UNIVERSIDAD
DE GRANADA

Programa de Doctorado en Tecnologías de la Información y la Comunicación

Algoritmos avanzados de procesamiento de señal basados en técnicas de Deep Learning para descripción y caracterización de señales sismo-volcánicas

MEMORIA PRESENTADA POR

Manuel Marcelino Titos Luzón

Julio de 2018

DIRECTORES

María del Carmen Benítez Ortúzar y María Luz García Martínez

Departamento de Teoría de la señal,
Telemática y Comunicaciones

Editor: Universidad de Granada. Tesis Doctorales
Autor: Manuel Marcelino Titos Luzón
ISBN: 978-84-1306-030-9
URI: http://hdl.handle.net/10481/54078

La memoria titulada "*Algoritmos avanzados de procesamiento de señal basados en técnicas de Deep Learning para descripción y caracterización de señales sismo-volcánicas* ", que presenta D. Manuel Marcelino Titos Luzón para optar al grado de doctor, ha sido realizada dentro del Programa Oficial de Doctorado en "*Tecnologías de la Información y la Comunicación*", en el Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada bajo la dirección de los doctores Dña. María del Carmen Benítez Ortúzar y Dña. María Luz García Martínez.

El doctorando y los directores de la tesis garantizamos, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo la dirección de los directores de la tesis, y hasta donde nuestro conocimiento alcanza, en la realización del trabajo se han respetado los derechos de otros autores a ser citados cuando se han utilizado sus resultados o publicaciones.

Granada, Julio de 2018

Los directores

Fdo: María del Carmen Benítez Ortúzar

Fdo: María Luz García Martínez

El doctorando

Fdo: Manuel Marcelino Titos Luzón

Esta tesis doctoral ha sido desarrollada con la financiación de una beca predoctoral FPI (código BES-2013-066730) adscrita al Ministerio de Ciencia, Innovación y Universidades. También ha sido subvencionada por los proyectos TEC-2012-31551 y TEC2015-68752 del Ministerio de Ciencia Innovación y Universidades.

Agradecimientos

En la esfera del reloj de pared del maestro Pío Baroja, yacía un lema estremecedor que señalaba el paso de las horas: Todas hieren, la última mata.

Muchas han sido las campanadas que han sonado en nuestro alma y nuestro corazón. Las dos manillas de ese reloj que ignora la marcha atrás, y hoy, con un pie en la mucha vida que hemos dejado atrás y el otro en la esperanza de la mucha que nos queda por vivir, escribo estas líneas para hablar con sentido del humor, y discurrir con buena voluntad y ya veremos si también con suerte, de todas aquellas personas que han hecho posible esta tesis doctoral.

En primer lugar y sabiendo que mis directoras Carmen y Luz no solo me permiten el inciso, sino que me obligan a hacerlo, me gustaría dedicar esta tesis de manera muy especial a mis abuelos Manuel y Mercedes, Marcelino y Estrella(que aunque hace tiempo que la vida nos separó, sé que desde dónde estéis, hoy, sonreiréis). Vuestro ejemplo diario, vuestra humildad, vuestra perseverancia, vuestra lucha y el sin fin de valores que desde niño me habéis inculcado me han hecho llegar a dónde estoy. Simplemente me queda deciros GRACIAS. Esta tesis es por y para vosotros/as.

Esta tesis tiene también una dedicación especial a mis padres y hermanos. Vosotros habéis sido protagonistas la etapa más especial e importante de mi vida, con sus altos y sus bajos. Habéis sufrido conmigo y por mi. A mis padre Manuel y Mercedes, por los ejemplos de perseverancia y constancia que los caracterizan y que me ha inculcado siempre, por el valor mostrado para salir adelante y por su amor. A mis hermanos Marcelino, Mercedes y Mónica, por enseñarme cuánto valgo y estar presente aún cuando no lo he notado. Por vuestra fe en mí. GRACIAS. Esta es vuestra recompensa.

Con el mismo cariño que a mi familia tengo que dedicar este trabajo a mi familia académica. Muy especialmente a mis Directoras Carmen Benítez y Luz García, por darme la oportunidad de vivir la que considero la mejor etapa de mi vida, por inculcarme los valores de rigurosidad y concisión que un trabajo científico debe tener, por vuestra enorme paciencia y dedicación, por el esfuerzo, y sobre todo, la confianza que habéis depositado en mí durante todos estos años. El camino no ha sido fácil y vuestra ayuda ha sido fundamental para alcanzar este objetivo con los que considero, excelentes resultados. GRACIAS.

Quisiera también agradecer el apoyo y la formación que durante todos estos años he recibido del que personalmente considero mi tercer Director, Jesús Ibáñez. Tus clases magistrales sobre sismología volcánica y tus consejos no solo me han abierto las puertas a este apasionante mundo, sino que me han dado la oportunidad de conocer lo que significa una campaña sísmica y llegar hasta la Antártida. GRACIAS.

Me gustaría agradecer muy especialmente al Departamento de Teoría de la señal, Telemática y Comunicaciones (que siempre consideraré mi casa) y al Instituto Andaluz de Geofísica. Al primero por acogerme y brindarme la oportunidad de vivir la etapa más especial de vida. Ni uno sol@ de los investigadores que lo componen merecen el descrédito que estos días tiene la Universidad pública. Ejemplo de trabajo, profesionalidad, dedicación, rigurosidad, disciplina y CIENCIA es lo que describen sus siglas. Ángel de la Torre, José Carlos, Isaac, Sonia, Pepe, Pedro, Gabriel, Juanma y muy especialmente a mi padrino Juanjo y más que un amigo Ángel Bueno, GRACIAS. Gracias por vuestros consejos, vuestra dedicación, vuestra humildad. Gracias. Al segundo, por inculcarme en vuestras campañas el valor de la humanidad, tan deteriorado

en estos días. Por inculcarme el valor del compañerismo elevado a la enésima potencia. Por hacer posible un sueño, la ANTÁRTIDA. Gracias Javi por confiar en mí a pesar de no ser geofísico y brindarme la oportunidad de cumplir un sueño. Gracias a mis padres antárticos Beni y Enrique, por vuestro bautismo polar y acogerme en vuestro seno como a un hijo. Gracias a mi hermano antártico Luís, del que no es necesario que diga nada, él sabe que no hay letras que describan lo vivido. Gracias a mi buen amigo Alfonso de "Arfacar" memorables fueron las noches en Randazzo. Agradecer también la confianza, la amabilidad y cercanía de dos hombres que aunque veo poco, siempre tienen una sonrisa para mi, Pepe Morales y Gerardo Alguacial. Y como no, gracias a mi buen amigo Paco Carrión, del que siempre aprendo una lección de vida. Las fragatas y las bergantinas siempre me acompañarán.

Evidentemente no puede faltar el agradecimiento a mis compañeros doctorandos. Ángel, Rafael, Sergio Ramírez y García ,Sara, Pablo, Daniel, Manuel Parras, Juanan, Rafa, Maillo, Elena, José Ángel, etc. por tantas y tantas experiencias, horas de charla, de temores, y también de alegrías. Os deseo lo mejor en estos años, sois el más fiel reflejo de valor de esta tierra. GRACIAS.

No podía faltar el agradecimiento a mi familia (en especial a mis primos) y amigos (el Laboratorio, Juanma, Miguel, Jose, Fede...), por ser más de lo que les pedí y de lo que en algunas ocasiones merecía. Por dar más de lo que necesité. Por brindarme confianza y apoyo antes de que lo notara e incluso pidiera. GRACIAS.

Quería dejar para el final, un agradecimiento muy especial, a dos personas. La primera de ellas es mi buen amigo Manuel Álvarez, Manolo para los amigos. Contigo conocí el mundo de la investigación y fuiste quién me incitó a realizar los estudios de doctorado que hoy termino. En aquel momento no disponías de financiación, y aunque querías trabajar conmigo no dudaste en recomendarme la posibilidad de trabajar con Carmen y Luz. La segunda la conocí en el Fin del Mundo. Es el Brigada Francisco Jarana. Ejemplo de amor por todo lo relacionado con Andalucía. Por acogerme como a un hermano y estar conmigo cuando las cosas se ponían grises oscuras. Por las innumerables experiencias que tuvimos la oportunidad de vivir juntos en las inhóspitas latitudes polares. Por todos los andaluces que hemos tenido la suerte de representar a nuestra tierra con orgullo dónde otr@s tantos hacían gala de la suya sin dejar de menospreciar la nuestra. Gracias Manolo. Gracias Paco!

Mi agradecimiento también a todas aquellas personas que no por no citarlas han sido menos importantes en el término de esta memoria.

Y como no, a mis mascotas!! Por todos las experiencias vividas, por crecer juntos, por hacerme reír, porque siempre tuvisteis un saludo amable para el que fue vuestro amigo, pero sobre todo por enseñarme el concepto de fidelidad y nobleza.

GRACIAS A TOD@S!!!!

Índice general

Índice de tablas

Índice de figuras

Parte I

Prefacio

Resumen y Objetivos

Summary

In highly populated areas where people are living near active volcanoes, volcano monitoring is an important task which helps to quantify the risk of potential eruptions. The myriad of seismic processes deep beneath the surface of volcanoes requires the development of early warning systems that can inform human societies about threats from volcanic hazards. Given the social and high economic impact generated by active volcanoes, there are several scientific disciplines to study volcanoes.

Based on the study and analysis of volcanic regions using seismic data, volcanic seismology describes and characterizes the wide range of signals related to the surface manifestation of complex (physical and chemical) processes occurred in the Earth's interior [46, 4, 106]. Even though each eruptive scenario has different characteristics (magma rheology, morphology of the volcanic structure, position and origin of the magmatic source) that derive on a large variety of different seismic signals, there is a remarkable observation: many volcanoes show comparable seismic signal characteristics that can be associated to different seismo-volcanic sources [50]. Therefore, one of the fundamental and most challenging objectives in volcano-seismology is to determine what always happens before an eruption in any scenario, which is differentiating between eruptions and that is only characteristic of some types of eruptive scenarios.

The objective of seismic analysis is not only limited to identify volcano seismic signals, but to determinate the source that originates them. Early warning systems are mainly based on the analysis of the considered prior precursor events. By definition, a seismic precursor is any seismic signal prior to potential eruption and whose origin is related to the dynamic processes (like fluids) that cause it. In this sense, the development of an automatic system able to evaluate potential seismic sources and their relationship to the present volcanic process in real time will allow a more effective management of volcanic risk.

Whilst direct applications of machine learning and signal processing could serve to improve monitoring and eruption forecasting, nature itself imposes several limitations that need to be taken into consideration. Hence, the implementation of automatic and robust monitoring systems is a difficult task:

1. The seismic signals are not entirely clear: seismic waves radiated by volcanic sources contain information not only about volcanic dynamic but also about the inner complex structure of the volcanic edifice affecting the seismic wavefield (source and attenuation effects), increasing the difficulty of geophysical interpretation from recorded signals [162, 151].

2. Generalization capabilities from machine learning models are inferred from ac-

quired data previously analyzed by geophysicists. However, data completeness plays an essential role when building volcano monitoring systems, as results will be strongly influenced by a human factor, including, but not limited to, lack of unified criteria, labelling fatigue, and time needed to analyze continuous streams of data. [24].

The vast amount of seismic data registered during an eruption requires robust and reliable systems able to operate in real time and tackle the mentioned drawbacks. The main objective of this thesis is to propose a deep learning volcano-seismic recognition system that can exploit the information contained within recorded seismic signals and improve generalization capabilities during real eruptive scenarios.

Inspired on recent advances in the field of neural networks and volcano-seismology [32, 27, 90], its main contributions are:

1. **Development of an automatic recognition system for isolated volcano-seismic events:** The monitoring of active volcanoes generates huge amounts of data that vulcanological observatories can hardly manage in the short term. Often, each recording is analyzed by geophysicists based on their experience and knowledge of a particular volcano. As a result, current datasets are composed by most relevant seismic signals that can be encountered in each volcano. Seismic observatories analyze the rate, type and location of seismic events. However, this wealth of data requires high recognition rates at a lower computational cost, with enough reliability and robustness to support the study. The approach followed in this work is to parse raw waveforms into a set of descriptive features using signal processing algorithms that can be further used in geological modelling frameworks. Deep learning could encode geophysical knowledge through hierarchical features that can enhance current early warning systems at scale.

2. **Development of a continuous recognition system:** The classification of isolated events, whilst useful to classify at scale, is still insufficient to manage eruptive crisis and issue early warnings in real time. During eruptive crisis, seismic data is registered as a continuous stream. Unlike isolated and segmented signals, these continuous seismic registers are temporal sequences with an indeterminate number of concatenated events. In this sense, the detection and classification of volcano-seismic events from real-time seismic data is a sequential problem which involves a complex and high dimensional dynamic signals which require efficient models able to capture the long temporal dependencies of seismic data.

According to the requirements of each system, several artificial neural networks architectures and several data parameterization schemes have been proposed. For the isolated recognition systems, we have found that deep architectures based on pre-training initialization and multiple processing layers to learn abstract representations of data could be a particularly effective strategy to increase the overall generalization capabilities of the systems. In this sense, we tested two different DNNs, DBN (Deep Belief Networks) and SDA (Stacked Denoising AutoEncoder) with seismic data recorded at "Volcán de Fuego", Colima (Mexico).

In the case of continuous recognition system, we find that RNNs (Recurrent Neural Networks) can be applied as statistical models able to exploit temporal information. We tested three recurrent architectures with seismic data from Deception Island (Antarctica) over different seismic periods: vanilla-RNN, LSTM and GRU. In order to

explore their generalization capabilities with data recorded in different time periods we further tested the models with data from a recent seismic survey in 2017.

The classification results obtained in both approaches outperform the recognition rate obtained by classical architectures as Support Vector Machine, Random Forest, Hidden Markov Models and Gaussian Mixture Models. We find that sDA and DBN can classify seismic events with higher precision and recall than classical architectures. Moreover, deep architectures are more sensitive to detect events that occur simultaneously in time, such as explosions and tremors. With regards to continuous recognition systems, attained results have shown that vanilla-RNN, LSTM and GRU classify volcano-seismic events with good accuracy, and memory cells (LSTM and GRU) enhance the detection of long-term signals.

In conclusion, classifiers based on deep neural networks can be deployed in real-environments to monitor the seismicity of restless volcanoes, and enhance current early warning systems. However, given the nature and size of volcanic snapshots (dataset), the use of raw volcanic events as training data results on non useful representations, and therefore, a direct application of state-of-the-art deep learning architectures is still a challenge.

Structure

The rest of this work is organized into three large blocks as follows:

- First, Part 1 serves as an introduction to the classification of volcano-seismic signals and describes the theoretical framework of the proposed architectures. Chapter 1 provides the fundamental concepts of volcano-seismology and active volcano monitoring. Chapter 2 addresses the development of a recognition system based on supervised learning and introduces the related research in the field. Finally, Chapters 3 and 4 provide the theoretical background of deep neural networks architectures (DNNs and RNNs), describing and identifying the main drawbacks when deploying these architectures.

- Part 2 describes the experimental methodology followed in this work, with detailed results. Chapter 5 stands as a small introduction about the volcanic signals that compose our data sets. Chapter 6 motivates the need to parameterize the data, and how to exploit the capabilities of deep learning architectures in order to extract features from our datasets. Taking into account that that systems performance is highly dependent on how accurately parameters of the model can be estimated, we propose two different parameterization schemes. Chapter 7 proposes a novel approach in the field of volcano seismology to classify volcano-seismic events based on fully-connected Deep Neural Networks (DNNs). Two DNN architectures with different weights initialization are studied: stacked Denoising Autoencoders (SDA) and Deep Belief Networks (DBN). Using a combined feature vector of Linear Prediction Coefficients (LPC) and statistical properties, we evaluate classification performance on seven different classes of isolated seismic events. The results obtained are compared to well-established techniques as Multilayer Perceptron (MLP), Support Vector Machine (SVM) Random Forest (RF), Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM). Following this methodology, Chapter 8 proposes RNNs ((LSTM, GRU and Vanilla) for detection and classification of continuous sequences of volcano-seismic events at Deception Island Volcano, Antarctica. The best configuration trained with data from seismic records obtained from 1995 to 2002, will be tested with

data from recent seismic survey performed at Deception Island Volcano in 2017 by the XXX Spanish Antarctic scientific expedition. This experiment explores how RNNs can perform continuous monitoring of volcanic-activity when terrain and seismic sources changes.

- Finally Part 3, discusses the interpretations of the final results and motivates future research lines in order to enhance effectiveness of the proposed systems.

Objetivos

Teniendo presente el éxito que las estrategias de aprendizaje profundo están obteniendo en muchas áreas del conocimiento en problemas de muy diferente naturaleza y complejidad, en este trabajo, se planteó el estudio y el análisis de dichas técnicas con el objetivo de desarrollar sistemas automáticos de reconocimiento de señales sismo-volcánicas rápidos y eficientes.

El desarrollo de un sistema automático de reconocimiento en el ámbito de la sismología volcánica puede ser abordado desde dos perspectivas, las cuales definen cada uno de los objetivos principales de esta tesis y que a continuación detallamos de forma esquematizada:

- **Clasificación en aislado de eventos sismo-volcánicos**

  - **Motivación de la necesidad de un proceso de parametrización o extracción de características con el que describir los eventos sismo-volcánicos** previo al proceso de clasificación, con el que guiar el aprendizaje y ajuste de los modelos. Teniendo presente las particularidades de las señales sismo-volcánicas, se explorará la viabilidad de usar como vectores de características, los obtenidos por algunas arquitecturas específicas presentes en el estado del arte. A partir de este estudio se motivará la necesidad de un proceso de parametrización, basado en técnicas de procesado de señal, con el que describir y caracterizar las señales sismo-volcánicas.
  - **Estudio del estado del arte de los sistemas de clasificación automática en el área de la sismología volcánica** con el objetivo de descubrir debilidades que nos sirvan como base para definir los objetivos de diseño de nuestra propuesta.
  - **Estudio de los diferentes esquemas de parametrización empleados en la clasificación de eventos sismo-volcánicos** con el objetivo de modelar la información realmente relevante de los eventos. Se realizará un análisis de las principales técnicas de descripción de datos, con especial énfasis en las características diseñadas para modelar eventos sismo-volcánicos. La mejor parametrización encontrada se propondrá como esquema de construcción del vector de características de entrada a los modelos.
  - **Análisis, diseño, implementación y evaluación de un sistema de clasificación basado en el paradigma del aprendizaje profundo.** Tras el estudio de los diferentes esquemas de parametrización empleados en la clasificación de eventos sismo-volcánicos, nuestro objetivo es desarrollar un sistema que de forma simple y precisa, intente mejorar los sistemas de clasificación hasta ahora presentes. Con el objetivo de evaluar la escalabilidad de nuestras propuestas, los mejores modelos encontrados serán evaluados frente a los mejores modelos presentes en el estado del arte.

- **Detección y clasificación en continuo de eventos sismo-volcánicos:**

  - **Motivación de la necesidad de un proceso de parametrización o extracción de características con el que describir los eventos sismo-volcánicos** previo al proceso de clasificación, con el que guiar el aprendizaje y ajuste de los modelos. Teniendo presente las particularidades de las señales sismo-volcánicas, se explorará la viabilidad de usar como vectores de

características, los obtenidos por algunas arquitecturas específicas presentes en el estado del arte. A partir de este estudio se motivará la necesidad de un proceso de parametrización, basado en técnicas de procesado de señal, con el que describir y caracterizar las señales sismo-volcánicas.

- **Estudio del estado del arte de los sistemas de clasificación automática en el área de la sismología volcánica** con el objetivo de descubrir debilidades que nos sirvan como base para definir los objetivos de diseño de nuestra propuesta.
- **Estudio de los diferentes esquemas de parametrización empleados en la clasificación de eventos sismo-volcánicos** con el objetivo de modelar la información realmente relevante de los eventos. Se realizará un análisis de las principales técnicas de descripción de datos, con especial énfasis en las características diseñadas para modelar eventos sismo-volcánicos. La mejor parametrización encontrada se propondrá como esquema de construcción del vector de características de entrada a los modelos.
- **Análisis, diseño, implementación y evaluación de un sistema de clasificación basado en el paradigma del aprendizaje profundo.** Tras el estudio de los diferentes esquemas de parametrización empleados en la clasificación de eventos sismo-volcánicos, nuestro objetivo es desarrollar un sistema que de forma simple y precisa, intente mejorar los sistemas de clasificación hasta ahora presentes. Con el objetivo de evaluar la escalabilidad de nuestras propuestas, los mejores modelos encontrados serán evaluados frente a los mejores modelos presentes en el estado del arte. No obstante, el análisis, diseño, implementación y evaluación de este tipo de sistemas deberá ser adaptado a escenarios de uso en tiempo real o cuasi-real.

Parte II

Introduction to automatic recognition systems for volcano-seismic signals

Capítulo 1

Volcanic activity

This chapter introduces the general concepts of volcanic seismology and active volcano monitoring. Section 1.1 motivates the dangers associated with active volcanism on human society and the importance of monitoring tasks. Section 1.2 presents volcanic seismology as one of the most reliable approaches for the strategies for volcanic control and it describes from a geophysical point of view, the seismic signals most frequently registered during an eruptive process. Section 1.3 motivates source and attenuation effects as the main phenomena conditioning the waveform and spectral content of a seismograms, revealing the complexity of the automatic classification process. Finally, section 1.4 describes the main strategies used in volcano monitoring and it introduces signal processing as the cornerstone of any early warning system.

1.1. Introduction

Coupled with earthquakes and meteorological disasters, volcanic eruptions and, to a lesser extent, volcanic activity, are the most severe natural hazards, affecting the climate and being an important contributing factor in global environmental change, as global warming [206]. Almost 500 million people around the world live with active volcanoes [152]. Thus, given the high social and economic impact (see in Table 1), generated by active volcanism on human society, the monitoring of volcanoes and the development of efficient early warning systems are vitally required tools to advise governments and inform the society about threats from natural hazards.

Based on the pioneering studies conducted in the first half of the 20th century, by [156, 179, 110], volcanic seismology has been emerging, becoming one of the basic pillars for prediction and forecasting volcanic risk. Volcanic activity and its interactions with the environment give rise to a range of physical processes (as fracturing rocks, escape of pressurized gases or ground deformation) that can be monitored. As a result, seismographs can register a wide range of volcano-seismic signals that reflect the nature and underlying physics of the source process. By analyzing these seismic-events, we can identify the active sources of emission, and thus, improve our knowledge about the state of the volcano.

Hazard	*Threat to life*	*Threat to property*	*Areas affected*
ash and pumice fall	*low (except for aviations)*	*depends on thickness*	*L,R,N,I*
pyroclastic flows	*very high*	*very high*	*L,R*
lava flows	*low*	*very high*	*L*
lahars/flooding	*high to moderate*	*high*	*L,R*
gases/dust/acid rain	*low to moderate*	*moderate*	*L,R*

Tabla 1.1: Summary of the effects and extents of major volcanic hazards [191]. L=Local; R=Regional; N=National; I=International.

1.2. Volcanic seismology

Between the late of 1960s and early 1970s, advances in seismic sensors and storage systems provided more accurate data from internal volcano sources, allowing volcanologists to focus their attention on the seismic signals preceding or accompanying a volcanic eruption. After this avid advance, many attempts were made trying to associate the recorded signal with potential source mechanisms, but the intrinsic difficulty of these signals, containing information not only of the volcanic dynamic but also about the inner complex structure of the volcanic edifice, makes many of them remain an unknown still to these days. A little later, between the late of 1980s and early 1990s, new advances in seismic sensors improving both, portability and battery life, allowed the development of new monitoring techniques, which in turn meant an improvement in the study of source mechanisms [207] that had come to a standstill a few decades ago.

In that regard, Volcanic Seismology is one of the most reliable approaches for the strategies for volcanic control. Seismic signals reflect directly the exchange of elastic energy between volcanic processes and environment. Volcano-seismic signals are the result of stress and relaxation processes and/or an exchange of energy due to pressure changes or movement of the magma, fluids, etc, that are transmitted from depth to the surface through the mechanical properties of the crust [48].

The study and analysis of a volcanic region using seismic data requires to detect and to register the signals from volcanic activity, to subsequently carry out the identification and classification considering many aspects such as the parameters of the source (position, spatio-temporal evolution, energy quantification, etc) and the medium (structure of speed, attenuation, heterogeneities, etc) [107]. Therefore, the description and characterization of the different types of signals becomes a vital activity in order to know what is happening inside the volcano and it allows us to know different aspects of the volcanic system, as the dynamics or mechanisms of fluid transport and the effects derived from these dynamics. Often, the description and characterization of the different types of volcano-seismic signals are based on two main approaches:

- The waveform and spectral content, having as principal drawback the high variability of the signals, mainly related to the type of volcano, added to, the propagation and source effects (section1.3).

- The source mechanisms that generate the signals, based on a solid physical criterion which often requires unavailable specific information as velocity models, source location or magnitude.

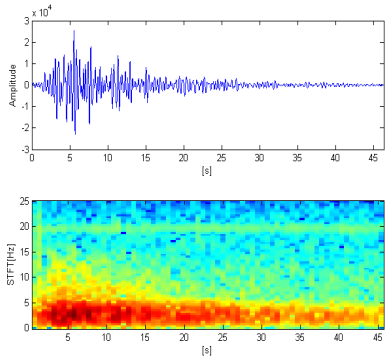
1.2.1. Volcano-seismic events and source models

Volcanoes are the surface manifestation of dynamic and complex processes occurred in the Earth's interior coupling physical and chemical processes. Due to the complexity of these processes, a large variety of different seismic signals can be recorded in these environments [46, 4, 106], but the large number of different terms for classifying these signals and the still lack of criteria commonly agreed about the source mechanisms which cause them are some of the problems to describe correctly the volcano seismology [143]. Despite of this variety, there is a remarkable observation: many volcanoes show comparable seismic signal characteristics that can be associated to different volcano-seismic sources [50]. Therefore, seismic signals are often classified into event families that could help to evaluate potential seismic sources and their relationship to the present volcanic process [108, 133].

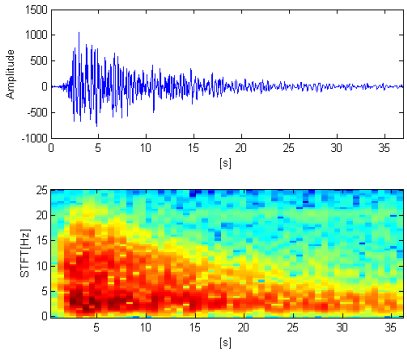
Following [187, 145, 143, 48, 7], we will try to classify the volcano-seismic signals based on the waveform, the presence or absence of identifiable phases, the spectral content and source mechanism. However, the shape and the spectral content of the signals can be significantly altered by the propagation medium 1.3.

- **Long Period Events (LPE), Figure 1.2.1a :** In general they are quasi-monochromatic signals with a narrow frequency band centered, in the majority of the cases, between 1 to 6 Hz. Their source models are associated to volumetric modes of deformation of the propagation medium. In general the proposed models are related to resonance of the medium as a consequence of fluid displacement inside of the volcanic edifice or the generation of pressure transients in fluids. We can mention as example, a crack in which a resonance occurs when the fluids (magma, gas or water) are ascending towards the surface or the existence of pressure transients within the fluid-gas mixture inside of the volcanic edifice, causing also resonance phenomena [49]. All proposed models are able to explain the behavior of the observed features in time and spectral domains. They are usually located in particular areas of the volcanic structure where fluids generate disturbances. They have been used as short-term precursors of volcanic eruptions. In many cases, they appear in time forming the so call "seismic swarms": thousands of LPE events in a short time period, often overlapped. these seismic signals remain in duration to small VTE earthquakes but with different frequency contain showing a clear harmonic signature [49]

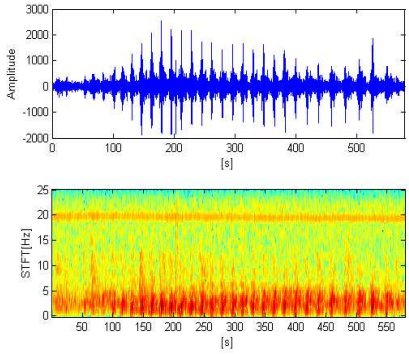
- **Volcano Tectonic Earthquakes (VTE), Figure 1.2.1b :** VTE events are *classical earthquakes* originated inside of volcanic environments. Their main characteristic is a signal with a broad frequency contains reaching up to 40 Hz with duration from a few to tens seconds. They are the result of a brittle response of the medium caused by seismic stress producing a shear failure of the volcanic edifice generating a broad set of seismic waves. This seismic stress could be produced by several causes, from local tectonic regime to fluid (water, gas or magma) displacement inside of the volcanic edifice. The consequence of this fracturing of the medium is the generation of two kinds of seismic waves (body waves) with different propagation velocity: P-waves (longitudinal displacement) associated to change of Pressure in the medium, and S-waves (transverse motion) associated to shear displacement of the elastic medium. They can appear spread in space and time inside of the volcanic edifice. they have been used as long-term precursors of the volcanic activity, appearing from days to months or years before the eruption. Often, the VTE events are categorized into two



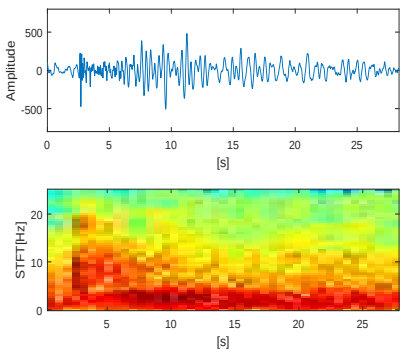
(a) Long Period Event (LPE)



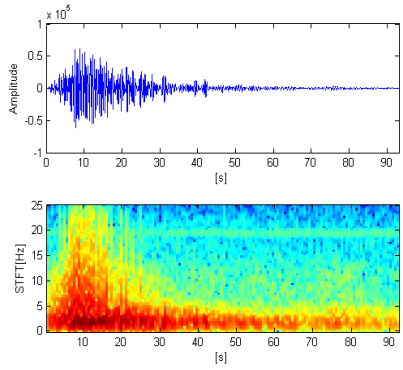
(b) Volcano-Tectonic Earthquake (VTE)



(c) Tremor (TRE)

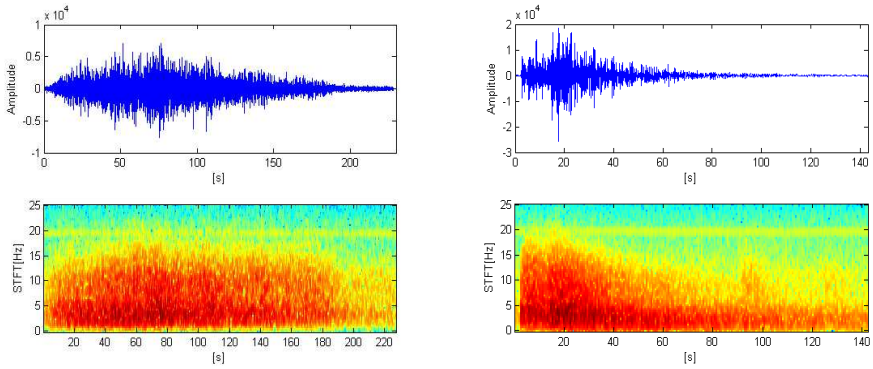


(d) Hybrid Event (HYB)



(e) Explosion (EXP)

Figura 1.2.1: Seismogram and Spectrogram of volcano-seismic signals registered at Volcán de Fuego", Colima (Mexico) and Deception Island, Antartica.



(f) Lava Flow (COL) (g) Regional Earthquake (REG)

Figura 1.2.1: Seismogram and Spectrogram of volcano-seismic signals registered at Volcán de Fuego", Colima (Mexico) and Deception Island, Antartica

different types:

- **VTE deep (VTE-A) :** commonly characterized by being located at depths greater than 2 km and by having clear onsets of P and S wave arrivals. The spectral content rises above 10 Hz, so impulsivity is very prominent.
- **VTE shallow (VTE-B) :** located at depths between 1 and 2 km. Mainly characterized by having spectral content reachs up to 10 Hz. The onset of P wave arrival is not impulsive, so they are considered emergent events. As a direct consequence of this lack of impulsivity, to determine the onset of S wave arrival S become a challenge.

▪ **Volcanic tremor (TRE), Figure 1.2.1c :** These events are in general characterized by harmonic signals with sustained amplitude and highly variable duration, lasting from minutes to hours or even months. Their spectral characteristics resemble LPE events with quasi-monochromatic signature, but in some cases their peak of frequency could reach up 10 Hz or more. Sources of volcanic tremor are diverse, from inner pressure disturbance to external gas emission, debris avalanches or pyroclastic flows, among others. Some theories suggest that, when the source of the tremor is located inside the volcanic edifice, its source is identical to those associated with LPE events, being the tremor the consequence of a non-lineal overlapping of multiple LPE events. This overlapping model is the consequence of the observation of the similar spectral characteristics, and that both TRE and LPE events share, in many cases, space inside of the volcanic edifice, appearing associated in the time. Volcanic tremors can be categorized following viscosity regimes and spectral contens into three types:

- **Harmonic Tremor (TRE):** monochromatic wave trains governed by several dominant frequencies (resonants) between 1 and 5 Hz, could vary the number of resonants within the same event. Although the source model associated with this type of volcanic tremor is not entirely clear, some authors such as [194] suggests a decompression on seismic parameter profiles in a gas-charged magma. However, the most accepted source for many authors

is the one suggested by [132], where the harmonic tremor is motivated as fluctuations of hidrotermal systems due to changes in both, pressure and temperature in themselves. The associated viscosity regime following [207] is high.

- **Spacmodic Tremor (TRS):** mainly characterized by relative variations in both, amplitude and frequency contents, this type of tremor is commonly associated to swarms of VTEs. As direct cosecuence of a less monochromatic signature, the spectral content is more sparse (between 1 and 10 Hz), often observing a single dominant frequency within this range. The associated viscosity regime following [207] is low.
- **Short-Pulsant Tremor (SPT):** a special type of tremor associated with the superposition of both, a trains waves (forming temporal *pulses*) and a spasmodic tremor (TRS). Similarly to TRS, the spectral content is more sparse and a single dominant frequency is often observed.

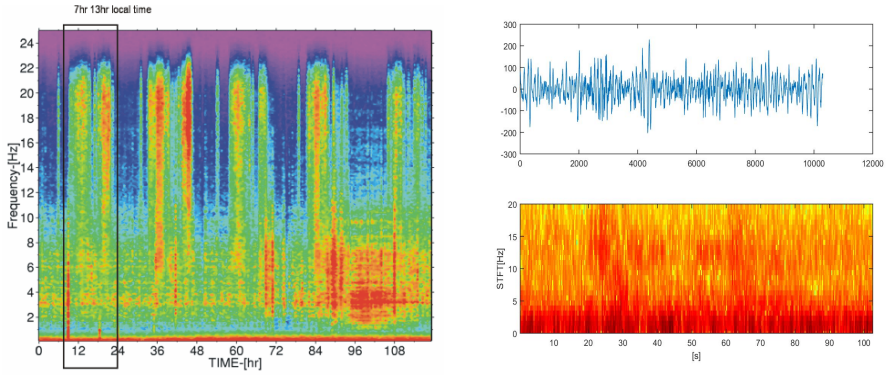
- **Hybrid events (HYB), Figure 1.2.1d :** They are often associated with the conjunction of events of different nature. They have a beginning characterized by the arrival of high frequency signals with a wide spectral band (up to 10 Hz) where the P and S phases can be easily identified. After the first arrival, a second signal of similar characteristics to LPE event is usually recorded. The origin or source model of these kind of events can be explained by the increase of pressure that it leads to the rupture of an area, producing an earthquake. The pressure-induced fracture is filled with volcanic fluids following the LPE event source previously mentioned, generating the signal of low frequencies registered. Then, the hybrid events usually are related to imminent pre-eruptive episodes.

- **Explosions (EXP), Figure 1.2.1e :** These signals are associated to the external activity of the volcano due to the sudden emission of gas and ash to the atmosphere (explosion). Since mostly of them can be visible and recorded by video, it is possible to associate the external effect to the signals recorded in the seismometers. They are characterized by an initial short duration LPE event followed by high frequency signals with a narrow energy peak with peaks located at different frequencies, from 4 up to 20 Hz.

- **Lava Flow (COL), Figure 1.2.1f :** As mentioned above, a class of tremor is originated by debris flow located at the volcano surface. Since they can be monitored using video record, it is possible to associate the surface lava movement with the generation of this type of volcanic tremor. These events provide very useful information on the final consequence of the internal activity in a volcano. These events exhibit frequency content between 5 to 10 Hz.

- **Regional Earthquakes (REG), Figure 1.2.1g :** Tectonic earthquakes might occur anywhere in the earth if there is enough elastic strain energy stored to drive the fracture propagation along a fault plane. They normally have a bigger duration than volcano-tectonic earthquakes, but similar spectral content.

- **Environmental Noise (NOISE), Figure 1.2.2 :** Overlapped over any seismic signal, there is a type of signal, mainly of low amplitude, originated by multiple natural and artificial sources. This signal is named *seismic noise* and typically contaminates the registered seismic signals. As natural sources, we can mention wind, atmospheric pressure variation or rain (Figure 1.2.2b). In case of artificial



(a) Cultural noise [207]. (b) Environmental noise (wind)

Figura 1.2.2: Environmental noise recorded in different seismic stations. a) Cultural noise recorded at Mt. Merapi (Indonesia). As the station is located near to farming area, human activity can be clearly recognized in cycles of 24 h. b) Ambiental noise recorded at Deception Island (Antarctica) during a windy day.

sources, this noise is known as *cultural noise* and is mainly introduced by nearby populations and human activity (Figure 1.2.2a). In some cases, this noise could interfere the frequency range in which most of the volcanic spectral content is located. Considering the locality of these type of events, efficient methods based on simultaneous occurrence in different nearby stations can be used to isolate them from those originated by volcanic sources.

1.3. Problems related to Volcanic Seismology: Source and Attenuation Effects

The seismic waves contain information not only of the volcanic dynamic but also about the inner complex structure of the volcanic edifice affecting the seismic wave-field and its interpretation [162, 151]. In most volcanoes a pronounced and rough topography introduces new complex effects, such as interferences, severe attenuation effects, or changes in the path followed by the direct seismic waves [211]. As final result, the same original seismic signal is recorded with different shape and wave-field characteristics according to the site of the seismometer. In addition, at the same seismic station, similar seismic-sources generate different signal pattern according to the way in which the source radiates energy [46, 47]. All of these effects can be mostly divided into path effects (attenuation) and source effects (energy and radiation pattern).

1.3.1. Attenuation Effects

The main phenomenon conditioning the spectral content and shape of a seismogram is the seismic attenuation [167, 65]. Seismic attenuation is the contribution of both, the energy lost by in-elasticity (intrinsic attenuation) and the energy lost by dissipation (scattering attenuation). The effect of the attenuation is a visible loss of

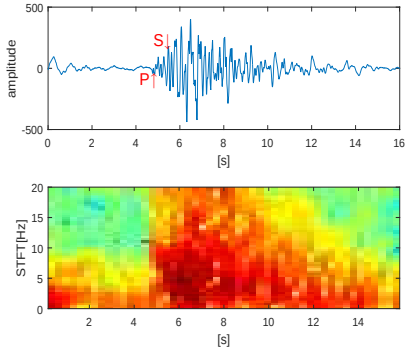
energy being more effective at higher frequencies and directly dependent of the distance receiver-source. In volcanic environments where the complexity and heterogeneity of the medium is more pronounced, this effect strongly modifies the seismic waveform, producing several phenomena such as: arrivals of scattered seismic waves in the last part of the seismograms [64]; reducing the high frequency energy contribution in the seismograms [139]; changing the magnitude and laws of scale [95]; or introducing distortion in the spectra of the LPE events [160], among others. These effects increase the difficulty to discriminate the type of event.

Figure 1.3.1 illustrates how the attenuation effect can introduce a bias in the recognition patter of a Volcano-Tectonic earthquake (VTE). Three VTE events, recorded at the same seismic station, but with different hypocentral distance (source-receiver distance) are illustrated. Figure 1.3.1a depicts a VTE with S-P time of around 1 second (a received-source distance lower than 3-4 km). The spectral contains reaches up to 20 Hz, the upper limit of the used scale, composing a "classical" VTE seismogram. Figure 1.3.1b depicts a VTE event was recorded with S-P time of around 2 seconds (distance close to 8 km). The spectral shape is different and the high frequency content does not exceed 15 Hz. Finally, in Figure 1.3.1c, the S-P time is close to 3 seconds (around 12 km of distance). This distance is not too large for the size of the volcanic environment, and many VTE events can be recorded at highest distances. However, observing the spectrogram, it can be appreciated the loss of the high frequency content, and the main presence of frequencies lower than 6 Hz. In this case, the VTE appears with clear similitude to LPE events, being easily confused.

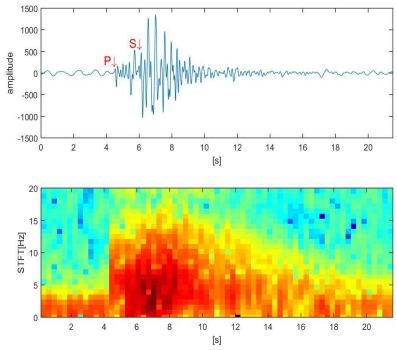
Another important aspect related to attenuation effects can be observed by the peak to peak amplitude degradation of the recorded signal. Figure 1.3.2 depicts two examples of volcanic tremor (TRE) recorded at the same seismic station (at same scale for visualization purposes). Whilst the frequency pattern shown in their spectrogram suggest they have been generated by the same source mechanism, the noticeable differences in their energy level indicates strong attenuation effects, coming to be confused the less energetic signal (almost 10 times lower than the expected level for an evident volcanic tremor) with environmental noise (NOISE).

1.3.2. Source Effects

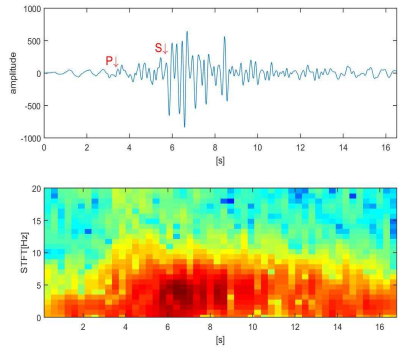
The source of volcano-seismic signals is associated to the interaction of geophysical systems and it is highly dependent on the volcanic environment. Source effects can be related to the interaction of water and hot rocks, among many others factors. For example, at Deception island, evidences of aquifers and hot materials placed near the surface widely confirmed using seismic tomographies in velocity and attenuation [171]. Interactions between water and hot rocks generate a sudden change of phase at depth, with its associated pressure step and radiation of high frequency seismic waves. In addition, the presence of several and complex fault systems in the area [7] induces low-frequency seismic waves swarms as the result of fluid auto oscillations filling the crack. When the interaction between water and hot-rocks is simple, simple oscillations are registered. However, in case of multiple interactions, the continuous change of phase and resonance of the faults generates an overlapping of signals.



(a) VTE non-attenuated

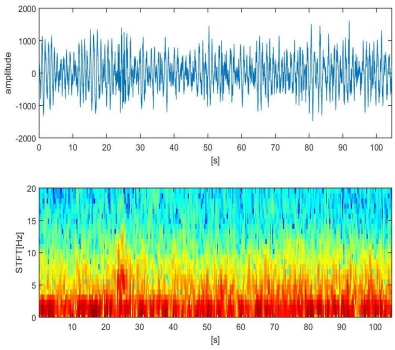


(b) VTE attenuated

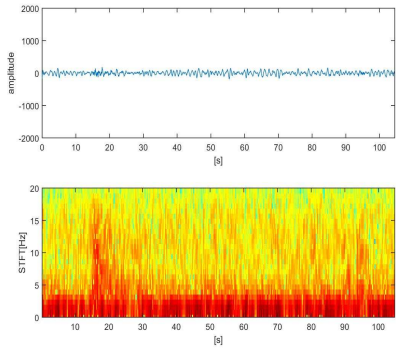


(c) VTE very attenuated

Figura 1.3.1: Attenuation effects. Seismograms and spectrograms showing how seismometer location affects registered shape and wave-field characteristics of volcano-seismic signals.



(a) Energetic tremor



(b) Attenuated tremor

Figura 1.3.2: Attenuation effects. Seismograms and spectrograms showing how two different volcanic tremors, with similar frequency patterns in their source mechanism as shown in their spectrogram, but with evident differences in the energy level due to attenuation effects.

1.4. Monitoring Volcanoes

Volcanic activity, as natural hazard, has many features in common with extreme weather or earthquakes. Generally, they are complex phenomena described by multi-parameter physical processes. As complex nonlinear systems, many of the processes that govern them are variable and change suddenly, making them unstable and unpredictable. In addition, the amount of energy released in a volcanic eruption (close to that released by several tens of thousands of atomic bombs), as well as the effects and extents of major volcanic hazards above-mentioned, make neccesary the development of new techniques for prediction and forecasting of volcanic risk.

1.4.1. Design of a seismic network

As mentioned above, the use of volcano-seismic signals as monitoring tool is justified because all eruptive processes are preceded or accompanied by anomalous seismic activity [207]. In order to acquire and store the information related to these anomalies (characterized by the registration of a wide variety of events), the observatories need to determine what instrumentation and what places are suitable for observation. The stations distribution is extremely important, it being necessary to choose places that potentially minimize the source and attenuation effects (section1.3). The number of stations deployed is another important issue. In most cases, given the cost of components and the maintenance, in terms of economic and human resources during activity stages or extreme weather conditions, between four and six stations are distributed around the volcanoes. However, there are cases on which it is necessary to deploy seismic antennas (arrays), since the application of these techniques has as main advantages the improvement in evaluating the radiated wavefield properties, velocity structure and the source location [207, 48]. The spatial coverage approximations most used in volcanic seismology are:

- **Two scales network:** spatial coverage can be considered as volcanic and non volcanic regions. Therefore, in order to distinguish between volcano and regional or local seismicity, two networks are deployed: one large scale network extending into non volcanic regions ($\triangle <$ 20 km) and other one concentrated on the flanks and on the top of the volcano ($\triangle \sim$ 0-2 km). The large scale networks are widely used to localize deep-seated sources of magmatic activity and tectonic seismicity. In contrast, given that most of the volcano-seismic signals are shallow, small in amplitude and very rich in terms of spectral content, a small scale network is also used.

- **Seismic antennas (arrays):** volcano-seismic signals are closely linked to several constraints as superficial ground velocity structures or sources location. Thus, in order to build maps of radiated wavefield to evaluate the properties of these constraints, seismic arrays composed by several seismic stations (between 6 and 12) are deployed in a diameter area of 100 m ($\triangle <$ 100 m) [9, 8].

- **Network of seismic arrays:** a special use case of seismic antennas is in the form of a network. Each station distributed into volcanic area, is a small seismic array composed by a broadband seismometer coupled to several vertical short-period sensors. This approach compared to singular seismic antennas improves both, sources location and wavefield representation.

However, when the observatories do not have enough resources, only one seismic station is deployed. This approach has been used since 1960s, obtaining excellent results and being the basis of many volcano-seismic recognition systems nowadays when polarization methods, ground motion studies, seismic spectral amplitude measurement (SSAM) and real-time amplitude measurement techniques are used (RSAM) [174, 67].

Another important factor related to the quality of the observations, is the type of sensor distributed. Despite the significant improvements in seismic instrumentation, the quality of the seismic signals registered, directly reflected in the seismogram (motion, velocity and acceleration of the ground), are influenced by the type of sensor used. Since a seismogram is the convolution of seismic signals (affected by source and path site) and the impulse response of the sensor, the same seismic event, registered in the same place, results in a different seismogram while using different seismometers. Although the wide range of seismometers, a simple description can be made in function of frequency response of the instrument, working in a specific frequencies interval $[f_L, f_H]$:

- **Narrow band or short-period sensor:** characterized by lower cutoff frequency greater than or equal to 1 Hz ($f_L \geq$ 1 Hz), this kind of sensor digitizes data using sampling frequency of 50 Hz ($F_S \geq$ 50 Hz) and between 12 and 16 bits per sample, so data are stored using integer type.
- **Broadband or long-period sensor:** mainly distinguished by a greater dynamic range, lower cutoff frequency greater than or equal to 0.01 Hz ($f_L \geq$ 0,01 Hz) and higher sampling frequency ($F_S \geq$ 200 Hz), this kind of sensors are being widely used. Unlike narrowband sensors, in their digitization process, they can use between 20 and 32 bits per sample, so data can be stored using both, double and integer types.

1.4.2. Signal processing as the cornerstone of any early warning system

Coupled with signal processing methods, volcanic seismology has been applied in several volcanoes as cornerstone of early warning systems for volcanic risk [13, 51, 107, 74]. When a volcano shows signs of activity, vulcanologist need to provide information on hazardous volcanic phenomena and their effects. As the requirements for prediction are rigorous, multidisciplinary works, involving geophysics, signal processing, artificial intelligent, between others, are carried out to ensure the safety of society. Some of the most important methods currently used in real scenarios to predict and forecast volcanic eruption are based on this transversality:

- **Count of precursor events:** the variation of volcanic activity can be measured counting the number of precursor events. Thus, inverting the acceleration curve in the frequency of this measurement, the date of the eruptive process can be estimated [205]. However, as the number of precursor events during a pre-eruptive period can be really high, vulcanologist rely on artificial intelligence and signal processing methods to detect, classify and finally count them. These multidisciplinary collaborations have become the state of the art in volcano-seismic recognition systems (VSR), being applied in active volcanoes with satisfactory results[27, 157].
- **Automatic seismic wave arrival detection and picking:** automatic detection and precise picking of the arrival times of seismic waves are challenge

and tedious task for earthquake early detection systems. So, based mainly on signal processing methods, automatic wave arrival detection systems are every important tools desirable by all seismic observatories [11, 77, 215].

- **Characterization of long-period events:** according to [49], long-period volcano seismicity can be used in eruption forecasting, since these events are present in both, pre-eruptives and eruptive episodes. Therefore, the study and characterization of long period events as well as the seismicity or source mechanism associated with them, are the most used approaches to monitoring active volcanoes. Many efforts in signal processing and artificial intelligence fields have been made over the lasts years to create automatic systems able to extract representative features of the signals that allow an effective detection and characterization of these signals

- **Description and location of magma:** following [177], the material transported in an eruptive process coupled with the source mechanisms which produce it, are reflected in signals with very large period. Therefore, detect efficiently these signals and then invert their moment tensor, is a interesting task, since several features involved in the eruption (characterization of microseismic event failures, identify which fractures are contributing to transport material, advanced understanding of fracture propagation) and consequently with the magma (estimate fluid flow enhancement, understand stress-strain field and fracture orientations), can be estimated.

- **High resolution 3D tomography:** in order to approximate the internal structure of the volcano and check the existence of fluids in cracks and magmatic chambers, velocity maps of P and S wave arrivals are developed. Similarly to count of precursor events and characterization of long period events, several signal processing techniques are used to detect these arrivals [77, 78].

- **Automatic recognition systems of volcano-seismic signals:** given the large amount of data (signals) acquired over time, a historical study of the behavior of the volcano is an advance in the knowledge of its dynamics in light of new eruptive stages. So, vulcanologists, based on their experience and knowledge of the volcano, create representative databases composed by several volcano-seismic signals of different nature. After that, each signals is characterized by signal processing methods, resulting on a set of descriptive features, which later, will be used to train intelligent automatic classification systems. Once the intelligent models has been trained, historical and recent records are analyzed, extracting new information that human experts had skipped.

Capítulo 2

Reconocimiento autómatico de señales sismo-volcánicas

El reconocimiento automático de señales sismo-volcánicas, desde el punto de vista del aprendizaje automático, puede ser abordado como un problema más de clasificación. Dada la gran variedad de técnicas existentes con las que abordar este tipo de problemas y dado que muchas de ellas han sido ya aplicadas en el área de la sismología volcánica, en este capítulo, daremos una visión general de los fundamentos de la clasificación automática centrándonos en el reconocimiento de señales sismo-volcánicas..

En la sección 2.1 describiremos los conceptos generales en el ámbito del aprendizaje automático y el reconocimiento de patrones. En la sección 2.2 estudiaremos las particularidades asociadas a los problemas de detección y clasificación de las señales mediante un enfoque supervisado. Para ello, describiremos el proceso de construcción de un sistema de reconocimiento desde la adquisición de los datos hasta la evaluación final, motivando en cada etapa los problemas asociados más relevantes.

Finalmente en la sección 2.3, haremos una revisión de los trabajos de investigación más relevantes en el área, haciendo especial hincapié en aquellos que nos servirán como base comparativa de los modelos que se proponen como base de esta tesis y que detallaremos en los siguientes capítulos.

2.1. Introducción

La búsqueda de patrones o regularidades capaces de describir y analizar un conjunto de datos o medidas (generalización del comportamiento a partir del análisis), es un problema fundamental y recurrente en la Ciencia a lo largo de la historia. Uno de los ejemplos más representativos de este problema fueron las extensas observaciones astronómicas realizadas por Tyge Ottesen Brahe (Tycho Brahe) en el siglo XVI, las cuáles le permitieron a Johannes Kepler descubrir las leyes empíricas del movimiento planetario, lo que a su vez supuso un avance importante en el desarrollo de la mecánica clásica. El reconocimiento automático de patrones, es por tanto, la búsqueda de regularidades en un conjunto de datos mediante reglas, algoritmos o modelos computacionales que permiten a un sistema realizar acciones de forma inteligente, como tomar decisiones o hacer predicciones ante nuevos escenarios o nuevos datos del problema [24].

Un sistema de reconocimiento automático de patrones (Pattern Recognition System), es un modelo adaptativo descrito por un conjunto de parámetros (θ) [24]. Generalmente, estos parámetros se ajustan a los datos del problema (training set) durante la fase de aprendizaje o entrenamiento (learning or training phase). En este sentido, un sistema entrenado puede ser expresado por una función $Y(x)$ definida por el conjunto de parámetros (θ) que mejor se adaptan a la forma analítica de los datos. Atendiendo a la forma en que los sistemas ajustan sus parámetros y a la forma de los datos que describen el problema, se pueden distinguir varios tipos de aprendizajes (Figura 2.1.1):

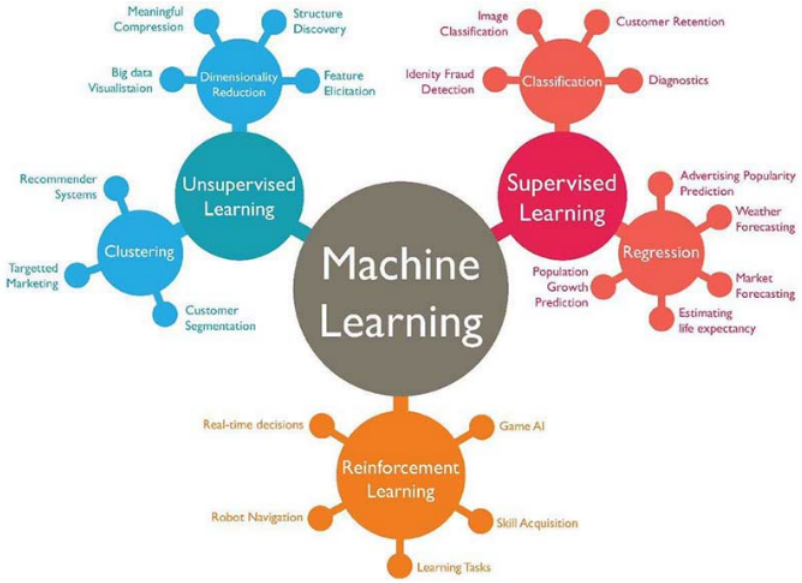


Figura 2.1.1: Tipos de aprendizaje automático atendiendo a la forma en que los modelos extraen el conocimiento. Imagen obtenida de[2]

- **Aprendizaje supervisado:** cada una de las instancias del problema tiene asociada una etiqueta (label). Durante la fase de aprendizaje, el sistema va tomando instancias del problema (que pueden estar o no parametrizadas) como entradas e intenta predecir la etiqueta asociada a dicha instancia como salida. El objetivo es por tanto, minimizar el error (obtenido a la salida del sistema) con respecto a los parámetros del modelo. Algunos de los problemas más ampliamente abordados desde el aprendizaje supervisado son los de **clasificación** y **regresión** [63, 44]. En los problemas de clasificación, cada instancia se asocia a una única etiqueta o categoría dentro de un conjunto finito de categorías conocidas. En los problemas de regresión, en cambio, la salida del sistema para cada instancia se asocia a una o varias variables numéricas. Para ello, se construye una función analítica que se ajuste a los datos, minimizando el error entre los valores predichos por el modelo y los realmente observados. La principal diferencia entre ambas estrategias reside en el dominio de las variables objetivo obtenidas a la salida de los sistemas. En problemas de clasificación, estas son de carácter discreto (categórico), mientras que en problemas de regresión, la salida es siempre un valor real y continuo.

- **Aprendizaje no supervisado:** los datos que representan el problema no tie-

nen asociada ninguna etiqueta. Atendiendo a los objetivos y la naturaleza del problema, las técnicas de aprendizaje no supervisado se utilizan principalmente en problemas de agrupamiento o clustering [93, 3], estimación de la densidad de probabilidad y de reducción de dimensionalidad:

- **Agrupamiento o clustering:** el objetivo en este tipo de problemas es encontrar características representativas en los datos, de manera que estos puedan ser agrupadas a partir de ellas. El número de grupos no tiene que estar necesariamente definido a priori, siendo el modelo el encargado de encontrar la distribución óptima de grupos con los que agrupar los datos. Una misma instancia puede pertenecer a uno o más grupos.
- **Estimación de la densidad de probabilidad:** en este tipo de problemas se intenta determinar la distribución de los datos dentro del espacio de las entradas. Los métodos más conocidos son los histogramas y las estimaciones basadas en núcleos (KDE- Kernel Density Estimation) [186].
- **Reducción de dimensionalidad:** a menudo, los datos relativos a un problema presentan dimensiones muy elevadas, lo que dificulta enormemente trabajar con ellos. La reducción de dimensionalidad se basa en la proyección de los datos de un espacio de alta dimensionalidad en otro espacio dónde los datos se representen en muchas menos dimensiones. La proyección de un espacio en otro, se puede asociar con la extracción de información relevante de los datos, de manera que la información redundante queda representada en las dimensiones no seleccionadas. Algunos ejemplos clásicos de esquemas de parametrización y reducción de dimensionalidad incluyen el análisis de componentes principales (PCA) y el análisis discriminante de Fisher (FDA) [114, 144].

- **Aprendizaje semi-supervisado:** utiliza datos de entrenamiento tanto etiquetados como no etiquetados. La distribución de datos es desbalanceada, es decir, la cantidad de datos no etiquetados es bastante superior a la cantidad de datos etiquetados. El objetivo de estas técnicas es extraer información relevante de los datos no etiquetados mediante métodos no supervisados, para posteriormente ajustar o introducir información a priori asociado a los datos en el modelo con métodos supervisados.

- **Aprendizaje por refuerzo:** este tipo de técnicas se basan en la búsqueda de acciones adecuadas que un agente software debe escoger dentro un entorno específico, con el objetivo de maximizar una función de "recompensa" [197]. El aprendizaje no está guiado por ninguna de las técnicas anteriormente descritas, sino que el sistema va aprendiendo a partir de observaciones extraídas de las respuestas del entorno ante determinadas acciones que él mismo genera (ensayo y error). Una característica importante del aprendizaje por refuerzo es la compensación entre la exploración, en la que el sistema intenta llevar a cabo nuevos tipos de acciones para evaluar su eficacia, y la explotación, en la que el sistema hace uso de acciones que se sabe que producen una gran recompensa. El abuso de uno u otro enfoque puede suponer resultados o rendimientos muy pobres.

- **Aprendizaje multitarea:** a partir de reglas y modelos que el sistema infiere analizando datos de diferentes problemas con múltiples representaciones, se obtienen soluciones conjuntas a problemas de diferente naturaleza pero que de alguna manera están relacionados [36].

- **Aprendizaje por transferencia:** también conocido como transferencia inductiva, la idea general es usar el conocimiento adquirido al resolver problemas donde hay una gran cantidad de datos disponibles en entornos donde la cantidad de datos es escasa [168, 209]. La creación de datos etiquetados es costosa, por lo que aprovechar al máximo los conjuntos de datos existentes es clave. El tamaño del corpus de datos que describe el problema a tratar determinará el método de trabajo desde la perspectiva de la transferencia inductiva. Si el nuevo conjunto de datos es muy pequeño, se usarán modelos pre-entrenados y se ajustarán las capas finales del mismo, evitando así el sobreajuste. En cambio, si el nuevo conjunto de datos es lo suficientemente grande, los modelos pre-entrenados serán usados como inicializadores, siendo recomendable un nuevo ajuste a partir de estos valores iniciales del sistema.

- **Aprendizaje activo:** es un caso especial de aprendizaje semi-supervisado en el que un algoritmo de aprendizaje usa interactivamente la información entrante (del usuario u otra fuente de información) para mejorar su conocimiento del dominio, y así optimizar su inferencia. Este tipo de aprendizaje es especialmente útil en escenarios donde el etiquetado manual es muy costoso y por tanto abundan los datos no etiquetados. [119, 185]

2.2. Detección y clasificación supervisada de eventos sismo-volcánicos

La detección y clasificación automática de señales sísmicas de origen volcánico (o sismo-volcánicas) es un problema muy complejo. Además de los problemas subrayados en la sección 1.3, relacionados con los efectos de fuente y propagación de las señales a través de la estructura volcánica, los observatorios vulcanológicos se enfrentan diariamente a otros problemas relacionados con el monitoreo volcánico (sección 2.2.1).

A grandes rasgos, un sistema de alerta temprana se fundamenta en el análisis de la actividad sísmica de los eventos considerados precursores de erupciones. Este análisis estudia en detalle las señales que van llegando en tiempo real al observatorio, detectando los diferentes eventos sismo-volcánicos que se van sucediendo y su evolución temporal. Ante un episodio pre-eruptivo, la dinámica intrínseca del volcán cambia, incrementando enormemente la actividad sísmica y por consiguiente, el número de eventos sismo-volcánicos que son analizados. Aunque el solapamiento o la simultaneidad en el tiempo de varios eventos no está ligado a una etapa pre-eruptiva o eruptiva, es en este tipo de escenarios, cuando más comúnmente se producen. Si a los problemas de propagación y fuente, añadimos el incremento de actividad en situaciones de emergencia y junto a ella, los problemas de simultaneidad (entre otros), observamos que el desarrollo de un sistema de alerta temprana fiable y eficiente, es una tarea de gran complejidad.

Desde el punto de vista de Machine Learning o aprendizaje automático, un sistema de alerta temprana de erupciones volcánicas y más concretamente, el problema del reconocimiento y clasificación de eventos sismo-volcánicos, puede ser abordado con diferentes técnicas y desde diferentes estrategias. El exhaustivo análisis que en los observatorios, los expertos vulcanólogos realizan sobre las señales registradas por los sismógrafos, deriva en bases de datos etiquetadas. El conocimiento experto implícito en forma de etiqueta asociado a los datos, así como la eficiencia en tareas de clasificación y

su menor coste computacional, postulan las técnicas de aprendizaje supervisado como primera alternativa para el desarrollo de este tipo de sistemas.

Considerando el aprendizaje supervisado como base, el diseño de un sistema de reconocimiento automático de señales sismo-volcánicas (VSR- Volcano-Seismic Recognition System), puede ser descrito y estructurado en varias etapas [148]:

1. **Adquisición de los datos:** esta etapa comprende desde la adquisición de las señales directamente del medio hasta la construcción final de la base de datos. Una vez las señales han sido registradas y digitalizadas por los sensores, son analizadas por personal experto y agrupadas (o etiquetadas) en función de sus características o disimilitudes. El agrupamiento o etiquetado de los datos podría ser considerado como una función matemática, $E(x)$, que dada una entrada (señal), le asigna una etiqueta o conjunto de pertenencia (tipo de evento).

2. **Acondicionamiento y parametrización de los datos:** las señales son parametrizadas y acondicionadas con el objetivo de facilitar el aprendizaje del sistema y poder representar los datos matemáticamente en un espacio de características lo más linealmente separable posible.

3. **Entrenamiento o aprendizaje [24, 63]:** a partir de los datos etiquetados (que pueden o no estar parametrizados), el sistema infiere una serie reglas, conocidas como reglas de decisión, que posteriormente serán la base de la función de clasificación ($C(x)$). Como se ha citado en la sección 2.1, un sistema de reconocimiento automático es un modelo adaptativo descrito por un conjunto de parámetros (θ). El ajuste de dichos parámetros corresponde a la inferencia de las reglas de decisión, afectando por tanto la descripción analítica de $C(x)$. Varios son los métodos que se pueden aplicar en el ajuste de los parámetros:

   a) **Métodos paramétricos:** $C(x)$ modela analíticamente los datos, es decir, el conjunto de parámetros (θ) se adapta a la forma analítica de los datos durante el aprendizaje. La mayoría de clasificadores automáticos se encuentran dentro de este grupo.

   b) **Métodos no paramétricos:** infieren las reglas de decisión del análisis de los datos, sin proponer ningún modelado analítico implícito. Los clasificadores basados en la estimación de densidad mediante Parzen y los clasificadores basados en árboles de decisión son los ejemplos más representativos de este grupo.

4. **Clasificación o reconocimiento:** una vez se han inferido la reglas de decisión y se ha construido $C(x)$, se espera que de manera automática el clasificador asigne una clase de pertenencia a un evento dado en función de sus características. La aplicación de $C(x)$ a todo el conjunto de datos genera un espacio conocido como espacio de características (Ω_x), en el que cada subconjunto de eventos pertenecientes a una misma clase (x_c) es delimitado por el espacio conocido como región de decisión (w_c) y una frontera de decisión (Figura 2.2.1). La capacidad de clasificar correctamente nuevos eventos que difieren de los utilizados durante la etapa de aprendizaje se conoce como generalización y es un término ampliamente usado como medida de calidad del clasificador. Una descripción más detallada sobre los tipos de clasificadores y técnicas de clasificación asociadas a las señales sismo-volcánicas se hará en la sección 2.2.3.

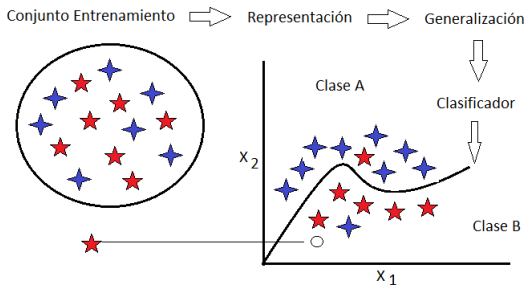


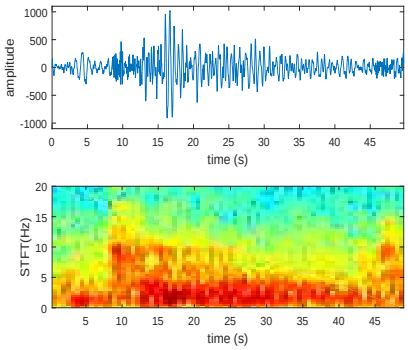
Figura 2.2.1: Proceso de clasificación. Cada elemento es representado de forma vectorial mediante un conjunto de características (x_1, x_2). El conjunto de elementos parametrizados forman el espacio de características (Ω_x), quedando dividido en dos regiones de decisión pertenecientes a las clases A y B. Una vez el sistema ha sido entrenado, $C(x)$ delimitará virtualmente ambas regiones con la frontera de decisión (w_c), de manera que ante la llegada de un nuevo elemento, este sea clasificado en una u otra clase en función de sus características.

5. **Evaluación del sistema:** el rendimiento del sistema se mide con respecto al conocimiento aportado por el personal experto. Como hemos citado en el punto 1, el agrupamiento o etiquetado experto de los datos podría ser considerado como una función matemática, $E(x)$, que dada una entrada le asigna una etiqueta o conjunto de pertenencia. El objetivo del clasificador es por tanto, aproximar $C(x)$ a $E(x)$ tanto como sea posible $(C(x) \approx E(x))$. Aunque existen varios modelos para medir el rendimiento del sistema o la bondad de la estimación, todos se basan en la minimización de una función coste o error que tiene en cuenta los errores de clasificación cometidos durante el proceso de evaluación, es decir, se evalúa el número de elementos que no han sido correctamente clasificados. Un clasificador se considera óptimo cuando el error de clasificación es mínimo, es decir, una regla o un conjunto de reglas de decisión son óptimas cuando el error de clasificación es mínimo.

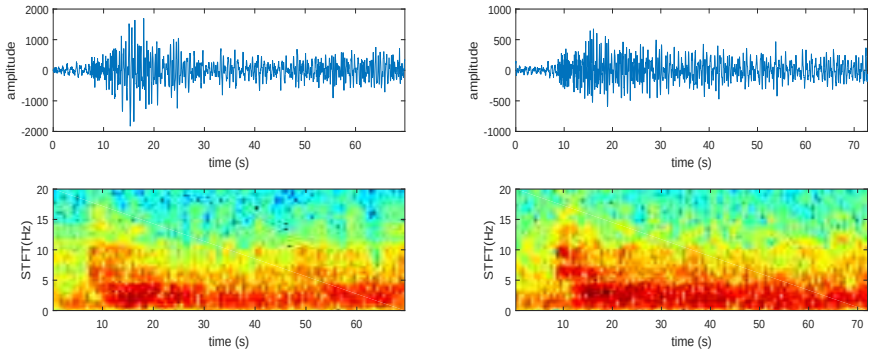
2.2.1. Adquisición de los datos: problemas relacionados con la naturaleza de los eventos sismo-volcánicos

El proceso de adquisición de datos sismo-volcánicos y más concretamente, la construcción de bases de datos robustas y fiables, es un problema muy complejo que requiere de conocimiento experto debido a la naturaleza de las señales sísmicas. Además de los problemas relacionados con los **efectos de sitio (fuente y propagación)** a través de la estructura volcánica (sección 1.3), las señales sísmicas presentan:

- **Problemas de variabilidad temporal y espectral:** la fuente o proceso físico generador del sismo es el origen de esta variabilidad, pudiéndose registrar señales del mismo tipo con diferente duración, diferente contenido espectral y diferente origen, en función del estado del volcán o incluso de su actividad sísmica en un determinado momento. Este mismo problema se ve aumentado si se estudian datos de diferentes volcanes y más aún, si los volcanes son de distinto tipo.
- **Superposición o simultaneidad de eventos:** durante una etapa activa del volcán, muchos eventos suceden de forma simultánea en el tiempo. En algunas



(a) El sistema de clasificación es capaz de detectar eventos que los operadores humanos no han considerado o no han detectado y son correctos, insertando un error en la matriz de confusión y reduciendo por tanto el rendimiento de los sistemas. La señal, etiquetada como Silencio-Tremor es clasificada como Silencio-Terremoto VolcanoTectónico- Tremor.



(b) Un cambio en la dinámica del volcán que deriva en el origen y la forma característica de algunos eventos. Izquierda: Evento etiquetado como Silencio-Tremor. Derecha: un cambio en la dinámica del volcán produce un pequeño evento híbrido al comiendo del tremor. Si la base de datos no es actualizada y los sistemas reentrenados, el evento híbrido puede no ser detectado.

Figura 2.2.2: Efectos de la incompletitud de la base de datos.

ocasiones, el origen de unos y otros eventos no tiene porqué estar asociado, pero en la mayoría de los casos, la forma continuada en el tiempo de ciertos eventos es la principal causa de aparición de otros eventos, superponiéndose ambos en una misma señal, afectando a la eficacia de los sistemas de reconocimiento automático, capaces de detectar eventos de forma secuencial.

- **Problemas derivados de la instrumentación:** la forma de onda y el espectro de las señales registradas están estrechamente influenciados por las características técnicas de las estaciones sísmicas usadas en la adquisición de las mismas. La instrumentación utilizada en el proceso de adquisición de los datos puede afectar a la forma de onda de la señal ya que cada instrumento puede presentar unas características en cuanto a la respuesta en frecuencia o ganancias diferentes.

El conjunto de estos problemas derivan en otro de mayor índole: **la fiabilidad de las bases de datos** creadas. Una base de datos se considera robusta y fiable si su tamaño es relativamente grande (gran número de eventos de cada clase), es decir, si representa suficientemente a cada uno de los eventos considerados; si la relación señal/ruido (SNR) de las señales es adecuada y si el etiquetado de los datos se ha llevado a cabo por expertos vulcanólogos cuyo criterio de análisis es claro y unificado.

A menudo, este último criterio se ve seriamente afectado, pues la falta de criterios objetivos definidos en cuanto a la descripción de las clases (motivada por los problemas asociados a las señales) conduce a situaciones de ambigüedad, donde el etiquetado de los eventos se realiza a partir de la experiencia del experto, en lo que se conoce como problemas de subjetividad.

Otro factor de gran relevancia relacionado con la fiabilidad es la **incompletitud** del etiquetado de las bases de datos. En el proceso de creación de la base de datos, muchos eventos no son considerados o incluso si ocurren de forma simultánea a otros, solo es considerado en el etiquetado el de mayor relevancia geofísica. Esto deriva en bases de datos incompletas y desbalanceadas que sesgan el rendimiento de los sistemas.

En la Figura 2.2.2 se muestra un ejemplo del efecto de la incompletitud de la base de datos. En el volcán objeto de estudio, algunos tremors volcánicos venían precedidos de un pequeño terremoto volcano-tectónico asociado a la fractura del conducto lávico debido al movimiento migratorio del magma. Este evento fue etiquetado por un grupo de expertos como Tremor volcánico, descartando por tanto el pequeño sismo del inicio del evento. El uso de sistemas de clasificación más novedosos y sofisticados permite detectar este pequeño sismo al inicio del evento, insertando un error en la matriz de confusión y reduciendo la bondad de la estimación cuando realmente es un acierto. Otro tipo de problema bastante generalizado derivado de la incompletitud de las bases de datos, es el asociado con el cambio de dinámica del propio volcán. En función del estado del volcán y su actividad sísmica, algunos eventos cambian sus características, apareciendo señales que nunca antes fueron registradas y por lo tanto que no están dentro de la bases de datos etiquetadas. Los sistemas que fueron entrenados sin tener en cuenta estas señales están obligados a clasificarlas, decrementando notablemente el rendimiento y debiendo ser nuevamente reentrenados para adaptarse a la nueva dinámica del volcán.

2.2.2. Etapa de aprendizaje

El proceso mediante el cual un sistema automático (clasificador) adquiere conocimiento (reglas y regiones de decisión) se conoce como aprendizaje. Como hemos citado anteriormente, existen dos métodos claramente diferenciables:

- Por un lado, encontramos los métodos no paramétricos, como los basados en árboles de decisión o los que usan medidas de similitud, que generan las reglas de decisión a partir del análisis de los datos, sin proponer ningún modelado analítico implícito.

- Por otro lado, encontramos los métodos paramétricos, que modelan analíticamente los datos en función de un conjunto de parámetros ($C(x) = C(x;\theta)$). Durante la etapa de aprendizaje (también conocida como etapa de ajuste de parámetros), cada uno de los parámetros ($\theta_p \epsilon \theta$) se ajusta de forma óptima al objetivo del problema.

El proceso de ajuste y evaluación del conocimiento adquirido por el sistema generalmente se lleva a cabo mediante tres conjuntos de datos: conjunto de entrenamiento o sintonización, conjunto de test, conjunto de validación, extraídos todos ellos aleatoriamente de la base de datos.

El objetivo del proceso de ajuste es, por tanto, obtener una alta eficiencia de clasificación en el conjunto de entrenamiento. Para ello, se ajustan los parámetros de la función $C(x;\theta)$ de manera que la clase asignada a cada instancia sea la correcta.. Este ajuste se realiza de forma iterativa, es decir, en cada iteración se hace un ajuste de los parámetros y se evalúa la desviación.

Para evaluar la desviación se hace uso del conjunto de validación. En cada iteración, una vez los parámetros han sido ajustados, se evalúa la capacidad de generalización del sistema (la capacidad que tiene el sistema para clasificar datos que no corresponden al conjunto de entrenamiento y que nunca ha visto). Si el rendimiento obtenido por el sistema en la iteración n no mejora el rendimiento obtenido por el sistema en la iteración *n-1*, el proceso de ajuste finaliza. Generalmente, los criterios de parada suelen activarse después de varias iteraciones sin mejora.

El rendimiento o bondad de estimación final del sistema es evaluado con el conjunto de test (sección 2.2.4). Una vez el proceso de ajuste ha terminado, se selecciona el modelo que mayor rendimiento ha obtenido en el conjunto de validación y se evalúa con el conjunto de test. El resultado obtenido, es el resultado de clasificación final del sistema.

Como hemos visto anteriormente (sección 2.2), el éxito de un sistema está determinado por la capacidad del método de aprendizaje de ajustar sus parámetros al objetivo del problema. Aunque generalmente, cada clasificador tiene sus propios algoritmos o métodos de aprendizaje, todos ellos pueden ser enmarcados en dos grandes grupos:

2.2.2.1. Aprendizaje mediante optimización de una función coste o error

Un clasificador ($C(x)$) puede ser evaluado definiendo una función coste o error sobre él. En términos estadísticos, el error clasificación está asociado con la probabilidad de que el sistema clasifique erróneamente una instancia debido al solapamiento de los modelos de clase en las regiones de decisión. El error de clasificación se asocia por tanto a una variable aleatoria e cuya probabilidad viene dada por [55]:

$$\begin{aligned} p(e;C) &= \sum_{w} p(e,w) = \sum_{c} p(e|w_c)p(w_c) \\ &= \sum_{x} p(e,x) = \sum_{x} p(e|x)p(x) \end{aligned} \tag{2.2.1}$$

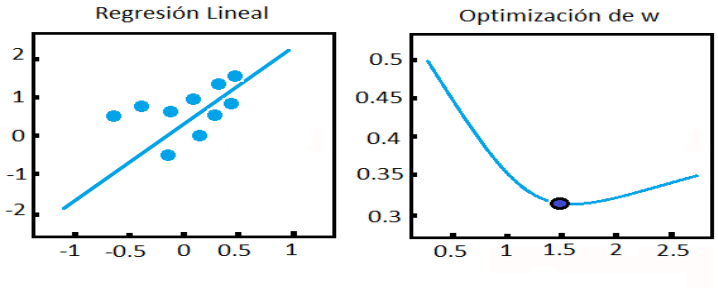


Figura 2.2.3: Ejemplo de ajuste mediante la optimización de una función de error. Supongamos un problema de regresión lineal ($y = w^t x + b$), donde w corresponde con los parámetros a determinar. Supongamos también un conjunto de entrenamiento que consta de diez puntos de datos, descritos por una única característica. El vector de parámetros w, contendrá un único parámetro, $w1$, asociado a la única características que describe los datos. (Izquierda) El objetivo de la regresión lineal es ajustar $w1$ de manera que $y = w1 * x$ se acerque lo más posible a todos los puntos de entrenamiento. (Derecha) La optimización del valor del parámetro $w1$ minimiza el error cuadrático medio entre los puntos de entrenamiento y la recta.

Siendo x la partición o conjunto de test, w el conjunto de clases al que puede asignarse cada instancia, $w_c$una clase específica del conjunto de clases y C, el modelo que define el clasificador en función de sus parámetros.

Un clasificador se considera estadísticamente óptimo ($C\,(x) \equiv C^*\,(x)$) si sus reglas de decisión minimizan la probabilidad de error $p(e)$:

$$C^*\,(x) \equiv \arg\min_{C} \{p(e; C)\} \tag{2.2.2}$$

Suponiendo un clasificador estadísticamente óptimo, se conoce como error de Bayes ($p_b(e)$), al mínimo valor posible asociado a la probabilidad de error dada una instancia:

$$\begin{aligned} p_b(e|x, C) \equiv \min_{C} \{p(e|x; C)\} &= \sum_{w_c \neq w_a} P(w_c|x; C) \\ &= 1 - \max_{C} \{P(w|x; C)\} \\ &= p(e|x; C = C^*) \end{aligned} \tag{2.2.3}$$

Donde $\sum_{w_c \neq w_a} P(w_c|x; C) = 1 - P(w_a|x; C)$ corresponde con el error de clasificación o probabilidad de clasificar una instancia x perteneciente a la clase w_a, erróneamente.

Siguiendo este enfoque, el ajuste óptimo de los parámetros θ que definen un clasificador puede ser planteado como un problema general de optimización. Para ello, el error de clasificación debe ser definido como una función que represente el error cometido por el modelo con respecto al conocimiento experto implícito en las etiquetas asociadas a las instancias del conjunto de entrenamiento. El objetivo es ajustar los parámetros del modelo de manera que la función de error sea minimizada (Figura 2.2.3). Aunque existe una gran variedad de funciones que pueden ser usadas en tareas de clasificación, generalmente son dos las que más se usan:

- **Error cuadrático medio (Mean Square Error-MSE) [85]:** el MSE de un estimador $C(x)$ descrito por un conjunto de parámetros θ con respecto a una

función de estimación desconocida $D(x)$ que determina la clase de pertenencia (w_c) de una instancia se define como:

$$\begin{aligned} MSE(C(x)) &= E\left[(D(x) - C(x;\theta))^2\right] = E[(w_c - C)^2] \\ &= E[(w_c - E[w_c])^2] + E[(E[w_c] - C)^2] \\ &= sesgo^2(C) + var(C) \end{aligned} \tag{2.2.4}$$

Donde el $sesgo^2(C)$ corresponde con el desajuste o desviación del modelo construido con respecto al modelo ideal y $var(C)$ corresponde con la varianza o variablidad del modelo, representando la sensibilidad del mismo y cómo varían sus predicciones o estimaciones. Generalmente, el desajuste se asocia a la asunción de hipótesis incorrectas en el modelado o incluso a un estado de sub-entrenamiento (subfitting). El compromiso de la desviación frente a la variabilidad es un problema de vital importancia y ampliamente estudiado [76], pues supone que los parámetros del clasificador estén o no óptimamente ajustados y por consiguiente el éxito en la tarea de clasificación. Un estudio más en detalle de este problema y sus efectos será llevado a cabo en la sección 2.2.2.3. Otra forma ampliamente usada de medir el error de clasificación mediante el MSE es haciendo uso de los vectores de predicciones. Supongamos que Y_{train} son las predicciones del modelo asociadas al conjunto de entrenamiento y que Y son las etiquetas o valores verdaderos de las instancias dicho conjunto. El error de clasificación expresado en términos de MSE se define como:

$$MSE(C(x)) = \frac{1}{n}\sum_{i}^{n}(Y_{train} - Y)^2 \tag{2.2.5}$$

- **Entropía cruzada o probabilidad logarítmica negativa(Cross-Entrophy-CE) [85]:** La entropía cruzada para dos distribuciones p y q sobre el mismo espacio de probabilidad se define como:

$$CE(p,q) = H(p) + D_{KL}(p||q) \tag{2.2.6}$$

Siendo $H(p)$ la entropía de p y $D_{KL}(p||q)$ la entropía relativa o divergencia de Kullback-Leibler entre p y q. Siguiendo el supuesto del MSE en el que Y_{train} son las predicciones del modelo asociadas al conjunto de entrenamiento, Y son las etiquetas o valores verdaderos de las instancias de dicho conjunto, y siendo ambas variables discretas, la entropía cruzada como error de clasificación se define como:

$$\begin{aligned} CE(p,q) = -\sum_{x} p(Y|x)\log q(Y_{train}|x) = \\ -\sum_{x} p(w_c|x)\log q(w_c|x) \end{aligned} \tag{2.2.7}$$

Donde $p(w_c|x)$ corresponde con la probabilidad de pertenencia real de la instancia x a una determinada clase w_c y $q(w_c|x)$ corresponde con la probabilidad de pertenencia a la clase w_c que el modelo le ha asignado a la instancia x. Como se puede observar en la Figura 2.2.4, cuando la probabilidad de pertenencia a la clase real asignada por el modelo ($q(w_c|x)$) es baja, la función de error se dispara, penalizando al modelo enormemente. Este tipo de medida del error es ampliamente usado en clasificadores cuya salida se corresponde

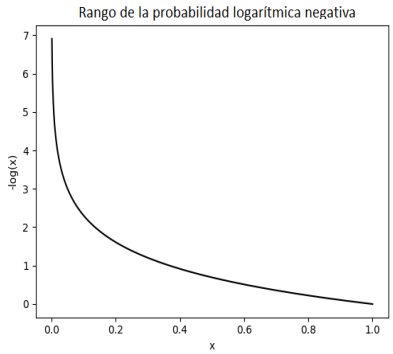


Figura 2.2.4: Representación gráfica de la probabilidad logarítmica negativa.

con las probabilidad de pertenencia asociadas a los datos de entrada, como por ejemplo los clasificadores cuya salida está determinada por la función softmax.

La minimización de la función de error es una tarea compleja, pudiéndose solucionar de forma algebraica en contadas ocasiones. Normalmente esta minimización se realiza mediante algoritmos iterativos (gradiente descendente), técnicas avanzadas de optimización local y heurísticas, que no aseguran la convergencia hacia una solución óptima del problema [85].

La forma analítica de la función de error con respecto a los datos, a menudo contiene varios mínimos locales, por lo que los modelos pueden encontrar varias soluciones al problema. El hallazgo de la solución óptima está determinado por el conocimiento a priori extraído de los datos que se le da al modelo. Este conocimiento sitúa el punto de partida del modelo en una determinada región dentro de la función de error más susceptible de encontrar la solución óptima.

2.2.2.2. Aprendizaje mediante estimadores estadísticos

De forma análoga al aprendizaje mediante la optimización de una función de error, el aprendizaje mediante estimadores estadísticos se basa en la optimización de una función coste, pero esta vez definida en términos estadísticos. Desde el punto de vista bayesiano, cada uno de los parámetros $\theta_p \epsilon\, \theta$ que describen el modelo, es una variable aleatoria independiente con una función de probabilidad marginal asociada y cuya probabilidad a posteriori se actualiza mediante una función de verosimilitud [24].

El objetivo del aprendizaje basado en estimadores estadísticos es por tanto el ajuste de los parámetros $\theta_p \epsilon\, \theta$ de forma óptima. Dicho ajuste o estimación se aborda desde dos perspectivas diferentes:

- **Estimación frecuentista [201]:** sea $F(E(x); C(x;\theta))$ una función de coste que representa el error cometido por el modelo $C(x;\theta)$ al estimar la clase de pertenencia de una instancia x con respecto a la función experta $E(x)$. Sea w la clase de pertenencia real de la instancia x y sea $\hat{w}$ la clase de pertenencia asignada por el modelo a dicha instancia. Según [201], la función de riesgo asociado al modelo $R(E(x); C(x;\theta))$: se define como el valor esperado de la función de coste $E[F(E(x); C(x;\theta))]$. En este sentido, el aprendizaje del modelo se puede abordar

como un problema general de optimización en el que el objetivo es encontrar los valores $\theta_p \epsilon\, \theta$ que hacen mínimo el riesgo:

$$\begin{aligned} R(E(x); C(x;\theta)) &\equiv E[F(E(x); C(x;\theta))] \\ &= \sum_x \sum_w F(E(x); C(x;\theta)) p(x,w) \\ &= \sum_x \sum_w F(w,\hat{w}) p(x|w) p(w) \end{aligned} \tag{2.2.8}$$

Una de las reglas de minimización del riego más usadas es la basada en la Máxima Probabilidad a Posteriori (MAP). Dada una instancia x, se define el riesgo condicional $R(\hat{w}|x)$: como el coste de asociar la instancia x con la clase $\hat{w}$:

$$R(\hat{w}|x) \equiv E[F(w;\hat{w}|x)] = \sum_w F(w,\hat{w}) P(w|x) \tag{2.2.9}$$

La regla MAP, que desde un punto de vista bayesiano asigna a la instancia x la clase w que hace máxima la marginal $P(w|x)$, es decir, $\hat{w} = \arg \max_w \{p(w|x)\}$, puede ser inferida minimizando el riesgo condicional bayesiano $R(w_i|x)$. Teniendo presente que el riesgo condicional se puede asociar con la probabilidad condicional de error, $R(w_i|x) \equiv p_{error}(w_i|x) = 1 - p(w_i|x)$, la minimización del riesgo se llevará a cabo minimizando $p_{error}(w_i|x)$, que es exactamente, encontrar la clase $\hat{w}$ que maximiza $p(w|x)$, es decir la clase que propone la regla MAP.

- **Estimación bayesiana**: A diferencia de los estimadores basados en el enfoque frecuentista, los estimadores bayesianos evalúan el riesgo de aproximar $C(x;\theta)$ mediante la probabilidad a posteriori $p(C(x;\theta)|x)$ del modelo y una función de coste $F(E(x); C(x;\theta))$. Dada una hipótesis de partida o un conocimiento a priori sobre una variable aleatoria θ representado por $p(\theta)$, y dada una función de verosimilitud de θ a partir de los datos de entrenamiento X, $L(X|\theta)$, la probabilidad a posteriori bayesiana queda definida como:

$$p(\theta|X) = \frac{p(X|\theta)}{p(X)} = \frac{L(X|\theta)p(\theta)}{\int L(X|\theta)p(\theta)d\theta} \tag{2.2.10}$$

El riesgo condicional, conocido ahora como el riesgo condicional de Bayes $R_B(C(x;\theta)|X)$ queda definido por tanto como la esperanza de un coste o error $E[F(E(X); C(X;\theta))]$ tomado sobre la probabilidad a posteriori:

$$\begin{aligned} R_B(C(X;\theta)|X) &= E[F(E(X); C(X;\theta))] \\ &= \int E[F(E(X); C(X;\theta))] p(\theta|X) d\theta \end{aligned} \tag{2.2.11}$$

Equivalentemente a los estimadores frecuentistas, un estimador bayesiano de θ es aquel que minimiza el riesgo bayesiano para todos los datos del conjunto de entrenamiento. Dependiendo de la forma analítica que describe la función coste o error, cada estimador bayesiano tendrá unas determinadas características. El uso del MSE como función de error en este tipo de estimadores es muy común, por lo que el mínimo error cuadrático medio (MMSE- Minimum Mean Square Error) es una de las técnicas de optimización más usadas. Un apunte importante en cuanto a los estimadores bayesianos es la influencia del tamaño de las bases de datos en las estimaciones. A medida que la base de datos es de mayor tamaño, la hipótesis de partida o probabilidad a priori se aproxima a una distribución normal, lo que reduce la influencia de la probabilidad a posteriori en la estimación final.

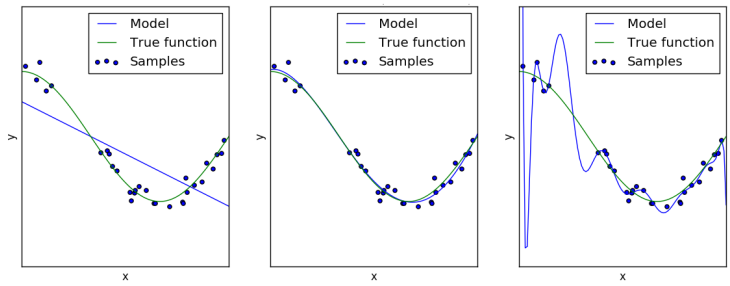


Figura 2.2.5: Compromiso de la desviación frente a la variabilidad. Suponiendo un conjunto de datos generados sintéticamente (muestreando aleatoriamente valores para la x y siendo después evaluados en una función cuadrática). (Sub-ajuste) La función lineal no puede capturar la curvatura presente en los datos. (Ajuste óptimo) La función cuadrática se adapta bien a los datos; ante la llegada de nuevos datos su capacidad de generalización es alta. (Overfitting) Los datos se han sobreajustado con un polinomio de alto grado. La solución pasa prácticamente por todos los puntos de entrenamiento, pero no con la forma analítica adecuada. Ante la llegada de nuevos puntos, la capacidad de generalización se verá gravemente afectada [85].

2.2.2.3. Compromiso de la desviación frente a la variabilidad. Introducción al uso de la regularización

Como hemos señalado anteriormente, en el ajuste de los parámetros θ de un clasificador, el compromiso de la desviación frente a la variabilidad es un problema de vital importancia y ampliamente estudiado [76]. Uno de los factores que más afecta a este compromiso es la complejidad de la base de datos.

En términos matemáticos, una base de datos se considera compleja si la variabilidad de las instancias que representan cada clase es alta. Cuanto mayor sea la variabilidad de una clase, mayor será la extensión de la región de decisión que el clasificador debe manejar para clasificar correctamente sus instancias, y por consiguiente su varianza.

En función de la varianza y de la cantidad de instancias bien clasificadas con respecto al modelo ideal (desviación), los clasificadores pueden ser agrupados en:

- **Fuertes o inestables:** caracterizados por una escasa desviación, son capaces de aproximar estrechamente la forma analítica de la base de datos. Suelen ser modelos complejos e inestables, es decir, con muy alta variabilidad. Un modelo fuertemente ajustado e inestable tiende a decrementar su capacidad de generalización. El ajuste excesivo de los parámetros a los datos de entrenamiento (overfitting), genera regiones y fronteras de decisión muy específicas, por lo que ante la llegada de nuevas instancias ligeramente distintas, la capacidad de clasificación se ve afectada (Figura 2.2.5).

- **Débiles o estables:** caracterizados por una notable desviación, estos modelos aproximan levemente la forma analítica de la base de datos. La variabilidad de las predicciones es menos extensa, por lo que se consideran modelos estables. De forma análoga al overfitting, una alta desviación o escaso ajuste de los parámetros a los datos de entrenamiento, genera regiones y fronteras de decisión muy simples, generalmente lineales, no siendo capaces de discriminar distribuciones de datos no lineales (Figura 2.2.5).

De forma general, los sistemas de reconocimiento y más concretamente los algoritmos que los guían, están diseñados para que funcionen en tareas muy específicas ("No free Luch Theorem") [210]. Esta especificidad se consigue construyendo un conjunto de preferencias en el algoritmo de aprendizaje, que a menudo se traduce en modelos sobreentrenados, muy ajustados a los datos de entrenamiento. Hasta el momento, el único método que hemos analizado para inferir conocimiento automáticamente, se basa en la aproximación de la función que describe la forma analítica de los datos de entrenamiento. Esta capacidad de representación se consigue modificando (agregando o eliminando) el número de funciones que el algoritmo puede elegir dentro del espacio de soluciones [85]. Siguiendo con el ejemplo propuesto en la Figura 2.2.5, podemos ver que el comportamiento del modelo se ve fuertemente afectado no solo por la cardinalidad del espacio de soluciones sino por la identidad específica de las funciones que lo componen. La permisibilidad de funciones polinomiales de alto grado en el espacio de soluciones que presumiblemente cumplen el criterio de minimización de error empujarán al algoritmo a escoger este tipo de representaciones, no siendo de ninguna manera óptimas.

Una técnica bastante extendida para evitar este tipo de sobreajustes es permitir a los algoritmos de aprendizaje la preferencia de determinadas soluciones sobre otras dentro del espacio de soluciones. Expresar preferencias por una función sobre otra es una forma general de controlar la capacidad de un modelo de incluir o excluir miembros del espacio de soluciones. Esta preferencia es conocida como **regularización** y se define como la modificación del algoritmo de aprendizaje con el objetivo de reducir el error de generalización pero no el error de entrenamiento. Aunque existen varias técnica para expresar preferencias por las diferentes soluciones, tanto implícita como explícitamente, una de las más usadas se basa en la incorporación de un término de penalización llamado regularizador en la función de coste o error. Un estudio más detallado de las diferentes técnicas regularizadoras será llevado a cabo en la sección 3.5, teniendo presente que este trabajo profundiza en los métodos de Deep Learning.

2.2.3. Técnicas de clasificación y clasificadores en el área de la sismología volcánica

La naturaleza de las señales sismo-volcánicas, la forma misma de adquirirlas y su posterior análisis en los observatorios, derivan en varias técnicas a la hora de clasificar automáticamente dichas señales. Dependiendo de la estructura de los datos se pueden distinguir dos aproximaciones diferentes:

- **Clasificación de forma aislada:** cada instancia de la base de datos corresponde con un único evento. Generalmente, en la construcción de bases de datos de este tipo de estructuras, los eventos son seleccionados o segmentados sobre una señal continua para posteriormente asignarles unas etiqueta.
- **Clasificación de forma continua:** cada instancia de la base de datos contiene un número indeterminado de eventos que se presentan como un flujo temporal de la señal. La construcción de bases de datos con este tipo de estructuras se lleva a cabo detectando, delimitando y etiquetando temporalmente cada evento. La clasificación en tiempo real, en la que la información proviene directamente de la fuente ininterrumpidamente es un ejemplo claro de este tipo de técnicas.

Existen dos grandes grupos de clasificadores dentro del área del aprendizaje supervisado:

- **Clasificadores estadísticos y/o probabilísticos:** las reglas de decisión y el modelado de datos se construyen a partir de inferencia estadística. Cuando la salida de este tipo de clasificadores corresponde con la probabilidad de pertenencia de una instancia o vector de entrada a cada una de las clases o categorías, $P(w_c|\mathrm{x}{=}x_i)$, se les llama clasificadores probabilísticos y presentan la ventaja de poder medir la fiabilidad de la predicciones o clasificaciones. Atendiendo a la descripción analítica de la función que describe la probabilidad de pertenencia de una instancia a las diferentes clases o categorías, se pueden distinguir varios tipos de clasificadores: lineales, cuadráticos, etc. Es importante destacar que no todos los clasificadores estadísticos son clasificadores probabilísticos. Un estudio más detallado de los clasificadores estadísticos y más concretamente de su aproximación bayesiana puede encontrarse en la sección 2.3.1.3.

- **Clasificadores no estadísticos:** las reglas de decisión y el modelado de datos se construyen sin usar métodos estadísticos. Los ejemplos más representativos son los árboles de decisión (Decision Trees) (basados en reglas lógicas o heurísticas), los sistemas expertos (Expert Systems), k-NNs (k- Nearest Neighborhood) (basados en medidas de similitud) y la comparación de plantillas (Template Matching).

2.2.4. Evaluación de los modelos

Aunque existen muchas métricas para evaluar la bondad de estimación o rendimiento de un sistema de clasificación, el método más extendido es trasladar los resultados de clasificación a una matriz de confusión. En dicha matriz, los eventos son representados como correctamente clasificados (C), borrados (eventos que el sistema no ha detectado) (D), insertados (eventos que no estaban en el conjunto de testeo y que el modelo ha insertado) (I) y sustituidos (confusiones entre eventos) (S). El número total de eventos N corresponderá por tanto con la suma de elementos correctamente clasificados, elementos borrados y elementos sustituidos ($N = C + D + S$). Una vez la matriz de confusión es construida, cada métrica hace un estudio diferente de ella. De entre las métricas más ampliamente usadas destacan:

- **Accuracy o rendimiento base:** es la forma más elemental de evaluar el rendimiento de un sistema de reconocimiento, ya que solo se basa en la contabilización del número de errores cometidos al clasificar los elementos pertenecientes a la partición de test. Es importante destacar que la definición de esta métrica varía en función del tipo de problema que se está abordando. Generalmente, la inserción y el borrado de eventos se suelen dar en escenarios de clasificación continuos, donde el número de etiquetas que le corresponde a cada sismograma no es conocido de antemano:

$$\%Acc = 100\frac{N - D - S - I}{N} = 100\frac{C - I}{N} \tag{2.2.12}$$

  Aunque el número de eventos insertados aparece como penalización en el cálculo de la medida, dichos eventos podrían estar correctamente detectados por el sistema, y haber sido ignorados por los expertos durante la construcción del corpus de datos dada las dificultades y particularidades de este tipo de señales.De forma análoga, en escenarios de clasificación aislada donde a cada sismograma le corresponde una sola etiqueta (quedando la ocurrencia de inserciones fuera

de lugar), el rendimiento base queda descrito simplemente por el porcentaje de eventos correctamente clasificados:

$$\%Acc = 100\frac{C}{N} \tag{2.2.13}$$

- **Valor-F o Medida-F [190]:** es una métrica basada en la precisión y la sensibilidad del sistema, entendiendo como precisión el porcentaje de etiquetas correctamente predichas ($\hat{Y}_i$) , y sensibilidad, como el porcentaje de etiquetas reales que fueron predichas (Y_i). En ambas ecuaciones, el valor n corresponde con el número de instancias o ejemplos considerados en la métrica.

$$\begin{aligned} precision &= \frac{TruePositives}{TruePositives + FalsePositives} = \\ &= \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \wedge \hat{Y_i}|}{|\hat{Y_i}|} \end{aligned} \tag{2.2.14}$$

$$\begin{aligned} sensibilidad &= \frac{TruePositives}{TruePositives + FalseNegatives} = \\ &= \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \wedge \hat{Y_i}|}{|Y_i|} \end{aligned} \tag{2.2.15}$$

Tanto la precisión como la sensibilidad son entendidas como medidas de la relevancia. Ambas medidas se definen como ideales si se acercan a 1, conociéndose ese estado como utilidad teórica. La fórmula general para evaluar un sistema de clasificación es::

$$F_\beta = (1+\beta^2) * \frac{precision * sensibilidad}{(\beta^2 * precision) + sensibilidad} \tag{2.2.16}$$

el valor de β denota la importancia que se le otorga a la precisión o la sensibilidad, de manera que un valor inferior a uno de β otorga mayor importancia a la precisión , mientras que un valor mayor que uno la otorga a la sensibilidad. Cuando el valor de β es igual a 1, el valor-F se considera como una media conjunta y es comúnmente conocido como F1-Score

$$\begin{aligned} F_1 &= F_{\beta=1} \\ &= 2 * \frac{precision * sensibilidad}{precision + sensibilidad} \end{aligned} \tag{2.2.17}$$

Esta métrica, en la que tanto la precisión como la sensibilidad tienen la misma importancia, ha sido usada como métrica evaluadora en sistemas multiclase, en los que se intenta discriminar entre varias clases o categorías, obteniendo para cada una de ellas una medida de precisión y sensibilidad.

2.3. Revisión del Estado del Arte en la clasificación automática de señales sismo-volcánicas

El reconocimiento automático de patrones así como los métodos de aprendizaje automático han sido extensamente aplicados en la clasificación y detección de eventos sismo-volcánicos durante las últimas cuatro décadas [19].

Uno de los trabajos pioneros en este ámbito fue [6], que introdujo la detección automática de las ondas características de un terremoto en una traza sísmica a finales de la década de los 70. Sabiendo de las limitaciones intrínsecas que presentaban las señales sísmicas (y sismo-volcánicas), mucho esfuerzo y mucho trabajo se centró en el estudio de parametrizaciones y desarrollo de métodos capaces de extraer características de los registros sísmicos con las que mejorar la eficiencia de los clasificadores [41, 40]. Estos estudios fueron durante la década de los '80s la base del modelado de patrones en tareas de detección y segmentación automática de señales.

A medida que la extracción de características iba siendo más eficaz, se pudo continuar con el estudio de clasificadores que había sido interrumpido años atrás. En este tiempo, principios de los 90', nacen las redes neuronales artificiales. Su aproximación como clasificador universal les otorgó una enorme popularidad, por lo que muchos trabajos de clasificación de señales sísmicas se basaron en ellas [72, 149, 141]. La aplicación de redes neuronales en el ámbito de la sismología volcánica se introdujo en [181, 166] (2005). Entrenado con características espectrales, [181] propone el uso de un modelo neuronal como clasificador binario, discriminando ruido y terremotos volcano-tectónicos pertenecientes al volcán Stromboli, Italia. Aprovechando el carácter explosivo de este mismo volcán, [166] (2006) empleó una red neuronal como detector de explosiones submarinas. Además de su uso como clasificadores, las redes neuronales también fueron usadas para entrenar mapas autoorganizados (SOM) con el objetivo de identificar e interpretar atributos y correlaciones entre eventos [124], analizar eventos de muy largo periodo [70] o incluso analizar temblores volcánicos pertenecientes a los volcanes Etna, Isla de Raoul, Ruapehu y Tongariro [125, 33, 31, 115].

A comienzos del siglo XXI, aunque las redes neuronales artificiales seguían siendo aplicadas en muchas áreas de la ciencia, fueron los clasificadores basados en redes bayesianas, como los Modelos Ocultos de Markov (HMM-Hidden Markov Models) los que cogieron gran protagonismo, ya que permitían la clasificación de registros continuos en tiempo real [153] (2001). No fue hasta mediados de la primera década (2005), cuando las Máquinas de Vector Soporte (SVM-Support Vector Machine) comenzaron a desplazar a las redes neuronales. [141] (2006) propuso una SVM de núcleos gaussianos para discriminar tremors volcánicos, coladas y explosiones pertenecientes al volcán de Stromboli.

Unos años más tarde (2007-2010), aprovechando las mejoras experimentadas en los sensores sísmicos, el análisis en tiempo real volvió a tomar gran protagonismo, y muchos observatorios vulcanológicos encontraban así la oportunidad de desarrollar una herramienta que fuese de ayuda en etapas pre-eruptivas, donde la elevada actividad y la gran cantidad de eventos registrados disminuye la capacidad de los operadores humanos de gestionar eficientemente la crisis. En este tiempo, basándose en el éxito conseguido en el reconocimiento automático de voz, [52, 109, 108, 133, 16] propusieron el uso de HMM como alternativa para la detección y clasificación de eventos sismo-volcánicos en tiempo real. Unos años más tarde (2012-2013), se empezó a apostar por esquemas combinados de reconocimiento capaces de adaptarse a entornos de monitorización multicanal. Fue entonces cuando los semi-HMM cogieron protagonismo dado su carácter híbrido (generativo-discriminativo), siendo empleados como sistemas de reconocimiento en el volcán Galeras [22, 23].

En esta misma fecha, debido a su sencillez y a su alta capacidad para modelar espacios de características complejas, los modelos basados en mezclas de gaussianas (GMMs-Gaussian Mixture Models) fueron usados como clasificadores capaces de actuar como selectores de características (mediante un algoritmo discriminativo) [10], y

como base para un sistema paralelo capaz de adaptar el número de componentes de forma óptima para cada tipo de evento [54].

Más recientemente, otra de las arquitecturas que más popularidad ha ido adquiriendo dada su sencillez y su estabilidad es el Random Forest, llegándose a convertir en la primera opción en tareas de clasificación en diferentes observatorios vulcanológicos [170, 97].

2.3.1. Breve reseña de los modelos usados en la clasificación de eventos sismo-volcánicos

Pese a que una descripción detallada de todas las técnicas anteriormente citadas y sus resultados quedan fuera del alcance de esta revisión, en esta sección se intentará dar una visión general de las técnicas que posteriormente serán usadas como base comparativa de los modelos propuestos en este trabajo.

Con el fin de organizar la información, las técnicas de clasificación serán agrupadas en función de la naturaleza del algoritmo de aprendizaje que las sostiene.

2.3.1.1. Máquinas de Vector Soporte (SVM)

Pertenecientes al grupo de clasificadores basados en análisis discriminativos, obtienen las fronteras de decisión aproximando una función discriminante mediante la combinación lineal, o no, de las características del espacio de descripción [201]. La función discriminante aproximada maximiza los márgenes entre fronteras conforme a una métrica concreta, por lo que dentro de los clasificadores basados en análisis discriminativo, estos modelos se les conoce como clasificadores de margen máximo. La idea es seleccionar un hiperplano de separación que equidista de los ejemplos más cercanos de cada clase, conocidos como vectores soporte.

La aproximación de la función discriminante en problemas complejos es a menudo una ardua tarea, por lo que las SVM usan núcleos o funciones de similitud capaces de transformar el espacio de descripción de características en otro de mayor dimensión donde dichas características pueden ser más fácilmente separadas de forma lineal (kernel trick) [26].

Siguiendo [39] y tomando como base un conjunto de datos binarios linealmente separable, una SVM puede ser definida como un discriminador lineal cuya función de clasificación $d_{SVM}(x)$ se corresponde con un hiperplano de separación $w^T + \Phi + b = 0$ de máximo margen en el espacio transformado de descripción de las características Ω_Φ. Dado un núcleo o una función de similitud $K(x_1, x_2) \equiv \Phi(x_1) \cdot \Phi(x_2)$ que transforma el espacio de descripción de características Ω_x en un espacio linealmente separable de mayor dimensionalidad Ω_Φ, una SVM queda definida como sigue:

$$\begin{aligned} d_{SVM}(x) &\equiv sgn\left\{w^T + \Phi + b\right\} = sgn\{\sum_s y_s \alpha_s K(x_s, x) + b\} \\ &= sgn\{\sum_s y_s \alpha_s \Phi(x_s) \cdot \Phi(x) + b\} \end{aligned} \tag{2.3.1}$$

Siendo $sgn(x)$ la función matemática que representa el signo de x que al tratarse de un problema de clasificación binaria corresponderá con la clase de pertenencia asignada por el modelo (± 1), y_s las etiquetas asociadas a cada vector x_s, α_s escalares multiplicadores de Lagranje y b el bias o desplazamiento de la frontera de decisión. Los núcleos de transformación más usados son el lineal $K_{lin}(x_1, x_2) = x_1.x_2$ y el de base radial

gaussiana $K_{RBF}(x_1, x_2) = \exp\left\{-0{,}5 \parallel \frac{(x_1 - x_2)}{\sigma} \parallel^2\right\}$. Durante la etapa de aprendizaje se maximiza el margen del hiperplano minimizando la norma del vector w que es una combinación lineal de los vectores soporte x_s y normal a dicho hiperplano. Aunque son modelos altamente escalables y con una elevada capacidad de generalización, son muy costosos de entrenar y evaluar, además de que la complejidad del entrenamiento aumenta si se quiere discriminar entre más de dos clases.

2.3.1.2. Random Forest (RF)

Concebidos como una modificación sustancial de los algoritmos de bagging [159], los RF[28] [161] son una combinación de árboles predictores no correlacionados

$$h = \{h_1(x), h_2(x), ..., h_n(x)\} \tag{2.3.2}$$

en los que cada árbol $h_k(x)$ realiza su predicción en función de un subconjunto de características extraídas de forma aleatoria del conjunto total de características que describen las instancias.

Para ello, cada uno de los árboles predictores aproxima la probabilidad de pertenencia del subconjunto de características (x_{sub}) a cada una de las clases w_ca partir de los parámetros que definen el árbol θ_k.

$$h_k(x_{sub}) = h_k(x_{sub}|\theta_k) = p(w_c|x_{sub}) \tag{2.3.3}$$

La idea, es promediar muchos modelos ruidosos (árboles) con el objetivo de reducir la variación sin aumentar el sesgo. Para ello, aprovechando la capacidad de los árboles para extraer estructuras de interacción compleja en los datos, cada modelo es entrenado independientemente. Ante la llegada de una nueva instancia, cada árbol asigna una clase de pertenencia, siendo la instancia finalmente clasificada (la función de clasificación final $f(x)$) en aquella clase que obtenga mayor cantidad de incidencias, es decir, mayor número de votos por todos los árboles.

$$f(x) = p(w_c|x_{sub}) = \sum_{1}^{n} p_n(w_c|x_{sub}, \theta_n) \tag{2.3.4}$$

Es uno de los algoritmos de clasificación más eficaces. El proceso de aprendizaje puede ser fácilmente paralelizado, lo que reduce enormemente su coste. No presenta restricciones ante ingentes cantidades de datos ni ante un elevado número de características descriptoras.

Además de extraer estructuras de interacción complejas de los datos, realza las características que son relevantes en la clasificación (selección de características). Contrariamente se ha observado que este tipo de modelos tienden a ser sobreajustados si la cantidad de datos no es lo suficientemente grande.

2.3.1.3. Clasificadores Probabilísticos

Un clasificador probabilístico es un modelo que aprende la distribución de probabilidad que siguen los datos, es decir, aprende la distribución de cada categoría o clase. Como hemos citado anteriormente, las reglas de decisión y el modelado de datos se construyen a partir de inferencia estadística.

Clasificadores de mezclas de Gaussianas (GMMs-Gaussian Mixture Models): estiman la distribución de probabilidad ($p(x|w_c)$) asociada a cada clase w_c mediante una combinación lineal de G gaussianas multivariadas $N(x;\mu_g,\sum_g)$. La asignación de pertenencia a una clase (regla de decisión) se realiza mediante el criterio bayesiano MAP y la combinación lineal de las G componentes gaussianas:

$$d_{GMM}(x) \equiv \arg\max_{w_c}\{p(w_c)p(x|w_c)\} = \arg\max_{w_c}\left\{p(w_c)\sum_{g=1:G}\alpha_g N_g(x;\mu_g,\Sigma_g)\right\} \tag{2.3.5}$$

Donde cada una de las G gaussianas multivariadas están ponderadas por el coeficiente α_g y se definen como:

$$\begin{aligned} N_g(x;\mu_g,\sum_g) &= \frac{1}{(2\pi)^{\frac{K}{2}}|\sum_g|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(x-\mu_g)\sum_g^{-1}(x-\mu_g)\right\} \\ &= ((2\pi)^k|\sum_g|)^{\frac{-1}{2}}\exp\left\{-\frac{1}{2}(x-\mu_g)\sum_g^{-1}(x-\mu_g)\right\} \end{aligned} \tag{2.3.6}$$

Siendo K el número de características que describen a las instancias, y $(\mu_g,\sum_g)$ la media y la matriz de covarianza de la g-ésima componente gaussiana del modelo. Los GMMs son capaces de modelar de forma sencilla espacios de características complejos (no lineales). El número de componentes gaussianas actúa como regularizador, controlando la complejidad y el nivel de ajuste del modelo a los datos.

Modelos Ocultos de Markov (HMM- Hidden Markov Models): concebidos como una generalización de los procesos de Markov, los HMMs son modelos probabilísticos estocásticos diseñados para reconocer patrones estructurados en secuencias. Aunque teóricamente fueron desarrollados en los años 60's, no fue hasta mediados de los 70's cuando se aplicaron por primera vez al reconocimiento automático del habla [14, 113]. Desde entonces, han sido ampliamente usados en el modelado de procesos en diversas áreas de la Física y la Ingeniería. Desde el punto de vista de procesado de señal, la obtención de un modelo estadístico de la señal como proceso aleatorio con estructura temporal es muy importante. En primer lugar, conocer el modelo de una señal facilita la comprensión de la fuente que la ha generado y la construcción de sistemas de predicción, reconocimiento e identificación. Y en segundo lugar, si se dispone de un modelo de señal, su posterior procesado se ve fuertemente facilitado.

Construidos a partir de características estadísticas de la señal, los modelos probabilísticos se basan en la suposición de que la señal puede ser caracterizada mediante un proceso aleatorio cuyos parámetros pueden ser estimados. Para ello, se parte de un conjunto S con N estados, $S=\{s_1,s_2,s_3,...,s_N\}$, interconectados entre sí y descritos por una matriz de probabilidades iniciales de los estados I, de forma que en cada instante de tiempo t, el modelo se encuentra en uno de los estados q_t, y a su vez genera un símbolo (observable) o_t. El resultado es un doble modelado estocástico del proceso, por un lado, se modela el espacio de características (símbolos) y por el otro, el espacio de la secuencialidad del evento (tiempo). Cada uno de estos estados q_t, tiene asociada una probabilidad $b_i(x_t)=P(o_t|q_t)$ de emitir un símbolo o_t en un instante de tiempo t, a partir de un vector de características x_t, verificando la siguiente condición de normalización:

$$\sum_{t=1}^{T} b_i(x_t) = 1(i=1,...,N) \tag{2.3.7}$$

Las transiciones entre estados, definidas como $a_{ij} = P(q_{t+1} = s_j | q_t = s_i)$, son las responsables del modelado temporal, y verifican también la condición de normalización:

$$\sum_{j=1}^{N} a_{ij} = 1(i = 1, ..., N) \tag{2.3.8}$$

De este modo, un HMM queda perfectamente definido proporcionando la matriz B de probabilidades de producción de símbolos y la matriz A de probabilidades de transición entre estados.

La secuencia de estados visitados para desplazarse desde el estado inicial hasta estado final se conoce como camino Q. Cada uno de los posibles caminos tiene asociada una probabilidad $P(Q|O, \lambda)$, siendo O el conjunto de símbolos emitidos por el modelo en cada estado y $\lambda = A, B, I$ el conjunto de parámetros que definen el modelo. El camino Q asociado a una secuencia de símbolos está inicialmente oculto, siendo decodificado en la fase de evaluación. En dicha fase, a partir de los modelos previamente entrenados y de las reglas que interconectan los modelos formando estructuras, se define una red de decodificación. Esta red de decodificación posibilita el reconocimiento de flujo continuo de datos tratándolo como una estructura secuencial. Una vez la red de decodificación ha sido creada, el cálculo del camino óptimo Q para una secuencia x de vectores observados dado el modelo λ, es llevado a cabo mediante el algoritmo de Viterbi [75]:

$$Q = \arg \max_{\forall q = \{q_1, ...q_T\}} \{p(q|x; \lambda)\} = \arg \max_{\forall q = \{q_1, ...q_T\}} \{p(x, q; \lambda)\} \tag{2.3.9}$$

2.3.1.4. Redes Neuronales Artificiales (ANN-Artificial Neural Networks)

Pertenecientes al conjunto de algoritmos conocidos como bio-inspirados, las redes neuronales artificiales son modelos computacionales que matemáticamente asemejan el comportamiento biológico de las neuronas y la estructura cerebral de un mamífero. Dicho cerebro se considera un sistema altamente complejo. Su unidad básica, la neurona, está masivamente distribuida, encontrando aproximadamente 10 billones de neuronas en la corteza cerebral y 60 trillones de conexiones neuronales. La manera en la que el cerebro responde ante los estímulos del mundo exterior y adquiere el conocimiento está directamente relacionada con las conexiones neuronales y su función principal, la transmisión de impulsos nerviosos.

Conocidos también como sistemas conexionistas, sistemas distribuidos paralelos o sistemas adaptativos, las redes neuronales se componen de una serie de unidades de procesamiento interconectados (neuronas) que operan en paralelo, adaptándose simultáneamente de forma modular y escalable con el flujo de información y las reglas de adaptación [176]. Cada unidad de procesamiento (neurona) [142] representa desde un punto de vista matemático, la respuesta a un estimulo mediante la combinación lineal de una serie de entradas ponderadas.

Las ANNs se consideran aproximadores universales, ya que usando la configuración o topología adecuada, pueden aproximar cualquier función continua, por muy compleja que sea. Esta características las convierte en modelos flexibles y potentes en los que basar un sistema de clasificación capaz de resolver problemas complejos y en los que a priori se tiene poca información. Llegando probablemente a ser el clasificador más usado en el reconocimiento de patrones, el perceptron multicapa (MLP-MultiLayer Perceptron) ha sido el origen de un amplio conjunto de técnicas y arquitecturas (Deep

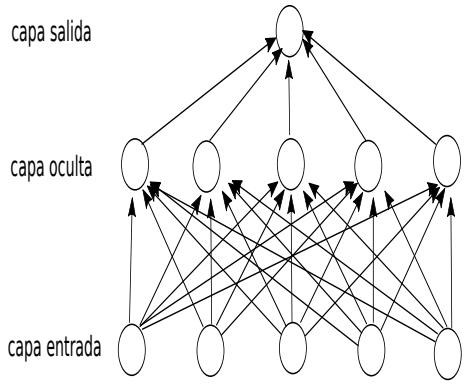


Figura 2.3.1: Ejemplo de una red neuronal (MLP) con una única capa oculta.

Learning), que basándose en los principios biológicos de su antecesor, se han convertido en el estado del arte en muchas áreas de la ciencia.

Un MLP o red neuronal multi-capa es una red neuronal compuesta de varias capas de neuronas entre la entrada y la salida de la red, en la que cada capa intermedia u oculta realiza una transformación no lineal de los datos. Cada transformación proyecta los datos de entrada en un nuevo espacio en el que pueden ser linealmente separables (Figura 2.3.1). Aunque un MLP con una sola capa oculta se puede considerar un aproximador universal, desde hace algunos años, se ha extendido el beneficio sustancial del uso de muchas capas ocultas (Deep Learning).

Formalmente, un MLP de una sola capa oculta, es una función $f : R^D \rightarrow R^L$, donde D es el tamaño del vector de entrada x y L es el tamaño del vector de salida y. En términos analíticos y matriciales, esta función se describe como:

$$y = f(x;\theta) = G(b^{(2)} + W^{(2)}(S(b^{(1)} + W^{(1)}x))) \tag{2.3.10}$$

Siendo $b^{(2)}, b^{(1)}$los vectores bías o sesgo que limitan la activación de las neuronas, $W^{(2)}, W^{(1)}$las matrices de pesos y G, S las funciones de activación de las neuronas. Como hemos citado anteriormente, cada neurona representa la respuesta a un estímulo mediante la combinación lineal de una serie de entradas ponderadas. En ese sentido, la respuesta al estímulo (vector de entrada) de la capa intermedia, o capa oculta, se obtiene obtiene aplicando una función de activación S a la combinación lineal ponderada entre los pesos de las unidades de entrada y la propia entrada.:

$$h(x) = S(b^{(1)} + W^{(1)}x) \tag{2.3.11}$$

Donde $W^{(1)} \epsilon R^{DxD_h}$ es la matriz de pesos que conecta el vector de entrada con la capa oculta. Cada columna de la matriz $W_i^{(1)}$representa los pesos del vector entrada a la *i-ésima* unidad oculta. Las funciones de activación más ampliamente usadas son la función tangente hipérbolica ($tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$), la función sigmoide ($\sigma(x) = \frac{1}{(1+e^{-x})}$) y la función ReLU o rectificador ($ReLU(x) = \text{máx}(0, x)$). El vector de salida se obtiene de forma análoga, utilizando como vector de entrada las salidas de la función de activación de la capa oculta:

$$o(x) = G(b^{(2)} + W^{(2)}(h(x)) \tag{2.3.12}$$

La función de activación G puede ser escogida en función del problema que se está abordando. En nuestro caso, al tratarse de un problema de clasificación, asociaremos la salida de cada neurona de la capa de salida con la probabilidad de pertenencia que el modelo asigna a cada una de las clases. Para ello, G corresponderá con la función softmax:

$$softmax(x) = \frac{\exp^x}{\sum_{k=1}^{K} \exp^{x_k}} \tag{2.3.13}$$

El aprendizaje del modelo se basa en el ajuste del conjunto de parámetros $\theta = \left\{b^{(2)}, b^{(1)}, W^{(2)}, W^{(1)}\right\}$ minimizando una función de coste o error. Para ello, se obtiene y propaga el gradiente del error (error cuadrático medio, entropía cruzada,etc) con respecto a los parámetros ($\frac{\delta E}{\delta \theta}$) mediante el algoritmo de retropropagación (back-propagation error) [85] .

La gran versatilidad de las redes neuronales ha derivado en multitud de variantes y en multitud de técnicas de entrenamiento que deberían ser descritas en este espacio, pero teniendo presente que este trabajo se basa en la aplicación de muchas de ellas en el área de sismología volcánica, dedicaremos un capítulo a su estudio y descripción.

2.4. Conclusiones

En este capítulo hemos abordado una descripción general del aprendizaje y reconocimiento automático de patrones. Para esto:

- Hemos definido brevemente los tipos de aprendizaje automático existentes y hemos clasificado los diferentes problemas en cada uno de estos tipos en función de su naturaleza.
- Teniendo presente las particularidades asociadas a las registros sismo-volcánicos y aprovechando el exhaustivo análisis que los expertos vulcanólogos realizan sobre estos como medida de control de riesgo volcánico, hemos descrito y estructurado el diseño de un sistema de reconocimiento automático de señales sismo-volcánicas (VSR- Volcano-Seismic Recognition System) bajo un enfoque de aprendizaje supervisado.
- Descritas las etapas de diseño y motivados sus problemas asociados, hemos realizado una revisión sobre los sistemas de reconocimiento más relevantes del área, analizando detenidamente aquellos que nos servirán como base comparativa de los modelos propuestos en esta tesis en capítulos posteriores.

Capítulo 3

Deep Learning: DNNs y CNNs

Pese a que todas las técnicas anteriormente citadas habían resultado ser bastante eficaces en multitud de problemas de reconocimiento automático, a mitad de la década de los 2000 (2006-2007), un nuevo conjunto de técnicas, conocido como aprendizaje profundo (Deep Learning) [85], se consolidó como el framework de última generación en áreas de la ciencia como el reconocimiento de voz [98, 61] o la visión por computador [96].

El aprendizaje profundo, entendido como un marco tecnológico en el que se encuentran diferentes tipos específicos de arquitecturas software con las que representar funciones de complejidad creciente a partir de la agregación o apilamiento de múltiples capas de procesamiento, se ha convertido en el estado del arte en muchas disciplinas y de muy distinta índole.

Aunque su mayor impacto en la ciencia se produjo hace aproximadamente una década (con las mejoras tanto en los dispositivos de procesamiento hardware como en los métodos de ajuste y entrenamiento), la idea de usar modelos bioinspirados con múltiples capas de procesamiento data de comienzos de los 60's. No obstante, a día de hoy, siguen derivándose modelos novedosos que poco a poco están cogiendo protagonismo en áreas tan novedosas como la clasificación de imágenes hiperespectrales [216, 42, 43, 175], predicción meteorológica [66, 134], predicción de ciclones o segmentación y detección de objetos en imágenes satelitales [118, 199, 91, 105].

Este capítulo, por tanto, se centrará en el marco teórico tanto de las redes neuronales profundas (DNNs) como de las redes convolucionales (CNN). En la sección 3.1 comenzaremos describiendo la base teórica de las DNNs y analizaremos la importancia de los esquemas de inicialización en el rendimiento final del sistema. En la sección 3.2 motivaremos el concepto de pre-entrenamiento no supervisado y analizaremos las implicaciones de su uso como estrategia para construir DNNs a partir del apilamiento de arquitecturas más simples, como las máquinas restringidas de Boltzmann (RBM) o los auto-encoder (AE). En la sección 3.3 abordaremos la descripción del proceso de optimización y ajuste de los parámetros basado en el descenso estocástico del gradiente. En la sección 3.4 describiremos la base teórica de las CNNs, puesto que las usaremos como base comparativa de los modelos propuestos en capítulos posteriores. Finalmente, en la sección 3.5, describiremos algunas de las técnicas avanzadas de regularización con las que combatir las situaciones de sobreajuste que este tipo de sistemas, por su naturaleza, a menudo encuentran.

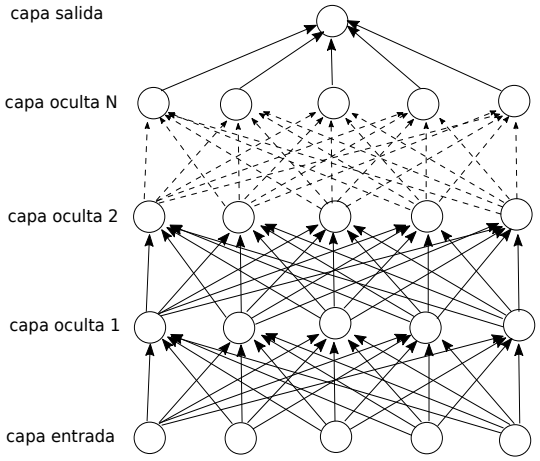


Figura 3.1.1: Ejemplo de red neuronal profunda.

3.1. Redes neuronales profundas (DNN-Deep Neural Networks)

Las redes neuronales profundas, también conocidas como perceptrones de múltiples capas MLPs, son los modelos de aprendizaje profundo por excelencia. Como ya introdujimos en la sección 2.3.1.4, una red neuronal es un modelo bioinspirado compuesto de varias capas de neuronas (unidades de procesamiento) entre la entrada y la salida de la red, en la que cada capa (capa oculta) realiza una transformación no lineal de la información que llega hasta ella. Esta transformación no lineal proyecta la información en un nuevo espacio de representación linealmente separable. Cada una de las capas ocultas está compuesta por una serie de unidades de procesamiento (neuronas), que a su vez están conectadas con todas y cada una de las neuronas de la capa inmediatamente superior e inferior.

Siguiendo [85], el objetivo de una DNN es aproximar una función f^* a partir del ajuste óptimo de un conjunto de parámetros θ (que definen cada una de las conexiones que unen las unidades de procesamiento entre capas). Como se puede apreciar en la Figura 3.1.1, una red neuronal es un grafo dirigido que aproxima una función f^* como composición de una red de funciones $f^{(1)}, f^{(2)}, f^{(3)}$, de manera que$f^{(1)}$ es la primera capa oculta del modelo, $f^{(2)}$ la segunda y $f^{(3)}$la tercera:

$$f^*(x) = f^{(3)}(f^{(2)}(f^{(1)}(x))) \tag{3.1.1}$$

La profundidad del modelo viene determinada por el número de funciones que operan en la función de composición, siendo considerados profundos aquellos en los que operan dos o más funciones [98]. La idea de usar múltiples capas de unidades, o múltiples representaciones vectoriales de los datos, deriva de la neurociencia. En ese sentido, en muchas ocasiones, la elección de las funciones de representación ($f^{(i)}(x)$) asociadas a cada capa, están estrechamente relacionadas con las funciones que las neuronas biológicas implementan y que han sido derivadas de observaciones neurocientíficas. Las DNNs abordan las limitaciones de las arquitecturas clásicas, permitiendo que los modelos aprendan representaciones jerárquicas y abstractas de los datos y mejorando la generalización estadística [129].

3.1.1. Importancia de los esquemas de inicialización

El entrenamiento mediante el que las DNNs adquieren el conocimiento a través de una serie de reglas de decisión que posteriormente serán la base de la función f^* que se intenta aproximar, no es muy diferente del proceso de entrenamiento que sigue cualquier otro modelo de aprendizaje automático basado en gradiente descendente. Las DNNs se ajustan usando optimizadores iterativos basados en gradientes que minimizan una función de error previamente definida:

- Dado un conjunto de datos $S = \{(x_1, y_1), (x_2, y_2), ...(x_n, y_n)\}$ donde $x_i \epsilon \mathbb{R}^d$ se consideran ejemplos aproximados y ruidosos de f^*, e $y_i \epsilon \mathbb{R}^c$ se corresponde con la etiqueta de cada uno de estos ejemplos, el gradiente entre lo obtenido por el modelo $f^*(x_i)$ y el valor real de su etiqueta y_i, medido con una función de error, es propagado hacia atrás a cada uno de los parámetros que componen el modelo, siendo estos ajustados y siendo las reglas de decisión por tanto, actualizadas (sección 3.3).

Una de las características importantes de las redes neuronales, sean o no profundas, es que las no linealidades de sus funciones de activación hacen que las funciones de error sean analíticamente no convexas, por lo que a diferencia de las funciones convexas que en teoría convergen a partir de cualquier parámetro inicial, el descenso del gradiente estocástico no garantiza la convergencia en el proceso de optimización [85]. En este sentido, los esquemas de inicialización de los parámetros tienen especial relevancia, ya que son los responsables de:

- Que el algoritmo consiga converger y encuentre un mínimo en la función de error, o por el contrario, encuentre dificultades numéricas y falle por completo.

- Determinan la velocidad de convergencia (suponiendo que el algoritmo converge), y a que punto convergen, pudiendo encontrar puntos en la función de error muy diferentes. Este es un problema bastante importante, ya que puntos con a priori, igual error, pueden derivar en sistemas con errores de generalización muy variable, repercutiendo directamente en el rendimiento final de los mismos.

En su mayoría, las estrategias de inicialización existentes son simples, teniendo como objetivo la obtención de propiedades que mejoren la convergencia. Sin embargo, el conocimiento que hasta la fecha se tiene de estas propiedades es muy reducido.

- En primer lugar se desconoce cuáles de las propiedades que mejoran la convergencia se conservan una vez el proceso de aprendizaje comienza a avanzar.

- En segundo lugar, como hemos señalado anteriormente, cabe la posibilidad de que puntos iniciales que pueden ser beneficiosos desde el punto de vista de la optimización, pueden ser perjudiciales desde el punto de vista de la generalización.

En este escenario, la única propiedad conocida es que los parámetros iniciales necesitan romper la simetría entre unidades [85], es decir, si dos unidades ocultas con la misma función de activación están conectadas a las mismas entradas, estas unidades deben tener diferentes parámetros iniciales. De lo contrario, el algoritmo de aprendizaje (determinista), aplicado a un función coste o error, actualizará constantemente ambos parámetros en la misma medida, rompiendo el proceso de aprendizaje en una de las unidades.

Este hecho motiva la inicialización aleatoria de los parámetros, intentando así que cada unidad calcule una función diferente del resto. Dentro de la inicialización aleatoria, los pesos se suelen asociar a valores aleatorios extraídos de distribuciones gaussianas o uniformes. La elección de una u otra distribución es aún objeto de estudio [129, 85]. No ocurre lo mismo con la escala inicial del rango de valores que toman las distribuciones, la cual ha sido exhaustivamente estudiada y ha demostrado afectar profundamente en el proceso de optimización y por consiguiente, en la capacidad de generalización de los modelos [85].

Aunque a priori se puede pensar que una inicialización con valores altos puede evitar redundancias en las funciones base, es decir, debilitar la simetría entre unidades, la multiplicación matricial asociada al cómputo de estas funciones puede derivar en problemas numéricos e interrumpir el proceso de aprendizaje:

- Por un lado, puede ocurrir que los valores de que se propagan tanto directa, como inversamente, sean tan grandes que no produzca conocimiento alguno, incurriendo por tanto en lo que en la literatura se conoce como **desbordamiento del gradiente** (Exploding Gradients Problem) [164].

- Por otro lado, valores muy elevados pueden dar lugar a valores extremos que provocan la saturación de la función de activación, causando la pérdida completa del gradiente a través de unidades saturadas.
  Contrariamente, la elección de valores muy próximos a cero podría ocasionar la no activación de las unidades, y aplicando la lógica anterior, que durante el cómputo de las funciones base y la multiplicación matricial asociada, los valores que se propagan sean tan pequeños que no produzca conocimiento alguno, incurriendo por tanto en lo que en la literatura se conoce como **desvanecimiento del gradiente** (Vanishing Gradients Problem)[164] .

3.1.1.1. Inicialización de los parámetros

Con el objetivo de intentar paliar la problemática anteriormente expuesta (desvanecimiento/desbordamiento de gradientes) surgieron heurísticas capaces de acotar la escala inicial de los valores de los parámetros relacionando el valor de los mismos con el número de unidades de entrada y salida que estos interconectan. En su versión más primitiva, los valores de los parámetros de una capa completamente conectada con m entradas y n salidas son muestreados de una distribución uniforme descrita como:

$$W_{ij} \sim U(-\frac{1}{\sqrt{m}}, \frac{1}{\sqrt{m}}) \tag{3.1.2}$$

Este enfoque, aunque tuvo bastante repercusión, suponía un grave problema a medida que se incluían más capas en los modelos, ya que la varianza de activación de unidades y de gradientes propagados entre las capas era muy dispar. Con el objetivo de igualar dichas varianzas entre capas y violando la suposición de que los modelos se basan en operaciones matriciales lineales, [82] obtuvo lo que en la literatura se conoce como Inicialización Normalizada (Normalized Initialization), dando lugar a dos esquemas diferentes:

$$W_{ij} \sim U(-\sqrt{\frac{6}{m+n}}, \sqrt{\frac{6}{m+n}}) \tag{3.1.3}$$

si se parte de una distribución uniforme, o

$$\sigma = \sqrt{\frac{2}{m+n}}$$
$$W_{ij} \sim N(0, \sigma) \tag{3.1.4}$$

si se parte de una distribución gaussiana (normal).

Siguiendo la misma suposición anterior, surgieron otros esquemas de inicialización aleatorios basados en matrices ortogonales en los que se escogía cuidadosamente un factor de escala g o ganancia, que tuviese en cuenta la no linealidad aplicada en cada capa [180]. No obstante, la elección del valor de$g$ no es tan sencillo y arbitrario, ya que al aumentarlo o disminuirlo, incrementaremos o disminuiremos el número de neuronas que se activan, y por tanto el número de gradientes a propagar durante el ajuste, lo que nuevamente no situaría a las puertas del problema de desvanecimiento o desbordamiento del gradiente. Sin embargo, un ajuste óptimo de este factor, preservaría los modelos de estas problemáticas y permitiría el entrenamiento de los mismos con cientos de capas ocultas [195].

Observando las ecuaciones 3.1.2, 3.1.3 y 3.1.4, se concluye que a medida que aumentamos el tamaño de la capa ($m>>$), las reglas de acotado disminuyen la cota de la inicialización o rango de valores disponibles para la inicialización de cada parámetro ($\frac{1}{\sqrt{m}}$, $\sqrt{\frac{6}{m+n}}$, $\sqrt{\frac{2}{m+n}}$).

Este problema se intentó abordar con lo que se conoce como Inicialización Dispersa (Sparse Initialization). La idea de este esquema es inicializar todas las unidades o neuronas de manera que todas tengan k parámetros distintos de cero. El objetivo es mantener la cantidad de información entrante a cada unidad independiente del número de entradas que hasta ella llegan, evitando que esta información disminuya a medida que incrementamos el tamaño de la capa [138]. No obstante, dado que este enfoque impone valores iniciales relativamente grandes en aquellos parámetros distintos de cero, el ajuste vía gradiente descendente de los mismos es bastante costoso.

Alternativamente a todos estos esquemas se encuentran los esquemas de ajuste manual, en los que a partir de pequeños conjuntos de datos se observa la desviación estándar de las activaciones o de los gradientes, de manera que si los valores de los parámetros son demasiados pequeños, el rango de activaciones se reducirá, lo que nos indicará la necesidad de aumentar el valor de los parámetros iniciales. De igual modo, en vez de usar el rango de activaciones, el valor inicial de los parámetros puede ser argumentando en función de la desviación estándar de los gradientes propagados [146]. Aunque este enfoque puede ser automatizado y presenta a priori un menor coste computacional, no ha sido tan ampliamente usado.

3.1.1.2. Inicialización basada en pre-entrenamiento

Los problemas de los esquemas de inicialización anteriormente expuestos se minimizan cuando se dispone de grandes bases de datos de entrenamiento. En ese caso pierden relevancia ya que generalmente se consigue que el proceso de optimización de la función de error converja.

Contrariamente, cuando la cantidad de datos no es lo suficientemente grande, aunque se disponga de un esquema de inicialización a priori óptimo, la convergencia en el proceso de optimización no está garantizada:

- En primer lugar, se está intentando optimizar un modelo en el que la diferencia entre el número de parámetros y de datos es muy reducida.

- En segundo lugar, si la profundidad del modelo no es excesiva y se tiene una cantidad de parámetros consecuente con el número de datos, puede ocurrir que los modelos converjan a un óptimo local, decrementando así la capacidad de generalización y por tanto su rendimiento.

En este sentido, el éxito del entrenamiento sin pre-entrenamiento previo de un modelo para resolver una tarea específica dependerá de la complejidad del modelo y de la complejidad de la tarea.

Una de las estrategias más eficaces y ampliamente extendidas a la hora de abordar problemas de contrastada dificultad, es la construcción de modelos complejos a partir de otros más simples, es decir, construir modelos eficaces en la resolución de tareas simples con el objetivo de afrontar a posteriori la resolución de tareas más complejas a partir de una combinación de modelos simples. Este tipo de estrategias son conocidas como estrategias de pre-entrenamiento y de forma general, pueden ser vistas como un proceso de inicialización del proceso de optimización, capaz de acelerar y mejorar la convergencia de la minimización de la función de error. Dos son las variantes más extendidas:

- Por un lado, siguiendo [18, 85], cada una de las capas que componen la red profunda es entrenada de forma supervisada (MLP), tomando como entrada la salida de la capa anterior. Para ello, se comienza entrenando de forma supervisada una arquitectura con una sola capa oculta. Una vez entrenada, se construye una nueva arquitectura con una sola capa oculta en la que la capa de entrada será la salida de la capa oculta del primer modelo, siendo la capa oculta de este segundo modelo, la segunda capa oculta del modelo profundo. Este procedimiento será repetido tantas veces como capas ocultas se deseen (Figura 3.1.2).

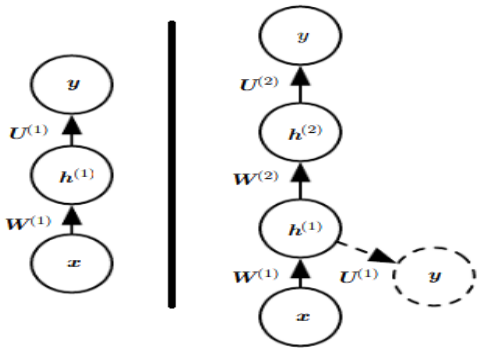


Figura 3.1.2: Proceso de construcción de una DNN mediante pre-entrenamiento supervisado. Izquierda: MLP con una sola capa oculta. Derecha: La salida de la capa oculta del primer MLP será usada como entrada en un nuevo MLP.

- Por otro lado, siguiendo [17, 85, 102], cada una de las capas que componen la red profunda es entrenada de forma no supervisada usando arquitecturas capaces de extraer información representativa de la estructura interna de los datos. Las arquitecturas usadas durante este proceso son el Denoising Auto Encoder (DA) y la Máquina Restringida de Botlzmann (Restricted Boltzmann Machine-RBM) (Figura 3.1.3):

- Según [203], siguiendo un esquema similar al expuesto en el pre-entrenamiento supervisado, los DA pueden ser apilados formando una red profunda conocida como SDA (Stacked Denoising AutoEncoder) (sección 3.2.2), en la que la representación latente encontrada por la capa inmediatamente anterior es usada como entrada en la siguiente capa. Cada capa se entrena de forma individualizada minimizando el error al reconstruir su entrada, que como hemos señalado anteriormente, es la representación latente de la capa inmediatamente anterior.
- De forma análoga, siguiendo [102], las RBM pueden ser apiladas y entrenadas mediante algoritmos voraces en lo que se conoce como redes de creencia profunda (DBN-Deep Belief Networks) (sección 3.2.1). DBN son modelos gráficos capaces de modelar la distribución de probabilidad conjunta entre los datos de entrada y las N capas ocultas.

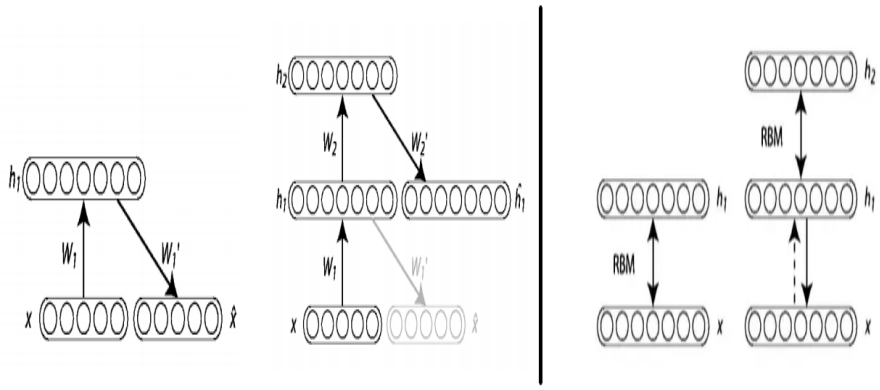


Figura 3.1.3: Proceso de construcción de una DNN mediante pre-entrenamiento no supervisado. Izquierda: construcción de un SDA a partir del apilamiento de DAs. Derecha: construcción de una DBN a partir del apilamiento de RBMs

3.1.1.3. Ventajas del pre-entrenamiento no supervisado frente al supervisado

Aunque ambas estrategias han demostrado ser útiles acelerando y mejorando la convergencia de la solución encontrada, han sido los enfoques no supervisados los que más repercusión han experimentado [68, 69]:

- En primer lugar, el pre-entrenamiento no supervisado tiene como consecuencia un efecto de regularización, inicializando los parámetros del modelo en un punto de la región del espacio de parámetros que de alguna manera hace que el proceso de optimización sea más efectivo, logrando minimizar la función de error. Este hecho se consigue a partir del aumento de la magnitud de los parámetros (pesos) del modelo. Dado el carácter no lineal de las funciones de activación de las neuronas, el aumento de la magnitud de los parámetros tiene el efecto de acentuar dichas no linealidades y en consecuencia, hacer de la función de coste

una función más compleja, complicando el recorrido de distancias significativas del procedimiento de descenso de gradiente durante la optimización del modelo, y por tanto actuando como regularizador.

- En segundo lugar, otorga al modelo un conocimiento a priori de la distribución de datos que se puede explicar por la forma en que las capas ocultas han sido previamente entrenadas. En ambos casos, usando DA o RBM, se maximiza un límite inferior [204, 101] que conduce a una minimización de la divergencia Kullback-Leibler entre la distribución verdadera de datos y la aproximada por los parámetros del modelo [25]. Este conocimiento se puede aprovechar para construir representaciones de los datos estadísticamente confiables que puedan ser usadas para predecir la pertenencia a una determinada clase en futuras tareas de clasificación [204, 101], ya que las transformaciones de características aprendidas de los datos, que a su vez son características predictivas de los principales factores de variación en $P(X)$, son también características predictivas con respecto a Y [68].

- Finalmente, en la situación estándar de entrenamiento con descenso de gradiente estocástico, las mejoras en la generalización no disminuyen cuando el tamaño de corpus de datos es grande. Los cambios introducidos en los parámetros de los modelos durante el pre-entrenamiento tienen un mayor impacto en la convergencia del procedimiento de optimización.

3.2. DBNs y SDAs como DNNs

Recientemente se ha llegado a la conclusión de que redes de cierta profundidad no necesitan de un pre-entrenamiento no supervisado para combatir el problema del sobreajuste cuando los corpus de datos no son lo suficientemente grandes, permitiendo una mejor retropropagación sin desvanecimiento o desbordamiento del gradiente si se usan técnicas como batch normalization [111], dropout [193], residual learning [96], Adam solver [117], Glorot initialization [82] y funciones de activación ReLU. No obstante, es importante destacar que los resultados en los que se apoyan estas conclusiones se han extraído de corpus de datos que tienen cientos de miles de instancias.

Como hemos señalado en la sección anterior, la construcción de DNNs como apilamiento de modelos más simples permite una mejor y más rápida convergencia que los enfoques basados en inicializaciones específicas. No obstante, aunque este tipo de estrategias suelen ser computacionalmente mucho menos costosas que las basadas en inicializaciones específicas, rara vez garantizan una solución óptima, siendo necesario un proceso de optimización general con el que ajustar eficazmente los parámetros del modelo (fine-tuning). De forma general, esta optimización se lleva a cabo minimizando una función de error (sección 3.3) que evalúa el error de las clasificaciones a partir de la salida de la capa softmax (Figura 3.2.1).

Aunque ambas arquitecturas (DBN y SDA) son consideradas DNNs, el uso de DAs o RBMs durante la etapa de pre-entrenamiento, derivará en modelos pertenecientes a familias diferentes dentro de la literatura.

3.2.1. RBM como base de una DBN

Propuestas en [102], las DBNs son modelos generativos compuestos por el apilamiento de varias RBMs capaces de modelar la distribución conjunta de una observación

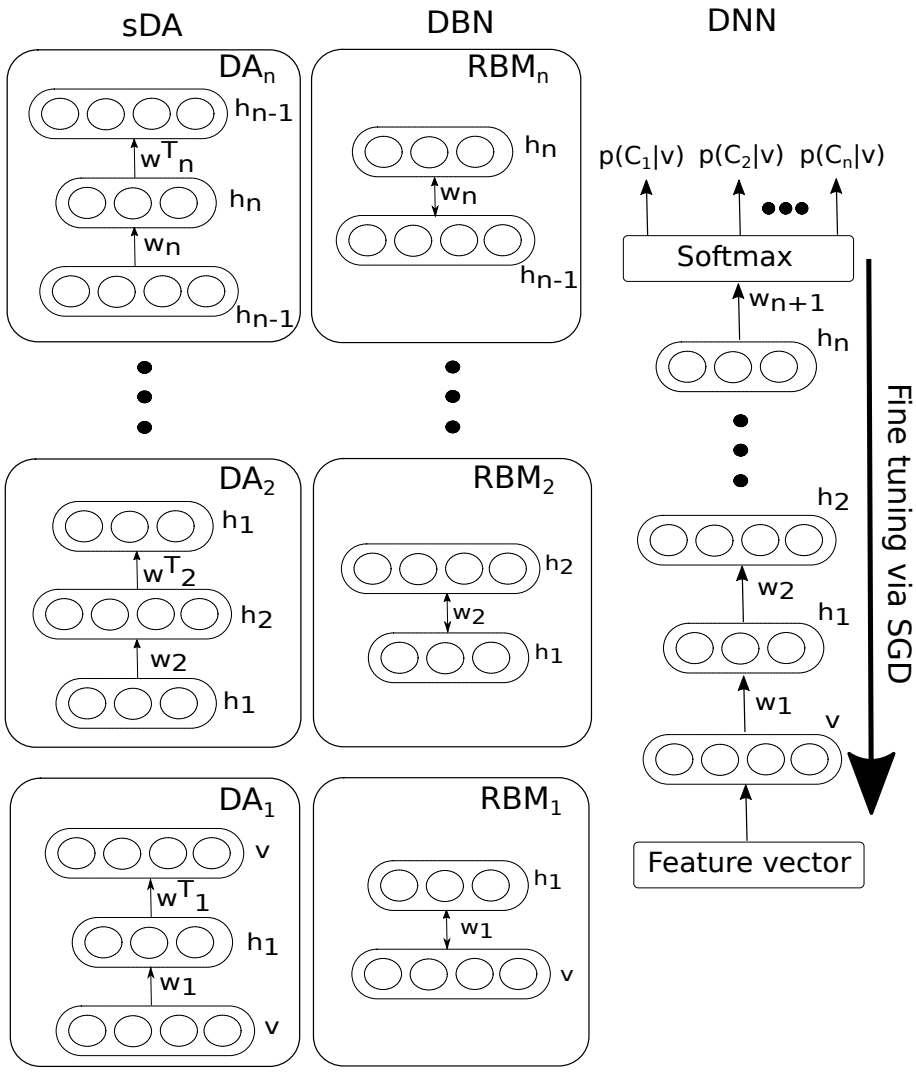


Figura 3.2.1: Construcción de una DNN a partir del apilamiento de a: DA b: RBM. Finalmente, se describe el proceso de optimización mediante el descenso de gradiente estocástico (SGD) .

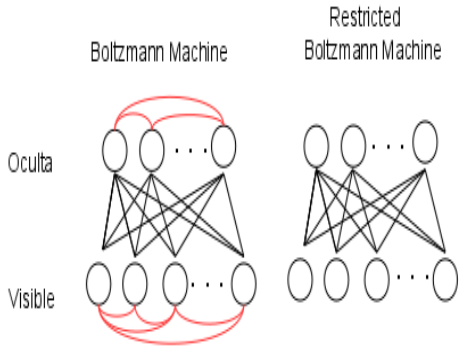


Figura 3.2.2: Descripción gráfica de una Máquina de Boltzmann y una Máquina Restringida de Boltzmann (RBM). La principal diferencia entre ambas es la falta de conexiones entre unidades de un mismo nivel en la RBM.

o vector de entrada x y las N capas ocultas que definen el modelo como:

$$p(x, h^1, h^2, ...h^N) = \left(\prod_{k=0}^{N-2} p(h^k|h^{k+1}) \right) p(h^{l-1}|h^l) \tag{3.2.1}$$

Siendo $x = h^0$, $p(h^{k-1}|h^k)$ la distribución condicional de la unidades visibles de la capa $k-1$ dadas las unidades ocultas de la RBM de la capa k y $p(h^{l-1}|h^l)$ la distribución conjunta de la RBM de la última capa.

La optimización por separado de cada una de las capas, fuerza al modelo a mejorar el límite variacional de la distribución de los datos de entrenamiento, incrementando la probabilidad de $p(x)$ [99].

Por definición, una RBM [73, 101] es un modelo gráfico asociado a un grafo bipartito no dirigido (o Markov Random Field (MRF)) que define una función de probabilidad sobre un conjunto de unidades binarias estocásticas. Como se muestra en la Figura 3.2.2, las unidades estocásticas de la capa inferior se conocen como unidades visibles (v_j), mientras que las unidades de la capa superior se denominan unidades ocultas (h_i). En una RBM las unidades de un mismo nivel no están conectadas entre sí, lo que implica que estas sean condicionalmente independientes, y por consiguiente sus probabilidades condicionadas vienen dadas por:

$$p(h|v) = \prod_{i=1}^{N} p(h_i|v) \tag{3.2.2}$$

$$p(v|h) = \prod_{j=1}^{M} p(v_j|h) \tag{3.2.3}$$

Siendo N y M, el número de unidades de la capa oculta y la capa visible, respectivamente.

En este sentido, viendo la RBM como una red neuronal estocástica, y usando la función sigmoide como función de activación en sus unidades, la probabilidad condicional

de las activaciones se define como:

$$p(h_i = 1|v;\theta) = \sigma\left(\sum_{j=1}^{M} w_{ji}v_j + a_i\right) \tag{3.2.4}$$

$$p(v_j = 1|h;\theta) = \sigma\left(\sum_{i=1}^{N} w_{ji}h_i + b_j\right) \tag{3.2.5}$$

donde w_{ij} representa la interacción simétrica entre la j-ésima unidad visible v_j y la i-ésima unidad oculta h_i, y b_j e a_i los bías de las unidades ocultas y visibles respectivamente.

En un MRF, la distribución para un conjunto dado de parámetros (pesos de las conexiones) θ, viene descrita en términos de una función de energía $E(v, h; \theta)$:

$$p(v, h; \theta) = \frac{exp(-E(v, h; \theta))}{Z} \tag{3.2.6}$$

siendo Z un factor de normalización o función de partición dado por la expresión:

$$Z = \sum_{v,h} e^{-E(v,h;\theta)} \tag{3.2.7}$$

y siendo E la función de energía, que aunque podría tener diferentes expresiones en función de la distribución de probabilidad que sigan las unidades visibles y las unidades ocultas, dado que en nuestro caso han sido restringidas a valores binarios (y por tanto siguen una distribución de Bernouilli) se define como:

$$E(v, h; \theta) = -\sum_{i=1}^{N}\sum_{j=1}^{M} w_{ji}h_i v_j - \sum_{j=1}^{M} b_j v_j - \sum_{i=1}^{N} c_i h_i \tag{3.2.8}$$

Finalmente, basándonos en la naturaleza restringida del modelo, podemos definir la distribución marginal de las unidades visibles como:

$$p(v;\theta) = \sum_{h} p(v, h; \theta) = \frac{1}{Z}\sum_{h} e^{-E(v,h;\theta)} \tag{3.2.9}$$

Donde de nuevo, w_{ij} representa la interacción simétrica entre la j-ésima unidad visible v_j y la i-ésima unidad oculta h_i, y b_j e a_i los bías de las unidades ocultas y visibles respectivamente.

Los algoritmos de aprendizaje de este tipo de modelos se basan en el ascenso del gradiente de una función de error, generalmente descrita por el logaritmo de la función de verosimilitud que compara la diferencia de dos patrones, el de entrada y el de salida del modelo [101, 73]. Teniendo presente el carácter generativo del modelo, se aprovechará la reconstrucción de la entrada (ecuación 3.2.9) a partir de la información latente capturada por las unidades ocultas para definir la función de verosimilitud:

$$\begin{aligned} \ln p(v|\theta) &= \ln \frac{1}{Z}\sum_{h} e^{-E(v,h;\theta)} \\ &= \ln \sum_{h} e^{-E(v,h;\theta)} - \ln \sum_{v,h} e^{-E(v,h;\theta)} \end{aligned} \tag{3.2.10}$$

siendo, finalmente, el ascenso del gradiente de dicha función, el encargado de guiar el proceso de aprendizaje del modelo:

$$\frac{\partial \ln p(v|\theta)}{\partial \theta} = \frac{\partial}{\partial \theta}\left(\ln \sum_h e^{-E(v,h;\theta)}\right) - \frac{\partial}{\partial \theta}\left(\ln \sum_{v,h} e^{-E(v,h;\theta)}\right)$$

$$= -\frac{1}{\sum_h e^{-E(v,h;\theta)}} \sum_h e^{-E(v,h;\theta)} \frac{\partial E(v,h;\theta)}{\partial \theta} + \tag{3.2.11}$$

$$\frac{1}{\sum_{v,h} e^{-E(v,h;\theta)}} \sum_{v,h} e^{-E(v,h;\theta)} \frac{\partial E(v,h;\theta)}{\partial \theta} \tag{3.2.12}$$

Sabiendo que la probabilidad condicionada del modelo puede ser reescrita en términos de la función de energía como:

$$p(h|v;\theta) = \frac{p(v,h;\theta)}{p(v;\theta)} = \frac{\frac{1}{Z} e^{-E(v,h;\theta)}}{\frac{1}{Z}\sum_h e^{-E(v,h;\theta)}} = \frac{e^{-E(v,h;\theta)}}{\sum_h e^{-E(v,h;\theta)}} \tag{3.2.13}$$

la ecuación 3.2.11 puede ser reescrita como:

$$\frac{\partial \ln p(v|\theta)}{\partial \theta} = -\sum_h p(h|v) \frac{\partial E(v,h;\theta)}{\partial \theta} + \sum_{v,h} p(v,h;\theta) \frac{\partial E(v,h;\theta)}{\partial \theta} \tag{3.2.14}$$

$$= E_{data}(v_i h_j) - E_{model}(v_i h_j) \tag{3.2.15}$$

donde el primer término (E_{data}) corresponde con el valor esperado observado de la función de energía en el conjunto de entrenamiento (muestreando h_j dados los v_i de acuerdo con el modelo) , y el segundo término (E_{model}) corresponde al valor esperado de la función de energía bajo la distribución del modelo.

3.2.1.1. Divergencia Contractiva y Divergencia Contractiva Persistente como algoritmo de entrenamiento de una RBM

Como hemos citado anteriormente, calculando el gradiente del logaritmo de la función de verosimilitud se puede derivar la regla de actualización de los parámetros de una RBM:

$$\triangle w_{ij} = \frac{\partial \ln p(v|\theta)}{\partial \theta} = E_{data}(v_i h_j) - E_{model}(v_i h_j) \tag{3.2.16}$$

Desde un punto de vista computacional, el cálculo de esta expresión puede llegar a ser demasiado costoso, ya que el número de operaciones crece exponencialmente con el número de variables del modelo (especialmente cuando se calcula el segundo término), siendo imprescindible el uso de técnicas de Monte Carlo (MCMC) [38, 24] para aproximarlas.

Los dos métodos basados en estas técnicas y más ampliamente usados para el entrenamiento de RBM son la Divergencia Contractiva (CD-k) y la Divergencia Contractiva Persistente (PCD) [34, 101].

- **CD-k:** cada vector de entrada del conjunto de datos de entrenamiento se usa para inicializar el proceso de muestreo de Gibbs [38] y actualizar el valor del conjunto de unidades ocultas. Estas unidades ocultas se utilizarán posteriormente para calcular las unidades visibles que reconstruyen el vector de entrada. Este procedimiento puede repetirse una o más veces, produciendo la muestra v^k después de

k iteraciones del muestro de Gibbs. Una vez el muestro de Gibbs ha finalizado, el gradiente correspondiente a un ejemplo de entrenamiento y referenciado en la ecuación 3.2.14, puede ser aproximado por:

$$\frac{\partial \ln p(v|\theta)}{\partial \theta} \approx CD^k(\theta, v^0) = -\sum_h p(h|v^0)\frac{\partial E(v^0,h)}{\partial \theta} + \sum_h p(h|v^k)\frac{\partial E(v^k,h)}{\partial \theta} \tag{3.2.17}$$

El número iteraciones utilizadas durante el muestreo (o el número muestras utilizadas para aproximar $E_{model}(v_i h_j)$) es un híper parámetro del algoritmo. Siendo generalmente ajustado a 1, en lo que se conoce como CD-1.

- **PCD:** definido como una aproximación de la CD, este algoritmo reemplaza la muestra v^0 por una muestra de la cadena de Gibbs que es independiente de cada vector de entrada. En lugar de inicializar el proceso de muestreo de Gibbs para cada vector de entrada del conjunto de datos de entrenamiento, mantiene el estado inicial de la cadena "persistente", es decir, el estado de la cadena inicial se corresponderá con el estado de la muestra v^k del proceso anterior, obteniendo la muestra $v^{k'}$ que corresponderá con $E_{model}(v_i, h_j)$ tras k pasos del proceso de Gibbs. La idea fundamental que subyace a la PCD es que se podría suponer que las muestras del proceso de muestro de Gibbs permanecen cerca de la distribución estacionaria si la tasa de aprendizaje es lo suficientemente pequeña y, por lo tanto, el modelo cambia ligeramente entre actualizaciones de parámetros [73]. Al igual que ocurría con la CD, el número iteraciones utilizadas durante el muestreo (o número de muestras utilizadas para aproximar $E_{model}(v_i, h_j)$) es un hiper parámetro del algoritmo.

3.2.1.2. Gaussian-Bernoulli RBM

Generalmente, el uso de RBMs está condicionado a observaciones representadas por vectores binarios (distribuciones binarias), pero a menudo, es necesario modelar distribuciones que tienen una función de probabilidad continua. Siguiendo [73], existen varias formas de abordar este problema. La primera de ellas escala los datos de entrada al intervalo [0, 1] y modela la probabilidad de que las variables visibles sean uno. Es decir, en lugar de muestrear valores binarios, se considera como el estado actual de la variable $v_i = p(v_i = 1|h)$, siendo v_i la i-ésima variable visible y h el conjunto de variables ocultas. .

La segunda y más utilizada, es usar una RBM con neuronas binarias en la capa oculta y valores reales en la capa visible. Para ello la probabilidad de las variables visibles $p(v_i = 1|h;\theta)$, se modela a partir de una distribución Normal cuya media corresponde con la combinación lineal de los pesos y el valor de las unidades ocultas y cuya varianza es la unidad. Este tipo especial de RBM se conoce como Gaussian-Bernoulli RBM y queda completamente definida como sigue:

$$p(h_j = 1|v;\theta) = \sigma\left(\sum_{i=1}^{I} w_{ij} v_i + a_j\right) \tag{3.2.18}$$

$$p(v_i = 1|h;\theta) = N\left(\sum_{j=1}^{J} w_{ij} h_j + b_i, 1\right) \tag{3.2.19}$$

dónde i y j, corresponden con la i-ésima y j-ésima unidades visibles y ocultas respectivamente.

Aunque el proceso de entrenamiento y el cálculo del gradiente de la función de verosimilitud no se ven afectados por el uso de este tipo de RBM, la distribución sobre las unidades visibles y ocultas dados los parámetros del modelo definida en términos de la función de energía, $p(v,h;\theta) = \frac{\exp(-E(v,h;\theta))}{Z}$, se ve levemente afectada, ya que la distribución gaussiana que modela la probabilidad de las variables visibles cambia levemente la forma analítica de la función de energía E:

$$E(v,h;\theta) = -\sum_{i=1}^{I}\sum_{j=1}^{J} w_{ij}v_i h_j - \frac{1}{2}\sum_{i=1}^{I}(v_i - b_i)^2 - \sum_{j=1}^{J} a_j h_j \tag{3.2.20}$$

3.2.1.3. Justificación y ventajas del apilamiento de RBMs en el pre-entrenamiento

El entrenamiento y apilamiento de RBMs además de ser una estrategia para construir DNNs, ayuda a mejorar la aproximación de la distribución de probabilidad que modela los datos de entrenamiento.

Esta conclusión es extraída de manera teórica por [102] con el siguiente análisis: Supongamos una DBN con solo dos capas ocultas, siendo h^1, h^2 con sus correspondientes pesos W^1, W^2 cada una de las capas ocultas. El logaritmo de la probabilidad de los datos de entrada (x), se define como:

$$\log p(x) = KL\left(Q(h^1|x)||p(h^1|x)\right) + H_{Q(h^1|x)} + \\ + \sum_{h} Q(h^1|x)\left(\log p(h^1) + \log p(x|h^1)\right) \tag{3.2.21}$$

Siendo KL la divergencia Kullback-Leibler entre la distribución a posteriori $Q(h^1|x)$ de la RBM perteneciente a la primera capa (suponiendo que es independiente) y la probabilidad $p(h^1|x)$ para la misma capa pero definida sobre el conjunto de la DBN, es decir, teniendo en cuenta la probabilidad a priori $p(h^1, h^2)$ y siendo $H_{Q(h^1|x)}$ la entropía de la distribución $Q(h^1|x)$.

Según [102], al inicializar ambas capas ocultas como $W^2 = W^{1^T}$, la divergencia KL es nula, ya que $Q(h^1|x) = p(h^1|x)$. Por tanto, si se entrena el primer nivel de RBM y después se optimiza la Ecuación 3.2.21 con respecto a W^2 manteniendo W^1 fijo (esquema de pre-entrenamiento no supervisado, sección 3.1.1.2), se puede aumentar la probabilidad de $p(x)$ y por consiguiente, mejorar la fiabilidad de los sistemas emitiendo probabilidades de pertenencia más elevadas:

- Al aislar los términos que solo dependen de W^2 de la ecuación 3.2.21 se obtiene:

$$\log p(x) = \sum_{h} Q(h^1|x)\left(\log p(h^1)\right) \tag{3.2.22}$$

  Por consiguiente, optimizar la ecuación 3.2.21 con respecto a W^2, equivale a entrenar una nueva RBM, usando la salida de $Q(h^1|x)$ (distribución a posteriori de la primera RBM) como distribución de entrada.

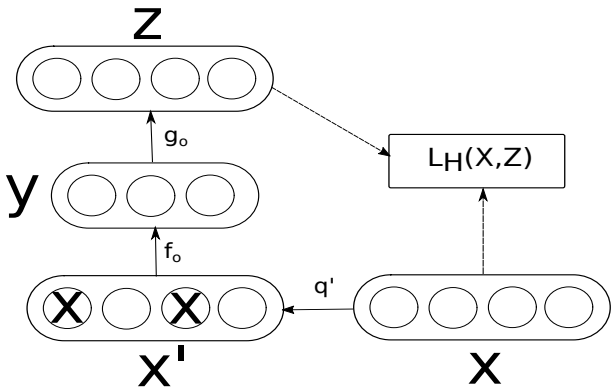


Figura 3.2.3: Descripción gráfica de un DA como una variante estocástica del Auto Encoder (AE) clásico en el que los datos de entrada son corrompidos con un determinado porcentaje de ruido

3.2.2. DA como base de un SDA

Los SDA [203] son modelos compuestos por el apilamiento de varios Denoising Auto Encoder (DA) . Por definición, un DA es una variante estocástica del Auto Encoder (AE) clásico en el que los datos de entrada son corrompidos con un determinado porcentaje de ruido, desplazando cada vector de entrada a una zona diferente del espacio de características y obligando al modelo a aprender información latente de relevancia con la que reconstruir la entrada e impidiendo que aprenda a modelar la función identidad. Principalmente utilizados para reducir la dimensión de los datos y aprender características representativas con los que reconstruirlos, el DA es uno de los algoritmos no supervisados más ampliamente usados.

De forma analítica, un AE, crea un mapeo determinista (codificación) de una entrada $x \epsilon [0,1]^d$ a partir una representación (oculta) $y \epsilon [0,1]^{d^{'}}$ (Figura 3.2.3):

$$y = f_\theta(Wx + b) \tag{3.2.23}$$

Donde f_θ corresponde con una función no lineal (como por ejemplo la función sigmoide), W con los parámetros del proceso de codificación y b con el bías de las unidades de la capa oculta. A partir de esta codificación (y), también conocida como representación latente, se reinvierte el mapeo (decodificación) obteniendo una reconstrucción del vector de entrada $z \epsilon [0,1]^d$:

$$z = g_{\theta'}(W^{'}y + b^{'}) \tag{3.2.24}$$

De forma análoga a la función de codificación, $g_{\theta'}$ corresponde con una función no lineal, $W^{'}$con los parámetros del proceso de decodificación (que pueden o no ser una versión traspuesta de W) y $b^{'}$con el bías de las unidades de la capa de salida.

El objetivo del modelo, es por tanto, optimizar el conjunto de parámetros $(W, W^{'}, b, b^{'})$ de manera que error medio de reconstrucción entre x y z sea mínimo.

Con el objetivo de evitar que el DA aprenda la función identidad, lo normal es crear un cuello de botella, de forma que existan menos unidades ocultas que unidades visibles, es decir, capas de codificación de menor dimensión que la capa de entrada. No obstante, hay casos en los que es necesario capturar la riqueza de la distribución de los datos de entrada, y por lo tanto, interesa que las capas de codificación sean mayores que la capa de entrada.

3.2.2.1. Funciones de error como medida de la reconstrucción

El error de reconstrucción, dependiendo de las características de los datos de entrada, puede ser evaluado desde varios enfoques. Si lo que se quiere evaluar es el grado de semejanza entre los vectores de entrada x y salida z, generalmente la medida de error usada es el error cuadrático medio:

$$L(x, z) = ||x - z||^2 \tag{3.2.25}$$

Si por el contrario, la entrada es interpretada como un vector de probabilidades o un vector binario, la medida de error que mejor se ajusta es la entropía cruzada:

$$L_H(x, z) = -\sum_{k=1}^{d}[x_k log(z_k) + (1 - x_k) log(1 - z_k)] \tag{3.2.26}$$

Donde x_k, z_k corresponden con los valores k-ésimos de los vectores binarios de entrada y salida de longitud d, respectivamente.

En cualquier caso, la codificación de una entrada una vez ajustado el modelo, puede ser vista como una representación distribuida que captura los principales factores de variación de los datos minimizando el límite inferior variacional sobre la probabilidad de un modelo generativo específico. Según [102], esta representación es de alguna manera similar a la obtenida por el Análisis de Componentes Principales (PCA) durante el proceso de proyección sobre los principales factores de variación de los datos. Si el proceso de codificación es llevado a cabo mediante funciones lineales y el modelo se entrena a partir del error cuadrático medio como medida de error, entonces, las k unidades ocultas proyectarán la entrada en el espacio de las primeras k componentes principales de los datos.

3.3. Optimización: aprendizaje basado en gradiente

El procedimiento de optimización (o adquisición de conocimiento) llevado a cabo durante el entrenamiento de una red neuronal (profunda o no) es similar al llevado a cabo en otros modelos lineales basados en descenso de gradiente dentro del área del aprendizaje automático, residiendo la mayor diferencia entre ambos, en las funciones no lineales implícitas que las unidades ocultas o neuronas de las redes implementan, las cuales permiten el modelado de funciones de pérdida más complejas.

Estas funciones, por norma general no convexas, no garantizan la convergencia del modelo hacia una solución óptima (a diferencia de los modelos de regresión lineal o los algoritmos de optimización convexos que garantizan la convergencia a partir de cualquier parámetro inicial) y como hemos señalando en la sección 3.1.1, son muy sensibles a los valores iniciales de los parámetros del modelo desde los que da comienzo la optimización.

Como en cualquier otro problema de optimización, el aprendizaje basado en gradiente intenta minimizar o maximizar una función $J(\theta)$, conocida como función objetivo, función de pérdida o función de error, alterando el valor de θ (siendo θ los parámetros del modelo), en dirección opuesta al gradiente de dicha función ($\nabla_\theta J(\theta)$).

Una vez se ha computado el coste o error cometido por el modelo para uno o varios datos de entrenamiento (batch), haremos uso del algoritmo de retropropagación [178], a menudo simplemente llamado backpropagation, para calcular el gradiente. Es importante destacar, que generalmente, el concepto de retropropagación está mal

empleado, llegando a ser interpretado como el algoritmo de aprendizaje que siguen las redes neuronales. En realidad, la retropropagación se refiere al método con el que se obtiene el gradiente de cualquier función (no solo para aquellas empleadas en el proceso de ajuste de las redes neuronales), mientras que la propagación de ese gradiente y por tanto el aprendizaje, se consigue con otro algoritmo conocido como descenso estocástico del gradiente [85].

Teniendo presente que en las siguientes secciones se presentará un estudio detallado del aprendizaje basado en gradiente, es necesario definir una función de error con la que guiar y facilitar la comprensión del proceso.

Dado que en la mayoría de los casos, nuestro objetivo será obtener una predicción ($f(x)$) a partir de unos determinados datos de entrada al modelo (x), podremos hacer uso del principio de máxima verosimilitud entre la salida de modelo y las etiquetas asociadas a los datos de entrada (y) como medida de error.

Aunque existen otras, una de las aproximaciones de máxima verosimilitud más ampliamente usada es la entropía cruzada. Definida como el logaritmo negativo de la probabilidad, esta medida puede cambiar su forma analítica dependiendo de si el problema es o no binario:

- Si nos encontramos ante un problema de clasificación binaria, el error del clasificador puede ser medido como:

$$J = L_H(y, f(x)) = -ylog(f(x)) - (1-y)(1 - log(f(x))) \tag{3.3.1}$$

siendo $f(x) = p(y = 1|x)$, que generalmente se obtiene como la salida de la única neurona de la capa de salida.

- En el caso de problemas con múltiples clases (>3), la capa de salida contendrá tantas neuronas como clases, lo que requiere el uso de la función softmax para normalizar las probabilidades de cada una de ellas. El error del clasificador se medirá haciendo uso de la probabilidad de pertenencia asignada a la clase correcta ($f(x)_c = p(y = c|x)$). La forma analítica de la función de error será por tanto:

$$J = L_H(y, f(x)) = -log\, f(x)_c \tag{3.3.2}$$

3.3.1. Regla de la cadena

De forma general, en los algoritmos de aprendizaje automático, cuando se hace referencia al uso del gradiente, en realidad se está haciendo referencia al gradiente de la función coste con respecto a los parámetros del modelo $\nabla_\theta J(\theta)$. En este sentido, el procesamiento numérico de este tipo funciones es parte del núcleo del proceso de aprendizaje y por tanto, de vital importancia.

Por otro lado, es importante destacar que en la mayor parte de los problemas de aprendizaje automático, las funciones abordadas son composiciones de dos o más funciones, por lo que el proceso de derivación numérico deberá ser abordado usando la regla de la cadena (eficientemente implementada en términos computacionales en el algoritmo de retropropagación a partir del producto de Jacobianos y gradientes (ecuación 3.3.4)):

- Sean $x \epsilon \mathbb{R}^m$, $y \epsilon \mathbb{R}^n$, $g : \mathbb{R}^m \longrightarrow \mathbb{R}^n$ y $h : \mathbb{R}^n \longrightarrow \mathbb{R}$, donde $y = g(x)$ y $z = h(y) = h(g(x))$. Según la regla de la cadena:

$$\frac{dz}{dx_i} = \sum_j \frac{dz}{dy_j}\frac{dy_j}{dx_i} \tag{3.3.3}$$

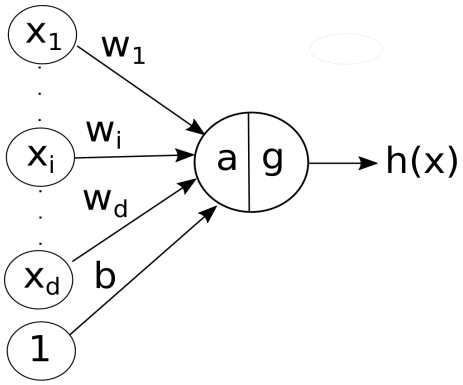


Figura 3.3.1: Arquitectura de una neurona artificial.

que en notación vectorial puede ser reescrita como:

$$\nabla_x z = \left(\frac{\partial y}{\partial x}\right)^\top \nabla_y z \tag{3.3.4}$$

donde $\frac{\partial y}{\partial x}$ es la matriz Jacobiana de g.

3.3.2. Cálculo detallado de los gradientes

Siguiendo la definición dada en la sección 3.1 (ecuación 3.1.1), en la que una red neuronal puede ser vista como una función compuesta, en la que cada capa es una función de la capa anterior y apoyándonos en la Figura 3.3.1, que describe la arquitectura asociada a una neurona artificial, los gradientes en cada capa se pueden calcular a partir del cálculo de las derivadas parciales:

1. **Cálculo de la derivada parcial del error con respecto a la salida de la red:** Teniendo presente que la salida de la red se corresponde con las salidas de las neuronas de la capa de salida (capa $n+1$), la derivada del error con respecto a la salida de la red se abordará considerando $h(x)^{(n+1)} = f(x)$. Siendo, L_H la función de error descrita en la ecuación 3.3.2 e y la etiqueta asociado al vector de entrada, la derivada del error con respecto $f(x)$ queda descrito por:

$$\frac{\partial}{\partial f(x)} L_H(y, f(x)) = \frac{-1}{f(x)} \tag{3.3.5}$$

Abordando el supuesto de un problema con múltiples clases, podemos definir la etiqueta de cada vector de entrada como un vector con un número de posiciones igual al número de clases abordadas en el problema y en el que todas las posiciones contienen ceros excepto la posición que representa la clase a la cual corresponde dicho vector de entrada. Teniendo esta consideración en mente,el gradiente que contiene la derivada parcial de la ecuación 3.3.5 se define como:

$$\nabla_{f(x)} L_H(y, f(x)) = \frac{-e(y)}{f(x)} \tag{3.3.6}$$

siendo $e(y)$ el vector correspondiente a la etiqueta anteriormente descrito.

2. **Cálculo de las derivadas parciales de la salida antes de la activación :** Sabiendo que $h(x)$ (ecuación 3.3.7) se corresponde con la aplicación de una función de activación (sigmide) g sobre la pre-activación $a(x)$

$$h(x) = g(a(x)^{(n+1)}) = g(\sum_i w_i x_i + b) \tag{3.3.7}$$

y teniendo presente que La derivada parcial de la salida con respecto a la pre-activación aplicando la regla de la cadena queda descrita por:

$$\frac{\partial}{\partial a^{(n+1)}(x)} L_H(y, f(x)) = \frac{\partial L_H(y, f(x))}{\partial f(x)} \frac{\partial f(x)}{\partial a^{(n+1)}(x)} \tag{3.3.8}$$

Sabiendo que

$$\frac{\partial f(x)}{\partial a^{(n+1)}(x)} = f(x)(1 - f(x)) \tag{3.3.9}$$

y considerando el valor del gradiente de la ecuación 3.3.5, se obtiene que:

$$\frac{\partial}{\partial a^{(n+1)}(x)} L_H(y, f(x)) = -(1 - f(x)) \tag{3.3.10}$$

Finalmente el gradiente se obtiene reemplazando el vector correspondiente a la etiqueta anteriormente descrito $e(y)$ en la ecuación 3.3.10:

$$\nabla_{a^{(n+1)}(x)} L_H(y, f(x)) = -(e(y) - f(x)) \tag{3.3.11}$$

3. **Cálculo de las derivadas parciales de los parámetros en la capa k-ésima de la red:** Como norma general, los gradientes de los parámetros asociados a cada una de las neuronas de la capa k-ésima se calcularán de manera similar a las señaladas anteriormente aplicando la regla de cadena [94, 85]:

$$\frac{\partial}{\partial h^{(k)}(x)_j} L_H(y, f(x)) = \sum_i \frac{\partial L_H(y, f(x))}{\partial a^{(k+1)}(x)_i} \frac{\partial a^{(k+1)}(x)_i}{\partial h^{(k)}(x)_j} \tag{3.3.12}$$

$$= \sum_i \frac{\partial L_H(y, f(x))}{\partial a^{(k+1)}(x)_i} W_{ij}^{(k+1)} \tag{3.3.13}$$

$$\frac{\partial}{\partial a^{(k)}(x)_j} L_H(y, f(x)) = \frac{\partial L_H(y, f(x))}{\partial h^{(k)}(x)_j} \frac{\partial h^{(k)}(x)_j}{\partial a^{(k)}(x)_j} \tag{3.3.14}$$

$$= \frac{\partial L_H(y, f(x))}{\partial h^{(k)}(x)_j} g'(a^{(k)}(x)_j) \tag{3.3.15}$$

$$\frac{\partial}{\partial W_{ij}^{(k)}(x)_j} L_H(y, f(x)) = \frac{\partial L_H(y, f(x))}{\partial a^{(k)}(x)_i} \frac{\partial a^{(k)}(x)_i}{\partial W_{ij}^{(k)}(x)_j} \tag{3.3.16}$$

$$= \frac{\partial L_H(y, f(x))}{\partial a^{(k)}(x)_i} h_j^{(k-1)} \tag{3.3.17}$$

$$\frac{\partial}{\partial b_i^{(k)}} L_H(y, f(x)) = \frac{\partial L_H(y, f(x))}{\partial a^{(k)}(x)_i} \frac{\partial a^{(k)}(x)_i}{\partial b_i^{(k)}} \tag{3.3.18}$$

$$= \frac{\partial L_H(y, f(x))}{\partial a^{(k)}(x)_i} \tag{3.3.19}$$

Siendo $h^k = g(a^k(x))$, W_{ij}^k y b_i^k los valores de la unidades ocultas tras aplicarle una función activación no lineal, los pesos o valores de las conexiones y el valor de los bías de la capa k-ésima respectivamente, y siendo g' la derivada de primer orden de la función de activación (sigmoide).

Finalmente, cada parámetro del modelo será actualizado en función del valor de su gradiente con respecto a la función de error. Para ello, estas derivadas parciales deben ser generalizadas como vectores de gradientes con los que propagar dichos valores desde la capa de salida hasta la primera capa oculta del modelo.

$$\nabla_{W^{(k)}} L_H(y, f(x)) = \nabla_{a^k(x)} L_H(y, f(x)) h^{(k-1)}(x)^\top \tag{3.3.20}$$

$$\nabla_{b^{(k)}} L_H(y, f(x)) = \nabla_{a^k(x)} L_H(y, f(x)) \tag{3.3.21}$$

$$\nabla_{h^{(k-1)}(x)} L_H(y, f(x)) = W^{(k)\top} \nabla_{a^k(x)} L_H(y, f(x)) \tag{3.3.22}$$

$$\nabla_{a^{(k-1)}(x)} L_H(y, f(x)) \tag{3.3.23}$$

$$= \nabla_{h^{(k-1)}(x)} L_H(y, f(x)) \odot [..., s'(a^{(k-1)}(x)_j), ...] \tag{3.3.24}$$

3.3.3. Variantes del aprendizaje basado en gradiente

Al igual que cualquier otro método de optimización, el aprendizaje basado en gradiente puede ser abordado desde diferentes enfoques, estando cada uno de ellos caracterizado por la forma y la cantidad de datos que se usan en el cálculo del gradiente de la función de coste. Dado el alto número de técnicas existentes en la literatura y dado que nuestro objetivo en este trabajo no es la descripción exhaustiva de todas ellas, en esta sección presentaremos solo la variante que hemos usado y aquellas de las que se deriva.

3.3.3.1. Descenso en gradiente por lotes

El caso más elemental es el descrito por el algoritmo de descenso en gradiente estocástico. En cada una de las iteraciones del algoritmo, se computa el gradiente de la función coste haciendo uso del conjunto completo de datos de entrenamiento. Una vez calculado el gradiente, los parámetros del modelo son actualizados como [85]:

$$\theta = \theta - \eta(\nabla_\theta J(\theta)) \tag{3.3.25}$$

siendo n la tasa de aprendizaje asociada a los parámetros del modelo. La elección del valor de este hiperparámetro es una tarea compleja. Una tasa de aprendizaje demasiado pequeña derivaría en una convergencia extremadamente lenta, mientras que una tasa de aprendizaje demasiado grande podría obstaculizar la convergencia y hacer que la función de pérdida fluctúe alrededor del mínimo o incluso hacer que llegue a divergir. En cualquier caso, el cálculo del gradiente a partir de un conjunto completo de datos, ralentiza el proceso de optimización y podría llegar a ser inabordable para conjuntos de datos demasiados grandes que no pueden ser completamente alojados en memoria.

3.3.3.2. Descenso en gradiente estocástico

A diferencia del escenario anterior, el descenso en gradiente estocástico actualiza los parámetros a partir del gradiente de cada uno de los ejemplos de entrenamiento (x_i, y_i) [85]:

$$\theta = \theta - \eta(\nabla_\theta J(\theta; x_i, y_i))) \tag{3.3.26}$$

La frecuente actualización de los pesos, a menudo de gran variación, además de disminuir el coste computacional, deriva en grandes fluctuaciones de la función de coste, lo que complica bastante la convergencia hacia un óptimo global. Esto hecho, de forma general, conduce a los modelos hacia óptimos locales alternativos.

3.3.3.3. Descenso en gradiente por mini-lotes

Finalmente, el descenso en gradiente por mini-lotes, aprovechando lo mejor de cada una de las técnicas anteriores, actualiza los parámetros del modelo a partir de pequeños subconjuntos de datos (conocidos como mini-batch), reduciendo la varianza de las actualizaciones y haciendo la convergencia más estable [85]:

$$\theta = \theta - \eta(\nabla_\theta J(\theta; x_{i:i+n}, y_{i:i+n}))) \tag{3.3.27}$$

3.3.4. Algoritmos de optimización asociados al descenso en gradiente

Aunque a priori podría parecer que estas estrategias satisfacen los requisitos de convergencia necesarios en la optimización del problema, en la práctica, presentan algunos inconvenientes de especial relevancia que hacen necesario el diseño de nuevas estrategias de optimización [85]:

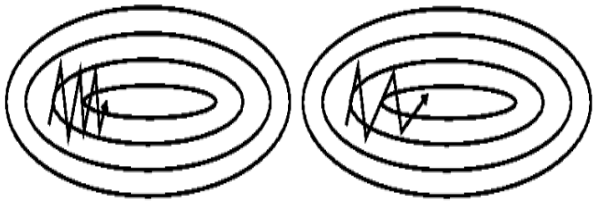
- El éxito de la convergencia está estrechamente relacionado con la naturaleza de los datos. Si estos son escasos y desbalanceados, es posible que la actualización de los parámetros bajo un mismo esquema sea indeseable, siendo necesario definir esquemas ponderados que otorguen actualizaciones de mayor magnitud a los ejemplos atípicos o menos numerosos.

- La naturaleza no convexa de las funciones de error deriva en numerosos puntos de inflexión en los que de forma general, existen valores de error similares. Estas puntos derivan en gradientes prácticamente nulos en cualquier dimensión (dirección), que implican la detención del proceso de aprendizaje ya que se detiene la convergencia.

Al igual que ocurría con las variantes del aprendizaje basado en gradiente, dado el alto número de optimizaciones existentes en la literatura y dado que nuestro objetivo en este trabajo no es la descripción exhaustiva de todas ellas, en esta sección presentaremos solo las variantes más ampliamente usadas y que han supuesto la base de nuestros experimentos.

3.3.4.1. Momentum

Uno de los principales inconvenientes de la optimización basada en gradiente es que la convergencia se ralentiza mucho alrededor de los óptimos locales. Esta situación está



(a) sin momentum (b) con momentum

Figura 3.3.2: Comparativa de la convergencia del algoritmo de descenso en gradiente: a) sin la optimización basada en momentum. b) con la optimización basada en momentum

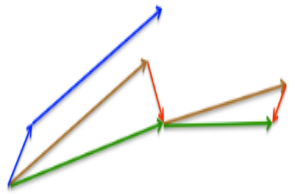


Figura 3.3.3: Descripción gráfica del Gradiente acelerado de Nesterov comparado con la optimización basada en momentum. Los vectores azules corresponden a optimizador basado en momentum. El vector marrón y el vector rojo corresponden a la dirección del gradiente en la iteración anterior y la dirección actual del gradiente, a partir de los cuales se actualiza la dirección final del gradiente. Imagen obtenida de [100]

motivada por la diferencia de pendiente entre dimensiones existente en determinadas áreas de la función de error. La optimización basada en momentum [172] acelera la convergencia y amortigua las oscilaciones (Figura 3.3.2) añadiendo a la información de actualización de los pesos, un porcentaje de la información asociada a la actualización del instante de tiempo anterior (v_{t-1}):

$$v_t = \gamma v_{t-1} \nabla_\theta J(\theta) \tag{3.3.28}$$

$$\theta = \theta - v_t \tag{3.3.29}$$

El término momentum γ (normalmente ajustado a 0.9), el encargado de regular la magnitud del cambio en las actualizaciones en función de la dirección de sus gradientes. Para las dimensiones cuyos gradientes apuntan en la misma dirección que el gradiente del instante de tiempo anterior esta magnitud se ve aumentada, en caso contrario, la magnitud se reduce. Como resultado, se obtiene una convergencia más rápida y una oscilación reducida.

3.3.4.2. Gradiente acelerado de Nesterov

La aceleración de la convergencia y la reducción de las oscilaciones alrededor del óptimo local no siempre se consigue de forma satisfactoria con la inclusión del momentum como optimizador. En algunas ocasiones se necesitan mecanismos capaces de conocer hacia dónde se dirige la convergencia y si es necesario, o no, ralentizar el proceso de optimización [150].

Para ello, es preciso calcular en gradiente del error con respecto a la hipotética posición futura de los parámetros $(\theta - \gamma v_{t-1})$, no con respecto la posición actual:

$$v_t = \gamma v_{t-1} \nabla_\theta J(\theta - \gamma v_{t-1}) \tag{3.3.30}$$

$$\theta = \theta - v_t \tag{3.3.31}$$

A diferencia de la optimización basada en momentum, en la que la magnitud de las actualizaciones se aumenta o disminuye en función de la dirección de los gradientes, el gradiente acelerado de Nesterov actúa del siguiente modo (Figura 3.3.3):

1. Se mueve en la dirección del gradiente acumulado previo (línea marrón)
2. Mide el gradiente actual y realiza una corrección sobre el mismo con respecto a 1 (línea rojo)
3. La dirección en la que se mueve el gradiente final se obtiene a partir de los dos procedimientos anteriores (línea verde), ralentizando y acotando el proceso de convergencia.

3.3.4.3. Adagrad

Una vez el proceso de convergencia ha sido adaptado a la forma de la función de error, el siguiente paso en la tarea de optimización es regular la magnitud de la actualización de cada parámetro en función de su importancia. Esta es precisamente la idea que hay detrás del método Adagrad [62], el cual aplica actualizaciones de pequeña magnitud para parámetros asociados con datos que se producen con frecuencia, y actualizaciones de mayor magnitud para parámetros asociados con datos poco frecuentes.

Para llevar a cabo esta actualización individualizada es necesario calcular el gradiente de la función de error en cada instante de tiempo t con respecto a cada parámetro del modelo i:

$$g_{t,i} = \nabla_\theta J(\theta_{t,i}) \tag{3.3.32}$$

Una vez calculado, cada parámetro i es actualizado como

$$\theta_{t+1,i} = \theta_{t,i} - \eta g_{t,i} \tag{3.3.33}$$

Como se puede apreciar, la magnitud de las actualizaciones se regulan aplicando tasas de aprendizaje (η) dinámicas. Dichas tasas se obtienen a partir de los gradientes previamente calculados para cada parámetro. Cada parámetro i tiene asociada una matriz diagonal $(G_{t,ii})$ en la que cada posición corresponde con la suma de los cuadrados de los gradientes con respecto a dicho parámetro en cada instante de tiempo t. Finalmente, la tasa de aprendizaje en cada iteración se adapta en función del último valor obtenido en la matriz $G_{t,ii}$ como:

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii} + \epsilon}} g_{t,i} \tag{3.3.34}$$

El término ϵ corresponde con un término de suavizado introducido con el objetivo de evitar una posible división por cero.

Aunque este método cuenta con la ventaja de eliminar el proceso de ajuste manual de la tasa de aprendizaje, puede ocurrir que el algoritmo no converja. A medida que la

suma acumulada de gradientes ($G_{t,ii}$) crece con el avance del proceso de entrenamiento, la tasa de aprendizaje disminuye, pudiéndose llegar a detener la convergencia si el valor de los gradientes acumulados es lo suficientemente grande.

3.3.4.4. Adadelta

Adadelta [214] es una extensión de Adagrad que busca reducir el agresivo decremento de la tasa de aprendizaje. A diferencia de *Adagrad* que acumula todos los gradientes asociados a cada parámetro ($G_{t,ii}$), este método, solo acumula una parte de de ellos.

Para ello, define la suma de gradientes como un promedio ponderado entre el gradiente actual (g_t) y un subconjunto de todos los gradientes acumulados hasta el momento, en el que unos gradientes tienen más pesos que otros:

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2 \tag{3.3.35}$$

Siguiendo las ecuaciones 3.3.33 y 3.3.34 y reemplazando la matriz diagonal $G_{t,ii}$ por el $E[g^2]_t$, la actualización de los parámetros bajo el método *Adadelta* queda definido como:

$$\triangle\theta_t = -\frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t = -\frac{\eta}{RMS[g]_t + \epsilon} g_t \tag{3.3.36}$$

Donde $RMS[g]_t + \epsilon$ corresponde con el valor cuadrático medio del gradiente.

Una característica importante a tener en cuenta al usar esta regla es que la magnitud de las actualizaciones no coincide con la magnitud de los parámetros, es decir, el rango de unidades en los que varían las actualizaciones rara vez coinciden con el rango de valores en que se varían los parámetros. Para forzar esta condición, se define un promedio de actualizaciones con el que normalizar la magnitud de ambas medidas:

$$E[\triangle\theta^2]_t = \gamma E[\triangle\theta^2]_{t-1} + (1-\gamma)\triangle\theta_t^2 \tag{3.3.37}$$

Sabiendo que el valor cuadrático medio de este promedio es:

$$RMS[\triangle\theta]_t = \sqrt{E[\triangle\theta^2]_t + \epsilon} \tag{3.3.38}$$

y teniendo presente que este valor solo puede ser aproximado hasta el instante $t-1$, puesto que nos disponemos a calcular el valor de la actualización en t, la regla de actualización de los pesos *Adadelta* queda finalmente descrita como:

$$\triangle\theta_t = \frac{RMS[\triangle\theta]_{t-1}}{RMS[g]_t + \epsilon} g_t \tag{3.3.39}$$

$$\theta_{t+1} = \theta_t + \triangle\theta_t \tag{3.3.40}$$

3.3.4.5. RMSprop

Al igual que *Adadelta*, *RMSprop* es también una extensión de *Adagrad* que busca mejorar el brusco decrecimiento de la tasa de aprendizaje debido a la acumulación de gradientes. Aunque fue desarrollado de forma paralela en el tiempo a *Adadelta*, este método nunca ha llegado a estar publicado, siendo generalmente presentado en seminarios y conferencias por su autor [100].

En esencia, este método es idéntico a *Adadelta*, residiendo su única diferencia en el mantenimiento de la tasa de aprendizaje en la regla de actualización:

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2 \tag{3.3.41}$$

$$\triangle\theta_t = -\frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t \tag{3.3.42}$$

3.3.4.6. Adam

Al igual que *Adadelta* y *RMSProp*, *Adam* [117] es un algoritmo de optimización basado en tasas de aprendizaje adaptativas para cada parámetro. Siguiendo las similitudes con *Adadelta* y *RMSProp*, este método, para cada parámetro, además de almacenar el promedio ponderado del cuadrado de sus gradientes (valor cuadrático medio) (v_t), almacena su promedio ponderado (m_t):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{3.3.43}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{3.3.44}$$

Los valores β_1 y β_2 corresponden con porcentajes de decaimiento que regulan la influencia de los promedios anteriores en el cálculo de una nueva iteración y están inicializados en en intervalo $[0, 1)$. Dado que tanto m_t como v_t, son inicializados al comienzo del proceso de aprendizaje como vectores de ceros, los valores de las medias de los gradientes durante las primeras iteraciones del algoritmo serán próximos a cero, lo que afectará gravemente a la magnitud de las actualizaciones. Para contrarrestar este hecho, se calculan versiones corregidas tanto de tanto $m_t$como v_t:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1} \tag{3.3.45}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2} \tag{3.3.46}$$

Una vez introducido el sesgo hacia magnitudes iniciales diferentes de cero, la regla de actualización de los parámetros, de forma similar a la ya abordada en *RMSprop* y *Adadelta*, se define como:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t \tag{3.3.47}$$

3.3.4.7. AMSGrad

AMSGrad es una extensión del algoritmo Adam en el que, en vez de usar una media ponderada del cuadrado de los gradientes (valor cuadrático medio) (v_t), o una versión corregida de este ($\hat{v}_t$), usa el valor máximo de entre los cuadrados de los gradientes obtenidos hasta el momento:

$$\hat{v}_t = max(\hat{v}_{t-1}, v_t) \tag{3.3.48}$$

De esta forma, AMSGrad reduce los problemas convergencia asociados al uso del valor cuadrático medio en la regla de actualización de los parámetros de los métodos anteriores.

3.4. Redes Neuronales Convolucionales (CNN-Convolutional Neural Networks)

Las Redes Neuronales Convolucionales, también conocidas como Redes Convolucionales o CNNs, son un tipo especializado de red neuronal adaptada al procesamiento

de datos temporal y espacialmente ordenados, siendo capaces de manejar desde series temporales (generalmente dispuestas en vectores de 1-D), hasta imágenes (2-D).

Como su propio nombre indica, estos modelos emplean como función de pre-activación un tipo especializado de operación lineal conocida como convolución. Por tanto, las redes convolucionales son redes neuronales clásicas, en las que en al menos una de sus capas, en vez de la multiplicación general de matrices, implementa la operación de convolución [85, 129].

Definida como un tipo muy general de media móvil, la convolución, es una operación matemática que se aplica sobre dos funciones continuas x y w, generando otra función, s, que representa la magnitud en la que se superponen x y una versión trasladada e invertida de x.

$$s(t) = (x * w)(t) = \int_{-\infty}^{\infty} x(a)w(t-a)da \tag{3.4.1}$$

Teniendo presente que los datos reales que describen un problema, para ser tratados en términos computacionales, deben ser digitalizados, la variable temporal y espacial que los describe pasa a de un dominio continuo a uno discreto, quedando la operación de convolución discreta definida como:

$$s(t) = (x * w)(t) = \sum_{a=\infty}^{\infty} x(a)w(t-a) \tag{3.4.2}$$

Trasladando esta definición al dominio del aprendizaje automático, normalmente, x se asocia con la información de entrada de la capa de convolución, mientras que w, se corresponde con el núcleo (o kernel) de convolución.

Es importante destacar que muy a menudo, los datos, se corresponden con funciones multidimensionales, lo que implica que las operaciones de convolución se llevarán a cabo sobre varios ejes o dimensiones a la misma vez. Este hecho hace necesario el diseño de núcleos de convolución multidimensionales y en consecuencia, la definición de la forma analítica de este tipo de convolución. Como norma general, bastará con incluir una nueva dimensión en su forma analítica y operar sobre ella en todo su eje.

El ejemplo más representativo de convoluciones multidimensionales el es uso de señales bi-dimensionales (imágenes). En este sentido, la convolución se lleva a cabo a partir de un núcleo bi-dimensional que es desplazado a lo largo de ambos ejes (temporal y espacial) [85].

$$S(i,j) = (I * k)(i,j) = \sum_{m}\sum_{n} I(m,n)K(i-m, j-n) \tag{3.4.3}$$

Al igual que ocurre con una red neuronal clásica, las redes convolucionales pueden construirse a partir del apilamiento de múltiples capas de convolución. Cada una de estas capas, toma como entrada un conjunto apilado de características pertenecientes a la salida de la capa inmediatamente anterior, produciendo como salida otro conjunto de características, que puede ser de mayor o menor orden que el de entrada.

De acuerdo con esto, la salida de cada capa de la red puede ser descrita como el resultado un extractor de características (no lineal y espacialmente local), aplicado mediante una ventana deslizante sobre los datos de entrada.

Suponiendo datos de entrada bi-dimensionales, la extracción de características a partir de una capa convolucional se caracteriza por (Figura 3.4.1):

- **Núcleos (o Kernels) de convolución:** siguiendo la nomenclatura usada en dicha figura, cada mapa de características (O_s) está asociado con uno o más

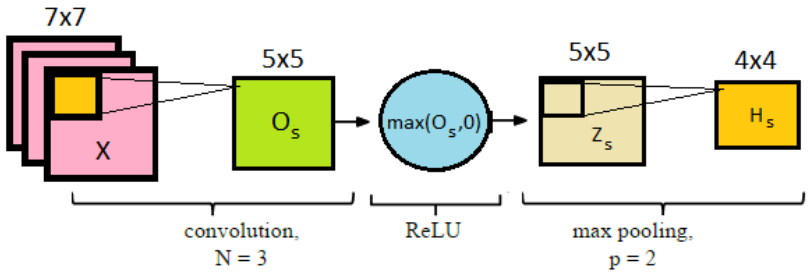


Figura 3.4.1: Descripción de un bloque de una capa de convolución en la que se extrae un solo mapa de características. La entrada, compuesta por un conjunto de datos de dimensión 7x7, se convoluciona con un núcleo 3x3. Una vez vez se ha obtenido el resultado de la convolución, se aplica la operación ReLU seguida de un max-pooling.

núcleos de convolución (también llamados filtros) . Analíticamente, estos mapas de características se obtienen como [129]:

$$O_s = \sum_r W_{sr} * X_r + b_s \tag{3.4.4}$$

Siendo X_r, W_{sr} y b_s, el r-ésimo canal de entrada, el núcleo de convolución para dicho canal y el término bías, respectivamente. Una característica muy importante de las redes convolucionales con respecto a las redes neuronales convencionales, es la drástica reducción de parámetros que éstas necesitan para extraer información relevante. Esta característica se debe al hecho de que los valores del núcleo son compartidos por todos los subconjuntos espaciales de los datos de entrada. Esta reducción de parámetros deriva además, en sistemas equivariantes a posibles traslaciones locales. Teniendo en cuenta que los valores del núcleo se pueden asociar con los pesos de las redes convencionales, la reducción de parámetros minimiza el riesgo asociado al sobreajuste de los modelos. La optimización de dichos parámetros se lleva a cabo a partir de la minimización de una función de error haciendo uso del descenso de gradiente estocástico.

- **Funciones de activación (no lineales):** siguiendo con la similitud con las redes neuronales convencionales, las redes convolucionales también aplican funciones de activación no lineales sobre el resultado de la convolución con el objetivo de obtener transformaciones no lineales de los datos de entrada. Aunque a grandes rasgos, las funciones de activación son compartidas entre ambos modelos, las redes convolucionales incorporan una variante de ellas que de forma general, es una de las más usada. Este tipo de función de activación, conocida como Maxout [86], se basa en el uso de K núcleos de convolución para unos mismos datos de entrada, es decir, la salida del bloque de convolución estará compuesta por K mapas de características diferentes:

$$Z_s = \{O_{Ks}, O_{ks+1}, ..., O_{Ks+k-1}\} \tag{3.4.5}$$

Una vez los K mapas de características han sido procesados, la función Maxout simplemente coge para cada posición espacial (i, j) (i en caso de mapas unidimensionales) el máximo valor de las características obtenidas:

$$Z_{s,i,j} = max\{O_{Ks,i,j}, O_{ks+1,i,j}, ..., O_{Ks+k-1,i,j}\} \tag{3.4.6}$$

- **Max-pooling:** concebida como herramienta para introducir invarianza sobre posibles traslaciones locales, esta función actúa directamente sobre el mapa de características extraído cogiendo el valor máximo de una subregión rectangular dentro del mismo:

$$H_{s,i,j} = \max_{p} Z_{s,Si+p,Sj+p} \tag{3.4.7}$$

  Siendo p el tamaño del subconjunto dentro del mapa de características y S el valor de los incrementos horizontales y verticales con los que delimitar la ubicación del resto de subconjuntos con las que se completará la operación. Nuevamente, si aplicamos la operación sobre una señal unidimensional, los incrementos se harán sobre una sola dimensión. Aplicada sobre el total del mapa de características, la operación de max-pooling se considera un procedimiento de submuestreo, que simplemente reduce el tamaño de dicho mapa [120].

3.5. Regularización

Tal y como introdujimos en la sección 2.2.2.3, cuando un modelo es relativamente complejo con respecto al tamaño de los datos de entrenamiento, lo normal es que termine sobreajustándose. Una técnica bastante extendida para evitar este hecho, es permitir a los algoritmos la preferencia de determinadas soluciones sobre otras dentro del espacio de soluciones. Esta preferencia, conocida como **regularización**, se define como la modificación del algoritmo de aprendizaje con el objetivo de reducir el error de generalización pero no el error de entrenamiento.

A pesar de que en la literatura existen bastantes técnicas con las que regularizar las soluciones de un determinado problema, aquí, solo describiremos algunas de las más usadas, que además, han sido las empleadas en capítulos posteriores durante la fase de experimentación.

Es importante destacar que las técnicas de regularización, tanto las aquí descritas como cualquiera de las existentes en la literatura, pueden ser aplicadas a cualquier arquitectura de red sin ninguna restricción, incluso a modelos inteligentes no neuronales, siempre y cuando basen su aprendizaje en la minimización de una función de error.

3.5.1. Regularización L1 y L2

Aunque existen varias técnicas para expresar preferencias por las diferentes soluciones, una de las más usadas es la incorporación de un término de penalización llamado regularizador (regularizer) en la función de coste o error. En el caso de las regularizaciones L1 y L2 (ecuación 3.5.1), el término de regularización ($R(w)$), actúa sobre el valor de los pesos, penalizando aquellos que tienen valores extremadamente altos y expresando preferencias sobre aquellos que son relativamente pequeños, con el objetivo de lograr fronteras de delimitación entre clases relativamente suaves [85].

$$w^* = \arg\min_{w} \frac{1}{N} \sum_{t} J(f(x_i; w), y_i) + \lambda R(w) \tag{3.5.1}$$

La diferencia entre ambas aproximaciones (L1 y L2) reside en la forma analítica del término. L1 describe la suma de los pesos del modelo ($R(w) = \sum_i |w_i|$) mientras que L2 describe la suma al cuadrado de los mismos ($R(w) = \sum_i w_i^2$). Aunque a grandes rasgos, ambos regularizadores son bastante similares, cada uno de ellos ofrece características de discriminación diferentes.

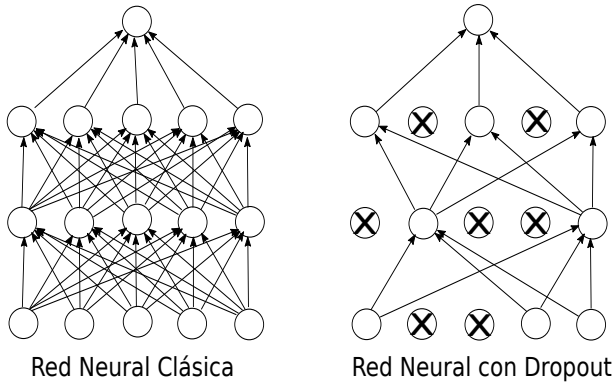


Figura 3.5.1: Regularización basada en dropout. Imagen extraída de [192]

Por un lado, la regularización L1, sitúa el valor de los pesos en una región del espacio de parámetros desde los que pueden ser modelados mediante una distribución laplaciana. Comparada con una distribución Normal, la función de densidad de la distribución laplaciana tiene mayor masa en 0 y en las colas [200], por lo que podemos concluir que L1 tiende a producir pesos grandes o cercanos a cero, fomentando la dispersión de los pesos y actuando de alguna manera como selector de características.

Por otro lado, L2, tiende a expresar preferencias por configuraciones de pesos relativamente pequeños (pero no necesariamente cero), pudiendo ser modelados mediante una distribución gaussiana. Esta característica reduce la dispersión y por tanto la capacidad de selección de características. No obstante, la solución analítica implícita en L2 incrementa su eficiencia computacional frente a L1.

El hiperparámetro λ generalmente es usado como medida de ponderación de la función de regularización.

3.5.2. Dropout

Otras de las técnicas de regularización que mejores resultados han ofrecido en el ámbito de las redes neuronales ha sido el dropout [192]. Para cada uno de los ejemplos de entrenamiento, la activación de cada neurona se asocia a una probabilidad p. Si dicha probabilidad, aleatoriamente escogida, es superior a 0.5, la neurona quedará activa, de lo contrario, esa neurona no tendrá repercusión alguna en ese ejemplo en concreto, y por tanto solo los parámetros de aquellas neuronas que quedaron activas serán actualizados (Figura 3.5.1). Si suponemos que las probabilidades asociadas a la activación de las neuronas siguen una distribución Normal, el 50 % de las neuronas de cada capa se mantendrán inactivas. Este proceso puede interpretarse como el muestreo de diferentes submodelos dentro del propio modelo.

Además, con la desactivación de parte del conjunto de neuronas se evita que el modelo se co-adapte al conjunto de entrenamiento, es decir, se evita que las neuronas sean dependientes del trabajo de neuronas vecinas y las obliga a extraer información útil de forma individualizada.

3.5.3. Early stopping

Durante el proceso de entrenamiento de un modelo, se suele observar que mientras el error de entrenamiento disminuye, el error de validación permanece constante,

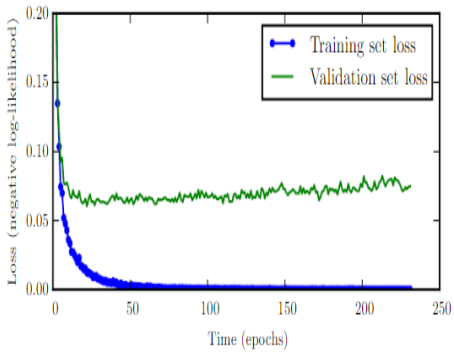


Figura 3.5.2: Curvas de aprendizaje indicando cómo el modelo está siendo sobreajustado. Se puede apreciar cómo el modelo es capaz de memorizar los datos de entrenamiento y diminuir su error durante el entrenamiento, a la vez que pierde capacidad de generalización y aumenta su error durante la validación. Imagen extraída de [85].

oscila o incluso aumenta. Esto se debe a la capacidad de representación del modelo, que es capaz de sobreajustarse a los datos de entrenamiento e incluso memorizarlos, disminuyendo así su capacidad de generalización (Figura 3.5.2).

Para evitar este comportamiento, durante el entrenamiento, se almacenan las configuraciones de los parámetros que mejores resultados de validación ofrecen, de manera que cuando el proceso de entrenamiento finaliza, el mejor modelo, es aquel que mejores resultados ha obtenido y no el último modelo evaluado. Si además de almacenar solo aquellas configuraciones que mejoran el error durante la validación, obligamos al algoritmo a detenerse cuando tras un cierto número de iteraciones previamente especificado, el error de validación no ha mejorado, estaremos regularizando el modelo a partir de una técnica conocida como Early Stopping [169, 37]. Varios son los aspectos de extendido uso:

- Dada su efectividad y su simplicidad, esta técnica no requiere de ningún cambio en el procedimiento de entrenamiento subyacente, por lo que la dinámica de aprendizaje no se ve afectada. A diferencia de otras técnicas como el decaimiento de los pesos, no corre el riesgo de quedar atrapadas en mínimos locales. Además, puede ser usada como complemento de las demás técnicas de regularización encontradas en la literatura agilizando el proceso de entrenamiento, es decir, reduciendo su coste computacional.

- Según [24, 85], restringe el espacio de búsqueda de los parámetros durante el entrenamiento a un volumen relativamente pequeño cercano al valor inicial de los mismos, implicando un efecto similar al obtenido en la regularización L2.

3.6. Conclusiones

En este capítulo hemos introducido de forma teórica los conceptos generales del aprendizaje profundo. Para ello:

- Basándonos en la influencia que los esquemas de inicialización tienen en la convergencia del proceso de optimización durante el aprendizaje (y por consiguiente

en el rendimiento final de los modelos), y en los diferentes problemas que los esquemas clásicos han demostrado tener a medida que incrementamos el tamaño y la profundidad de los modelos, se ha motivado el uso de estrategias de pre-entrenamiento como esquema de inicialización alternativo con el que mejorar la convergencia y como esquema de construcción de DNNs.

- Una vez motivado el uso del pre-entrenamiento, hemos analizado y comparado las aproximaciones (supervisado/ no supervisado) que de dicho pre-entrenamiento se pueden implementar. Para esto, hemos descrito las arquitecturas más ampliamente usadas (DA y RBM) y que son la base de los esquemas de pre-entrenamiento no supervisado. Considerando las ventajas que el pre-entrenamiento no supervisado ofrece frente al pre-entrenamiento supervisado, hemos presentado las diferentes familias de DNNs que a partir de dicho entrenamiento se pueden construir (SDA y DBN).

- Teniendo presente que el aprendizaje de las DNNs está basado en gradientes, hemos detallado su derivación teórica e introducido algunas de las técnicas más ampliamente usadas tanto en la propagación de los propios gradientes como en la optimización de dicha propagación.

- Finalmente, hemos introducido el concepto de red neuronal convolucional como un tipo específico de DNNs (dado su éxito en multitud de áreas dentro del ámbito del aprendizaje automático) y señalado algunas de las técnicas más usadas como estrategias de regularización

Capítulo 4

Modelado de secuencias temporales: Redes Neuronales Recurrentes RNNs

Desde el punto de vista del aprendizaje automático, la detección, la clasificación e incluso la predicción de objetivos a partir de registros continuos de datos, es un problema secuencial que involucra datos de mucha duración que requieren arquitecturas capaces de capturar su evolución temporal.

A diferencia de los problemas clásicos de clasificación, en los que a menudo los datos no siguen ninguna dependencia temporal, en los problemas secuenciales, los datos contienen estructuras temporales latentes que los relacionan y que precisan ser capturadas para dar una solución eficaz al problema.

En este tipo de problemas, el uso de algunas arquitecturas (MLP, SDA, DBN, SVM, RF, entre otras) queda restringido por su propia naturaleza:

- En primer lugar, presentan la limitación de trabajar con datos de entrada o vectores de características de longitud fija. Esta característica complica el uso de corpus de datos compuestos por ejemplos de longitud variable, siendo necesario adaptar todas las entradas a la longitud de la entrada más extensa, incrementando los recursos necesarios para abordar el problema, así como los parámetros a sintonizar del modelo.

- En segundo lugar, presentan la limitación de ofrecer soluciones de longitud fija, es decir, la salida de estas arquitecturas (dada la entrada), es de tamaño fijo. Esta característica restringe completamente este tipo de arquitecturas en problemas como la descripción de imágenes, en las que dado un vector de características de entrada (imagen), el objetivo es obtener una descripción de la escena, que de forma general se compone como una secuencia variable de palabras.

- Finalmente, (a excepción de algunas arquitecturas como los HMM) las arquitecturas clásicas, aunque son capaces de modelar de alguna manera la evolución temporal de los datos, lo hacen de forma general, es decir, no mantienen información de lo ocurrido en el pasado para modelar lo que ocurre en el presente. La falta de "memoria" obliga a los modelos a modelar las dependencias temporales como una característica más, impidiendo capturar su estructura temporal

latente en pequeños intervalos de tiempo.

En este capítulo, nos centraremos en la descripción de un tipo específico de arquitecturas dentro del marco del aprendizaje profundo, capaces de manejar datos complejos y de longitud variable, conocidas como redes neuronales recurrentes (RNN-Recurrent Neural Networks). En las secciones 4.1 y 4.2 comenzaremos describiendo la arquitectura y sus principales características. En la sección 4.3, analizaremos desde un punto de vista teórico, el problema de desvanecimiento y desbordamiento de gradientes que afecta a todos los modelos profundos en general y de forma muy especial a las RNNs. Una vez motivados los problemas de desvanecimiento y desbordamiento de gradientes, en las secciones 4.4 y 4.5 describiremos las arquitecturas RNN-LSTM y RNN-GRU que se han desarrollado para intentar paliarlos.

4.1. Redes Neuronales Recurrentes (RNNs)

Las redes neuronales recurrentes (RNN-Recurrent Neural Networks) son algoritmos de computación bio-inspirados capaces de capturar dependencias temporales entre los datos. Básicamente, una red neuronal recurrente es una red neuronal clásica capaz de operar en el tiempo, en la que la información latente del instante de tiempo t_{n-1} es usada para calcular la información latente en el instante de tiempo t_n.

Dadas sus capacidades de modelado temporal, las redes neuronales recurrentes se han convertido en el estado del arte de las disciplinas en las que los datos se pueden considerar series temporales, como el reconocimiento automático de voz [87], el procesamiento de imágenes o vídeo [89], el procesado de lenguaje natural [198] o traducción automática de textos [196, 45].

Aunque su aplicación dentro del área del remote sensing, no ha sido hasta el momento tan acentuada como en otras áreas, este tipo de arquitecturas están siendo ampliamente usadas en el pronóstico de futuros terremotos [123, 163], en el pronóstico meteorológico, más concretamente en el relacionado con las lluvias [116, 29], en el pronóstico del oleaje oceánico o el cauce de un río [15, 121], en el seguimiento de ciclones [131] o la clasificación de imágenes satelitales [188].

Siguiendo el esquema de la Figura 4.1.1, una RNN mapea una secuencia de entrada $X = x_0, x_1, ..., x_{n-1}$ en una secuencia de salida $Y = y_0, y_1, ..., y_{n-1}$ a partir de transformaciones no lineales de la información de entrada y de la información latente en el instante de tiempo anterior. Al igual que ocurría con las redes neuronales clásicas, las transformaciones no lineales proyectan la información en un nuevo espacio de representación linealmente separable, lo que repercute directamente en un mayor rendimiento en las posteriores tareas de predicción, detección o clasificación. Como se puede apreciar, cada instante t_i se puede considerar como una capa adicional en una red neuronal clásica, con la única diferencia con respecto a a éstas, que los parámetros en las RNNs son compartidos en el tiempo, lo que reduce drásticamente el número final de parámetros a sintonizar.

En su versión más simple (Vanilla), los parámetros θ de una RNN se componen por tres matrices de pesos:

- U que describe los parámetros que relacionan la información latente en instantes de tiempo anteriores con la información de la entrada actual.

- W o matriz de pesos recurrentes, que relaciona la información latente en el instante de tiempo t con la información latente en el instante de tiempo $t-1$

- V en la que se computa la salida de la red en cada instante de tiempo a partir de la información latente anteriormente obtenida.

Varias son las observaciones que se pueden hacer sobre el comportamiento de las RNNs apoyadas en la figura 4.1.1:

- El estado oculto que recoge la información latente en cada instante de tiempo (h_i) puede ser visto como la "memoria" de la red. Como hemos señalado anteriormente, el estado oculto se obtiene a partir de la información latente en el instante de anterior ($h_{(t-1)}$) y la información de la entrada actual ($x_{(t)}$) :

$$h_{(t)} = \sigma(x_{(t)} * U + h_{(t-1)} * W + b) \tag{4.1.1}$$

  siendo σ una función no lineal de activación de la neurona como tanh, sigmoide o ReLU. Es importante destacar que aunque el cálculo de $h_{(t)}$ solo está relacionado con $h_{(t-1)}$, en realidad, de forma indirecta, se está capturando información de todos los instantes anteriores, ya que $h_{(t-1)}$ depende de la información obtenida en $h_{(t-2)}$, que a su vez depende de la información obtenida en $h_{(t-3)}$, así sucesivamente hasta el instante de tiempo inicial. Por lo tanto, la salida del modelo en cada instante de tiempo no solo se basa en la información de entrada actual y la información inmediatamente anterior, sino que de alguna manera se está obteniendo información de salida relacionada con toda la información de entrada anterior.

- Aunque de forma natural las RNNs ofrecen información de salida (y_i) en cada instante de tiempo, éstas pueden ser modificadas en función del problema a abordar, de manera que la información de salida solo se obtenga en el instante de tiempo final (t_{n-1}) y no en cada instante de tiempo. Manteniendo las similitudes con las redes neuronales clásicas, en problemas de clasificación, la salida de las RNNs se obtendrá a partir de la función softmax computada sobre la combinación lineal de la información contenida en los estados ocultos y la matriz de pesos V, con el objetivo de obtener las probabilidades normalizadas de pertenencia a cada clase :

$$y_{(t)} = f(x)_t = softmax(V * h_{(t)}) \tag{4.1.2}$$

El proceso de entrenamiento de una RNN es similar al proceso de entrenamiento de una red neuronal clásica. Ambos se basan en la propagación del gradiente de la función de error con respecto a los parámetros del modelo, pero en el caso particular de las RNN, debido a que los parámetros son compartidos, el gradiente en cada instante tiempo depende no solo de la información actual, sino también de la información relacionada con instantes de tiempo anteriores. Por lo tanto, para propagar el gradiente del error es necesario hacer uso de un método alternativo conocido como propagación del error a través del tiempo (BPTT-Back-Propagation Error Through Time) que será descrito en la sección 4.2 [208].

4.1.1. Extensiones de las RNNs

De una forma similar a lo ocurrido con las redes neuronales clásicas, a lo largo de los años, se han ido desarrollando tipos más sofisticados de RNNs con los que intentar abordar los requisitos de los problemas que se planteaban, así como mejorar las prestaciones del modelo básico Vanilla.

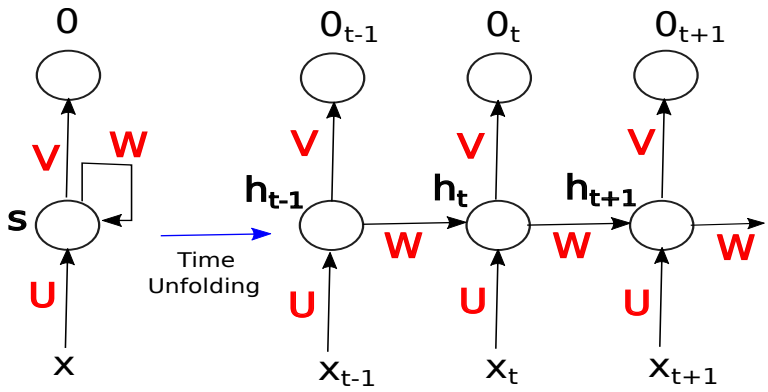


Figura 4.1.1: Descripción de una RNN en su variante más básica (Vanilla)

Aunque podríamos realizar una taxonomía en función del cómputo de la información oculta ($h_{(t)}$), en esta subsección nos centraremos en describir algunas de las propuestas más interesantes relacionadas con la profundidad de los modelos así como con su forma de capturar la dependencia temporal, dejando para futuras secciones la taxonomía de las RNNs en función de su capacidad de memoria.

4.1.1.1. RNNs Bidireccionales

Este tipo alternativo de arquitectura se basa en la idea de que la salida del modelo en el instante de tiempo t_i no solo depende de la información contextual previa, sino también de la información contextual futura [183]. Como se puede observar en la Figura 4.1.2, son arquitecturas bastante simples compuestas por dos capas ocultas que procesan información en sentido opuesto. El resultado final en cada instante de tiempo se calcula a partir de la información latente de ambas capas. Han sido ampliamente usadas en la generación automática de texto y reconocimiento automático de la voz, ya que para predecir una palabra faltante en una secuencia, se estudia tanto el contexto izquierdo como el derecho.

4.1.1.2. RNNs Bidireccionales Profundas

Teniendo presente que las RNNs son redes neuronales clásicas capaces de operar en el tiempo, es inmediato pensar en modelos recurrentes de cierta profundidad. En este sentido, las redes recurrentes profundas son redes recurrentes en las que entre la entrada y la salida se encuentran varias capas ocultas, cada una de ellas conectada de forma recurrente tal y como se describe en la Figura 4.1.1.

Las redes bidireccionales profundas son una extensión de las redes recurrentes profundas y las redes recurrentes bidireccionales (Figura 4.1.3), en las que cada estado oculto está modelado tanto por información contextual anterior y como por información contextual posterior [87]. Aunque en la práctica la profundidad y la bidireccionalidad brindan una mayor capacidad de modelado temporal, lo cierto es que estos modelos necesitan una enorme cantidad de datos con los que ajustar sus parámetros, por lo que su uso se limita a dominios en los que los datos son muy numerosos.

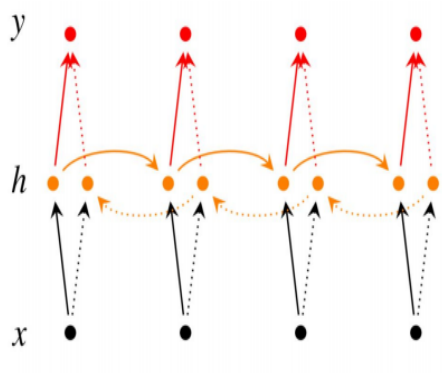


Figura 4.1.2: Descripción gráfica de una RNN bidireccional. Se puede apreciar como la salida de la red en cada instante de tiempo se obtiene a parter de información contextual pasada y futura. Imagen extraída de [1]

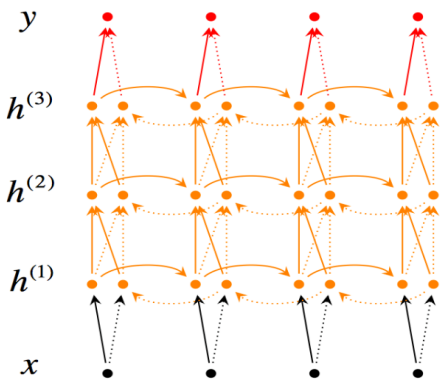


Figura 4.1.3: RNNs bidireccionales profundas. Imagen extraída de [1]

4.2. Descenso en gradiente estocástico y propagación del error a través del tiempo

Las RNNs, al igual que las DNNs, se entrenan a partir de la optimización de una función de error o pérdida. El objetivo por tanto es encontrar los parámetros que minimicen la función de error con respecto a unos determinados datos de entrenamiento. Para ello, es necesario definir una función de error. De forma general, las funciones de error empleadas en el entrenamiento de RNNs son las mismas que las empleadas en las redes clásicas, siendo la más común de entre todas ellas la entropía cruzada entre la salida del clasificador ($f(x)$) correspondiente a la capa softmax y la etiqueta de los datos de entrada (y):

$$E_t = L_{H_t}(y, f(x)) = -log\, f(x)_c \tag{4.2.1}$$

Siendo $f(x)_c = p(y = c|x)$ la probabilidad de que el clasificador asigne a los datos la etiqueta correcta. Esta elección está motivada en la naturaleza misma de la información de entrada, para la que en cada instante de tiempo t_i se dispone de C clases en la que ser clasificada, de manera que el error con respecto a las predicciones queda totalmente recogido a partir de la suma del error en cada instante de tiempo, siendo N el número total de predicciones o instantes de tiempo

$$E = \sum_{t=0}^{N-1} E_t = \sum_t L_{H_t}(y, f(x)) = -\sum_{t=0}^{N-1} log\, f(x)_c \tag{4.2.2}$$

Una vez calculado el error, el objetivo es actualizar los parámetros del modelo en la dirección negativa de su gradiente mediante un descenso estocástico (sección 3.3.3).

Para ello, como hemos señalado anteriormente, durante el entrenamiento de las RNNs se hace uso de una versión ligeramente modificada del algoritmo de propagación hacia atrás conocido como Backpropagation Through Time (BPTT). Dado que los parámetros son compartidos por todos los instantes de tiempo en la red, el gradiente de la salida en cada instante dependerá no solo de la información actual, sino también de la información relacionada con instantes de tiempo anteriores.

Partiendo de la función de error descrita en la ecuación 4.2.2 y de las ecuaciones 4.1.1 y 4.1.2 que describen de forma teórica el cálculo de $h_{(t)}$ e $y_{(t)}$ respectivamente, el cálculo de los gradientes con respecto a los parámetros del modelo se obtiene a partir de la regla de cadena.

El gradiente del error con respecto a la matriz de parámetros V se define como [94]:

$$\frac{\partial E_t}{\partial V} = \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial V} = \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial z_t}\frac{\partial z_t}{\partial V} \tag{4.2.3}$$

Sabiendo que $z_t = V * h_t$ y que $f(x)_t = \sigma(z_t)$, las derivadas parciales se obtienen como:

$$\frac{\partial E_t}{\partial f(x)_t} = -\frac{1}{f(x)_t} \tag{4.2.4}$$

$$\frac{\partial f(x)_t}{\partial z_t} = f(x)_t(1 - f(x)_t) \tag{4.2.5}$$

$$\frac{\partial z_t}{\partial V} = h_t \tag{4.2.6}$$

A partir de estos resultados, la ecuación 4.2.3 puede ser reescrita como:

$$\frac{\partial E_t}{\partial V} = \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial z_t}\frac{\partial z_t}{\partial V} = (f(x)_t - y_t) \otimes h_t \tag{4.2.7}$$

siendo $\oplus$ el producto tensorial o producto de Kronecker. Como se puede apreciar, este gradiente solo depende de la información asociada al instante de tiempo en el que se está calculando, por lo que su cómputo depende de una simple operación matricial.

Desafortunadamente, los gradientes del error con respecto a W y U no se pueden resolver de una manera tan directa, ya que ahora si que se necesita información realcionada con instantes de tiempo anteriores. Dado que el procedimiento de cálculo es el mismo tanto para W como para U, solo describiremos el asociado al gradiente con respecto a W:

$$\frac{\partial E_t}{\partial W} = \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial h_t}\frac{\partial h_t}{\partial W} \tag{4.2.8}$$

El cálculo de $h_t = \sigma(x_{(t)*}U + h_{(t-1)*}W + b)$ depende de W y $h_{(t-1)}$, que a su vez vuelve a ser dependiente de W y $h_{(t-2)}$, lo que hace que dicho gradiente sea dependiente de todos los instantes de tiempo anteriores. Por tanto, para calcular el gradiente, es necesario aplicar nuevamente la regla de la cadena y propagar los gradientes dependientes hasta $t = 0$:

$$\frac{\partial E_t}{\partial W} = \sum_{k=0}^{t} \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial h_t}\frac{\partial h_t}{\partial h_k}\frac{\partial h_k}{\partial W} \tag{4.2.9}$$

Estas dependencias temporales, además de suponer una dificultad durante el entrenamiento y una carga computacional añadida, conducen a los modelos a situaciones de desvanecimiento y desborde de gradientes que interrumpen el aprendizaje [164] y complican la captura de las dependencias temporal a largo plazo, causas que han motivado el diseño de nuevas arquitecturas y nuevos métodos con los que combatirlas.

4.3. Desvanecimiento y desborde de los gradientes

Como hemos descrito anteriormente, las RNNs tienen graves problemas para capturar dependencias temporales a muy largo plazo. Para entender por qué ocurre esto, es necesario revisar detalladamente el cómputo de los gradientes con respecto a los parámetros indirectamente relacionados con la función de error, es decir, las matrices de pesos U (información latente sobre instantes de tiempo anteriores) y W (matriz de pesos recurrentes entre capas ocultas). Dado que ambos parámetros sufren el mismo problema, solo detallaremos el asociado al parámetro W por simplicidad.

Siguiendo las ecuaciones 4.2.8 y 4.2.9 habíamos llegado a la conclusión de que el cómputo del gradiente con respecto a W derivaba en el cómputo de los gradientes dependientes de instantes anteriores. Teniendo presente que estamos derivando una función multivariable con respecto a sus variables, el resultado es una matriz Jacobiana cuyos puntos son las derivadas parciales de dicha función con respecto a cada una de sus variables. El gradiente de la función de error con respecto a W puede ser reescrito como:

$$\frac{\partial E_t}{\partial W} = \sum_{k=0}^{t} \frac{\partial E_t}{\partial f(x)_t}\frac{\partial f(x)_t}{\partial h_t}\left(\prod_{j=k+1}^{t}\frac{\partial h_j}{\partial h_{j-1}}\right)\frac{\partial h_k}{\partial W} \tag{4.3.1}$$

Conociendo que la derivada de la función tanh (o sigmoide) usada como función de activación en cada una de las neuronas tiende a cero a medida que crece o decrece el valor de la entrada (en lo que se conoce como estado de saturación de la neurona), se puede observar cómo a partir de valores pequeños en la matriz Jacobiana y múltiples multiplicaciones asociadas a diferentes instantes de tiempo, los valores del gradiente se reducen exponencialmente, llegando a desaparecer después de unos pocos instantes de tiempo, conduciendo a los gradientes de capas anteriores a 0, y por lo tanto interrumpiendo el aprendizaje de dependencias temporales a largo plazo. Este problema, ampliamente estudiado [164, 182], es conocido como el problema del desvanecimiento del gradiente (*vanishing gradient*).

Aunque este problema se puede encontrar en cualquier modelo de cierta profundidad, es en las arquitecturas recurrentes donde más regularmente se observa dado el rápido incremento de su número de capas encargadas de capturar las dependencias temporales. Algunos de los métodos propuestos para combatir este tipo de situaciones se basan en mejores esquemas de inicialización de los parámetros, el uso de funciones de activación ReLU (dado el carácter constante de su derivada) y el uso de arquitecturas específicas diseñadas explícitamente para combatir este problema como son las RNN-LSTM (LSTM-Long Short Term Memory) [104]y las RNN-GRU (GRU-Gated Recurrent Unit) [45].

Por otro lado, dependiendo de las funciones de activación y de los parámetros del modelo, podría ocurrir que los gradientes tendiesen a valores extremadamente grandes, estando estos siempre asociados a valores de la matriz Jacobiana elevados. En este caso, el problema es conocido como desborde del gradiente (*exploding gradient*) [164]. Es importante destacar, que aunque el desborde del gradiente también interrumpe el aprendizaje, no es tan preocupante como el problema del desvanecimiento:

- En primer lugar, cuando el gradiente se desborda, el valor asociado a los gradientes aparece como NaN (not a number), que a diferencia de lo que ocurre en el problema de desvanecimiento, permite ubicar en que momento se produce el desbordamiento y de alguna manera subsanarlo.

- En segundo, dado que el aprendizaje falla por un exceso de valor de los gradientes, una solución muy simple y efectiva es acotarlos a un umbral predefinido, en lo que se conoce como *norm clipping* [164]. De esta manera, los gradientes seguirán teniendo valores numéricos y en consecuencia, el proceso de aprendizaje no se verá interrumpido.

4.3.1. Verificación del desvanecimiento del gradiente

Comprobar si los modelos, y por tanto, los resultados, están siendo influenciados por el desbordamiento o el desvanecimiento del gradiente es una tarea compleja y en muchas ocasiones inexacta. Dos son los métodos más extendidos:

1. Basándonos en [164], se puede verificar si un modelo está o no siendo influenciado por el desbordamiento/desvanecimiento del gradiente, estudiando la norma de los gradientes durante la etapa de entrenamiento. Para ello, sabiendo que el gradiente en un determinado instante de tiempo se corresponde con la suma de los gradientes en los instantes de tiempo anteriores

$$\frac{\partial E}{\partial \theta} = \sum_{1 \leq t \leq T} \frac{\partial E_t}{\partial \theta} \tag{4.3.2}$$

$$\frac{\partial E_t}{\partial \theta} = \sum_{1 \leq k \leq t} \left(\frac{\partial E_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial \theta} \right) \tag{4.3.3}$$

donde $\frac{\partial h_k}{\partial \theta}$ corresponde con la derivada parcial en el estado h_k con respecto a θ , se puede comprobar que:

$$\frac{\partial h_t}{\partial h_k} = \prod_{t \geq i > k} \frac{\partial h_i}{\partial h_{i-1}} = \prod_{t \geq i > k} W^T diag(\sigma'(h_{i-1})) \tag{4.3.4}$$

donde $diag(\sigma'(h_{i-1}))$ corresponde con la matriz diagonal de la derivada de la función de activación del estado h_{i-1}. Teniendo presente que $|x * y| \leq |x| * |y|$ siendo x e y dos variables aleatorias, la norma del Jacobiano $\frac{\partial h_t}{\partial h_k}$ queda acotada por el producto de la norma de las matrices W^T y $diag(\sigma'(h_{i-1}))$:

$$\forall k, \parallel \frac{\partial h_t}{\partial h_k} \parallel \leq \parallel W^T \parallel \parallel diag(\sigma'(x_{i-1})) \parallel \tag{4.3.5}$$

Por otro lado, se puede probar analíticamente que la norma de W corresponde con el radio espectral λ_1 o valor absoluto del mayor de los autovalores de la matriz W^T, que a su vez está acotado por un valor $\gamma \epsilon \mathbb{R}$. Por tanto,

$$\nabla k, \parallel \frac{\partial h_t}{\partial h_k} \parallel \leq \parallel W^T \parallel \parallel diag(\sigma'(h_{i-1})) \parallel < \lambda_1 < \frac{1}{\gamma} \tag{4.3.6}$$

Más concretamente, si $\gamma > 1 \Longrightarrow \lambda_1 < 1$, lo que resultaría en situaciones de desvanecimiento de gradiente. De forma contraria, el desbordamiento del gradiente vendría dado por situaciones en los $\lambda_1 > 1$.

2. Comprobando implícitamente el valor de los gradientes propagados. Para ello, se analizan los valores de los gradientes propagados durante la etapa de entrenamiento. Por un lado, se considera que un modelo está siendo afectado por situaciones de desbordamiento cuando el gradiente propagado supera un determinado umbral o su valor corresponde con NaN. Por otro lado, se considera que un modelo está siendo afectado por situaciones de desvanecimiento de gradiente, cuando el 25 % de los gradientes propagados son próximos a $0 + \varepsilon$, siendo $\varepsilon = 10^{-8}$.

4.4. RNN-LSTM (Long Short Term Memory)

En la sección 4.3 se han introducido algunos de los mecanismos propuestos en la literatura para combatir el problema del desvanecimiento y desbordamiento de los gradientes. En el caso del desbordamiento, el mecanismo más ampliamente usado es el reescalado de la norma del propio gradiente una vez se ha superado un cierto umbral [164] . En el caso del desvanecimiento, se opta por el uso de términos de regularización que representan la preferencia por determinados valores de parámetros que ni incrementan ni disminuyen la magnitud de los gradientes [164].

Aunque analíticamente ambos métodos son relativamente simples, la búsqueda de un umbral adecuado en el caso del desbordamiento y la relativamente alta carga computacional asociada a la regularización de los parámetros en el caso del desvanecimiento, puso de manifiesto la necesidad de diseñar nuevas arquitecturas como redes

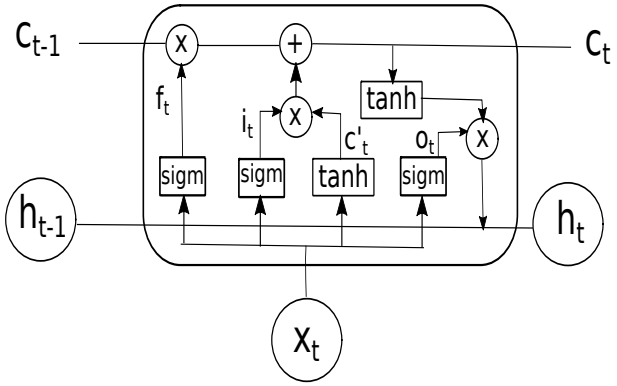


Figura 4.4.1: Descripción gráfica de la arquitectura LSTM

neuronales recurrentes en su variante LSTM, que son mucho menos sensibles a estos problemas.

En consecuencia con lo anteriormente expuesto, la Clasificación Temporal Conexionista (CTC) basada en RNN-LSTM [88], aprovechando la existencia de células de memoria más sofisticadas, permite que los gradientes fluyan sin interrupciones durante largos periodos de tiempo, identificando de forma robusta patrones de alto nivel y haciendo factible la implementación de un sistema de reconocimiento en el que no existen configuraciones híbridas, es decir, configuraciones compuestas por varias arquitecturas de diferentes características. Esta última apreciación es muy importante, ya que el enfoque híbrido necesita entrenar por separado cada uno de los módulos desde diferentes criterios, ajustando un costoso hiperparámetro adicional en cada etapa de entrenamiento y llegando a no encontrar una solución óptima con la que resolver la tarea final.

4.4.1. Arquitectura LSTM

Una RNN-LSTM [103] es básicamente una RNN clásica (Vanilla) en la que el cálculo del estado oculto h_t se obtiene a partir de un mecanismo de puertas (*gating*) que permite que la información fluya o quede retenida en el propio estado (célula de memoria) , capturando así las dependencias temporales a largo plazo. Al aprender los parámetros de sus puertas, la red aprende cómo debería de comportarse su memoria.

Descrita en la Figura 4.4.1, la arquitectura LSTM consta de:

- Tres puertas, i_t, f_t y o_t, conocidas como puerta de entrada, puerta de olvido o reseteo y puerta de salida. El objetivo de las puertas de entrada y de reseteo es regular que cantidad de información relacionada con la entrada y que cantidad de información relacionada con el estado anterior será usada para calcular el estado oculto del instante de tiempo actual. La puerta de salida, en cambio, regula que cantidad de información relacionada con el estado interno será expuesta para el cálculo del estado en el siguiente instante de tiempo. De forma analítica, el estado de estas puertas se define como:

$$i_t = \sigma(x_t * U^i + h_{t-1} * W^i) \tag{4.4.1}$$

$$f_t = \sigma(x_t * U^f + h_{t-1} * W^f) \tag{4.4.2}$$

$$o_t = \sigma(x_t * U^o + h_{t-1} * W^o) \tag{4.4.3}$$

Siendo U^i, W^i, U^f, W^f, U^o, W^o las matrices de pesos asociadas a cada una de las puertas y σ la función sigmoide.

- Un estado oculto provisional o candidato, $\tilde{c_t}$, qué corresponde con el estado oculto de una RNN clásica, calculado a partir de la información de entrada actual y del estado oculto inmediatamente anterior. La información relacionada con este candidato será posteriormente utilizada para calcular el estado oculto final.

$$c'_t = tanh(x_t * U^c + h_{t-1} * W^c) \tag{4.4.4}$$

U^c, W^c corresponden nuevamente con las matrices de pesos asociadas al cálculo del estado oculto provisional.

- Una célula de memoria interna, c_t, que relaciona la información contenida en la célula de memoria del instante de tiempo anterior y la información de entrada. De entre sus opciones, esta célula puede descartar por completo la información de la célula anterior, descartar por completo la información actual entrante, o lo más probable, combinar la información de ambas.

$$c_t = c_{t-1} * f_t + c'_t * i_t \tag{4.4.5}$$

- Un estado oculto, h_t, que corresponde con el estado oculto de la arquitectura y que finalmente se obtiene a partir de la información contenida en la celda de memoria y de la información expuesta por la puerta de salida anteriormente descrita.

$$h_t = tanh(c_t) * o_t \tag{4.4.6}$$

4.5. RNN-GRU(Gated Recurrent Unit)

Pese a los buenos resultados obtenidos por los modelos RNN-LSTM, su alto coste computacional y su dilatado tiempo de entrenamiento han obligando a los investigadores a desarrollar métodos alternativos con los que seguir haciendo de este tipo modelos, modelos dinámicos y competitivos. Algunas de las alternativas propuestas fueron: la parametrización de los datos/modelos en múltiples GPUs [92, 196] y la creación de esquemas de inicialización específicos para los pesos de las conexiones recurrentes [127]. No obstante, en determinadas situaciones, los modelos RNN-LSTM seguían sufriendo procedimientos de entrenamiento muy costosos e intensivos que los dejaban en clara desventaja frente a otras técnicas.

Considerados como una variante más simple de los RNN-LSTM pero que comparten prácticamente todas sus propiedades, los modelos RNN-GRU han llegado a ser el estado del arte en la resolución de multitud de problemas de modelado secuencial. Su principal ventaja es la competitividad, ya que la reducción de la carga computacional debido al diseño de un mecanismo de memoria menos sofisticado les permite modelar dependencias temporales a largo plazo a la vez minimizan el tiempo de entrenamiento.

Descrita en la Figura 4.5.1 , la arquitectura GRU consta de:

- Dos puertas, r_t y z_t, conocidas como puerta de reinicio (combinación de las puertas de entrada y reseteo de la arquitectura LSTM) y de actualización, respectivamente. El objetivo de la puerta de reinicio es determinar cómo se combina

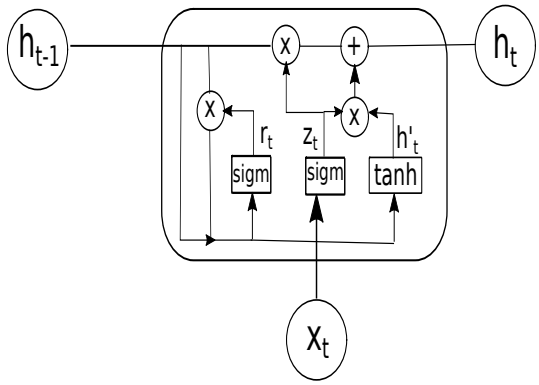


Figura 4.5.1: Descripción gráfica de la arquitectura GRU

la información entrante con la información contenida en la memoria del estado anterior. La puerta de actualización, en cambio, determina que cantidad de información relacionado con el estado anterior se debe conservar en este nuevo estado. De forma analítica, el estado de las puertas se define como:

$$r_t = \sigma(x_t * U^r + h_{t-1} * W^r) \tag{4.5.1}$$

$$z_t = \sigma(x_t * U^z + h_{t-1} * W^z) \tag{4.5.2}$$

Siendo U^r, W^r, U^z, W^z las matrices de pesos asociadas a las puertas.

- Un estado oculto provisional o candidato, $\tilde{h_t}$, que al igual que ocurría con la arquitectura LSTM, corresponde con el estado oculto de una RNN clásica, calculado a partir de la información de entrada actual y del estado oculto inmediatamente anterior. La información relacionada con este elemento será posteriormente utilizada para calcular el estado oculto final.

$$h_t^{'} = tanh(x_t * U^h + (h_{t-1} * r_t) * W^h) \tag{4.5.3}$$

- Un estado oculto, h_t, que corresponde con el estado oculto de la arquitectura y que finalmente se obtiene a partir de la información correspondiente al estado anterior y la contenida en el estado provisional correspondiente a la información actual entrante.

$$h_t = (1 - z_t) * h_{t-1} + z_t * h'_t \tag{4.5.4}$$

4.6. Conclusiones

En este capítulo hemos introducido las redes neuronales recurrentes como herramientas de modelado secuencial de series temporales y hemos descrito de forma teórica sus conceptos generales. Teniendo presente la profundidad que estos modelos alcanzan durante el entrenamiento, hemos descrito desde un punto de vista numérico, los problemas de desvanecimiento y desbordamiento de gradientes que tan negativamente afectan en el proceso de aprendizaje. Finalmente, hemos presentado dos variantes de la arquitectura neuronal recurrente clásica, LSTM y GRU, con las que a partir de mecanismos de memoria más sofisticados se pueden modelar dependencias temporales a largo plazo y combatir el problema del desvanecimiento/desbordamiento de gradientes.

Parte III

Implementación de sistemas de reconocimiento automático de señales sismo-volcánicas basados en técnicas de Deep Learning

Capítulo 5

Origen y adquisición de los datos: volcanes de Colima e isla Decepción

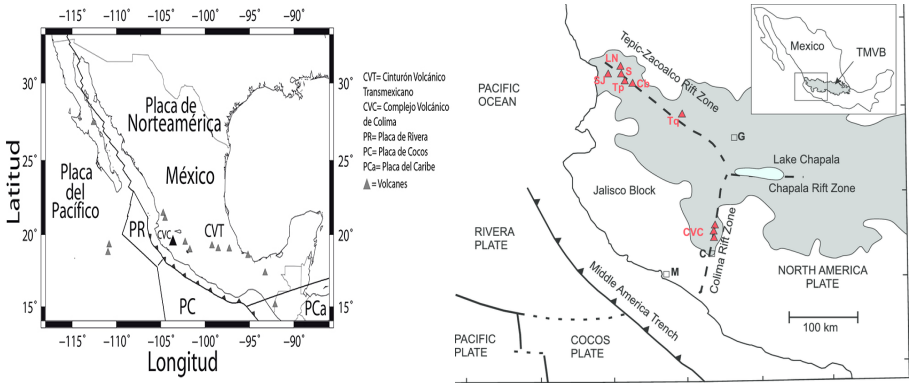
La elección de los registros que representarán cada uno de los eventos en el corpus de datos, es una tarea de vital importancia, ya que de ellos dependerá la capacidad de generalización y el correcto funcionamiento de los sistemas. Teniendo presente las particularidades de las señales en el marco de nuestro problema, y dado el alto coste temporal y humano necesario, la construcción de bases de datos muy grandes es todavía un reto. Por lo tanto, pese a que el tamaño de las bases de datos supone un grave problema en la consecución de nuestro objetivo (pues en su gran mayoría los conjuntos de datos sufren del problema de la incompletitud) se opta por construir bases de datos pequeñas pero muy fiables, que permitan construir sistemas robustos (aunque un poco más simples).

En este capítulo describiremos los registros que conforman nuestros corpus de datos y que fueron recogidos de los volcanes de Colima (México) e Isla Decepción (Antártida) respectivamente. Pese a que en la sección 1.2.1 se describen de forma general los eventos sismo-volcánicos más representativos, sabiendo que cada volcán sigue sus propias dinámicas y que los mecanismos fuente que generan los eventos pueden venir derivados por diferentes procesos físicos, se hace imprescindible la descripción detallada del marco tectónico de cada volcán y las características asociadas a sus eventos.

5.1. Volcán de Fuego de Colima, México

Junto a los volcanes de el Cántaro y el Nevado, el volcán de Fuego forma el Complejo Volcánico de Colima (CVC) (Figura 5.1.1). Esta cadena de estratovolcanes andesíticos, según [5], tuvo el comienzo de su actividad volcánica hace 1.7 Ma (millones de años), con la formación del volcán de el Cántaro.

Formado como consecuencia de la subducción de las placas de Cocos (PC) y Ribera (PR) bajo la placa Norteamericana, el Eje Volcánico Transversal o Cinturón Volcánico Trans-Mexicano (CVTE), es una zona de gran actividad sísmica en la que además del CVC se encuentran otros muchos volcanes activos como el Popocatépelt e Iztaccíhuatl, el Malinche, el Nevado de Toluca, el pico de Orizaba, etc. En concordancia con [12, 136],



(a) Complejo Volcánico de Colima. (CVC). (b) Marco sismo-tectónico del CVC.

Figura 5.1.1: a: Ubicación del Complejo volcánico de Colima (CVC) dentro del Cinturón Volcánico Trans-mexicano (CVTE). Imagen extraída del centro universitario de estudios e investigaciones de vulcanología de la Universidad de Colima b: Interpretación del CVC como área asociada con la fragmentación del bloque de Jalisco producida por el choque de tres grandes depresiones tectónicas o rifts. Imagen extraída de [56]

el CVC puede ser interpretado como un área asociada a la fragmentación del Bloque de Jalisco desde la placa Norteamericana a lo largo del graben (fosa tectónica) de Tepic-Zacoalco con una dirección NW-SE, y del graben de Colima con una orientación N-S. Dentro de este marco tectónico, [58, 79] sugieren que la cercanía de la falla de Tamazula y el sistema regional de fallas vecino han supuesto un factor importante en el proceso de migración de vulcanismo hacia el sur, siendo el volcán de Fuego, situado a 30 km de la ciudad de Colima, el más activo y a la vez el más austral del CVC. Este sistema de fallas, también ha sido asociado a la historia eruptiva del CVC, en la que cíclicamente se desarrollan grandes estratovolcanes que posteriormente colapsan formando extensos depósitos de escombros.

Considerado como el volcán más activo de Norteamérica, el volcán de Fuego ha experimentado 50 episodios eruptivos en los últimos 450 años. Tanto es así que en 2010,

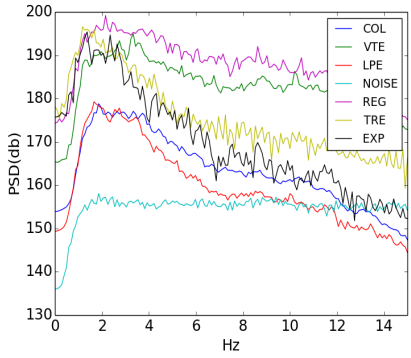


Figura 5.1.2: Contenido espectral promedio por tipo de evento. Los datos corresponden a las señales filtradas entre 1 y 25 Hz.

tuvo el índice de explosividad, asociado a la alerta por una posible erupción, más alto de todos los volcanes de México (*Volcano Explosivity Index-VEI* $\geq$4), prediciéndose así una nueva erupción en los siguientes 20 años. Esta nueva erupción no se hizo esperar, y solo cuatro años después, en el 2014, tuvo lugar.

Históricamente, este volcán ha sufrido grandes erupciones plinianas, siendo las acontecidas en 1818 y 1913 las dos últimas de mayor envergadura registradas. Como hemos señalado anteriormente, este tipo de erupciones generan domos muy altos que terminan siendo derrumbados por el contexto sismo-tectónico. En este sentido, no fue hasta la década de los 90s, cuando acompañado de las mejoras de los sensores sísmicos y de nuevas técnicas de monitoreo, se observó que las dinámicas del volcán generaban enjambres de LPEs y VTEs como episodios pre-eruptivos, que precedieron el derrumbe del domo y la consecuente generación de flujo piroclástico en la erupción de 1991, así como la explosión del 21 de julio de 1994. Este ciclo eruptivo se volvió a repetir años más tarde, cuando en 1997 el domo comenzó de nuevo a crecer, y colapsó en 1999 debido a las explosiones registradas.

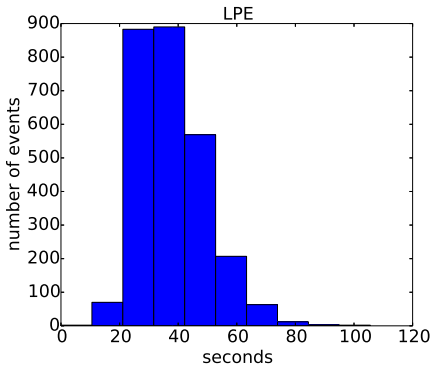
El ciclo eruptivo más importante registrado en este último siglo comenzó en 2004 y se extiende hasta nuestros días. El día 25 de septiembre de 2004, una sucesión de LPEs dio lugar a un incremento de actividad que desde entonces ha venido acompañada de derrumbes, flujos piroclásticos y explosiones con expulsión de material asociado a alturas de hasta 10 km cuya caída de ceniza afecta a poblaciones en un radio de 12 Km [12]. Desde finales de 2013, el volcán ha experimentado un incremento en la agresividad de las erupciones, llegándose a registrar caída de cenizas volcánicas asociadas a las explosiones en un radio de 25 Km, lo que ha obligado a evacuar en algunas ocasiones a ciertas poblaciones cercanas al volcán.

5.1.1. Eventos de el volcán de Fuego de Colima

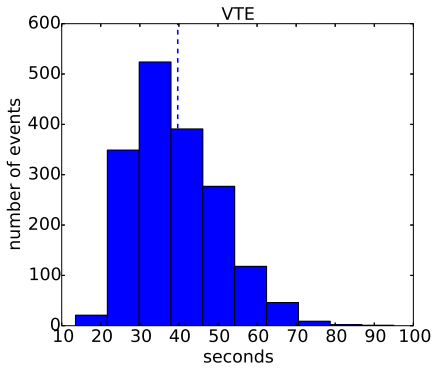
Los eventos que componen el corpus de datos de Colima corresponden a los episodios eruptivos de 1994, 1995, 1998 y 2005. Siguiendo las directrices propuestas en [207], los eventos sismo-volcánicos del volcán de Fuego pueden ser clasificados en función de su forma de onda y su contenido espectral a la vez que asociados a potenciales mecanismos fuente como sigue:

- **Eventos de largo período (LPE):** son señales casi monocromáticas con una banda de frecuencia estrecha centrada, en la mayoría de los casos, entre 1 y 6 Hz (Figura 5.1.2). Su modelo fuente está asociado a modos volumétricos de deformación del medio de propagación como el crecimiento del domo, llegando también a preceder episodios explosivos en determinadas etapas eruptivas como la registrada en 2005. Dados los diferentes mecanismos fuente con los que estos eventos pueden ser generados en este volcán, es posible encontrar diferentes tipos de estos eventos, atendiendo cada uno de ellos a diferentes criterios espectrales y frentes de onda. Observando la Figura 5.2.3a se puede ver que su duración media está aproximadamente en 35 segundos.

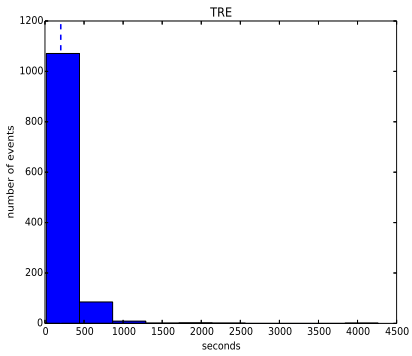
- **Terremotos volcano-tectónicos (VTE):** suelen estar relacionados con fragmentaciones del edificio volcánico debido a los esfuerzos ejercidos por la migración de fluidos (agua, gas o magma). La consecuencia de esta fragmentación del medio es la generación de las ondas sísmicas P y S, que aparecen en los sismogramas bien diferenciadas. La energía espectral está bastante distribuida, llegando a encontrar frecuencias de hasta 40 Hz (Figura 5.1.2). Generalmente, sus codas



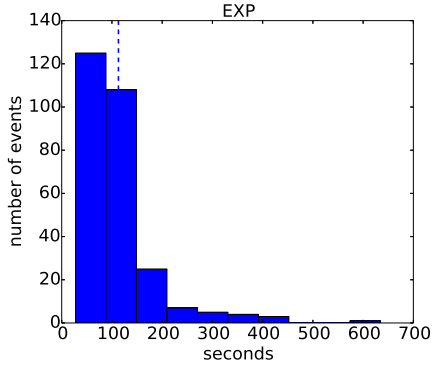
(a) Eventos de largo período



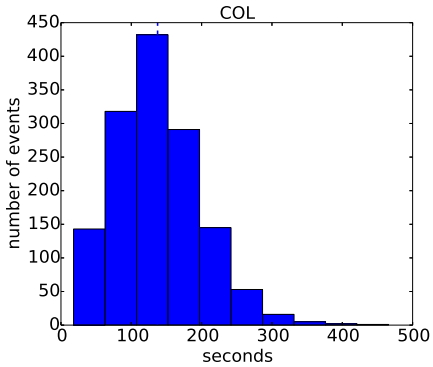
(b) Terremotos volcano-tectónicos



(c) Tremors volcánico

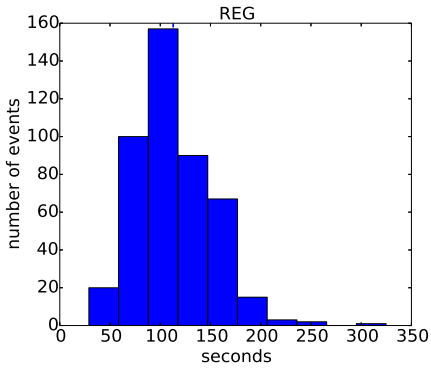


(d) Explosiones

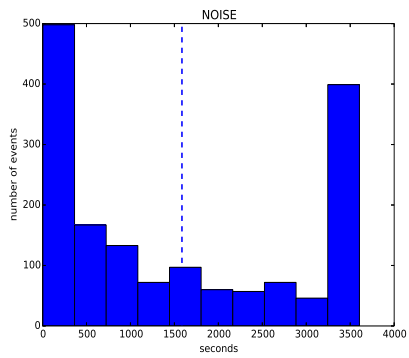


(e) Coladas

Figura 5.1.3: Histogramas de duración en segundos por tipo de evento de la base de datos del volcán de Colima.



(f) Terremotos regionales



(g) Ruido volcánico

Figura 5.1.3: Histogramas de duración en segundos por tipo de evento de la base de datos del volcán de Colima.

decaen exponencialmente. La duración media de este tipo de eventos es de 40 segundos (Figura 5.1.3b).

- **Tremors volcánicos (TRE):** están relacionados con cambios de presión en los fluidos presentes en la cámara magmática o incluso en las grietas del edificio volcánico. Su rango espectral se encuentra entre 1 y 10 Hz (Figura 5.1.2), por lo que en algunos casos se puede considerar ruido volcánico. De forma regular, estos eventos vienen descritos por señales armónicas con amplitud y duración muy variable (de minutos a horas o incluso meses). En muchas ocasiones, los tremors asociados al volcán de Fuego se suelen subclasificar en diversos tipos:

  - **Tremor armónico o resonante:** la forma de onda del sismograma está modulada por varias frecuencias dominantes o resonantes, en los que se pueden diferenciar de forma clara los armónicos. Una característica importante a tener en cuenta en este tipo de tremors es que las diferentes frecuencias resonantes van variando su posición en el espectro debido a los cambios de presión en las cavidades resonantes.
  - **Tremor espasmódico:** este tipo de tremor es comúnmente asociado con la concatenación de múltiples eventos de largo período, por lo que su rango de frecuencias llega incluso a los 15 Hz. A diferencia del tremor armónico, su amplitud es muy variable y no tiene frecuencias dominantes claras. De forma más o menos generalizada, estos tremors han sido registrados en la fase terminal de los episodios explosivos del volcán y durante episodios emisivos en los que se desprende gas y cenizas.
  - **Tremor pulsante:** caracterizado por la superposición de un tremor espasmódico y unos pulsos energéticos que secuencialmente se repiten en intervalos de entre 10 y 30 segundos. Su rango espectral se extiende hasta los 5 Hz.

- **Explosiones (EXP):** precedidas por un LPE de corta duración, son señales de alta frecuencia con picos espectrales ubicados en diferentes frecuencias desde los

4 hasta los 20 Hz (Figura 5.1.2). En este volcán, las explosiones están presentes prácticamente en todos los períodos activos, siendo más notable su registro cuando el domo se enfría, actuando como tapón y la presión bajo él comienza a destruirlo [12]. Regularmente se estructuran en dos fases de diferenciado contenido espectral que a menudo generan y se solapan con tremors espasmódicos y derrumbes. En primer lugar se observa una primera llegada poco energética relacionada con la profundidad y la fuerza de la explosión. Seguidamente, se registra una segunda llegada más energética y de mayor contenido espectral que se suele relacionar con la fragmentación del domo [202, 217]. La duración media es aproximadamente de 150 segundos (Figura 5.1.3d).

- **Coladas y colapsos de rocas (COL):** como hemos citado anteriormente, el crecimiento del domo en el interior del cráter termina con un desbordamiento de material por los flancos del edificio volcánico. Este desbordamiento genera flujos piroclásticos que han llegado a recorrer hasta 5 Km en episodios eruptivos como los del 1991, 1998 y 2005. El rango espectral va desde los 5 a los 15 Hz (Figura 5.1.2) y su duración media es de 150 segundos pudiendo llegar hasta los cinco minutos (Figura 5.1.3e).

- **Terremotos regionales (REG):** son sismos tectónicos de origen no volcánico que pueden ocurrir en cualquier parte de la tierra siempre y cuando haya suficiente energía de deformación elástica almacenada en los bordes de las placas o en las fallas cercanas. Normalmente tienen una duración media de 120 segundos, observando una diferencia en la llegada de las fases P y S de decenas de segundos, ya que en la mayoría de los casos provienen del CVC y de la zona de subducción de Colima, situada a varias decenas de kilómetros (Figura 5.1.3f). Su coda experimenta una caída exponencial en sus frecuencias más energéticas.

- **Ruido volcánico (NOISE):** cualquier evento que no pueda ser asociado con los ya descritos es considerado ruido volcánico. Generalmente aparece superpuesto sobre el resto de señales sísmicas, pero en ausencia de ellas, se puede caracterizar por una baja amplitud originada por múltiples fuentes naturales y artificiales. Como fuentes naturales, podemos mencionar el viento, la variación de la presión atmosférica o la lluvia. En el caso de las fuentes artificiales, este ruido se conoce como "ruido cultural" y es introducido principalmente por las poblaciones cercanas y la actividad humana. De forma generalizada, el ruido, ubicado entre 1 y 15 Hz interfiere en el rango de frecuencias en el que se encuentra la mayor parte del contenido espectral volcánico (Figura 5.1.2). Su duración media es de 1500 segundos (Figura 5.1.3g).

Teniendo presente el relativamente pequeño conjunto de datos del que disponemos (9932), se ha optado por incluir en una misma clase todos los eventos tipo TRE con el objetivo de encontrar características discriminativas con los que poder clasificarlos. De otra manera, el número de TREs por clase no serían suficientes como para encontrar una buena caracterización. Dicho esto, la distribución final de eventos por clase resulta en: **1738 VTE, 2699 LPE, 1170 TRE, 455 REG, 1406 COL, 278 EXP y 1586 NOISE.**

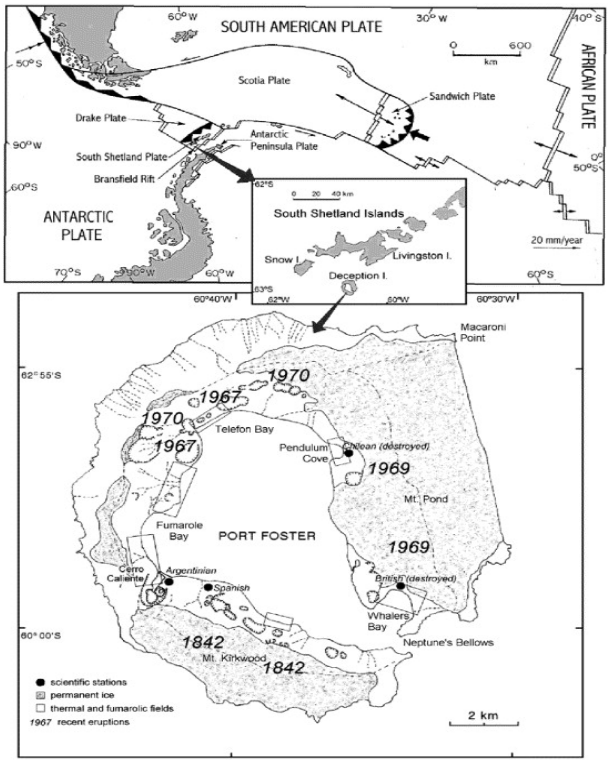


Figura 5.1.4: Completo tectónico asociado al volcán de Isla Decepción. Imagen extraída de [30]

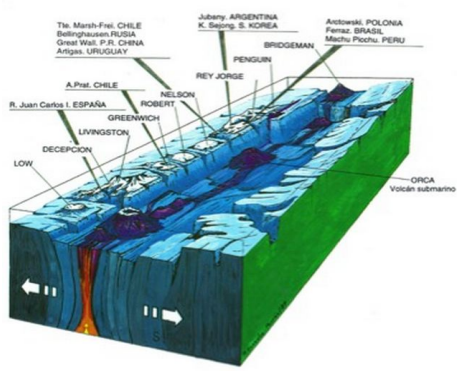


Figura 5.2.1: Fosa oceánica del Bransfield. Imagen extraída de [84]

5.2. Volcán de Isla Decepción, Antártida

La isla Decepción, perteneciente al archipiélago Shetland del Sur, en la Antártida, y situada en el estrecho de Bransfield, al noroeste de la península Antártica (62º 59 'S, 60º 41' W), es la cima de un cráter volcánico en la que tras el derrumbe de parte de la caldera, se introdujo agua del mar de la Flota. El volcán, de cuya erupción en el período cuaternario emergió la isla, se encuentra a 850 m bajo el nivel del mar y tiene un diámetro basal de entre 25-30 Km. Considerado como uno de los tres volcanes antárticos (junto al monte Erebus y la isla Buckle) más activos, ha entrado en erupción al menos seis veces en los últimos 200 años, siendo registradas las dos últimas en 1967 y 1970, que llegaron a destruir algunas de las bases militares allí presentes.

Actualmente, la parte emergida del volcán donde se encuentra la isla, se eleva 1400 metros por encima del fondo marino, siendo el Mt. Pond el punto más alto a 540 metros sobre el nivel del mar. Su forma de herradura lo convierte en el único volcán navegable del mundo.

El marco tectónico en el que se ubica este volcán es muy complejo (Figura 5.1.4): el proceso de subducción al que está sometida la fosa oceánica de las Shetland del Sur y que la llevó a separarse de la Península Antártica hace unos 2 Ma (millones de años), ha dado lugar a lo que hoy se conoce como el Rift del Bransfield [30]. Este rift, compuesto por tres cuencas extensionales activas, se caracteriza por una serie de volcanes submarinos, algunos de los cuales, como Decepción, siguen activos (Figura 5.2.1), lo que unido a otros procesos de subducción, como el asociado a la placa de Drake, explica la gran actividad sísmica de la isla.

Los valores numéricos de la Figura 5.1.4 relacionan la ubicación y la fecha de las erupciones del volcán de Decepción en los últimos 180 años. De forma general, todas ellas han sido de pequeño volumen y cercanas a la bahía interior. Las tres más recientes, relativas al 1967, 1969 y 1970, fueron avistadas y documentadas. Gracias a ello se conoce que la primera de estas erupciones (1967) tuvo de forma simultánea, dos focos eruptivos, separado el uno del otro por una distancia de 2 km. Compuesta por la expulsión de cenizas, bombas y vapor, la erupción uno de los focos, caracterizado por ser submarino, dio lugar a un nuevo islote ubicado en bahía Telefon. Unos años más tarde, en 1969, tuvo lugar una segunda etapa eruptiva que además de producir grietas en los glaciares y hielos del monte Pond, vino acompañada de flujos piroclásticos que destruyeron completamente la base científica chilena. Finalmente, en 1970, en las

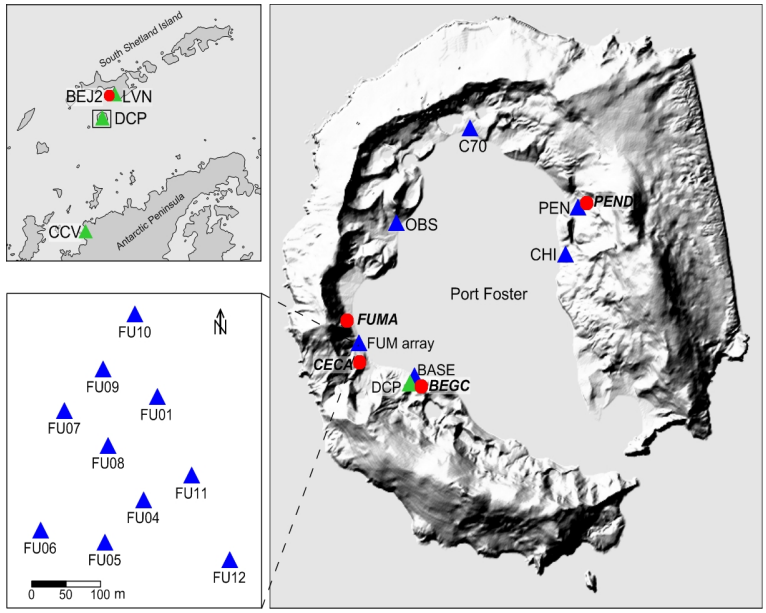


Figura 5.2.2: Ubicación de los sensores sísmicos (triángulos azules) durante las campañas sísmicas en la isla Decepción. Imagen extraída de [81]

cercanías de bahía Teléfono tuvo nuevamente lugar otra pequeña erupción que formó nuevos cráteres y modifico la orografía de la isla.

Desde entonces y hasta la fecha, la actividad volcánica de la isla se caracteriza por multitud de sismos, tanto regionales (asociados al complejo tectónico) como locales (asociados al vulcanismo y la inyección profunda de magma), y la existencia de áreas hidrotermales donde aparecen constantemente fumarolas (asociadas al proceso de evaporación del agua perteneciente al deshielo y filtraciones oceánicas) y aguas termales [30, 112]. Estos sistemas hidrotermales son un sujeto importante en la sismicidad de la isla, ya que son los responsables del origen de los LPEs.

5.2.1. Eventos del volcán de Isla Decepción

El corpus de datos del volcán de Decepción se ha construido a partir de datos adquiridos con sensores ubicados en diferentes áreas de la isla (Figura 5.2.2). Como se puede apreciar, el despliegue de estaciones se realiza de forma radial con el objetivo de registrar tanto la sismicidad local como la regional. Los puntos de color rojo corresponden con las bases científicas ubicadas en la isla. Los puntos azules corresponden a las estaciones sísmicas temporales desplegadas durante las campañas antárticas australes y el punto verde corresponde a la estación sísmica permanente (DCP), que registra información de forma ininterrumpida hasta agotar su batería una vez ha finalizado la campaña austral. Si se observa detenidamente, es fácil encontrar que la ubicación de las estaciones corresponde con los puntos donde se registraron las últimas erupciones (1967,1969 y 1970), así como en aquellos dónde se registran anomalías hidrotermales (bahía Fumarolas-FUM). A excepción de FUM que es un array sísmico con entre 9 y 11 sensores con el que se pretende detectar tanto la dirección como la velocidad aparente de los eventos sísmicos (dada la elevada actividad termal presente en la bahía, en la

que además de aguas calientes se registran de forma ininterrumpida grandes fumarolas y emisiones de gases y vapores de agua), el resto de estaciones son sensores con tres componentes que registran la información, la almacenan localmente en su memoria y la envían en tiempo real a la base mediante una red inalámbrica local (LAN).

En concordancia con la descripción de la actividad volcánica actual introducida en la sección anterior y siguiendo los criterios expuestos en [107], los eventos sismo-volcánicos asociados al volcán de isla Decepción que componen nuestro corpus de datos (recopilados durante las campañas 1994-1995, 1995-1996 y 2001-2002) se pueden caracterizar como sigue:

- **Eventos de largo período (LPE):** cuyo modelo fuente se relaciona con la dinámica de fluidos dentro del edificio del volcán: desde grietas en las que las se originan resonancias cuando los líquidos ascienden hacia la superficie, a la existencia de transitorios de presión dentro de la mezcla fluido-gas que también causan fenómenos de resonancia [48]. Están ubicados en la parte poco profunda del volcán, y su contenido de frecuencia está restringido a una banda estrecha entre 0.5 y 5 Hz. Generalmente tienen una duración menor de 60 segundos (Figura 5.2.3a) y su envolvente tiene forma de huso. En determinadas circunstancias, en Decepción se han registrado LPEs con una alta frecuencia dominante similar a los VTE, siendo necesario un análisis que separe las componentes para poder diferenciarlo.

- **Terremotos volcano-tectónicos (VTE):** son señales muy impulsivas generadas (dentro de un rango de profundidades) por el estrés sísmico. Este estrés es a menudo relacionado con procesos volcánicos como la interacción del agua con materiales calientes o migraciones de magma hacia la superficie que genera fracturas de rocas frágiles e incluso deformaciones. Cuando se produce una fractura sólida, se origina una onda sísmica en la que se pueden identificar las ondas P y S, y cuyo frente difiere en menos de 4 segundos. El contenido espectral de esta señal es muy amplio, alcanzando los 30 Hz.

- **Tremor volcánico (TRE):** es considerado un signo de alta actividad dentro del volcán. Aunque se han observado tremors resonantes en torno a frecuencias de 2 Hz, en Decepción, el tipo de tremor más comúnmente registrado es el espasmódico, cuyo contenido espectral está por debajo de los 5 Hz y cuya duración varía desde unos pocos minutos hasta meses. Algunas teorías consideran que estos eventos vienen generados por movimientos de magma, mientras que otras sugieren fluctuaciones de gas. Dado que todavía se desconocen los mecanismos de su fuente, la importancia y el momento entre la primera aparición de tremor y la posible actividad eruptiva sigue siendo un tema de discusión [184]. En muchos casos, la identificación y la distinción del tremor frente al ruido de fondo es una tarea bastante compleja, que requiere análisis avanzado de señales [8].

- **Eventos híbridos (HYB):** relacionados con episodios eruptivos inminentes, se caracterizan por una fase inicial de alta frecuencia, de corta duración, seguida de una segunda señal idéntica al LPE. Su origen puede explicarse por los incrementos de presión que provocan los terremotos. Las fracturas inducidas por presión se llenan de fluidos volcánicos, generando un tren de impulsos con un contenido espectral similar al de los LPEs. Tal y como se concluye en [106], en Decepción no existe una diferenciación clara con la que poder clasificar fehacientemente los eventos HYBs de los LPEs.

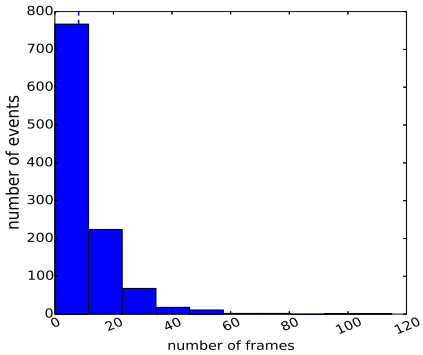
- **Silence (SIL):** son señales principalmente de baja amplitud. Se registran cuando las fuentes sísmicas internas dentro del volcán no emiten ninguna información sísmica. Su contenido espectral llega hasta los 10 Hz. En nuestro corpus de datos, este tipo de señales generalmente anteceden y preceden al resto de eventos.

La ocurrencia de estos eventos no sigue ningún patrón temporal, pudiendo estar su aparición influenciada por factores externos como los meteorológicos, la carga oceánica y el deshielo [147]. En este sentido, es posible encontrar una diferencia notable en la actividad sísmica de una campaña con respecto a otra, no implicando un aumento del riesgo volcánico.

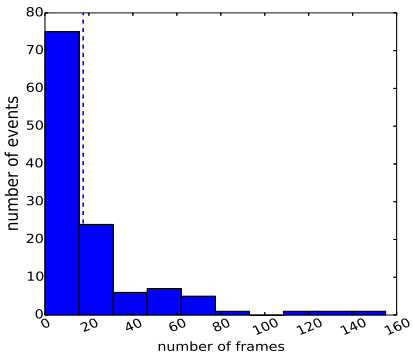
El corpus de datos final consta de 512 registros continuos de datos, con un total de 2193 eventos distribuidos como sigue: 75 VTE, 765 LPE, 77 TRE, 54 HYB y 1222 SIL.

5.3. Conclusiones

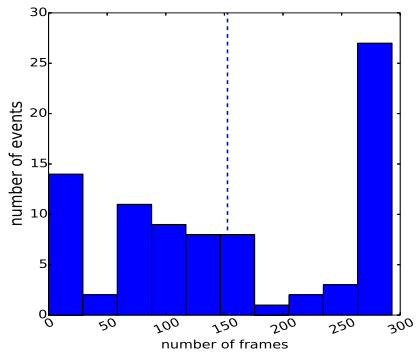
En este capítulo hemos descrito los marcos tectónicos y los principales eventos sismo-volcánicos del volcán de Fuego de Colima (México) e Isla Decepción (Antártida), que forman nuestro corpus de datos. Además de hacer uso de sus principales características espectrales y geofísicas, cada evento ha sido asociado a una fuente sísmica, dando así una mayor perspectiva de su relevancia dentro de un escenario eruptivo.



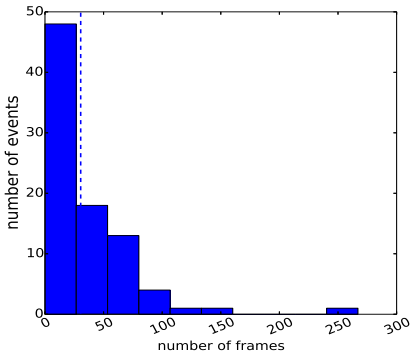
(a) Eventos de largo período (LPE)



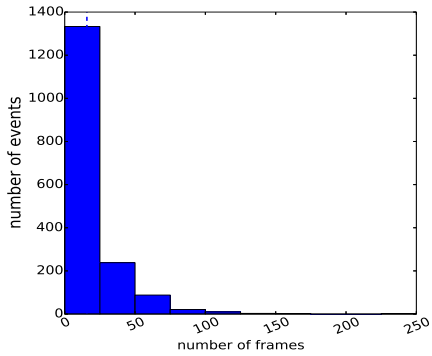
(b) Terremotos volcano-tectónicos (VTE)



(c) Tremor (TRE)



(d) Eventos híbridos (HYB)



(e) Silencio (SIL)

Figura 5.2.3: Histogramas que resumen la distribución de duración en términos del número de frames de las señales sísmicas registradas durante las campañas sísmicas de 1994-1995, 1995-1996 y 2001-2002 en el volcán de Isla Decepción. Las líneas discontinuas reflejan la duración media de cada evento. Todas las señales hacen uso de ventanas de 4 seg con un solapamiento de 3.5 seg.

Capítulo 6

Parametrización o Extracción de Características

La extracción de características es un proceso crucial para construir clasificadores fiables cuando no se dispone de grandes cantidades de datos y cuando los sistemas de clasificación no sean en sí mismos extractores de características.

Partiendo de la base de que no siempre se consigue, la información discriminativa extraída mediante la parametrización, además de ser de algún modo invariante, se puede representar en un espacio de características que de forma general, permite una mejor separación lineal que aquellas representaciones obtenidas directamente de los datos [98]. Aunque uno de los principales atributos del aprendizaje profundo es reemplazar el proceso de extracción de características por características extraídas directamente por el modelo durante la etapa de aprendizaje [85], las limitaciones impuestas por la naturaleza de las señales sismo-volcánicas nos obligan a abordar el problema de la clasificación desde un enfoque clásico, haciendo uso de la experiencia y el conocimiento adquirido durante décadas en el procesado de señales geofísicas.

En este capítulo se motivará la necesidad de parametrizar las señales para poder construir un sistema de clasificación robusto, y se detallará el proceso de extracción de características llevado a cabo en cada uno de los escenarios del problema planteados, clasificación en aislado y clasificación en continuo. En la sección 6.1 se describirá el preprocesado de las señales asociado a la normalización del uso de diferentes sistemas de adquisición. En la sección 6.2 se explorará la viabilidad de usar arquitecturas basadas en redes neuronales (CNNs y RNNs) como extractores de características a partir de los cuales construir un posterior sistema de clasificación en aislado. En la sección 6.3, una vez observada la necesidad de representar a las señales por un vector de parámetros, se usará una red neuronal clásica (MLP) para, de forma experimental, decidir el vector de parámetros más adecuado para los datos disponibles. En la sección 6.4, siguiendo la metodología descrita en la sección 6.2, se motivará la necesidad del uso de un proceso de extracción de características previo a la tarea de detección y clasificación en continuo. Finalmente, en la sección 6.5, una vez motivada la necesidad de la parametrización, se describirá en profundidad el esquema propuesto.

	T1	T2	T3	T4	AVG
DeepCNN-3Layers	52.91	53.94	58.43	55.25	55.13
DeepCNN-5Layers	43.82	46.45	47.64	46.27	46.04
RNN-LSTM	50.35	49.07	48.719	48.71	49.08

Tabla 6.1: Resultados obtenidos clasificando datos sismo-volcánicos de Colima sin parametrizar con RNN y CNN. T1,T2,T3,T4 corresponden con los resultados para cada uno de los test usados en la validación cruzada. AVG corresponde con la media de estos resultados. La medida de rendimiento corresponde con el porcentaje de eventos bien clasificados (accuracy) y viene dada en %.

6.1. Prepocesamiento de las señales

En primer lugar, los sismogramas se filtran y se normalizan los efectos del uso de diferentes sistemas de adquisición. Las operaciones realizadas durante la fase de preprocesado pueden agruparse en dos categorías. Por un lado, están las operaciones destinadas a detectar y manipular datos considerados imperfectos;; y por otro lado, se consideran aquellas operaciones cuya finalidad es transformar los datos para hacerlos más manejables. Aplicado a nuestro problema, el preprocesado se corresponde con:

- **Filtrado en la banda de 1 a 25 Hz:** conociendo que la mayor parte de la energía de los eventos sismo-volcánicos se concentra en la banda $[f_{l,} = 1, f_h = 25]$ Hz [48, 207], se ha optado por aplicar un filtro de Butterworth de 4^{o} orden a cada registro. Es importante destacar, que como ningún corpus de datos consta de eventos de muy largo período (VLP- Very Long Period), filtrar por encima de 1 Hz eliminará bastante ruido sísmico de fondo [165].

- **Submuestreo de los datos:** con el objetivo de poder usar registros pertenecientes a diferentes sistemas de adquisición, todas las señales se submuestrean a una frecuencia de 50 Hz. Aquellas señales cuya amplitud quede fuera del rango dinámico del sensor y aquellas otras cuya SNR sea demasiado baja, serán excluidas del corpus de datos, ya que las primeras no están correctamente descritas y las últimas contienen tanto ruido que no ofrecen ninguna información.

Pese a que casi todos los sensores registran la información en tres componentes, en este estudio se hará uso solo de la información registrada en la componente vertical.

6.2. Trascendencia de la parametrización en la clasificación en aislado

Desde el punto de vista del aprendizaje automático, el apilamiento de varias capas de transformaciones no lineales para extraer representaciones jerárquicas abstractas de los datos con las que reemplazar el proceso de extracción clásico de características (parametrización), parece, a priori, ser un buen enfoque para caracterizar eventos sismo-volcánicos. Basándonos en las propiedades que definen a cada uno de los eventos (sección 1.2.1) y los inconvenientes asociados con ellos (sección 2.2.1), podemos concluir que en un mismo período eruptivo, se pueden encontrar eventos con una duración muy diferente: de minutos a días en el caso de temblores volcánicos, o de pocos segundos en el caso de explosiones, dando lugar a dependencias temporales de largo alcance y diferente duración que son difíciles de modelar.

Teniendo en cuenta las limitaciones físicas impuestas por las señales, el uso de datos sismo-volcánicos sin parametrizar en arquitecturas profundas queda condicionado por:

- En primer lugar, las DNNs necesitan vectores de entrada de igual longitud. Esta limitación puede ser abordada desde dos perspectivas:

  - Por un lado, si se opta por el uso de datos sin parametrizar, es decir, la señal en el dominio del tiempo, se deberán adaptar las longitudes de los vectores de entrada a aquellas de mayor orden. Esto conducirá a vectores de entrada extremadamente grandes, que a su vez incurrirán en millones de parámetros a estimar, ya que se deberán usar capas adyacentes con muchas unidades ocultas con las que ir capturando características más abstractas. Este enfoque se presenta inabordable dado el tamaño de nuestro corpus de datos.
  - Por otro lado, las señales pueden ser parametrizadas, obteniendo todas la misma dimensión y reduciendo el número de parámetros a estimar. Aunque en la sección 3.4 hemos motivado el uso de las CNNs como extractores de características [120], en nuestro caso, no es la mejor opción debido principalmente al hecho de que las señales no son de igual longitud.

- En segundo y último lugar, aunque dentro del ámbito del aprendizaje profundo existen arquitecturas capaces de manejar datos de diferente dimensión, como son las redes neuronales recurrentes (RNN), la extensa duración de eventos como el tremor volcánico o la colada, pueden conducir a los modelos a situaciones de subajuste (underfitting), debido al problema del desvanecimiento o desbordamiento del gradiente (sección 4.3) [164] que actúa como una restricción para aprender las dependencias a largo plazo.

Siguiendo[57] se han realizado una serie de experimentos para comprobar tanto la versatilidad de las CNNs como extractoras de características, como el uso de las RNNs para modelar directamente las secuencias temporales de distinta duración que conforman la base de datos. La Tabla 6.1 muestra los resultados obtenidos.

Ni el uso de arquitecturas convolucionales, ni el uso de arquitecturas recurrentes con las que manejar datos de diferente longitud, han brindado buenos resultados:

- En el caso de las CNNs, se probaron configuraciones de 3 y 5 capas ocultas con 16 hasta 32 filtros de convolución. Basándonos en los porcentajes de reconocimiento obtenidos, se observa una ligera mejoría con las configuraciones con 3 capas. Este resultado está motivado en el tamaño del corpus de datos, a partir del cual es muy difícil ajustar modelos de más de tres capas. No obstante, ninguna de las configuraciones obtiene porcentajes de reconocimiento aceptables a partir de los cuales implementar un sistema de alerta temprana. Por lo tanto, podemos concluir que las CNNs como extractores de características aplicadas a nuestro corpus de datos, no tienen aplicabilidad.

- En el caso de las RNNs, dada la longitud de algunos de los registros del corpus de datos, se probó una LSTM en la que el número de unidades de la capa oculta, varió entre 10 y 100. Su mejor configuración se encontró en 70 unidades. Los porcentajes de reconocimiento obtenidos son muy similares a los obtenidos por las CNNs.

En este sentido, concluimos que la parametrización de los datos sismo-volcánicos es un paso crucial para proporcionar información útil como entrada al clasificador dado el tamaño de nuestra base de datos para la base de datos que disponemos.

6.3. Clasificación en aislado: parametrización basada en LPC.

Las señales geofísicas, y más concretamente, las señales sísmicas, han sido ampliamente estudiadas y parametrizadas desde diferentes enfoques (forma de onda, contenido espectral, polarización y atributos relacionados con la geometría de red sísmica) [10, 170]. No obstante, una de las parametrizaciones más usadas dado su escaso coste computacional y su sencillez ha sido la codificación predictiva lineal [158].

La codificación predictiva lineal (LPC-Linear Prediction Coefficients) se basa en la codificación de la información mediante la combinación lineal de k coeficientes más una señal de error:

$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + e[n] \tag{6.3.1}$$

de manera que el error obtenido entre la señal original $s[n]$ y la señal reconstruida a partir de dicha codificación sea mínimo.

Este enfoque permite modelar cualquier señal independientemente de su duración, de forma simple y poco costosa. Aunque de forma general, la parametrización predictiva lineal se suele asociar a entornos de audio digital y a sistemas de procesado de voz, en el área de la geofísica aplicada ha tenido gran repercusión demostrando ser una representación robusta de la información [181, 71, 80, 59, 140].

6.3.1. Elección del esquema de codificación

Motivado el uso de la codificación predictiva como extractor de características, es preciso determinar qué esquema de codificación vamos a seguir para parametrizar las señales de nuestro corpus de datos:

- Dado que la mayor parte de los eventos no son estacionarios, es decir, su forma de onda va variando a lo largo de la duración en tiempo del evento, una forma de caracterizar esta evolución temporal es dividir el evento en un número n de segmentos (no solapados) [60, 12].

- En segundo lugar, es necesario definir el número de coeficientes con los que codificar cada segmento. Según los estudios abordados en [181], un evento sismo-volcánico puede ser robustamente representado mediante 5 coeficientes LPC.

Siguiendo estas dos pautas se han dispuesto las siguientes aproximaciones:

- Cada evento sismo-volcánico independientemente de su duración se representa con entre 1 y 5 segmentos no solapados de igual longitud.

- Cada segmento es representado con 5, 8, 10, 12 y 15 coeficientes LPC.

- Además de la información aportada por los LPC, cada sismograma es complementado con características estadísticas [53] de su evolución tanto el dominio del tiempo como en el dominio de la frecuencia. Para ello, se calculan las sumas

acumuladas de la señal en el tiempo (amplitud) y en el dominio de la frecuencia (FFT-Fast Fourier Transform), añadiendo como características los valores de dichas sumas correspondientes a los percentiles 20 %, 50 % y 80 %.

6.3.2. Estudio experimental

En esta sección se llevará a cabo un estudio experimental de los esquemas de parametrización expuestos en la sección 6.3.1. Se medirá su capacidad de representación comparando el rendimiento (accuracy) de clasificadores que los aplican. El objetivo de este estudio será obtener el número de segmentos y el número de coeficientes por segmento que mejor representa la evolución temporal de la señal.

6.3.2.1. Eleeción del número de segmentos

Los datos (una vez pre-procesados) han sido codificados mediante las diferentes aproximaciones y evaluados en un MLP con una sola capa oculta en el que el número de unidades varió entre 25, 50, 100, 200, 300, 500, 1000 y 2000.

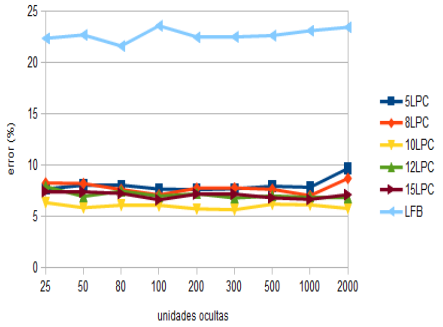
De forma complementaria, cada segmento asociado ha sido parametrizado mediante un banco de filtros (LFB- Log Filter Bank) a partir de los cuales se obtiene la evolución temporal de su energía espectral [173, 213]. El objetivo principal de este nuevo experimento es comparar la robustez de la representación de los LPC frente a LFB con 16 filtros.

A tenor de los resultados obtenidos (Figura 6.3.1), se puede observar que las mejores aproximaciones usan codificadores predictivos, dividiendo la señal en dos y tres segmentos, siendo la aproximación de tres segmentos levemente mejor que la de dos.

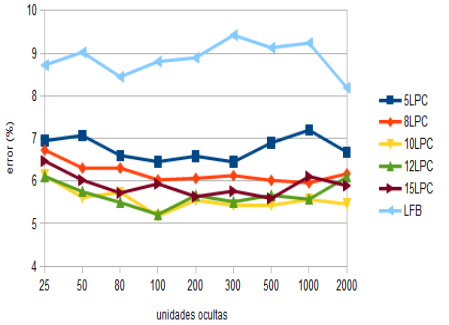
Es también importante destacar, cómo las parametrizaciones basadas en la evolución temporal de la energía espectral (LFB) obtienen tasas de error más elevadas cuando se usan esquemas con menos de tres segmentos.

6.3.2.2. Elección del número de coeficientes LPC por segmento

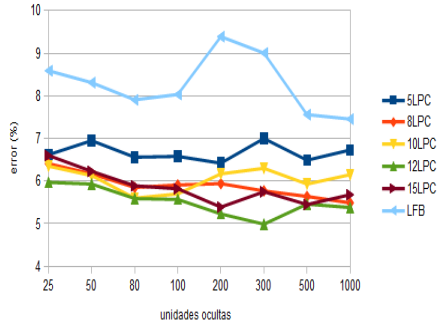
Habiendo motivado la elección de la codificación predictiva lineal como método de parametrización y habiendo fijado la descripción de las señales mediante tres segmentos dada su menor tasa de error, es necesario hacer un estudio del número de coeficientes que se usarán para codificar cada segmento, o dicho de otro modo, el número de coeficientes que describirán cada segmento de la señal.



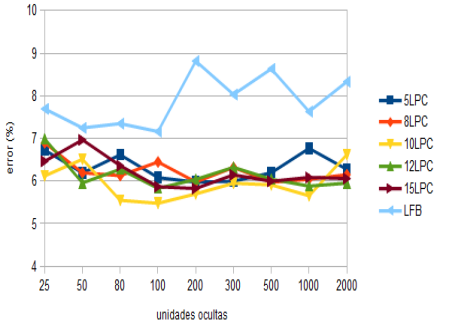
(a) 1 segmento



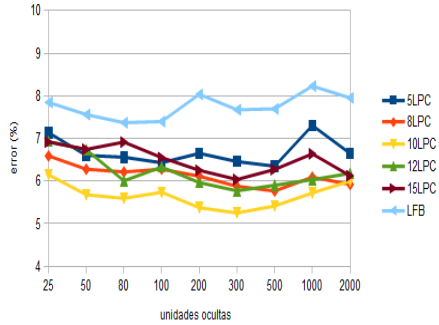
(b) 2 segmentos



(c) 3 segmentos



(d) 4 segmentos



(e) 5 segmentos

Figura 6.3.1: Comparativa del porcentaje de error de clasificación para las parametrizaciones LPC y LFB. Análisis para diferente número de coeficientes LPC con entre 1 y 5 segmentos por señal . LFB con 16 filtros.

A diferencia del experimento anterior, en el que las configuraciones del MLP variaron entre 25, 50, 100, 200, 300, 500, 1000 y 2000 unidades ocultas, en este experimento, se ha llevado a cabo una búsqueda exhaustiva en torno a la configuración óptima del MLP para cada una de las codificaciones propuestas.

	NOI	EXP	REG	COL	VTE	TRE	LPE	AVG
3 LPC	94.85	62.23	81.25	93.90	88.45	73.73	86.75	80.08
5 LPC	94.24	54.79	80.80	92.80	89.24	73.56	89.59	79.17
6 LPC	95.83	65.43	83.48	94.18	89.24	79.76	89.51	82.32
7 LPC	95.10	62.77	82.14	92.80	90.37	78.65	90.62	81.61
8 LPC	95.22	69.68	80.36	92.00	89.69	79.21	91.48	82.26
9 LPC	94.73	68.62	79.91	92.80	91.05	80.12	91.17	82.39
10 LPC	95.10	70.21	79.02	93.22	88.67	81.19	90.06	82.30
12 LPC	96.08	65.96	79.91	94.98	90.37	81.24	92.11	82.68
15 LPC	94.98	70.74	78.57	93.09	89.70	81.75	90.85	82.58

Tabla 6.2: Resultados obtenidos por un MLP en su configuración óptima variando el número de coeficientes LPC por segmento y siendo cada señal descrita mediante tres segmentos. Los valores corresponden con el porcentaje promedio (precisión) (%) de eventos correctamente clasificados para cada tipo de evento.

Las tablas 6.2 y 6.3 resumen los resultados obtenidos por un MLP en su configuración óptima con respecto a los criterios Precision y F1 score variando el número de coeficientes LPC por segmento.

	NOI	EXP	REG	COL	VTE	TRE	LPE	AVG
3 LPC	97.40	79.87	90.39	97.00	94.34	86.61	93.41	91.29
5 LPC	97.11	75.33	90.03	96.45	94.68	86.67	94.85	90.73
6 LPC	97.91	82.07	91.53	97.11	94.70	89.89	94.74	92.57
7 LPC	97.54	79.70	90.74	96.43	95.21	89.38	95.35	92.05
8 LPC	97.60	83.55	89.77	96.03	94.93	89.52	95.77	92.45
9 LPC	97.35	83.45	90.12	96.42	95.53	90.21	95.58	92.66
10 LPC	97.54	84.39	89.03	96.62	94.41	90.74	95.03	92.54
12 LPC	98.03	81.14	90.01	97.50	95.22	90.62	96.07	92.66
15 LPC	97.51	84.50	89.41	96.53	94.86	90.65	95.48	92.71

Tabla 6.3: Resultados obtenidos por un MLP en su configuración óptima variando el número de coeficientes LPC por segmento y siendo cada señal descrita mediante tres segmentos. Los valores corresponden con el porcentaje F1_score promedio (%) de eventos correctamente clasificados para cada tipo de evento.

Dado que nuestro objetivo es estudiar la capacidad de representación y abstracción de los modelos neuronales profundos, la selección de la parametrización usada se hará a partir de dos características:

- Cantidad de parámetros a sintonizar, teniendo en cuenta el tamaño del corpus de datos del que se dispone.
- Tasa de error obtenida durante el proceso de clasificación.

Teniendo en cuenta ambas restricciones, se ha optado por la elección de una codificación basada en 5 coeficientes. Como se puede apreciar, esta parametrización es la que peor porcentaje de reconocimiento ofrece, siendo un buen punto de partida para analizar si las abstracciones jerárquicas implícitas a los modelos profundos ayudan en la tarea de clasificación. Además, esta parametrización aborda durante la búsqueda

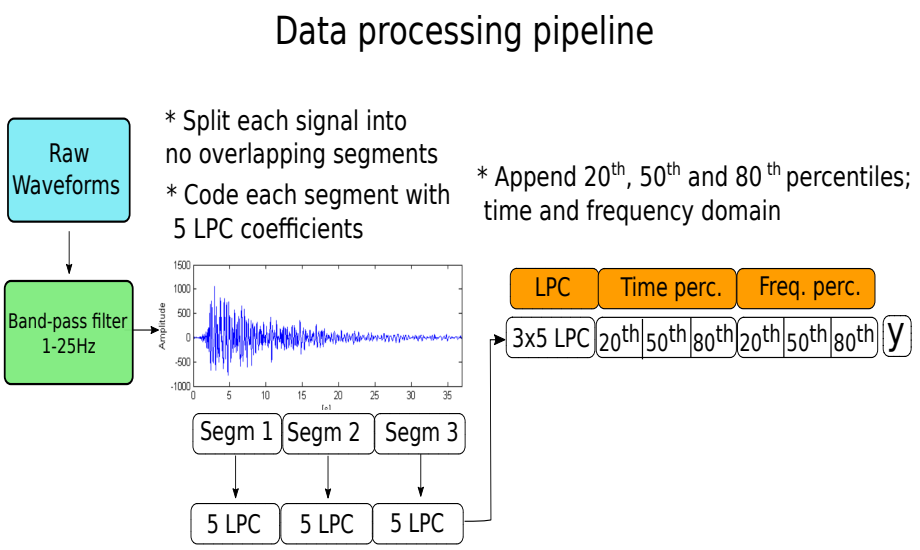


Figura 6.3.2: Descripción general de la etapa de preprocesado y extracción de características. Cada sismograma se divide en tres segmentos no superpuestos. Todos los sismogramas están representados un vector de 21 características, independientemente de su duración y de forma de onda.

de la configuración óptima un número de parámetros a sintonizar consecuente con el tamaño de nuestro corpus de datos.

Por lo tanto, independientemente de su duración y forma de onda, cada sismograma será descrito por un vector de 21 características, $(3 \cdot k) + 6$, siendo $k = 5$ el orden de los coeficientes LPC, *3* el número de segmentos, y *6* el número de características correspondientes a los percentiles 20 %, 50 % y 80 %, en ambos dominios, tiempo y frecuencia. (Figura 6.3.2).

6.4. Clasificación en continuo ¿Por qué es importante nuevamente la parametrización?

Teniendo presente que nuestro corpus de datos, correspondiente al volcán de Isla Decepción (sección 5.2), está compuesto por registros que contienen un número indeterminado de eventos, capturar la evolución temporal de estos sin ninguna parametrización, basándonos única y exclusivamente en la información temporal de la señal (amplitud), es una tarea muy complicada, ya que la forma de onda de la señal se ve altamente afectada por las características del medio (sección1.3).

Por tanto, el uso de información sin parametrizar disminuye la capacidad de los modelos de encontrar representaciones o características discriminativas con las que resolver de manera eficaz la detección y clasificación de los eventos.

En esta sección se llevarán a cabo una serie de experimentos a partir de los cuales se evaluarán los resultados obtenidos por varias arquitecturas haciendo uso de datos parametrizados y datos sin parametrizar.

Una vez motivada la necesidad de la parametrización, se describirá en profundidad el esquema propuesto.

6.4.1. Estudio experimental

El estudio experimental asociado a la justificación de la necesidad de un proceso de parametrización se llevará a cabo a partir de dos esquemas de parametrización diferentes:

- Por un lado, LBF, que aunque también se ve afectada por las particularidades del medio, proporciona características muy representativas y muy discriminativas que mejoran el modelado de la evolución temporal.
- Por otro lado, un esquema basado en coeficientes de predicción lineal, dado el alto rendimiento obtenido como esquema en la clasificación en aislado.

Para ello, se usarán las tres variantes de la arquitectura RNNs (Vanilla, LSTM, GRU) (sección 4.1), de manera que cada una de ellas será entrenada usando los datos parametrizados bajo cada uno de los esquemas. Una vez entrenadas, se analizarán los resultados de clasificación obtenidos.

Es importante destacar que el entrenamiento de las RNNs se lleva a cabo de forma iterativa en diferentes ventanas de tiempo. Cada uno de los registros se divide en segmentos o frames de duración determinada. Esta duración determina el ancho de la ventana de análisis, que posteriormente será parametrizada y representada mediante un vector de características que será la entrada al modelo.

6.4.1.1. Elección del tamaño de la ventana de análisis

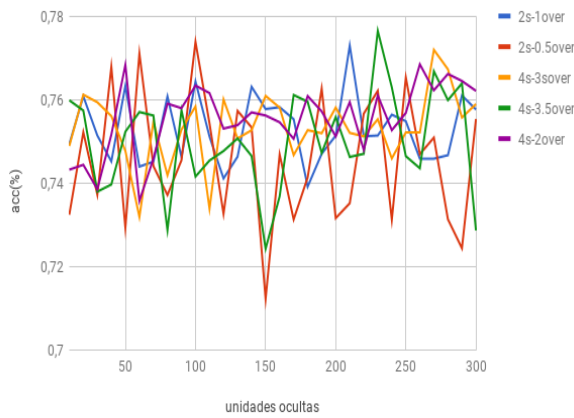


Figura 6.4.1: Estudio del rendimiento o accuracy por frame asociado al modelo RNN-LSTM usando diferentes configuraciones en su capa oculta y diferentes tamaño de ventana de análisis.

El ancho de la ventana o tamaño del frame es un parámetro de vital importancia, ya que de él dependerá que la evolución temporal de los eventos sea capturada correctamente.

De forma general, este parámetro se ajusta en función de los datos y de su evolución temporal. Analizando detenidamente los eventos que componen nuestro corpus de datos, se ha observado que la duración mínima de los mismos es de aproximadamente 5 segundos.

Conf.	2s-0.5s	2s-1s	4s-2s	4s-3s	4s-3.5s
RNN-LSTM	77.38	77.29	76.84	77.19	77.64

Tabla 6.4: Resumen de los mejores resultados obtenidos por modelos RNN-LSTM (Figura 6.4.1) usando diferentes configuraciones de ventanas sobre datos en crudo, sin ningún tipo de parametrización (raw). Los resultados están expresados en términos de ventanas o frames correctamente clasificados (%). Conf. corresponde a las configuraciones de ventanas usadas: el primer valor corresponde a el tamaño de la ventana, mientras que el segundo corresponde con la superposición entre ventanas adyacentes.

Con el objetivo de obtener el tamaño de ventana óptimo, se han evaluado varios esquemas con datos sin parametrizar en una RNN-LSTM en la que el número de unidades ocultas varió entre 10 y 300(Figura 6.4.1):

- Ventanas de 2 segundos con superposiciones de 1.5 y 1 segundos.
- Ventanas de 4 segundos con superposiciones de 3.5, 3 y 2 segundos.

El uso de estas longitudes de ventana está motivado por:

- Ventanas con una duración menor de 2 segundos no tendrían ninguna aplicabilidad puesto que el evento de menor duración observado tiene una duración aproximada de 5 segundos. Además, teniendo presente la naturaleza de las RNNs, el uso de ventanas de análisis muy reducidas podrían conducir a situaciones de desvanecimiento y desborde de gradiente.
- Ventanas con una duración muy superior a cinco segundos dejarían de modelar los eventos de menor duración. Aunque una ventana de análisis grande supone una mayor resolución espectral, dado el método entrenamiento, en el que cada ventana debe tener una etiqueta asociada, los eventos de menor duración que pudiesen estar presentes dentro de la misma quedarían enmascarados, puesto que dicha ventana debería ser etiquetada o asociada al evento predominante o de mayor duración. Esta restricción dificultaría enormemente el modelado o caracterización de los eventos de menor duración.

La Tabla 6.4 resume los mejores resultados obtenidos. A tenor de dichos resultados, se concluye que la mejor de las configuraciones se corresponde con ventanas de 4 segundos, solapadas 3.5 segundos, tal y como se describe en la Figura 6.5.1.

6.4.1.2. Elección del número de LPC por segmento

Una vez motivado el uso de ventanas de 4 segundos solapadas 3.5 segundos, es necesario estudiar el número óptimo de LPC por ventana para evaluar el impacto de la parametrización de los datos. Para ello, cada ventana fue codificada con:

- 3, 5, 6, 7, 8, 9, 10, 12, 15 LPC.

La Tabla 6.5 resume los mejores resultados obtenidos por las tres arquitecturas RNNs variando el número de unidades ocultas (entre 10 y 300). El mejor resultado se obtuvo con 5 LPC. En la sección D del apéndice, se encuentra un estudio ampliado de estos resultados.

	RNN-Vanilla	RNN-GRU	RNN-LSTM
3 LPC	72.52	78.40	77.18
5 LPC	77.25	80.35	79.86
6 LPC	77.29	79.89	78.91
7 LPC	76.36	79.14	79.23
8 LPC	76.13	79.47	77.72
9 LPC	77.17	79.22	78.24
10 LPC	76.92	80.02	78.22
12 LPC	77.27	79.10	77.10
15 LPC	76.19	79.43	78.14

Tabla 6.5: Comparativa del porcentaje de acierto en la clasificación para las arquitecturas recurrentes propuestas. Estudio de ventanas de análisis codificadas con diferente número de coeficientes predictivos lineales. Los resultados están expresados en %.

	RNN-Vanilla	RNN-GRU	RNN-LSTM
Raw data	73.83	77.10	77.64
LPC (5 coeffients)	77.25	80.35	79.86
LPC (5 coefficients)+($\triangle, \triangle\triangle$)	77.34	80.40	79.31
LFB	79.83	84.07	83.56
LFB+($\triangle, \triangle\triangle$)	82.39	85.43	84,88

Tabla 6.6: Rendimiento general de los sistemas en su configuración óptima, usando datos sin parametrizar y datos parametrizados con diferentes técnicas. Los resultados se han obtenido a nivel de frames, vienen dados en % y corresponden con el promedio de los resultados de los cuatro test usados en la validación cruzada.

6.4.1.3. Comparativa y resultados

En esta sección se han comparado los esquemas de parametrización propuestos. Cada esquema de parametrización ha sido evaluado en cada variante de la arquitectura RNN (Vanilla, LSTM, GRU), en la que el número de unidades de su capa oculta varió entre 10 y 300.

En el caso de la parametrización basada en LPC, cada ventana ha sido codificada con 5 coeficientes, dada su mejor tasa de reconocimiento. La parametrización basada en evolución temporal de la energía espectral se llevó a cabo a partir de una aproximación conocida como banco de filtros en escala de frecuencias logarítmica (LFB), en el que el número de filtros se ajustó a 16.

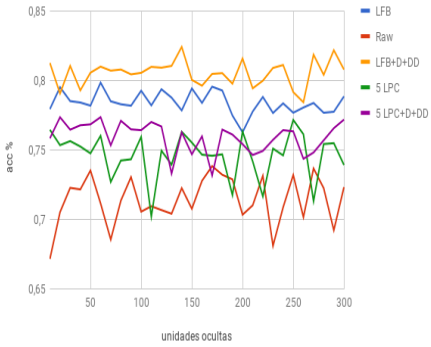
Además, se realizó un nuevo experimento en el que cada ventana de análisis es complementada con información contextual correspondiente a las derivadas temporales de primer y segundo orden ($\triangle, \triangle\triangle$) de cada componente.

La Figura 6.4.2 describe los resultados obtenidos por cada arquitectura en función del esquema de parametrización y del número de unidades de su capa oculta.

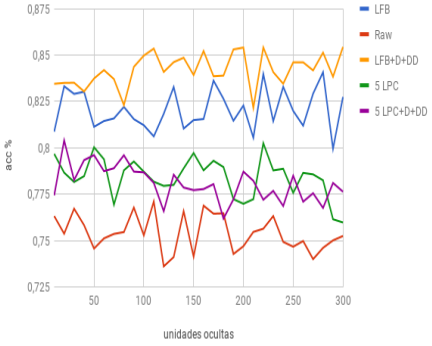
La Tabla 6.6 resume los mejores resultados obtenidos por cada arquitectura en función de la parametrización. Varias son las conclusiones obtenidas:

- La parametrización de los registros mejora significativamente la tasa de detección y clasificación.

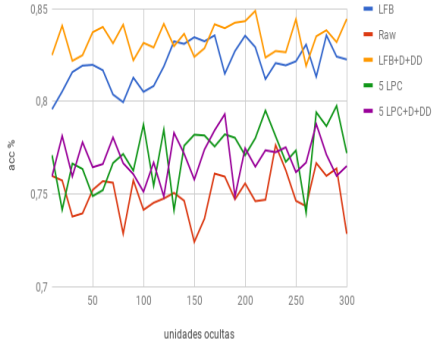
- Las arquitecturas más complejas (LSTM y GRU), obtienen siempre un mejor



(a) Vanilla



(b) GRU



(c) LSTM

Figura 6.4.2: Estudio del número óptimo de unidades de la capa oculta usando los esquemas de parametrización LFB y 5 LPC ($+(\triangle, \triangle\triangle)$)

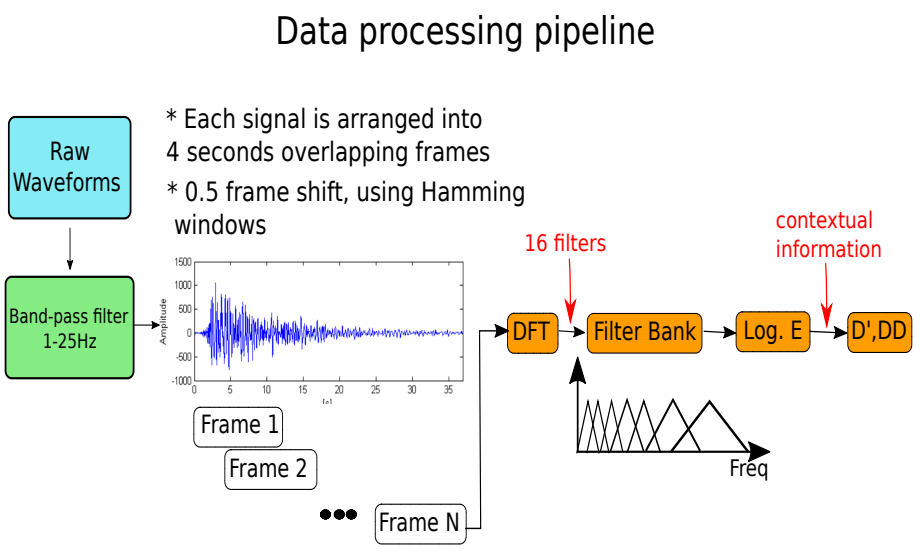


Figura 6.5.1: Descripción general de la etapa de preprocesamiento y extracción de características propuesta para la detección y clasificación es continuo. .

porcentaje de detección y clasificación en todas las parametrizaciones. Esto se debe a la gran variabilidad de la duración de las señales, cuyas dependencias temporales quedan mejor modeladas por este tipo de arquitecturas.

- Pese a que la parametrización LPC mejora el porcentaje de reconocimiento con respecto a los datos sin parametrizar, son las parametrizaciones basadas en la evolución temporal de la energía espectral (LFB), las que realmente aportan una mejora significativa en el rendimiento de los sistemas.

- A medida que la parametrización es más representativa (LFB+($\triangle, \triangle\triangle$)), la diferencia en cuanto al porcentaje de reconocimiento de los modelos se reduce. Esta apreciación puede ser explicada desde un punto de vista algorítmico:

    - Una parametrización muy descriptiva ayuda a los modelos menos complejos (RNN-Vanilla) a modelar la evolución espectral del evento, mejorando la detección tanto al inicio como al final del mismo. Contrariamente, cuando la parametrización es menos descriptiva, la delimitación de los eventos no es tan clara y por consiguiente, los modelos menos complejos reducen notablemente su capacidad de detección.

6.5. Descripción detallada del esquema de parametrización asociada a la clasificación en continuo

A partir de los resultados obtenidos en la sección 6.4.1.3 podemos concluir que la parametrización LFB una buena aproximación para abordar la detección y clasificación de eventos sismo-volcánicos en tiempo real. Su esquema de parametrización consta de varias etapas:

- **Enventanamiento o fragmentación (windowing):** el registro de datos, continuo, se divide en segmentos o frames de duración determinada. Esta duración determina el ancho de la ventana de análisis, que posteriormente será parametrizada y representada mediante un vector de características. Para mejorar la robustez, los segmentos se suavizan con una ventana de Hamming y solapan unos con otros.
- **Análisis mediante banco de filtros:** para cada uno de los segmentos, se halla su transformada rápida de Fourier (FFT- Fourier Fast Transform). Su espectro es analizado en la banda de 1 a 25 Hz. Para ello:
  - El eje de frecuencias se traslada a escala logarítmica.
  - Una vez el eje ha sido logarítmicamente escalado, se construye un banco de filtros triangulares (en nuestro caso 16) de igual duración y solapados al 50 % que cubran todo el intervalo de análisis.
  - Finalmente, el rango o intervalo que cubre cada filtro en escala logarítmica es representado en la escala lineal de frecuencias. Los filtros centrados en bajas frecuencias serán por tanto, más estrechos que los centrados en aquellas más altas, permitiendo un énfasis del análisis en las bajas frecuencias que es donde se encuentra la mayor parte de la energía de los eventos sismovolcánicos. El vector de características para cada ventana o segmento, estará compuesto por 16 componentes, correspondientes al valor del logaritmo de la energía de cada banda espectral.
- **Decorrelación de componentes:** debido al solapamiento de los canales o bandas espectrales, mucha de la información capturada por los filtros es redundante. Esta redundancia se refleja en un alto grado de correlación entre las componentes que forman el vector de características. Con el objetivo de decorrelacionar dichas componentes y eliminar parte de la información redundante, se puede aplicar la transformada discreta del coseno (DCT). No obstante, las redes neuronales son menos susceptibles a entradas altamente correlacionadas y por lo tanto, la DCT no es un paso estrictamente necesario. [98].
- **Incorporación de información dinámica y energética:** Además de lo anterior, como una versión mejorada de la parametrización, además de la información proporcionada por el banco de filtros correspondiente a la energía del segmento, al vector de características se añade información contextual. Para ello se calculan las derivadas temporales de primer y segundo orden ($\triangle, \triangle\triangle$) de cada componente [122]. Una vez calculadas ambas derivadas, el tamaño del vector de características se habrá triplicado con respecto al inicial.

6.6. Conclusiones

En este capítulo hemos motivado la necesidad de parametrizar las señales para poder construir sistemas de clasificación robustos. Dos han sido los casos de estudio.

- Clasificación en aislado:
  - Se ha explorado la viabilidad de usar las CNNs y RNNs como extractores de características a partir de las cuales implementar los sistemas de clasificación sin necesidad de un proceso de parametrización previo, observando

que en nuestro corpus de datos, este tipo de técnicas no obtienen buenos resultados.

- Observada la necesidad de representar las señales con un vector de parámetros, se han estudiado dos esquemas con los que abordarla: LPC y LFB. Haciendo uno de una red neuronal clásica con una sola capa oculta (MLP), de forma experimental, se han evaluado una serie de vectores de parámetros obtenidos a partir de diferentes configuraciones de estos esquemas de parametrización, concluyendo que en nuestro corpus de datos, el esquema basado en LPC obtiene mejores resultados

- Clasificación en continuo:

  - Siguiendo la metodología descrita en el proceso de clasificación en aislado, hemos motivado la necesidad de uso de un proceso de extracción de características previo a la implementación de un sistema de reconocimiento en continuo. Haciendo uso de un RNN-LSTM, se han probado diferentes tamaños de ventana de análisis sobre datos sin parametrizar, eligiendo como tamaño final, el que mejor resultados de clasificación obtuvo.
  - Una vez elegido el tamaño de ventana de análisis, la señales han sido parametrizadas siguiendo los esquemas señalados anteriormente: LPC y LBF. Tras evaluar diferentes vectores de características obtenidos a partir de diferentes configuraciones de estos esquemas de parametrización, se concluye que, para nuestro corpus de datos, la mejor parametrización es LFB.
  - Finalmente, hemos descrito detalladamente la implementación de LFB.

Capítulo 7

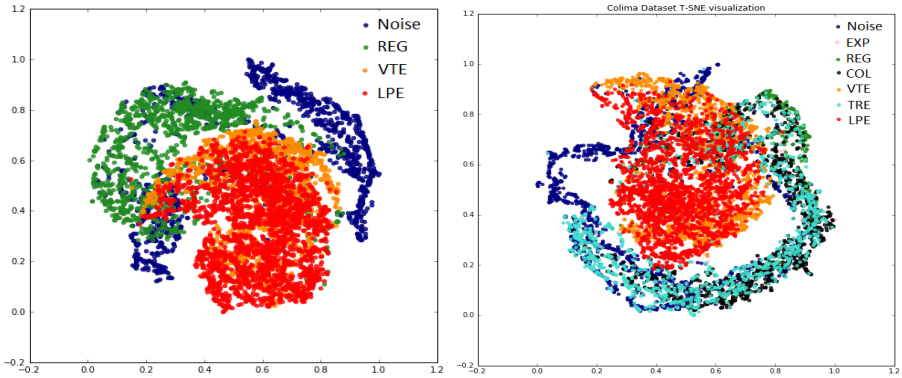
Clasificación en aislado de eventos sismo-volcánicos pertenecientes al volcán de Fuego (Colima) mediante SDAs y DBNS

El monitoreo de volcanes activos genera ingentes cantidades de datos que los observatorios vulcanológicos difícilmente pueden manejar a corto plazo. A menudo, esta información es analizada de forma aislada, es decir, no se hace un estudio completo de toda la serie sísmica, sino que se extraen (aíslan) y estudian los eventos más relevantes con el objeto de profundizar en el conocimiento de la dinámica histórica del volcán.

En este capítulo se aborda la implementación de un sistema de reconocimiento en aislado de eventos sísmicos de origen volcánico basado en técnicas de aprendizaje profundo (Deep Learning), más concretamente, en SDAs y DBNs. Estas técnicas se aplicaran sobre la base de datos del Volcán de Fuego de Colima, descrita en el capítulo 5.

En la sección 7.4 se comenzará describiendo la configuración y el ajuste de cada arquitectura. En la sección 7.5 se analizará la impacto del pre-entrenamiento no supervisado como una de las fases de construcción en el rendimiento final de los sistemas. Con el objetivo de medir la robustez de nuestra propuesta, en la sección 7.6 compararemos los resultados obtenidos con otras técnicas presentes en el estado del arte, como las Máquinas de Vector Soporte, Random Forest, Modelos de Mezclas de Gaussianas y Modelos Ocultos de Markov.

Finalmente, conociendo la relevancia que ciertos eventos tienen frente al resto en una erupción volcánica, en la sección 7.6.2 evaluaremos y analizaremos la confiabilidad (la emisión de probabilidades de pertenencia) de nuestra propuesta para cada tipo de evento, poniendo de manifiesto que desde un punto de vista geofísico, en el que la evaluación de los sistemas no solo se basa en un mejor porcentaje de reconocimiento, sino también en una mayor medida de confianza en el reconocimiento, los sistemas de aprendizaje profundo basados en abstracciones jerárquicas se presentan como una seria alternativa.



(a) t-SNE parcial (b) t-SNE total

Figura 7.2.1: Análisis t-SNE (t-Distributed Stochastic Neighbor Embedding) asociado al corpus de datos del volcán de Fuego, Colima. Las gráficas describen los datos haciendo uso de la Componente-1 frente a la Componente-2. a) Análisis t-SNE usando un subconjunto del total de clases de eventos. b) Análisis t-SNE usando todas las clases eventos

7.1. Criterios de evaluación

Como ya vimos en la sección 2.2.4, aunque en el aprendizaje automático existen muchas métricas para evaluar el rendimiento de un sistema de clasificación, en este trabajo nos hemos centrado en dos de las más ampliamente usadas: **Accuracy o rendimiento base y F_1-Score** (basado en la precisión y fiabilidad de los sistemas)**.**

Es importante destacar que los sistemas de reconocimiento en el ámbito de la sismología volcánica deben ser evaluados desde un punto de vista geofísico:

- Por un lado, no todos los tipos de eventos tienen la misma trascendencia, por lo que la calidad del reconocimiento podría ponderarse en función de la relevancia de los objetos reconocidos con éxito en una crisis eruptiva.

- Por otro lado, no todos los sistemas clasifican con el mismo grado de confiabilidad, es decir, no emiten la misma probabilidad de pertenencia, por lo que un sistema que clasifique los eventos de forma más confiable (con probabilidades más altas de pertenencia a las clases) aun teniendo el mismo rendimiento que otro, primará como elección en un observatorio vulcanológico.

En este sentido, además de un análisis basado en el rendimiento, nuestra propuesta será también evaluada por la confianza de sus clasificaciones.

7.2. Exploración de los datos

La naturaleza de las señales sismo-volcánicas y el entorno en el que se registran reducen la información característica de cada evento y por tanto, dificultan su clasificación. En este sentido, sería conveniente realizar un estudio o análisis de los resultados basándonos en cada uno de los tipos de eventos.

Para ello, nos apoyaremos en la técnica t-SNE (t-Distributed Stochastic Neighbor Embedding)[137], diseñada para reducir la dimensionalidad de los datos y que es especialmente útil en tareas de visualización y obtención de información significativa sobre la distribución de datos en espacios no lineales.

En esta técnica, los datos altamente dimensionales (con cientos de características) se proyectan en espacios de una dimensión muy reducida (2 o 3 dimensiones), permitiendo así una mejor visualización. Esto se consigue construyendo dos distribuciones de probabilidad y disminuyendo la divergencia de Kullback-Leibler entre ellas: la primera distribución P, se construye sobre pares de objetos de alta dimensión, de tal manera que los objetos similares tienen una alta probabilidad de pertenecer a la misma distribución, mientras que los objetos diferentes tienen una probabilidad asignada de pertenencia a esta distribución, mucho menor. La segunda distribución Q, recoge los puntos en el mapa de baja dimensión. Una vez las distribuciones están creadas, se minimiza la divergencia Kullback-Leibler entre la distribución de probabilidad conjunta de los puntos de datos en el espacio de alta dimensión, P y la distribución de probabilidad conjunta Q de los puntos de representación en el espacio dimensional mas sencillo, obteniendo así una disposición diferente en el espacio de representación para cada tipo de objeto.

La Figura 7.2.1 describe el análisis t-SNE de un subconjunto del total de clases (Figura 7.2.1a) y el total de clases (Figura 7.2.1b) que componen nuestro corpus de datos.

Aunque en las próximas secciones se llevará a cabo un estudio detallado de cada evento, en esta sección extraeremos algunas conclusiones básicas los datos:

- A partir de la Figura 7.2.1a, se observa que los eventos cuyas características están bien definidas pueden ser fácilmente separables, siendo el ruido el más eficazmente capturado. De forma general, los VTEs suelen ser confundidos con LPEs debido a la atenuación de las altas frecuencias durante la propagación de la propia señal y a la caída exponencial de su energía e incluso con REG si estos se producen a muy amplias distancias. Observando dicha figura, se distingue la frontera de los VTEs entre los LPEs y los REGs. No obstante, al tratarse de eventos locales a la sismicidad volcánica, la confusión entre VTEs y LPEs se traduce en puntos de características dispersos entre ambas clases en el espacio de representación.

- Al observar la Figura 7.2.1b, se puede apreciar como la delimitación de fronteras entre clases adyacentes en el espacio de características no está tan clara cuando se tienen en cuenta todas las clases de eventos. Además de las conclusiones anteriormente señaladas, en esta figura se observa que los ruidos se entremezclan tanto con TRE como con COL. Este hecho podría estar motivado en la naturaleza del ruido. Cuando éste tiene una componente de alta frecuencia es fácil asociarlo con ruido cultural, ambiental e incluso con COL. Sin embargo, cuando el ruido tiene componentes en bajas frecuencias, es difícil distinguirlo de los TRE, ya que su única diferencia suele ser la amplitud pico a pico entre ambas señales.

7.3. Metodología experimental

La metodología seguida para el desarrollo de los experimentos realizados se puede describir en los siguientes puntos:

- Como norma general para todos los problemas de clasificación abordados en este trabajo, el corpus de datos, una vez pre-procesado y parametrizado, será se dividido en dos conjuntos: conjunto de entrenamiento (75 %) y de test (25 %). En el caso del corpus de datos correspondiente al volcán de Fuego de Colima, el total de instancias que componen el conjunto de entrenamiento es de 7000.

- Con el objetivo de evitar el sobre entrenamiento o sobre ajuste de los modelos, además de la regularización (sección 3.5), se ha utilizado el criterio de *early stopping* (sección 3.5.3), que detiene el proceso de entrenamiento si no se consiguen mejoras en la validación de los modelos durante un número predefinido de iteraciones. Para ello, es necesario disponer de un conjunto de validación con el que evaluar si los modelos deben o no seguir siendo entrenados antes de ser finalmente testeados. El conjunto de validación se extrae del conjunto de test, por lo que ambos contienen un 12.5 % del total de los eventos del corpus de datos (1166). Las configuraciones que mejor rendimiento obtienen en el conjunto de validación son los que finalmente se analizan y evalúan en el conjunto de test.

- Dado que el tamaño de los corpus de datos de los que disponemos no es muy extenso, la evaluación de la capacidad de generalización de los modelos propuestos se debe realizar mediante la técnica de validación cruzada (*Cross Validation*)[83]. En este sentido, el conjunto de datos se ha dividido en 4 subconjuntos que posteriormente irán siendo rotados, de manera que tres de ellos se dedicarán a entrenamiento (75 %) y el restante a test y validación (25 %). Cada modelo es entrenado y testeado con cada uno de estos subconjuntos, siendo el resultado de clasificación final la media de los cuatro resultados de test.

- Con el objetivo de comparar la capacidad de generalización de los modelos propuestos, se han incluido experimentos con cuatro clasificadores de naturaleza muy diferente a los neuronales.

    - Por un lado, se ha decidido hacer uso de Máquinas de Vector Soporte (con núcleos lineales y radiales), Modelos de Mezclas de Guassianas (GMM) y Modelos Ocultos de Markov (HMM) (ver sección 2.3) ya que han sido ampliamente usadas en este aréa para clasificar señales sismo-volcánicas [52, 109, 108, 16, 97, 141, 181].
    - Por el otro, se optó por el uso de Random Forest, teniendo presente el creciente éxito experimentado recientemente y su estado actual del arte en áreas como el Big Data, Data Mining o Remote Sensing. A diferencia de los modelos profundos, estos tipos de clasificadores pueden ser ejecutados en plataformas no específicas (GPU- Graphics Processing Unit) sin incrementar el tiempo de cómputo.

7.3.1. Requisitos tecnológicos

Los modelos neuronales han sido implementados en Theano [21], una librería desarrollada en Python que permite definir, optimizar y evaluar expresiones matemáticas que involucran matrices multidimensionales de manera eficiente. No obstante, dado el elevado número de operaciones matriciales que requiere tanto el entrenamiento como el testeo de los modelos profundos, es necesario el uso de hardware específico que reduzca el tiempo de cómputo.

El avance experimentado en las unidades de procesamiento gráfico (GPU- Graphics Processing Unit) durante el último lustro, las ha situado como herramienta básica en la resolución de cualquier problema basado en aprendizaje profundo, dando lugar a multitud de *wrapper* o subrutinas capaces de ejecutar código fuente de carácter no gráfico en este tipo de dispositivos. En este sentido, toda nuestra experimentación se ha llevado a cabo en dos GPUs:

- NVIDIA K40c (con 2880 núcleos CUDA, y 12 GB de memoria)
- NVIDIA GEFORCE GTX 1080 (con 2560 núcleos CUDA y 8 GB de memoria).

7.4. SDA y DBN como sistemas de clasificación

Teniendo presente tanto los resultados de clasificación obtenidos por las arquitecturas recurrentes y convolucionales en el corpus de datos de Colima, como las particularidades intrínsecas de este tipo de señales (sección 6.2), las redes neuronales clásicas con varias capas de transformaciones no lineales se postulan como una alternativa (dentro del marco Deep Learning) para clasificar eventos sismo-volcánicos de forma aislada.

Dado el tamaño del corpus de datos de Colima y el desbalanceo de sus clases, se ha optado por el pre-entrenamiento no supervisado y apilamiento de modelos más simples como alternativa de construcción de DNNs (sección 3.1) en vez de generar una DNN con un esquema de inicialización concreto.

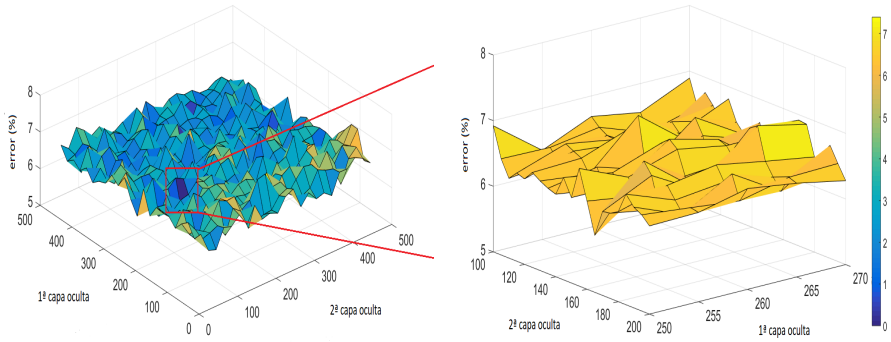
Se han considerado dos sistemas de clasificación: sDA (stacked Denoising AutoEncoder) basado en el apilamiento de modelos de carácter discriminativo (DA-Denoising AutoEncoder) y DBN (Deep Belief Networks) basado en el apilamiento de modelos de carácter generativo (RBM-Restricted Boltzmann Machine).

7.4.1. Configuración de la RBM como base de una DBN

El ajuste y configuración de una RBM como modelo base en la construcción de una DBN queda definido por el método de aprendizaje usado para inferir el conocimiento y el número de neuronas de la capa oculta. Como se argumentó en la sección 3.2.1, existen dos métodos con los que estimar el gradiente de una RBM: la divergencia contrastiva (CD) y la verosimilitud máxima estocástica o divergencia contrastiva persistente (PCD). En este trabajo se ha optado por la divergencia contrastiva dado su menor coste computacional y dado el carácter pre-inicializador de su objetivo.

No obstante, la estimación del gradiente de una RBM mediante la divergencia contrastiva tiene asociado el ajuste de varios hiperparámetros de vital importancia:

- **Tasa de aprendizaje:** es necesario definir una tasa de aprendizaje (learning rate) con el que modificar el ajuste de los pesos. Durante la fase de experimentación se evaluaron varios en el rango [0.000001,0.01], encontrando el mejor de ellos en 0.001.
- **Número de pasos del muestreo de Gibbs:** pese a que muchos trabajos solo aplican un solo paso $k = 1$, en lo que se conoce como Divergencia Contrastiva-1 ($CD_{k=1}$), en este estudio se han aplicado varios valores de k, encontrando $k = 10$ como mejor valor. Este valor de k puede ser explicado por el tamaño del corpus de datos. Cuando se disponen de grandes cantidades de datos, la aproximación de la distribución de probabilidad que describe los datos se puede alcanzar ejecutando



(a) Porcentaje de error obtenido en una parte del grid de búsqueda

(b) Porcentaje de error de clasificación obtenido en una subregión del entorno del óptimo local

Figura 7.4.1: Búsqueda de la DBN óptima con dos capas ocultas. a) Resumen de los porcentajes de error obtenidos en una rejilla de búsqueda de 500x500 unidades ocultas. b) Subregión de búsqueda entorno al óptimo local en una rejilla de [250,270]x[100,200] unidades ocultas.

un solo paso del muestreo de Gibbs. En cambio, cuando el conjunto de datos no es tan grande, es necesario ejecutar más pasos del muestreo, aproximando un poco mejor el valor esperado bajo la distribución definida por el modelo para así hacerlo converger.

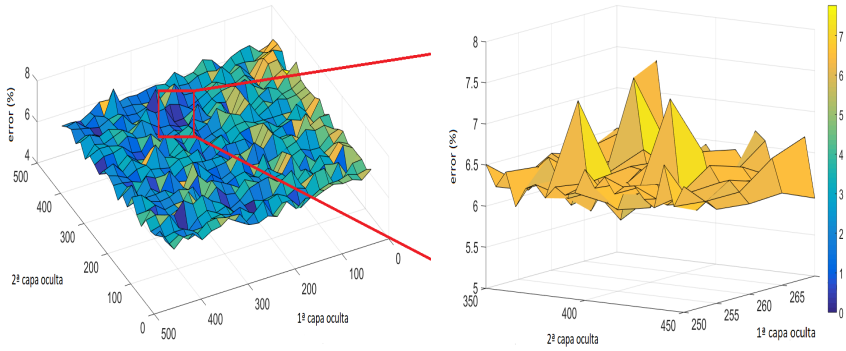
- **Número de épocas o iteraciones de pre-entrenamiento:** es otro hiperparámetro de mucha importancia, ya que regula el grado de entrenamiento de los modelos y por tanto la capacidad de representación de las características extraídas. Generalmente el número de iteraciones del pre-entrenamiento se asocia con la calidad de los predictores de características extraídos, que posteriormente serán de especial utilidad para discriminar entre clases. El número de iteraciones de pre-entrenamiento que mejor resultados ha ofrecido en nuestro corpus de datos ha sido 10. Un número de iteraciones por debajo de este valor ofrecía rendimientos más bajos, mientras que un valor superior no ofrecía mejoras significativas notables con respecto al incremento del tiempo de cómputo asociado.

- **Tamaño del batch:** dado el tamaño de nuestro conjunto de entrenamiento, y tras evaluar varios tamaños, se llegó a la conclusión de que el tamaño de batch óptimo era de 10 instancias.

Finalmente, dado que nuestros datos no son binarios, no se puede utilizar una RBM en la entrada, habiéndose optado por usar una Gaussian-bernoulli RBM (sección 3.2.1.2) como modelo base en la capa de entrada en la construcción de nuestros modelos.

7.4.2. Configuración del DA como base de un SDA

Al igual que ocurría con la RBM, el ajuste y configuración de un DA como modelo base en la construcción de un sDA queda definido por el método de aprendizaje usado



(a) Porcentaje de error obtenido en una parte del grid de búsqueda .

(b) Porcentaje de error de clasificación obtenido en una subregión del entorno del óptimo local

Figura 7.4.2: Búsqueda del sDA óptimo con dos capas ocultas. a) Resumen de los porcentajes de error obtenidos en una rejilla de búsqueda de 500x500 unidades ocultas. b) Subregión de búsqueda entorno al óptimo local en la rejilla de [250, 270] x[350,450] unidades ocultas.

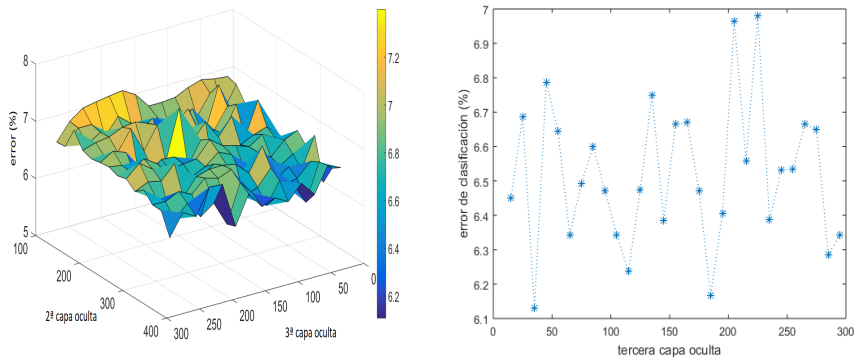
para inferir el conocimiento y el número de neuronas de la capa oculta (sección 3.2.2).

La minimización de la función de error o reconstrucción de un DA tiene asociado el ajuste de varios hiperparámetros:

- **Distorsión de los datos de entrada:** un 10 % de cada vector de entrada es distorsionado con ruido gaussiano aditivo, ya que supone una elección natural para entradas de valores reales. El error de minimización por tanto, se optimiza minimizando la entropía cruzada entre la salida del modelo y la entrada sin distorsionar [204].
- **Tasa de aprendizaje:** dado que se está trabajando con gradientes, es necesario definir una tasa de aprendizaje con el que modificar el ajuste de los pesos. Durante la fase de experimentación se evaluaron varios en el rango [0.000001,0.01], encontrando el mejor de ellos en 0.001.
- **Número de épocas o iteraciones de pre-entrenamiento:** como ocurría en el caso de las RBMs, el número iteraciones de pre-entrenamiento que mejor resultados ha ofrecido en nuestro corpus de datos ha sido 10. Un númeor de iteraciones por debajo de este valor ofrecía rendimientos más bajos, mientras que un valor superior no ofrecía ninguna mejora significativa.
- **Tamaño del batch:** Aprovechando el tamaño de batch óptimo obtenido en la experimentación con DBNs, el tamaño de batch con el que guiar el descenso de gradiente estocástico fue ajustado a 10 instancias.

7.4.3. Búsqueda de los modelos o configuraciones óptimas

Las mayores dificultades que surgen al diseñar DNNs son la elección del número óptimo de capas ocultas, el número óptimo de unidades por capa y el ajuste de la tasa de aprendizaje.



(a) (Porcentaje de error obtenido en una parte del grid de búsqueda (capa oculta 1=260)

(b) Porcentaje de error de clasificación obtenido en una subregión del entorno del óptimo local

Figura 7.4.3: Búsqueda de la DBN óptima con tres capas ocultas. a) Resumen de los porcentajes de error obtenidos en una rejilla de búsqueda de 300x400 unidades ocultas, habiendo fijado el número de unidades de la primera capa oculta a 260. b) Subregión de búsqueda entorno al óptimo local, habiendo fijado el número de unidades de la primera y la segunda capa oculta.

Algunos de los algoritmos tradicionales para resolver este problema se basan en la búsqueda en grid, búsqueda aleatoria o incluso la configuración manual llevaba a cabo por personal experto [20].

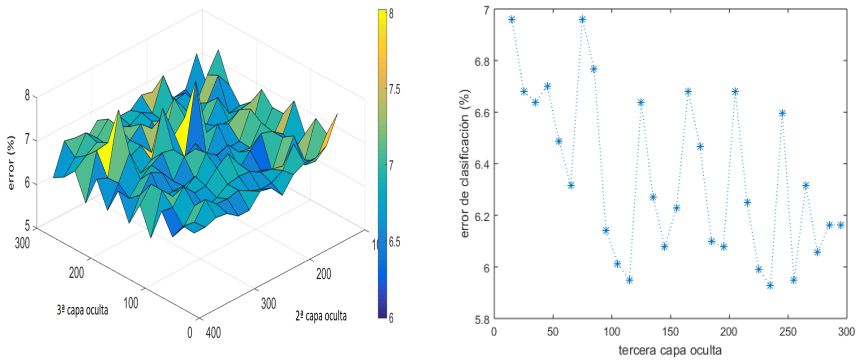
Debido al alto coste computacional y temporal requerido para entrenar DNNs, la optimización de los hiperparámetros y búsqueda del mejor modelo se ha llevado a cabo en forma de grid, tal y como propone [126]. Partiendo de una primera optimización bayesiana de los hiperparámetros [189], se fueron evaluando modelos de forma iterativa, llegándose a analizar un total de 250100 configuraciones, con hasta tres capas ocultas.

Para ambas arquitecturas (sDA y DBN), el número de unidades por capa varió desde 25 hasta 1250, con incrementos de 20. Las figuras 7.4.1 y 7.4.2 resumen la búsqueda en grid anteriormente citada.

Como se puede observar, la búsqueda encuentra un óptimo local para la DBN y el sDA en 250-165 y 260-385 unidades ocultas respectivamente:

- En el caso de la DBN, el error de clasificación varió en el rango 5.86 % al 7.39 %. A partir de la figura 7.4.1a se puede observar cómo la configuración de unidades por capa afecta enormemente a la tasa de reconocimiento final obtenida. Desde el punto de vista del aprendizaje automático, estas cambiantes tasas de error pueden estar motivadas por la falta de datos, lo que implica que unas configuraciones aproximen mejor que otras la distribución de probabilidad conjunta entre los datos y el modelo, y con ello, la probabilidad de éxito en la clasificación.

- En el caso del SDA, el error de clasificación obtenido varió entre 5.71 % y 7.75 %. Contrariamente a los resultados obtenidos por la DBN, los resultados obtenidos por el SDA no fueron tan fluctuantes (Figura 7.4.2a). Esta conclusión se asocia a la naturaleza discriminativa de su función objetivo. Al tratarse de un modelo que extrae una representación característica con la que reconstruir los datos de entrada, la diferencia entre configuraciones no tiene tanta repercusión en la tasa



(a) Porcentaje de error obtenido en una parte del grid de búsqueda (capa oculta 1=260)

(b) Porcentaje de error de clasificación obtenido en una subregión del entorno del óptimo local

Figura 7.4.4: Búsqueda del sDA óptimo con tres capas ocultas. a) Resumen de los porcentajes de error obtenidos en una rejilla de búsqueda de 300x400 unidades ocultas, habiendo fijado el número de unidades de la primera capa oculta a 260. b) Subregión de búsqueda entorno al óptimo local, habiendo fijado el número de unidades de la primera y la segunda capa oculta.

final de reconocimiento, ya que la información discriminativa que se extrae en las diferentes configuraciones es similar. No obstante, se pueden apreciar algunos picos espurios en los resultados. Estos picos pueden estar motivados nuevamente en la falta de datos e incluso en configuraciones que han aprendido a modelar ruido, lo que afecta directamente a su capacidad de generalización.

En el caso de configuraciones con tres capas ocultas, debido al elevado número de parámetros, fue necesario reducir el análisis exploratorio, por lo que número de unidades por capa oculta se disminuyó, llegando a explorar configuraciones de hasta 400 unidades ocultas. Los mejores resultados se obtuvieron en configuraciones vecinas a las encontradas con dos capas ocultas. Las figuras 7.4.3 y 7.4.4 resumen la búsqueda en grid habiendo fijado el número de unidades ocultas de la primera capa a 260. Como se puede observar, los modelos óptimos para la DBN y el sDA se encuentran con 260-385-35 y 260-385-235 unidades ocultas.

- En el caso de la DBN, el error de clasificación obtenido varió entre el 6.13 % y el 7.79 %. Analizando la Figura 7.4.3a, se observa cómo la curva de superficie que relaciona las unidades ocultas de la segunda y tercera capa ocultas(habiendo fijado las unidades ocultas de la primera) sigue oscilando en un rango de valores relativamente amplio debido a la posible falta de datos para poder aproximar estrechamente la distribución conjunta de probabilidad. En cambio, en el caso del SDA, aunque oscila, el rango en el que lo hace es menor, entre el 6.6 % y el 7.3 %. Nuevamente esto se debe al carácter discriminativo del modelo. Es importante destacar, que de forma similar a lo que ocurría con solo dos capas ocultas, la inclusión de una tercera también produce picos espurios que disparan el error hasta 8.10 %, pero en lineas generales, la curva de superficie asociada al SDA oscila en menor magnitud que la asociada a la DBN.

La tasa de aprendizaje asociada al proceso de optimización posterior al pre-entrenamiento

	Acc. Global 50 %	Acc. Global 75 %	Acc. Global 100 %
DNN-H2-Glorot	92.22±0.6	92.71±0.43	93.31±0.58
DBN-H2-preTra	**93.06±0.97**	**92.83±0.7**	**94.04±0.68**
DNN-H2-Glorot	**91.83±0.82**	92.77±0.72	93.17±0.66
sDA-H2-preTra	91.21±1.4	**92.83±1**	**94.32±0.66**
DNN-H3-Glorot	91.58±0.64	92.35±0.69	93.24±0.74
DBN-H3-preTra	**92.2±0.57**	**92.8±0.44**	**93.87±0.69**
DNN-H3-Glorot	91.67±0.52	92.63±0.63	93.09±0.55
sDA-H3-preTra	**92.09±0.76**	**92.97±0.53**	**94.1±0.68**

Tabla 7.1: Efectos de la inicialización con respecto al tamaño del conjunto de datos. Se han evaluado las mejores arquitecturas obtenidas mediante la búsqueda en grid, con 2 y 3 capas ocultas. Cada columna de la tabla corresponde con el tamaño del conjunto de datos. Los resultados obtenidos representan el porcentaje de aciertos de clasificación (accuracy).

no supervisado, se evaluó en el rango [0.000001, 0.01], encontrando su mejor valor en 0.04. El estudio comparativo y las conclusiones extraídas de este estudio se abordará en la sección 7.6.

7.5. La importancia del pre-entrenamiento

Con el objetivo de evaluar los efectos del pre-entrenamiento como esquema de inicialización en nuestro problema, se han realizado una serie de experimentos en los que se mide la capacidad de generalización de los modelos cuando se modifica el tamaño del corpus de datos.

Para ello, los mejores modelos encontrados mediante la búsqueda en grid han sido evaluados (con y sin la etapa de pre-entrenamiento) en conjuntos de datos cuyo tamaño representa el 50, el 75 y el 100 % del total de la base de datos.

Los modelos a los que se ha eliminado la etapa de pre-entrenamiento no supervisado y por tanto, su inicialización se ha llevado a cabo solo y exclusivamente a partir del esquema de Glorot (*Uniform*) [82] se referenciarán en este estudio como DNNs. Los modelos pre-entrenados de forma no supervisada, se referenciarán por tanto, como SDAs y DBNs

La Tabla 7.1 muestra el efecto del pre-entrenamiento en nuestro corpus de datos teniendo presente que:

- Todos los modelos han sido entrenados bajo el mismo método de optimización, Adam, descrito brevemente en la sección 3.3.4[128].
- El uso de la normalización por batch no tiene ningún efecto en nuestro problema, ya que los datos han sido normalizados en media y varianza durante la etapa de pre-procesado.
- Se ha evaluado una configuración de dropout siguiendo [193] con valor de probabilidad asociada a la poda p=0.2, no habiendo encontrado ninguna mejora en cuanto al rendimiento de los modelos.

A partir de este estudio se concluye que:

- Sólo hay un caso en el que los modelos no pre-entrenados se comportan mejor que los modelos pre-entrenados, y es cuando se usa un SDA con 2 capas y el 50 % de la base de datos. Esta tendencia cambia cuando aumenta el número de capas ocultas y el tamaño del conjunto de datos. Esto puede reflejar una mejora en la captura de características abstractas que posteriormente se traduce en una mejora relativa del rendimiento final del modelo. Dicha mejora puede explicarse en cómo los modelos están pre-entrenados. La RBM intenta aproximar la distribución de probabilidad que describe los datos a través de sus entradas. Para ello hace uso de la divergencia contractiva CD_k (CD-Contrastive Divergence), siendo más efectiva cuando los datos crecen. El DA intenta minimizar un error de reconstrucción y aprender tanto como sea posible de la distribución de datos. Las funciones de reconstrucción no se pueden aprender correctamente. En cambio, cuando hay más datos disponibles, se extraen características más representativas que a su vez optimizan las funciones de reconstrucción, obteniendo un mejor rendimiento general del sistema.

7.6. Estudio comparativo

El rendimiento general de los sistemas se evaluará mediante el porcentaje de reconocimiento obtenido. Debido a la gran cantidad de experimentos realizados (aproximadamente 250100), solo se presentarán los resultados de las mejores configuraciones. Cuatro serán los análisis que se realizarán:

1. En primer lugar, dado que algunos eventos tienen una mayor trascendencia con respecto a otros en la ocurrencia de una erupción volcánica, es necesario realizar un estudio pormenorizado del porcentaje de reconocimiento por tipo de evento. Para ello se analizarán las medidas de Precisión, Sensibilidad y F1.

2. En segundo lugar, siguiendo la filosofía clásica de los trabajos de investigación en el área del aprendizaje automático, los modelos propuestos serán evaluados frente a modelos base cuyo rendimiento ha supuesto mejoras en el estado del arte dentro del área de estudio como son las Máquinas de Vector Soporte, los Random Forest, los Modelos de Mezclas de Gaussianas y los Modelos Ocultos de Markov.

3. En tercer lugar, dado que estamos trabajando sobre un problema real que puede suponer un riesgo vital para las poblaciones que conviven en entornos volcánicos, se hace imprescindible la evaluación de los sistemas desde la confiabilidad de los mismos, es decir, el análisis de la clasificación no solo se basará en el porcentaje de reconocimiento sino también en la confianza o valor de probabilidad con el que se emiten las clasificaciones. Para ello, se definirán una serie de umbrales asociados a las probabilidades de pertenencia emitidas por la capa softmax (salida) desde los que se analizará e interpretará la confianza de las predicciones, aportando a los observatorios vulcanológicos conclusiones importantes a tener en cuenta a la hora de evaluar un sistema de clasificación.

4. Finalmente, evaluaremos los modelos propuestos en un escenario de clasificación real, en el que fijado un umbral de confianza (probabilidad mínima), se llevará a cabo un estudio en el que todas las clasificaciones que no superen dicho umbral serán descartadas. El uso de este umbral en las predicciones puede ser de gran

	Noise		*EXP*		*REG*		*COL*	
	PR	*RC*	*PR*	*RC*	*PR*	*RC*	*PR*	*RC*
GMM	*96.12*	*95.27*	*57.85*	*56.17*	*79.42*	*78.47*	*92.93*	*90.60*
HMM	***99.07***	*94.01*	*50.24*	***91.38***	*71.67*	*88.57*	*95.16*	***98.00***
SVM-Rad	*97.52*	*96.20*	*78.98*	*65.96*	*92.72*	*85.27*	*93.87*	*95.66*
SVM-Lin	*97.52*	*96.20*	***87.07***	*53.72*	*94.26*	*87.95*	*91.66*	*96.88*
RF-120	*97.62*	*95.47*	*86.57*	*61.70*	*92.09*	*88.39*	*93.53*	*96.07*
MLP-H1	*97.53*	*96.69*	*82.39*	*69.68*	*93.72*	*86.61*	*95.24*	*97.69*
DBN-H2	*97.20*	***97.92***	*82.39*	*69.68*	*92.27*	***90.63***	*97.03*	*97.56*
sDA-H2	*97.78*	*97.06*	*84.91*	*71.81*	***94.31***	*88.84*	*96.26*	*97.69*
DBN-H3	*97.55*	*97.43*	*82.82*	*71.81*	*91.47*	*86.16*	***97.16***	*97.42*
sDA-H3	*97.78*	*96.94*	*83.44*	*72.34*	*93.40*	*88.39*	*96.10*	*97.01*

	VTE		*TRE*		*LPE*	
	PR	*RC*	*PR*	*RC*	*PR*	*RC*
GMM	*88.28*	*90.96*	*82.49*	*82.48*	*91.98*	*92.29*
HMM	*85.97*	*95.22*	*89.98*	*66.67*	*93.10*	*86.84*
SVM-Rad	*93.03*	*93.77*	*85.9*	*83.39*	*91.91*	*95.9*
SVM-Lin	*92.75*	*94.11*	*85.22*	*76.82*	*89.45*	***96.29***
RF-120	*93.32*	*94.9*	*86.90*	*85.95*	*92.27*	*96.06*
MLP-H1	***93.68***	*95.70*	*87.25*	*86.13*	*94.03*	*95.66*
DBN-H2	*92.96*	*95.70*	*89.63*	*88.32*	*94.66*	*95.03*
sDA-H2	*93.03*	***96.72***	***89.64***	*89.96*	*95.11*	*95.11*
DBN-H3	*93.37*	*95.70*	*87.97*	*89.42*	*94.35*	*94.79*
sDA-H3	*92.79*	*96.15*	*88.41*	***90.51***	***95.56***	*94.95*

Tabla 7.2: Precisión (PR) y Sensibilidad (RC) por clase obtenidos para cada uno de los modelos propuestos así como para los modelos base con lo que los comparamos. Los valores están expresados en %.

ayuda en los observatorios vulcanológicos, ya que si las predicciones son muy fiables, los expertos geofísicos podrán manejar más eficientemente la crisis eruptiva. Si por el contrario, las predicciones son muy poco fiables, los expertos se verán obligados a revisar cada uno de los eventos minuciosamente, decrementando la eficiencia de la gestión.

7.6.1. Rendimiento general de los sistemas propuestos

Un precursor sísmico es cualquier señal sísmica que antecede a una erupción y cuyo origen está relacionado con los procesos dinámicos (fluidos) que la ocasionan (VTE, LPE, TRE). Como adelantamos en la sección 1.2.1, en diferentes escenarios eruptivos, en función de sus características (reología, morfología de la estructura volcánica, posición y origen de la fuente magmática) se pueden tener diferentes procesos en los que se podrán encontrar patrones comunes y diferenciadores de cada uno.

Por tanto, la identificación segura de los eventos considerados precursores es de vital importancia, ya que a partir de ellos se podrá determinar o al menos aproximar, una posible alerta temprana.

Antes de comenzar el análisis de los resultados, describiremos brevemente el ajuste de los modelos base que usaremos para comparar:

- Perceptron Multicapa-MLP (con sólo una capa oculta), en el que la optimización del número de neuronas de la capa oculta se realizó en el rango [10 a 2000], encontrando su mejor configuración en 500. El learning rate asociado al perceptrón se ajustó a partir de una optimización bayesiana [189] seguida de una búsqueda aleatoria en los contornos vecinos. El mejor resultado se obtuvo con un learning rate de 0.04.

- Gaussian Mixture Model (GMM), en el que se evaluaron hasta 151 gaussianas, encontrando el mejor resultado con 44.

- Máquinas de Vector Soporte (SVM), implementándose tanto la versión lineal como la versión radial.

- Random Forest (RF), en el que mediante una búsqueda en grid se evaluaron diferentes números de estimadores (hasta 500), encontrando el mejor resultado con 120.

- Hidden Markov Models (HMM), para el que se dispusieron dos enfoques diferentes:

    - Por un lado, para encontrar una configuración competitiva única o de propósito general, evaluamos el modelo con 5, 9 y 11 estados ocultos. Las probabilidades de emisión de un vector de características en cualquier estado, se modelaron evaluando hasta con 16 gaussianas multivariantes con matrices de covarianza diagonales. El mejor resultado se obtuvo con 10 estados y 16 gaussianas por estado.
    - Por otro lado, teniendo en cuenta la gran variabilidad de las señales, también consideramos la implementación de un HMM en el que la diferencia con respecto a la configuración anterior, es que ahora el número de estados del modelo es distinto para cada tipo de evento. Como hemos descrito anteriormente, para modelar las probabilidades de emisión en cualquier estado, se evaluaron hasta 16 gaussianas multivariadas con matrices de covarianza diagonales. Los resultados obtenidos con este enfoque no mejoraron a los obtenidos con la configuración de propósito general anterior.

Los resultados de clasificación obtenidos por cada modelo en su configuración óptima están recogidos en la Tabla 7.2. Cada columna de la tabla describe la precisión y la sensibilidad asociados a cada tipo de evento. A continuación se incluye un análisis detallado de estos resultados:

7.6.1.1. Análisis por arquitectura

A partir de los resultados de la Tabla 7.2 y la Figura 7.2.1, se observa que:

- Las arquitecturas o modelos base, GMM, HMM, RF, SVM y MLP, generalmente obtienen resultados equiparables a los obtenidos por las DNNs en aquellos eventos cuyas características están bien definidas, como son los NOISE, REG, VTE Y LPE. Apoyándonos en la Figura 7.2.1a, podemos concluir que esto se debe a la separabilidad implícita que existe en el espacio de representación de características de estos eventos y que estas arquitecturas modelan exitosamente.

	NOISE	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Glob
GMM	95.69	56.86	78.81	91.74	89.59	82.57	92.13	83.91	89.31±0.18
HMM	96.47	64.83	79.16	96.56	90.36	76.53	89.85	84.82	89±0.44
RF-120	96.53	71.88	90.13	94.78	94.10	86.41	94.13	89.71	92.80±0.61
SVM-Lin	96.85	66.39	91.03	94.19	93.42	80.80	92.75	87.92	91.55±0.80
SVM-Rad	96.86	71.86	88.81	94.75	93.39	84.63	93.87	89.17	92.32±0.76
MLP	97.11	75.33	90.03	96.45	94.68	86.67	94.85	90.73	93.57±0.70
DBN-H2	**97.56**	75.42	91.43	**97.29**	94.30	88.98	94.85	91.40	94.04±0.68
sDA-H2	97.41	77.78	**91.51**	96.97	94.84	**89.78**	**95.11**	**91.92**	**94.32±0.66**
DBN-H3	97.00	77.00	89.00	97.00	**95.00**	89.00	95.00	91.29	93.87±0.69
sDA-H3	97.00	**78.00**	91.00	97.00	94.00	89.00	95.00	91.57	94.10±0.68

Tabla 7.3: F1_Score por clase obtenido usando Precisión y Sensibilidad. Rendimiendo global (Acc) expresado en %.

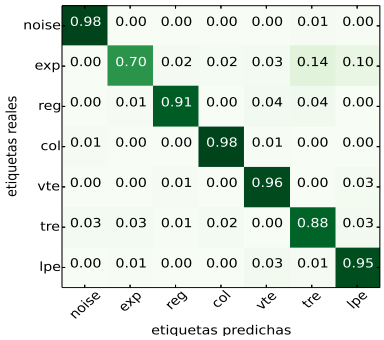
- Aunque en líneas generales, los porcentajes en términos de precisión (PR) o especificidad comparados con las DNNs son muy similares para todos los eventos, existe una pequeña diferencia con respecto a estos cuando se mide la sensibilidad (RC) de las clasificaciones. Esto se debe, como anteriormente hemos citado, a la separabilidad de los datos. Al observar la Figura 7.2.1b, se puede apreciar como la delimitación de fronteras entre clases adyacentes en el espacio de características no está tan clara cuando se tienen en cuenta todas las clases de eventos. Por tanto:

  - La representación de fronteras entre clases mediante funciones no complejas llevado a cabo por los modelos base, no se ajusta tanto a la forma en la que se disponen los datos, y por consiguiente, reduce el porcentaje de elementos que debieron ser correctamente clasificados y que posiblemente pertenezcan o se ubiquen en la proximidad de dichas fronteras.
  - Este hecho se ve mejorado en el caso de las DNNs, en los que el modelado de fronteras se realiza mediante funciones complejas, permitiendo un mejor ajuste y por tanto, un incremento de la sensibilidad.

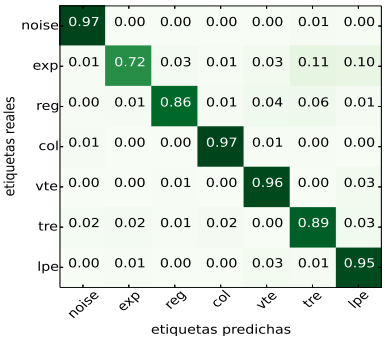
7.6.1.2. Análisis por eventos

Para realizar el análisis por eventos nos apoyaremos tanto en las matrices de confusión de la Figura 7.6.1, como en el análisis t-SNE de la Figura 7.2.1b, anteriormente expuesto.

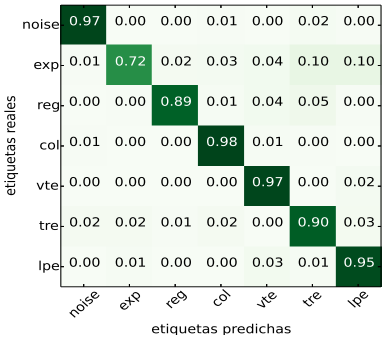
- Ruido: de forma general el ruido es reconocido con una alta precisión y una alta sensibilidad por todos los sistemas. De las matrices de confusión se extrae que el ruido es principalmente confundido con coladas (Col) y tremors (TRE), conclusión que además se puede corroborar observando el análisis t-SNE. No obstante, se puede apreciar que las confusiones con respecto al TRE son mucho más numerosas. Una explicación a esta situación podría ser la degradación de la amplitud pico a pico de las señales relacionada con los efectos de atenuación descritos en la sección 1.3. Cuando el Ruido tiene una componente de alta frecuencia es fácil asociarlo con ruido cultural (actividades humanas) o incluso ambiental (viento). Sin embargo, a bajas frecuencias, es difícil distinguir directamente entre Ruido y TRE, excepto cuando se aplican técnicas de avanzadas como arrays sísmicos o análisis de polarización.



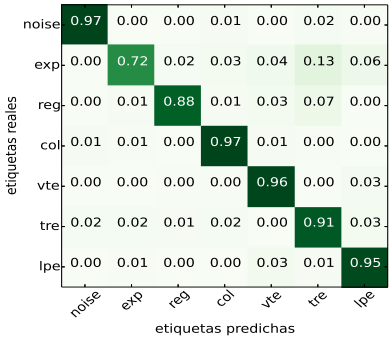
(a) DBN-H2



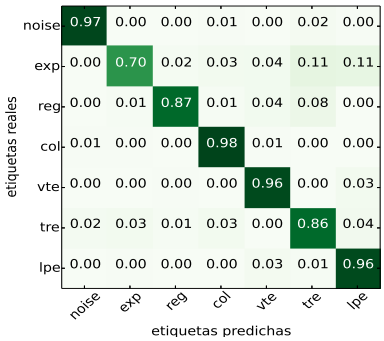
(b) DBN-H3



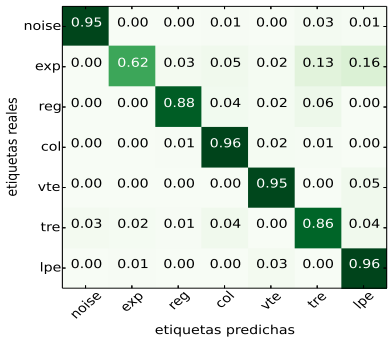
(c) sDA-H2



(d) sDA-H3

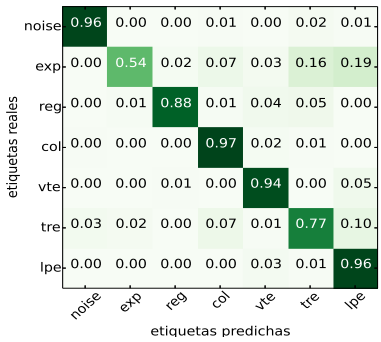


(e) MLP-H1

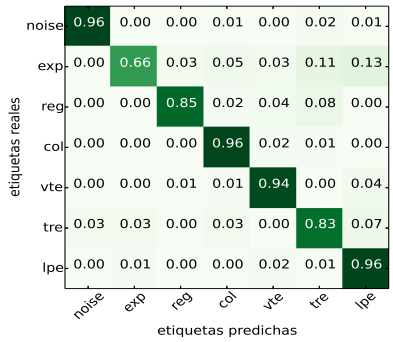


(f) RF-120

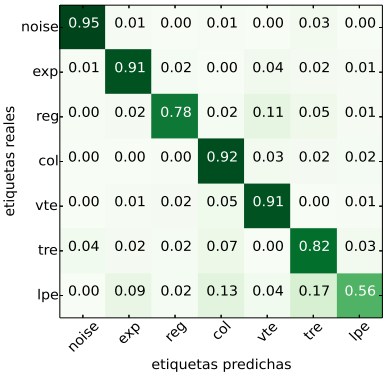
Figura 7.6.1: Matrices de confusión normalizadas asociadas a las arquitecturas implementadas. Los resultados corresponden con la precisión (ecuación 2.2.14) promedia obtenida en los cuatro conjuntos de tests usados en la validación cruzada.



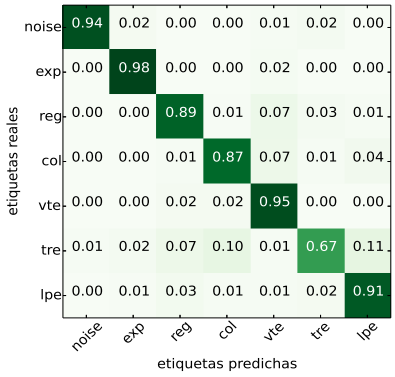
(g) SVM-Lin



(h) SVM-RBF



(i) GMM



(j) HMM

Figura 7.6.1: Matrices de confusión normalizadas de las arquitecturas implementadas. Los resultados corresponden con la precisión (ecuación 2.2.14) promedia obtenida en los cuatro conjuntos de tests usados en la validación cruzada.

- EXP: la menor sensibilidad obtenida por los modelos en la clasificación de EXP se puede interpretar como la consecuencia de la superposición de varias señales implícita al proceso explosivo y que deriva en la aparición de flujos de lava y/o desprendimientos de rocas, emisión de gases, etc. Esta superposición de señales afecta directamente a la eficacia de los sistemas, aumentando la cantidad de falsos positivos e insertando más errores en todas las clases. Si se observa el análisis t-SNE se concluye que las explosiones se disponen heterogéneamente en el espacio de representación. Esta conclusión se puede apreciar de forma clara en las matrices de confusión, en las que sistemáticamente las EXP son confundidas con eventos de muy distintas características, siendo nuevamente el TRE el más numeroso. Desde un punto de vista geofísico, las numerosas confusiones con respecto al TRE estarían motivadas en la duración y el contenido espectral de ambos eventos, que de forma general son muy similares. Las EXP tienen una gran componente de TRE asociada a la emisión de gas, que puede llegar a durar varias decenas de minutos, siendo la información visual asociada al proceso eruptivo la variable de más peso en el proceso de discriminación entre ambas señales.

- REG: pese a que los terremotos regionales son muy característicos, la sensibilidad con la que los modelos los clasifican se ve afectada. En las matrices de confusión se observa un error medio de confusión de en torno al 7 % con respecto al TRE. Nuevamente, está confusión está motivada en los efectos de atenuación. Tras la llegada de los paquetes de ondas P y S, los terremotos (regionales y volcano-tectónicos) experimentan una caída exponencial de energía que generalmente se conoce como coda. Si los terremotos son lo suficientemente grandes (duración y magnitud) y se encuentran a centenares de kilómetros de la estación sísmica que los registra, el decaimiento exponencial de la coda adquiere un contenido espectral muy similar al del TRE. Observando el análisis t-SNE, se puede apreciar como algunos de los REG se solapan con los TRE en varias zonas del espacio de representación. La indeterminada y compleja frontera entre dichos eventos repercute directamente en el descenso de la sensibilidad de las clasificaciones.

- COL: al igual que el Ruido, las COL son clasificadas con una muy buena precisión y sensibilidad en todos los modelos. Observando las matrices de confusión se concluye que generalmente las COL se confunden con VTE, lo que podría estar motivado por la similitud del contenido espectral de ambas señales La llegada de los paquete de ondas impulsivas P y S se verán influenciadas por la profundidad de la fuente sísmica y por tanto, las muy altas frecuencias pueden ser atenuadas, llegándose a confundir con el contenido espectral de las COL.

- VTE-LPE: de forma general los terremotos volcano-tectónicos son reconocidos con una elevada sensibilidad. Esto hecho se debe a la especificidad de las características que lo describen. No obstante, los resultados en cuanto a la precisión disminuyen significativamente. Observando la Figura 7.2.1a, se puede apreciar como algunos de estos eventos están dispuestos en una zona del espacio de representación más característica de los LPE que de los propios VTE. Esta característica coincide con los porcentajes de confusión presentes en las matrices de confusión. Una posible explicación a este comportamiento es la distancia hipocentral a la que se encuentran las fuentes que los generan, del sensor que los registra. A medida que la diferencia de llegada de las ondas P y S se incrementa (mayor distancia hipocentral), disminuye el contenido espectral en altas frecuencias de la señal registrada. A partir de distancias cercanas a los 12 km (dependiendo del

modelo de velocidad) las diferencias espectrales entre VTE y LPE son mínimas, por lo que ambos eventos son fácilmente confundidos.

- TRE: los resultados de clasificación del TRE siguen la lógica expuesta en los eventos anteriormente estudiados. No obstante, el mayor porcentaje de TRE mal clasificados se corresponden con LPE. De forma general, los contenidos espectrales de ambos eventos son prácticamente idénticos, habiéndose llegado a describir un tipo específico de TRE como un conjunto encadenado de LPE, siendo la diferencia más característica entre ambos, la duración. Mientras un LPE dura apenas unos minutos, un TRE puede durar incluso días. En este sentido, el factor humano asociado al proceso de etiquetado de los datos juega un papel de vital importancia, ya que muchas veces, durante este proceso, cuando los TRE son ininterrumpidos y demasiado largos, no son aislados, sino que se etiquetan pequeños segmentos representativos de los mismos. Estos segmentos, como hemos citado anteriormente, tienen prácticamente la misma duración y el mismo contenido espectral que un LPE, por lo que los modelos tienden a confundirlos. Es importante destacar, que desde un punto de vista geofísico, este error en la clasificación suele ser despreciado.

7.6.1.3. Análisis por número de capas ocultas

Observando detenidamente la Tabla 7.3, podemos apreciar como el número de capas ocultas afecta notablemente al porcentaje de rendimiento en función de dos variables:

- Número de eventos por clase

- Tipo de red subyacente usada durante la etapa de pre-entrenamiento.

El incremento de capas ocultas en los sDAs mejora los resultados cuando el número de eventos de una determinada clase es reducido, es decir, cuando una clase no está lo suficientemente representada, el uso de varias capas de abstracción incrementa la capacidad de discriminación de los sistemas. En este sentido, cuando una clase está suficientemente representada, los resultados obtenidos aumentando el número de capas ocultas son similares, ya que las características en los primeros niveles de abstracción extraídas son muy representativas, es decir, las funciones de reconstrucción quedan muy optimizadas y son lo bastante discriminativas, por lo que un incremento en el nivel de abstracción no tiene mucho efecto. Esta conclusión se puede observar claramente en la Tabla 7.3, en la que los sistemas mejoran la clasificación de las EXP siendo la clase menos numerosa cuando se incrementa el número de capas ocultas.

Análogamente, para las DBN, cuando una clase no está lo suficientemente representada, el incremento de capas ocultas (o niveles abstracción), mejora los resultados de clasificación. La inclusión de mayores niveles de abstracción ayuda en la aproximación de la distribución de probabilidad que modela los subconjuntos de datos escasos. No obstante, aunque una clase esté lo suficientemente representada e independientemente de que la distribución de probabilidad quede bien aproximada con un menor número de capas ocultas, la inclusión de mayores niveles de abstracción incrementa la fiabilidad de las clasificaciones, es decir, al clasificar los diferentes eventos en sus respectivas clases, lo hace asignando una mayor probabilidad de pertenencia (sección 3.2.1.3).

En la sección 7.6.2 se realizará un estudio más detallado de este hecho.

	NOISE	EXP	REG	COL	VTE	TRE	LPE
	CE	CE	CE	CE	CE	CE	CE
MLP-H1	0.0299	0.2747	0.1147	0.0430	0.0541	0.1206	0.0703
DBN-H2	0.0171	0.1603	0.0669	0.0219	0.0402	0.0686	0.0356
sDA-H2	0.0202	0.2273	0.0980	0.0343	0.0506	0.0981	0.0511
DBN-H3	**0.0133**	**0.1117**	**0.0485**	**0.0140**	**0.0274**	**0.0480**	**0.0232**
sDA-H3	0.0153	0.2276	0.0586	0.0280	0.0423	0.082	0.0478

Tabla 7.4: Entropía cruzada (CE) por clase obtenida por los mejores modelos.

7.6.2. Confianza de las clasificaciones

Los resultados analizados previamente sugieren que de forma general, los clasificadores, independientemente de la arquitectura, obtienen buenos resultados cuando trabajan con eventos sismo-volcánicos de forma aislada. Pero estos resultados, además del punto de vista de porcentaje de reconocimiento, deben ser analizados desde un punto de vista geofísico, donde puede ser interesante conocer la confianza de la clasificación.

El análisis de eventos complejos, a menudo resuelto por personal experto, ayuda a entender la evolución de los volcanes en etapas eruptivas. Disponer de un sistema experto, capaz de clasificar además de eficaz, de forma muy fiable, aportaría a los observatorios mucha agilidad a la hora de gestionar alertas tempranas. En este sentido, un análisis de la fiabilidad o confianza de los clasificadores sobre la base de las probabilidades de pertenencia emitidas en su capa de salida ayudará a entender si los modelos propuestos presentan alguna ventaja en la detección de eventos confusos (superpuestos), pudiendo ser de gran utilidad para los observatorios y en consecuencia, un aporte importante en al área de estudio.

Dado el rendimiento obtenido (Tabla 7.3), en el que supera al resto de modelos (GMM, HMM, SVM y RF), tomaremos el MLP como modelo base con el que comparar la confianza de las predicciones. Para ello, haremos uso de las probabilidades obtenidas por la capa softmax (capa de salida). Dos son las aproximaciones más comunes para medir dicha confianza:

- Por un lado, aprovechando la Entropía Cruzada (Ecuación 2.2.7) como función de coste o error, se puede obtener una medida de la confianza de las predicciones a partir de la suma de las probabilidades de los eventos por clase, es decir, para cada clase, se suman los errores obtenidos por la Entropía Cruzada para cada evento. Como se puede observar en la Figura 2.2.4, a medida que aumenta la probabilidad de pertenencia asignada sobre la clase correcta, disminuye el valor de la Entropía Cruzada . Este hecho se puede constatar fácilmente en la Tabla 7.4, que refleja la medida de entropía para cada clase y las mejores configuraciones de cada arquitectura: las arquitecturas profundas presentan una menor entropía, siendo la DBN con tres capas ocultas, la arquitectura más fiable.

- Por otro lado, se calcula la CDF empírica de probabilidades de pertenencia a la clase dadas por el clasificador. Esto se hace calculando, para cada clase, la suma acumulada del histograma normalizado de probabilidades de pertenencia . Este enfoque presenta la ventaja de la representación gráfica, en la que se pueden apreciar entre otras cosas, los porcentajes de eventos etiquetados con probabilidades dadas.

La Figura 7.6.2 resume la función de distribución acumulada (CDF) de las probabilidades de pertenencia asignadas por clase. Observando la Figura 7.6.2c, se aprecia cómo las DNNs asignan mayores probabilidades de pertenencia a la clase explosión (EXP) comparada con el MLP. Más concretamente, en el mejor de los casos, el 60 % de los eventos etiquetados como explosión fueron clasificados por DBN-H3 con una probabilidad mayor del 80 %, mientras que el MLP, lo hizo con una probabilidad aproximada del 57 %. Algo similar ocurre en el caso de los tremors (TRE), donde las DNNs clasificaron el 80 % de los eventos con un 20 % más de confianza que el MLP.

De forma general, se puede apreciar que:

- En todos, los casos los modelos propuestos obtienen una mayor fiabilidad en la clasificación, siendo más pronunciada en aquellas clases en las que la superposición de eventos es más probable, es decir, en aquellas clases más confusas o complejas.

- Las DBNs obtienen casi siempre mejores resultados y más fiables que los sDAs. Una explicación a este hecho puede ser el método de pre-entrenamiento y su arquitectura subyacente. Dado que las RBMs aproximan la distribución de probabilidad que siguen los datos de entrada, las DBNs obtienen un conocimiento a priori que beneficia a la posterior emisión de probabilidades. Este hecho se puede constatar en las clases cuyo número de instancias es menor que el resto. Como no se dispone de suficientes instancias, el conocimiento a priori extraído por los DAs, aunque es lo suficientemente representativo, genera cierta incertidumbre, lo que afecta directamente a la confianza de la clasificación.

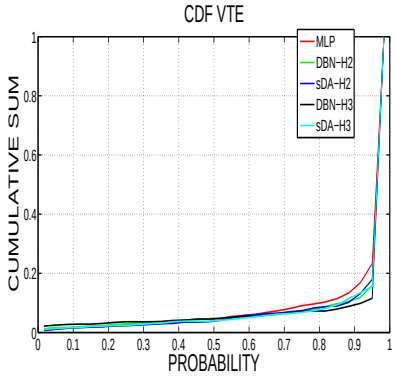
Otro apunte interesante a destacar de este análisis, es el potencial beneficio de aumentar el número de capas dependiendo de la arquitectura subyacente. Si observamos detenidamente todas las CDFs, comprobaremos cómo:

- La extracción de características discriminativas durante el pre-entrenamiento de los SDAs, siempre y cuando el número de instancias sea los suficientemente grande, deriva en probabilidades de pertenencia similares cuando aumentamos el número de capas ocultas, es decir, cuando se dispone de suficientes instancias, las probabilidades de pertenencia emitidas por un SDA con 2 capas ocultas y las emitidas por uno con 3 capas ocultas, son prácticamente las mismas. Esto se debe a la propia naturaleza de los modelos; cuando la información extraída es lo suficientemente representativa no se necesita aumentar la profundidad del modelo para obtener mejores resultados. Las Figuras 7.6.2a, 7.6.2e y 7.6.2d, resumen esta conclusión.

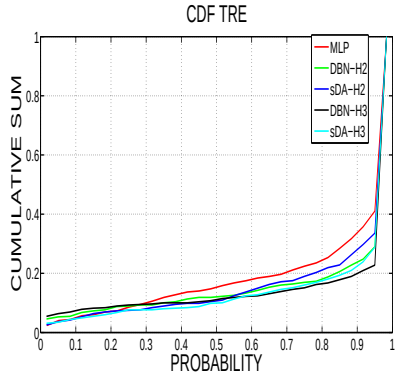
- En cambio, en las DBNs, el aumento de capas ocultas incrementa siempre la confianza de las del modelo. Como ya describimos en la sección 3.2.1.3, el incremento de niveles ocultos en una DBN mejora la aproximación de la distribución de probabilidad que describe los datos y por tanto, se incrementa la confianza de los modelos.

7.6.2.1. Modelos desde un punto de vista geofísico

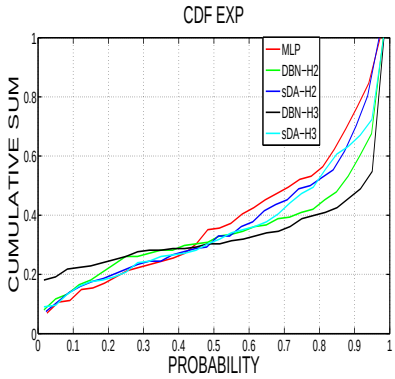
Teniendo presente que de forma general, en los observatorios, sería conveniente emplear umbrales de incertidumbre en los que basar la veracidad de las clasificaciones, se pueden considerar dos tipos de sistemas:



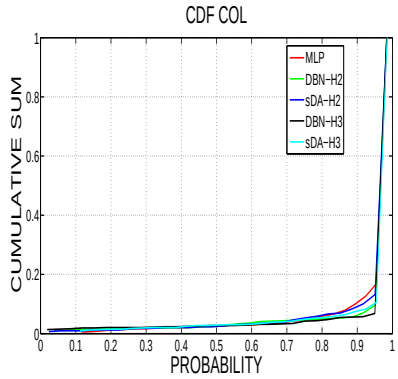
(a) Instancias clasificadas como VTE.



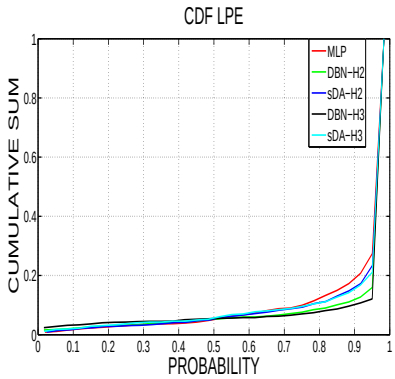
(b) Instancias clasificadas como TRE.



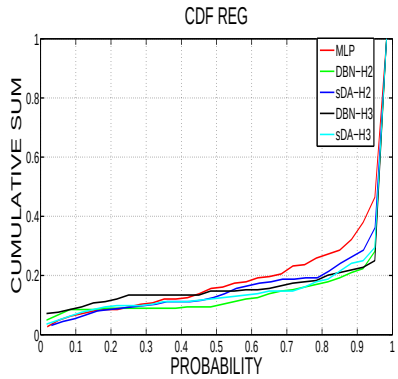
(c) Instancias clasificadas como EXP.



(d) Instancias clasificadas como COL.



(e) Instancias clasificadas como LPE.



(f) Instancias clasificadas como REG.

Figura 7.6.2: Función de distribución acumulada (CDF) para diferentes tipos de eventos. El eje x representa las probabilidades de clase asignadas por los modelos. El eje y representa la suma acumulada normalizada de eventos clasificados dentro de esa clase. Para un rendimiento óptimo del clasificador, el gráfico ideal tenderá hacia la probabilidad uno sobre el eje de probabilidades, ya que el vector de probabilidad de salida será menos disperso.

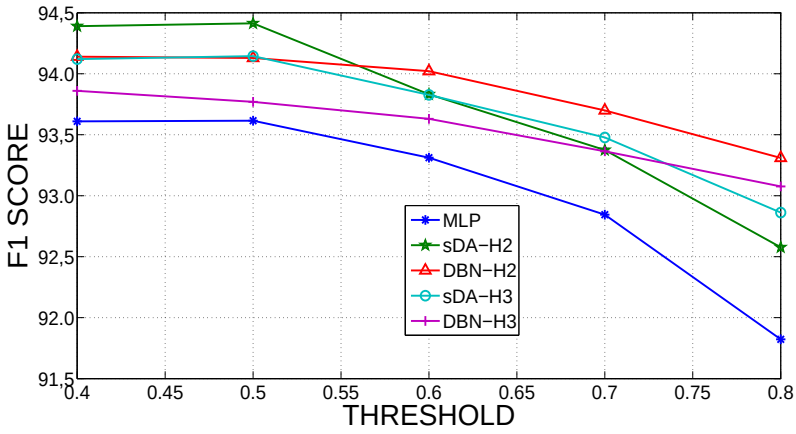


Figura 7.6.3: Estudio del rendimiento de los sistemas en términos de F1 score a medida que varía el umbral de incertidumbre.

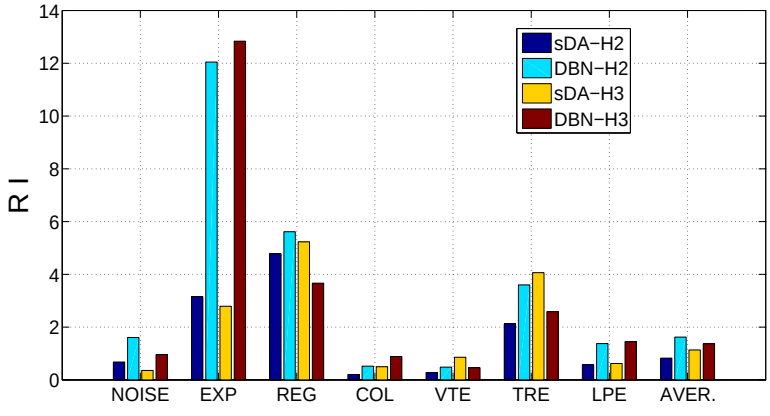


Figura 7.6.4: Mejora relativa por evento de las DNNs comparadas con el MLP para un umbral de confianza de 0.8.

- Para umbrales de probabilidad de pertenencia a la clase bajos, los sistemas serán muy sensibles, por lo que se detectarán muchos eventos con una probabilidad ligeramente superior al umbral, siendo muchos de ellos clasificados en una manera errónea.
- Con umbrales de probabilidad de pertenencia a la clase altos, los sistemas solo serán capaces de detectar señales muy claras, dejando fuera del reconocimiento aquellas señales cuya naturaleza sea difícil de caracterizar.

Como se puede apreciar en la Figura 7.6.2, todas las CDF pueden ser divididas en tres regiones de confianza (probabilidad) distintivas: una región baja en torno a 0.4, una región alta en torno al 0.9 y una meseta intermedia entre ambas. El estudio de la dispersión de los resultados en esta meseta nos da información de cómo los modelos profundos emiten las probabilidades de pertenencia. Con base en este criterio, se realizará un estudio del rendimiento de los sistemas, en término de F1 score, variando el valor del umbral entre 0.4 y 0.8, contemplando así tanto los modelos sensibles como los precisos.

Observando la Figura 7.6.3, se concluye que incluso para sistemas altamente sensibles (umbral 0.4), el rendimiento obtenido por las arquitecturas profundas supera casi en un 1 % al obtenido por el MLP. A medida que el umbral de probabilidad de pertenencia a la clase aumenta, la diferencia de rendimiento entre ambos tipos arquitecturas se distancia. Estos resultados confirman la hipótesis de partida: los modelos profundos, además de clasificar eventos muy característicos, como Ruido, VTE, COL y LPE, clasifican eventos complejos (EXP, TRE) con probabilidades más altas.

Otro apunte importante a destacar de estos resultados es el comportamiento de los sistemas DBN, ya que son los que menor decremento del rendimiento experimentan a medida que incrementamos el umbral de pertenencia. Esto, como ya argumentamos en la sección 7.6.2, se debe a la naturaleza de sus arquitecturas subyacentes, las cuales aproximan la distribución de probabilidad que siguen los datos de entrada (sección 3.2.1), beneficiando la posterior emisión de probabilidades y por tanto minimizando el efecto del uso de umbrales de pertenencia.

Siguiendo el criterio de varios expertos vulcanólogos y con el objetivo de evitar errores excesivos, se ha fijado el umbral de probabilidad de pertenencia en 0.8 y se han obtenido las mejoras relativas (RI- Relative Improvement) (ecuación 7.6.1) para cada clase de eventos de los modelos profundos sobre el modelo base de mayor rendimiento (MLP).

$$RI = \frac{Acc_{model} - Acc_{MLP}}{Acc_{MLP}} \tag{7.6.1}$$

Analizando la Figura 7.6.4, se puede observar que algunos de los eventos más complejos de discriminar (debido a la superposición implícita de eventos en su aparición), como son las explosiones (EXP) , presentan una mejora relativa (RI) del 13 % para la DBN-H3 y del 2.5 % para el sDA-H3. Esta mejora relativa tan pronunciada está motivada nuevamente por la naturaleza de las arquitecturas subyacentes que dan origen a los sistemas DBN. La emisión de probabilidades de pertenencia elevadas en eventos muy complejos o difíciles de discriminar forma parte del valor añadido de este tipo característico de sistemas.

Otro apunte interesante son las mejoras significativas de hasta un 4 %, conseguidas en ambas arquitecturas, en la detección de tremors volcánicos. Esta mejora es de especial relevancia ya que este tipo de eventos están presentes en todos los procesos eruptivos y su pronta detección supone una ventaja en la gestión de la futura alerta temprana.

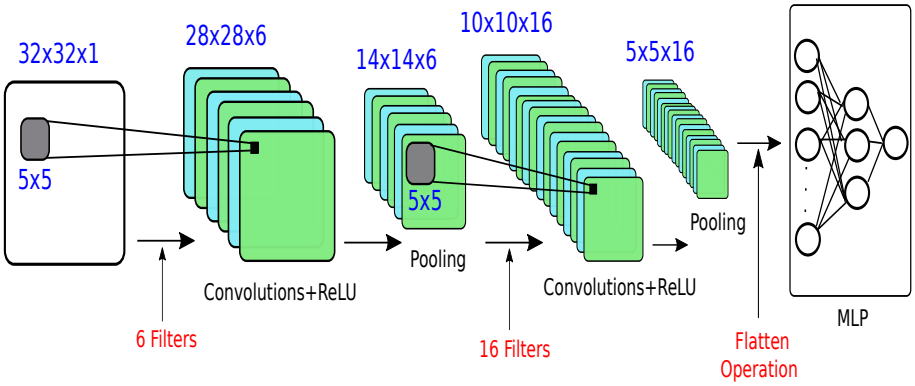


Figura 7.7.1: LeNet-5.

Modelo	Parametrización	Rendimiento (Acc. Global %)
DeepCNN-3H	Raw	55.13
DeepCNN-5H	Raw	46.04
CNN LeNet +128	Espectrograma	84.14
CNN LeNet +512	Espectrograma	92.88
CNN LeNet Based-2 FC	Espectrograma	78.83

Tabla 7.5: Resultados obtenidos clasificando datos sismo-volcánicos de Colima con modelos convolucionales, haciendo uso de arquitecturas previamente entrenadas con otros tipos de datos (Transfer Learning). Acc. Global corresponde con la media de estos resultados obtenidos en los cuatro conjuntos de test.

7.7. Transfer-Learning y CNN

Uno de los enfoques más ampliamente usados hoy en día en escenarios donde la cantidad de datos que describen el problema es reducida, es la transferencia inductiva (TL-Transfer Learning). Como ya argumentamos en la sección 2.1, el TL usa el conocimiento adquirido en la resolución de problemas en los que se dispone de una gran cantidad de datos en entornos donde la cantidad de datos es escasa [168, 209]. Generalmente, en este tipo de escenarios, los modelos pre-entrenados y ajustados en otros dominios son usados casi en su totalidad como solución al problema planteado, siendo necesario un simple ajuste de las capas finales haciendo uso del nuevo conjunto de datos con el fin de no incurrir en escenarios de sobreajuste.

Teniendo presente el éxito que las redes neuronales convolucionales (sección 3.4) han experimentado (dentro del área del Deep Learning) en campos como la visión por computador (CV), el procesamiento natural del lenguaje (NLP) o el reconocimiento automático de la voz (ASR), se planteó la idea de incorporar como base de nuestro sistema de clasificación alguno de los modelos que mejores resultados están ofreciendo en cualquiera de estos dominios, para finalmente ajustar con nuestro conjunto de datos las capas finales del mismo.

Este planteamiento se fundamenta en la idea de que las características de bajo nivel (primeras capas) contienen información genérica (detectores de bordes, detectores de regiones de colores, etc), mientras que progresivamente, las últimas capas van extrayendo características más específicas asociadas a la información que describe cada

una de las clases que figuran en el conjunto de datos con el son entrenadas. En este sentido, apoyándonos en el espectrograma como parametrización asociada a los datos, se planteó el uso de la red LeNet como modelo base [130], una red convolucional con 7 niveles de profundidad (propuesta en 1998) que clasifica dígitos manuscritos digitalizados en imágenes en escala de grises de 32x32 píxeles (Figura 7.7.1). La capacidad de procesar imágenes de mayor resolución requeriría un mayor número de capas, por lo que dado el contenido de los espectrogramas y los recursos hardware de los que se dispone, se ha optado por este modelo.

Como se puede apreciar en la Figura 7.7.2, a las señales, una vez han sido preprocesadas siguiendo el modelo propuesto en la sección 6.1, se les calculan sus espectrogramas, que posteriormente, durante la etapa de entrenamiento serán usados como vector de entrada al modelo (LeNet) en pequeños lotes de 10. La salida de dicho modelo está conectada o una o varias capas completamente conectadas (Fully Connected-FC) cuyos parámetros serán ajustados durante esta misma etapa. En este esquema, la red LeNet queda aislada durante la tarea de clasificación, siendo realmente relevante durante la extracción de características.

Es importante destacar, que con este método de trabajo, las operaciones de convolución se realizan sobre un espacio bidimensional (2-D), a diferencia de las convoluciones que se presentaron en la Tabla 6.1, en las que las operaciones de convolución se realizaban en el dominio del tiempo (1-D). Los mejores resultados obtenidos se han resumido en la Tabla 8.1. En dicha tabla también se han incluido los resultados obtenidos con las arquitecturas convolucionales en una dimensión (1-D) para facilitar la comparación entre ambos. Como se puede observar se han considerado tres modelos:

- CNN-LeNet +128: en la que a la salida del modelo se ha añadido una capa completamente conectada de 128 unidades ocultas.
- CNN-LeNet + 512: en la que a la salida del modelo se ha añadido una capa completamente conectada con 512 unidades ocultas.
- CNN-LeNet Based-2 FC: en la que a la salida del modelo se han añadido dos capas completamente conectadas de 128 unidades ocultas cada una.

A tenor de los resultados obtenidos, podemos concluir que el uso de CNN previamente ajustadas y más concretamente, la extracción jerárquica de características que implementan, pueden ser un buen punto de partida desde el que construir sistemas de clasificación fiables, aprovechando las características de invariabilidad y localidad que las operaciones de convolución nos ofrecen.

7.8. Conclusiones

Varias han sido las conclusiones extraídas tras un profundo análisis de las DNNs y su aplicabilidad al problema de la clasificación aislada de señales sismo-volcánicas.

- En primer lugar, se ha motivado la necesidad de una parametrización eficiente con la que guiar el proceso de clasificación, ya que la naturaleza y el tamaño de los corpus de datos de los que disponemos impiden el uso directo de registros volcánicos y por lo tanto, una aplicación directa de las últimas arquitecturas de aprendizaje profundo.
- En segundo lugar, se ha demostrado que tanto sDAs como DBNs son capaces de clasificar eventos sismo-volcánicos con mayor precisión y sensibilidad que los

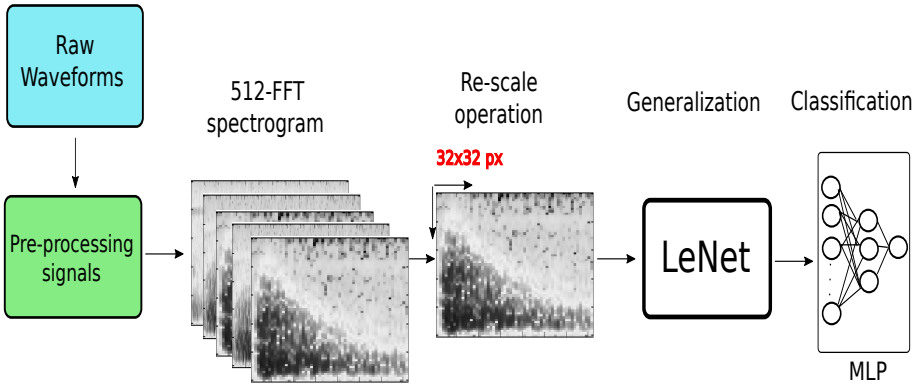


Figura 7.7.2: Descripción del esquema de trabajo propuesto haciendo uso del modelo LeNet.

arquitecturas clásicas, mejorando significativamente la detección de eventos complejos. El uso de varias capas de transformaciones no lineales proyecta los datos de entrada en un nuevo espacio de representación en el que pueden ser más fácilmente separables y por tanto en la construcción de funciones complejas con las que se modelar y adaptar mejor las fronteras de decisión entre clases dentro del espacio de representación. Esta mejora en la discriminación entre clases contribuye a la emisión de probabilidades de pertenencia más elevadas, repercutiendo directamente en una mayor fiabilidad de las clasificaciones.

- En tercer lugar, se ha confirmado que el uso de arquitecturas subyacentes diferentes durante la etapa de pre-entrenamiento repercute notablemente tanto en el rendimiento como en la confianza de los sistemas. Se concluye que cuando se dispone de suficientes eventos de una determinada clase, el incremento de capas ocultas no afecta al porcentaje de reconocimiento, ya que las funciones de reconstrucción (en el caso de los DA) y las distribuciones de probabilidad (en el caso de las RBM), están lo suficientemente bien aproximadas y por tanto las características extraídas tras la etapa de entrenamiento son lo suficientemente discriminativas que la inclusión de una nueva capa no mejora en gran medida los resultados obtenidos. En cambio, si que se observa una mejora en cuando a la confianza de los sistemas cuando se introduce una nueva capa de transformaciones no lineales. Por un lado, en el caso del apilamiento de DA, un mayor número de niveles de abstracción deriva como hemos dicho anteriormente, en un espacio de representación más fácilmente separable, y por tanto, las probabilidades de pertenencia se incrementan. Por otro lado, en el caso de apilamiento de RBM, siguiendo el razonamiento expuesto en [102] y argumentado en la sección 3.2.1.3, se puede concluir que la inclusión de nuevos niveles de abstracción, producirá aproximaciones de la distribución de probabilidad menos divergentes, incrementando las probabilidades de pertenencia emitidas.

- En cuarto lugar, tras un exhaustivo análisis de los resultados, apoyándonos en matrices de confusión y técnicas actuales de exploración visual de los datos, se ha observado que la gran parte de los errores de clasificación cometidos por los modelos se deben a la naturaleza de las señales sismo-volcánicas y especialmente a los efectos de fuente y atenuación. Este hecho pone de manifiesto la dificultad

implícita al problema que estamos abordando y la necesidad de crear corpus de datos mucho más completos y de mayor envergadura.

- En quinto lugar, partiendo de la conclusión inmediatamente anterior, se ha estudiado la confiabilidad de los sistemas propuestos imponiendo ciertos umbrales de confianza en las clasificaciones. Este estudio ha demostrado como la sensibilidad de los modelos base (MLP, SVM y RF) se ve seriamente afectada a medida que se exige una mayor especificidad de los mismos. Esta conclusión cobra especial relevancia ya que el uso de umbrales de pertenencia es una práctica muy común llevada a cabo en los observatorios vulcanológicos.

- En sexto lugar, se ha introducido el uso de modelos previamente entrenados en otros dominios como base de conocimiento y extractor de características (Transfer Learning) desde la que construir sistemas fiables y eficientes. Este punto necesita todavía de un estudio más profundo, ya que lo aquí abordado es una pequeña introducción y un sondeo de su uso sobre un dataset muy específico. A tenor de los resultados obtenidos se preveen mejoras significativas y avances importantes en esta línea de investigación.

Por tanto, cerramos este capítulo, concluyendo que la implementación (en entornos reales) de clasificadores basados en redes neuronales profundas como herramienta de monitoreo volcánico puede mejorar los sistemas actuales de alerta temprana.

Capítulo 8

Clasificación continua de eventos sismo-volcánicos mediante RNNs

La naturaleza de las señales sismo-volcánicas, la forma misma de adquirirlas y su posterior análisis en los observatorios, demandan sistemas capaces de trabajar en tiempo real. La clasificación de eventos aislados, aunque útil, sigue siendo insuficiente para gestionar crisis eruptivas y emitir alertas tempranas en tiempo real. El tiempo transcurrido desde que los eventos considerados precursores comienzan a manifestarse, hasta que finalmente el volcán entra en erupción, es finito y acotado. Una parte importante de este tiempo, se consume en el proceso de detección y delimitación de los eventos, reduciendo el tiempo de análisis y retrasando el lanzamiento de alertas, lo que finalmente se refleja en un incremento del riesgo ante una posible erupción.

Dada la estructura de los registros sísmicos, en los que el flujo temporal de la señal se presenta como un número indeterminado de eventos, y dado el coste tanto temporal como humano de su análisis, el diseño de un sistema capaz de detectar y clasificar en tiempo real (o cuasi-real) dicha información se convierte en una actividad prioritaria.

Desde el punto de vista del aprendizaje automático, la detección y clasificación de eventos sismo-volcánicos a partir de registros continuos de datos, es un problema secuencial que involucra señales de duraciones y características muy diversas que requiere arquitecturas capaces de capturar su evolución temporal. Hasta el momento (y siempre desde nuestro conocimiento), los sistemas de reconocimiento desplegados en observatorios capaces de trabajar en tiempo real, se basan en Modelos Ocultos de Markov (HMM) [52, 55].

En este capítulo, tomando como referencia los resultados obtenidos por los modelos neuronales en el problema de la clasificación aislada y aprovechando las mejoras obtenidas por las RNNs (modelos secuenciales pertenecientes al ámbito del Deep Learning) en diferentes áreas del conocimiento, se aborda la implementación de un sistema de detección y clasificación de eventos sismo-volcánicos desde tres arquitecturas diferentes: RNN-Vanilla, RNN-Long Short Term Memory y RNN-Gated Recurrent Units (sección 4.1).

En la sección 8.1 se presenta la metodología experimental seguida durante el desarrollo y evaluación de los sistemas. En la sección 8.2, definiremos una gramática que analizará a posteriori el conjunto de predicciones de las RNNs y adecuará sus etique-

tas. Con el objetivo de medir la robustez de estas arquitecturas, en la sección 8.3, compararemos los resultados obtenidos con otras técnicas presentes en el estado del arte, como los Modelos Ocultos de Markov. Finalmente, en la sección 8.4, mediremos la capacidad que tienen nuestras propuestas para adaptarse a las nuevas y cambiantes dinámicas del volcán. Para ello, los modelos entrenados y ajustados con corpus de datos serán evaluados en un corpus de datos pertenecientes a las campañas sísmicas antárticas de 1994–1995, 1995–1996 y 2001–2002, serán evaluados en un corpus de datos perteneciente a la campaña del año 2017.

8.1. Metodología experimental

- Siguiendo los criterios establecidos en la sección 7.3, el corpus de datos, una vez pre-procesado y parametrizado, será dividido en dos conjuntos: conjunto de entrenamiento (80 %) y de test (20 %). Con el objetivo de evitar el sobre entrenamiento o sobre ajuste de los modelos se ha utilizado el criterio de early stopping (sección 3.5.3).

- El conjunto de validación se extrae del conjunto de test, por lo que ambos contienen un 10 % del total de las instancias del corpus de datos. Los modelos que mejor rendimiento obtienen en el conjunto de validación serán los que finalmente se analizarán y evaluarán en el conjunto de test.

- Al igual que ocurría con el corpus de datos del Volcán de Fuego, Colima, el tamaño del corpus de datos de Isla Decepción del que disponemos no es muy extenso. En este sentido, para evaluar la capacidad de generalización de los sistemas es necesario hacer uso de la técnica de validación cruzada (Cross Validation) [83]. El conjunto de datos se ha dividido en 4 subconjuntos que posteriormente irán siendo rotados, de manera que tres de ellos se dedicarán a entrenamiento (75 %) y el restante a test y validación (25 %). Cada modelo es entrenado y testeado con cada uno de estos subconjuntos, siendo el resultado de clasificación final la media de los cuatro resultados de test.

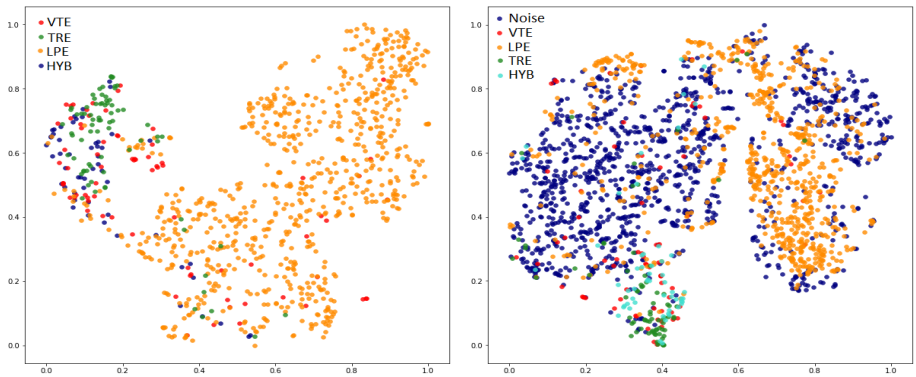
- Con el objetivo de comparar la capacidad de generalización de los modelos propuestos, se han incluido experimentos con todas las parametrizaciones expuestas en el capítulo 6.

- Dado que estamos trabajando en la construcción de un sistema en tiempo real, se hace imprescindible la evaluación del mismo mediante su capacidad de generalización. Para ello, los modelos serán testeados en un corpus de datos perteneciente a una campaña sísmica más reciente. Los resultados obtenidos serán comparados con los obtenidos por un Modelo Oculto de Markov entrenado y testeado en las mismas circunstancias.

8.2. Exploración de los datos

Al igual que en el caso de la clasificación en aislado, antes de proceder con la experimentación y análisis de los resultados, sería conveniente realizar un estudio de los datos.

La Figura 8.2.1 describe el análisis t-SNE de un subconjunto del total de clases (Figura 7.2.1a) y el total de clases (Figura 7.2.1b) que componen nuestro corpus de



(a) t-SNE parcial (b) t-SNE total

Figura 8.2.1: Análisis t-SNE (t-Distributed Stochastic Neighbor Embedding) asociado al corpus de datos de Isla Decepción, Antártida. Las gráficas describen los datos haciendo uso de la Componente-1 frente a la Componente-2. a: Análisis t-SNE usando un subconjunto del total de clases de eventos. b: Análisis t-SNE usando todas las clases eventos

datos. Además de una clara descompensación o desbalanceo entre las diferentes clases que componen el corpus de datos, algunas de las conclusiones básicas extraídas de estas imágenes son:

- En la Figura 8.2.1a, se observa una clara frontera entre la clase LPE y el resto de eventos. De forma general, los VTEs y los HYB comparten regiones del espacio de representación debido a la similitud de sus características. En el caso de los TRE, su región en el espacio de representación también está próxima a la de los VTEs e HYB. Este hecho puede estar motivado en la componente de bajas frecuencias que todos comparten, debido a la atenuación de las altas frecuencias y a la caída exponencial de la energía que VTEs e HYB experimentan por los efectos de propagación (sección 1.3.1).

- Al observar la Figura 8.2.1b, se puede apreciar como la delimitación de fronteras entre clases adyacentes en el espacio de características no está tan clara cuando se introducen los silencios que anteceden y siguen a cada uno del resto de eventos. Además de las conclusiones anteriormente señaladas, en esta figura se observa que los SIL se disponen en todo el espacio de representación. Este hecho podría estar motivado en la naturaleza del propio evento. Cuando éste tiene una componente de alta frecuencia es fácil asociarlo con ruido ambiental y meteorológico. Sin embargo, cuando tiene componentes en bajas frecuencias, es difícil distinguirlo de los TRE, y por tanto, de algunos LPEs, ya que su única diferencia suele ser la amplitud pico a pico entre ambas señales.

8.3. Evaluación de los sistemas

Las arquitecturas recurrentes, obedeciendo a su naturaleza, asignan una etiqueta o clase de pertenencia a cada ventana de tiempo analizada (sección 4.1). Desde la pers-

pectiva del aprendizaje automático, los resultados deberían ser analizados a partir de estas predicciones. Sin embargo, en al ámbito de nuestro problema, estas predicciones deben ser trasladadas a nivel de evento, pudiendo encontrar errores en la emisión de etiquetas, como por ejemplo eventos de un determinado tipo con una duración inadecuada o incluso eventos entremezclados. Para evitar este tipo de escenarios, sería útil aplicar algún tipo de gramática basada en el conocimiento geofísico de los eventos que se reconocen. Dado que los sistemas serán entrenados y evaluados con datos del volcán de Isla Decepción (Antártida), el conocimiento y la experiencia de personal experto, adquiridos durante decenas de campañas desde el año 1986, servirán como herramienta para mejorar los resultados de reconocimiento.

Teniendo presente la duración media de los eventos, representada en la Figura 5.2.3, una de las soluciones más viables y ampliamente usadas en este tipo de escenarios, es la incorporación de reglas que permitan verificar la coherencia de las predicciones. Para ello, definiremos una gramática que analizará a posteriori el conjunto de predicciones de reconocimiento de los frames y adecuará la etiqueta de aquellas que no tengan sentido desde el punto de vista geofísico:

- **Detección de eventos de muy corta duración entre dos eventos bien reconocidos:** la llegada o finalización de un evento es registrada por los sistemas con un cierto retraso. Durante este pequeño intervalo de tiempo (aproximadamente dos segundos), se emiten etiquetas poco fiables, ya que los sistemas detectan un cambio en el flujo de datos pero no disponen aún de información entrante relevante con la que hacer la clasificación. En este escenario, de forma general, se registran eventos de muy corta duración entre dos eventos claramente detectados. Aprovechando la probabilidad de pertenencia obtenida mediante la capa softmax (capa de salida), las etiquetas espurias son corregidas asignándoles la clase del evento de entre los dos que la delimitan que haya obtenido mayor probabilidad de pertenencia. Supongamos la siguiente emisión de etiquetas asociada a un subconjunto de ventanas entrantes, en el que las etiquetas 0 corresponden con SIL, 2 con LPE y 3 con VTE:

0	0	0	0	0	0	0	2	2	3	3	3	3	3	...

    : se puede observar que la duración del evento LPE que el sistema ha detectado no corresponde con la duración real promedio de este tipo de evento reflejada en la Figura 5.2.3. Aprovechando la probabilidad de pertenencia emitida por la capa softmax para cada una de las clases adyacentes, este error espurio queda corregido asignando dichas etiquetas a la clase de mayor probabilidad de las clases colindantes. Si suponemos que las ventanas asociadas a LPE tienen una probabilidad de pertenencia a SIL mayor que a VTE, el vector de salida final resultará en el reconocimiento de un evento tipo SIL y otro VTE :

0	0	0	0	0	0	0	0	0	3	3	3	3	3	...

- **Detecciones consecutivas de diferentes eventos de corta duración entre eventos bien reconocidos:** como ya hemos argumentado ampliamente en a lo largo del texto, una de las particularidades más importantes de las señales sismo-volcánicas es el solapamiento o superposición en el tiempo de eventos. Dicha superposición se traduce en señales con contenidos espectrales muy heterogéneos. Este hecho conlleva la emisión estocástica de etiquetas, donde durante un cierto intervalo de tiempo, cada ventana entrante es asociada a un evento diferente, no cumpliendo ninguno de ellos el requisito de duración mínima. Pese a que este comportamiento podría ser considerado erróneo o poco deseable,

desde un punto de vista geofísico, tiene una ventaja muy importante, ya que indica aquellas partes de la señal que deberían ser analizadas minuciosamente a posteriori. Ante la incompletitud de las bases de datos de las que disponemos, en las que los eventos etiquetados son mayormente aquellos bien diferenciados que garantizan un correcto entrenamiento de los modelos, y dado que los sistemas en tiempo real deben emitir forzosamente una etiqueta para cada ventana de tiempo, se ha optado por presentar un nuevo tipo de evento, el cual llamaremos evento desconocido, con el que representaremos la detección consecutiva de diferentes eventos de corta duración entre dos eventos bien reconocidos y que será considerado una inserción, con el objetivo de ser posteriormente analizado por personal experto.

- | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 2 | 2 | 4 | 4 | 4 | 1 | 1 | 1 | 1 | ... |: entre las etiquetas 0 y 1 se detectan una serie de ventanas pertenecientes a diferentes tipos de eventos, pero ninguno de ellos cumple la restricción de la duración mínima. Al tratarse de la emisión estocástica de etiquetas, se interpreta una posible superposición de señales con contenidos espectrales muy heterogéneos, por lo que la gramática propuesta asignará a este subconjunto de etiquetas el tipo evento desconocido, que posteriormente será estudiado por un experto en el observatorio:

| 0 | 0 | 0 | 0 | 0 | 0 | ED | 1 | 1 | 1 | 1 | ... |

Tras un análisis a posteriori de los resultados a partir de las reglas anteriormente expuestas, se obtendrán las etiquetas de salida del sistema. Dichas salidas serán evaluadas siguiendo los criterios descritos, para finalmente, dar como medida del rendimiento del sistema definido en el capítulo 2.

Con el objetivo de analizar la capacidad de adaptación de los modelos a las nuevas dinámicas del volcán, los mejores modelos obtenidos para cada arquitectura, que fueron entrenados con campañas de recogida de registros sísmicos en los años 1994 a 2002, serán usados para clasificar un corpus de datos perteneciente a la campaña antártica del año 2017.

El rendimiento general de los sistemas se evaluará en función del porcentaje de registros bien reconocidos. Es importante recordar que el criterio usado para medir el porcentaje de reconocimiento correcto en problemas de clasificación en continuo tiene en cuenta los errores de inserción cometidos. Estos errores deberán ser analizados detenidamente con la ayuda de personal experto con el objetivo de valorar el rendimiento real de los sistemas, ya que posiblemente, muchas de esas inserciones se puedan asociar a errores de incompletitud de la base de datos.

8.4. Estudio comparativo

Basándonos en la necesidad de la parametrización de las señales (motivada en la sección 6.4) y en el proceso de extracción de características propuesto, las arquitecturas recurrentes descritas en la sección 4.1 serán evaluadas con diferentes parametrizaciones. Debido a la gran cantidad de experimentos realizados, solo se presentarán los resultados de las mejores configuraciones.

Tres serán los análisis que se describirán:

- En primer lugar, se analizarán las capacidades de modelado temporal presentes en las arquitecturas propuestas. Para ello, además de analizar los resultados

desde el punto de vista del porcentaje de reconocimiento, se llevará a cabo un estudio detallado a partir de los mapas de activación de las unidades ocultas obtenidos directamente de los modelos en el conjunto de test.

- En segundo lugar, dado que estamos trabajando en la construcción de un sistema en tiempo real, se hace imprescindible la evaluación del mismo mediante su capacidad de generalización. Para ello, los modelos serán entrenados con un corpus de datos y testeados con otro perteneciente a una campaña sísmica más reciente. Esta situación además de darnos una idea de la capacidad de adaptación que tienen estas arquitecturas ante nuevas dinámicas eruptivas, nos dará información sobre la robustez de las características extraídas para modelar la secuencialidad de los datos.

- Finalmente, aprovechando el umbral de las probabilidades de pertenencia emitidas por la capa softmax (salida), se analizará e interpretará la confianza de las predicciones, aportando a los observatorios vulcanológicos conclusiones importantes a tener en cuenta a la hora de evaluar un sistema de clasificación.

8.4.1. Rendimiento general de los sistemas propuestos

Antes de comenzar el análisis de los resultados, describiremos brevemente el proceso de ajuste seguido para encontrar la configuración óptima de cada arquitectura.

Siguiendo un enfoque similar al descrito en [189], el número de unidades de la capa oculta cada una de las arquitecturas ha sido evaluado en el intervalo $[10, 300]$. Al igual que ocurría con las DNNs, el ajuste de los parámetros está determinado por la estimación de un gradiente, lo que hace necesario definir una tasa de aprendizaje. Durante la fase de experimentación se evaluaron varios en el rango [0.001,0.1], con un momentum fijado en 0.9. Dado que los modelos pueden verse afectados por el desbalanceo de clases y por la longitud de los eventos de algunas de ellas, es necesario introducir técnicas de regularización más allá de las implícitas en el proceso mismo de propagación del gradiente. En nuestro caso, hemos usado dos:

- Por un lado, optamos por el término de regularización L_2 obteniendo así una medida de la complejidad del modelo en función del valor de sus pesos, discriminando aquellos cuya medida sea demasiado alta y evitando en parte el desbordamiento del gradiente.

- Por otro lado, como ya citamos en la Metodología experimental (sección 8.1), se ha utilizado el criterio de *early stopping* ajustado a 10 iteraciones.

Todos los modelos han sido entrenamos bajo el mismo método de optimización, *Adam* (sección 3.3.4.6). El uso de la normalización por batch no tiene ningún efecto en nuestro problema, ya que los datos han sido normalizados en media y varianza durante la etapa de preprocesado. Es importante destacar que tampoco se han tenido en cuenta técnicas de dropout, ya que el número de parámetros con los que se han obtenido los mejores resultados no es demasiado alto. El tamaño de batch para optimizar el descenso estocástico del gradiente obtuvo su mejor resultado con 10 instancias. Tras este exhaustivo proceso de búsqueda, las mejores configuraciones de número de neuronas para la capa oculta fueron:

	RNN-Vanilla	RNN-GRU	RNN-LSTM
Raw	170	110	240
LFB+($\triangle, \triangle\triangle$)	140	300	210
LFB	60	20	130
LPC+($\triangle, \triangle\triangle$)	60	20	180
LPC	250	220	290

Tabla 8.1: Número óptimo de neuronas en la capa oculta para cada arquitectura con las diferentes parametrizaciones abordadas en el capítulo.

La Tabla 8.5 recoge los mejores resultados obtenidos por cada uno de los modelos en su conguración óptima. Cada fila de la tabla describe la parametrización usada (capítulo 6).

Como se puede observar, además de las parametrizaciones basadas en un banco de filtros, se ha introducido información relativa a la predicción lineal, pero esta vez sin usar los percentiles estadísticos, ya que a diferencia de la clasificación en aislado, dónde se usaban para dar información de la estadística de la estructura de evolución temporal del evento que la DNN no podía capturar de ningún modo, las RNNs si la capturan. Se han estudiado parametrizaciones de los frames con diferente número de coeficientes entre 3 y 15. El número de óptimo de coeficientes por frame obtenido en este corpus de datos ha sido 5. Un estudio más detallado de los resultados parametrizando cada frame con diferentes coeficientes se puede encontrar en la sección D. La Tabla 8.1 recoge las configuraciones óptimas de cada arquitectura en función de la parametrización usada.

Los tiempos de entrenamiento asociados a las configuraciones óptimas de cada arquitectura en función de la parametrización usada están resumidos en la Tabla 8.2.

Independientemente de la mejora obtenida mediante el uso de parametrizaciones basadas en LFB frente a parametrizaciones basadas en LPC, de estos resultados se pueden extraer otras conclusiones:

- En primer lugar, pese a que los resultados son prometedores, dado que la parametrización LPC ha sido aplicada a nivel de frame, la introducción de arquitecturas más complejas con las que paliar el problema del sub-ajuste no tiene apenas efecto. Los resultados obtenidos por las tres arquitecturas con esta configuración son muy similares. Desde nuestro punto de vista, esto se debe a la poca información sobre la evolución temporal pasada que puede capturarse con este tipo de parametrización. Como las características capturadas responden a información temporal (amplitud), la evolución del evento entrante no puede ser correctamente modelada, o al menos, no es correctamente modelada a muy largo plazo, ya que los mecanismos de memoria implementados detectan cambios en la información más rápidamente, actualizando por tanto su información.

- Siguiendo un razonamiento similar al argumentado en el punto anterior, se puede observar como la arquitectura clásica (Vanilla) obtiene generalmente un mayor número de inserciones que el resto de arquitecturas independientemente de la parametrización usada. Esta apreciación se debe a la incapacidad de modelado temporal de eventos muy largos, los cuales, de forma general, divide en varias partes.

- Los resultados obtenidos con las arquitecturas LSTM y GRU en cuanto a eventos correctamente clasificados, son muy similares para cualquiera de las parametrizaciones usadas. En cambio, si se aprecia una diferencia sustancial cuando se

	RNN-Vanilla	RNN-GRU	RNN-LSTM
Raw	2857.21	7640.18	40118.83
LFB+($\triangle, \triangle\triangle$)	1239.31	15107.96	16834.63
LFB	437.21	577.62	6810.25
LPC+($\triangle, \triangle\triangle$)	450.78	572.05	11795.95
LPC	2600.06	8086.50	27777.81

Tabla 8.2: Comparativa de los tiempos de entrenamiento asociados a las configuraciones óptimas de cada arquitectura, recogidas en la Tabla 8.1, en función de la parametrización usada. Los valores están expresados en segundos (s).

analiza el porcentaje de eventos insertados. Estás inserciones, como veremos más adelante, deberán ser analizadas por expertos geofísicos, ya que más que errores, podrían ser vistas como información no detectada durante el proceso de etiquetado de la base de datos. No obstante, la arquitectura LSTM requiere un mayor tiempo de entrenamiento y la sintonización de mayor número de parámetros que la arquitectura GRU.

- Finalmente, se puede comprobar como la incorporación de información dinámica al vector de características (derivadas temporales de primer y segundo orden), mejora la tasa de reconocimiento. En este sentido, los mecanismos de memoria pueden ser actualizados de una forma más acertada, incrementando así el rendimiento de los sistemas.

8.4.1.1. Estudio detallado de los resultados en función del desvanecimiento y desborde de los gradientes

Como hemos argumentado en las secciones 4.1 y 4.3, uno de los principales inconvenientes de las redes neuronales recurrentes cuando se abordan datos de larga duración, es la correcta propagación de los gradientes durante la etapa de entrenamiento. A menudo, el desbordamiento/desvanecimiento de gradientes afecta negativamente en el ajuste de los parámetros, impidiendo la captura de dependencias temporales a largo plazo disminuyendo la capacidad de generalización o de adaptación dinámica de los sistemas.

Comprobar si los modelos, y por tanto, los resultados, están siendo influenciados por este tipo de problemas es una tarea compleja y en muchas ocasiones inexacta.

En este trabajo, siguiendo la metodología descrita en la sección 4.3 se han realizado dos tipos diferentes de experimentos para detectar si las configuraciones óptimas (Tabla 8.1) están teniendo o no desbordamiento o desvanecimiento del gradiente:

1. Estudiando la norma de los gradientes durante la etapa de entrenamiento. La Tabla 8.3 resume los radios espectrales de las matrices de pesos recurrentes W_{rec} de cada modelo en su configuración óptima y en función de la parametrización utilizada durante la etapa de entrenamiento. Como se puede apreciar, ningún valor está enormemente lejos de 1 ni estrechamente cercano a 0, a excepción de valores asociados a los arquitectura más simple (Vanilla), que aunque sufren una desviación superior a 1, esta no llegar a ser tan grande como para producir gradientes de una magnitud intratable.

	RNN-Vanilla	RNN-GRU	RNN-LSTM
Raw	1.2027	0.2397	0.3056
LFB+($\triangle, \triangle\triangle$)	1.73	0.3163	0.2895
LFB	1.6924	0.1790	0.2544
LPC+($\triangle, \triangle\triangle$)	1.6134	0.1611	0.2806
LPC	3.48	0.2865	0.3158

Tabla 8.3: Comparativa de los valores del radio espectral (λ_1) asociado a la matriz de pesos recurrentes de cada arquitectura en su configuración óptima y en función de cada parametrización.

2. Comprobando implícitamente el valor de los gradientes propagados. Para ello, se ha escogido el registro sismo-volcánico de mayor duración, y se han analizado los valores de los gradientes propagados durante la etapa de entrenamiento. Por un lado, se considera que un modelo está siendo afectado por situaciones de desbordamiento cuando el gradiente propagado supera un determinado umbral o su valor corresponde con NaN (Not a Number). Por otro lado, se considera que un modelo está siendo afectado por situaciones de desvanecimiento de gradiente, cuando el 25 % de los gradientes propagados son próximos a $0 + \varepsilon$, siendo $\varepsilon = 10^{-8}$. La Tabla 8.4 recoge el valor del percentil 25 % de los valores de los gradientes propagados en cada una de las matrices de parámetros (pesos) para la arquitectura LSTM. Cada columna corresponde a cada una de las parametrizaciones usadas y en sus configuraciones óptimas. *Input, forget y output* corresponden a las matrices asociadas a los mecanismos de activación y desactivación de las puertas de entrada, de reseteo y de salida. *Cell* corresponde a las matrices asociadas a la celda de memoria. *h_y* corresponde a la matriz de parámetros que describen las conexiones entre la capa oculta y la capa de salida. El estudio asociado al resto de arquitecturas se ha ubicado en el Anexo E para facilitar la lectura. Como se puede apreciar, ninguna matriz presenta porcentajes de gradientes propagados similares a cero por encima del 25 %, por lo que unido al estudio anteriormente expuesto, concluimos que el desvanecimiento de gradiente no está influyendo en los porcentajes de clasificación obtenidos.

8.4.1.2. Análisis detallado de los resultados

Una cuestión importante que se desprende de los resultados obtenidos (Tabla 8.5), es la diferencia entre elementos borrados y elementos insertados. Mientras que el número de elementos borrados es relativamente pequeño, el número de elementos insertados es demasiado grande, llegando a decrementar el rendimiento de los sistemas hasta en un 15 %. Esta característica puede significar dos cosas:

1. El etiquetado de los datos sea incompleto, y algunas de estas inserciones sean producto de la capacidad de los modelos para detectar información que el personal experto no detectó. Además, la detección de eventos superpuestos aumentará la emisión de etiquetas de eventos desconocidos, y por tanto, el número de inserciones.

2. Las inserciones correspondan con errores de etiquetado cometidos por los modelos y por lo tanto, los resultados correspondan con el rendimiento real de los

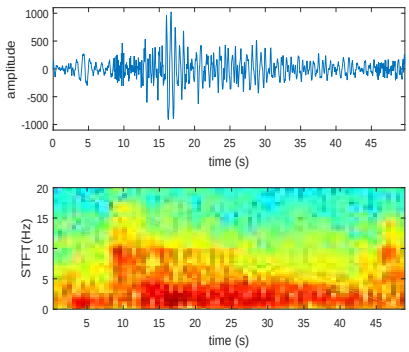
		LPC	LPC+($\triangle, \triangle\triangle$)	LBF	LBF+($\triangle, \triangle\triangle$)	Raw
W_xi	input	0.0289	0.0203	0.0171	0.0054	0.0085
W_hi		0.0088	0.0049	0.0121	0.0009	0.0038
W_ci		0.0448	0.0419	0.0932	0.0031	0.0109
W_xf	forget	0.0169	0.0192	0.0101	0.0030	0.0037
W_hf		0.0068	0.0045	0.0114	0.0008	0.0041
W_cf		0.0471	0.0155	0.1449	0.0024	0.0123
W_xo	output	0.0112	0.0173	0.0867	0.0048	0.0023
W_ho		0.0048	0.0043	0.0964	0.0012	0.0052
W_co		0.0438	0.0141	0.0867	0.0027	0.0144
W_xc	cell	0.1818	0.0988	0.0102	0.0150	0.0087
W_hc		0.0996	0.0571	0.0058	0.0034	0.0180
W_hy	h_y	0.3560	0.2168	0.1586	0.0147	0.0955

Tabla 8.4: Comparativa del percentil 25 % de los valores de los gradientes propagados a través de las arquitecturas LSTM. Cada columna corresponda con la configuración óptima de cada parametrización. Cada fila corresponde con una de las matrices de parámetros asociadas a la arquitectura.

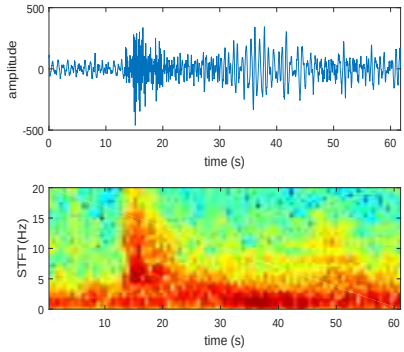
	LFB+($\triangle, \triangle\triangle$)		LFB	
	Acc	Cor	Acc	Cor
RNN-Vanilla	76.49±6.38	92.03±2.39	74.15±5.97	89.54±4.02
RNN-GRU	81.62±2.43	93.02±2.36	81.76±3.34	91.60±3
RNN-LSTM	82.57±1.78	94.00±2.45	79.28±4.41	93.81±3.03

	LPC+($\triangle, \triangle\triangle$)		LPC	
	Acc	Cor	Acc	Cor
RNN-Vanilla	75.87±6.02	87.45±3.02	72.39±6.1	86.90±2.6
RNN-GRU	74.68±3.71	89.37±3.35	75.38±2.14	89.29±4.29
RNN-LSTM	76.66±6.34	88.14±2.98	78.53±3.64	90.08±3.44

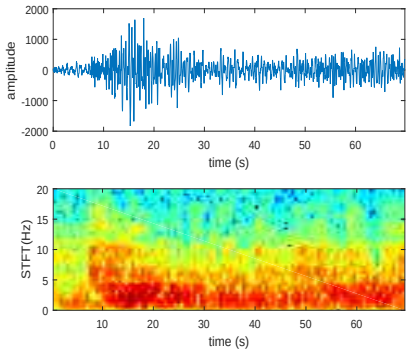
Tabla 8.5: Rendimientos obtenidos por las arquitecturas recurrentes (haciendo uso de la gramática descrita en la sección 8.3) en función de la parametrización usada. Los resultados están expresados en % y corresponde con el promedio obtenido en los cuatro conjuntos de tests usados en la validación cruzada.



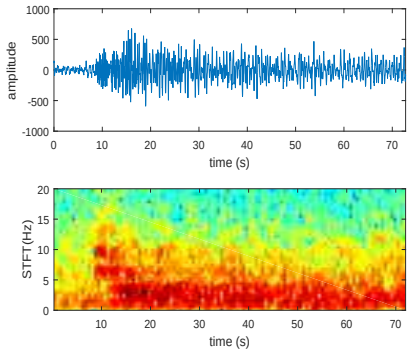
(a) Etiquetas: SIL-TRE



(b) Etiquetas: SIL-VTE-SIL-LPE



(c) Etiquetas: SIL-TRE



(d) Etiquetas: SIL-HYB-TRE

Figura 8.4.1: Estudio pormenorizado de los casos donde los modelos han obtenido un elevado número de inserciones. Los espectrogramas han sido seleccionados del conjunto de test.

mismos.

Teniendo presente ambas hipótesis, es necesario realizar un estudio pormenorizado de los casos donde los modelos han obtenido un elevado número de inserciones o borrados con el objetivo de valorar fehacientemente los resultados obtenidos. Para explicar el procesamiento que hemos hecho, nos apoyaremos en la Figura 8.4.1:

- Figura 8.4.1a: las etiquetas asociadas por los expertos vulcanólogos a este segmento de señal fueron SIL-TRE. Sin embargo, todas las arquitecturas introdujeron un pequeño VTE previo a la ocurrencia del TRE. Tras una supervisión a posteriori por algunos de los expertos que etiquetaron el conjunto de datos, podemos considerar como correcta la detección obtenida por los sistemas. La fuente de este tipo de TRE, a menudo, está precedida de un pequeño VTE, pero dada su pequeña magnitud y su corta duración, fue omitido de la etiqueta original. Por lo tanto, el error de inserción en este caso es un error de incompletitud, lo que resulta en una mejora del rendimiento de los sistemas.
- Figura 8.4.1b: las etiquetas asociadas a este segmento son SIL-VTE-SIL-LPE.

En este caso, el segundo evento detectado como SIL tiene una duración menor que la considerada para este tipo de eventos en las reglas incorporadas para verificar la coherencia de las predicciones (Figura 5.2.3e), por lo que los sistemas lo ignoran y lo tratan como parte de la coda del VTE. La salida final consta de tres eventos, SIL-VTE-LPE. Dado que un evento híbrido puede ser considerado como la concatenación de un VTE y un LPE, tras aplicar la gramática, la salida final se resume en SIL-HYB, introduciendo un borrado si la comparamos con la etiqueta original. Tras un estudio a posterori de este segmento, nuevamente, los expertos geofísicos consideran correcta la salida.

- Figuras 8.4.1c y 8.4.1d: estos espectrogramas corresponden a dos señales diferentes, SIL-TRE y SIL-HYB-TRE que todas las arquitecturas han reconocido como SIL-TRE, introduciendo un borrado y reduciendo por tanto el porcentaje de eventos bien reconocidos. Al observar ambos espectrogramas, podemos concluir que la presencia del evento HYB no es fácilmente distinguible, siendo necesaria la forma de onda para llegar a detectarlo. En este caso, el factor humano tiene un rol esencial, ya que la inclusión o no del evento HYB en el proceso de etiquetado depende de la subjetividad geofísica del operador humano.

Finalmente, en concordancia con el capítulo 7, abordamos un estudio de los resultados de clasificación apoyándonos tanto en las matrices de confusión de la Figura 8.4.2, como en el análisis t-SNE de la Figura 8.2.1 anteriormente expuesto:

- Ruido/Silencio: de forma general el ruido es reconocido con una alta precisión y una alta sensibilidad por todos los sistemas. De las matrices de confusión se extrae que el ruido es principalmente confundido con tremors (TRE) o no reconocido, conclusión que además se puede corroborar observando el análisis t-SNE. Una explicación a esta situación podría ser la degradación de la amplitud pico a pico de las señales relacionada con los efectos de atenuación descritos en la sección 1.3. Cuando el Ruido tiene una componente de alta frecuencia es fácil asociarlo con ruido cultural (actividades humanas) o incluso ambiental (viento). Sin embargo, a bajas frecuencias, es difícil distinguir directamente entre Ruido y TRE, excepto cuando se aplican técnicas de avanzadas como arrays sísmicos o análisis de polarización.

- HYB: Su característico comienzo en altas frecuencias hace que los modelos generalmente confundan este tipo de eventos con VTEs. Aunque después de la primera llegada se registra una segunda señal de características similares al evento de LPE, desde el punto de vistas del aprendizaje máquina, si entre estas dos señales no existe un pequeño intervalo de tiempo, generalmente GRU y Vanilla asociarán la llegada de la segunda señal con la coda de la primera.

- VTE-LPE: Observando la Figura 8.2.1a, se puede apreciar como algunos de los VTEs están dispuestos en una zona del espacio de representación de los LPE. Esta característica coincide con los porcentajes de confusión presentes en las matrices de confusión. Una posible explicación a este comportamiento es la distancia hipocentral a la que se encuentran las fuentes que los generan, del sensor que los registra. A medida que la diferencia de llegada de las ondas P y S se incrementa (mayor distancia hipocentral), disminuye el contenido espectral en altas frecuencias de la señal registrada. A partir de distancias cercanas a los 12 km (dependiendo del modelo de velocidad) las diferencias espectrales entre VTE y LPE son mínimas, por lo que ambos eventos son fácilmente confundidos.

Tras la llegada de los paquetes de ondas P y S, los terremotos experimentan una caída exponencial de energía. Dicho decaimiento adquiere un contenido espectral muy similar al del TRE. Observando el análisis t-SNE, se puede apreciar como algunos de los VTEs se solapan con los TRE en varias zonas del espacio de representación, lo que nuevamente queda reflejado en las matrices de confusión.

- TRE: Aunque como hemos citado anteriormente, son muchas las confusiones con VTE debido a los efectos de atenuación, el mayor porcentaje de TRE mal clasificados se corresponden con LPE. De forma general, los contenidos espectrales de ambos eventos son prácticamente idénticos, habiéndose llegado a describir un tipo específico de TRE como un conjunto encadenado de LPE, siendo la diferencia más característica entre ambos, la duración. Mientras un LPE dura apenas unos minutos, un TRE puede durar incluso días. En este sentido, el factor humano asociado al proceso de etiquetado de los datos juega un papel de vital importancia, ya que muchas veces, durante este proceso, cuando los TRE son ininterrumpidos y demasiado largos, no son aislados, sino que se etiquetan pequeños segmentos representativos de los mismos. Estos segmentos, como hemos citado anteriormente, tienen prácticamente la misma duración y el mismo contenido espectral que un LPE, por lo que los modelos tienden a confundirlos. Es importante destacar, que desde un punto de vista geofísico, este error en la clasificación suele ser despreciado.

8.4.2. Análisis de los resultados mediante mapas de activación

Obtener información acerca de lo que los modelos están detectando e intentar dar explicación a cómo desarrollan su conocimiento es todavía hoy una línea prematura de investigación. Además, dependiendo del área en el que se esté trabajando, esta información podrá ser representada en mayor o menor medida en una forma inteligible para el humano.

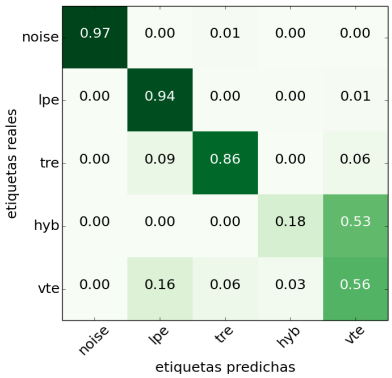
Un ejemplo de ello es la clasificación de imágenes, donde uno de los métodos más extendidos para analizar qué tipo de características se están aprendiendo es la visualización e interpretación de los mapas de activación de las neuronas de la capa oculta [154, 212, 35].

Los trabajos hasta ahora presentados, se basan en el análisis de mapas de prominencias o correlaciones abstractas de neuronas, en una forma similar a la usada en la segmentación de imágenes, en la que el objetivo es simplificar y/o cambiar la representación de una imagen en algo que sea más significativo y fácil de analizar.

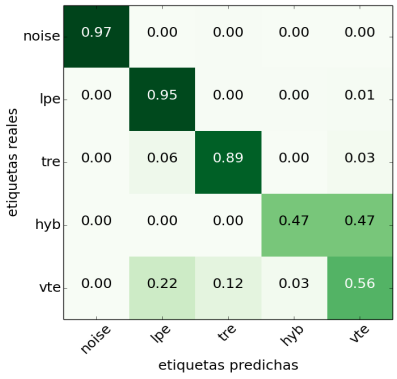
Sin embargo, la interpretabilidad de las redes neuronales, cuya eficacia reside en la captura de características a través de sus capas ocultas, se puede enfocar mediante un diccionario semántico [155]. La activación de cada neurona es emparejada con una visualización de la misma, cambiando la relación con el objeto (matemático) adyacente detectado.

En este escenario, las activaciones no son simples índices o valores numéricos (abstractos), sino que las activaciones se asocian a representaciones icónicas, como por ejemplo partes de un objeto dentro de un determinado escenario.

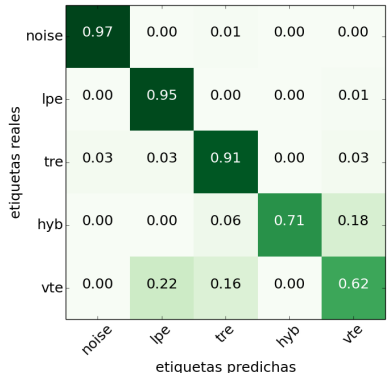
Como hemos citado anteriormente, dado el carácter simbólico de las imágenes, la representación de características se presenta de forma natural como un conjunto de abstracciones visuales. En cambio, construir representaciones a partir de abstracciones o combinaciones de neuronas que se disparan en una determinada ubicación espacial en problemas como el procesamiento automático de la voz o el reconocimiento automático



(a) Vanilla



(b) GRU



(c) LSTM

Figura 8.4.2: Matrices de confusión normalizadas asociadas a las arquitecturas implementadas haciendo uso de la parametrización LBF+Δ + $\Delta\Delta$. Los resultados corresponden con la precisión (ecuación 2.2.14) promedia obtenida en los cuatro conjuntos de tests usados en la validación cruzada.

de señales sismo-volcánicas, no es una tarea sencilla, máxime si la información a partir de la cual se construyen está de algún modo parametrizada.

8.4.2.1. Análisis por eventos

En esta sección se abordará el estudio de los mapas de activación de las unidades de la capa oculta de la arquitectura que mejor resultados ha ofrecido (LSTM), usando como parametrización el LBF+△+△△.

Para ello, se han seleccionado algunos de los ejemplos más representativos (LPE, VTE, TRE) (Figura 8.4.3). Cada uno de estos ejemplos está representando, por un lado, con las características espectrales del registro sísmico en el tiempo (espectrograma) y por el otro, por la activación de la unidades de la capa oculta para cada una de las ventanas en las que se ha dividido la información de entrada. Es importante destacar, que aunque la arquitectura LSTM implementa un mecanismo de puertas y células de memoria con diferentes funciones no lineales, el valor de las unidades ocultas se obtiene a partir de la aplicación de la función *tanh* (sección 4.1), por lo que el rango numérico en la activación de las neuronas estará entre $[-1, 1]$.

Varias han sido las conclusiones extraídas de este análisis:

1. Como los eventos están representados por características espectrales, las unidades ocultas se activan o desactivan dependiendo del contenido espectral. Cuando el contenido espectral viene dado por bajas frecuencias, solo se activan unas pocas unidades (Figura 8.4.3a). Es importante observar cómo, cuando el contenido espectral cambia, la unidad activa se desactiva, y algunas de las unidades que estaban desactivadas se activan, lo que indica la llegada de un evento emergente. Esta propiedad puede ser utilizada como una herramienta de revisión del proceso de etiquetado de la base de datos.

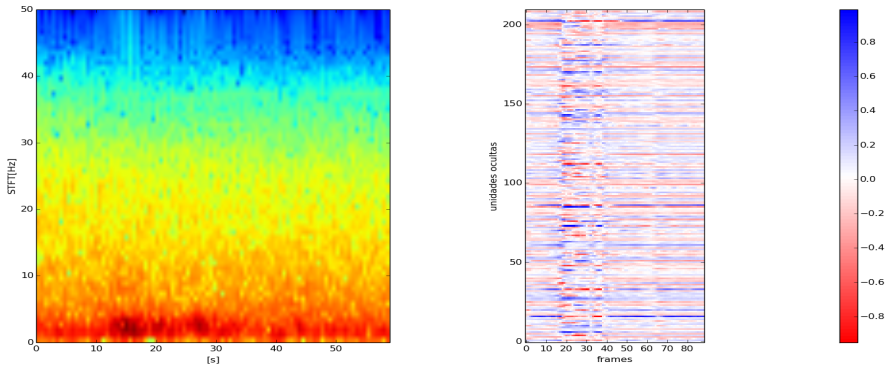
2. El tiempo durante el que las unidades permanecen activas dependerá del contenido espectral del evento (Figuras 8.4.3b y 8.4.3c). Se puede apreciar fácilmente como a medida que los eventos tienen un contenido espectral más amplio, el número de neuronas que se excitan aumenta. Además, el nivel de excitación de las unidades también depende del contenido espectral del segmento entrante. Para frecuencias bajas, la excitación es menor que para frecuencias altas, donde se obtienen valores muy positivos y muy negativos (teniendo en cuenta que estamos usando la función tanh como función de activación).

3. Este comportamiento nos sugiere que (como ya hemos citado anteriormente) algunas de las etiquetas obtenidas a partir de los modelos recurrentes pueden mejorar a las asignadas por los operadores humanos. En este sentido, encontramos viable un trabajo futuro en el que aplicar las redes neuronales recurrentes como algoritmos de picking automático.

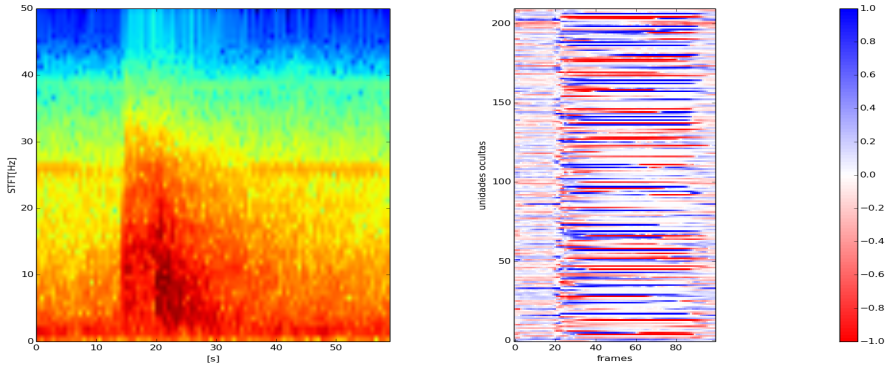
8.4.2.2. Análisis por arquitecturas

Como estudio complementario, se ha abordado una comparativa de los resultados haciendo uso del resto de arquitecturas implementadas en este trabajo. Para ello, se han comparado los mapas de activación de las 3 arquitecturas. Como ejemplos representativos se han seleccionado un VTE y un LPE, puesto que ambos van acompañados de la clase SIL al comienzo y al final del registro (Figura 8.4.4).

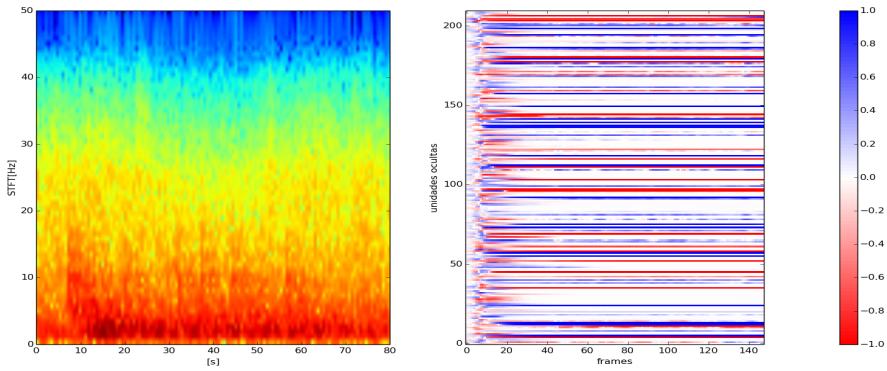
De los resultados se pueden extraer varias conclusiones:



(a) SIL-LPE-SIL-LPE



(b) VTE



(c) HYB-TRE

Figura 8.4.3: Mapas de activación pertenecientes a algunos de los registros sísmicos recogidos en la Isla Decepción durante las campañas de 1994–1995, 1995–1996 y 2001–2002.

1. Los mapas de activación de las arquitecturas LSTM y GRU tienen cierta similitud. Esto se debe al uso de celdas de memoria. Fruto de ello es la más clara detección de la llegada de eventos emergentes. El cambio en el contenido espectral es detectado de forma más temprana por las arquitecturas más complejas, ya que estas vienen "arrastrando" un estado de poca actividad. Al detectar un cambio significativo en la información entrante, tanto GRU como LSTM actualizan el estado de su celda de memoria y por tanto el estado de sus puertas, cambiando por completo la dinámica del sistema, que a partir de este momento intentará inferir qué tipo de evento se está registrando.

2. Aunque del mapa de activación de la arquitectura Vanilla también se puede extraer, de una forma menos clara, la llegada y el final de cada evento, es importante destacar que la ausencia de células de memoria y el uso de la información a muy corto plazo, deriva en una alta actividad neuronal. En cada instante de tiempo, el modelo intenta inferir el tipo de evento entrante, sin tener en cuenta ninguna información pasada a excepción de la inmediatamente anterior, por lo que generalmente, la mayor parte de las neuronas están activas. De esta observación también se desprende que, a medida que los mecanismos que implementan las células de memoria son más complejos, se disminuye la actividad neuronal. Si comparamos los mapas de activación de las arquitecturas LSTM y GRU, se puede ver como para LSTM un mayor número de neuronas permanecen inactivas o poco activas después de detectar la llegada de un nuevo evento. El uso de un modelo más complejo deriva en una mayor especialización de las neuronas, siendo necesario por tanto una menor actividad neuronal.

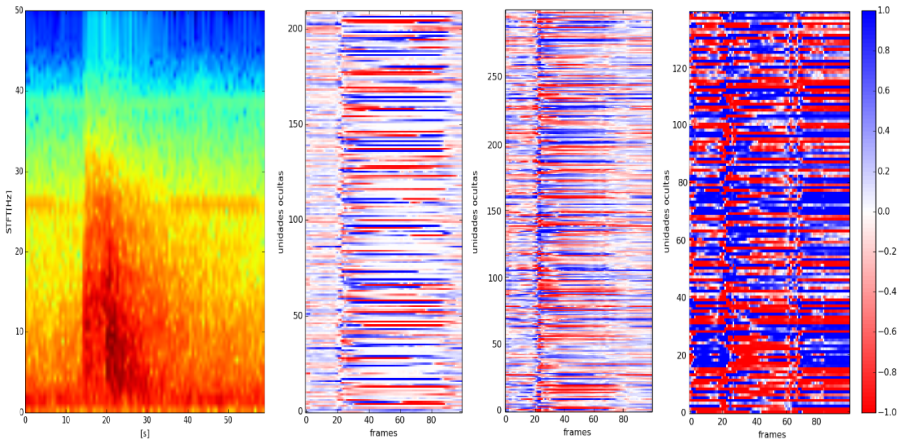
3. La delimitación de los eventos una vez detectados está también fuertemente influenciada por el tipo de modelo implementado. La capacidad de modelar dependencias temporales a muy largo plazo necesita cambios en la información entrante prolongados para evitar así la constante actualización del estado de la célula de memoria. En este sentido, las arquitecturas GRU y LSTM situarán el final de los eventos generalmente, algunas ventanas de tiempo después (retardo) que la arquitectura Vanilla. Este retardo, a su vez, dependerá de la información entrante, siendo más acusado en eventos cuyo contenido espectral cambia más suavemente.

8.4.2.3. Análisis por parametrización

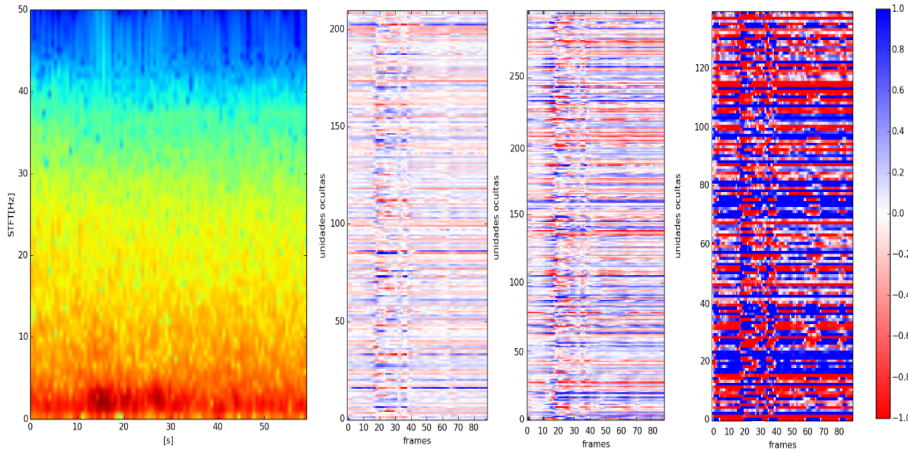
Finalmente, con el objetivo de extraer información acerca de cómo afecta el uso de diferentes parametrizaciones en la actividad neuronal, se han obtenido los mapas de activación de los eventos hasta ahora estudiados mediante LBF+Δ+$\Delta\Delta$ haciendo uso de LPC+Δ + $\Delta\Delta$como parametrización. Para facilitar el análisis de los resultados, se han considerado solamente los resultados asociados a la arquitectura LSTM (Figura 8.4.5).

De los resultados se han extraído las siguientes conclusiones:

1. La llegada de las ondas sísmicas de más contenido espectral es capturada eficientemente por ambas parametrizaciones. Sin embargo, usando la parametrización basada en LPC, los sistemas presentan dificultades a la hora de delimitar el final o la caída energética de los eventos. Cuando tras la llegada de un evento con contenido espectral amplio (VTE), los valores enérgicos no se reestablecen

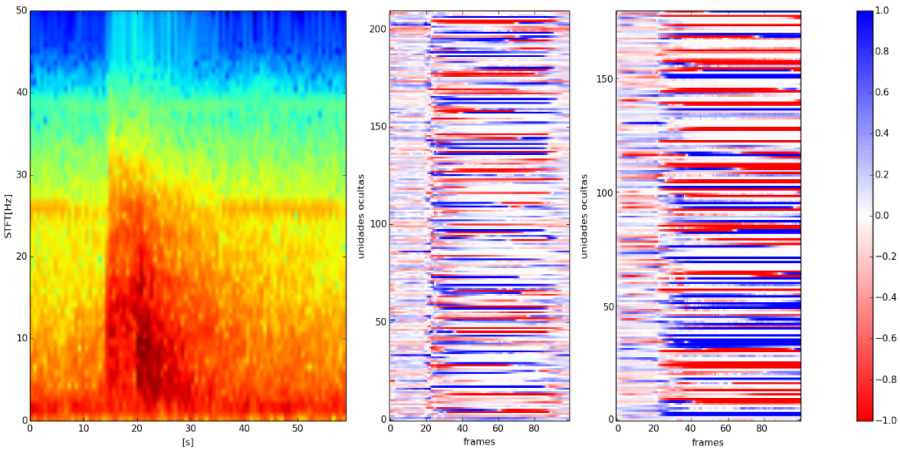


(a) VTE

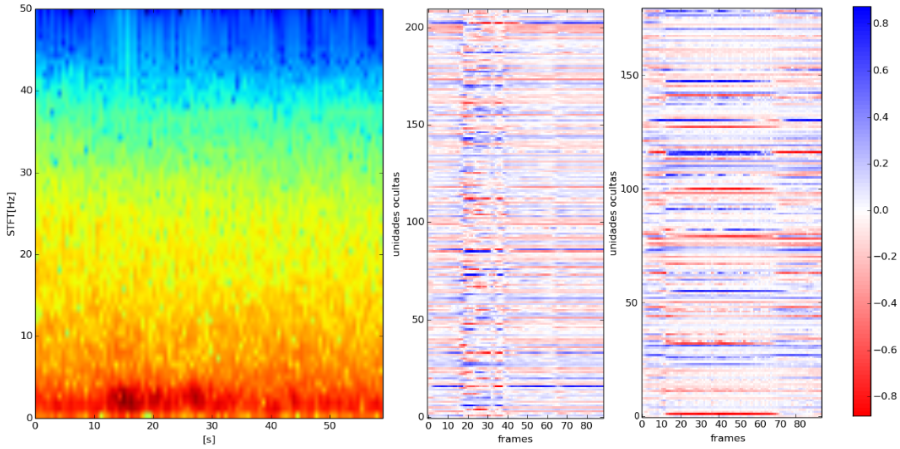


(b) SIL-LPE-SIL-LPE

Figura 8.4.4: Comparativa de los mapas de activación pertenecientes a las diferentes arquitecturas empleadas en este trabajo y en la base de datos de la Isla Decepción. Cada imagen representa el espectrograma asociado al registro sísmico junto a los mapas de activación asociados a las arquitecturas LSTM (izquierda), GRU (centro) y Vanilla (derecha).



(a) VTE



(b) SIL-LPE-SIL-LPE

Figura 8.4.5: Comparativa de los mapas de activación asociados a la arquitectura LSTM y pertenecientes a las dos mejores parametrizaciones ((LBF+Δ + $\Delta\Delta$) y (LPC+Δ+$\Delta\Delta$). Cada imagen representa el espectrograma asociado al registro sísmico junto a los mapas de activación asociados a ambas parametrizaciones: LBF+Δ+$\Delta\Delta$ (izquierda), y LPC+Δ + $\Delta\Delta$ (derecha).

por completo, LPC, basada en los valores de la señal en el dominio del tiempo (amplitud), no permite una correcta detección del cambio en las propiedades de la señal y por tanto, no actualiza adecuadamente la célula de memoria (Figura 8.4.5a). Si por el contrario, tras la llegada de un evento con contenido espectral relativamente amplio (LPE), se reestablecen los valores enérgicos de la señal, los sistemas detectarán el cambio con un retardo amplio. Esta característica tiene implicaciones muy importantes, ya que ante la llegada de un enjambre o tren de eventos, el sistema los detectará como uno solo de gran duración, incurriendo en un elevado número de elementos borrados, y por tanto, decrementando su rendimiento. Este comportamiento se puede apreciar en la Figura 8.4.5b, en la que el sistema detecta la llegada de un solo evento LPE, cuando la realidad corresponde a dos de ellos muy seguidos en el tiempo.

2. La actividad neuronal en ambos casos es muy similar, lo que nos lleva a pensar que la activación de las neuronas en el caso de la parametrización (LPC+Δ+$\Delta\Delta$) está relacionada con la impulsividad, teniendo en cuenta que dicha parametrización no tiene información directa de la energía en las diferentes bandas de frecuencia. En este sentido, ante la llegada de eventos poco impulsivos como pueden ser los LPE, la actividad neuronal será menor, y por tanto, la gran parte de sus neuronas estarán en un estado de poca actividad.

8.5. Análisis de la capacidad de generalización de los sistemas propuestos

Uno de los mayores retos en el reconocimiento automático de señales sismo-volcánicas (VSR) es la construcción o desarrollo de modelos robustos capaces de readaptarse fácilmente a las dinámicas altamente cambiantes de las fuentes sísmicas, de manera que si las dinámicas internas del volcán cambian con el tiempo, el sistema debería ser capaz de proporcionar resultados de supervisión eficientes, evitando que su rendimiento decaiga bruscamente y la supervisión automática sea ineficaz.

Basándonos en los excelentes resultados obtenidos usando datos de campañas sísmicas relativamente antiguas y basándonos también en el eficaz reetiquetado que la propia naturaleza de estos modelos ofrecen durante su testeo, analizamos la capacidad de adaptación de las configuraciones anteriormente estudiadas con un corpus de datos perteneciente al mismo volcán pero esta vez a distinta campaña, más concretamente a la campaña sísmica del año 2017 (XXX Campaña Antártica Española).

El objetivo de este experimento fue comprobar si los modelos hasta ahora ajustados podrían tener alguna aplicabilidad como herramienta de monitoreo en futuras campañas antárticas, ayudando por tanto a los expertos hasta allí desplazados en su labor de vigilancia volcánica. Para ello, se dispuso de un registro sísmico de 3.5 horas perteneciente al día 2/1/2017. La elección de este día vino motivada por su elevada sismicidad, descrita en los cuadernos de bitácora asociados a la campaña. Dado que los modelos fueron entrenados con datos pre-procesados y parametrizados, para poder obtener un estudio equitativo del rendimiento en este nuevo escenario, fue necesario filtrar las señales en su mismo rango de frecuencias $[1, 50]$ Hz. Este corpus de datos está compuesto por registros sísmicos directamente obtenidos de los sensores allí desplegados, sin ninguna supervisión humana, es decir, se desconoce a priori los eventos que componen el conjunto.

	LFB + (Δ,ΔΔ)		LFB	
	Acc (%)	Cor (%)	Acc (%)	Cor (%)
RNN-Vanilla	68.82	81.48	54.12	77.37
GRU-RNN	59.27	75.68	54.12	74.31
LSTM-RNN	65.33	79.75	75.22	80.42
HHM	42.55	60.79	-	-

Tabla 8.6: Rendimiento obtenido por los modelos recurrentes habiendo sido entrenados con datos de las campañas sísmicas 1994-1995, 1995-1996, 2001-2002 y testeados con datos de la campaña 2016-2017. En el caso de la arquitectura HMM y teniendo presenta su carácter comparativo, solo se han llevado a cabo los experimentos en la mejor parametrización.

Siguiendo el esquema de trabajo de secciones anteriores, y con el objetivo de comparar la capacidad de generalización de los modelos propuestos, se ha introducido el porcentaje de clasificación del mejor sistema de clasificación basado en HMM, entrenado con datos pertenecientes a campañas anteriormente estudiadas y testeado en este nuevo conjunto de datos. Basándonos en los resultados de la Tabla 8.5, y más concretamente en la parametrización que mejores resultados ofrece (LBF+$\Delta + \Delta\Delta$), se han propuesto los siguientes experimentos:

- Por un lado, para encontrar una configuración de propósito general, se han evaluado modelos con 5, 9 y 11 estados ocultos. Las probabilidades de emisión de un vector de características en cualquier estado, se modelaron evaluando hasta con 16 gaussianas multivariantes con matrices de covarianza diagonales.

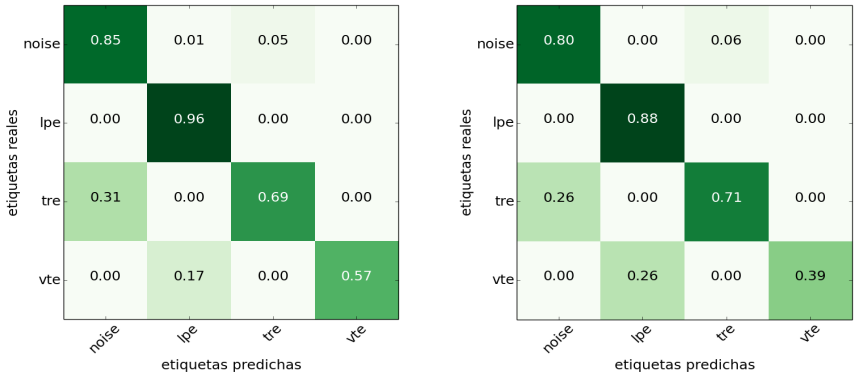
- Por otro lado, teniendo en cuenta la gran variabilidad de las señales, también consideramos la implementación de un HMM para cada tipo de evento, es decir, sabiendo que cada evento tiene características temporales diferentes, propusimos utilizar configuraciones de modelos diferentes basadas en la duración promedio de los eventos. Para eventos relativamente cortos como los LPE y VTE (menos de tres minutos), propusimos modelos con 5 estados ocultos. Para eventos intermedios con duraciones entre 3 y 7 minutos (SIL e HYB), usamos 10 estados, y finalmente, para eventos grandes como TRE, elegimos 15 estados. Como hemos descrito anteriormente, para modelar las probabilidades de emisión en cualquier estado, se evaluaron hasta 16 gaussianas multivariantes con matrices de covarianza diagonales.

Los mejores resultados se obtuvieron haciendo uso del segundo enfoque, estando cada estado modelado por 12 gaussianas multivariantes.

Desde el punto de vista geofísico, este experimento tiene un atractivo especial, ya que los nuevos registros podrían contener eventos que pueden diferir de los eventos prototipo cuidadosamente seleccionados para entrenar los sistemas.

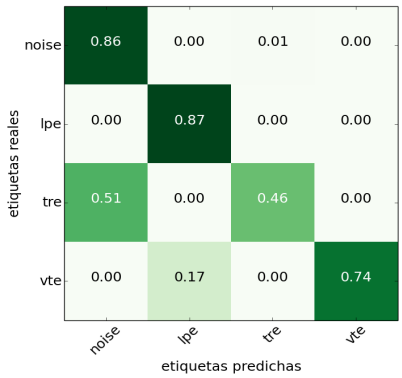
La Tabla 8.6 y la Figura 8.5.1 recogen los resultados de la prueba. Aunque los resultados son esperanzadores, se puede apreciar un notable descenso en el rendimiento de las arquitecturas GRU y Vanilla-RNN. Tras un exhaustivo estudio a posteriori de los resultados, esta disminución de la eficacia podría explicarse desde el punto de vista geofísico.

Como ya mencionamos en la sección 1.3, la atenuación y los efectos fuente condicionan enormemente el rendimiento de todos los sistemas. Un ejemplo de ello se puede

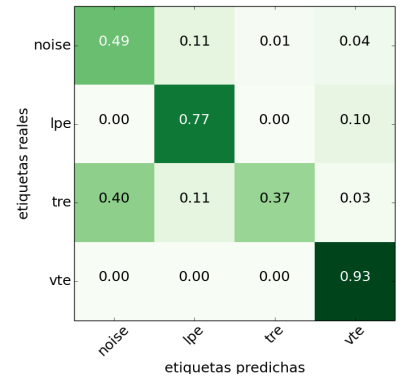


(a) Vanilla

(b) GRU



(c) LSTM



(d) HMM

Figura 8.5.1: Matrices de confusión normalizadas asociadas a las arquitecturas implementadas. Los resultados corresponden con la precisión (ecuación 2.2.14) obtenida en un conjunto de datos pertenecientes a la campaña sísmica del año 2017.

apreciar en la Figura 8.5.2 , que representa tres VTEs registrados en la misma estación sísmica, pero con diferente distancia hipocentral.

A medida que la diferencia en el tiempo de llegada de las ondas P y S se va incrementando, es decir, la distancia entre la fuente y la estación que registra la señal es mayor, la forma de onda y las características espectrales que describen los eventos se modifican, llegando a cambiar tanto, que pueden incluso ser confundidas con la forma de onda y características espectrales de otros eventos registrados en el entorno volcánico. Esta degradación de las características propias de la señal se ve claramente reflejada en las probabilidades de pertenencia emitidas por los sistemas en su capa de salida. Para la Figura 8.5.2a, cuya distancia hipocentral está en torno a los 3-4 km (ya que la diferencia P-S es de aproximadamente 1 segundo), los sistemas emiten probabilidades de pertenencia muy heterogéneas. Dado que el contenido espectral de la señal está por encima de los 20 Hz y la forma de onda apenas está deformada, la probabilidad de pertenencia asignada a la clase VTE se acerca al 70 %.

En el caso de la Figura 8.5.3b, la distancia hipocentral es de aproximadamente 8 Km (una diferencia P-S de 2 segundos). Aquí ya la forma espectral comienza a cambiar y las altas frecuencias no superan los 15 Hz. En consecuencia, la probabilidad de pertenencia asignada al evento VTE disminuye casi un 20 %, acercándose al 52 %.

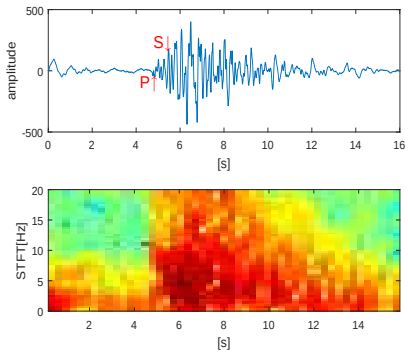
Finalmente, en la Figura 8.5.2c, la distancia hipocentral se aproxima a los 12 km (diferencia S-P de 3 segundos). Aunque no es una distancia demasiado grande dentro del entorno volcánico de Isla Decepción, y se pueden llegar a registrar VTEs a distancias mayores, la atenuación de los contenidos de alta frecuencia es ya muy pronunciada. La prácticamente única presencia de frecuencias inferiores a 6 Hz lleva a los sistemas a emitir probabilidades de pertenencia muy homogéneas. Dadas las similitudes tanto en la forma de onda como en el contenido espectral con algunos de los eventos LPE usados durante la etapa de entrenamiento, las probabilidades de pertenencia asignadas para ambas clases no superan el 40 %, generando una enorme confusión en el reconocimiento.

Otro aspecto importante a tener en cuenta de los efectos de atenuación es la degradación de la amplitud pico a pico de la señales. Dos eventos generados por un mismo mecanismo fuente pero a diferente distancia de la estación que los registra, pueden ser claramente confundidos por los sistemas de reconocimiento aunque su contenido espectral sea idéntico si los niveles de energía de su forma de onda (amplitud) presentan diferencias notables.

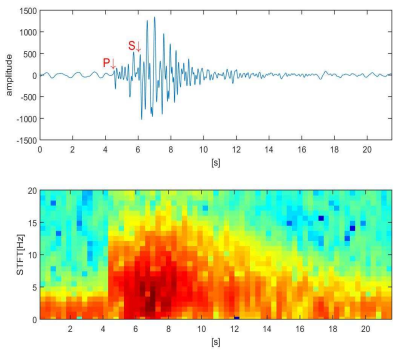
Un ejemplo de ello se puede apreciar en la Figura 8.5.3. Mientras que los patrones espectrales de ambos espectrogramas sugieren que ambas señales han sido generadas por el mismo mecanismo fuente, las gran diferencia en su nivel de energía indica fuertes efectos de atenuación. En este sentido, la Figura 8.5.3a, dada su amplitud, fue etiquetada como TRE, mientras que la Figura 8.5.3b, con un nivel de energía casi 10 veces menor que la primera, fue asociada con un SIL.

Desde un punto de vista geofísico, este tipo de inconsistencias no tienen ninguna relevancia, pues un TRE tan poco energético puede ser considerado como SIL. No obstante, este tipo de contradicciones en el etiquetado automático suponen el 5 % del error total obtenido. Muchos de los TRE poco energéticos son etiquetados como SIl, incurriendo en un elevado número de elementos borrados. Si tenemos en cuenta esta apreciación, el rendimiento final de los sistemas mejora ostensiblemente.

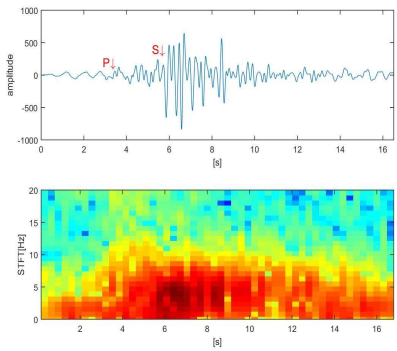
Para finalizar, centraremos el efecto fuente como influencia directa en la remarcada diferencia de rendimiento entre el sistema RNN-LSTM y los sistemas RNN-Vanilla y RNN-GRU.



(a) VTE sin atenuar

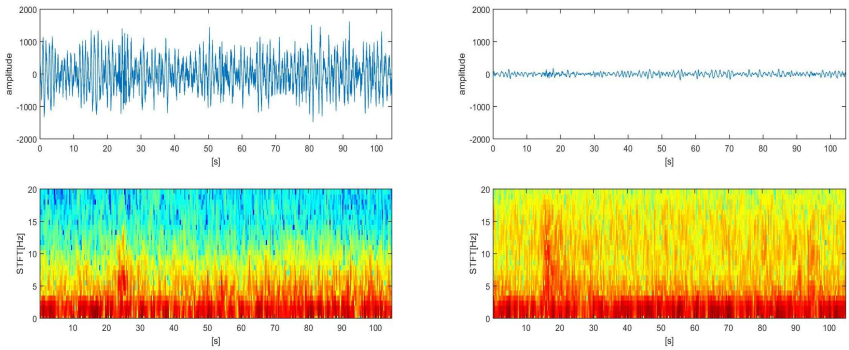


(b) VTE atenuado



(c) VTE muy atenuado

Figura 8.5.2: Efectos de atenuación. Los sismogramas y espectrogramas reflejan cómo la ubicación del sismómetro condiciona la forma y las características del campo de ondas sísmicas.



(a) TRE energético (b) TRE atenuado

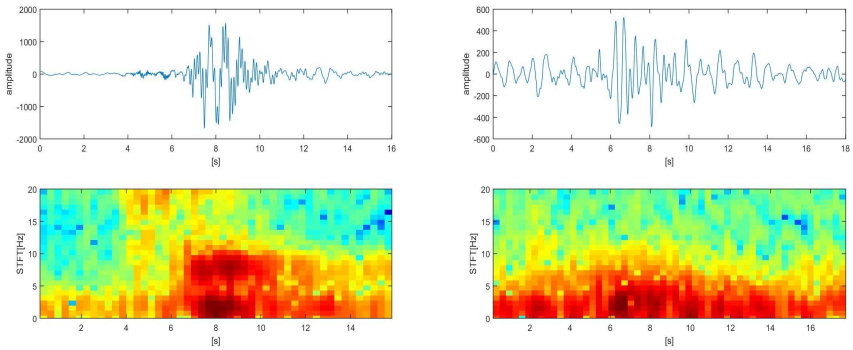
Figura 8.5.3: Efectos de atenuación. Los sismogramas y espectrogramas reflejan cómo dos TREs, con similar o idéntico mecanismo fuente pero a diferente distancia de la estación que los registra, presentando un mismo patrón en su contenido espectral, pueden ser confundidos por los sistemas dadas sus diferencias en el nivel de energía de su forma de onda (amplitud).

En el entorno volcánico de Isla Decepción, los LPEs a menudo son precedidos por un transitorio de presión explosivo en la región de la fuente, por lo que las estaciones que están ubicadas cerca de la misma registran una primera llegada de ondas de alta frecuencia (asemejando la forma de onda del evento a la de un HYB), mientras que las estaciones más alejadas de la fuente solo registran la información espectral clásica de este tipo de eventos.

La Figura 8.5.4 ilustra esta situación. El transitorio explosivo registrado en la Figura 8.5.4a se identifica con un paquete de ondas de alta frecuencia (hasta 20 Hz) ubicado en torno a los 4 segundos en el sismograma. Al no observarse un decaimiento exponencial de la energía y ante le inminente llegada de un paquete de ondas de baja frecuencia, el evento, que en un principio podía haber sido asociado con un VTE por su impulsiva llegada, se termina asociando a un LPE. Esta señal corresponde con un registro recogido a una distancia muy cercana a la fuente.

En cambio, la señal registrada en la Figura 8.5.4b representa el mismo LPE pero registrado en una estación más distante de la fuente presentando un claro decremento de energía asociado al efecto de atenuación. En este escenario, es plausible la confusión entre los eventos LPE y HYB cercanos a la fuente: los sistemas RNN-Vanilla y RNN-GRU, con un mecanismo de memoria menos complejo (menos puertas internas), reconocen cambios en la información entrante más rápidamente, insertando un VTE de corta duración en el etiquetado y disminuyendo por tanto el rendimiento. En el caso del sistema RNN-LSTM, el paquete de altas frecuencias es ignorado dada su corta duración y por lo tanto, no se realizan actualizaciones internas dentro de los estados de las celdas de memoria, siendo reconocido en este caso el evento de mayor duración, el LPE.

El hecho de que tanto la RNN-Vanilla como la RNN-GRU detecten un evento de corta duración al comienzo del registro no debería considerarse un error. Más bien revela la necesidad de crear corpus de datos completos en el que se incluyan eventos de todas las características.



(a) LPE registrado cerca de la fuente. Un paquete de ondas de alta frecuencia asociadas a un transitorio de presión explosivo en la fuente preceden la llegada del paquete de ondas de baja frecuencia

(b) LPE clásico

Figura 8.5.4: Efectos de fuente. Los sismogramas y espectrogramas reflejan cómo la fuente de la señal sísmica influye en la forma de onda y el contenido espectral registrado.

8.6. Conclusiones

Varias han sido las conclusiones extraídas tras el análisis de las RNNs y su aplicabilidad al problema de la detección y clasificación de señales sismo-volcánicas en sistemas de tiempo real.

- En primer lugar, se ha motivado la necesidad de una parametrización eficiente con la que guiar el proceso de detección, ya que la naturaleza y el tamaño de los corpus de datos de los que disponemos impiden el uso directo de registros volcánicos con los que guiar el ajuste de los sistemas.
- En segundo lugar, se ha demostrado que las RNNs pueden ser aplicadas como modelos estadísticos capaces de explotar la información temporal de los registros sísmicos.
- En tercer lugar, los resultados, analizados de forma exhaustiva a posteriori por expertos geofísicos, han mostrado que la naturaleza de las señales sismo-volcánicas y especialmente los efectos de fuente y atenuación, deben ser considerados cuidadosamente a la hora de crear corpus de datos. No obstante, las RNNs han demostrado ser capaces de adaptarse a las situaciones impuestas por estos factores, ajustando la confiabilidad de sus predicciones, y poniendo de manifiesto que este tipo de sistemas deberían ser implementados de forma paralela, es decir, la detección y clasificación de los eventos se debería realizar en función de la información relativa a varias estaciones. Este hecho conlleva la creación de corpus de datos mucho más complejos y de mayor envergadura, por lo que el desarrollo de sistemas de este tipo se proponen como trabajo futuro.
- En cuarto lugar, haciendo uso de los mapas de activación de las unidades ocultas de los modelos, se ha observado que:

- La ausencia de células de memoria sofisticadas y el uso de la información a muy corto plazo en las RNNs clásicas, deriva en una alta actividad neuronal con la que detectar los eventos. A medida que los mecanismos que implementan las células de memoria son más complejos, LSTM y GRU se disminuye la actividad neuronal. El uso de un modelo más complejo deriva en una mayor especialización de las neuronas, siendo necesario por tanto una menor actividad neuronal.
- La delimitación temporal de los eventos una vez detectados, está fuertemente influenciada por el tipo de modelo implementado y por el esquema de parametrización escogido. Por un lado, la capacidad de modelar dependencias temporales a muy largo plazo necesita cambios en la información entrante prolongados para evitar así la constante actualización del estado de la célula de memoria de los modelos. Por tanto, las arquitecturas GRU y LSTM sitúan el final de los eventos generalmente, algunas ventanas de tiempo después que la arquitectura Vanilla. Por otro lado, aunque la llegada de las ondas sísmicas es bien capturada por ambas parametrizaciones (LPC y LFB), LPC presenta dificultades a la hora de delimitar el final o la caída energética de los eventos. Esta característica tiene implicaciones muy importantes, ya que ante la llegada de un enjambre o tren de eventos, el sistema los detecta como uno solo de gran duración, incurriendo en un elevado número de elementos borrados, y por tanto, decrementando su rendimiento.

- Finalmente, se ha probado que este tipo de sistemas tienen la capacidad de generalizar datos correspondientes a diferentes periodos sísmicos de una manera muy eficaz, pudiendo ser usados como una herramienta de etiquetado de registros de datos pertenecientes a campañas recientes

Por tanto, cerramos este capítulo, concluyendo que la implementación (en entornos reales) de clasificadores basados en redes neuronales recurrentes como herramienta de monitoreo volcánico puede mejorar los sistemas actuales de alerta temprana en tiempo real que hasta ahora trabajan en los observatorios.

Parte IV

Conclusiones y líneas de investigación futuras

Capítulo 9

Conclusions

Automatic monitoring systems able to parse raw volcanic data into human-readable information consistently at scale, without explicit programming, and with generalization capabilities across different eruptive periods, would enhance human knowledge about volcanoes. However, volcanic sources are located on very adverse conditions, and nature itself imposes several wave propagation effects that complicate the interpretation of recorded volcano-seismic data. The human cost associated with data labelling and scientific analysis reveals as one of the main bottlenecks when real-time specialized monitoring systems need to be deployed on volcanic-environments.

Deep Learning has become a breaking point in the machine learning discipline, being recently adopted by the geoscience and remote sensing communities. Despite its great modelling and generalization capabilities, deep learning architectures have not experienced an uprising as baseline systems for the automatic recognition of volcanic signals. Hence, in volcanic seismology, deep learning lacks a recognizable application to perform robust classification with extensive applications to multiple eruptive scenarios. The results obtained from this research enlight the capabilities of deep neural networks, as they exploit the information contained within recorded seismic signals, learning long-term dependencies and increasing the overall generalization on real eruptive scenarios. However, the uniqueness of seismic signals pose a challenge for traditional deep learning algorithms, and different approaches for both, isolated and continuous data, need to be adopted. Concretely:

- For isolated classification systems:

    1. DNNs (SDAs y DBNs) have shown that they are able to classify seismic events with higher precision and recall than classical architectures. Moreover, deep architectures are more sensitive to detect complex events, such as explosions and tremors. The use of multiple non-linear processing layers projects the data into a new representation space in which categorization is easier, improving the construction of complex functions that can model decision boundaries between classes within the space of representation. This discriminative improvement contributes to the emission of higher membership probabilities, directly affecting the reliability of classifications.
    2. In our dataset, the underlying architectures during the pre-training stage has a significant impact on the performance and confidence of the systems. The inclusion of hidden layers does not affect the recognition rate of certain

classes if enough data are available: After pre-training step, reconstruction functions for DAs and probability distributions for RBMs are adequately well approximated (features extracted are sufficiently discriminatives), so inclusion of new layers does not improve the results obtained. However, a more linear separable representation space as a direct consequence of deeper models, although it does not improve de recognition rate, contributes to the emission of higher membership probabilities [102].

3. Data analysis using confusion matrices and visual exploration techniques has highlighted the implicit difficulty of the addressed problem. The groups of misclassified events correspond to volcano-seismic signals that have been attenuated or masked with noise. However, even if DAs and RBMS do have a low rate of misclassified events, these experimental results emphasize the importance of creating more complete and representative dataset.
4. When the confidence of the classifications is analyzed by specific thresholds, the reliability of the shallow algorithms decreases. Compared to deep ones, the sensitivity of shallow neural models is seriously affected as more specificity is required. This conclusion acquires special relevance since the use of confidence thresholds in the classifications is a common practice carried out in vulcanological observatories.

- For continuous classification systems:

1. RNNs (Vanilla, LSTM and GRU) have shown that they can be applied as statistical models able to exploit temporal information and model temporal dependencies of continuous seismic streams.
2. The results analyzed by geophysicist have shown that RNNs are scalable models able to adapt to the constraints imposed by attenuation effects and adjust the reliability of their predictions depending on the incoming information. Furthermore, these results have motivated the need to develop models that work in parallel, detecting and classifying signals from distinct seismic stations.
3. Analysis based on activation map of hidden units have demonstrated that in Deception Island dataset:

   a) The absence of sophisticated memory cells and the use of very short term information in Vanilla RNN, leads to a high neuronal activity detecting events. However, LSTM and GRU, implementing more sophisticated memory cells, achieve greater specialization of the units requiring lower neuronal activity.

   b) Automatic segmentation task after event detection is strongly influenced by both, recurrent architecture and parameterization scheme chosen. On the one hand, the capacity to model long term dependencies requires prolonged changes on the incoming information to avoid constant updating memory cell status. Therefore, GRU and LSTM architectures place the end of the events generally, some frames after Vanilla architecture. On the other hand, although the arrival of seismic waves is well captured by LPC and LBF parameterizations, LPC presents some difficulties defining the end or energy drop of events. This characteristic has important implications, since the arrival of train of events or

swarm, the system detects them as a long single one, incurring in an elevated number of misclassified events, and therefore, decreasing their performance.

4. Finally, RNNs have the ability to efficiently generalize data corresponding to different seismic periods, so they can be used as labelling tool for recent survey data. This may lead to massive datasets with consistently classified events, yielding standardized datasets that can be used for further analysis in the science of seismology.

To conclude, classifiers based on deep neural networks can be deployed in real-environments to monitor the seismicity of restless volcanoes, and enhance current early warning systems in real time. More concretely, depth helps to increase the overall generalization capabilities and reliability of the systems, outperforming classical architectures. In this sense, deep neural architectures are very promising alternatives who with signal processing algorithms could serve to improve monitoring and eruption forecasting strategies.

Capítulo 10

Líneas de investigación futuras

La multidisciplinariedad implícita al reconocimiento automático de señales sismo-volcánicas y su relativamente corta existencia como línea de investigación, derivan en multitud de nuevas ideas y nuevos marcos experimentales que se caracterizan por su inmediata aplicabilidad.

Basándonos en las demandas y retos impuestos por los expertos vulcanólogos para desarrollar sistemas de alerta temprana eficientes y eficaces, es este capítulo, describiremos algunas de las ideas más interesantes que han surgido durante el desarrollo de la tesis y que creemos serían de gran interés.

10.1. Funciones de error ponderadas

Uno de los principales inconvenientes en el reconocimiento automático de señales sismo-volcánicas es el desbalanceo de los corpus de datos. Como hemos citado a lo largo del trabajo, cada escenario eruptivo tiene sus propias características (reología del magma, morfología de la estructura volcánica, posición y origen de la fuente magmática), las cuales condicionan los tipos de señales sísmicas que podemos encontrar. En este sentido, en un mismo volcán se pueden dar episodios eruptivos muy diferentes, que resultan en corpus de datos desbalanceados, en los que unos tipos de eventos son cientos de veces más numerosos que otros.

Además de las técnicas específicas de desbalanceo desarrolladas en el área del aprendizaje automático [135], teniendo presente la naturaleza de las técnicas usadas en este trabajo, en las que el conocimiento se adquiere a través de la optimización de una función de error, el uso de medidas capaces de ponderar los errores en función de la relevancia de cada evento (o incluso de la cantidad de eventos de una determinada clase que componen el corpus de datos) podrían suponer un gran avance en la eficacia de los sistemas, ya que podrían sesgar la discriminación de los a priori, eventos más complejos, bien por sus propias características o bien por ser los menos numerosos.

Basándonos en la definición de la función de error descrita en la sección 2.2.2.1, en la que el error de clasificación representa el error cometido por el modelo al clasificar una determinada instancia x con respecto al conocimiento experto implícito asociado a su etiqueta y, una función ponderada se define como una función de error en la que se incluye un término de ponderación α:

$$J = \alpha L_H(y, f(x)) \tag{10.1.1}$$

A diferencia de otras disciplinas, el hiperparámetro α debería ser ajustado en base al conocimiento histórico del volcán, es decir, α vendría dado por la relevancia que cada evento tenga dentro del escenario eruptivo que se está analizando, teniendo los eventos más relevantes, un mayor factor de ponderación.

10.2. Clasificación multietiqueta

A menudo, el proceso de etiquetado de los corpus de datos tiene asociado un fuerte factor humano que repercute directamente en el éxito el fracaso de la futura tarea de reconocimiento automático. Como ya abordamos en el capítulo 1, de forma general, los registros sísmicos se etiquetan en función del conocimiento y de la experiencia del experto que lleva a cabo dicha tarea, por lo que un mismo registro, revisado por diferentes expertos, puede derivar en un suceso diferente de eventos sismo-volcánicos.

A este hecho, que desde el punto de vista del aprendizaje automático condiciona seriamente la caracterización de los eventos, hay que añadirle la más que probable simultánea superposición u ocurrencia de eventos en un mismo instante de tiempo y que es registrada en un solo registro sísmico.

Hasta la fecha, los sistemas de reconocimiento abordaban la detección y clasificación de los diferentes eventos desde la perspectiva de la clasificación multiclase, es decir, un segmento perteneciente a un registro sísmico es clasificado como un determinado evento dentro de un conjunto de posibles clases de eventos. Por lo tanto, teniendo presente la física subyacente característica de la sismología volcánica, se propone el desarrollo de sistemas capaces de llevar a cabo una clasificación multietiqueta, es decir, el desarrollo de sistemas capaces de detectar y clasificar varios eventos en un mismo segmento.

Siguiendo la filosofía de este trabajo, y usando redes neuronales profundas, este esquema podría ser abordado variando la función de error e incluyendo una capaz de medir el error multiobjetivo de las clasificaciones. No obstante, para llevar a cabo el desarrollo de este sistema, los corpus de datos deberían ser revisados y nuevamente etiquetados permitiendo la optimización de los sistemas con base en un criterio multiobjetivo.

10.3. RNNs como algoritmos de picking automático

Tras los resultados obtenidos por las RNNs en la detección y clasificación de eventos sismo-volcánicos en registros continuos y tras el análisis de sus mapas de activación asociados (capítulo 8), se concluyó, que este tipo de arquitecturas podían ser eficazmente entrenadas en la detección de las llegadas de las ondas P y S de los sismos tanto locales al volcán, como regionales.

Aprovechando la componente temporal intrínseca a este tipo de modelos, los registros sísmicos pueden ser analizados en pequeñas ventanas de tiempo haciendo posible la detección de las llegadas de los paquetes de ondas P y S, y por tanto, actuando como algoritmos de picking automático.

Su fácil despliegue permite además que cada una de las estaciones sísmicas instaladas alrededor del volcán, puedan ser analizadas en tiempo real, ayudando además en la posible determinación de su fuente.Llevando a cabo una adaptación de los corpus de datos, pensamos que este tipo de arquitecturas podrían ser una herramienta muy útil

en la detección de sismos, obteniendo importantes índices de reconocimiento y junto a otras técnicas, suponer un gran avance en el campo.

10.4. RNNs bidireccionales

Nuevamente, partiendo de los resultados obtenidos por las RNNs, surgió la idea de hacer uso de las RNNs bidireccionales (sección 4.1.1) como arquitecturas con las que abordar la clasificación y detección de eventos sismo-volcánicos en tiempo real.

El uso de información contextual antes y después del instante de tiempo sujeto de estudio brinda la oportunidad de realizar un análisis más confiable y más profundo de los datos entrantes, lo que directamente puede repercutir en una mejora significativa de la eficacia de los sistemas, siendo además de gran ayuda en el proceso de etiquetado de datos. El mayor inconveniente de este enfoque es la enorme cantidad de datos que demanda el proceso de entrenamiento, pero dada la tendencia creciente de estudio de este área y dado también el creciente volumen de datos que algunos observatorios van registrando día tras día, el desarrollo de este tipo de sistemas se presenta como un reto asequible.

Si además de usar arquitecturas innovadoras capaces de manejar información contextual anterior y posterior, los sistemas son también entrenados en escenarios de clasificación multietiqueta como los anteriormente expuestos, los observatorios vulcanológicos experimentarán un incremento de las prestaciones a nivel de gestión de riesgos, pero sobre todo, a nivel de análisis de datos, lo que mejorará el conocimiento de las dinámicas eruptivas y por consiguiente, los procesos internos que las preceden y aceleran.

Capítulo 11

Divulgación científica

11.1. Artículos en revistas especializadas

- Titos, M., Bueno, A., García, L., & Benítez, C. (2018). A Deep Neural Networks Approach to Automatic Recognition Systems for Volcano-Seismic Events. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(5), 1533-1544.
- Titos, M., Bueno, A., García, L., Benítez, C. & Ibáñez. J (2018). Detection and Classification of Continuous Volcano-Seismic Signals with Recurrent Neural Networks. IEEE Transactions on Geoscience and Remote Sensing.
- García, L., Alvarez, I., Titos, M., Díaz-Moreno, A., Benítez, M. C., & de la Torre, A. (2017). Automatic Detection of Long Period Events Based on Subband-Envelope Processing. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10(11), 5134-5142.
- IBÁÑEZ, Jesús M., et al. TOMO-ETNA experiment at Etna volcano: activities on land. Annals of Geophysics, 2016.
- Romero, J. E., Titos, M., Bueno, Á., Álvarez, I., García, L., de la Torre, Á., & Benítez, M. C. (2016). APASVO: a free software tool for automatic P-phase picking and event detection in seismic traces. Computers & Geosciences, 90, 213-220.
- García, L., Álvarez, I., Benítez, C., Titos, M., Bueno, Á., Mota, S., ... & Prudencio, J. (2016). Advances on the automatic estimation of the P-wave onset time. Annals of Geophysics, 59(4), 0434.

11.2. Ponencias en Congresos Internacionales

- Manuel Titos; Ángel Bueno; Luz García; Carmen Benítez; Jesús Ibáñez; Luz García. Classification of volcano-seismic events based on Deep Neural Networks. Fall Meeting, AGU. New Orleans, United States of America 2017.
- Ángel Bueno; Manuel Titos; Luz García; Carmen Benítez; Jesús Ibáñez. Automatic Seismic-Event Classification with Convolutional Neural Networks. Fall Meeting, AGU New Orleans, United States of America 2017

- Angel Bueno; Rodriguez, Manuel Titos Luzon; Luz Garcia Martinez; Carmen Benitez Ortuzar; Alejandro Diaz Moreno; Silvio De Angelis; and Jesus Ibañez Godoy. A Bayesian Neural Network approach for the classification of volcano-seismic events. EGU, Viena 2018.

- Angel Bueno Rodriguez; Manuel Titos Luzon;Alejandro Diaz Moreno; Silvio De Angelis; Lucciano Zuccarello; Luz Garcia Martinez; Carmen Benitez Ortuzar; and Jesus Ibañez Godoy. Convolutional Neural Networks and their application to automatic classification ofvolcano-seismic events. Cities on Volcanoes 10. Nápoles, Italia 2018

- Manuel Marcelino Titos Luzón; Angel Bueno Rodriguez; Luz García; Isaac Álvarez; Sonia Mota; Jesus Ibañez Godoy and Carmen Benitez Ortuzar. Recurrent Neural Networks as AutomaticVolcano-Seismic Recognition Systems in real-time. Cities on Volcanoes 10. Nápoles, Italia 2018

- Carmen Benítez; Manuel M. Titos Luzón; Luz García; Isaac Álvarez; Ángel De la Torre. A study of classifiers based on NN, HMM, GMM and SVM for the VT, LP and Noises discrimination task. Cities on Volcanoes 8. Yogyakarta, Indonesia 2014.

- Manuel Marcelino Titos Luzón; Isaac Álvarez; Luz García; Ángel De la Torre; Anaïs Boué; Philippe Lesage; Raúl Arambula; Gabriel Reyes-Dávila. Automatic detection of volcanic tremors and other long-duration volcanic events. Cities on Volcanoes 8. Yogyakarta, Indonesia 2014.

11.3. Estancias de investigación

- TomoEtna 2014. Campaña de recogida de datos sísmicos. Sicilia. Junio-Julio, 2014.

- Universidad de Sherbrooke (Canadá). Febrero-Mayo, 2015.

- Universidad de Saboya (Francia). Septiembre, 2016.

- XXX Campaña antártica española. Base Gabriel de Castilla. Enero-Abril, 2017.

11.4. Proyectos de investigación

- (2013-2016) APASVO (Algoritmos avanzados de procesado de señal para la descripción y caracterización de señales sismo-volcánicas).

- (2013-2015) MEDiterranean SUpersite Volcanoes. MED-SUV. EU Grant agreement no: 308665. UGR Partner 14

- (2016-2020) KNOWLEDGE EXTRACTION OF THE STATE OF ACTIVE VOLCANOES AND ITS APPLICATION TO THE MODELLING OF ERUPTION FORECAST BY ADVANCED SEISMIC SIGNAL ANALYSIS. Knowaves. TEC2015-68752-R.

11.5. Entrevistas en medios de comunicación

- El escarabajo verde, Paralelo 60 (La 2, TVE). http://www.rtve.es/alacarta/videos/el-escarabajo-verde/escarabajo-verde-antartida-capitulo-1-paralelo-60/4613887/

Apéndice

Apéndice A

Experimentación complementaria a la clasificación en aislado

A continuación se detalla la experimentación complementaria asociada al problema de la clasificación en aislado de eventos sismo-volcánicos. Cada una de las siguientes tablas corresponde con el rendimiento (medido en términos de F1_Score) por clase obtenido representando cada uno de los tres segmentos diferentes número de LPC y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	97.21	74.88	89.25	95.96	93.96	87.16	93.36	90.25	92.98±0.39
SVM-Lin	96.57	63.65	87.23	94.55	93.83	81.53	92.06	87.06	91.27±0.69
SVM-Rad	97.15	66.60	89.95	94.83	93.77	83.99	92.86	88.45	92.02±0.30
MLP	97.40	79.87	90.39	97.00	94.34	86.61	93.41	91.29	93.41±0.47
DBN-H2	97.52	**80.42**	90.09	97.38	94.29	**88.57**	**94.46**	91.82	**93.99±0.26**
sDA-H2	97.46	78.38	89.84	97.65	**94.46**	88.13	94.12	91.44	93.84±0.45
DBN-H3	**98.00**	79.00	**92.00**	**98.00**	94.00	88.00	94.00	**92.00**	93.82±0.27
sDA-H3	97.00	75.00	91.00	97.00	94.00	86.00	93.00	90.00	93.05±0.49

Tabla A.1: F1_Score por clase obtenido representando cada segmento con 3 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	97.10	78.99	91.63	96.13	95.12	88.86	95.24	91.87	94.16±0.48
SVM-Lin	97.29	73.51	87.69	96.48	94.31	83.44	93.93	89.52	92.81±0.74
SVM-Rad	96.93	78.50	89.61	94.47	94.34	86.73	94.27	90.69	93.11±0.58
MLP	97.91	82.07	**91.53**	97.11	94.70	89.89	94.74	92.57	94.44±0.62
DBN-H2	**98.11**	81.39	90.43	97.10	95.12	**90.72**	95.55	92.63	94.81±0.53
sDA-H2	97.50	**83.02**	91.07	97.50	95.39	90.09	**95.63**	**92.88**	**94.85±0.55**
DBN-H3	97,61	81,06	89,59	**97,76**	**95,55**	89,09	95,20	92,27	94.57±0.66
sDA-H3	97.93	81.97	90.38	97.43	95.15	90.16	95.21	92.60	94.70±0.76

Tabla A.2: F1_Score por clase obtenido representando cada segmento con 6 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	96.59	72.33	87.85	93.89	94.04	87.00	94.02	89.39	92.60±0.64
SVM-Lin	97.28	71.53	87.71	95.70	93.86	84.05	93.52	89.09	92.51±0.43
SVM-Rad	96.97	79.89	87.81	94.87	94.36	86.42	95.26	90.80	93.37±0.39
MLP	97.54	79.70	90.74	96.43	95.21	89.38	95.35	92.05	94.38±0.38
DBN-H2	97.90	76.97	90.19	96.26	94.89	89.48	95.33	91.57	94.21±0.13
sDA-H2	97.53	79.38	89.95	96.40	94.79	89.43	95.17	91.81	94.21±0.32
DBN-H3	**98,21**	78,33	89,33	96,46	95,11	**91,25**	95,59	92,04	94.63±0.44
sDA-H3	97.77	**81.44**	**91.15**	**96.87**	**95.71**	89.09	**95.62**	**92.52**	**94.68±0.42**

Tabla A.3: F1_Score por clase obtenido representando cada segmento con 7 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	96.72	75.28	89.01	93.86	94.67	86.81	94.25	90.09	92.92±0.54
SVM-Lin	97.10	76.43	88.84	95.74	93.81	84.49	93.72	90.02	92.77±0.42
SVM-Rad	96.92	79.54	88.55	93.81	93.77	85.45	94.23	90.32	92.75±0.57
MLP	97.60	83.55	89.77	96.03	94.93	89.52	95.77	92.45	94.51±0.48
DBN-H2	97.24	82.01	90.18	96.40	94.90	89.53	95.88	92.31	94.46±0.48
sDA-H2	97.54	**84.98**	90.55	**96.84**	95.11	90.73	**96.07**	**93.12**	**94.96±0.41**
DBN-H3	97,31	82,79	**91,55**	96,33	**95,23**	89,68	95,60	92,64	94.57±0.47
sDA-H3	**97.80**	84.29	87.84	96.43	94.44	**91.04**	95.82	92.52	94.64±0.85

Tabla A.4: F1_Score por clase obtenido representando cada segmento con 8 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	96.32	71.43	89.64	94.26	94.58	86.47	93.78	89.50	92.64±0.93
SVM-Lin	97.08	74.57	90.37	96.07	94.75	84.11	93.67	90.09	92.96±0.33
SVM-Rad	96.85	72.70	90.28	94.52	94.76	86.44	94.89	90.06	93.13±0.44
MLP	97.35	83.45	90.12	96.42	95.53	90.21	95.58	92.66	94.640±0.29
DBN-H2	97.53	82.46	90.96	**96.97**	**95.67**	90.26	95.63	92.78	94.81±0.28
sDA-H2	97.28	84.30	91.25	96.92	95.59	91.25	**95.82**	93.20	95.00±0.21
DBN-H3	**97,56**	83,48	86,84	96,87	94,60	**91,75**	95,65	92,39	94.68±0.30
sDA-H3	97.22	**84.60**	**91.46**	96.87	95.62	91.39	95.80	**93.28**	**95.02±0.26**

Tabla A.5: F1_Score por clase obtenido representando cada segmento con 9 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	96.91	74.18	86.98	94.36	94.44	87.39	94.02	89.76	92.88±0.62
SVM-Lin	96.92	74.94	88.68	95.95	93.61	84.06	93.30	89.64	92.51±0.16
SVM-Rad	97.12	76.92	88.37	94.82	93.91	86.57	94.46	90.31	93.05±0.56
MLP	97.54	**84.39**	89.03	96.62	94.41	90.74	95.03	92.54	94.40±0.62
DBN-H2	97.37	79.43	87.66	96.33	94.19	91.01	95.62	91.66	94.23±0.41
sDA-H2	**97.60**	82.49	**89.24**	**97.19**	95.05	**91.58**	95.57	**92.68**	**94.81±0.56**
DBN-H3	97,55	82,56	88,29	96,69	**95,09**	90,52	**96,27**	92,43	94.64±0.39
sDA-H3	97.38	80.96	88.12	96.66	94.93	91.32	95.88	92.18	94.61±0.29

Tabla A.6: F1_Score por clase obtenido representando cada segmento con 10 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	97.21	74.39	89.30	93.64	93.82	88.71	94.29	90.19	93.03±0.71
SVM-Lin	96.98	83.17	**90.38**	96.80	93.95	86.82	94.11	91.74	93.61±0.56
SVM-Rad	97.71	72.47	89.25	93.43	93.52	87.37	94.54	89.75	92.85±0.57
MLP	98.03	81.14	90.01	97.50	95.22	90.62	96.07	92.66	95.00±0.76
DBN-H2	**98.40**	81.92	89.53	96.76	94.62	91.51	96.04	92.68	94.94±0.85
sDA-H2	98.10	83.67	89.21	96.90	**95.28**	**91.70**	**96.62**	**93.07**	**95.26±0.44**
DBN-H3	97,50	**84,61**	88,70	96,54	94,32	91,27	95,18	92,59	94.76±0.60
sDA-H3	98.10	83.11	88.54	**97.37**	94.84	91.41	96.17	92.79	95.02±0.62

Tabla A.7: F1_Score por clase obtenido representando cada segmento con 12 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

	SIL	EXP	REG	COL	VTE	TRE	LPE		
	F1	F1	F1	F1	F1	F1	F1	F1 AVG	Acc. Global
RF-120	97.04	81.17	90.53	93.28	94.35	90.71	94.80	91.70	93.71±0.70
SVM-Lin	96.78	**85.83**	**91.14**	96.58	94.73	86.94	94.64	92.38	93.97±0.71
SVM-Rad	96.73	79.03	89.07	93.98	94.59	85.45	94.68	90.50	93.03±0.51
MLP	97.51	84.50	89.41	96.53	94.86	90.65	95.48	92.71	94.61±0.32
DBN-H2	97.51	84.31	88.68	95.88	94.73	90.69	**96.02**	92.54	94.61±0.52
sDA-H2	**97.93**	85.46	89.55	**96.85**	**95.10**	91.39	95.83	**93.16**	**95.00±0.19**
DBN-H3	97,44	84,65	89,26	96,47	94,76	**91,41**	95,90	92,84	94.49±0.29
sDA-H3	97.44	84.65	89.26	96.47	94.76	**91.41**	95.90	92.84	94.78±0.22

Tabla A.8: F1_Score por clase obtenido representando cada segmento con 15 coeficientes de predicción lineal y los percentiles 20º, 50º y 80º de las sumas acumuladas de la señal en el dominio del tiempo y el dominio de la frecuencia.

Apéndice B

Mejores configuraciones encontradas durante el estudio asociadas a cada parametrización

A continuación se detallan las mejores configuraciones obtenidas en el problema de clasificación en aislado con cada una de las parametrizaciones usadas. La Tabla B.1 corresponde a las configuraciones con 2 capas ocultas, mientras que la Tabla B.2 corresponde a las configuraciones con 3 capas ocultas. El número de unidades de entrada (que corresponde con el tamaño del vector de características) y de salida (que corresponde con las siete clases de salida) han sido omitidos en esta tabla por simplicidad, pero es compartido por todos los modelos.

	DBN-H2	sDA-H2
3LPC	280-185	205-115
6LPC	105-245	265-255
7LPC	260-165	260-255
8LPC	255-125	290-135
9LPC	255-125	260-35
10LPC	275-155	290-190
12LPC	225-125	265-140
15LPC	265-95	280-270

Tabla B.1: Configuraciones óptimas obtenidas para sDA y DBN con dos capas ocultas. Cada uno de los valores representa el número de neuronas o unidades ocultas de capa oculta.

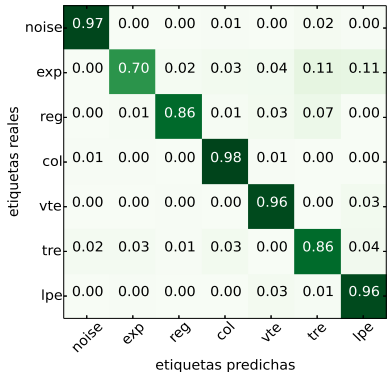
	DBN-H3	sDA-H3
3LPC	145-185-145	85-265-125
6LPC	25-25-225	125-205-105
7LPC	265-245-65	185-105-25
8LPC	125-185-185	185-65-165
9LPC	205-125-245	225-105-25
10LPC	265-205-205	185-85-45
12LPC	185-265-165	185-285-225
15LPC	285-285-85	205-65-105

Tabla B.2: Configuraciones óptimas obtenidas para sDA y DBN con tres capas ocultas. Cada uno de los valores representa el número de neuronas o unidades ocultas de capa oculta.

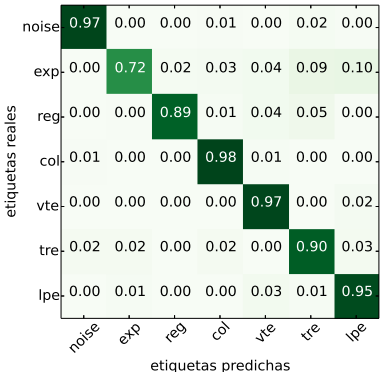
Apéndice C

Matrices confusión asociadas al estudio mediante umbrales probabilísticos

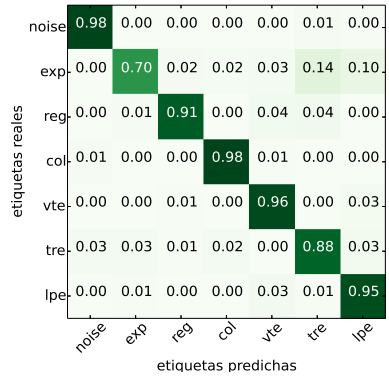
A continuación se detallan las matrices de confusión asociadas al análisis de los sistemas de clasificación en aislado imponiendo umbrales probabilísticos en las predicciones de la salida. Cada figura corresponde con la matriz de confusión asociada a cada uno de los modelos estudiados. El rango de valores estudiados para los umbrales ha variado desde 0.4 hasta 0.8.



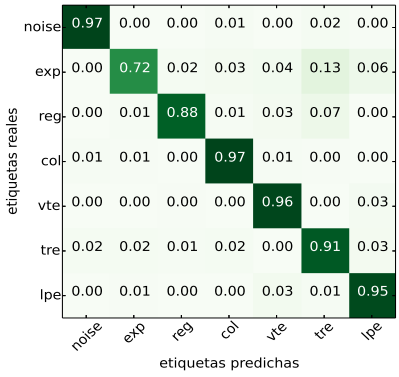
(a) MLP



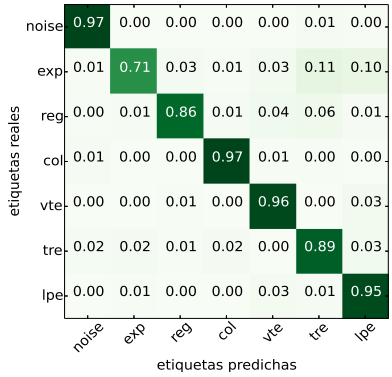
(b) sDA-2H



(c) DBN-2H

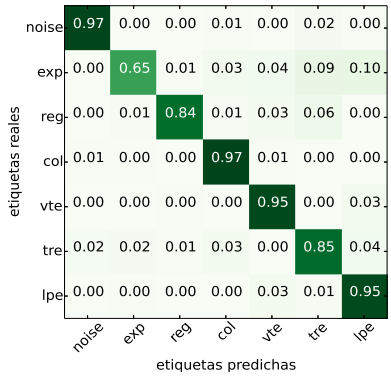


(d) sDA-3H

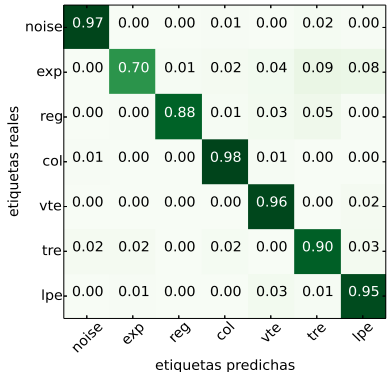


(e) DBN-3H

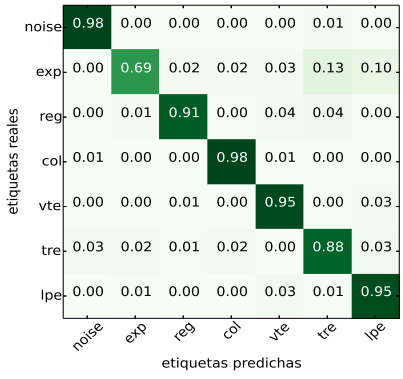
Figura C.0.1: Matrices de confusión asociadas al proceso de clasificación fijando el umbral en 0.4



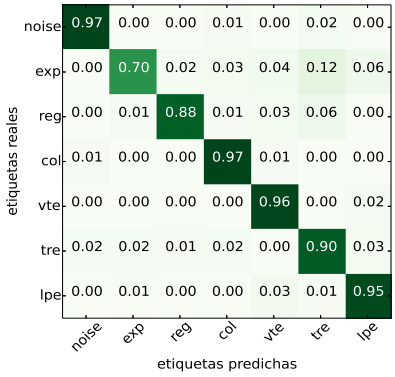
(a) MLP



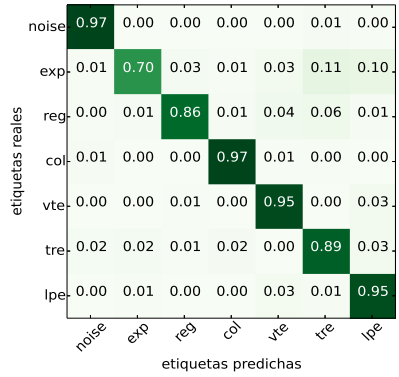
(b) sDA-2H



(c) DBN-2H

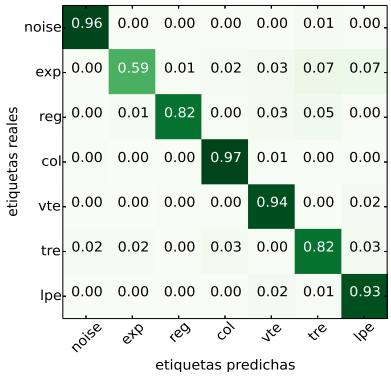


(d) sDA-3H

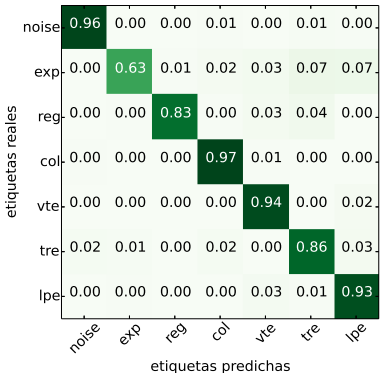


(e) DBN-3H

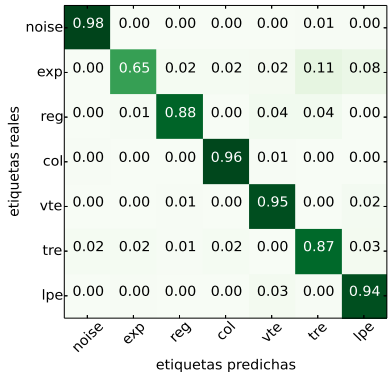
Figura C.0.2: Matrices de confusión asociadas al proceso de clasificación fijando el umbral en 0.5



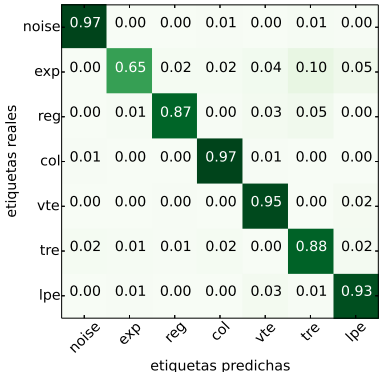
(a) MLP



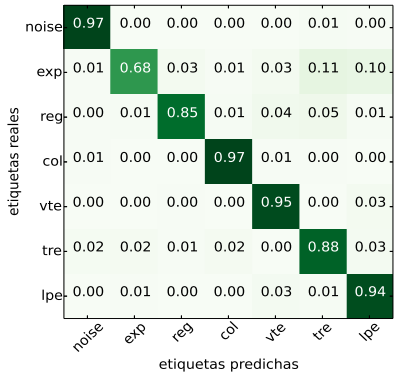
(b) sDA-2H



(c) DBN-2H

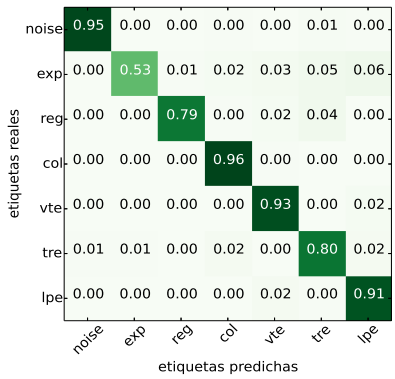


(d) sDA-3H

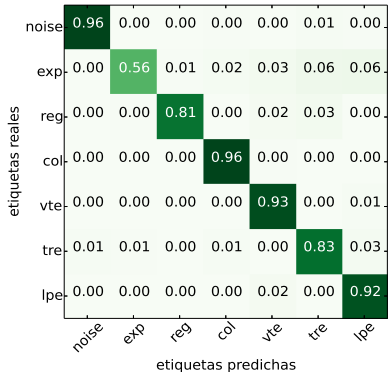


(e) DBN-3H

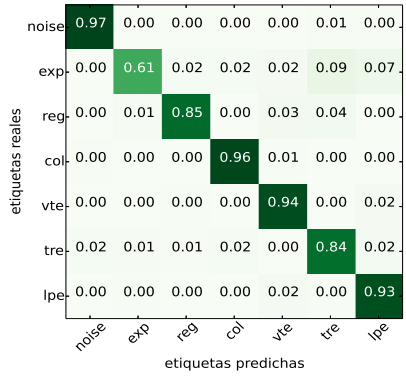
Figura C.0.3: Matrices de confusión asociadas al proceso de clasificación fijando el umbral en 0.6



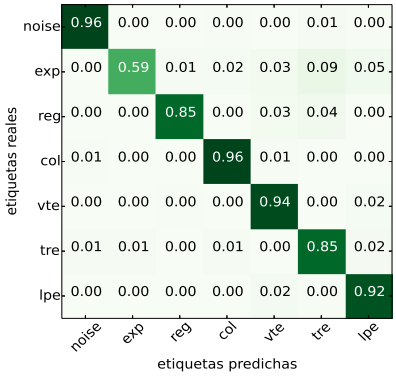
(a) MLP



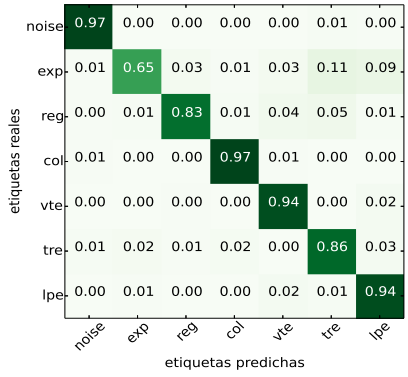
(b) sDA-2H



(c) DBN-2H

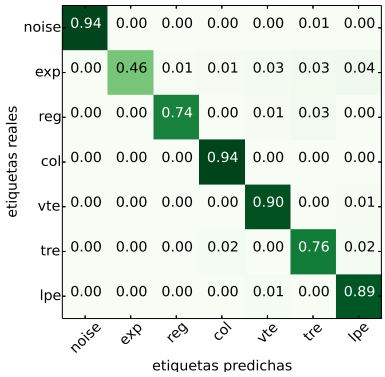


(d) sDA-3H

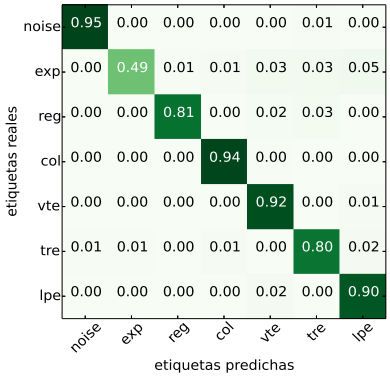


(e) DBN-3H

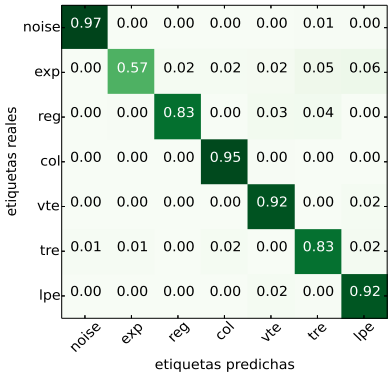
Figura C.0.4: Matrices de confusión asociadas al proceso de clasificación fijando el umbral en 0.7



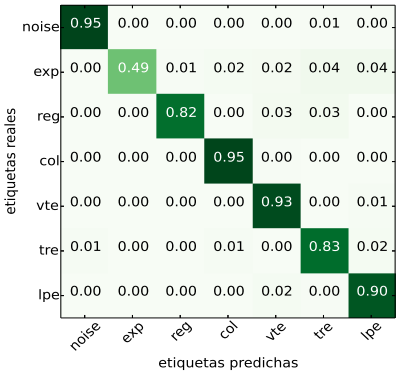
(a) MLP



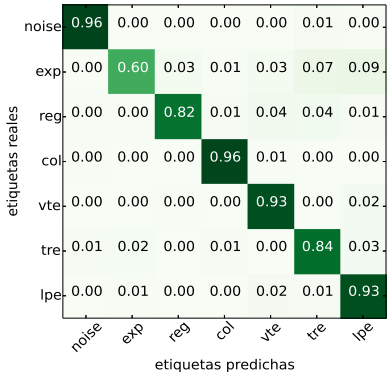
(b) sDA-2H



(c) DBN-2H



(d) sDA-3H



(e) DBN-3H

Figura C.0.5: Matrices de confusión asociadas al proceso de clasificación fijando el umbral en 0.8

Apéndice D

Análisis asociado a la parametrización LPC usando RNN

A continuación se detallan los resultados obtenidos por las RNNs (Vanilla, GRU y LSTM) haciendo uso de diferente número de LPC con los que parametrizar cada frame. La primera columna de las tablas se corresponde con el número de unidades de la capa oculta. Los resultados están expresados en %.

	Vanilla	GRU	LSTM
10	68,287694	78,295571	73,656094
20	68,250883	77,940571	73,981917
30	65,483427	78,291458	74,631679
40	69,853038	77,563947	73,272502
50	66,96738	78,041416	74,866271
60	69,946355	77,398539	75,914645
70	69,466823	78,237712	71,57976
80	67,65601	78,053784	71,803695
90	65,377986	77,120423	74,493754
100	70,441133	77,768755	73,597968
110	67,565262	77,875614	73,445928
120	67,054123	78,315145	75,174457
130	68,127435	77,807891	74,156618
140	65,617764	77,598453	74,233729
150	67,344624	78,048158	75,366354
160	66,686267	75,586647	74,632734
170	66,110653	75,847208	73,668051
180	68,184584	77,21718	74,844438
190	70,443958	77,913344	74,098933
200	65,780079	75,957417	75,775051
210	67,317128	78,40004	74,848086
220	72,521782	77,313459	69,617873
230	69,930756	78,145194	77,179515
240	69,859326	76,144606	74,881279
250	68,181205	75,917792	73,674643
260	66,835421	78,102738	74,216425
270	71,074879	77,596807	73,723423
280	69,144672	77,758646	66,441149
290	69,516563	77,179015	74,206555
300	69,278181	77,032608	73,585099

Tabla D.1: 3 LPC

	Vanilla	GRU	LSTM
10	76,438606	79,673344	77,103931
20	75,332153	78,66022	74,155855
30	75,625539	78,155434	76,652062
40	75,236046	78,471684	76,359284
50	74,746203	80,03056	74,897724
60	76,006621	79,381686	75,215882
70	72,688907	76,937622	76,67917
80	74,231339	78,786224	77,166528
90	74,32121	79,263997	76,2694
100	75,927967	78,712368	78,752041
110	70,202792	78,17291	75,45591
120	74,936587	77,94553	78,479064
130	73,938417	78,002733	74,134576
140	76,295286	78,891695	77,61395
150	75,520325	79,719186	78,206265
160	74,651694	78,785276	78,164768
170	74,573827	79,303002	77,570772
180	74,674857	78,961849	78,222966
190	71,741223	77,235228	78,038615
200	76,319623	76,984727	77,099091
210	74,168825	77,230501	78,015375
220	71,649933	80,250418	79,509604
230	75,087124	78,781253	78,121054
240	74,600154	78,869623	76,743859
250	77,150548	77,571517	77,357626
260	76,117063	78,649008	73,994255
270	71,310091	78,567326	79,415953
280	75,413108	78,246081	78,664905
290	75,476021	76,149356	79,761821
300	73,909676	75,972891	77,215934

Tabla D.2: 5 LPC

	Vanilla	GRU	LSTM
10	72,281283	78,545302	75.,73452
20	77,190351	78,459483	76,999724
30	76,478618	79,667968	77,310348
40	75,398195	77,908874	76,948941
50	77,038836	78,159624	77,28368
60	76,687056	78,191185	77,189428
70	74,933797	76,334524	76,257217
80	75,716662	75,397754	76,438725
90	74,969053	78,730106	75,96162
100	73,828	78,72501	78,768337
110	73,025662	79,646099	77,418637
120	74,928117	79,368544	78,911215
130	73,380125	79,077947	76,755011
140	75,312459	77,675408	77,164084
150	73,775613	78,451824	77,620959
160	75,757754	79,790545	77,212638
170	76,816618	76,347077	76,823014
180	75,361675	77,916765	74,179345
190	75,086629	76,555777	77,264774
200	76,678777	79,158008	76,072568
210	75,281727	79,115272	76,447564
220	76,445055	77,897805	77,224928
230	72,109711	78,732663	76,062095
240	75,120056	78,098428	76,957572
250	73,888862	78,226447	77,399093
260	75,855398	78,822255	76,932907
270	75,457156	77,100778	77,47975
280	75,745255	79,568124	76,622975
290	74,99001	78,788936	76,155549
300	74,875474	75,977582	76,605493

Tabla D.3: 6 LPC

	Vanilla	GRU	LSTM
10	76,230991	77,633989	72,220969
20	75,739378	78,765905	76,54202
30	72,119462	78,59714	76,032734
40	75,768435	78,476071	75,694108
50	72,286254	79,13748	77,258092
60	74,209136	78,728175	76,232165
70	73,825097	77,574086	70,187831
80	76,124525	79,055363	76,952255
90	73,054159	78,60781	75,42128
100	76,094466	77,840614	75,812763
110	74,047679	77,654409	77,712423
120	74,750257	78,160512	77,845246
130	71,760523	78,545398	75,871605
140	73,331189	77,280957	72,663265
150	75,584137	78,446054	72,62398
160	74,702978	78,703713	77,321517
170	75,434983	75,687838	77,720547
180	74,810171	78,362972	75,261664
190	74,58396	76,279187	76,010579
200	76,153255	77,669477	72,813416
210	74,513817	78,877437	77,103949
220	74,492103	77,173543	76,378977
230	74,361408	78,869617	75,952983
240	70,464325	77,154946	76,334941
250	73,818254	78,815991	79,228097
260	74,404263	75,413859	76,207745
270	74,124187	77,58947	76,310855
280	75,240284	78,297842	79,039097
290	76,263183	74,349135	75,595158
300	74,92398	78,452969	75,239158

Tabla D.4: 7 LPC

	Vanilla	GRU	LSTM
10	75,502777	77,704144	74,004287
20	73,563826	78,363615	74,204475
30	74,155074	78,073233	76,706672
40	76,106191	77,397454	77,106339
50	76,111329	77,989274	72,624242
60	76,132852	78,118479	77,707803
70	75,848675	78,110099	77,343738
80	72,93818	77,222466	77,720797
90	74,976164	78,510904	76,047415
100	71,765375	75,231624	76,156598
110	75,079411	77,126092	76,859111
120	75,114352	77,610457	76,012599
130	74,493527	78,672844	74,639404
140	72,915107	79,370481	75,879687
150	74,957371	78,133309	75,476748
160	74,410534	78,930014	75,494307
170	76,094055	78,897697	75,527
180	68,924135	76,264107	76,396829
190	75,641954	77,596724	76,739269
200	74,82748	77,352023	76,487708
210	71,760082	78,428876	74,033433
220	74,744749	76,341462	76,34455
230	75,999451	76,60681	75,878382
240	74,82127	77,239376	76,905853
250	73,886514	76,975024	75,980633
260	73,226887	77,480209	74,333036
270	74,005771	77,538359	72,763216
280	73,18095	76,440245	77,047873
290	75,262135	76,033509	76,432467
300	72,648883	77,02781	75,343418

Tabla D.5: 8 LPC

	Vanilla	GRU	LSTM
10	72,650462	77,88046	76,225847
20	76,615274	77,603149	75,085884
30	77,143044	78,52757	76,884544
40	74,405921	79,087633	78,155667
50	72,262734	77,587628	76,07885
60	74,369347	76,658142	72,567701
70	74,09904	78,192878	74,784857
80	76,007885	76,168758	75,81048
90	73,550981	77,591962	75,936019
100	73,168135	78,097475	76,788688
110	75,639486	76,72478	77,266717
120	72,142231	78,143483	77,471638
130	74,933743	78,493541	74,933434
140	74,484593	75,819927	77,242565
150	72,921956	77,073276	75,222218
160	71,75805	76,645416	76,059914
170	73,654228	76,972038	74,505627
180	75,467318	76,517308	77,604878
190	75,553417	76,180875	76,474917
200	74,548465	76,126695	74,335629
210	75,31386	77,366829	74,310577
220	72,963935	77,202868	75,367498
230	76,095974	74,787295	74,017763
240	74,207973	77,918601	72,587633
250	74,541783	76,489002	74,515998
260	76,781166	79,22473	76,458979
270	74,912566	77,062523	72,697657
280	74,6907	77,7996	77,186507
290	73,425949	78,588098	76,647878
300	75,400311	76,910424	77,003741

Tabla D.6: 9 LPC

	Vanilla	GRU	LSTM
10	71,898532	77,021098	72,386312
20	74,117929	76,807666	75,860476
30	72,731113	78,737998	75,355321
40	75,264859	80,019248	76,412773
50	72,420859	78,557169	76,620895
60	73,428655	77,918708	76,86134
70	73,90123	76,3973	76,16269
80	75,255007	78,456056	76,90146
90	74,32912	76,550192	74,69874
100	75,458306	78,75182	76,618499
110	73,478949	77,178639	76,526642
120	72,100502	76,81396	75,317246
130	72,417462	77,134824	76,483417
140	73,952538	78,000408	72,052443
150	72,560215	76,071936	74,758434
160	73,797303	76,865029	73,636067
170	76,076102	78,637367	74,45184
180	76,915002	75,124156	75,316536
190	74,823785	77,715671	73,609889
200	70,763707	78,279507	75,133348
210	74,606317	76,603562	78,223997
220	73,747563	76,333404	75,735241
230	71,920496	77,659875	75,037891
240	73,666376	76,41511	75,922853
250	75,364387	77,906108	74,500525
260	73,232776	76,268291	75,262386
270	74,962211	77,261353	74,150813
280	73,135942	77,703476	76,256877
290	73,967457	76,490986	76,970541
300	75,17308	77,688133	75,846756

Tabla D.7: 10 LPC

	Vanilla	GRU	LSTM
10	74,837285	79,099011	76,925004
20	75,387198	75,679541	73,795027
30	70,694172	77,466875	74,023533
40	73,929489	77,483886	75,279117
50	73,349714	78,487301	76,249605
60	74,005234	76,580083	75,400048
70	75,590599	77,260715	76,308209
80	74,184084	77,531946	73,576349
90	71,558315	77,919149	76,984155
100	75,240159	78,761661	76,689416
110	71,878588	78,293902	73,693055
120	72,354448	76,216054	75,076616
130	74,743962	73,18753	73,935717
140	75,299978	77,500451	75,411701
150	74,602461	78,425795	77,070558
160	74,772203	77,529919	77,102411
170	74,074769	75,457478	75,318575
180	72,395074	75,658292	71,807444
190	72,638071	77,857763	74,980778
200	74,14484	75,304866	75,311148
210	75,837392	76,885432	76,710773
220	76,387322	76,92287	76,295447
230	74,02842	75,844276	74,663579
240	75,89187	78,694332	76,467073
250	72,373438	76,347214	74,13919
260	75,094199	77,061701	74,544966
270	74,162066	76,606524	76,198208
280	77,171588	77,161968	73,867774
290	74,860334	75,287849	76,893973
300	72,960997	75,217712	74,107003

Tabla D.8: 12 LPC

	Vanilla	GRU	LSTM
10	70,657802	76,517427	75,140679
20	73,699123	78,718579	71,486825
30	72,122627	76,823694	76,906633
40	75,2608	78,843045	75,927299
50	74,966407	79,426253	76,055896
60	74,700147	75,80635	78,141224
70	72,835213	77,362245	75,819618
80	74,425292	78,100473	76,88936
90	73,953021	78,691959	76,849902
100	72,790295	76,471949	75,877404
110	73,008442	77,951646	76,150727
120	74,0453	76,903206	75,938261
130	72,657937	78,566015	75,586861
140	73,944384	76,710141	75,86531
150	74,736053	77,445066	77,995086
160	73,818654	74,79825	75,655776
170	75,330335	77,80211	74,042135
180	72,289807	77,149624	74,193347
190	75,347352	76,360071	76,410538
200	76,087928	77,715349	72,83783
210	74,562335	77,718127	74,732417
220	70,156991	77,653533	71,63738
230	73,950624	78,414714	76,078463
240	74,036294	77,083707	76,181126
250	74,830157	76,213402	73,811173
260	71,857131	77,364063	75,406981
270	75,645369	76,260334	76,079535
280	74,184865	76,671863	73,449373
290	74,967086	76,771063	73,317122
300	75,656974	77,326274	77,065229

Tabla D.9: 15 LPC

Apéndice E

Estudio desvanecimiento/desborde del gradiente usando RNN

A continuación se detallan los resultados obtenidos durante el estudio del desvanecimiento y desborde de los gradientes haciendo uso del percentil 25 % de los valores de los gradientes propagados a través de las arquitecturas. Estos valores se han obtenido durante haciendo uso del evento de más duración dentro del conjunto de test.

		LPC	LPC+($\triangle, \triangle\triangle$)	LBF	LBF+($\triangle, \triangle\triangle$)	Raw
W_xz	update	0.1337	0.1994	0.1848	0.0064	0.0501
W_hz		0.0339	0.1724	0.2373	0.0013	0.0503
W_xr	reset	0.0335	0.0377	0.0925	0.0082	0.0120
W_hr		0.0183	0.0389	0.0894	0.0098	0.0230
W_xh	candidato	0.2811	0.1737	0.3200	0.0739	0.0374
W_hh		0.0823	0.2539	0.2057	0.0252	0.0791
W_hy	salida	0.2921	0.3107	0.2387	0.1610	0.3443

Tabla E.1: Comparativa del percentil 25 % de los valores de los gradientes propagados a través de las arquitecturas GRU. Cada columna corresponda con la configuración óptima de cada parametrización. Cada fila corresponde con una de las matrices de parámetros asociadas a la arquitectura.

	LPC	LPC+($\triangle, \triangle\triangle$)	LFB	LFB+($\triangle, \triangle\triangle$)	Raw
W_uh	0.0788	0.1632	0.0986	0.0488	0.0673
W_hh	0.0878	0.4774	0.2523	0.1290	0.1876
W-hy	0.5954	0.9136	0.5771	0.6369	0.3168

Tabla E.2: Comparativa del percentil 25 % de los valores de los gradientes propagados a través de las arquitecturas Vanilla. Cada columna corresponda con la configuración óptima de cada parametrización. Cada fila corresponde con una de las matrices de parámetros asociadas a la arquitectura.

Bibliografía

[1] Redes neuronales bidireccionales. `http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/`.

[2] Scheme of machine learning area. `https://www.datasciencecentral.com/`.

[3] Rakesh Agrawal, Ramakrishnan Srikant, et al. Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499, 1994.

[4] G. Alguacil, J. Almendros, E. Del Pezzo, A. Garcia, J. Ibañez, M. La Rocca, J. Morales, and R. Ortiz. Observations of volcanic earthquakes and tremor at deception island-antarctica. *Annals of Geophysics*, 42(3), 1999.

[5] JF Allan and ISE Carmichael. Lamprophyric lavas in the colima graben, sw mexico. *Contributions to Mineralogy and Petrology*, 88(3):203–216, 1984.

[6] Rex V Allen. Automatic earthquake recognition and timing from single traces. *Bulletin of the Seismological Society of America*, 68(5):1521–1532, 1978.

[7] J. Almendros, E. Carmona, and J. Ibañez. Precise determination of the relative wave propagation parameters of similar events using a small-aperture seismic array. *Journal of Geophysical Research: Solid Earth*, 109(B11), 2004.

[8] J Almendros, JM Ibáñez, G Alguacil, E Del Pezzo, and R Ortiz. Array tracking of the volcanic tremor source at deception island, antarctica. *Geophysical Research Letters*, 24(23):3069–3072, 1997.

[9] Javier Almendros, Jesús M Ibáñez, Gerardo Alguacil, and Edoardo Del Pezzo. Array analysis using circular-wave-front geometry: an application to locate the nearby seismo-volcanic source. *Geophysical Journal International*, 136(1):159–170, 1999.

[10] Isaac Alvarez, Luz Garcia, Guillermo Cortes, Carmen Benitez, and Ángel De la Torre. Discriminative feature selection for automatic classification of volcano-seismic signals. *IEEE Geoscience and Remote Sensing Letters*, 9(2):151–155, 2012.

[11] Isaac Álvarez, Luz García, Sonia Mota, Guillermo Cortés, Carmen Benítez, and Ángel De la Torre. An automatic p-phase picking algorithm based on adaptive multiband processing. *IEEE Geoscience and Remote Sensing Letters*, 10(6):1488–1492, 2013.

[12] R Arámbula-Mendoza, Philippe Lesage, C Valdés-González, NR Varley, G Reyes-Dávila, and C Navarro. Seismic activity that accompanied the effusive and explosive eruptions during the 2004–2005 period at volcán de colima, mexico. *Journal of Volcanology and Geothermal Research*, 205(1-2):30–46, 2011.

[13] Alejandra Arciniega-Ceballos, Bernard A Chouet, and Phillipe Dawson. Very long-period signals associated with vulcanian explosions at popocatepetl volcano, mexico. *Geophysical Research Letters*, 26(19):3013–3016, 1999.

[14] James Baker. The dragon system–an overview. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):24–29, 1975.

[15] C. Balas, L. Koç, and L. Balas. Predictions of missing wave data by recurrent neuronets. *Journal of waterway, port, coastal, and ocean engineering*, 130(5):256–265, 2004.

[16] Mark S Bebbington. Identifying volcanic regimes using hidden markov models. *Geophysical Journal International*, 171(2):921–942, 2007.

[17] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *Advances in neural information processing systems*, pages 153–160, 2007.

[18] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.

[19] Carmen Benitez, Alberto Alos, Janire Prudencio, Isaac Alvarez, Angel de la Torre, et al. A comparative study of classifiers based on hmm, gmm and svm for the vt, lp and noises discrimination task. In *EGU General Assembly Conference Abstracts*, volume 16, 2014.

[20] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

[21] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: A cpu and gpu math compiler in python. In *Proc. 9th Python in Science Conf*, pages 1–7, 2010.

[22] M Beyreuther and J Wassermann. Hidden semi markov model based earthquake classification system using weighted finite state transducers. *Nonlinear Processes in Geophysics*, 18(1):81, 2011.

[23] Manuele Bicego, Carolina Acosta-Muñoz, and Mauricio Orozco-Alzate. Classification of seismic volcanic signals using hidden markov model based generative embeddings. *IEEE Transactions on Geoscience and Remote Sensing*, 51(6):3400–3409, 2013.

[24] C Bishop. Pattern recognition and machine learning (information science and statistics), 1st edn. 2006. corr. 2nd printing edn. *Springer, New York*, 2007.

[25] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[26] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[27] Anaïs Boué, Philippe Lesage, Guillermo Cortés, Bernard Valette, and G Reyes-Dávila. Real-time eruption forecasting using the material failure forecast method with a bayesian approach. *Journal of Geophysical Research: Solid Earth*, 120(4):2143–2161, 2015.

[28] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[29] J. Cao and J. Wang. Global asymptotic stability of a general class of recurrent neural networks with time-varying delays. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 50(1):34–44, 2003.

[30] Enrique Carmona, Javier Almendros, Inmaculada Serrano, Daniel Stich, and Jesús M Ibáñez. Results of seismic monitoring surveys of deception island volcano, antarctica, from 1999–2011. *Antarctic Science*, 24(5):485–499, 2012.

[31] R Carniel, L Barbui, and AD Jolly. Detecting dynamical regimes by self organizing map (som) analysis: an example from the march 2006 phreatic eruption at raoul island, new zealand kermadec arc. *Bollettino di Geofisica Teorica ed Applicata*, 54(1), 2013.

[32] Roberto Carniel, Mauro Di Cecca, and Olivier Jaquet. A user-friendly, dynamic web environment for remote data browsing and analysis of multiparametric geophysical data within the multimo project. *Journal of volcanology and geothermal research*, 153(1-2):80–96, 2006.

[33] Roberto Carniel, Arthur D Jolly, and Luca Barbui. Analysis of phreatic events at ruapehu volcano, new zealand using a new som approach. *Journal of Volcanology and Geothermal Research*, 254:69–79, 2013.

[34] Miguel A Carreira-Perpinan and Geoffrey E Hinton. On contrastive divergence learning. In *Proceedings of the tenth international workshop on artificial intelligence and statistics*, pages 33–40. Society for Artificial Intelligence and Statistics NP, 2005.

[35] Shan Carter and Michael Nielsen. Using artificial intelligence to augment human intelligence. *Distill*, 2(12):e9, 2017.

[36] Rich Caruana. Multitask learning. In *Learning to learn*, pages 95–133. Springer, 1998.

[37] Rich Caruana, Steve Lawrence, and C Lee Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408, 2001.

[38] Markov Chains. Gibbs fields, monte carlo simulation, and queues, 1999.

[39] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27, 2011.

[40] C Chen. Comparison of seismic features extracted by digital signal processing techniques. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'77.*, volume 2, pages 148–150. IEEE, 1977.

[41] CH Chen. Seismic pattern recognition. *Geoexploration*, 16(1-2):133–146, 1978.

[42] Yushi Chen, Hanlu Jiang, Chunyang Li, Xiuping Jia, and Pedram Ghamisi. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10):6232–6251, 2016.

[43] Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected topics in applied earth observations and remote sensing*, 7(6):2094–2107, 2014.

[44] Vladimir Cherkassky and Filip Mulier. *Learning from data: Concepts, theory, and methods*. Wiley New York, 1998.

[45] K. Cho, B. Van Merriënboer, C. Gulcehre, Caglar, D. Bahdanau, F. Bougares, H. Schwenk, Holger, and Y.Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[46] B. Chouet. Dynamics of a fluid-driven crack in three dimensions by the finite difference method. *Journal of Geophysical Research: Solid Earth*, 91(B14):13967–13992, 1986.

[47] B. Chouet. Resonance of a fluid-driven crack: Radiation properties and implications for the source of long-period events and harmonic tremor. *Journal of Geophysical Research: Solid Earth*, 93(B5):4375–4400, 1988.

[48] B. Chouet. Volcano seismology. *Pure and Applied Geophysics*, 160(3-4):739–788, 2003.

[49] Bernard A Chouet. Long-period volcano seismicity: its source and use in eruption forecasting. *Nature*, 380(6572):309, 1996.

[50] Bernard A Chouet and Robin S Matoza. A multi-decadal view of seismic methods for detecting precursors of magma movement and eruption. *Journal of Volcanology and Geothermal Research*, 252:108–175, 2013.

[51] Bernard A Chouet, Robert A Page, Christopher D Stephens, John C Lahr, and John A Power. Precursory swarms of long-period events at redoubt volcano (1989–1990), alaska: their origin and use as a forecasting tool. *Journal of Volcanology and Geothermal Research*, 62(1-4):95–135, 1994.

[52] G. Cortés, R. Arámbula, L. Gutiérrez, C. Benítez, J. Ibánez, P. Lesage, I. Alvarez, and L. Garcia. Evaluating robustness of a hmm-based classification system of volcano-seismic events at colima and popocatepetl volcanoes. In *Geoscience and Remote Sensing Symposium,2009 IEEE International,IGARSS 2009*, volume 2, pages II–1012. IEEE, 2009.

[53] Guillermo Cortés, Raúl Arámbula, LigdamisA Gutiérrez, Carmen Benítez, Jesús Ibánez, Philippe Lesage, Isaac Alvarez, and Luz Garcia. Evaluating robustness of a hmm-based classification system of volcano-seismic events at colima and popocatepetl volcanoes. In *Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009*, volume 2, pages II–1012. IEEE, 2009.

[54] Guillermo Cortés, Luz García, Isaac Álvarez, Carmen Benítez, Ángel de la Torre, and Jesús Ibáñez. Parallel system architecture (psa): An efficient approach for automatic recognition of volcano-seismic events. *Journal of Volcanology and Geothermal Research*, 271:1–10, 2014.

[55] Guillermo Cortés Moreno. Reconocimiento de señales sismo-volcánicas mediante canales específicos basados en modelos ocultos de markov. 2016.

[56] Julia M Crummy, Ivan P Savov, Carlos Navarro-Ochoa, Daniel J Morgan, and Marjorie Wilson. High-k mafic plinian eruptions of volcán de colima, méxico. *Journal of Petrology*, 55(11):2155–2192, 2014.

[57] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das. Very deep convolutional neural networks for raw waveforms. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 421–425. IEEE, 2017.

[58] Universidad Nacional Autónoma de México. Instituto de Geología and Abel Cortés. *Carta geológica del complejo volcánico de Colima*. UNAM, Instituto de Geología, 2005.

[59] Edoardo Del Pezzo, Anna Esposito, Flora Giudicepietro, Maria Marinaro, Marcello Martini, and Silvia Scarpetta. Discrimination of earthquakes and underwater explosions using neural networks. *Bulletin of the Seismological Society of America*, 93(1):215–223, 2003.

[60] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, et al. Recent advances in deep learning for speech research at microsoft. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8604–8608. IEEE, 2013.

[61] Li Deng, Dong Yu, et al. Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387, 2014.

[62] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[63] Richard O Duda, Peter E Hart, David G Stork, et al. *Pattern classification*, volume 2. Wiley New York, 1973.

[64] J. Ibañez E. Del Pezzo, M. La Rocca. Observations of high-frequency scattered waves using dense arrays at teide volcano. *Bulletin of the Seismological Society of America*, 87(6):1637–1647, 1997.

[65] J. Ibañez E. Del Pezzo, M. Simini. Separation of intrinsic and scattering q for volcanic areas: a comparison between etna and campi flegrei. *Journal of volcanology and geothermal research*, 70(3-4):213–219, 1996.

[66] Mohamed Elhoseiny, Sheng Huang, and Ahmed Elgammal. Weather classification with deep convolutional neural networks. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 3349–3353. IEEE, 2015.

[67] Elliot T Endo and Tom Murray. Real-time seismic amplitude measurement (rsam): a volcano monitoring and prediction tool. *Bulletin of Volcanology*, 53(7):533–545, 1991.

[68] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 11(Feb):625–660, 2010.

[69] Dumitru Erhan, Pierre-Antoine Manzagol, Yoshua Bengio, Samy Bengio, and Pascal Vincent. The difficulty of training deep architectures and the effect of unsupervised pre-training. In *Artificial Intelligence and Statistics*, pages 153–160, 2009.

[70] AM Esposito, F Giudicepietro, L DAuria, S Scarpetta, MG Martini, M Coltelli, and M Marinaro. Unsupervised neural analysis of very long period events at stromboli volcano using the self organizing maps. *Bulletin of the Seismological Society of America*, 98(5):2449–2459, 2008.

[71] AM Esposito, F Giudicepietro, S Scarpetta, L DAuria, M Marinaro, and M Martini. Automatic discrimination among landslide, explosion-quake, and microtremor seismic signals at stromboli volcano using neural networks. *Bulletin of the Seismological Society of America*, 96(4A):1230–1240, 2006.

[72] S Falsaperla, S Graziani, G Nunnari, and S Spampinato. Automatic classification of volcanic earthquakes by using multi-layered neural networks. *Natural Hazards*, 13(3):205–228, 1996.

[73] Asja Fischer and Christian Igel. An introduction to restricted boltzmann machines. In *Iberoamerican Congress on Pattern Recognition*, pages 14–36. Springer, 2012.

[74] Tobias P Fischer, Meghan M Morrissey, V Marta Lucía Calvache, M Diego Gomez, C Roberto Torres, John Stix, and Stanley N Williams. Correlations between so2 flux and long-period seismicity at galeras volcano. *Nature*, 368(6467):135–137, 1994.

[75] G David Forney. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[76] Jerome H Friedman. On bias, variance, 0 1 loss and the curse of dimensionality. *Data mining and knowledge discovery*, 1(1):55–77, 1997.

[77] Luz García, Isaac Álvarez, Carmen Benítez, Manuel Titos, Ángel Bueno, Sonia Mota, Angel De la Torre, José C Segura, Gerardo Alguacil, Alejandro Díaz-Moreno, et al. Advances on the automatic estimation of the p-wave onset time. *Annals of Geophysics*, 59(4):0434, 2016.

[78] Luz García, Isaac Alvarez, Manuel Titos, Alejandro Díaz-Moreno, M Carmen Benítez, and Angel de la Torre. Automatic detection of long period events based on subband-envelope processing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(11):5134–5142, 2017.

[79] Víctor Hugo Garduño-Monroy, Ricardo Saucedo-Girón, Zenón Jiménez, Juan Carlos Gavilanes-Ruiz, Abel Cortes-Cortés, and Rosa María Uribe-Cifuentes. La falla tamazula, límite suroriental del bloque jalisco y sus relaciones con el complejo volcánico de colima, méxico. *Revista Mexicana de Ciencias Geológicas*, 15(2):132–144, 1998.

[80] F. Giacco, A. Esposito, M. Antonietta, S. Scarpetta, F. Giudicepietro, and M. Marinaro. Support vector machines and mlp for automatic classification of seismic signals at stromboli volcano. In *Neural Nets WIRN09: Proceedings of the 19th Italian Workshop on Neural Nets, Vietri Sul Mare, Salerno, Italy May 28-30 2009*, volume 204, page 116. IOS Press, 2009.

[81] 2015. Report on Deception Island (Antarctica). Global Volcanism Program. Deception island. `https://volcano.si.edu/showreport.cfm?doi=10.5479/si.GVP.BGVN201506-39003`.

[82] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[83] Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[84] Oscar González-Ferrán et al. *Volcanes de Chile*. Instituto Geográfico Militar, 1995.

[85] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[86] Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. *arXiv preprint arXiv:1302.4389*, 2013.

[87] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[88] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM, 2006.

[89] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015.

[90] Fredric M Ham, Ishwarya Iyengar, Bereket M Hambebo, Milton Garces, John Deaton, Anna Perttu, and Brian Williams. A neurocomputing approach for monitoring plinian volcanic eruptions using infrasound. *Procedia Computer Science*, 13:7–17, 2012.

[91] Junwei Han, Dingwen Zhang, Gong Cheng, Lei Guo, and Jinchang Ren. Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Transactions on Geoscience and Remote Sensing*, 53(6):3325–3337, 2015.

[92] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[93] John A Hartigan. Clustering algorithms. 1975.

[94] Mohamad H Hassoun. *Fundamentals of artificial neural networks*. MIT press, 1995.

[95] J. Havskov, J. Peña, J. Ibañez, L. Ottemoller, and C. Martinez-Arevalo. Magnitude scales for very local earthquakes. application for deception island volcano (antarctica). *Journal of volcanology and geothermal research*, 128(1):115–133, 2003.

[96] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[97] Clément Hibert, Floriane Provost, Jean-Philippe Malet, André Stumpf, Alessia Maggi, and Valérie Ferrazzini. Automated classification of seismic sources in large database using random forest algorithm: First results at piton de la fournaise volcano (la réunion). In *EGU General Assembly Conference Abstracts*, volume 18, page 12895, 2016.

[98] G. Hinton, Li Deng, D. Yu, G-E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. Sainath. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012.

[99] G. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[100] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, page 14, 2012.

[101] Geoffrey E Hinton. A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*, pages 599–619. Springer, 2012.

[102] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[103] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[104] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[105] Fan Hu, Gui-Song Xia, Jingwen Hu, and Liangpei Zhang. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11):14680–14707, 2015.

[106] J. Ibañez, J. Almendros, E. Carmona, C. Marti, M. Abril, et al. The recent seismo-volcanic activity at deception island volcano. *Deep Sea Research Part II: Topical Studies in Oceanography*, 50(10):1611–1629, 2003.

[107] Jesús M Ibánez, Edoardo Del Pezzo, Javier Almendros, Mario La Rocca, Gerardo Alguacil, Ramón Ortiz, and Alicia García. Seismovolcanic signals at deception island volcano, antarctica: Wave field analysis and source modeling. *Journal of Geophysical Research: Solid Earth*, 105(B6):13905–13931, 2000.

[108] J.M. Ibánez, C. Benítez, L. Gutiérrez, G. Cortés, A. García-Yeguas, and G. Alguacil. The classification of seismo-volcanic signals using hidden markov models as applied to the stromboli and etna volcanoes. *Journal of Volcanology and Geothermal Research*, 187(3):218–226, 2009.

[109] J.M. Ibanez, C. Benítez, L. Gutiérrez, G. Cortés, A. García-Yeguas, and G. Alguacil. The classification of seismo-volcanic signals using hidden markov models as applied to the stromboli and etna volcanoes. *Journal of Volcanology and Geothermal Research*, 187(3):218–226, 2009.

[110] Giuseppe Imbò. *Sismicità del parossismo vesuviano del marzo 1944*. Stabil. Tipogr. G. Genovese, 1954.

[111] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456, 2015.

[112] J. Almendros G. Saccorotti E. Del Pezzo M. Abril R. Ortiz J. Ibañez, E. Carmona. The 1998–1999 seismic series at deception island volcano, antarctica. *Journal of volcanology and geothermal research*, 128(1):65–88, 2003.

[113] Frederick Jelinek. Continuous speech recognition by statistical methods. *Proceedings of the IEEE*, 64(4):532–556, 1976.

[114] Ian T Jolliffe. Principal component analysis and factor analysis. In *Principal component analysis*, pages 115–128. Springer, 1986.

[115] AD Jolly, P Jousset, JJ Lyons, R Carniel, N Fournier, B Fry, and C Miller. Seismo acoustic evidence for an avalanche driven phreatic eruption through a beheaded hydrothermal system: an example from the 2012 tongariro eruption. *Journal of Volcanology and Geothermal Research*, 286:331–347, 2014.

[116] M. Karamouz, S. Razavi, , and S. Araghinejad. Long-lead seasonal rainfall forecasting using time-delay recurrent neural networks: a case study. *Hydrological Processes*, 22(2):229–241, 2008.

[117] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[118] Rita Kovordányi and Chandan Roy. Cyclone track forecasting based on satellite images using artificial neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6):513–521, 2009.

[119] Anita Krishnakumar. Active learning literature survey. Technical report, Technical Report, University of California, Santa Cruz, 2007.

[120] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[121] D. Kumar, K. Raju, , and T. Sathish. River flow forecasting using recurrent neural networks. *Water resources management*, 18(2):143–161, 2004.

[122] Kshitiz Kumar, Chanwoo Kim, and Richard M Stern. Delta-spectral cepstral coefficients for robust speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 4784–4787. IEEE, 2011.

[123] K. Kurach and K. Pawlowski. Predicting dangerous seismic activity with recurrent neural networks. In *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on*, pages 239–243. IEEE, 2016.

[124] M. Kuroda, A. Vidal, A. Maria, and A. De Carvalho. Interpretation of seismic multiattributes using a neural network. *Journal of Applied Geophysics*, 85:15–24, 2012.

[125] H Langer, S Falsaperla, M Masotti, R Campanini, S Spampinato, and A Messina. Synopsis of supervised and unsupervised pattern classification techniques applied to volcanic tremor data at mt etna, italy. *Geophysical Journal International*, 178(2):1132–1144, 2009.

[126] Hugo Larochelle, Dumitru Erhan, Aaron Courville, James Bergstra, and Yoshua Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pages 473–480. ACM, 2007.

[127] Quoc V Le, Navdeep Jaitly, and Geoffrey E Hinton. A simple way to initialize recurrent networks of rectified linear units. *arXiv preprint arXiv:1504.00941*, 2015.

[128] Quoc V Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, and Andrew Y Ng. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 265–272. Omnipress, 2011.

[129] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[130] Yann LeCun et al. Lenet-5, convolutional neural networks. *URL: http://yann. lecun. com/exdb/lenet*, page 20, 2015.

[131] R. Lee and J.Liu. Tropical cyclone identification and tracking system using integrated neural oscillatory elastic graph matching and hybrid rbf network track mining techniques. *IEEE Transactions on Neural Networks*, 11(3):680–689, 2000.

[132] Robert C Leet. Saturated and subcooled hydrothermal boiling in groundwater flow channels as a source of harmonic tremor. *Journal of Geophysical Research: Solid Earth*, 93(B5):4835–4849, 1988.

[133] L.Gutiérrez, J.M Ibánez, G. Cortés, J. Ramírez, C. Benítez, V. Tenorio, and A. Isaac. Volcano-seismic signal detection and classification processing using hidden markov models. application to san cristóbal volcano, nicaragua. In *Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009*, volume 4, pages IV–522. IEEE, 2009.

[134] Yunjie Liu, Evan Racah, Joaquin Correa, Amir Khosrowshahi, David Lavers, Kenneth Kunkel, Michael Wehner, William Collins, et al. Application of deep convolutional neural networks for detecting extreme weather in climate datasets. *arXiv preprint arXiv:1605.01156*, 2016.

[135] Victoria López, Alberto Fernández, Salvador García, Vasile Palade, and Francisco Herrera. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, 250:113–141, 2013.

[136] James F Luhr, Stephen A Nelson, James F Allan, and Ian SE Carmichael. Active rifting in southwestern mexico: Manifestations of an incipient eastward spreading-ridge jump. *Geology*, 13(1):54–57, 1985.

[137] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[138] James Martens. Deep learning via hessian-free optimization. In *ICML*, volume 27, pages 735–742, 2010.

[139] C. Martinez-Arevalo, F. Bianco, J. Ibañez, and E. Del Pezzo. Shallow seismic attenuation and shear-wave splitting in the short period range of deception island volcano (antarctica). *Journal of volcanology and geothermal research*, 128(1):89–113, 2003.

[140] Stefano Masiello, Antonietta M Esposito, Silvia Scarpetta, Flora Giudicepietro, Anna Esposito, Maria Marinaro, et al. Application of self organized maps and curvilinear component analysis to the discrimination of the vesuvius seismic signals. In *WSOM*, 2006.

[141] M. Masotti, S. Falsaperla, H. Langer, S. Spampinato, and R. Campanini. Application of support vector machines to the classification of volcanic tremor at etna, italy. *Geophysical research letters*, 33(20), 2006.

[142] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[143] Stephen R McNutt. Volcanic seismology. *Annu. Rev. Earth Planet. Sci.*, 32:461–491, 2005.

[144] Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Scholkopf, and Klaus-Robert Mullers. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop.*, pages 41–48. IEEE, 1999.

[145] TAKESHI MINAKAMI. Prediction of volcanic eruptions. In *Developments in Solid Earth Geophysics*, volume 6, pages 313–333. Elsevier, 1974.

[146] Dmytro Mishkin and Jiri Matas. All you need is a good init. *arXiv preprint arXiv:1511.06422*, 2015.

[147] Vanessa Jiménez Morales, Javier Almendros, and Enrique Carmona. Detection of long-duration tremors at deception island volcano, antarctica. *Journal of Volcanology and Geothermal Research*, 347:234–249, 2017.

[148] Guillermo Cortes Moreno. *Reconocimiento de señales sismo-volcánicas mediante canales específicos basados en modelos ocultos de Markov*. PhD thesis, Universidad de Granada, 2015.

[149] Michael D Murphy and James A Cercone. Neural network techniques applied to seismic event classification. In *System Theory, 1993. Proceedings SSST'93., Twenty-Fifth Southeastern Symposium on*, pages 343–347. IEEE, 1993.

[150] Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/k^ 2). In *Doklady AN USSR*, volume 269, pages 543–547, 1983.

[151] Jürgen Neuberg and Tim Pointer. Effects of volcano topography on seismic broad-band waveforms. *Geophysical Journal International*, 143(1):239–248, 2000.

[152] CG Newhall. Volcano warnings. *Encyclopaedia of volcanoes, Academic, New York*, pages 1185–1197, 2000.

[153] Matthias Ohrnberger. Continuous automatic classification of seismic signals of volcanic origin at mt. merapi, java, indonesia. 2001.

[154] Chris Olah. Visualizing representations: Deep learning and human beings. *colah. github. io*, 2015.

[155] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 3(3):e10, 2018.

[156] Fusakichi Ômori. The usu-san eruption and earthquake and elevation phenomena. *Bulletin of the Imperial Earthquake Investigation Committee*, 5:1, 1911.

[157] Mauricio Orozco-Alzate, Carolina Acosta-Muñoz, and John Makario Londoño-Bonilla. The automated identification of volcanic earthquakes: concepts, applications and challenges. In *Earthquake Research and Analysis-Seismology, Seismotectonic and Earthquake Geology*. InTech, 2012.

[158] Douglas O'Shaughnessy. Linear predictive coding. *IEEE potentials*, 7(1):29–32, 1988.

[159] Nikunj C Oza. Online bagging and boosting. In *Systems, man and cybernetics, 2005 IEEE international conference on*, volume 3, pages 2340–2345. Ieee, 2005.

[160] A. Jolly P. Jousset, J. Neuberg. Modelling low-frequency volcanic earthquakes in a viscoelastic medium with topography. *Geophysical Journal International*, 159(2):776–802, 2004.

[161] Mahesh Pal. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1):217–222, 2005.

[162] M Palo, JM Ibáñez, M Cisneros, M Bretón, E Del Pezzo, E Ocana, J Orozco-Rojas, and AM Posadas. Analysis of the seismic wavefield properties of volcanic explosions at volcan de colima, mexico: insights into the source mechanism. *Geophysical Journal International*, 177(3):1383–1398, 2009.

[163] A. Panakkat and H. Adeli. Recurrent neural network for approximate earthquake time and location prediction using multiple seismicity indicators. *Computer-Aided Civil and Infrastructure Engineering*, 24(4):280–292, 2009.

[164] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. *International Conference on Machine Learning*, pages 1310–1318, 2013.

[165] Jon Peterson et al. Observations and modeling of seismic background noise. 1993.

[166] E. Del Pezzo, A. Esposito, F. Giudicepietro, M. Marinaro, M. Martini, and S. Scarpetta. Discrimination of earthquakes and underwater explosions using neural networks. *Bulletin of the Seismological Society of America*, 93(1):215–223, 2003.

[167] E. Del Pezzo, J. Ibañez, J. Morales, and R. Maresca A. Akinci. Measurements of intrinsic and scattering seismic attenuation in the crust. *Bulletin of the Seismological Society of America*, 85(5):1373–1380, 1995.

[168] Lorien Y Pratt. Discriminability-based transfer between neural networks. In *Advances in neural information processing systems*, pages 204–211, 1993.

[169] Lutz Prechelt. Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, 11(4):761–767, 1998.

[170] F Provost, C Hibert, and J-P Malet. Automatic classification of endogenous landslide seismicity using the random forest supervised classifier. *Geophysical Research Letters*, 44(1):113–120, 2017.

[171] J. Prudencio, L. De Siena, J. Ibanez, E. Del Pezzo, A. Garcia-Yeguas, and A. Diaz-Moreno. The 3d attenuation structure of deception island (antarctica). *Surveys in Geophysics*, 36(3):371–390, 2015.

[172] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999.

[173] Lawrence R Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*, volume 14. PTR Prentice Hall Englewood Cliffs, 1993.

[174] JA Rogers and CD Stephens. Ssam: Real-time seismic spectral amplitude measurement on a pc and its application to volcano monitoring. *Bulletin of the Seismological Society of America*, 85(2):632–639, 1995.

[175] Adriana Romero, Carlo Gatta, and Gustau Camps-Valls. Unsupervised deep feature extraction for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3):1349–1362, 2016.

[176] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[177] CA Rowe, RC Aster, PR Kyle, JW Schlue, and RR Dibble. Broadband recording of strombolian explosions and associated very-long-period seismic signals on mount erebus volcano, ross island, antarctica. *Geophysical research letters*, 25(13):2297–2300, 1998.

[178] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533, 1986.

[179] K Sassa. Anomalous deflection of seismic rays in volcanic districts. *Mem. Coll. Science, Kyoto Imp. Univ. Ser. A*, 19:65–78, 1936.

[180] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[181] S. Scarpetta, F. Giudicepietro, EC. Ezin, S. Petrosino, E. Del Pezzo, M. Martini, , and M. Marinaro. Automatic classification of seismic signals at mt. vesuvius volcano, italy, using neural networks. *Bulletin of the Seismological Society of America*, 95(1):185–196, 2005.

[182] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[183] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[184] D. Seild, R. Schock, and M. Riuscetti. Volcanic tremor at etna, a model for hydraulic origin. *Bulletin Volcanologique*, 44(1):43–56, 1981.

[185] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

[186] Simon J Sheather and Michael C Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 683–690, 1991.

[187] D Shimozuru, N Osada, K Horigome, M Sawada, A Okada, M Shibano, S Matsumoto, K Sasaki, and Y Hosoga. Volcanic and seismic characteristics of izu islands, brief summary of the special project of prediction of volcanic eruptions. *Bull. Volc. Soc. of Japan*, 17:66–87, 1972.

[188] J. Simpson and TJ. McIntire. A recurrent neural network classifier for improved retrievals of areal extent of snow cover. *IEEE Transactions on Geoscience and Remote Sensing*, 39(10):2135–2147, 2001.

[189] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

[190] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

[191] RSJ Sparks and WP Aspinall. Volcanic activity: frontiers and challenges in forecasting, prediction and risk assessment. *The State of the Planet: Frontiers and Challenges in Geophysics*, pages 359–373, 2004.

[192] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[193] Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In *Advances in neural information processing systems*, pages 2377–2385, 2015.

[194] Susan Sturton and Jürgen Neuberg. The effects of a decompression on seismic parameter profiles in a gas-charged magma. *Journal of volcanology and geothermal research*, 128(1):187–199, 2003.

[195] David Sussillo and LF Abbott. Random walk initialization for training very deep feedforward networks. *arXiv preprint arXiv:1412.6558*, 2014.

[196] I. Sutskever, O. Vinyals, , and QV. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[197] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[198] t. Mikolov, S. Kombrink, L. Burget, J. Černockỳ, and S. Khudanpur. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5528–5531. IEEE, 2011.

[199] Jiexiong Tang, Chenwei Deng, Guang-Bin Huang, and Baojun Zhao. Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine. *IEEE Transactions on Geoscience and Remote Sensing*, 53(3):1174–1185, 2015.

[200] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[201] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[202] Nick Varley, Raúl Arámbula-Mendoza, Gabriel Reyes-Dávila, Richard Sanderson, and John Stevenson. Generation of vulcanian activity and long-period seismicity at volcán de colima, mexico. *Journal of Volcanology and Geothermal Research*, 198(1-2):45–56, 2010.

[203] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[204] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010.

[205] Barry Voight. A relation to describe rate-dependent material failure. *Science*, 243(4888):200–203, 1989.

[206] Peter Langdon Ward. *What Really Causes Global Warming?: Greenhouse Gases Or Ozone Depletion?* Morgan James Publishing, 2015.

[207] J. Wassermann. *IASPEI New manual of seismological observatory practice*, volume 1, chapter Chapter 13: Volcano seismology, page 42 pp. GeoForschungs-Zentrum Potsdam, 2002.

[208] Paul J Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[209] Jeremy West, Dan Ventura, and Sean Warnick. Spring research presentation: A theoretical foundation for inductive transfer. *Brigham Young University, College of Physical and Mathematical Sciences*, 1, 2007.

[210] David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[211] A. Yeguas, A. Garcia, J. Almendros, R. Abella, and J. Ibanez. Quantitative analysis of seismic wave propagation anomalies in azimuth and apparent slowness at deception island volcano (antarctica) using seismic arrays. *Geophysical Journal International*, 184(2):801–815, 2011.

[212] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[213] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, et al. The htk book. *Cambridge university engineering department*, 3:175, 2002.

[214] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[215] Haijiang Zhang, Clifford Thurber, and Charlotte Rowe. Automatic p-wave arrival detection and picking with multiscale wavelet analysis for single-component recordings. *Bulletin of the Seismological Society of America*, 93(5):1904–1912, 2003.

[216] Liangpei Zhang, Lefei Zhang, and Bo Du. Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 4(2):22–40, 2016.

[217] Vyacheslav M Zobin, Carlos J Navarro-Ochoa, and Gabriel A Reyes-Dávila. Seismic quantification of the explosions that destroyed the dome of volcán de colima, mexico, in july–august 2003. *Bulletin of volcanology*, 69(2):141–147, 2006.