Universidad de Granada

Departamento de Arquitectura y Tecnología de Computadores

ICA Incompleto Paralelo: Una nueva herramienta para el análisis de datos fMRI

Tesis Doctoral

Ingo Rudolf Keck

Granada, 2006

Editor: Editorial de la Universidad de Granada
Autor: Ingo Rudolf Keck
D.L.: Gr. 1754 - 2006
ISBN: 978-84-338-4086-8

Thanks

I would like to thank all the people that have helped me with this dissertation, especially

- My wife Salua Nassabay, who had to endure me all the time I was working on the dissertation. Without her constant guide this work would not have been possible.
- Prof. Dr. Carlos G. Puntonet for giving me the opportunity to work with him and for many useful advices and personal help.
- Prof. Dr. Elmar W. Lang for the possibility to work in his group and for an open ear every time I needed one.
- The members of the group of Prof. Lang for showing me the fine points of ICA.
- My parents for never giving up the hope in me and sustaining me all the time.
- and many more people I will name later.

Contents

List of Figures

Resumen

En la actualidad las comunicaciones adquieren cada día más importancia. En muchas situaciones del mundo real se trabajan con el envío y recepción de información, la cual generalmente, en el camino hacia su destino sufre diferentes distorsiones que corrompen la señal, resultando al final una mezcla de información que requiere ser procesada con el fin de obtener resultados correctos. Existe infinidad de campos donde el procesamiento de señales cumple diversas funciones fundamentales; estos campos comprende entre otros: la comunicación de datos, voz e imágenes, sismología, médica, acústica, sonar, instrumentación, robótica, etc.

El problema de la separación ciega de señales consiste en la recuperación de las señales originales a partir de las mezclas detectadas por sensores, conociendo tan sólo estas últimas. Estas mezclas de señales tiene lugar en el medio en que se propagan y en los sensores y como característica de este método es que a priori no se cuenta con ninguna información de las señales originales ni de la forma en que fueron mezcladas.

Un claro ejemplo que permite tener una idea general de este problema es el conocido efecto "Cocktail Party", el cual consiste en la habilidad que posee el ser humano de percibir y separar una voz de un fondo de ruido o de un conjunto de voces hablando de manera simultanea. Una vez realizada esta separación, las voces recuperadas o fuentes individuales, se les denomina "componentes independientes."

El presente trabajo doctoral tiene como objetivo la utilización de la herramienta del análisis de componentes independientes aplicado a señales del cerebro "fMRI". Para

entender bien como funciona fMRI vale la pena mencionar que el cerebro de las personas se encuentra dividido en dos hemisferios (derecho y izquierdo), y cada uno se divide en lóbulos los cuales desempeñan diferentes tareas relacionadas con el sentir, oír, ver, etc. Cada acción, pensamiento, sensación es producida por la actividad de las células localizadas en un área específica del cerebro. Cuando un grupo de células son alteradas por alguna razón comienzan a disparar, aumentando la corriente eléctrica, transmisión y el metabolismo. Esto también conlleva a que las necesiten más sangre porque necesitan más energía. La activación de un área del cerebro produce la recepción de más sangre ya que se está produciendo una dilatación de las venas sanguíneas. Por tanto, cualquier acción del cerebro causa un aumento de sangre en el punto de la corteza que rige la acción.

Las imágenes de resonancia magnética permiten ver la anatomía de los órganos internos del cuerpo. Las imágenes de resonancia magnética funcional son resultado de los valores de la intensidad de la señal que se produce con el aumento de sangre.

La presente memoria se encuentra estructura como sigue:

Capítulo 1: Introducción a la Separación Ciega de fuentes. En este capítulo se presenta la definición y formulación matemática de la técnica de Análisis de Componentes Independientes aplicado al problema de la Separación Ciega de Señales. En primer lugar se presentan los fundamentos estadísticos y su importancia en la separación de señales. A continuación se introduce en el tema de teoría de la información y los fundamentos básicos necesarios para la técnica ICA. Seguidamente se introducen las técnicas del Análisis de Componentes Principales (PCA) y el Análisis de Componentes Independientes (ICA) con sus correspondientes definiciones matemáticas, restricciones y algoritmos. Finalmente se realiza una comparación entre PCA e ICA.

Capítulo 2: ICA Reliability. En este capítulo se desarrollan fundamentos teóricos y

prácticos que permiten aplicaciones simples y directas de los conceptos que se aplican a lo largo del presente trabajo. En él realizo diferentes pruebas de confiabilidad, funcionamiento y comparación en sistemas reales del método análisis de componentes independientes, con énfasis en fMRI.

Capítulo 3: Aplicación de ICA: fMRI. En este capítulo se presenta una introducción a los Fundamentos de la Resonancia Magnética Funcional "fMRI", junto con las ventajas y desventajas que implican el utilizar análisis de componentes independientes para aplicaciones en investigación cerebral. Seguidamente se presentan resultados experimentales aplicado a los modelos linear; de forma que permiten demostrar la efectividad de ICA en datos de fMRI, para finalmente extraer las conclusiones tanto en aspectos positivos como negativos del método.

Capítulo 4: Incomplete ICA. En este capítulo se desarrolla una introducción teórica en el tema de Incomplete partiendo de su definición y formulación matemática. Seguidamente se introduce al tema de "clustering con Incomplete ICA", desarrollando en primera estancia ejemplos que demuestras la efectividad del método para trabajar con valores distintos a señales del cerebro. A continuación se aplica dicho método a fMRI y se describen las conclusiones obtenidas.

Capítulo 5: Incomplete ICA paralelo. En este capítulo se desarrolla una descripción aplicando el método de incomplete ICA clustering. Por medio de un ejemplo se demuestran las ganancias que resultan usando el método paralelo.

En la sección de "Conclusiones y Trabajos Futuro" se resumen las principales aportaciones que se han realizado en la materia con el desarrollo del presente trabajo doctoral. Igualmente se presentan trabajos futuros y líneas de investigación en los que vale la pena seguir investigando.

Summary

Nowadays communication gains more and more importance every day. In many real world situations one works in sending and receiving information. Generally, on the way to the receiver the signal suffers of different distortions that corrupt the signal, so that in the end one receives a mixture of information that has to be processed to find the correct results. There exists a huge number of fields where the processing of signals has fundamental importance; these are: data communication, speech and image processing, seismology, medicine, acoustics, sonar, instrumentation, robotic, etc.

The problem of blind source separation consist of the recovery of the original signals based on the mixtures detected by the sensors when only these mixtures are known. The main characteristic of this method is that no information about the original sources or how they were mixed together has to be known.

One widely known example of this concept that gives a general idea of this problem is the so called "Cocktail Party" effect. It shows that humans are able to hear and separate different voices from a background of noise and other voices that are speaking at the same time. After the separation has taken place these recovered voices or individual sources are called "independent components."

In this dissertation the main theme is to utilise the tool independent component analysis (ICA) on brain activity data sets gained from functional magnetic resonance imaging (fMRI). To understand how the brain works one has to mention that the human brain is separated in two hemispheres (right and left) and each of them divides itself

into parts that are related to different types of activities like sensing, smelling, seeing, etc. Every action, thought and sensation is produced by activation of the neural cells in the brain. When a group of neurones start firing for some reason, the current in the neural dendrites increases and transmission of information happens. This leads to an increase in blood flow as the cells need more energy. So, every increase of activation in the brain leads also to an increase of blood flow in the same region of the brain.

Magnetic resonance imaging allows us to see the anatomy of the internal organs of the human body. Functional MRI allows us to see the change in blood flow in the brain and thus the activity in the brain.

This dissertation has the following structure:

Chapter 1: Introduction in blind source separation. In this chapter the definitions and the mathematical formulation of the independent component analysis are presented. First the fundaments of mathematical statistics are presented and the importance of signal separation. Then the basics of information theory are presented so that all the necessary steps to introduce independent component analysis are complete. After this principal component analysis (PCA) and independent component analysis (ICA) are described together with their definitions, restrictions and algorithms. Finally, a comparison of PCA and ICA is given.

Chapter 2: ICA Reliability. In this chapter the fundamental theories and methods are presented that allow the simply and straightforward application of the methods given in this dissertation. The reliability and the stability testing methods for ICA are presented with the emphasis on fMRI.

Chapter 3: Application of ICA: fMRI. In this chapter an introduction to the fundamentals of fMRI is presented, together with the advantages and disadvantages of ICA applied to fMRI. The general linear model method and ICA are applied to experimental data and thus the positive and negative sides of ICA are demonstrated.

Chapter 4: Incomplete ICA. In this chapter first an introduction to the theory and the idea of incomplete ICA is given. Then the method of clustering with incomplete ICA is presented. First various examples are presented that show the efficiency of this method in clustering. Then the application of this method to fMRI data sets is presented.

Chapter 5: Parallel incomplete ICA. In this chapter a description how to apply the incomplete ICA clustering algorithm is given. An example shows the gains resulting from the parallel version of the method.

In the section "Conclusions and future work" a summary of the work done in this dissertation is give. At the same time future work and the fields of research are given that should be taken in account for future research.

1 Introducción a la Separación Ciega de Fuentes

En este capítulo se dará una introducción al concepto de la *separación ciega de fuentes* (BSS) con énfasis en el *análisis de componentes independientes* (ICA). Se comienza primero dando una descripción de BSS, seguida de una corta introducción de los fundamentos matemáticos necesarios en estadística y teoría de la información para entender el concepto del análisis componentes independientes.

Con la introducción no pretendo dar pruebas de teoremas; sino por el contrario, utilizar los ejemplos para ayudar en la comprensión del mismo.

Las ideas de la *separación ciega de fuentes (BSS)* se remontan a muchos años atrás, con los artículos publicados al rededor de 1966 de C.R. Rao [36,37]. En 1980 Benveniste y col. publicaron en [9] un interesante artículo sobre la deconvolución ciega y muestran un método que se asemeja a la forma como se estima ICA con deferentes multicanales instantáneos. La expresión *análisis de componentes independientes* fue forjada alrededor del año 1987 por Herault y Jutten, quienes se basaron en las semejanzas del análisis de componentes principales [23].

En el modelo de la separación ciega de fuentes los datos $\vec{x}$ se obtienen de una mezcla de las fuentes $\vec{s}$ (ver figura 1.1). Este problema es más amplio y necesita más restricciones para poder llegar a una correcta solución. Estas restricciones resultan de la manera en como el sistemas de datos se genera para mantener un modelo válido. Como ejemplo se

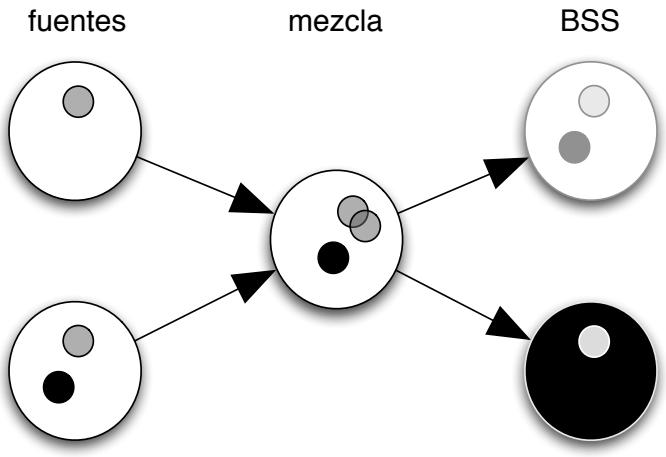


Figure 1.1: Esta figura muestra el concepto de BSS: el sistema de datos se interpreta como mezcla de fuentes subyacentes. BSS intenta reconstruir las fuentes de las mezclas originales - esto es a menudo solamente posible hasta la permutación y el escalamiento.

tiene que una restricción común es aceptar en el problema solamente modelos lineares BSS, esto significa,

$$\vec{x} = \mathbf{A}\vec{s} \tag{1.1}$$

donde $\mathbf{A}$ es a $m \times n$-matrix.

Sin embargo, definir correctamente BSS y sus herramientas se requiere de matemática en teoría estadística y de la información.

1.1 Fundamentos Estadísticos

En esta sección se da una corta introducción a los fundamentos estadísticos y de la teoría de la información necesarios y útiles para entender la separación ciega de fuentes (BSS) y el análisis componentes independientes (ICA).

Primero es necesario definir las herramientas matemáticas básicas. Estas se pueden encontrar en muchos libros de probabilidad y de estadística, como por ejemplo [4,14,35]. Por lo tanto en este capítulo no se demostraran las pruebas de los teoremas con el fin de hacer más comprensible el tema.

1.1.1 Espacio de Probabilidad

Definition 1.1.1 *Sea el conjunto Ω no vacío, es decir, $\neq \emptyset$. Un sistema de subconjuntos $\mathcal{A} \subset P(\Omega) = \{A \subset \Omega\}$ esta llamado* σ-algebra *en Ω, si*

- $\Omega \in \mathcal{A}$,
- $A \in \mathcal{A} \Rightarrow \Omega \backslash A \in \mathcal{A}$,
- $A_i \in \mathcal{A} \Rightarrow \bigcup_{i=1}^{\infty} A_i \in \mathcal{A}$. *para todo* $i \in \mathbb{N}$

Si existe el conjunto de eventos Ω y una σ-algebra, todos los $A \in \mathcal{A}$ son llamados *eventos*.

Definition 1.1.2 *Si $\mathcal{A}$ es σ-algebra en Ω, entonces $P : \mathcal{A} \mapsto \mathbb{R}$ se llama* medida de la probabilidad*, si*

- $P(A) \geq 0$ *para todo* $A \in \mathcal{A}$
- $P(\Omega) = 1$ *y*
- $P(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$ *para parejas disjuntas (mutually disjoint)* A_i.

Estos tres requisitos son también conocidos como *axiomas de Kolmogorov.* $(\Omega, \mathcal{A}, P)$ se conocen como el *espacio de la probabilidad* y $P(A)$ es la probabilidad del evento A. Por la definición se tiene

$$0 \leq P(A) \leq 1$$

Definition 1.1.3 *Una transformación* $X : \Omega \mapsto \mathbb{R}$ *es llamada* variable aleatoria *en el espacio de la probabilidad* $(\Omega, \mathcal{A}, P)$*, si para cada subconjunto* $A_i = \{\omega : X(\omega) \leq a\}$ *con* $a \in \mathbb{R}$ *también se tiene* $A_i \in \mathcal{A}$*.*

Definition 1.1.4 *La* función de densidad de probabilidad *(pdf) de* x *esta definida por*

$$\begin{aligned} F_x : \quad \mathbb{R} &\to \quad [0,1] \\ x_0 &\mapsto \quad F_x(x_0) = P(x \leq x_0) := P(x^{-1}(-\infty, x_0]) \end{aligned}$$

Es interesante observar que F_x es constante y continuo el aumento en el lado derecho y

$$\begin{aligned} F(-\infty) &= 0 \\ F(\infty) &= 1. \end{aligned}$$

A continuación se hará referencia sobre las variables aleatorias continuas. Estas resultan importantes en la manera que

$$p_x : \mathbb{R} \to \mathbb{R}, x_0 \mapsto p_x(x_0) := \left. \frac{dF_x(x)}{dx} \right|_{x=x_0}$$

existe y es continuo, $p_x(x)$ se llama *densidad* o *función de densidad de la probabilidad*

(pdf) de x. Un ejemplo importante es el pdf gaussina:

$$p(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \tag{1.2}$$

m es el centro de la distribución y σ es la raíz cuadrada de su variación.

1.1.2 Distribuciones Multidimensionales

La definición que hasta ahora se había dado unidimensional de variables aleatorias y de sus distribuciones, puede ser ampliada a múltiples dimensiones.

Definition 1.1.5 *$\vec{x} = (x, \ldots, x_n)^T$ se llama* vector aleatorio*, si*

$$\vec{x}^{-1}((a, b] \times \ldots \times (a_n, b_n]) \in \mathcal{A}.$$

Los vectores aleatorios también tienen distribuciones y densidades de probabilidad.

Definition 1.1.6 *La distribución de un vector aleatorio $\vec{x}$ se define como*

$$F_{\vec{X}} = P(\vec{x}^{-1}((-\infty, x_1 \ \times \ldots \times (-\infty.x_n]))$$

y la densidad de $\vec{x}$ como

$$p_{\vec{x}}(\vec{x}_0) := \left.\frac{\partial^n}{\partial x_1 \ldots \partial x_n} F_{\vec{x}}(\vec{x})\right|_{\vec{x}=\vec{x}_0}$$

La integral sobre la probabilidad de densidad en una dimensión tiene que ser igual a 1:

$$\int_{\mathbb{R}^n} p_{\vec{x}} = 1 \tag{1.3}$$

La matriz de componentes independientes utiliza operadores de variables aleatorios. El siguiente teorema resulta especialmente útil en estos casos:

Theorem 1.1.1 *(Transformación de densidades)*
Si $\mathbf{A}$ *esta en el Lie-group de valores invertibles* $n \times n$*-matrices Gl(n) (también llamado* grupo linear general*), entonces la densidad de probabilidad de* $\mathbf{A}\vec{x}$ *puede ser calculada como:*

$$P_{\mathbf{A}\vec{x}}(\mathbf{A}\vec{x}) = \frac{1}{|det\mathbf{A}|} p_{\vec{x}}(\vec{x}).$$

Cross-Correlación y Esperanzas

El *momento de primer orden* de un vector aleatorio $\vec{x}$ es *el valor de la esperanza*, también llamado *media*:

Definition 1.1.7

$$E(\vec{x}) := \int_{\mathbb{R}^n} \vec{x} p_{\vec{x}}(\vec{x}) d\vec{x}$$

se llama esperanza *de* $\vec{x}$ *o* media $\vec{m}_{\vec{x}}$.

Es también posible calcular la media del producto entre una matriz $\mathbf{A}$ y un vector aleatorio $\vec{x}$:

$$E(\mathbf{A}\vec{x}) = \mathbf{A}E(\vec{x}) \tag{1.4}$$

$$E(\vec{x}\mathbf{B}) = E(\vec{x})\mathbf{B} \tag{1.5}$$

La forma más simple para comparar los vectores es calculando su producto escalar. Para los vectores aleatorios esto da también información sobre sus distribuciones. Si ambos vectores son los mismos entonces esto se llama (auto-)correlación:

Definition 1.1.8 *El valor de la esperanza del producto de un vector aleatorio $\vec{x}$ con su traspuesta*

$$\begin{aligned} R_{\vec{x}} &= E(\vec{x}\vec{x}^T) \\ &= E\begin{pmatrix} x_1x_1 & \cdots & x_1x_n \\ \vdots & \ddots & \vdots \\ x_nx_1 & \cdots & x_nx_n \end{pmatrix} = \begin{pmatrix} E(x_1x_1) & \cdots & E(x_1x_n) \\ \vdots & \ddots & \vdots \\ E(x_nx_1) & \cdots & E(x_nx_n) \end{pmatrix} \end{aligned}$$

se llama (auto-)correlación *de $\vec{x}$.*

$R_{\vec{x}}$ se define simétrico y positivo ($\vec{a}^T R_{\vec{x}} \vec{a} \geq 0$). La auto-correlación proporciona un momento de segundo orden de $\vec{x}$:

Definition 1.1.9 *$R_{\vec{x}} = (r_{ij})_{ij}$ se denominan los momentos de segundo orden de $\vec{x}$.*

Si se elimina la media de la correlación se obtiene la covarianza:
Definition 1.1.10

$$C_{\vec{x}} = R_{\vec{x}-\vec{m}\vec{x}} = E\left[(\vec{x} - \vec{m}_{\vec{x}})(\vec{x} - \vec{m}_{\vec{x}})^T\right]$$

se denominan (auto-)covarianza *o* momentos de segundo orden centrales *de $\vec{x}$.*

En el caso uni-dimensional este momento se llama varianza. Es la desviación estándar σ elevado a 2:

Definition 1.1.11 *$Var(x) = c_x = E((x - m_x)^2) = \sigma_x^2$ se llama* varianza. *σ es la desviación estándar de la distribución de $\vec{x}$*

Usando los dos vectores aleatorios de la definición 1.1.8 se obtiene la definición de crosscorrelación:

Definition 1.1.12 *$R_{\vec{x}\vec{y}} = E(\vec{x}\vec{y}^T)$ se llama* crosscorrelación *de dos vectores aleatorios $\vec{x}$ y $\vec{y}$.*

Mientras que la autocorrelación de un vector aleatorio proporciona la información sobre semejanzas en la distribución del vector aleatorio consigo mismo, la crosscorrelación compara las distribuciones de los dos vectores aleatorios. Normalmente resulta útil eliminar la media de los vectores aleatorios antes de realizar el calculo de la crosscorrelación. El resultado de este procedimiento se denomina crosscovarianza:

Definition 1.1.13 $C_{\vec{x},\vec{y}} = E((\vec{x}-\vec{m}_{\vec{x}})(\vec{y}-\vec{m}_{\vec{y}})^T)$ *se llama* crosscovarianza *de los vectores aleatorios* $\vec{x}$ *y* $\vec{y}$. $\vec{m}_{\vec{x}}$ *y* $\vec{m}_{\vec{y}}$ *son sus respectivas medias.*

Una vez mencionadas estas definiciones podemos dar otro paso hacia el tema de la separación ciega de fuentes.

1.1.3 Decorrelación e Independencia Estadística

Como se ha visto existe siempre una correlación entre dos vectores aleatorios. Sin embargo, si esta correlación es igual a cero, estos dos vectores no se correlacionan. Por consiguiente resulta interesante analizar que ocurre en este caso.

Definition 1.1.14 $\vec{x}$ *y* $\vec{y}$ *se denominan* decorrelados, *si* $C_{\vec{x}\vec{y}} = 0$. $\vec{x}$ *es* decorrelacionado mutuamente *si* $C_{\vec{x}}$ *es diagonal (es decir* $E((x_i - m_{x_i})(x_j - m_{x_j}) = 0$ *para todo* $i \neq j$*).*

Un caso especial es el vector aleatorio decorrelacionado y centrado, porque es un paso común en el pre-procesamiento de estadísticos de alto orden:

Definition 1.1.15 *Si* $\vec{m}_{\vec{x}} = 0$ *y* $C_{\vec{x}}$ *es la identidad* $\mathbb{I}$*, entonces* $\vec{x}$ *se conoce como* blanqueado.

Resulta interesante observar que $\vec{x}$ es blanqueado solamente cuando $\mathbf{A}\vec{x}$ es blanqueado para $\mathbf{A} \in O(n) = \mathbf{A}|\mathbf{A}\mathbf{A}^T = \mathbb{I}$. Esto puede ser fácilmente visto en

$$C_{\mathbf{A}\vec{x}} = E(\mathbf{A}\vec{x}\vec{x}^T\mathbf{A}^T) = \mathbf{A}E(\vec{x}\vec{x}^T)\mathbf{A}^T = \mathbf{A}\mathbf{A}^T = \mathbb{I}.$$

Para conseguir resultados superiores a los estadísticos de segundo orden, tenemos que utilizar las funciones de distribución de la probabilidad de vectores aleatorios.

Definition 1.1.16 $\vec{x}$ *se denomina* independencia mutua *si*

$$p_{\vec{x}}(x_1, \ldots, x_n) = p_{x_1}(x_1) \cdots p_{x_n}(x_n).$$

$p_{x_i}(x_i)$ *se llama* densidad marginal.

De manera similar, para dos vectores aleatorios:

Definition 1.1.17 *Dos vectores aleatorios* $\vec{x}$, $\vec{y}$ *son* estadísticamente independientes *si*

$$p(\vec{x}, \vec{y}) = p(\vec{x})p(\vec{y}).$$

Para demostrar el significado de estas definiciones podemos mirar algunos ejemplos:

- Sea $\vec{x}$ un vector gaussiano n-dimensional. La función de la densidad de probabilidad es:
$$p_{\vec{x}}(\vec{x}) = \frac{1}{(2\pi)^{\frac{n}{2}}(\det C_{\vec{x}})^{\frac{n}{2}}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{m}_{\vec{x}})^T C_{\vec{x}}^{-1}(\vec{x} - \vec{m}_{\vec{x}})\right)$$
con la media $\vec{m}_{\vec{x}}$ y la covarianza $C_{\vec{x}}$ de $\vec{x}$.

- Sea $\vec{x}$ un vector gaussiano bi-dimensional con $C_{\vec{x}} = \mathbb{I}$, $\vec{m}_{\vec{x}} = 0$ (blanqueado). La función de la densidad de probabilidad es:
$$p_{\vec{x}}(x_1, x_2) = C \exp\left(-\frac{1}{2}(x_1^2 + x_2^2)\right) = C \exp\left(-\frac{1}{2}x_1^2\right) \exp\left(-\frac{1}{2}x_2^2\right).$$
Obviamente $\vec{x}$ es mutuamente independiente.

- Sea $\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \vec{y} = \mathbf{A}\vec{x}, \mathbf{A}^{-1} = \frac{1}{2}\mathbf{A}$. $\vec{x}$ es un gaussiano blanqueado, bi-

dimensional. Entonces tenemos:

$$\begin{aligned} p_{\vec{y}}(\vec{y}) &= \frac{1}{2} p_{\vec{x}}(\mathbf{A}^{-1}\vec{y}) \\ &= \frac{1}{2} C \exp\left(-\frac{1}{2}\frac{1}{4}(y_1^2 + y_2^2)^2 + (y_1 - y_2)^2\right) \\ &= \frac{1}{2} C \exp\left(-\frac{1}{4} y_1^2\right) \exp\left(-\frac{1}{4} y_2^2\right), \end{aligned}$$

así $\vec{y} = \mathbf{A}\vec{x}$ se obtiene resultados mutuamente independientes.

El anterior ejemplo puede formularse de una manera más general:

Theorem 1.1.2 *Si $\vec{x}$ es gaussiana (tiene una distribución gaussiana) entonces $\vec{y} = A\vec{x}$ también es gaussino con la covarianza $C_{\vec{y}} = AC_{\vec{x}}A^T$, $\vec{m}_{\vec{y}} = A\vec{m}_{\vec{x}}$. Si $\vec{x}$ es gaussiano y esta decorrelacionado, entonces es también mutuamente independiente.*

Regresando de nuevo al caso general, podemos también definir las probabilidades para las variables aleatorias relacionadas:

Definition 1.1.18 *La densidad de probabilidad restrictiva de $\vec{x}$ y dado $\vec{y}_0$ se define:*

$$p_{\vec{x},\vec{y}}(\vec{x}|\vec{y}) = \frac{p_{\vec{x},\vec{y}}(\vec{x}, \vec{y}_0)}{p_{\vec{y}}(\vec{y}_0)}$$

Si $\vec{x}$ y $\vec{y}$ son independientes entonces $p_{\vec{x}|\vec{y}} = p_{\vec{x}}$ as $p_{\vec{x},\vec{y}} = p_{\vec{x}} p_{\vec{y}}$.

La generalización de esta regla se denomina *Bayes' rule*:

Theorem 1.1.3 *Las probabilidades restrictas siguen las relaciones:*

$$p_{\vec{x},\vec{y}}(\vec{x}_0|\vec{y}_0) = p_{\vec{x}|\vec{y}}(\vec{y}_0|\vec{x}_0) p_{\vec{x}}(\vec{x}_0) = p_{\vec{y}|\vec{x}}(\vec{x}_0|\vec{y}_0) p_{\vec{y}}(\vec{y}_0)$$

y

$$p_{\vec{y}|\vec{x}}(\vec{y}_0|\vec{x}_0) = \frac{p_{\vec{x}|\vec{y}}(\vec{x}_0|\vec{y}_0) p_{\vec{y}}(\vec{y}_0)}{p_{\vec{x}}(\vec{x}_0)}$$

Las gaussianas tiene una interesante característica: las probabilidades restrictivas de las mismas son también gaussianas.

Momentos de Alto Orden

Además de la covarianza y la correlación, existen más momentos estadísticos. La definición de éstos se conoce como *momentos de alto orden*:

Definition 1.1.19

$$\alpha_j := E(\vec{x}^j) := \int \vec{x}^j p_x d\vec{x}$$

se llama momentos de jth orden *de* $\vec{x}$.

Definition 1.1.20

$$\mu_j := E((x\alpha_1)^j)$$

se llama momento central de jth orden *de* $\vec{x}$.

Los primeros cuatro momentos centrales tienen nombres especiales:

- $\alpha_1 = \mu$ es el valor de la esperanza $\vec{x}$.
- μ_2 es la *varianza* de $\vec{x}$.
- μ_3 se conoce como *skewness* de $\vec{x}$.
- Usando μ_4 la *kurtosis* de $\vec{x}$ se define:

Definition 1.1.21 *La* kurtosis *de un vector aleatorio* $\vec{x}$ *se define como:*

$$kurt(x) := \mu_4 - 3\mu_2^2$$

con el 4th momento central μ_4 *y el 2do momento central* μ_2 *de* $\vec{x}$.

A continuación se comentarán las características de la kurtosis:

- kurt(βx) =kurt(x)

- Si x y y son estadísticamente independientes, entonces tenemos:

$$\text{kurt}(x + y) = \text{kurt}(x) + \text{kurt}(y)$$

1.1.4 Teoría de Estimadores

Comenzando con T experimentos o muestras $\vec{x} = (x(1), \ldots, x(T))$ deseamos estimar los parametros $\Theta_1, \ldots, \Theta_n$. Esta clase de proyección $\hat{\Theta} : \mathbb{R}^T \to \mathbb{R}^n$ se conoce como *estimador*. A continuación presento dos ejemplos importantes para esta teoría:

- El estimador del medio es la media de la muestra:

$$\hat{\mu} = \frac{1}{T} \sum_{i=1}^{T} x(i).$$

- El estimador de la varianza es la varianza de la muestra:

$$\sigma^2 = \frac{1}{T-1} \sum_{i=1}^{T} (x(j) - \hat{\mu})^2.$$

Tenemos a menudo una familia $\hat{\Theta}^{(T)}$ de estimadores. Esta Familia se denomina *online* cuando los estimadores pueden ser calculado con frecuencia:

$$\hat{\Theta}^{(j+1)} = \vec{h}(x(j+1), \hat{\Theta}(j)),$$

Pero, así mismo existe el caso contrario donde éstos no puede ser calculado con frecuencia. Este caso se denomina *batch*. Como ejemplo podemos tomas los estimadores medios:

$$\hat{\mu}^{(j+1)} = \frac{1}{j+1}(j\hat{\mu}^{(j+1)} + x(j+1))$$

Este proceso de valoración, lleva consigo un error de valoración, el cual se define de la siguiente manera:

Definition 1.1.22 *Sea* $\Theta(x) \in \mathbb{R}^n$ *un valor relacionado con la variable aleatoria* x *la cual puede ser estimada y* $\hat{\Theta}$ *el estimador. Entonces*

$$\tilde{\Theta}(x, x(1), \ldots, x(T)) := \Theta(x) - \hat{\Theta}(x(a), \ldots, x(T))$$

Se denomina error de estimación *la relación entre las variables* x *y* $x(i) \in \mathbb{R}$.

Naturalmente, para un buen estimador $\tilde{\Theta}$ debe de ser lo más cercano posible a 0, si $x(1), \ldots, x(T)$ son muestras de x.

Otra característica que a menudo se espera obtener de un estimador, es que él converja al un valor verdadero si el tamaño de la muestra converge al infinito:

Definition 1.1.23 *Si* $x_1, \ldots, x_t$ *son las iid-muestras de variables aleatorias con distribuciones como* x*, entonces* $\hat{\Theta}$ *se llama* estimador imparcial *de* Θ*, si*

$$E(\tilde{\Theta}) = 0,$$

es decir,

$$E(\Theta(x)) = E(\hat{\Theta}(x_1, \ldots, x_T))$$

Nuevamente se tiene, que la media de la muestra resulta un buen ejemplo ya que es un estimador imparcial del medio de la distribución μ:

$$E(\hat{\mu}) = \frac{1}{T}\sum_{j=1}^{T} E(x_j) = \frac{1}{T}TE(x) = E(x) = \mu$$

Teorema del Límite Central

La suma de variables de distribución uniforme independientes, tienen una característica interesante: convergen en una distribución gaussiana . Esto se describe con el *teorema del límite central*:

Theorem 1.1.4 *Sea $z_1, z_2, \ldots$ un sistema de datos con distribución independiente (iid) de L^2-variables aleatorias. Sea $x_k := \sum_{i=1}^{k} z_i$; $y_k := \frac{x_k - m_{x_k}}{\sigma_{x_k}}$ la norma de x_k. Entonces puede decirse:*
Cada $k \to \infty$, y_k converge en una gaussina normalizada.

1.1.5 Máxima Probabilidad

El método de la *máxima probabilidad* (ML) permite encontrar un estimador óptimo $\hat{\Theta}_{ML}$ dentro de una familia de estimadores Θ. El estimador de toda la probabilidad $\hat{\Theta}_{ML}$ para la muestra dada $x(1), \ldots, X(T)$ de una variable aleatoria x satisface la siguiente ecuación:

$$p(\vec{x}|\hat{\Theta}_{ML}) = p_{\vec{x}|\ \Theta}(\vec{x}|\hat{\Theta}_{ML}(\vec{x})) \text{ es máxima.} \tag{1.6}$$

De ésta ecuación se puede derivar:

$$\frac{\partial}{\partial \Theta_i} \ln p(\vec{x}|\Theta)|_{\Theta=\hat{\Theta}_{ML}} = 0, \text{ for } i = 1, \ldots, n. \tag{1.7}$$

Debido a $p(\vec{x}|\Theta) = \prod_{j=1}^{T} p(x(j)|\Theta)$ el resultado en

$$\frac{\partial}{\partial \Theta_i} \sum_{j=1}^{T} \ln p(x(j)|\Theta)|_{\Theta=\hat{\Theta}_{ML}} = 0. \tag{1.8}$$

Ejemplo: Sean $x(1), \ldots, x(T)$ muestras de una distribución gaussina, donde los parámetros μ y σ^2 tienen que ser estimados:

$$p(\vec{x}|\mu, \sigma^2) = (2\pi\sigma^2)^{-T/2} \exp(-\frac{1}{2\sigma^2} \sum_{j=1}^{T} (x(j) - \mu)^2).$$

Entonces obtenemos la siguiente ecuación para el registro-probabilidad:

$$\ln p(\vec{x}|\mu\sigma^2) = -\frac{T}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{j=1}^{T} (x(j) - \mu)^2$$

La máxima probabilidad esta dada por:

$$\frac{\partial}{\partial \mu} \ln p(\vec{x}|\hat{\mu}_{ML}, \sigma^2_{ML}) = \frac{1}{\sigma^2_{ML}} \sum_{j=1}^{T} (x(j) - \hat{\mu}_{ML} \stackrel{!}{=} 0$$

Con lo que se obtiene:

$$\hat{\mu}_{ML} = \frac{1}{T} \sum_{j=1}^{T} x(j).$$

Aparte de ese caso, se tiene también:

$$\frac{\partial}{\partial \sigma^2} = -\frac{T}{2\hat{\sigma}^2} + \frac{1}{2\hat{\sigma}^4_{ML}} \sum_{j=1}^{T} (x(j) - \hat{\mu})^2 \stackrel{!}{=} 0$$

y

$$\hat{\sigma}^2_{ML} = \frac{1}{T} \sum_{j=1}^{T} (x() - \hat{\mu}_{ML})^2.$$

Estos estimadores no son insesgados, sino que son estimadores sesgados.

1.2 Teoría de la Información

A continuación describo los fundamentos básicos de la teoría de la información necesarios para entender ICA.

1.2.1 Entropía

La *entropía* es una medida de información para una variable aleatoria. Esta se define de la siguiente manera:

Definition 1.2.1 *La* entropía (diferencial) *de una variable aleatoria $\vec{x}$ con densidad de probabilidad $p_{\vec{x}}$ es:*

$$H(\vec{x}) = -E_{\vec{x}}(\log p_{\vec{x}}) = -\int_{\mathbb{R}^n} p_{\vec{x}}(\vec{x}') \log p_{\vec{x}}(\vec{x}') d\vec{x}'$$

Demostrando esta idea con un ejemplo de una densidad uniforme se tiene:

$$P_x(\vec{x}') := \begin{cases} \frac{1}{a} \text{ if } \vec{x}' \in [0, a] \\ 0 \text{ para todos los otros casos} \end{cases}$$

La entropía diferencial de x es

$$H(x) = -\int_0^a \frac{1}{a} \log \frac{1}{a} d\vec{x}' = -\log \frac{1}{a} = \log a$$

Theorem 1.2.1 *Si* $\mathbf{A}$ *es inversible,*

$$H(\mathbf{A}x) = H(x) + log(\det(\mathbf{A})).$$

Theorem 1.2.2 $H(x_{gauss}) \geq H(x)$ *para todo* x *con media* 0 *y varianza* 1.

1.2.2 Información Mutua

Definition 1.2.2 *La* información mutua (MI) *de* $\vec{x}$ *se define como:*

$$I(\vec{x}) = \sum_{i=1}^{n} H(x_i) - H(\vec{x})$$

con la entropía marginal $H(x_i)$ *y la entropía común* $H(\vec{x})$*.*

Theorem 1.2.3 *La información mutua es siempre una cantidad positiva:* $I(\vec{x}) \geq 0$*. Es igual a* 0 *solo si* $\vec{x}$ *es mutuamente independiente.*

Theorem 1.2.4 *La información mutua de un vector aleatorio es independiente del escalamiento y de la permutación:*

$$I(\mathbf{LP}\vec{x}) = I(\vec{x}),$$

donde la matriz de escalamiento es $\mathbf{L}$ *y la matriz de permutación* $\mathbf{P}$

Una matriz de escalamiento $\mathbf{L}$ es simplemente una matriz diagonal, es decir $l_{ij} = 0$ para $i \neq j$. Una matriz de permutación $\mathbf{P}$ es una matriz que tiene el valor 1 exactamente una vez en cada columna y fila.

1.2.3 Negentropía

Definition 1.2.3 *La* negentropía *de* $\vec{x}$ *se define como:*

$$J(\vec{x}) := H(\vec{x}_{gauss,Cov(\vec{x})}) - H(\vec{x}),$$

donde $\vec{x}_{gauss,Cov(\vec{x})}$ *es una gaussiana con covarianza* $Cov(\vec{x})$ *.*

Es interesante observar

$$H(\vec{x}_{\text{gauss},\Sigma}) = \frac{1}{2}\log(\det\Sigma) + \frac{n}{2}(1+\log 2\pi)$$

Theorem 1.2.5 *Si* $\mathbf{A}$ *es inversible, entonces se tiene:* $J(\mathbf{A}\vec{x}) = J(\vec{x})$.

Generalmente la negentropía debe de ser aproximada, ya que resulta difícil de calcular. Unos buenos resultados pueden obtenerse cuando se usa la aproximación polinómica de *Gram-Charlier* y de *Edgeworth*. Estos dos autores utilizan *Chebishev-Hermit-polynoms*:

$$\frac{\partial^i \phi(x)}{\partial x'^i} = (-1)^i H_i(x')\phi(x') \tag{1.9}$$

Con lo anteriormente citado puede construirse un sistema ortogonal como el siguiente:

$$\delta_{ij} = \int \phi(x')H_i(x')H_j(x')dx' \tag{1.10}$$

$$\phi(x') = \frac{1}{\sqrt{2\pi}}\exp(-\frac{x'^2}{2}) \tag{1.11}$$

La distribución de la probabilidad se puede aproximar como:

$$\hat{p}(x)|_{x'} = \phi(x')\left[1 + \kappa_3(x)\frac{H_3(x')}{3!} + \kappa_4(x)\frac{H_4(x')}{4!} + \ldots\right] \tag{1.12}$$

con

$$\kappa_e(x) = E(x^3)$$

$$\kappa_4(x) = E(x^4) - 3 = \text{kurt}(x)$$

Comúnmente esto resulta ser una mala aproximación. Con lo que se puede recurrir a la siguiente ecuación para la entropía:

$$H(x) \cong -\int \hat{p}_x(x') \log \hat{p}_x(x') dx' \cong -\int \phi(x') \log \phi(x') dx' - \frac{\kappa_3^2(x)}{2 * 3!} - \frac{\kappa_4^2(x)}{2 * 4!} \quad (1.13)$$

Con este resultado y utilizando la simetría, se puede obtener la siguiente aproximación para la negentropía:

$$J(x) \approx \frac{1}{12}\kappa_3^2(x) + \frac{1}{48}\kappa_4^2(x) = \frac{1}{48}\kappa_4^2(x) \quad (1.14)$$

Ahora disponemos de todas las herramientas para comenzar con las diferentes técnicas de la separación ciega de fuentes. En primer lugar describiré el análisis de componente principales.

1.3 Análisis de Componentes Principales

El *análisis de componentes principales (PCA)* es una transformación que maximiza la varianza. En física es conocida como *transformación de los ejes principales* y tiene dos características importantes:

- El eje que corresponde al valor propio más grande, está en la dirección de la máxima variación del sistema de datos.

- Los ejes son ortogonales.

Las características estadísticas de PCA pueden ser definidas como la transformación de un vector aleatorio; de modo que la matriz de covarianza del vector resultante sea la matriz de identidad $\mathbb{I}$. PCA también puede utilizarse para casos en donde se necesita una reducción del ruido; ya que con este método se conserva la parte principal de la varianza. Por lo tanto los vectores propios con los valores propios más pequeños se fijan en cero.

Existen muchos métodos que permiten calcular las componentes principales. Matemáticamente esto corresponde a calcular los vectores propios (eigenvectors) y los valores propios (eigenvalues) de la matriz de covarianza del vector aleatorio:

$$\mathbf{D} = \mathbf{V}^T \mathrm{cov}(\vec{x})\mathbf{V} \tag{1.15}$$

donde $\mathbf{V}$ es la matriz con los vectores propios de la matriz de covarianza de $\vec{x}$ como columnas y $\mathbf{D}$ es una matriz diagonal con los valores propios correspondientes en la diagonal. La ecuación para la matriz de PCA $\mathbf{W}$ es:

$$\mathbf{W} = \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T \tag{1.16}$$

La prueba para demostrar esto se puede obtener de una manera muy simple:

$$\begin{aligned} \text{cov}(\mathbf{W}\vec{x}) &= E(\mathbf{W}\vec{x}\vec{x}^T\mathbf{W}^T) = \mathbf{W}\text{cov}(\vec{x})\mathbf{W} \\ &= \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T\text{cov}(\vec{x})\mathbf{V}\mathbf{D}^{-\frac{1}{2}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{D}\mathbf{D}^{-\frac{1}{2}} \\ &= \mathbb{I} \end{aligned}$$

Otra posibilidad, es la optimización con la siguiente función de coste:

$$J(\vec{\omega}) =< (\vec{x}^T\frac{\vec{\omega}}{||\vec{\omega}||})^2 >= \frac{\vec{\omega}^T < \vec{x}\vec{x}^T > \vec{\omega}}{||\vec{\omega}||^2}$$

donde $\vec{x}$ es el vector de los datos ya centralizados y $\vec{\omega}$ los nuevos ejes del sistema de coordenadas. La reducción al mínimo de esta función de coste significa:

$$\vec{\nabla}J(\vec{\omega}) \stackrel{!}{=} 0 \tag{1.17}$$

Con lo que se obtiene:

$$\frac{< \vec{x}\vec{x}^T > \vec{\omega}||\vec{\omega}||^2 - (\vec{\omega}^T < \vec{x}\vec{x}^T > \vec{\omega})\vec{\omega}}{||\vec{\omega}||^4} \stackrel{!}{=} 0 \tag{1.18}$$

y

$$< \vec{x}\vec{x}^T > \vec{\omega} = \frac{(\vec{\omega}^T < \vec{x}\vec{x}^T > \vec{\omega})}{||\vec{\omega}||^2}\vec{\omega} = \lambda\vec{\omega} \tag{1.19}$$

con

$$\lambda = \frac{(\vec{\omega}^T < \vec{x}\vec{x}^T > \vec{\omega})}{||\vec{\omega}||^2}.$$

1.3.1 PCA con Redes Neuronales

Es también posible utilizar una red neuronal para PCA: Sea $\vec{x}'$ el vector de entrada, $\vec{w}$ los pesos y $\vec{v}$ el vector de salida. La regla del aprendizaje Hebbiano es:

$$\vec{v} = \vec{w}^T \vec{x}' \tag{1.20}$$

con

$$\Delta w_i = \eta v_i x_i' \tag{1.21}$$

Es importante conocer que resulta necesario la normalización de los pesos. Una forma para hacer esto es la normalización propuesta por Oja con paso de tiempo k:

$$\vec{w}^{k+1} = \frac{\vec{w}^k + \eta \vec{v}^k \vec{x}'^k}{||\vec{w}^k + \eta \vec{v}^k \vec{x}'^k||} \approx w^4 + \eta v^4 (x'^4 - v^4 w^4) + O(\eta^2) \tag{1.22}$$

De lo anterior se obtiene:

$$\Delta w_i = \eta v_i x_i - \eta v^2 w_i \tag{1.23}$$

$\vec{w}$ forma entonces un vector de valores propios $< \vec{x}\vec{x}^T >$ con $\lambda_{\max}$.

Una clara desventaja de trabajar con redes neuronales es la lentitud de las mismas; pero igualmente podemos obtener diferentes ventajas de las mismas, entre las que vale la pena resaltar la posibilidad de trabajar en línea (online) y no de manera estacionaria, la capacidad que tienen de aprender de la experiencia y de abstraer características esenciales a partir de entradas que representan alguna información irrelevante.

Análisis del Subespacio

Usando la *regla del subespacio* por Oja [32, 33] es posible encontrar las combinaciones lineares de las componentes principales, es decir, los subspacios:

$$\Delta w_{ij} = \eta v_i (x_i - \sum_{k=1}^{M} v_k w_{k_j}) \tag{1.24}$$

Usando la *regla de Sanger* (APEX) [39] se pueden obtener las componentes ordenados dependiendo de la varianza:

$$\Delta w_{ij} = \eta v_i (x_i - \sum_{k=1}^{i} v_k w_{k_j}) \tag{1.25}$$

1.3.2 Blanqueado y Sphering

Según la definición en 1.1.15, una variable aleatoria $\vec{z}$ esta "blanqueada", si la media es cero y es matriz de la covarianza es igual a la matriz de identidad $\mathbb{I}$. Estos requisitos se pueden considerar como básicos en PCA:

$$\vec{z} = \mathbf{W}\vec{x} \tag{1.26}$$

usando

$$\mathbf{W} = \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T \tag{1.27}$$

donde $\mathbf{V}$ es la matriz de los vectores propios y $\mathbf{D}$ la matriz diagonal de los valores propios respectivos de la matriz de covarianza $\mathbf{C} = \text{cov}(\vec{z})$. Vale la pena mencionar que $\mathbf{W}$ no es la única solución válida, se puede recurrir a muchas más que son posibles.

La ecuación $\mathbf{W}$ también puede ser calculada de manera online con la siguiente regla:

$$\Delta \mathbf{W} = \gamma(\mathbb{I} - \vec{z}\vec{z}^T)\mathbf{W} \tag{1.28}$$

Ortogonalización Simétrica

A veces es necesario calcular una base ortogonal de un subespacio dado. Si por ejemplo los vectores de la base $\vec{a}1, \ldots, \vec{a}_m$, $\vec{a}_i = (a_{i1}, \ldots, a_{in})$, $m < n$ de un subespacio son dados y los vectores ortogonales de la base $\vec{w}_1, \ldots, \vec{w}_m$ que crea el mismo subespacio, se busca:

$$\vec{w}_i = \sum_j c_{ij} a_j \tag{1.29}$$

Entonces $\mathbf{W} = (\vec{w}_1 \ldots \vec{w}_m)$ puede ser calculado de $\mathbf{A} = (\vec{a}_1 \ldots \vec{a}_m)$ de manera que:

$$\mathbf{W} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-\frac{1}{2}} \tag{1.30}$$

Por lo que se tiene:

$$\mathbf{W}^T\mathbf{W} = \mathbb{I}. \tag{1.31}$$

1.4 Análisis de Componentes Independientes

La idea detrás del *análisis de componentes independientes (ICA)* se puede resumir como la utilización de la información estadística contenida en los datos mezclados, recogidos por diferentes fuentes, con el fin de encontrar las componentes básicas que permitan una reconstrucción de las señales originales. Esta característica del funcionamiento de este método es también lo que lo diferencia de otros ya que usa información a priori.

Definition 1.4.1 *Si* $\vec{x} : \Omega \to \mathbb{R}^m$ *es un vector aleatorio y* $\mathbf{W}$ *pertenece a los reales* $n \times m$ *es la matriz completa, entonces* $\mathbf{W}$ *se llama* Análisis de Componentes Independientes (ICA) *de* $\vec{x}$ *si* $\vec{y} := \mathbf{W}\vec{x}$ *es mutuamente independiente. Las componentes* y_i *de* $\vec{y}$ *se llaman* componentes independientes *(IC) de* $\vec{x}$*.*

Si $\mathbf{W}$ es un ICA de $\vec{x}$ la información mutua $I(\mathbf{W}\vec{x})$ es cero.

Definition 1.4.2 *Un ICA es llamado* cuadrado (square)*, si el número de columnas de la matriz de mezcla* $\mathbf{W}$ *es igual al número de sus filas, es decir,* $m = n$ *y* $\mathbf{W} \in GL(n)$*.*

Definition 1.4.3 *Una trasformación de* $\vec{x} \mapsto W(\vec{x})$ *se llama* algoritmo ICA*, si* $\mathbf{W}(\vec{x})$ *es un ICA de* $\vec{x}$*.*

1.4.1 El Modelo de ICA Linear (Sin Ruido)

En el análisis de componentes independientes, la variable aleatoria $\vec{x}$ se interpreta como la mezcla de las fuentes subyacentes $\vec{s}$. Esta mezcla puede ser linear o no linear. En esta tesis trataré solamente el problema linear de ICA: Aquí las fuentes independientes $\vec{s}$ son mezcladas por una matriz que se mezcla linear $\mathbf{A}$:

$$\vec{x} = \mathbf{A}\vec{s} \tag{1.32}$$

Un ICA pede separar un vector aleatorio mezclado $\vec{x}$ en sus componentes independientes tal y como se ha definido en 1.4.1:

$$\vec{y} = \mathbf{W}\vec{x} \tag{1.33}$$

Idealmente, las componentes independientes $\vec{y}$ podrían ser iguales a las fuentes $\vec{s}$ y la matriz de separación $\mathbf{W}$ podría ser inversa a la matriz de mezcla $\mathbf{A}$. Sin embargo, sin la información anterior de la distribución de las fuentes, el resultado de un ICA no es único, esto significa que resulta imposible determinar en que orden secuencial estaban las fuentes originales (permutación) y si un factor linear era parte de la matriz que se mezclaba o de la fuente.

Theorem 1.4.1 *Si* $\mathbf{W}$ *es un ICA de* $\vec{x}$*, entonces* $\mathbf{LPW}$ *es un ICA de* $\vec{x}$*, con el operador del escalamiento*

$$\mathbf{L} = \begin{pmatrix} \lambda_1 & & \\ & \dots & \\ & & \lambda_n \end{pmatrix}, \quad \lambda_i \neq 0$$

y el operador permutación $\mathbf{P}$ *(solamente uno* 1 *por fila y columna, el resto de los valores en la matriz son ceros).*

1.4.2 Restricciones en el Modelo de ICA

Para ser matemáticamente válido y encontrarse bien definido, el modelo de ICA tiene que satisfacer algunas restricciones:

- $\vec{x} \in L^2(\Omega, \mathbb{R})$, de modo que existan los valores de covarianza y esperanza.
- $E(\vec{x}) = 0$ esto significa $\vec{x}$ sea centrado. Porque cerca $\vec{x}' = \vec{x} - E(\vec{x})$ se tiene

$$\vec{y}' = \mathbf{W}\vec{x}' = \mathbf{W}\vec{x} - \mathbf{W}E(\vec{x})$$

así que $\mathbf{W}$ es un ICA de $\vec{x}$.

- La distribución de $\vec{x}$ y $\vec{s}$ tiene que ser blanqueado, es decir $\text{cov}(\vec{x}) = \mathbf{I}$ y $\text{cov}(\vec{s}) = \mathbf{I}$. Esto no es realmente una restricción impuesta por las características matemáticas del problema, sino que es una manera de mantener los algoritmos más simples. Siempre es posible blanquear datos no blanqueados $\vec{x}$ por $\vec{z}' := \mathbf{V}\vec{x}$ y entonces calculando ICA $\vec{y}' := \mathbf{W}'\vec{z}$, así que ICA de $\vec{x}$ simplemente resulta $\mathbf{W} := \mathbf{W}'\mathbf{V}$.

- $\mathbf{W} \in O(n) := \{\mathbf{W}|\mathbf{W}\mathbf{W}^T = \mathbf{I}\}$ and $\mathbf{A} \in O(n)$, porque $\mathbf{I} = \text{cov}(\vec{x}) = \mathbf{A}\text{cov}(\vec{s})\mathbf{A}^T = \mathbf{A}\mathbf{A}^T$.

Además, solamente una de las s_i puede tener una distribución gaussina. Si más s_i son gaussinas $\mathbf{A}\vec{s}$ sería gaussina para $\mathbf{A} \in O(n)$ con lo que se tendría que una separación por medio de la técnica ICA sería imposible:

Theorem 1.4.2 *(Comon 1994): El BSS de $(\vec{s}, \mathbf{A})$, es decir, las ICAs de $\mathbf{A}\vec{s}$ tienen la forma $\mathbf{LPA}^{-1}$, si más de un s_i es no gaussiano.*

Este teorema puede ser reformado por el teorema de Damois/Skitovitek después del blaqueamiento. Una generalización existe para los casos $m \neq n$ y para BSS post-no-linear. El teorema no puede aplicarse en modelos generales no lineales.

1.4.3 Algoritmos ICA

Para la separación de las fuentes, resultan equivalentes las siguientes condiciones:

- La información mutua de las fuentes es mínima.

- La negentropía y de las fuentes es máxima.

- La distribución de cada fuente es tan no-gaussiana como posible.

- La información entre las mezclas y las fuentes es máxima.

En las siguientes secciones daré una descripción corta de los diferentes algoritmos que utilizan las condiciones anteriormente mencionadas.

Maximum Nongaussianity

El teorema del límite central (1.1.4) proporciona una primera idea intuitiva: La distribución de una suma de variables independientes aleatorias es normalmente mucha más cercana a una distribución gaussina, que la distribución de la variable aleatoria. Dado $\vec{x}$ solamente tenemos que encontrar los coeficientes b_i tales que $Y = \sum_i b_i x_i$ es no-gaussina máxima. De esta manera podemos encontrar las componentes independientes.

Comenzado con el modelo de mezcla de ICA

$$\vec{x} = \mathbf{A}\vec{s} \tag{1.34}$$

y buscando un ICA con las características de:

$$\mathbf{W}\vec{x} = \mathbf{A}^{-1}\vec{x} = \vec{s} \tag{1.35}$$

Vale la pena resaltar que en primer lugar buscamos solamente soluciones para una componente; con lo que tenemos:

$$y = \sum_i b_i x_i = \vec{b}^T\vec{x} = \vec{b}\mathbf{A}\vec{s} =: \vec{q}^T\vec{s} \tag{1.36}$$

Tenemos que encontrar $\vec{b}$ donde $\vec{b}^T\vec{x}$ es lo más lejano posible de la distribución gaussina. Por consiguiente es necesario medir la "distancia" a la distribución gaussiana.

Kurtosis como una medida de la "Gaussianidad"

En la sección 1.1.21 hemos definido la kurtosis como:

$$\text{kurt}(y) = E(y^4) - 3(E(y^2))^2 \tag{1.37}$$

Si y tiene una distribución gaussina, entonces tenemos $E(y^4) = 3(E(y^2))^2$ de modo que $\text{kurt}(y) = 0$. De esta manera la kurtosis puede servir como una medida simple para determinar la distancia a la distribución gaussina; sin embargo debe ser observarse que $\text{kurt}(y) = 0$ puede aparecer también para no-gaussinas y.

Si los datos del sistema han sido blanqueados, se tiene:

$$\text{kurt}(y) = E(y^4) - 3 \tag{1.38}$$

En este caso la kurtosis es un momento de cuarto orden normalizado.

Algoritmo

Estamos buscando $\vec{s}$. Después del blanqueamiento $\vec{z} = \mathbf{V}\vec{x}$ tenemos que encontrar $\vec{w}$ tal que $\vec{w}^T\vec{z}$ es una no-gaussina máxima. Por medio de:

$$\vec{q} = (\mathbf{VA})^T\vec{w} \tag{1.39}$$

podemos encontrar $\mathbf{I} = (\mathbf{VA})(\mathbf{VA})^T$:

$$||q||^2 = \vec{q}^T\vec{q} = \vec{w}^T(\mathbf{VA})(\mathbf{VA})^T\vec{w} = \vec{w}^T\vec{w} = ||\vec{w}||^2 \tag{1.40}$$

Así con $\vec{q} \in S^{n-1}$ también tenemos $\vec{w} \in S^{n-1}$.

Enunciando el algoritmo de una manera mas corta, se puede decir: maximizando $\vec{w} \mapsto |\text{kurt}(\vec{w}^T\vec{x})|$ en S^{n-1} con $||\vec{w}|| = 1$ debido a que previamente se ha blanqueado.

La manera más común de maximizar una función es realizando actualizaciones locales en la dirección del gradiente. Esto resulta muy bueno si se está buscando máximos locales. Para esto hacemos:

$$w^{(n+1)} := w^{(w)} + \eta n \Delta w^{(n)}, \text{ mit } \Delta w = \text{grad}(f) \tag{1.41}$$

El gradiente de $|\text{kurt}(\vec{w}^T z)|$ es

$$\frac{\partial |\text{kurt}(\vec{w}^T z)|}{\partial \vec{w}} = 4 * \text{sign}(\text{kurt}(\vec{w}^T z))(E(\vec{z}(\vec{w}^T z)^3 - 3\vec{w}||\vec{w}||^2) \tag{1.42}$$

Esto es posible debido a que la optimización se produce de manera circular ($||\vec{w}|| = 1$), de esta manera el algoritmo resulta muy simple:

1. Calculamos el cambio para el siguiente paso: $\Delta w \propto \text{sign}(\text{kurt}(\vec{w}^T z) E(z(\vec{w}^T \vec{z})^3)$,
2. Fijamos los nuevos pesos: $\vec{w} \leftarrow \vec{w} + \eta \Delta \vec{w}$,
3. Normalizamos los nuevos pesos: $\vec{w} \leftarrow \frac{\vec{w}}{||\ \vec{w}||}$,
4. Continuamos con el primer paso hasta que se alcanzan los máximos locales.

Algoritmo de Punto Fijo (FastICA)

Una función f de S^n tiene un extremo en la posición $\vec{w}_0$, if $\text{grad} f(\vec{w}_0) \propto \vec{w}_0$. Para nuestro caso tenemos:

$$\vec{w} \propto \frac{\partial \text{kurt}(\vec{w}^T \vec{z})}{\partial \vec{w}} = E(\vec{z}(\vec{w}^T \vec{z})^3) - 3\vec{w} \tag{1.43}$$

Obteniendo el siguiente resultado para el algoritmo:

$$\vec{w} \leftarrow E(\vec{z}(\vec{w}^T \vec{z})^3) - 3\vec{w} \tag{1.44}$$

Con este método tenemos la ventaja que no hay parámetros que tienen que ser elegidos y que la convergencia es cúbica.

Generalización: Usando la Negentropía para medir la no-Gaussianidad

La negentropía según la definición dada en 1.2.3 se puede usar también como una medida de la distancia de una distribución a la distribución gaussina:

$$J(\vec{y}) = H(\vec{y}_{\text{Gauss}} - H(\vec{y}) \tag{1.45}$$

Una buena aproximación es:

$$J(\vec{y}) \approx \frac{1}{12}E(y^3)^2 + \frac{1}{48}\text{kurt}(y)^2 + \dots \tag{1.46}$$

Usando esta relación es posible generalizar el algoritmo de FastICA. Hasta ahora sólo hemos estimado una fila de $\mathbf{W}$. Sin embargo, la matriz de separación puede ser estimada con lo anterior debido a que las filas y las columnas de la matriz de separación blanqueada $\mathbf{W}$ son ortogonales ($E((\vec{w}_i^T\vec{z})(\vec{w}_j^T\vec{z})^T) = \vec{w}_i^T\vec{w}_j = \delta_{ij}$). Esto es utilizado por el algoritmo de la deflación de FastICA:

1. Eligir el número de IC m para estimar y fijar $p \leftarrow 1$,
2. Inicializar $\vec{w}_p$ aleatoriamente,
3. Una iteración de FastICA para $\vec{w}_p$
4. $\vec{w}_p \leftarrow \vec{w}_p - \sum_{j=1}^{p-1}(\vec{w}_p^T\vec{w}_j)\vec{w}_j$
5. $\vec{w}_p \leftarrow \frac{\vec{w}_p}{||\vec{w}_p||}$
6. Si el algoritmo no converge, regresar al paso 3.
7. Sistema de datos $p \leftarrow p + 1$,

8. Si $p < m$ regresar al paso 2.

ICA aplicando Estimación de la Máxima Probabilidad

Como se ha definido anteriormente la máxima probabilidad es una manera de estimar los parámetros más probables en una observación dada.

Comenzando de nuevo con el modelo linear de ICA

$$\vec{x} = \mathbf{A}\ \vec{s} \tag{1.47}$$

Sea $\mathbf{B} := \mathbf{A}^{-1}$. Lo que conduce a:

$$p_{\vec{x}}(\mathbf{A}\vec{s}) = |\det \mathbf{B}| p_{\vec{s}}(\vec{s}) = |\det \mathbf{B}| \prod_i p_i(s_i) \tag{1.48}$$

donde $p_i := p_{s_i}$ es la densidad del pobabilidad del i-th IC. De $\vec{x} := \mathbf{A}\vec{s}$ podemos derivar la siguiente expresión:

$$\vec{s} = \mathbf{B}\vec{x} \tag{1.49}$$

$\mathbf{B} = (\vec{b}_1, \ldots, \vec{b}_n)^T$ con las filas $\vec{b}_i^T$, so

$$s_i = (\mathbf{B}\vec{x})_i = b_i^T \vec{x} \tag{1.50}$$

Usando el anterior resultado para la ecuación de la probabilidad, obtenemos:

$$p_{\vec{x}}(\vec{x}) = |\det \mathbf{B}| \prod_i p_i(\vec{b}_i^T \vec{x}) \tag{1.51}$$

El likelihood si una muestra dada iid $\vec{x}(1), \ldots, \vec{x}(T) \in \mathbb{R}^n$ se define como:

$$L(\mathbf{B}) := \prod_{t=1}^{T} p(\vec{x}(t)|\mathbf{B}) = \prod_{t=1}^{T} (\prod_{i=1}^{n} p_i(\vec{b}_i^T \vec{x}(t)))|\det(\mathbf{B})| \tag{1.52}$$

El log-likelihood es

$$\ln L(\mathbf{B}) = \sum_{t=1}^{T} \sum_{i=1}^{n} \ln p_i(\vec{b_i^T} \vec{x}(t)) + T \ln |\det \mathbf{B}| \tag{1.53}$$

con la media de la muestra:

$$\frac{1}{T} \ln L(\mathbf{B}) = E\left(\sum_{i=1}^{n} \ln p_i(\vec{b_i^T} \vec{x}) \right) + \ln |\det \mathbf{B}| \tag{1.54}$$

El problema ahora es que aparte de el modelo paramétrico $\mathbf{B}$ también las densidades de las fuentes p_i tienen non-parametricly. Debido a esto, éste se denomina "semiparametric". Este problema puede ser solucionado usando la información anterior (known p_i) o realizando una aproximación de p_i con una familia paramétrica de las densidades de la probabilidad.

El siguiente teorema resulta útil para una aproximación de p_i por las familias binarias de densidades:

Theorem 1.4.3 *Si $\tilde{p}_i$ son las densidades aproximadas supuestas de los componentes independientes; si $y_i := b_i^T \vec{x}$ esta blanqueado, se puede realizar la siguiente relación:*

$$g_i(s_i) := \frac{\partial}{\partial s_i} \ln \tilde{p}_i = \frac{\tilde{p}_i'}{\tilde{p}_i}(s_i), \tag{1.55}$$

Entonces el estimador de ML es localmente constante ($\hat{\mathbf{B}} \rightarrow \mathbf{B}$ *localmente en la distribución para $T \rightarrow \infty$), si*

$$E(s_i g_i(s_i) - g'(s_i)) > 0 \tag{1.56}$$

Ahora la idea es construir clases de densidades con

$$\ln \tilde{p}^+(s) \;\; := \;\; \alpha_1 - 2 \ln \cosh(s) \tag{1.57}$$

$$\ln \tilde{p}^{-}(s) \;:=\; \alpha_2 - \left(\frac{s^2}{2} - \ln\cosh(s)\right) \tag{1.58}$$

con $\alpha_i > 0$ tal que $\int \tilde{p}^{\pm} = 1$. Se obtiene que estas densidades diferencian comparando la distribución gaussina: $\tilde{p}^{+}(s)$ es super-gaussiana, $\tilde{p}^{-}(s)$ es sub-gaussiana. El resultado es:

$$g_i^{+}(s_i) \;=\; -2\ln\cosh s_i \tag{1.59}$$

$$g_i^{+}(s_i)' \;=\; -2\tanh s_i \tag{1.60}$$

$$g_i^{-}(s_i) \;=\; -\frac{s_i}{2} + \ln\cosh s_i \tag{1.61}$$

$$g_i^{-}(s_i)' \;=\; -s_i + \tanh s_i \tag{1.62}$$

De esta manera a las densidades definidas en 1.57 y 1.58 satisfacen casi siempre la ecuación 1.56 debido a $E(s_i^2) = 1$. Sin embargo es necesario elegir densidad correcta para cada una de las componentes. Dependiendo de la opción de $\tilde{p}^{\pm}$ la kurtosis será máxima o mínima.

EL algoritmo de Bell-Seiynowski (1995)

Bell and Seiynowski usaron en su algoritmo (publicado en [5]) el siguiente gradiente log-likelihood:

$$\frac{1}{T}\frac{\partial \ln L(\mathbf{B})}{\partial \mathbf{B}} = (\mathbf{B}^T)^{-1} + E(\vec{g}(\vec{\mathbf{B}}\vec{x})\vec{x}) \tag{1.63}$$

con $\vec{g} := (g_i(y_1, \ldots, g_n(y_n))$ y las scorefunctions negativas

$$g_i = \frac{p_i'}{p_i}. \tag{1.64}$$

Usando esta ecuación, estos dos autores proponen:

$$\Delta \propto (\mathbf{B}^T)^{-1} + E(\vec{g}(\vec{\mathbf{B}}\vec{x})\vec{x}^T), \tag{1.65}$$

y para la versión estocástica

$$\Delta \propto (\mathbf{B}^T)^{-1} + \vec{g}(\vec{\mathbf{B}}\vec{x})\vec{x}^T. \tag{1.66}$$

El problema de este algoritmo es que con frecuencia no converge bien y requiere de mucho tiempo de computo. Sin embargo, estos problemas pueden ser enfrentados usando el gradiente natural.

Gradiente Natural por Amari

Amari y col. introducen en [1] el gradiente natural en el algoritmo de Bell-Seiynowski: Si en lugar de la geometría euclidiana se usa la estructura del grupo de Lie $Gl(n)$, el *gradiente natural* resulta:

$$\mathrm{grad}_{\mathrm{nat}}(\mathbf{B}) = \mathrm{grad}_{\mathrm{Eukl}} f(\mathbf{B})\mathbf{B}^T\mathbf{B} \tag{1.67}$$

de modo que

$$\Delta\mathbf{B} \propto (\mathbb{I} + e(g(\vec{x})\vec{y}^T))\mathbf{B} \tag{1.68}$$

con $\vec{y} := \mathbf{B}\vec{x}$ y la matriz de identidad $\mathbb{I}$.

Infomax

El algoritmo *Infomax* [5] se basa en la maximización de la información entre las entradas y las salidas de una red neuronal. La salida de la red es

$$y_i = \Phi_i(\vec{b_i^T}\vec{x}) + n \tag{1.69}$$

con la función monotónica de la activación Φ_i, la ecuación del perceptron $\vec{b_i^T}\vec{x}$ y el ruido gaussino n. Se tiene una entropía en la salida:

$$H(\vec{y}) = H(\Phi_{1(\vec{b_1^T}\vec{x}}, \ldots, \phi_n(\vec{b_1^T}\vec{x})) \tag{1.70}$$

En el caso del límite $|n| \to 0$ un máximo en la entropía corresponde a un máximo en el flujo de la información $MI(\vec{x}, \vec{y})$. Entonces tenemos

$$H(\vec{y}) = H(\vec{x}) + e(\log|\det\frac{\partial F}{\partial \mathbf{B}}|) \tag{1.71}$$

con

$$F(\vec{x}) = (\phi_1(\vec{b_1^T}\vec{x}), \ldots, \Phi_n(\vec{b_1^T}\vec{x})) \tag{1.72}$$

El criterio del log-likelihood es:

$$H(\vec{y}) = H(\vec{x}) + /sum_{i=1}^{n} E(\log\Phi_i'(\vec{b_i^T}\vec{x})) + \log|\det\mathbf{B}| \tag{1.73}$$

El algoritmo de infomax por consiguiente, corresponde a una valoración de la probabilidad si Φ_i aproxima las densidades de las fuentes.

1.4.4 ICA Métodos que utilizan Estructuras Temporales

Hasta este momento hemos tratado solamente las mezclas de variables aleatorias sin estructuras adicionales, es decir, iid muestras. No obstante para las señales temporales que dependen estadísticamente unas de otras, se requieren de otros métodos para su correcto análisis.

Definition 1.4.4 *Una secuencia de vectores aleatorios* $\vec{x}(t), t = 1, 2, \ldots$ *se conoce como* proceso estocástico discreto. $(\vec{x}(t))_t$ *se llama iid, si* $\vec{x}(t)$ *tiene distribución uniforme y es mutuamente independiente. A realisation (path) de* $(\vec{x}(t))_t$ *puede ser construido para* $\omega \in \Omega$ *por* $\vec{x}(1)(\omega), \vec{x}(2)(\omega), \ldots$ *(secuencia en* $\mathbb{R}^n$*).*

El valor de la esperanza del proceso es la secuencia sobre el valor de la esperanza de la variable aleatoria. Especialmente:

$$\vec{m}_{\vec{x}}(t) = E(\vec{x}(t)) = \int_{\mathbb{R}^n} \vec{x} p_{\vec{x}(t)}(\vec{x}) d\vec{x} \tag{1.74}$$

Definition 1.4.5 *La* auto-covarianza *y la* auto-correlación *del proceso* $(\vec{x}(t))_t$ *son definidos de la siguiente manera* $\tau \in \mathbb{N}_0, t > \tau$*:*

$$C_\tau^{\vec{x}}(t) = Cov(\vec{x}(t), \vec{x}(t-\tau)) \tag{1.75}$$

$$R_\tau^{\vec{x}}(t) = Cor(\vec{x}(t), \vec{x}(t-\tau)) \tag{1.76}$$

A continuación miraremos el modelo que se mezcla instantáneo

$$\vec{x}(t) = \mathbf{A}\vec{s}(t) \tag{1.77}$$

para los procesos estocásticos $\vec{x}, \vec{s}$, donde $\mathbf{A} \in Gl(n)$ y $C_\tau^{\vec{s}}$ diagonal para todos τ, t. Aquí $s_i(t)$ no tienen que ser solucionar el modelo no-gaussino. En lugar de la estadística de

alto orden, la estadística de segundo orden es suficiente para la auto-covarianza:

Decorrelación Temporal

Trabajando ahora sin la restricción, sea $\vec{m}_{\vec{x}}(t) = 0$ y $\mathbf{A} \in O(n)$. Sea $\tau \in \mathbb{N}$. Debido a la condición previa $\mathbf{A}^T = \mathbf{A}^{-1}$ tenemos:

$$\mathbf{A}^T \vec{x}(t) = \vec{s}(t) \tag{1.78}$$

$$\mathbf{A}^T \vec{x}(t-\tau) = \vec{s}(t-\tau) \tag{1.79}$$

Ahora podemos definir una auto-covariaza simétrica que permita servirnos de ayuda para solucionar este problema:

Definition 1.4.6 *La* auto-covarianza simétrica *se define como:*

$$\bar{C}_\tau^{\vec{x}} := \frac{1}{2}(C_\tau^{\vec{x}} + (C_\tau^{\vec{x}})^T) \tag{1.80}$$

Por esta definición resulta:

$$\bar{C}_\tau^{\vec{x}} = \frac{1}{2}(\mathbf{A} C_\tau^{\vec{s}} \mathbf{A}^T + (\mathbf{A} C_\tau^{\vec{s}} \mathbf{A}^T)^T) \tag{1.81}$$

$$= \mathbf{A} C_\tau^{\vec{s}} \mathbf{A}^T \tag{1.82}$$

y corresponde a una descomposición del valor propio.

Si asumimos $\bar{C}_\tau^{\vec{s}}$ (y con esto también $\bar{C}_\tau^{\vec{x}}$) tiene n diferentes valores propios, entonces la descomposición de $\bar{C}_\tau^{\vec{x}}$ es única (y también con la descomposición $\mathbf{A}$) la permutación y el escalamiento, porque cada espacio del valor propio tiene una dimensión. Esto también significa que el problema es separable.

El Algoritmo AMUSE

El algoritmo AMUSE puede formularse de una manera muy simple:

1. Sea $\vec{x}(t)$ blanqueado y sea $C_\tau^{\vec{x}}$ con n diferentes valores propios.

2. Se calcula la descomposición del valor propio de $C_\tau^{\vec{x}}$:

$$C_\tau^{\vec{x}} = \mathbf{W}^T D \mathbf{W} \tag{1.83}$$

   con $\mathbf{W} \in O(n)$ y D diagonal.

3. Entonces $\mathbf{W}$ es la matriz de separación:

$$\mathbf{W}^T = \mathbf{W}^{-1} \sim \mathbf{A} \tag{1.84}$$

Sin embargo, la condición que todos los n valores propios existen son muy fuerte, es a menudo un problema. Los valores propios son $\mathrm{Cov}(s_i(t), s_i(T-\tau))$ y tienen que ser muy diferentes, lo cual resulta especialmente problemático con señales que tienen espectros similares de energía. Una solución posible es utilizar múltiples traslados τ, que desafortunadamente da lugar a problemas en la diagonalisación simultánea en el mismo tiempo.

1.4.5 PCA versus ICA

A menudo, ICA se mal-interpreta como "una clase de PCA, que permite obtener resultados más estadísticos." Sin embargo, mientras que ambos métodos se basan claramente en estadística, los resultados derivados de ellos se diferencian substancialmente:

PCA: Los ejes de la transformación de la separación muestran la dirección de la mayor variación en el sistema de datos y en direcciones ortogonales.

ICA: Los ejes de la transformación de la separación muestran la dirección de la máxima independencia estadística máxima en el sistema de datos.

Figura 1.2 demuestra este efecto. Muestra la uniformidad en fuentes distribuidas s_1 y s_2, mezclado por la matriz de mezclar

$$\mathbf{A} = \begin{pmatrix} 0.90 & 0.80 \\ 0.04 & 0.30 \end{pmatrix}.$$

Como puede verse los ejes de PCA y de ICA no señalan en la misma dirección, de hecho después de aplicar PCA a los datos las fuentes serán más dependientes que antes. Igualmente con ICA no tomará los componentes resultantes basadas en la contribución a la variación, las componentes resultantes contribuirán de forma absoluta en mayor o menor cantidad de variación a las mezclas.

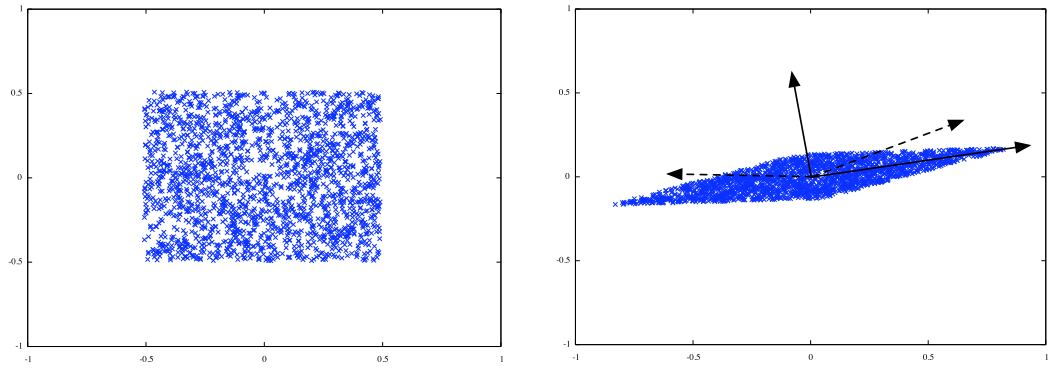


Figure 1.2: En el lado izquierdo de la figura, se muestran las fuentes antes de aplicar la matriz que de mezcla. En el lado derecho se muestra las mezclas obtenidas y los ejes de PCA (sólido) y de ICA (punteado).

2 ICA Reliability

In the real world ICA is applied unsupervised, that means that the right result of the analysis is not known before. This leads to the problem that it is not possible to decide how well the ICA algorithm has worked by simply comparing the result with the real sources. To make things further difficult unsupervised ICA algorithms will give an estimate for the demixing matrix even if the mixing process does not correspond to the ICA model.

So, in the application of ICA a mayor problem how to determine the reliability of the estimated independent components. This can be attributed to various reasons [18]:

- Algorithmic uncertainty: unstable algorithms may result in different results for different initial conditions.
- The ICA model may not be totally adequate to the underlying data generation process that generated the data set.
- The finite sample size induces statistical errors in the estimation that result in local minima. (see figure 2.1)

2.1 Methods of reliability testing

A common way to test the reliability of an ICA is to run the algorithm multiple times and compare the results. In this chapter I will present three popular methods that are

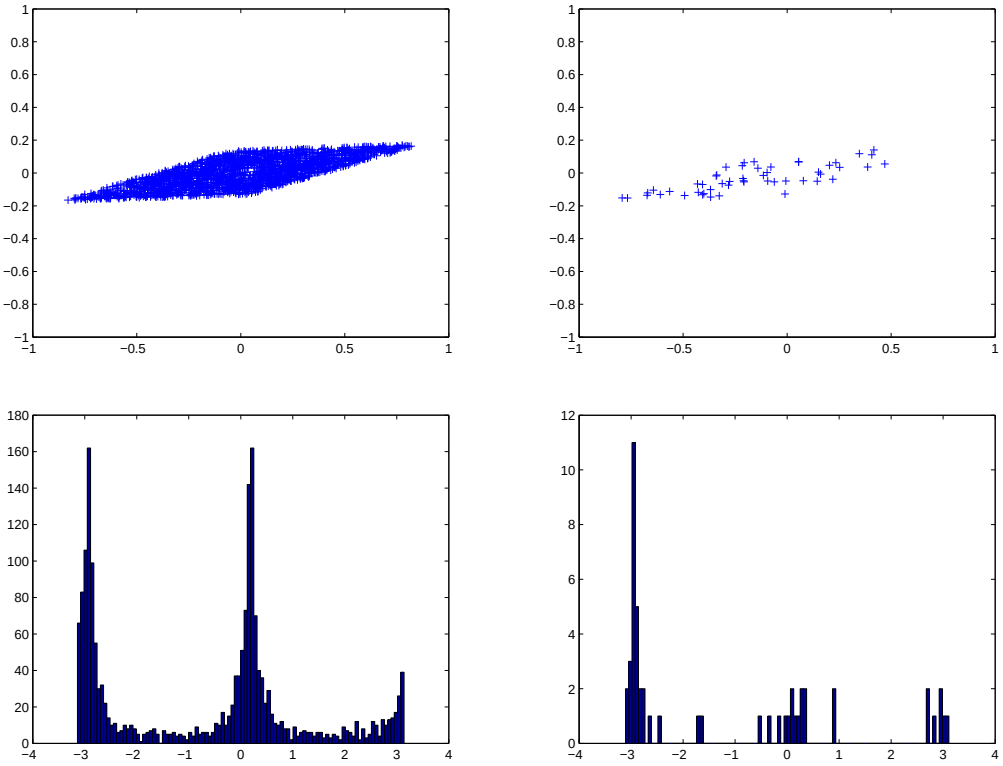


Figure 2.1: Scatterplots of the same two-dimensional mixtures (the same as in figure 1.2) above, below the histogram of the desity of the points relative to the angle. The plots on the left show 2000 datapoints, the right plots only 50 datapoints. While an ICA algorithm will easily find the optimal position of the demixing axes in the left data set, in the right data set many local minima exist in the score function for the demixing matrix.

possible.

2.1.1 Initial Settings of Stochastic Algorithms

In ICA the goal of the algorithms is to maximise or minimise a given score function, also known as energy function. Apart from the sign there is no difference between a maximum and a minimum, therefore I will treat only the case of minima in this thesis.

Analytic algorithms can calculate the entire function or it's derivatives and are thus able to calculate the global minima. Stochastic algorithms usually do not calculate the entire energy function, instead they usually depend on gradient descend algorithms to find the global minima. These kind of algorithms however are sensible to local minima in the score function and depending on the start point of the calculation they can trap themselves into a local minima instead of the global minima.

Most ICA algorithms are stochastic, e.g. FastICA and Infomax. Fortunately, being trapped in one local minima in the case of ICA does not mean that all the calculated independent components are flawed; due to the decoupled estimation of the components a subset of the components may be estimated correctly [19]. Starting the stochastic algorithm multiple times with different initial settings will result in different solutions and thus different independent components if local minima are present. In a good data set with few local minima the global will represent the main part of these solutions.

2.1.2 Resampling the Data Set

Another way to change the energy function is to change the data set or to create a new data set out of the original data set. This can eliminate local minima while they should not affect good global minima. However, care has to be taken that the characteristics of the data that is important for the ICA algorithm is not changed.

The two most popular methods are [40]:

Jackknife: One or few data points are cut out of the data set. If the data is not sparse this is a only a marginal change and thus the statistical characteristics of the data should not change.

Bootstrapping: Points of the original are picked randomly and with this points a new data set is created. This is only valid for iid samples as it will destroy temporal or spatial structures in the data set.

It is possible to change the bootstrapping method so that it retains the local structure of the original data. Meinecke et al propose in [28] a time structure preserving bootstrap where a series $\{a_t\}$ is defined by the bootstrapping with each a_t indicating how often the data point $\vec{x}(t)$ has been drawn. Using that they calculate the resampled time-lagged correlation matrix as

$$C(\tau) = \frac{1}{T}\sum_{t=1}^{T} a_t \vec{x}(t)\vec{x}^T(t+\tau) \tag{2.1}$$

with $\sum a_t = T$ and $a_t \in \{0, 1, 2, \ldots\}$.

Linear Filters

Another method proposed by Meinecke et al in [28] is the use of a (random) linear filter $\mathbf{F}$ on the mixed data:

$$\mathbf{F}x_i(t) = \sum_{\tau=0}^{T} f_\tau x_i(t-\tau) = \sum_j A_{ij}\mathbf{F}s_j(t) \tag{2.2}$$

They argue that since the mixing matrix $\mathbf{A}$ commutes with the filter operator and the filtered sources $s'_j = \mathbf{F}s_j(t)$ are still mutually independent, the filtered signals $x'_i(t) = \mathbf{F}x_i(t)$ "can be interpreted as linear mixtures of the filtered sources with the same mixing matrix $\mathbf{A}$."

2.1.3 Noise Injection

Stefan Harmeling et al propose in [17] to test the reliability of independent components by partially corrupting the data with stationary white gaussian noise. Starting with the ICA model

$$\vec{x} = \mathbf{A}\vec{s} \tag{2.3}$$

$$\vec{y} = \mathbf{W}\vec{x} \tag{2.4}$$

they first normalise the mixing matrix $\mathbf{A}$ so that it has unit-length columns. Then they define an so called energy matrix that contains the square roots of the estimated components on the diagonal, i.e. their standard deviations:

$$E = \begin{pmatrix} \sqrt{\frac{1}{T} Y_{1:} Y_{1:}^T} & & 0 \\ & \ddots & \\ 0 & & \sqrt{\frac{1}{T} Y_{n:} Y_{n:}^T} \end{pmatrix} \tag{2.5}$$

Then R instances of gaussian white noise in the form of $n \times T$ matrices $N^{(1)}, \ldots N^{(R)}$ are added to the estimated components. The noise has to be adjusted by the energy matrix:

$$\begin{aligned} \tilde{Y}^{(1)} &= \cos(\sigma) Y + \sin \sigma E N^{(1)} \\ &\vdots \\ \tilde{Y}^{(R)} &= \cos(\sigma) Y + \sin \sigma E N^{(R)} \end{aligned}$$

$0 \leq \sigma \leq \pi/2$ is the noise-parameter: $\sigma = 0$ does not add noise, $\sigma = \pi/2$ produces only noise. Since $\cos^2(\sigma) + \sin^2(\sigma) = 1$ it is garanteed that the noisy versions have in each component the same energy as the components in the original estimates. Harmeling et

al give $\sigma = \pi/8$ as a good value based on empirical evidence [17].

Additionally, the noisy components $\tilde{Y}$ can be mixed by randomly generated mixing matrixes $\mathbf{B}$ without changing their energies.

2.2 Comparison of the ICA Runs

After the data has been resampled in one or multiple of the ways shown before, the ICA algorithm can then be applied again to the now resampled data set. In doing this many times and comparing the results it is possible to estimate the reliability of the resulting components.

Like in the case of resampling various methods exist to compare the different runs of an ICA algorithm. In the following I will describe the methods that have been applied to fMRI data so far.

2.2.1 Clustering of the Estimated Demixing Matrix Columns

To cluster the estimated components a measure for the distance between the components is necessary. Himberg et al propose in [19] to simply use their mutual correlation coefficients. Therefore the resulting demixing matrixes $\mathbf{W}_i$ from M ICA runs are collected into a single matrix $\hat{\mathbf{W}} = [\mathbf{W}_1^T \mathbf{W}_2^T \cdots \mathbf{W}_M^T]^T$. The correlation matrix is then given by

$$\mathbf{R} = \hat{\mathbf{W}} \mathbf{C} \hat{\mathbf{W}}^T \tag{2.6}$$

with the covariance matrix $\mathbf{C}$ of the original data set $\mathbf{X}$. It should be noted that this is only valid if the data set is not resampled, for resampling a further normalisation step is necessary, so that the variance of the mixtures does not change.

The final similarity matrix is then defined by the absolute value of the elements of the

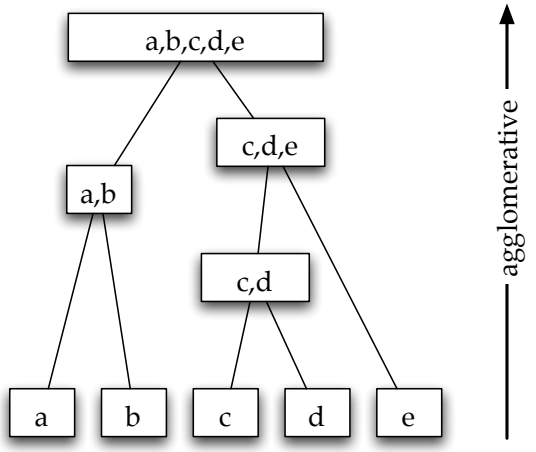


Figure 2.2: Agglomerative hierarchical clustering: This dendrogram shows the main features of AHC. Depending on the deepness of agglomeration more and more elements are formed into cluster, depending on their dissimilarity measure

correlation matrix:

$$\sigma_{ij} = |r_{ij}| \tag{2.7}$$

The elements of the dissimilarity matrix for the clustering algorithms can be calculated straightforward by

$$d_{ij} = 1 - \sigma_{ij} \tag{2.8}$$

Using this dissimilarity index it is now possible to cluster the estimated independent components from the different ICA runs. Himberg et al. [19] suggest agglomerative hierarchical clustering as described in [13, 16] in this case because the tree-like hierarchy (dendrogram) can guide the researcher in choosing the right number if clusters during the analysis. However, Himberg et al note that while there exist numerous reviews and studies on many agglomeration strategies and cluster validity indices, there is no easy way of selecting the optimal clustering strategy for a specific set of data. Instead the selection must be based on problem specific considerations.

In agglomerative hierarchical clustering clusters are constructed of subclusters that again contain subclusters and so on (see figure 2.2). The three basic agglomeration

methods that operate directly on the similarity matrix are the *single-link* (SL), *complete-link* (CL) and *group average-link* (AL). They are defined as follows:

SL: This method is also known as *nearest neighbour technique*. Here distance between clusters is defined as the distance between closest pairs. At each step of the hierarchical clustering the clusters for which the distance is the overall shortest, are merged together

CL: This method is also known as *farthest neighbour technique*. The distance between clusters is defined as the distance between the most distant pair in both clusters. Like in SL, at each step of the hierarchical clustering the clusters for which the distance is the overall shortest, are merged together.

AL: The distance between clusters is defined as the average distance between all pairs of elements in both clusters. Again, at each step of the hierarchical clustering the clusters for which the distance is the overall shortest, are merged together.

For this kind of analysis, J. Himberg also provides a MATLAB packaged called ICASSO[1] on his webpage.

2.2.2 Analysing Angles Between ICA Basisvectors

Harmeling et al [17] and Meinecke et al [28] follow a different approach in the analysis of the reliability of ICA. They argue that an ICA represents a basis transformation along the axes defined by the columns of the demixing matrix. Therefore it is sensible to compare different ICA runs by calculation the angles between the resulting demixing base vectors.

The main idea is to apply the ICA on already separated but resampled components of a first ICA so that the resulting demising matrix should be close to the identity matrix if

[1] http://www.cis.hut.fi/jhimberg/icasso

the components of the first ICA are stable. To compare the different demixing matrixes they use the fact that the matrix representation of the rotation Lie group $SO(N)$ can be parameterised by

$$R(\alpha) = \exp\left(\frac{1}{2}\sum_{i,j}\alpha_{ij}\mathbf{M}^{ij}\right) \tag{2.9}$$

with $(\mathbf{M}^{ij})_{ab} = \delta^i_a\delta^j_b - \delta^j_a\delta^i_b$ where the matrices $\mathbf{M}^{ij}$ are generators if the group and the α_{ij} are the rotation parameters (angles) of the rotation matrix R.

The algorithm is as follows (taken from [28]):

1. Estimate the separation matrix $\mathbf{W}$ using an ICA algorithm. Calculate the ICA-projections $\vec{y} = \mathbf{W}\vec{x}$.

2. Produce k surrogate data sets from $\vec{y}$ and whiten these data sets.

3. For each surrogate data set: produce a set of rotation matrices by performing ICA.

4. Calculate variances of rotation parameters (angels) α_{ij}.

5. For each ICA component calculate the uncertainty $U_i = \max_j \sqrt{\mathrm{Var}(\alpha_{ij})}$

This method to test the reliability of an ICA is very interesting because it allows also to identify stable subspaces in the data set created by independent components that were themselves highly unstable. The existence of these subspaces often is a sign that the applied ICA algorithm is not well suited to the data.

2.2.3 Comparison of the Estimated Components

While the previous two sections used the columns of the (de-)mixing matrix to compare the results of different ICA runs, of course it is also possible to compare the estimated independent components themselves.

Crosscovariance

Usually the crosscovariance between two components is used as a measure for similarity:

$$C(\vec{y}_i, \vec{y}_j) = E((\vec{y}_i - \vec{m}_{\vec{y}_i})(\vec{y}_j - \vec{m}_{\vec{y}_j})^T) \tag{2.10}$$

Applying this technique to fMRI quickly shows it's main shortcoming: fMRI data sets are rather noisy and this manifests itself in a high level of background noise in the independent components. Resampling of the data set usually leads to great changes in this noise background even in components that remain stable in the data related part. However, as the crosscovariance does not differentiate between the noise and the data part of the estimated component, the most part of the crosscovariance may consist of the noise part that changes. Thus stable components will appear unstable using the crosscovariance as measure for the similarity of two components.

A way to circumvent this problem is to cut out the noise part of the components before calculation the crosscovariance. A simple approach is to interpret the noise part as gaussian noise and to cut out all the values in the independent components that are below a certain threshold. I found from empirical evidence (chapter **??**) that 3 times the standard deviation is a good threshold value. However, especially in components with few activated voxels[2], the denoising puts the stress on the exact values of the activation within the component. This also can lead to unwanted changes in the value of the crosscovariance.

Overlap

Another possibility especially for fMRI data analysis is to use the spatial overlap between to components as measure for similarity. All voxels in the independent component, that have their value above the given threshold, are defined as "active". Then the overlap

[2] a voxel is the smallest unit of the brain data set, see chapter **??** for a introduction

between the active voxel of two components is calculated, based on the total number of active voxels in both components.

3 functional Magnetic Resonance Imaging

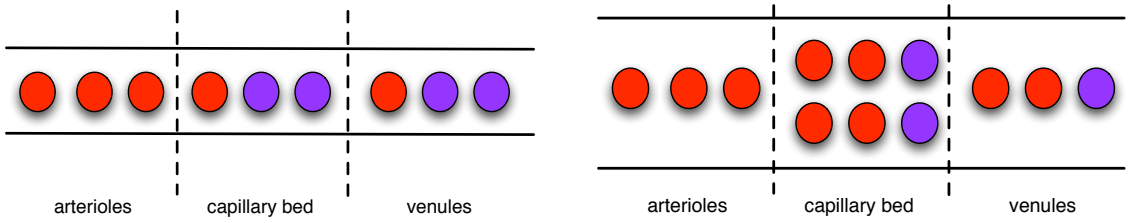
It has been known for a long time that local activity in the brain leads to an increased local flow of blood in the same region. This effect has been well studied [15] and is used in *functional Magnetic Resonance Imaging (fMRI)* to measure the activity in the brain.

3.1 Introduction to fMRI

In normal magnetic resonance imaging (MRI) it is not possible to distinguish between blood and living tissue. To measure the blood flow it is therefore necessary to use marker substances in the blood that can be detected via MRI and whose concentration is proportional to the blood flow. These substances are paramagnetic marker that influence the de-phasing of the spins in the MRI (figure 3.1). Two substances where used this way: Gandolinium compounds [7] and deoxyhaemoglobin [31].

Alignment of the spin of the protons by $B_0 \parallel z$
$\downarrow$
90° RF pulse
$\downarrow$
relaxation and de-phasation of the spins

Figure 3.1: Scheme of a MR image



(a) (b)

Figure 3.2: The BOLD effect: in times of normal activity (a) the brain consumes in this fictive example 2 parts of oxygen pro unit of time. The ratio of oxyhaemoglobin to deoxyhaemoglobin therefor is 1:2. In times of increased activity (b) the consumption of oxygen remains constant. But due to the increased blood flow there is more oxyhaemoglobine in the capillaries so that the ratio changes to 2:1 in the case (b).

The deoxyhaemoglobin is special because it is a substance that forms part of the body and has not to be injected into the blood of the subjects. It is possible to measure the blood flow with it because the consumption of oxygen of the brain is not directly connected to it's activity – in times of higher local activation the brain does not consume more oxygen at the first time compared to times of normal activation. However, the blood flow in the capillaries is increased and so the ratio of oxyhaemoglobin to deoxyhaemoglobin changes in favour of the oxyhaemoglobin in the capillaries and the veins (figure 3.2). This effect is called *BOLD (blood oxygenation level dependent contrast)*

This change in concentration does not follow the activity in the brain directly. It has some time lag and corresponds to a response function (figure 3.3), the so called *haemoglobin response function (HRF)*. Further difficulties for the analysis result because the HRF is not constant over the brain volume and different from subject to subject. Also the BOLD effect only corresponds to 1%–10% of the change in the fMRI signal.

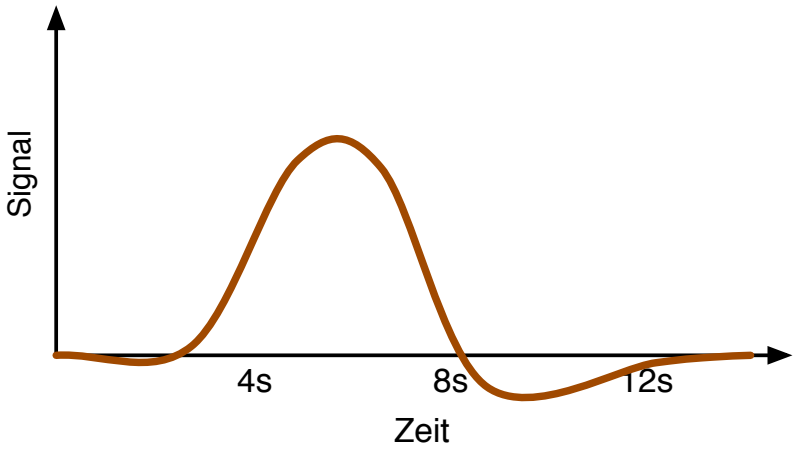


Figure 3.3: The haemoglobin response function (HRF) (fictive). The event of increased activation was at $t = 0$ s

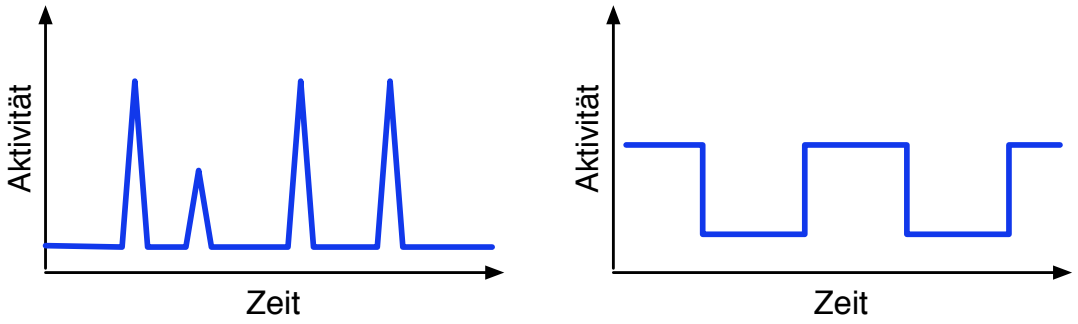
3.1.1 Design of an fMRI Experiment

The human brain is always active and occupied with diverse activities. If one wants to find out in an experiment, which parts of the brain are related to a given task, one has to measure the relative increase in activation. For this it is necessary to compare times with the task ("active") and times without the task ("inactive"). It is therefor important to design the experiment in a way that also times without tasks are measured so that the base state activity can be estimated.

Two kinds of design are used:

- Event design: Short events of stimulus are presented to the subject.
- Block design: The presented stimulus has some duration or the subject has to fulfil tasks

Figure 3.4 shows the time course of activity as example for a event design and a block design.



(a) (b)

Figure 3.4: the time course of the activation function for an event design (a) and a block design (b).

3.2 Advantages of ICA

ICA in the analysis of fMRI data is a rather young field of activity. One of the first publications was in 1998 by McKeown et al [25] who applied spatial ICA on fMRI data of a Stroop color-naming experiment and a word/number experiment. This initial publication soon was followed by various publications on the subject of applying ICA on fMRI data, see [10,26] for an overview. While these works established ICA as a technique that delivers results comparable to the well established fMRI analyse methods like the *general linear model (GLM)* [15], in this chapter I will present a case where ICA actually is able to outperform this classic method.

3.2.1 Model based approach: general linear model

The general linear model as a kind of regression analysis has been the classic way to analyse fMRI data in the past. Basically it uses second order statistics to find the voxels whose activations correlate best to given time courses. The measured signal for each voxel in time $\vec{y} = (y(t_1), ..., y(t_n))^T$ is written as a linear combination of independent variables $\vec{y} = \mathbf{X}\vec{b} + \vec{e}$, with the vector $\vec{b}$ of regression coefficients and the matrix X of the independent variables which in case of an fMRI-analysis consist of the assumed time courses in the data and additional filters to account for the serial correlation of fMRI data. The residual error $\vec{e}$ ought to be minimized. The normal equation $\mathbf{X}^T\mathbf{X}\vec{b} = \mathbf{X}^T\vec{y}$ of the problem is solved by $\vec{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\vec{y}$ and has a unique solution if $\mathbf{X}\mathbf{X}^T$ has full rank. Finally a significance test using $\vec{e}$ is applied to estimate the statistical significance of the found correlation.

As the model $\mathbf{X}$ must be known in advance to calculate $\vec{b}$, this method is called "model-based". It can be used to test the accuracy of a given model, but cannot by itself find a better suited model even if one exists.

3.2.2 Model free approach: BSS using independent component analysis

In case of fMRI data blind source separation refers to the problem of separating a given sensor signal, i.e. the fMRI data at the time t

$$\vec{x}(t) = \mathbf{A}\left[\vec{s}(t) + \vec{s}_{noise}(t)\right] = \sum_{i=1}^{n} a_i s_i(t) + \sum_{i=1}^{n} a_i s_{noise,i}(t)$$

into its underlying n source signals $\vec{s}$ with $a_i(t)$ being its contribution to the sensor signal, hence its mixing coefficient. $\mathbf{A}$ and $\vec{s}$ are unique except for permutation and scaling. The functional segregation of the brain [15] closely matches the requirement of spatially independent sources as assumed in spatial ICA. The term $s_{noise}(t)$ is the time dependent noise. Unfortunately, in fMRI the noise level is of the same order of magnitude as the signal, so it has to be taken into account. As the noise term will depend on time, it can be included as additional components into the problem. This problem is called "under-determined" or "over-complete" as the number of independent sources will always exceed the number of measured sensor signals $x(t)$.

Various algorithms utilizing higher order statistics have been proposed to solve the BSS problem. In fMRI analysis, mostly the extended Infomax (based on entropy maximisation [5,27]) and FastICA (based on negentropy using fix-point iteration [20]) algorithm have been used so far. While the extended Infomax algorithm is expected to perform slightly better on real data due to its adaptive nature, FastICA does not depend on educated guesses about the probability density distribution of the unknown source signals. However, Esposito et al show in [12] that both algorithms produce comparable results.

3.3 A fMRI Example: Wordprocessing Task Experiment

First, I present the implementation of the used algorithm used. Then an example of an event-designed experiment will be discussed and its BSS based analysis where it was possible to identify a network of brain areas which could not be detected using classic regression methods.

3.3.1 Applied spatial ICA

To implement spatial ICA for fMRI data, every three-dimensional fMRI image is considered as a single mixture of underlying independent components. The rows of every image matrix have to be concatenated to a single row-vector and with these image-vectors the mixture matrix $\mathbf{X}$ is constructed.

For FastICA the second order correlation in the data has to be eliminated by a "whitening" preprocessing. This is done using a principal component analysis (PCA) step prior to the FastICA algorithm. In this step a data reduction can be applied by omitting principal components (PC) with a low variance in the signal reconstruction process. However, this should be handled with care as valuable high order statistical information can be contained in these low variance PCs. The maximal variations in the timetrends of the supposed word-detection ICs in our example account only for 0.7 % of the measured fMRI Signal.

The FastICA algorithm calculates the de-mixing matrix $\mathbf{W} = \mathbf{A}^{-1}$. Then the underlying sources $\mathbf{S}$ can be reconstructed as well as the original mixing matrix $\mathbf{A}$. The columns of $\mathbf{A}$ represent the time-courses of the underlying sources which are contained in the rows of $\mathbf{S}$. To display the ICs the rows of $\mathbf{S}$ have to be converted back to three-dimensional image matrixes.

As noted before because of the high noise present in fMRI data the ICA problem will always be under-determined or over-complete. As FastICA cannot separate more

components than the number of mixtures available, the resulting IC will always be composed of a noise part and the "real" IC superimposed on that noise. This can be compensated by individually de-noising the IC. In correspondence with Dr. Karsten Specht[1], who conducted and designed the fMRI experiment, it was found from empirical evidence that to be considered a noise signal the value has to be below 10 times the mean variance in the IC which corresponds to a standard deviation of about 3. [42]

3.3.2 Example: Analysis of an event-based experiment

This experiment was part of a study to investigate the network involved in the perception of speech and the decoding of auditory speech stimuli. Therefor one- and two-syllable words were divided into several frequency-bands and then rearranged randomly to obtain a set of auditory stimuli. The set consisted of four different types of stimuli, containing 1, 2, 3 or 4 frequency bands (FB1–FB4) respectively. Only FB4 was perceivable as words.

During the functional imaging session these stimuli were presented pseudo-randomized to 5 subjects, according to the rules of a stochastic event-related paradigm. The task of the subjects was to press a button as soon as they were sure that they had just recognized a word in the sound presented. It was expected that in case of FB4 these four types of stimuli activate different areas of the auditory system as well as the superior temporal sulcus in the left hemisphere [41].

Prior to the statistical analysis the fMRI data were pre-processed with the SPM2 toolbox [29]. A slice-timing procedure was performed, movements corrected, the resulting images were normalized into a stereotactical standard space (defined by a template from the Montreal Neurological Institute) and smoothed with a gaussian kernel to increase the signal-to-noise ratio.

[1]Department of Biological and Medical Psychology and the Bergen Mental Health Center, University of Bergen, Norway

Classical fixed-effect analysis

First, a classic regression analysis with SPM2 was applied. No substantial differences in the activation of the auditory cortex apart from an overall increase of activity with ascending number of frequency bands was found in three subjects. One subject showed no correlated activity at all, two only had marginal activity located in the auditory cortex (figure 3.5 (c)). Only one subject showed obvious differences between FB1 and FB4: an activation of the left supplementary motor area, the cingulate gyrus and an increased size of active area in the left auditory cortex for FB4 (figure 3.5 (a),(b)).

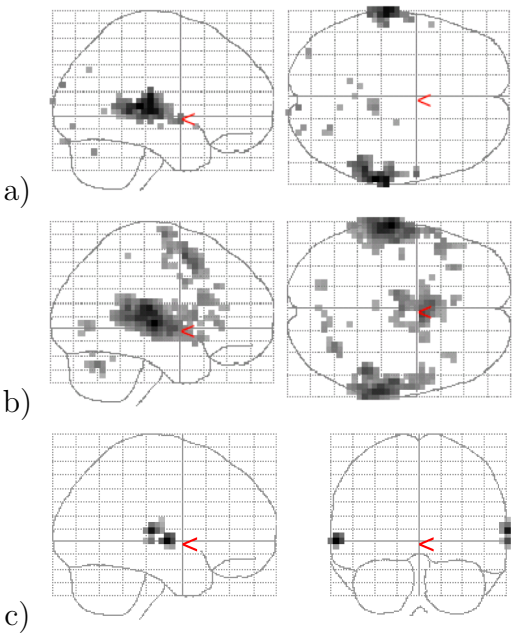


Figure 3.5: Fixed-effect analysis of the experimental data. No substantial differences between the activation in the auditory cortex correlated to (a) FB1 and (b) FB4 can be seen. (c) shows the analysis for FB4 of a different subject.

Spatial ICA with FastICA

For the sICA with FastICA [20] up to 351 three-dimensional images of the fMRI sessions were interpreted as separate mixtures of the unknown spatial independent activity signals. Because of the high computational demand each subject was analyzed individ-

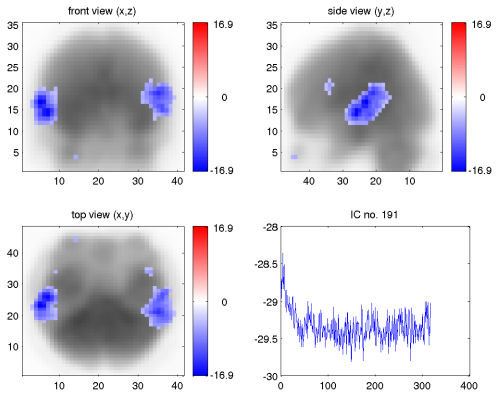


Figure 3.6: Independent component located in the auditory cortex and its time course.

ually instead of a whole group ICA as proposed in [11]. A principal component analysis (PCA) was applied to whiten the data. 340 components of this PCA were retained that correspond to more than 99.999% of the original signals. This is still 100 times greater than the share of ICs like that shown in figure 3.7 on the fMRI signal. In one case only 317 fMRI images were measured and all resulting 317 PCA components were retained.

Then the stabilized version of the FastICA algorithm was applied using tanh as non-linearity. The resulting 340 (resp. 317) spatially independent components (IC) were sorted into different classes depending on their structural localization within the brain. Various ICs in the region of the auditory cortex could be identified in all subjects, figure 3.6 showing one example. It should be noted that all brain images in this chapter are flipped, i.e. the left hemisphere appears on the right side of the picture. To calculate the contribution of the displayed ICs to the observed fMRI data the value of its voxels has to be multiplied with the time course of its activation for each scan (lower subplot to the right of each IC plot). Also a component located at the position of the supplementary motor area (SMA) could be found in all subjects.

The most interesting finding was an IC which represents a network of three simultaneously active areas in the inferior frontal gyrus (figure 3.7) in one subject. This network was suggested to be a center for the perception of speech in [41]. Figure 3.8 shows an

IC (of the same subject) that is assumed to be a network for the decision to press the button [42]. All other subjects except one had ICs that correspond to these networks, although often separated into different components. The time course of both components matches visually very well (figure 3.9) while their correlation coefficient remains rather low ($k_{corr} = 0.36$), apparently due to temporary time- and baseline-shifts.

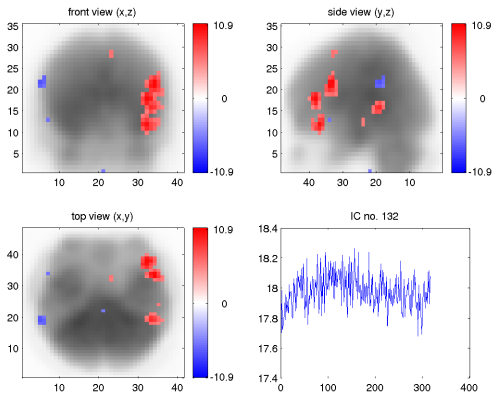


Figure 3.7: Independent component which correspond to a proposed subsystem for word detection.

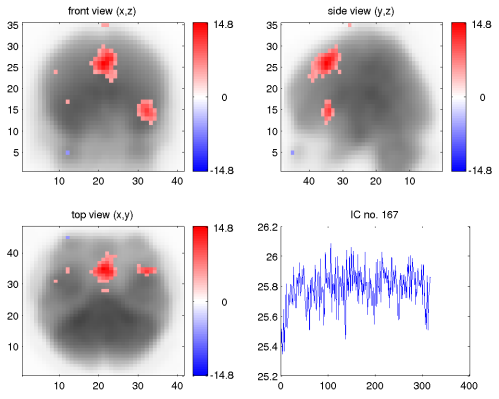


Figure 3.8: Independent component with activation in Broca's area (speech motor area).

Comparison of the regression analysis versus ICA

To compare the results of the fixed-effect analysis with the results of the ICA the correlation coefficients between the expected time-trends of the fixed-effect analysis and the time-trends of the ICs were calculated. No substantial correlation was found: 87 % of all these coefficients were in the range of -0.1 to 0.1, the highest coefficient found being 0.36 for an IC within the auditory cortex (figure 3.6). The correlation coefficients for the proposed word detection network (figure 3.7) were 0.14, 0.08, 0.19 and 0.18 for FB1–FB4. Therefor it is quite obvious that this network of areas in the inferior frontal gyrus cannot be detected with a classic fixed-effect regression analysis.

While the reasons for the differences between the activation-trends of the ICs and the assumed time-trends are still subject to on-going research, it can be expected that the results of this ICA will help to gain further information about the work flow of the brain concerning the task of word detection.

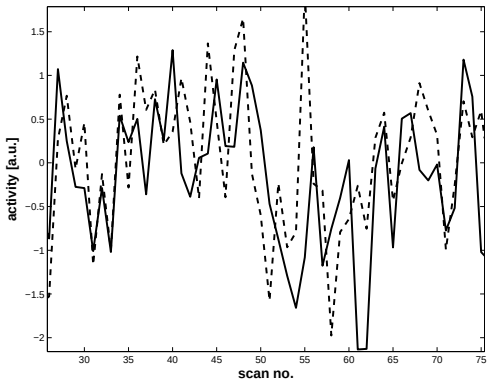


Figure 3.9: The activation of the ICs shown in figure 3.7 (dotted) and 3.8 (solid), plotted for scan no. 25–75. While these time-trends obviously appear to be correlated, their correlation coefficient remains very low due to temporary baseline- and time-shifts in the trends.

3.4 Conclusion and Open Problems

In this chapter I have shown that ICA can be a valuable tool to detect hidden or suspected links and activity in the brain that cannot be found using the classical approach of a model-based analysis like the general linear model. While clearly ICA cannot be used to validate a model (being in itself model-free), it can give useful hints to understand the internal organization of the brain and help to develop new models and study designs which then can be validated using a classic regression analysis.

However, various practical problems hinder ICA on its way to become an universal analysis tool for the brain research:

- manual work is necessary to find the interesting components
- the reliability of the components has to be tested

4 Incomplete ICA

In data processing one of the main questions is finding the hidden models that generate the signals. This information can then be used to detect the fundamental characteristics of the data or to compress the data without loosing important information.

In the investigation of this problem the research has focused on the human visual system (HVS) as one of the most efficient systems in this area. [34, 38] It not only compresses the data on the way from the eyes to the primary visual cortex highly sparse but also is able to detect borders, regions, textures and whole objects fast and secure.

Clustering of data seems to occur very naturally during data processing in the human brain. Objects consists usually of many different edges and textures, yet the brain manages to cluster all this regions into one separate object. Bell *et al* [6] showed that the filters which are produced by analyzing natural scenes with Independent Component Analysis (ICA) are similar to filters which are found in the visual cortex of animals. This leads to the idea that the visual cortex indeed applies some kind of ICA to learn these filters. Usually, a natural scene contains much more independent sources than perceived objects. In this article we concentrate on this question and show that an incomplete ICA will automatically perform clustering based on the appearance of the independent components in the mixtures.

In the following section we will show that clustering with ICA is related to dimension reduction of the data space. Based on this idea we develop a clustering algorithm.

4.1 Theory of Incomplete Independent Component Analysis

ICA can be used to solve the "blind source separation" (BSS) problem. It tries to separate a mixture of originally statistically independent components based on higher order statistical properties of these components:

4.1.1 The ICA Model

Let $s_1(t), \ldots, s_(t)$ be m independent signals with unit variance for simplicity, represented by a vector $\vec{s}(t) = (s_1(t), \ldots, s_m(t))^T$, where T denotes the transpose. Let the mixing matrix $\mathbf{A}$ generate n linear mixtures $\vec{x}(t) = (x_1(t), \ldots, x_n(t))^T$ from these source signals according to:

$$\vec{x}(t) = \mathbf{A}\vec{s}(t) \tag{4.1}$$

(Note that each column of the mixing matrix $\mathbf{A}$ represents the contribution of one source to each mixture.) Assume that only the mixtures $\vec{x}(t)$ can be observed. Then ICA is the task to recover the original sources $\vec{s}(t)$ along with the mixing matrix $\mathbf{A}$. For the complete case $n = m$ many algorithms exist to tackle this problem, e.g. Infomax (based on entropy maximization [5]) and FastICA (based on negentropy using fix-point iteration [20]), just to mention some of the most popular ones. The other cases like the more difficult overcomplete ($n < m$) and the more trivial undercomplete ($n > m$) case have also been widely studied in the literature, see e.g. [2, 44].

4.1.2 The Incomplete Case

In this chaper I concentrate on the incomplete case: What will happen if one tries to extract deliberately fewer sources than can be extracted from the mixtures $\vec{x}(t)$? We do not want to extract a subset of all independent sources, instead we try to cluster

all sources into fewer components than could be extracted in principle. In this way the incomplete case differs from the overcomplete case.

At the same time the incomplete case obviously makes a dimension reduction of the data set necessary. A common way to keep the loss of information minimal is to apply a principal component analysis to the mixtures $\vec{x}(t)$ and to do the dimension reduction based on the eigenvectors $\vec{e}_i$ corresponding to the smallest eigenvalues λ_i of the data covariance matrix $\mathbf{C}$ [22]. This is also a basic pre-processing step ("whitening") for many ICA algorithms (and can be done quite efficiently with neural networks), as it reduces the degrees of freedom in the space of the solutions by removing all second order correlations of the data and setting the variances to unity:

$$\vec{\tilde{x}} = \mathbf{E}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^T\vec{x}, \tag{4.2}$$

where $\mathbf{E}$ is the orthogonal matrix of eigenvectors of the covariance matrix of $\vec{x}$, with $\mathbf{C}(\vec{x}) = \mathrm{E}((\vec{x} - \mathrm{E}(\vec{x}))(\vec{x} - \mathrm{E}(\vec{x}))^T)$, and $\mathbf{\Lambda}$ the diagonal matrix of its eigenvalues.

This dimension reduction will cluster the independent components (IC) $s_i(t)$ based on their presence in the mixing matrix $\mathbf{A}$, as the covariance matrix of $\vec{x}$ depends on $\mathbf{A}$: [8]

$$\begin{aligned} \mathrm{E}(\vec{x}\vec{x}^T) &= \mathrm{E}(\mathbf{A}\vec{s}\vec{s}^T\mathbf{A}^T) && (4.3)\\ &= \mathbf{A}\mathrm{E}(\vec{s}\vec{s}^T)\mathbf{A}^T && (4.4)\\ &= \mathbf{A}\mathbf{A}^T && (4.5) \end{aligned}$$

If two columns in the mixing matrix $\mathbf{A}$ are almost identical up to a linear factor, i.e. are linearly dependent, this means that the two sources represented by those columns are almost identically represented (up to a linear factor) in the mixtures. A matrix with two linearly dependent columns does not have full rank, hence will have at least one zero eigenvalue due to its restricted dimension.

This also holds for the transpose $\mathbf{A}^T$ of the matrix $\mathbf{A}$ as the transpose has the same dimensionality as the original matrix, as well as for the product of both matrixes $\mathbf{A}\mathbf{A}^T$.

Setting this close-to-zero eigenvalue to zero in the course of a dimension reduction will thus combine two almost identical columns of $\mathbf{A}$ to a single one. This means that components that appear to be similar to each other in most of the mixtures will be clustered together into new components by the dimension reduction with PCA.

Another possibility is to use only parts of the original data set. This also will cause the ICA to form clusters of independent components with similar columns in the mixing matrix in the reduced data set.

4.1.3 Clustering with Incomplete ICA

For ICA the literature on clustering so far is based on the comparison of the independent components themselves. To name just a few published algorithms for this problem: the tree-dependent ICA [3] and the topographic ICA [21]

As shown in the section before, clustering based on the columns of the mixing matrix comes naturally to incomplete ICA. However, normally ICA is applied to find the basic independent components of the original data. So a second step is necessary to get from the clusters to their separate parts: The idea is to compare different ICA runs with a different level of dimension reduction applied beforehand. First a complete ICA is performed extracting the maximal number of independent components (ICs) from the data set. In a second run, an incomplete ICA is performed on a reduced data set which resulted from a dimension reduction during PCA pre-processing.

The independent components of the complete ICA without dimension reduction are then compared with the IC of several incomplete ICA runs. Independent components which form part of the components of the incomplete ICA are then grouped into the cluster which is represented by the IC of the incomplete ICA at hand. Hence the ICs of

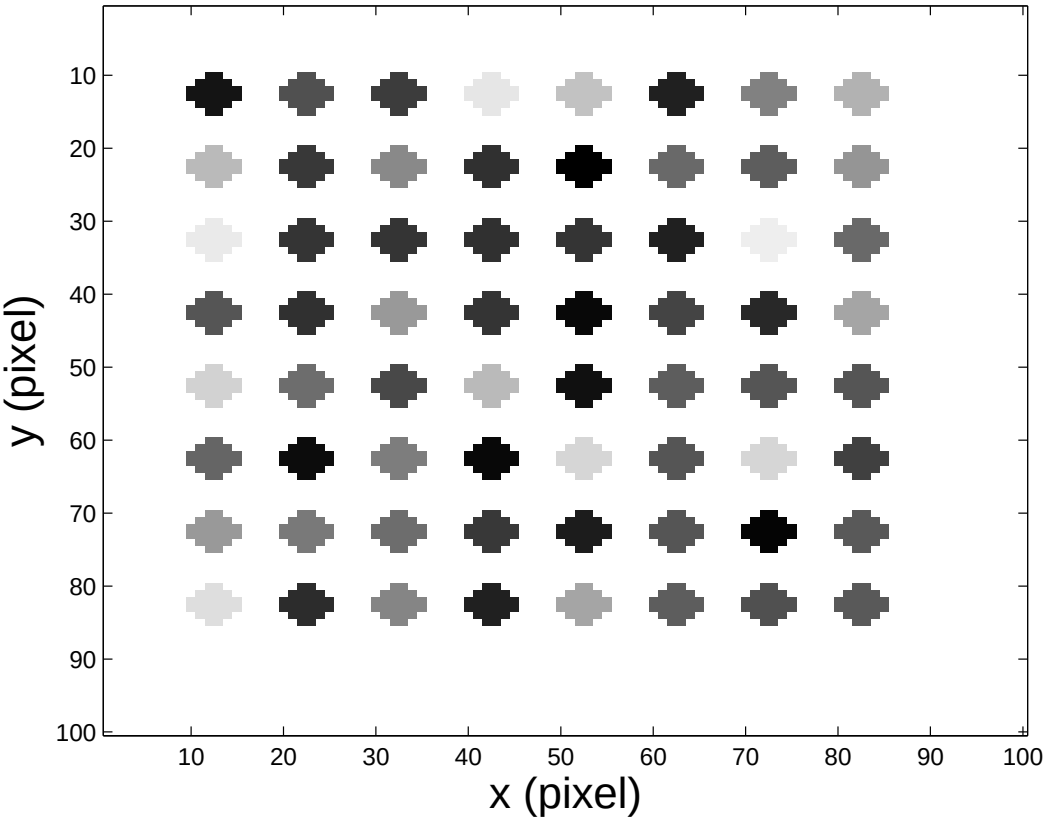


Figure 4.1: The figure shows one mixture of the toy data set. Note the almost undetectable A-set in the upper left corner and the C-set in the lower right corner.

any incomplete ICA form sort of prototype ICs of the clusters formed by ICs from the complete set. This leads immediately to an algorithm for clustering by incomplete ICA:

1. apply a standard ICA to the data set without dimension reduction: ICA1
2. apply a standard ICA to the data set with dimension reduction: ICA2
3. find the independent components in ICA1 that are similar to some forms of components in ICA2 for a further analysis of the independent components.
4. (optionally) repeat step 2 and 3 until the interesting clusters are found

As can be seen in our toy example in the next section, the actual size of the dimension reduction in step 2 is not critical, the algorithm seems to work well within a range of adequate numbers.

4.2 Examples

In this section I will demonstrate with a toy example that the clustering with incomplete ICA outperforms a standard *k-means* clustering algorithm. Then I will demonstrate that incomplete ICA will cluster the response of a filter to a set of images in a way that the results can be used to detect objects.

4.2.1 Toy Data

To test the quality of the clustering I chose to build a toy data set. 64 sources were created where each of the sources represents a circle in a square lattice of 100×100 lattice points. The mixing matrix was initialized randomly and then modified so that two sets of circles – one representing the letter "A" in the upper left corner, the other representing the letter "C" in the lower right corner – appeared together in the mixtures

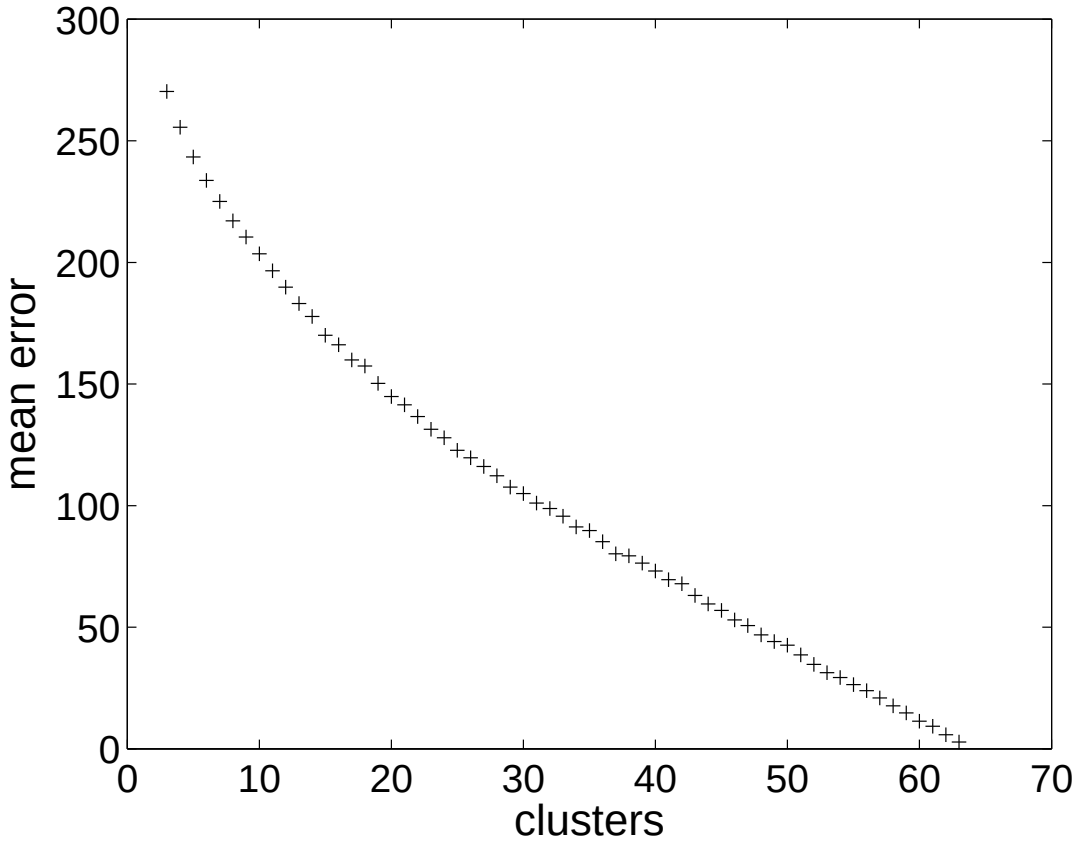


Figure 4.2: The figure on the right the overall mean error against the number of clusters that were used for the k-means analysis.

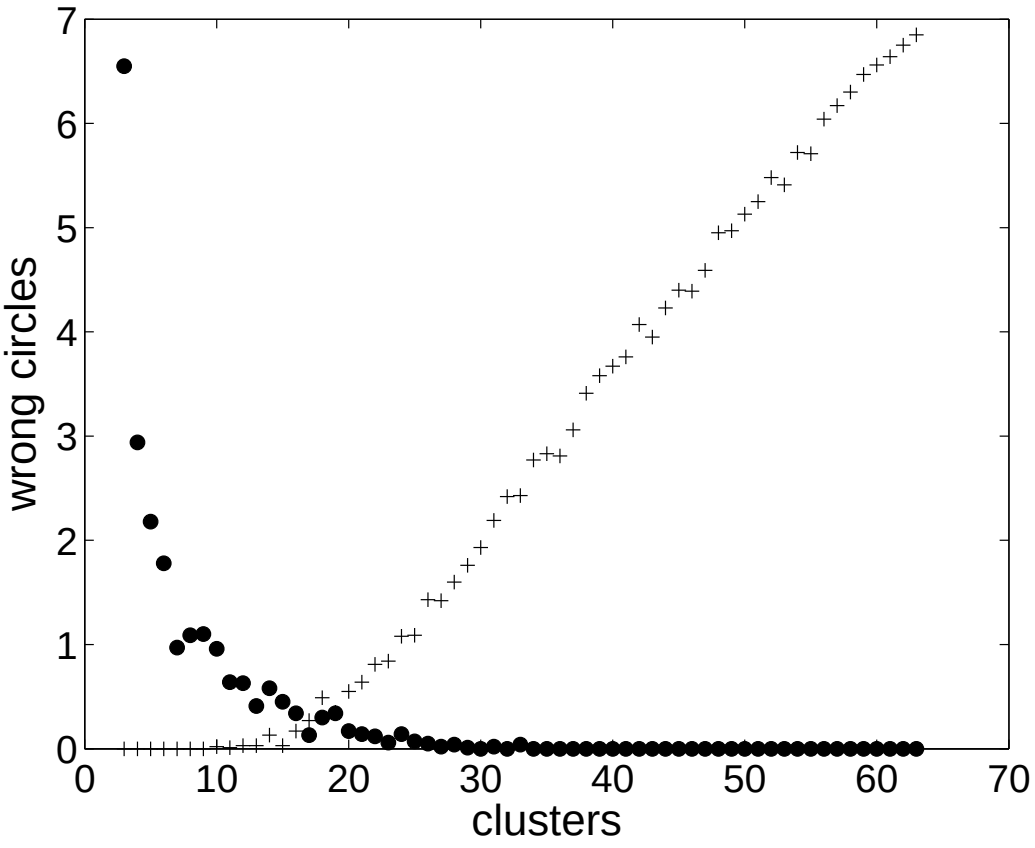


Figure 4.3: In the figure the number of wrong circles in the A-cluster (dots) and the number of A-circles in the wrong cluster (crosses) are plotted against the number of clusters. The *k-means* algorithm was used

by setting the columns of these sets to the same values, with random differences of up to 5%. Figure 4.1 shows one of these mixtures.

K-means analysis

A standard *k-means* clustering analysis was performed to cluster the columns of the mixing matrix $\mathbf{A}$. Figure 4.2 shows the mean overall error of 100 *k-means* analysis runs for 3 up to 63 clusters. It can be seen that in this case this statistic gives no hint on the number of clusters in the data set.

In figures 4.3 and 4.4 the mean number of wrong circles is plotted, in figure 4.3 for the

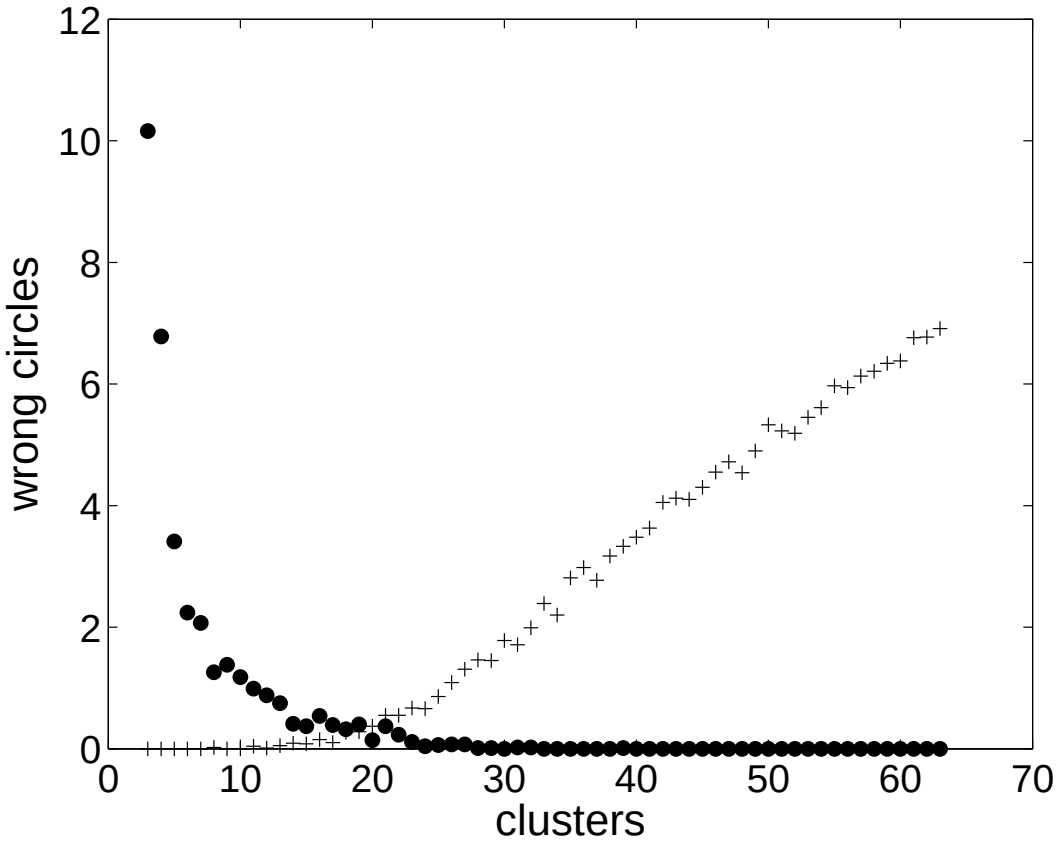


Figure 4.4: In the figure the number of wrong circles in the C-cluster (dots) and the number of C-circles in the wrong cluster (crosses) are plotted against the number of clusters. The *k-means* algorithm was used.

A-class circles, in figure 4.4 for the C-class circles. While the k-means analysis obviously clusters all the A-class circles in one cluster up to an overall number of searched-for clusters of 10, it fails to do so with an average error of almost 1 not-A circle. For more than 20 clusters this error disappears, but at the same time A-class circles appear in other clusters. The results for the C-class circles are practically the same as can be seen in figure 4.4.

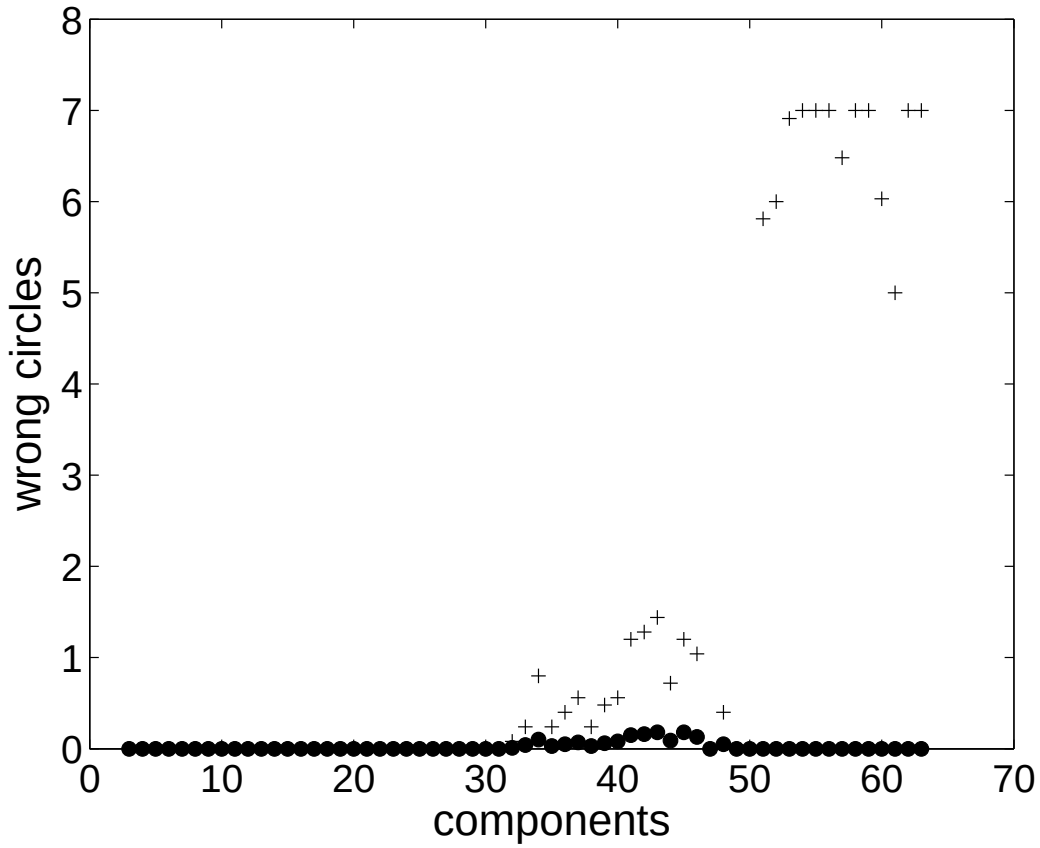


Figure 4.5: In the plot the number of wrong circles in the A-cluster (dots) and the number of A-circles in the wrong cluster (crosses) are plotted against the number of components for clustering with the incomplete ICA using the PCA dimension reduction.

Incomplete ICA

For this analysis the *FastICA* algorithm [20] was used. 3 up to 63 components (data reduction via a preceding PCA or via a random pick of the mixtures from the original data) were estimated in 100 runs for each analysis. As the ICA first had to de-mix the mixture of circles, a simple de-noising of the resulting components was necessary (every pixel with a level of 70% was counted as activated).

On figure 4.5 the plot for the number of falsely classified circles shows that the algorithm using PCA for dimension reduction worked well for 3 up to 31 used components,

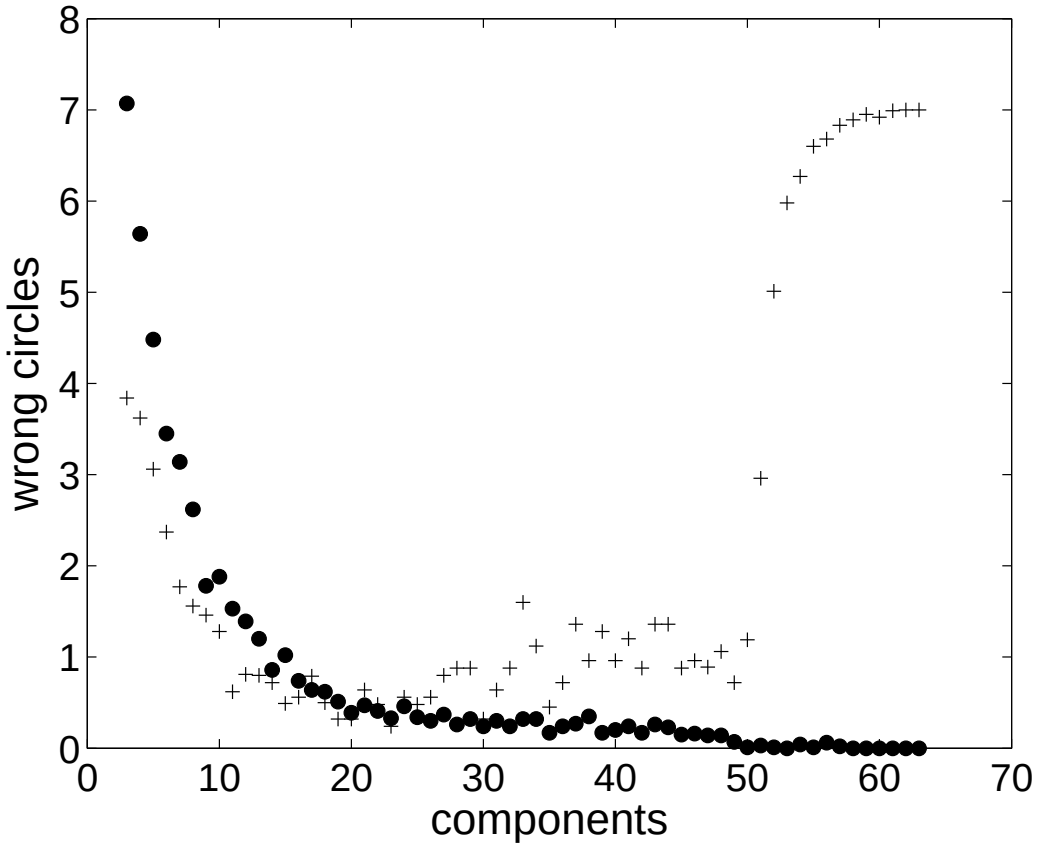


Figure 4.6: Here the number of wrong circles in the A-cluster (dots) and the number of A-circles in the wrong cluster (crosses) are plotted against the number of components for clustering with the incomplete ICA for the dimension reduction by randomly picking mixtures of the data set.

thus being remarkably stable. For more than 52 components the ICA separated the single circles of the A-Cluster, as can be expected due to the setting of the data. The incomplete ICA also was able to find the right column of the mixing matrix for the class-A circles for all used components in every run.

Figure 4.6 shows the results for the algorithm using a random subset of the original data set for dimension reduction. Obviously the algorithm here had problems in clustering the correct circles together. Also the ICA here was not always successful in finding the correct column of the mixing matrix for the A-class circles. It seems that the simple dimension reduction via the random picking of only some of the original mixtures erases too much of the information, so that the algorithm can not work properly anymore.

Object detection

Incomplete ICA can be used to detect objects in small movies: while each object typically consists of many separate independent components, these ICs will appear together in the result of the incomplete ICA because they also appear together in the mixtures. To demonstrate this we took 4 pictures of a palm tree in slightly different angles. This mimics the view of a passing observer (figure 4.7). Then a filter for vertical edges was applied (see figure 4.8 for an example). After this an incomplete ICA was used to separate the resulting filter responses of the four images.

As written before it can be expected that this ICA will show at least one IC with the highest values for that part of the image that represents an object within the filter responses in the image. At the same time this IC should have the overall highest values in his column of the mixing matrix as it should be the IC that consistently appears in the images.

Figure 4.9 shows the IC that showed the highest values in the columns of the mixing matrix. The trunk of the palm tree in the middle here clearly shows the highest values and thus can be marked as “object”.

Figure 4.7: The four black and white images that were used for the object detection test. each time the palm tree is in the centre of the image, but the angle of the view differs slightly.

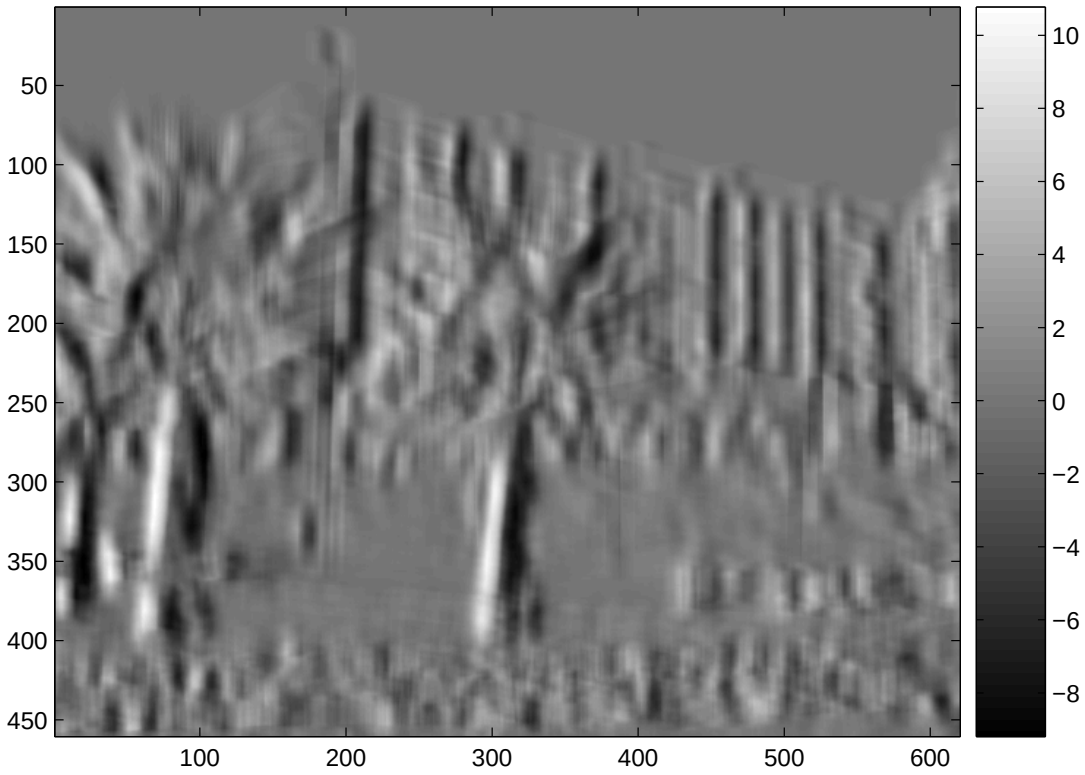


Figure 4.8: The filter response for one of the images. As expected the filter for vertical edges will have the highest results for the trunks of the trees and the buildings in the background.

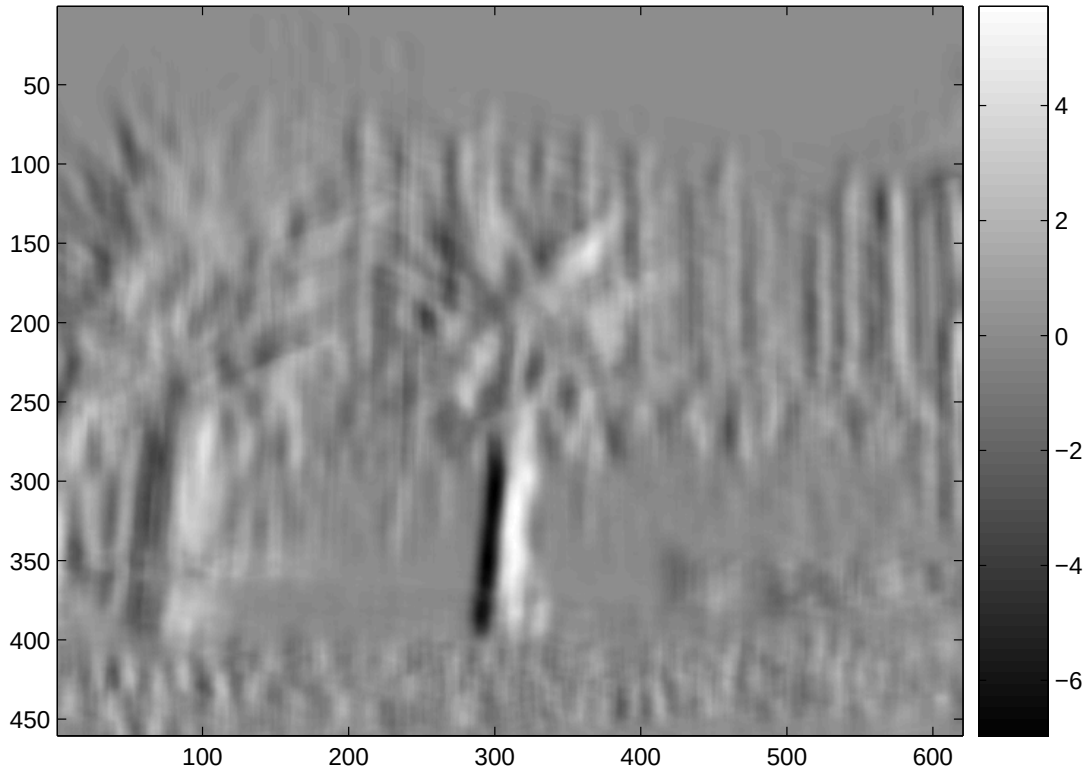


Figure 4.9: The component with the highest value in the incomplete ICA. The trunk of the palm tree was detected as object.

Combined with the results of other filters that are able to detect other parts of objects, the clustering feature of an incomplete ICA could be used to build a sophisticated object detection system.

4.3 Application on fMRI Data

Functional magnetic resonance imaging (fMRI) is one of the leading technologies for functional human brain research due to its high spatial resolution and noninvasiveness. fMRI also is a grateful example to utilize spatial independent component analysis (sICA) as the functional segregation of the brain (see [15]) closely matches the requirement of statistical spatial independence. However, a problem in the case of fMRI still is the mass of data that has to be analyzed to find interesting components: Each fMRI session will typically yield hundreds of different components.

Clustering can be part of the solution to this problem, as the regions of the brain that work together during the experiment also will form clusters of activation in the time domain that can be exploited to cluster the components that represent the collaborating parts of the brain.

Surprisingly, for independent component analysis the literature so far concentrates on the comparison of the independent components (ICs) themselves. Two recent published algorithms for these problem are the tree-dependent ICA by [3] and the topographic ICA by [21]. In tree-dependent ICA the assumption of independence is weakened and a transformation of the data into a tree of independent clusters of dependent sources is searched. In topographic ICA also the resulting components do not have to be completely independent: the variances corresponding to neighboring components have to be positively correlated while the other variances remain independent. Both algorithms have been applied to fMRI data by [30] with varying results.

However, in the search for cooperating networks of brain areas in fMRI not the independent components itself need to be similar, but the time-courses of their activations, i.e. their columns in the mixing matrix estimated by the ICA. In this article therefore we demonstrate a new algorithm that, instead of comparing the components itself, will cluster the components dependent on their appearance in the mixtures. For the fMRI

case this means that activation-maps that have similar time-courses will be clustered together by this algorithm.

4.3.1 spatial Independent Component Analysis

Let $s_1(t), \ldots, s_(t)$ be m independent signals with unit variance for simplicity, represented by a vector $\vec{s}(t) = (s_1(t), \ldots, s_m(t))^T$, where T denotes the transpose. Let the mixing matrix $\mathbf{A}$ generate n linear mixtures $\vec{x}(t) = (x_1(t), \ldots, x_n(t))^T$ from these source signals according to:

$$\vec{x}(t) = \mathbf{A}\vec{s}(t) \tag{4.6}$$

Note that each column of the mixing matrix $\mathbf{A}$ then represents the contribution of one source to each mixture.

Assume that only the mixtures $\vec{x}(t)$ can be observed. Then the task to recover the original sources $\vec{s}(t)$ along with the mixing matrix $\mathbf{A}$ is commonly referred to as "independent component analysis" (ICA). If the mixtures represent spatial vectors of data points, e.g. each $\vec{x}(t)$ one image of a fMRI session, this ICA is called "spatial" and the estimated sources will be spatial independent.

For the complete case $n = m$ (i.e. as many mixtures as sources) many algorithms exist to tackle this problem, e.g. Infomax (based on entropy maximization, see [5] for details) and FastICA (based on negentropy using fix-point iteration, see [20] for details), just to mention some of the most popular ones. The other cases like the more difficult over-complete ($n < m$) and the more trivial under-complete ($n > m$) case have also been widely studied in the literature, see e.g. [2, 44].

4.3.2 Clustering with incomplete ICA

The idea behind the clustering with a intentionally incomplete ICA is to compare different ICA runs with a different level of dimension reduction applied beforehand. First a

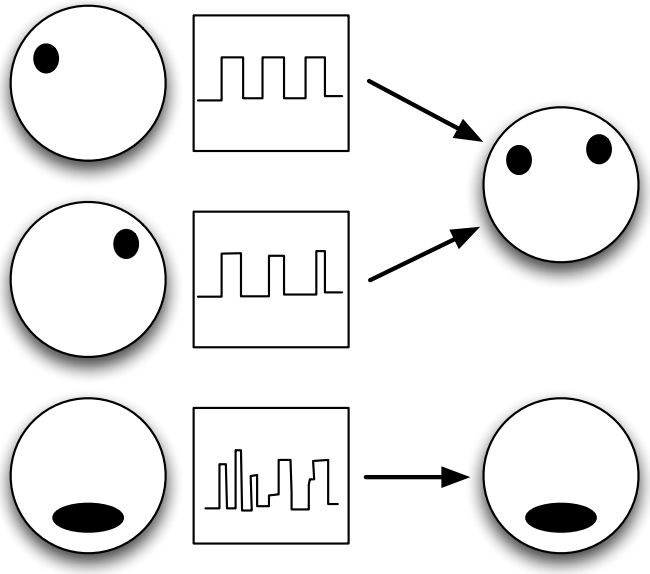


Figure 4.10: incomplete ICA: If the ICA is forced to interpret fewer sources into the data than are existent in the data then the algorithm will cluster together independent components that share similar columns (i.e. time-courses in the fMRI case) in the mixing matrix (above).

complete ICA is performed extracting the maximal number of independent components from the data set. In a second run, an incomplete ICA is performed on a reduced data set which resulted from a dimension reduction during PCA preprocessing.

The independent components (ICs) of the complete ICA without dimension reduction are then compared to the ICs of several incomplete ICA runs. Independent components which form part of the components of the incomplete ICA are then grouped into the cluster which is represented by the IC of the incomplete ICA at hand. Hence the ICs of any incomplete ICA form sort of prototype ICs of the clusters formed by ICs from the complete set. Figure 4.10 shows a schematic example.

4.3.3 fMRI workflow

The workflow for a fMRI analysis using this clustering approach will be as follows:

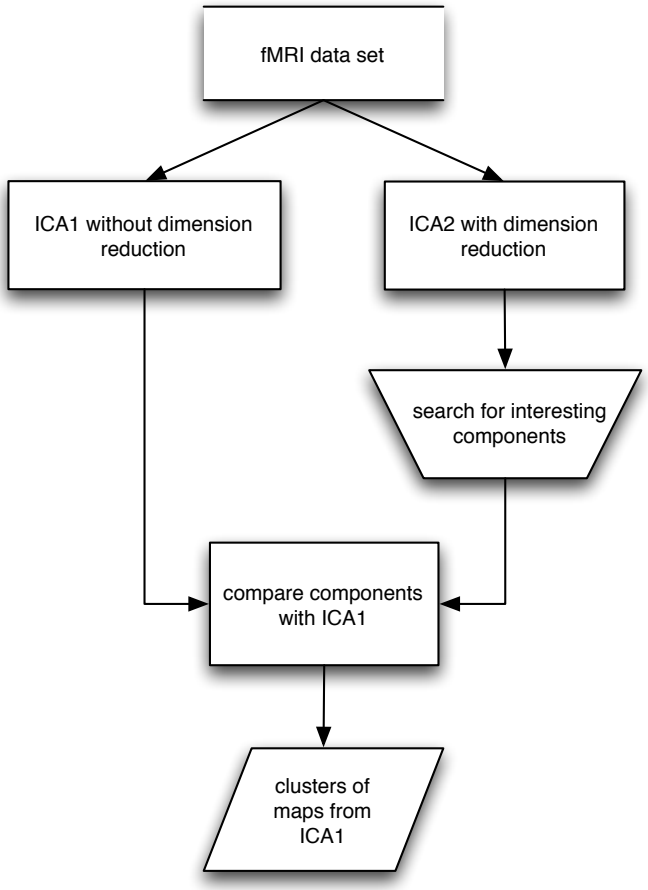


Figure 4.11: The workflow of a fMRI analysis using incomplete ICA for clustering

1. apply a standard ICA to the dataset without dimension reduction: ICA1
2. apply a standard ICA to the dataset with dimension reduction: ICA2
3. search for interesting "cluster" components in ICA2
4. repeat (2) and (3) for different levels of dimension reduction in ICA2 to find the best results
5. find the independent components in ICA1 that are similar to some forms of components in ICA2. These components and their time courses can now be analyzed further within the framework of normal brain research to understand their connectivity and collaboration.

Figure 4.11 shows a diagram of this workflow.

Due to the data reduction the components will be altered within the incomplete ICA, so for the interpretation of the dataset the results from the complete ICA1 should be used.

4.4 Analysis of a WCST fMRI example

The clustering algorithm was then applied to fMRI data of a modified *Wisconsin Card Sorting Test* of one subject. The subject has the task to sort subsequently presented cards of symbols with respect to an attribute, like color, shape of the symbol or number of displayed symbols. In the beginning, the subject does not know the sorting role and has to figure it out by trial and error. The test was modified in a way that the subject also had to look for the spatial position of the symbol. Control blocks were introduced where the subject did know in advance the attribute that was searched in the test.

The analyzed data set consisting of 467 scans was created within the projekt ModKog[1]

[1]financed by the German Bundesminsterium für Bildung und Forschung

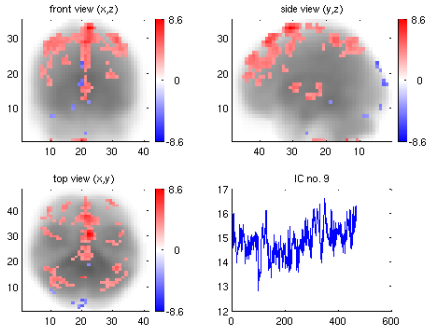


Figure 4.12: The searched-for activation pattern resulting from ICA2 for 10 dimensions. The images appear flipped. On the lower right corner the time course of this activation is displayed.

in the group of Prof. Dr. med. Gereon R. Fink at the institute for medicine at the Research Center Jülich, Germany, preprocessed to remove motion artefacts, normalized and filtered with a gaussian filter to increase the signal to noise ratio.

Spatial ICA was used for the analysis so that the independent components correspond to activation maps and the columns of the mixing matrix correspond to the time courses of this activation (for more information on ICA of fMRI data see e.g. [27]).

For the clustering with the incomplete ICA the data was first reduced via PCA to 450 dimensions, so that almost all information was retained. Then the (spatial) ICA1 was calculated using the *extended Infomax* algorithm.

For the (spatial) ICA multiple runs were made using different levels of dimension reduction. The results of these runs then were manually searched for the activation pattern that is typical for this modified *Wisconsin Card Sorting Test* and has been found earlier through a *classic general linear model* analysis [43]. It consists of a bilateral network of frontal and parietal areas that is relevant for the processing of spatial and object-oriented information and for the direction of attention.

Figure 4.12 shows the activation map that was the result of a reduction to 10 dimensions. The searched-for patterns appear only marginally while the time course (lower right corner of the figure) already shows the frequency of the basic blocks of the experi-

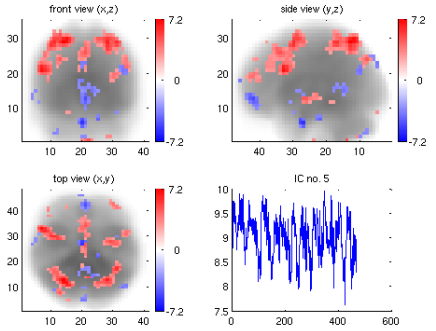


Figure 4.13: The searched-for activation pattern resulting from ICA2 for 20 dimensions.

ment.

Figure 4.13 shows the activation map that was the result of a reduction to 20 dimensions. The searched-for patterns here appear already well formed.

For 40 dimension the ICA splits the activation into two activation maps with different time courses (see Figure 4.14). Obviously the network consists of two sub-networks that are used to process the information.

For 50 dimensions the quality of the result further enhances, as can bee seen in figure 4.15. This effect is expected, as more and more information is available to the analysis and will be used to construct the components.

Then it was searched for the components in ICA1 that correspond to the two activation maps that were found in ICA2 with 50 dimensions. To achieve this the activation maps were compared and looked for components in ICA1 and ICA2 that have an overlap in their maps. Prior to the comparison the components were de-noised: all values of the activation maps that were below 4 times the standard deviation were neglected. This is also the de-noising scheme that was used for the figures in this section.

Figure 4.16 shows a subset of these components of ICA1 that overlap at least 10% with the components of ICA2 as shown in figure 4.15. These components that form the "cluster maps" of ICA2, are now open for a further study of their interplay using their time courses.

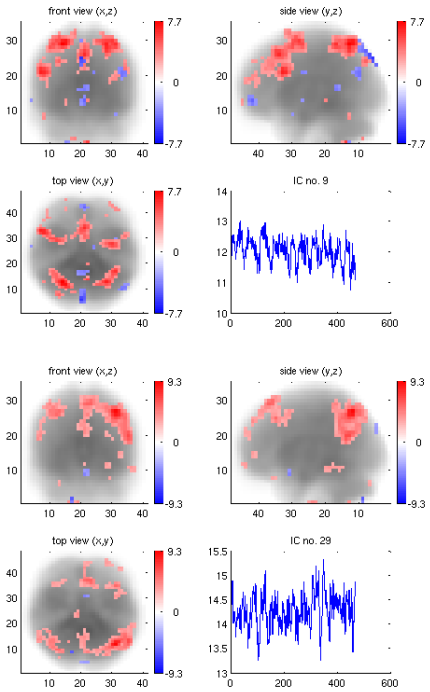


Figure 4.14: The searched-for activation patterns resulting from ICA2 for 40 dimensions. The ICA splits the network into two patterns with roughly similar time courses

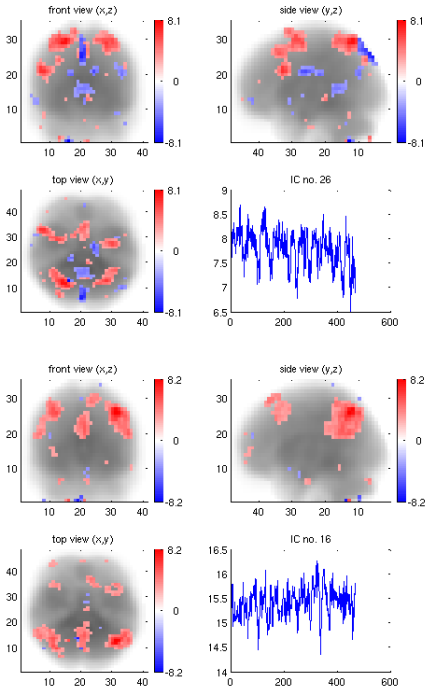


Figure 4.15: The searched-for activation patterns resulting from ICA2 for 50 dimensions.

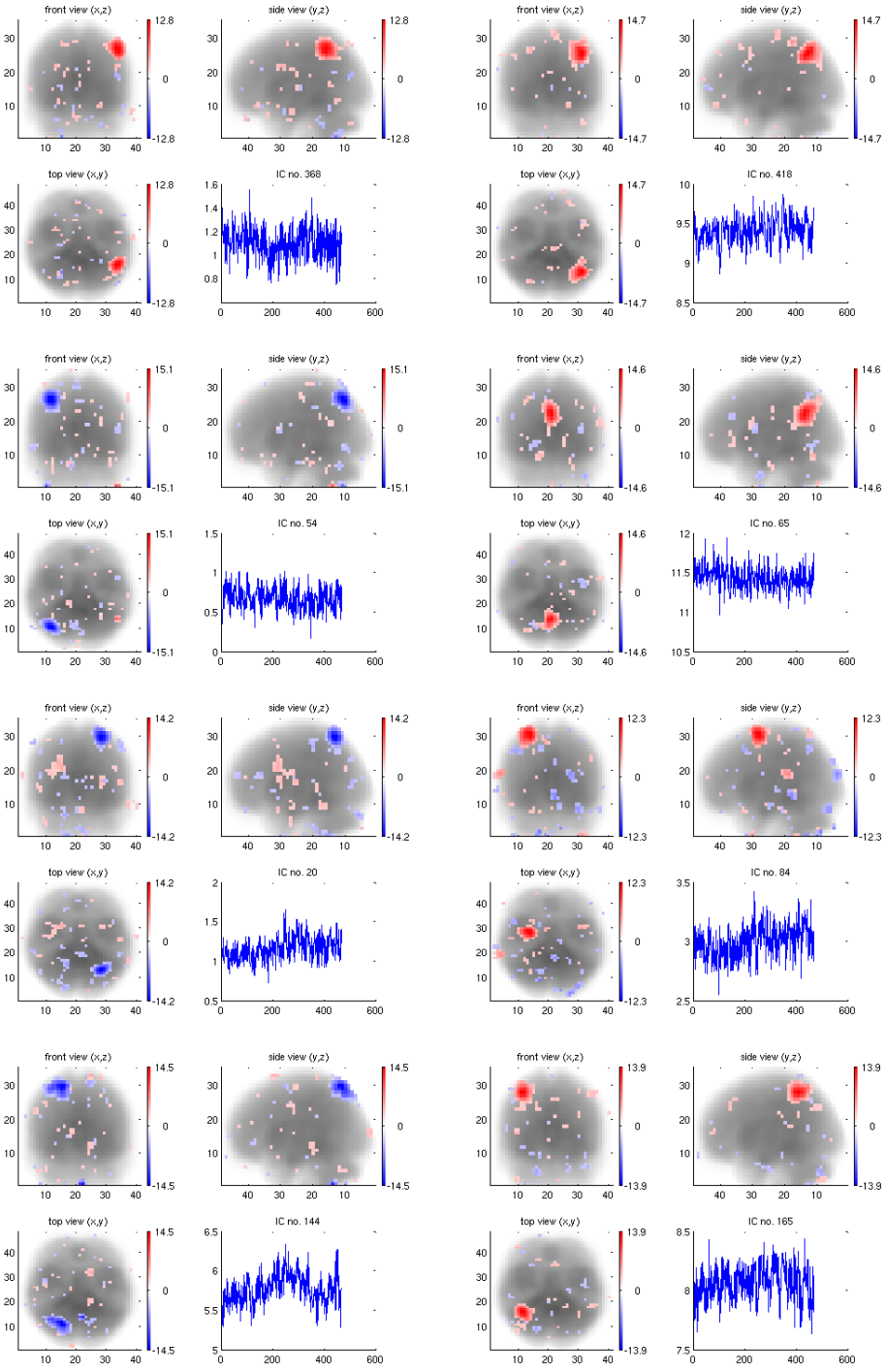


Figure 4.16: A subset of the components of ICA1 that are related to the activation maps shown in figure 4.15.

4.5 Conclusion

Based on the idea that an incomplete ICA will cluster independent components based on their distribution in the mixtures, thus allowing a basic form of object detection, it was possible to demonstrate in this chapter that this feature of ICA can be used to develop a promising new kind of clustering algorithm. It is also interesting to note that this algorithm was able to out-perform *k-means* in a toy example without any problems.

Furthermore it has been shown that by clustering using incomplete ICA it is possible to reduce the necessary manual work of a fMRI analysis with ICA. While this is only a part in the ongoing attempt to enhance the use of independent component analysis in medicine, it can be expected that this work will lead to a better understanding of the interconnectivity within the human brain.

5 Parallel incomplete ICA

Functional magnetic resonance imaging (fMRI, [24,31]) based on the blood oxygen level dependent contrast (BOLD) nowadays is one of the main technologies in human brain research. Using spatial independent component analysis (ICA) to interpret fMRI data (see e.g. [27] for an application) works well, as the functional segregation of the brain closely matches the requirement of statistical spatial independence. The advantage of ICA for the analysis of fMRI data is that it is a model free technique so that it is not necessary to know in advance how the brain will react to the stimulus that is presented to the subject.

Basically, spatial ICA separates the fMRI data set $\mathbf{X}$ into statistically independent sources $\mathbf{S}$ (in the case of fMRI they represent maps of activation in the brain) and their time courses, which are contained in the so called mixing matrix $\mathbf{A}$:

$$\mathbf{X} = \mathbf{AS} \tag{5.1}$$

A severe problem in the case of fMRI is the mass of data that has to be analysed manually to find the interesting components, as each fMRI session will typically yield hundreds of different components. Also, the calculations tend to be time consuming – a single processor desktop computer needs more than one hour to do a complete ICA of a one-subject data set of 400 normal-sized fMRI Images with the *Matlab* implementation of *FastICA* [20].

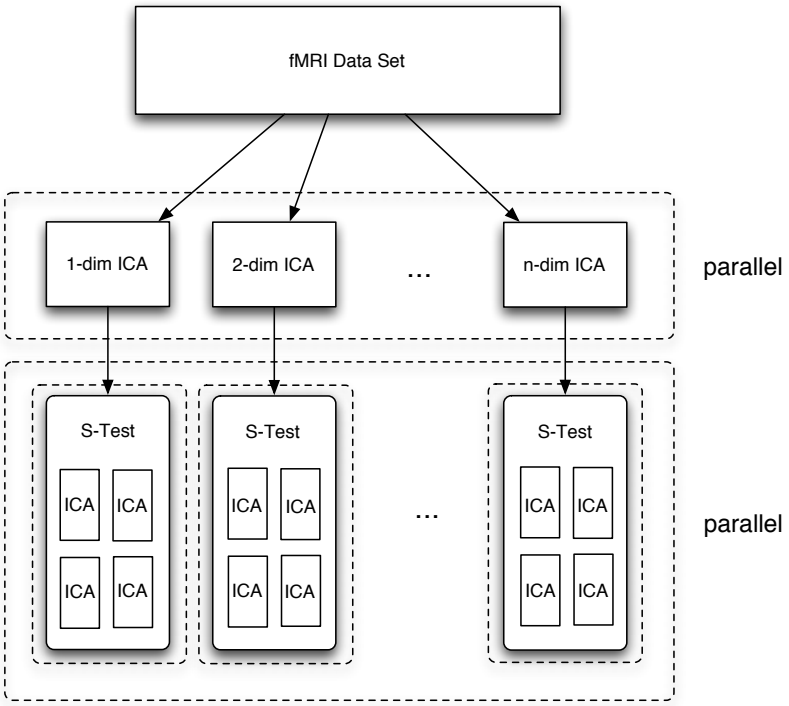


Figure 5.1: Diagram of parallel clustering with incomplete ICA. The dotted boxes represent the parts of the program that can be executed parallel.

The approach to solve these problems is twofold (see figure 5.1 for a diagram of our approach):

1. use multiple, incomplete ICA runs with increasing dimensions that will cluster the activation maps based on their similarity of their time courses in the data set. This way the researcher can start with a few clusters of a low-dim ICA and investigate the separate, localised activation patterns in a high-dim ICA that form this cluster.

2. use multiple, parallel ICA runs to test the reliability of the ICA based on the algorithm by Meinecke et al [28]

In this way it is possible to present a novel method of parallel clustering of the interesting independent components in fMRI data analysis while at the same time minimising the manual work of the brain researcher.

6 Conclusiones y Perspectivas

Como se ha visto a lo largo del desarrollo de esta tesis, la principal ventaja que posee ICA, que es la de ser un "modelo libre" (model-freeness), ha sido aplicada a la investigación con datos fMRI. Los métodos estadísticos de análisis clásicos basados en estadísticas de segundo orden como el GLM permiten la verificación de modelos, pero terminan siendo inútiles si no existe un modelo o si él mismo no está correcto. Igualmente no permite análisis de datos exploratorios. Caso contrario sucede con el método ICA. Esto es especialmente un problema en la investigación de redes cognitivas más altas en el cerebro, pues la activación en esta clase de experimentos tiende mucho para no ser correlacionada al protocolo del experimento y así que encontrar el modelo derecho puede ser casi imposible. Un problema adicional es que las estadísticas de segundo orden como el GLM no demostrarán si el modelo está correcto o no; pero puede entregar resultados falsos si el modelo es incorrecto. Esto pudo ser demostrado en el caso de un experimento realizado con el procesamiento de palabras donde los estadísticos de segundo orden no revelaron ninguna información interesante; mientras que al usar ICA era posible de manera muy simple detectar las dos redes del interés principal en los datos.

El análisis de datos exploratorio conlleva muchos problemas: Una gran cantidad de resultados tienen que ser buscados generalmente de manera manual para componentes interesantes, mientras que al mismo tiempo puede no tenerse una buena estabilidad y confiabilidad en el análisis. En los últimos años en las investigaciones enfocadas en el cerebro la búsqueda para localizar nuevas redes ha ganado más y más interés. Eso es otro

campo de investigación donde las estadísticas de segundo orden no pueden ser utilizadas eficientemente.

En esta tesis he desarrollado una nueva clase de algoritmo cluster basado en la aplicación de ICA en el caso incompleto. Este ICA incompleto reúne las componentes basándose en semejanzas de las columnas de la matriz de mezcla y de esta forma permite reunir para el caso fMRI las activaciones que tienen cursos de tiempos similares. Esto es un paso muy importante en la búsqueda de redes activas en el cerebro porque este tipo de redes demuestran semejanzas o similitudes en los cursos de tiempos pudiendo así, de una manera general ser detectadas. En esta tesis demuestro lo anteriormente dicho con un sistema de datos fMRI de un experimento modificado de la Wisconsin-Card-Sorting-Test. Sin embargo, el uso de este algoritmo cluster no se restringe solamente a datos de fMRI, también fue aplicado con éxito en un juego de datos y utilizado de manera simple para detectar objetos en una película.

Otra característica interesante del método de clusterización con ICA incompleto se presenta en el hecho de que es altamente paralelo. Tal y como he demostrado en esta tesis formular el algoritmo de forma paralela es facil y las ganancias en tiempo de computo son altas.

No obstante, aún quedan problemas abiertos. La búsqueda manual de componentes interesantes podría ser reducida si se pudiera encontrar una forma que permita detectar automáticamente los artefactos que carecen de interés o las componentes comunes que se relacionan con las actividades fundamentales del cuerpo humano y no con las del experimento. Otro interesante campo es la aplicación de ICA paralelo por ejemplo en algoritmos genéticos. Sin embargo, estos campos todavía requieren de más investigación.

En esta tesis he presentado un método innovador con el fin de dar solución a algunos de los problemas que han obstaculizado el éxito de ICA hasta ahora en el campo del análisis de datos del fMRI. El futuro dirá si estos métodos pueden ser aplicados extensamente a

ICA como ahora GLM, método comunmente usado en el análisis para fMRI.

Conclusions and Perspectives

As we have seen in this thesis, the main advantage of ICA applied to fMRI lies in it's model-freeness. While classic analysis methods based on second order statistics like the GLM allow the verification of models, they are useless if no model exists or if the model itself is not correct. Unlike ICA they do not allow exploratory data analysis. This is especially a problem in the investigation of higher cognitive networks in the brain, as the activation in this kind of experiments tends not to be very much correlated to the experiment protocol and so finding the right model can be almost impossible. Further problematic is that second order statistics like the GLM will not show whether the model is correct or not - instead it may give false results if the model is incorrect. I was able to show this in the case of a word processing experiment where the second order analysis did not reveal any interesting information, while using ICA it was possible to detect the two networks of main interest in the data very simply.

The exploratory data analysis however has it's own problems: Usually a great number of results have to be manually searched for interesting components, while at the same time stability and reliability of the analysis may not be very good. At the same time in brain research the search for networks in the brain has gained more and more interest in the recent years, and another field where second order statistic can not be used efficiently.

In this thesis I have developed a new kind of clustering algorithm based on intentionally applying ICA in the incomplete case. This incomplete ICA clusters the components based on similarities in the columns of the mixing matrix and thus will cluster in fMRI

activation that shows similar time-courses. This is a very important step in the search for active networks in the brain because active networks usually show similarities in the time-courses of their parts and thus can be detected very simple with this method. In this thesis I demonstrate this on an fMRI data set of a modified Wisconsin-Card-Sorting-Test experiment. However, the application of this clustering is not restricted to fMRI data alone, it was also applied successfully to a toy data set and used as a simple way to detect objects in a movie.

Another interesting characteristic of the method of clustering by incomplete ICA lies in the fact that it is in itself highly parallel. As I have shown in this thesis formulating the algorithm in a parallel way is straightforward and the resulting gains in terms of used calculation time are quite profound.

Still, open problems exist. The manual search for interesting components could be further reduced if a way could be found to automatically detect uninteresting artefacts or common components that are related to fundamental activities of the human body and not to the experiment. Also it could be quite interesting paralleling ICA itself, for example via genetic algorithm. However, these fields are still subject to heavy research.

In this thesis I have made an innovative attempt in solving some of the problems that have hindered the success of ICA so far in the field of fMRI data analysis. Future will tell whether these methods will be applied widely and will lead to ICA as common analysis method for fMRI just as is today the GLM.

Bibliography

[1] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[2] S. Amari. Natural gradient learning for over- and under-complete bases in ica. *Neural Computation*, 11:1875–1883, 1999.

[3] F.R. Bach and M.I. Jordan. Beyond independent components: Trees and clusters. *Journal of Machine Learning Research*, 4:1205–1233, 2003.

[4] H. Bauer. *Probability theory.* Walter de Gruyter, Berlin – New York, 1996.

[5] A.J. Bell and T.J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[6] A.J. Bell and T.J. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[7] J.W. Belliveau, D.N. Kennedy, R.C. McKinstry, B.R. Buchbinder, R.M. Weisskoff, M.S. Cohen, J.M. Vevea, T.J. Brady, and B.R. Rosen. Functional mapping of the human visual-cortex by magnetic-reonance-imaging. *Science*, 254:716–719, 1991.

[8] A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique based on second order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–444, 1997.

[9] A. Benveniste, M. Goursat, and G. Ruget. Robust identification of a nonminimum phase system: blind adjustment of a linear equalizer in data communications. *IEEE Transactions on Automatic Control*, 25(4):385–399, 1980.

[10] V. Calhoun, T. Adali, L. K. Hansen, J. Larsen, and J. Pekar. ICA of functional MRI data: An overview. In *Fourth International Symposium on Independent Component Analysis and Blind Source Separation (ICA 2003)*, pages 281–288, Nara, Japan, April 2003.

[11] V.D. Calhoun, T. Adali, G.D. Pearlson, and J.J. Pekar. A method for making group inferences from functional mri data using independent component analysis. *Human Brain Mapping*, 14:140–151, 2001.

[12] F. Esposito, E. Formisano, E. Seifritz, R. Goebel, R. Morrone, G. Tedeschi, and F. Di Salle. Spatial independent component analysis of functional mri time-series: To what extent do results depend on the algorithm used? *Human Brain Mapping*, 16:146–157, 2002.

[13] Brian S. Everitt. *Cluster Analyis*. Edward Arnold, London, third edition, 1993.

[14] L. Fahrmeir, I. Pigeot, and G. Tutz. *Statistik*. Springer Verlag Berlin, Heidelberg, 4. edition, 2004.

[15] R. S. J. Frackowiak, K. J. Friston, Ch. D. Frith, R. J. Dolan, and J. C. Mazziotta. *Human Brain Function*. Academic Press, San Diego, 1997.

[16] A.D. Gordon. A review of hierarchical classification. *Journal of the Royal Statistical Society, Series A*, 150(2):119–137, 1987.

[17] Stefan Harmeling, Frank Meinecke, and Klaus-R. Müller. Analysing ica components by injecting noise. In *Proc. Int. Workshop on Independent Component Analysis (ICA 2003)*, 2003.

[18] J. Himberg and A. Hyvärinen. Icasso: software for investigating the reliability of ica estimates by clustering and visualization. In *Proc. 2003 IEEE Workshop on Neural Networks for Signal Processing (NNSP2003)*, pages 259–268, 2003.

[19] Johann Himberg, Aapo Hyvärinen, and Fabrizio Esposito. Validating the independent components of neuroimaging time series via clustering and visualization. *Neuroimage*, 22(3):1214–1222, July 2004.

[20] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

[21] A. Hyvärinen, P.O. Hoyer, and M. Inki. Topographic independent component analysis. *Neural Computation*, 13(7):1525–1558, 2001.

[22] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13(4–5):411–430, 2000.

[23] C. Jutten and A. Taleb. Source separation: from dusk till dawn. In *Proc. 2nd Int. Workshop on Independent Component Analysis and Blind Source Separation (ICA2000), Helsinki, Finnland*, pages 15–26, 2000.

[24] K. K. Kwong, J. W. Belliveau, D. A. Chester, I. E. Goldberg, R. M. Weisskoff, B. P. Poncelet, D. N. Kennedy, B. E. Hoppel, M. S. Cohen, R. Turner, H-M. Cheng, T. J. Brady, and B. R. Rosen. Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation. *Proc. Natl. Acad. USA*, 89:5675–5679, 1992.

[25] M. McKeown, T. Jung, S. Makeig, G. Brown, S. Kindermann, A. Bell, and T. Sejnowksi. Analysis of fMRI data by blind separation into independent spatial components. *Human Brain Mapping*, 6:160–188, 1998.

[26] M.J. McKeown, L.K. Hansen, and T.J. Sejnowski. Independent component analysis of functional mri: what is signal and what is noise? *Current Opinion in Neurobiology*, 13(5):620–629, 2003.

[27] M.J. McKeown and T.J. Sejnowski. Independent component analysis of fmri data: Examining the assumptions. *Human Brain Mapping*, 6:368–372, 1998.

[28] Frank Meinecke, Andreas Ziehe, Motoaki Kawanabe, and Klaus-R. Müller. Assessing reliability of ica projections – a resampling approach. In T.-W. Lee, editor, *Proc. 3rd Int. Conference on Independent Component Analysis*, pages 74–79, 2001.

[29] FIL methods group. Spm2.

[30] Anke Meyer-Bäse, Fabian J. Theis, Oliver Lange, and Carlos G. Puntonet. Tree-dependent and topographic independent component analysis for fmri analysis. In *Volume 3195 of Lecture Notes in Computer Science*, pages 782–789, 2004.

[31] S. Ogawa, T. M. Lee, A. R. Kay, and D. W. Tank. Brain magnetic-resonance-imaging with contrast dependent on blood oxygenation. *Proc. Natl Acad. Sci. USA*, 87:9868–9872, 1990.

[32] E. Oja. A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3):167–173, 1982.

[33] E. Oja and J. Karhunen. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. Tech Report TKK-F-A458, Helsinki University of Technology, 1981.

[34] G. Pajares Martinsanz and J.M. De la Cruz García. *Visión por computador – Imágenes digitales y aplicaciones*. RA-MA Editorial Madrid, 2001.

[35] Athanasios Papoulis. *Probability, random variables, and stochastic processes*. McGraw-Hill, 3. edition, 1991.

[36] C.R. Rao. Characterisation of the distribution of random variables in linear structural relations. *Sankhya: The Indian Journal of Statistics*, 28(Series A, Pt. 2, 3):251–260, 1966.

[37] C.R. Rao. A decomposition theorem for vector variables with a linear structure. *Ann. Math Statistics*, 40:1845–1849, 1969.

[38] A. Rosenfield. Survey: Analysis and computer vision: 1999. *Computer vision and image understanding*, 78:222–302, 2000.

[39] T. D. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2(3):459 – 473, 1989.

[40] J. Shao and D. Tu. *The Jackknife and Bootstrap*. Springer, New York, 1995.

[41] K. Specht and J. Reul. Function segregation of the temporal lobes into highly differentiated subsystems for auditory perception: an auditory rapid event-related fmri-task. *NeuroImage*, 20:1944–1954, 2003.

[42] Karsten Specht. personal correspondance, 2004.

[43] Karsten Specht. personal correspondance, 2005.

[44] F.J. Theis, A. Jung, C.G. Puntonet, and E.W. Lang. Linear geometric ica: Fundamentals and algorithms. *Neural Computation*, 15:419–439, 2003.

Index