



Análisis de componentes independientes aplicado al estudio de la actividad cerebral

Diego Jose Sandoval Salazar

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Matemáticas
Bogota D.C., Colombia
2014

Análisis de componentes independientes aplicado al estudio de la actividad cerebral

Diego Jose Sandoval Salazar

Tesis presentada como requisito parcial para optar al título de:
Magister en Ciencias - Matemática Aplicada

Director:
Ph.D. Humberto Sarria Zapata

Línea de Investigación:
Matemática Aplicada

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Matemáticas
Bogota D.C., Colombia
2014

Dedicatoria

A mi padre (Q.E.P.D)

Resumen

Este trabajo presenta una propuesta para reducir la complejidad computacional del proceso de Análisis de Componentes Independientes [12] (ICA Independent Component Analysis). Para tal fin, se usan técnicas de Análisis Wavelet con el objeto de reducir los datos y por ende los cálculos de ICA. La evaluación de la eficiencia del método propuesto se hace en el marco de las señales generadas por electroencefalogramas (EEG) [7] para lograr determinar la región del cerebro en donde se genera la respuesta a un estímulo externo dado.

Abstract

This paper presents a proposal to reduce the computational complexity of Independent Component Analysis [12] (ICA). For this, Wavelet Analysis techniques are used in order to reduce the data and therefore the calculations of ICA. The evaluation of the efficiency of the proposed method is evaluated with the signals generated by electroencephalogram (EEG) [7] to achieve determining the region of the brain where the response to a given external stimulus is generated.

Contenido

Índice de figuras	xi
Introducción	xiii
1. Análisis de Componentes Independientes (ICA)	1
1.1. Motivación, el efecto Cocktail Party	1
1.2. Formulación del método ICA	3
1.3. Condiciones para el buen funcionamiento del método ICA	5
1.4. Construcción del método ICA	7
1.4.1. Blanqueo de datos	7
1.4.2. Teorema del límite central	10
1.4.3. Negentropía y medida de Nogaussianidad	11
1.4.4. Algoritmo en una dimensión	13
1.4.5. Algoritmo en varias dimensiones	14
1.4.6. Elección de la función de contraste G	15
1.5. Algoritmo en Matlab	16
2. ICA aplicado a la actividad cerebral	18
2.1. El electroencefalograma	18
2.2. ICA aplicado a la actividad cerebral	19
2.2.1. El estudio	20
2.2.2. Análisis ICA	22
3. Wavelets	25
3.1. Transformada de Fourier	25
3.1.1. Transformada Discreta (DFT)	25
3.1.2. Transformada de Fourier por ventanas STFT	26
3.2. Transformada Wavelet (CWT)	28
3.2.1. Características del análisis wavelet	28
3.2.2. Wavelet Haar	29
3.3. Wavelet Haar Inversa	30
3.4. Compresión de datos vía Wavelet	32

4. Analisis wavelet aplicado a EEG	33
4.1. Precisión al reducir datos con wavelet Haar	33
4.2. Análisis del costo computacional	34
5. Conclusiones y recomendaciones	36
5.1. Conclusiones	36
5.2. Recomendaciones	36
Bibliografía	38

Índice de figuras

1-1.	Efecto Cocktail party	2
1-2.	Señal Independiente A	3
1-3.	Señal Independiente B	3
1-4.	Señal Mezclada $M1=A+2.5*B$	3
1-5.	Señal Mezclada $M2=0.7*A-1.1*B$	4
1-6.	Señal Recuperada A	4
1-7.	Señal Recuperada B	5
1-8.	Señales independientes con fdp. no gaussiana	6
1-9.	Señales mezcladas	6
1-10.	Combinación de señales con densidad no Gaussiana	8
1-11.	Combinación de señales con densidad Gaussiana	8
1-12.	Datos blanqueados	10
1-13.	Funciones de densidad de las señales mezcladas	11
1-14.	Funciones de densidad de las señales	12
2-1.	Ubicación de electrodos	19
2-2.	EEGLAB Toolbox	20
2-3.	Imagen de EEG	21
2-4.	Ubicación de electrodos	21
2-5.	Imagen de EEG	22
2-6.	Señales independientes	23
2-7.	Resultados	24
3-1.	(Arriba) Representación discreta de la señal. (Abajo) Espectro de frecuencia mediante DFT.	26
3-2.	(arriba) señal $g(t)$. (abajo) Espectro usando DFT	27

3-3.	Función de ventana $h(t)$	27
3-4.	Comparación STFT Vs CWT	28
2-5.	Coeficientes Haar nivel 8	32
4-1.	Reducción de datos vía wavelet Haar. Ejemplo 1	34
4-1.	Reducción de datos vía wavelet Haar. Ejemplo 2	34
4-3.	Costo computacional	35

Introducción

El estudio del cerebro humano es un campo de conocimiento que siempre ha generado gran interés, durante los últimos años ha tenido avances muy significativos gracias a las nuevas teorías, procesos y tecnologías que permiten dar una idea de cómo funciona el cerebro humano. Sin embargo, aún se desconoce mucho sobre el mismo. Y muchas preguntas respecto a su funcionamiento aún son difíciles de contestar de manera acertada.

En la presente Tesis se realiza un estudio del algoritmo de Análisis de Componentes Independientes (ICA Independent Component Analysis) [12] con el fin de reducir el tiempo de cálculo, mediante eliminación y construcción de datos, irrelevantes y relevantes, usando Análisis Wavelet.

Para el proceso experimental numérico se toma como referencia las señales generadas por un cerebro humano y recogidas por un electroencefalograma en un estudio zonificación por regiones que tengan una misma respuesta respecto a un estímulo dado [7].

El cerebro humano está formado por neuronas, a través de las cuales, viajan los mensajes en términos de impulsos eléctricos. La electricidad que se genera en ciertas neuronas es la respuesta que la región del cerebro, en donde se encuentran dichas neuronas, emite a un estímulo dado. Sin embargo, es difícil determinar en qué región del cerebro se genera la respuesta, ya que la electricidad viaja por todo el cerebro y una medición puntual mediante un encefalograma mostrará un registro de actividad eléctrica sobre todo el cerebro.

El método ICA permite recoger la información de respuesta a un estímulo y separa las señales que se generaron independientemente en ciertas zonas del cerebro [4]. En este proceso, se usa una gran cantidad de información que hace que el cálculo sea lento, por esta razón es importante proponer métodos de cálculo que permitan reducir el tiempo de procesamiento.

En el capítulo 1, se presenta una descripción exhaustiva de la construcción del método ICA, también se muestra una construcción del algoritmo asociado en Matlab.

En el capítulo 2, se aplica el método ICA al estudio de la actividad cerebral con el fin de mostrar como funciona dentro del análisis de zonificación del cerebro como respuesta a un estímulo; para el análisis de los datos generados por el electroencefalograma (EEG) se utiliza un Toolbox de Matlab llamado EEGLab que permite el estudio, ilustración y análisis de los

datos.

En el capítulo 3, se hace una introducción breve al Análisis Wavelet y se describe la Wavelet Haar, que es una de las funciones wavelet computacionalmente menos costosa.

En el capítulo 4, se realizan varios experimentos de reducción de datos superfluos y construcción de datos relevantes usando Wavelets Haar, con el fin de disminuir los tiempos de cálculo del Método ICA aplicado a los electroencefalogramas. Además, se hace un análisis del costo y el ahorro computacional generado por el método propuesto.

Al final del trabajo se presentan algunas conclusiones y recomendaciones.

1 Análisis de Componentes Independientes (ICA)

Análisis de Componentes Independientes (ICA) es un método para la separación “ciega” de fuentes, basado en la independencia estadística de dichas fuentes. A diferencia del Análisis de Componentes Principales (PCA) en ICA, no se supone que las componentes del método sean ortogonales para su separación y únicamente asumimos la independencia estadística de las componentes. A continuación detallaremos una motivación para el método, así como la teoría necesaria para su construcción y aplicación.

1.1. Motivación, el efecto Cocktail Party

Supongamos que nos encontramos en una reunión, en donde se encuentran varias personas hablando al mismo tiempo (de aquí el nombre Cocktail Party). Nosotros estamos interesados en escuchar solamente a una de las personas que está hablando; lo que nosotros percibimos, es una mezcla de todas las voces y sonidos presentes en la reunión, sin embargo, el cerebro humano tiene la valiosa capacidad de aislar o separar los diferentes sonidos que está escuchando y concentrarse únicamente en la voz que se encuentra interesado.

A esta habilidad para seleccionar y reconocer solamente una fuente de sonido de un ambiente ruidoso se le conoce como Cocktail Party Effect [15], este fenómeno psico-acústico fue propuesto inicialmente por Colin Cherry en 1953 [11] y ha sido de gran interés en diversas áreas de estudio, entre ellas medicina, psicología, física y matemáticas.

Consideremos ahora la siguiente situación, hay dos locutores 1 y 2 (*Figura 1-1*), cada uno de ellos está hablando, por lo tanto, se generan dos señales acústicas s_1 y s_2 ; junto con los dos locutores, se encuentran dos micrófonos que graban el sonido que se genera cerca a ellos, los micrófonos no reciben independientemente cada una de las señales s_1 y s_2 que se están generando, en vez de eso reciben una mezcla de las dos señales, el método ICA permite recuperar las dos señales originales.

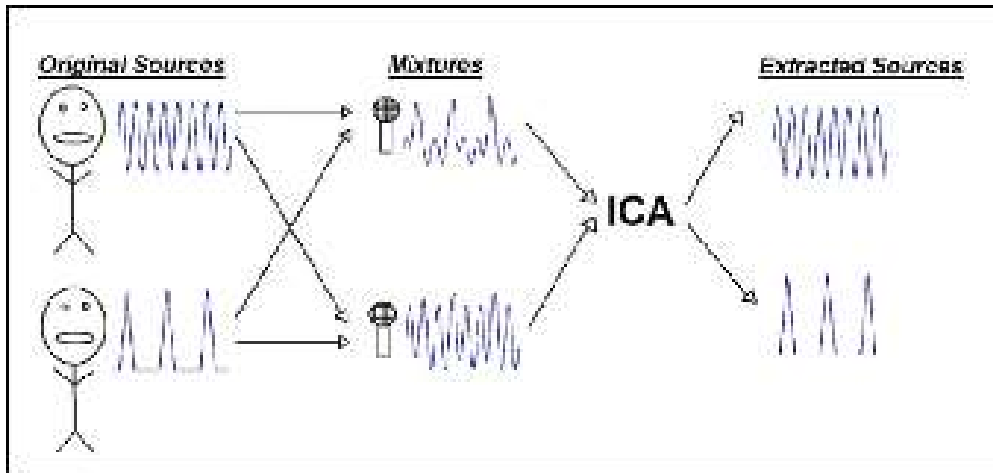


Figura 1-1: Efecto Cocktail party

En el siguiente ejemplo se tienen dos señales A (Figura 1-2) y B (Figura 1-3):

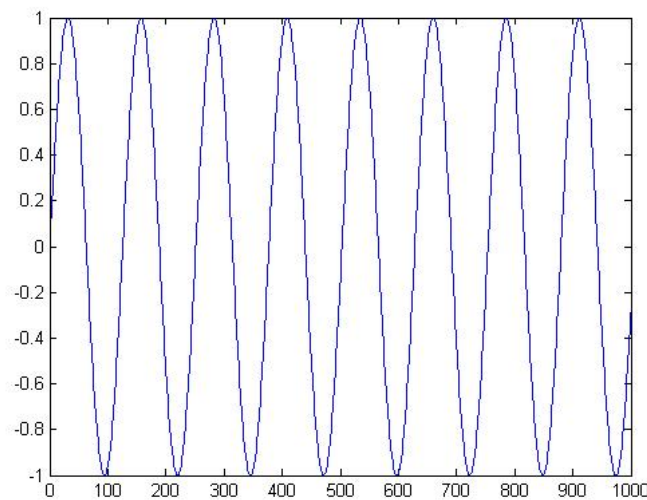


Figura 1-2: Señal Independiente A

que se mezclan linealmente para generar dos nuevas señales mezcladas $M1$ y $M2$. (Figuras 1-4 y 1-5)

Al aplicar el método ICA a las señales mezcladas $M1$ y $M2$, logramos recuperar las dos señales independientes salvo por un factor de escala (Figuras 1-6 y 1-7).

Podemos observar, que el método no garantiza la recuperación exacta de las señales originales, sin embargo, la única diferencia entre las señales independientes originales y las señales recuperadas al utilizar el método es un escalar (puede ser positivo o negativo)[10].

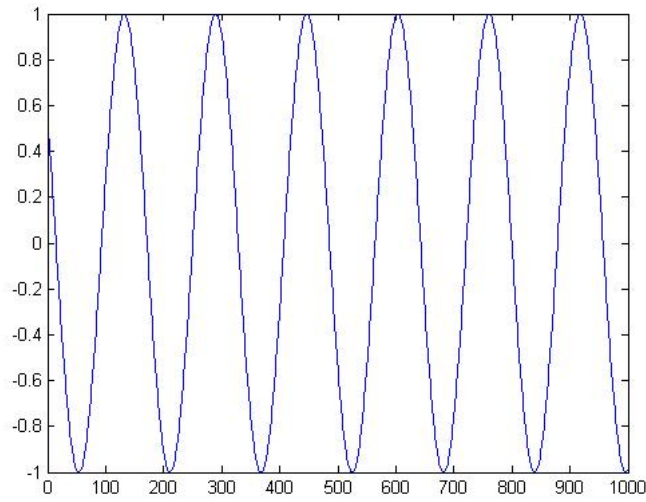


Figura 1-3: Señal Independiente B

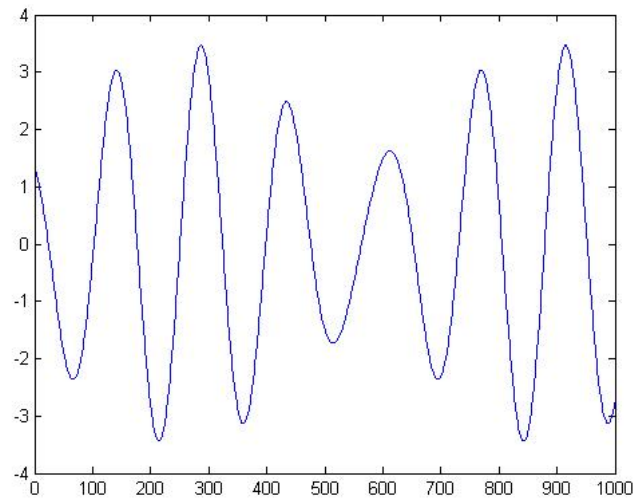


Figura 1-4: Señal Mezclada $M1=A+2.5*B$

1.2. Formulación del método ICA

Sea $\mathbf{x} = [x_1, x_2, \dots, x_m]$ un vector de m componentes que son observadas (*Figura 1-9*), asumimos que este vector se produce como una combinación lineal (mezcla) de m señales independientes (*Figura 1-8*) que denotaremos con el vector \mathbf{s} , $\mathbf{s} = [s_1, s_2, \dots, s_m]$, es decir, tenemos el

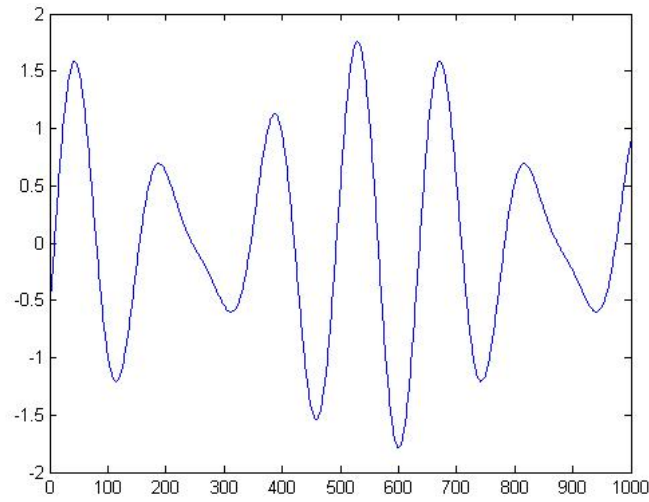


Figura 1-5: Señal Mezclada $M2=0.7*A-1.1*B$

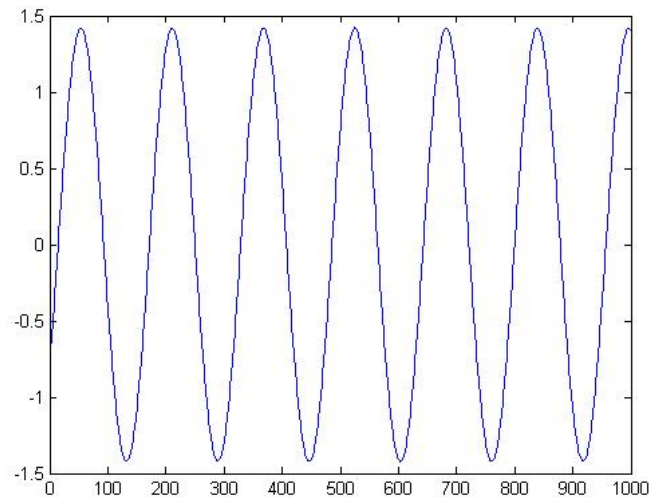


Figura 1-6: Señal Recuperada A.

sistema[13]:

$$\begin{aligned}
 x_1 &= a_{1,1}s_1 + a_{1,2}s_2 + \cdots + a_{1,m}s_m \\
 x_2 &= a_{2,1}s_1 + a_{2,2}s_2 + \cdots + a_{2,m}s_m \\
 &\vdots \\
 x_m &= a_{m,1}s_1 + a_{m,2}s_2 + \cdots + a_{m,m}s_m
 \end{aligned}$$

Los coeficientes $a_{i,j}$ determinan una matriz **A** que se conoce como **matriz de mezcla**, el vector **x** lo llamaremos **vector de mezclas** y **s** es el vector de componentes independientes.

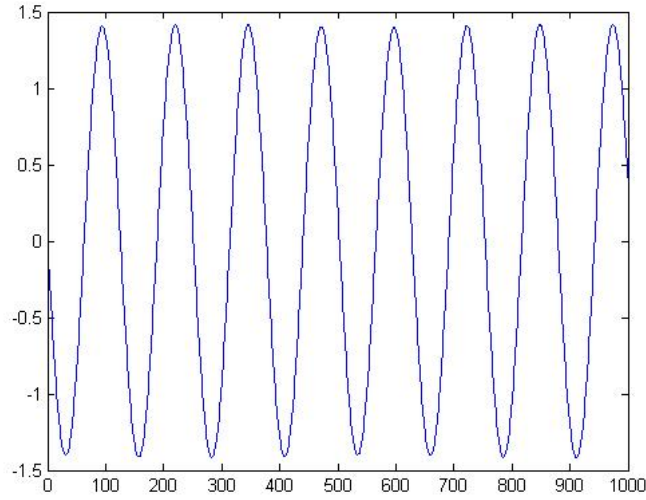


Figura 1-7: Señal Recuperada B.

Podemos escribir el anterior sistema en la forma:

$$\mathbf{x} = \mathbf{A}\mathbf{s}$$

En la práctica, sólo conocemos el vector de mezclas \mathbf{x} que se genera al muestrear algún evento. El método ICA consiste en aplicar un algoritmo que nos permita encontrar una matriz de desmezclado \mathbf{W} de tal modo que $\mathbf{y} = \mathbf{W}\mathbf{x}$ sea una buena aproximación del vector \mathbf{s} , es decir $\mathbf{y} \cong \mathbf{s}$.

1.3. Condiciones para el buen funcionamiento del método ICA

Para construir el algoritmo para el método, debemos asumir las siguientes condiciones que son necesarias para que el método genere buenos resultados:

- **Independencia estadística**

La independencia estadística es la idea primordial para toda la construcción del método, ya que queremos encontrar dentro de nuestras señales mezcladas, aquellas que son estadísticamente más independientes respecto a las demás. Podemos definir la independencia estadística como sigue:

Sean x_1, x_2, \dots, x_m un conjunto de variables aleatorias con función de densidad de probabilidad $f(x_1, x_2, \dots, x_m)$, entonces éstas variables son mutuamente independientes si:

$$f(x_1, x_2, \dots, x_m) = f_1(x_1)f_2(x_2)\dots f_m(x_m)$$

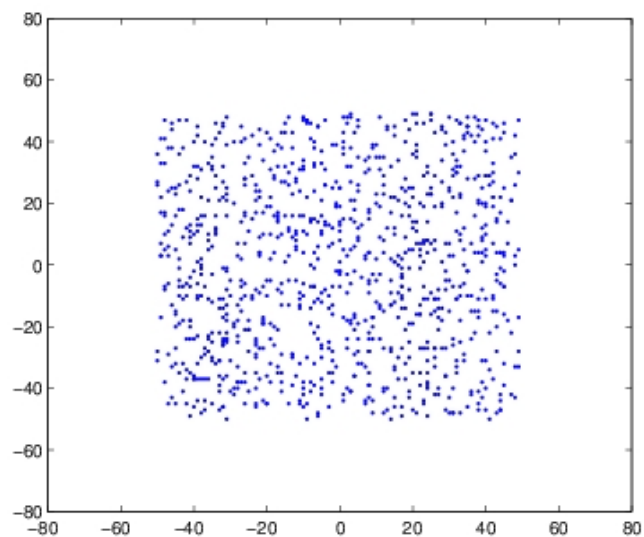


Figura 1-8: Señales independientes con fdp no gaussiana.

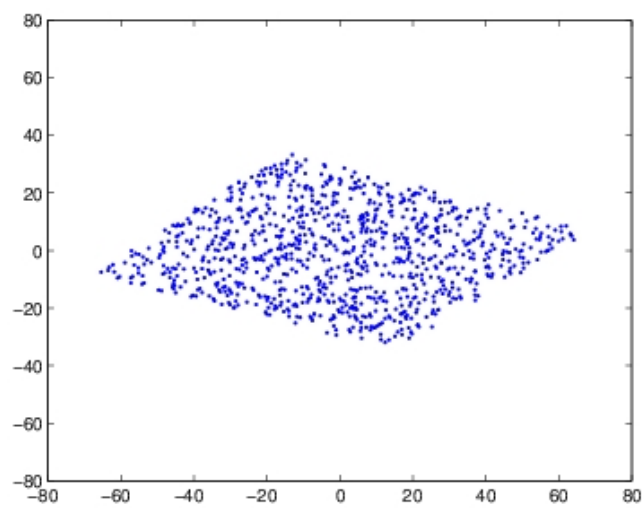


Figura 1-9: Señales mezcladas.

donde f_i es la función de densidad marginal de x_i . Asumiremos para la construcción del método que cada una de las señales s_i sea estadísticamente independiente respecto a las demás señales.

- **La matriz A debe ser cuadrada**

Para la construcción del método es necesario que la matriz de mezcla A sea cuadrada y de rango completo, es decir, que el número de fuentes s_i sea igual al número de mezclas x_i .

- **Se asume que el experimento está libre de ruido**

Todas las señales que se muestrean de las señales mezcladas \mathbf{x} deben tener solamente combinaciones lineales de las señales independientes \mathbf{s} , no debe haber información proveniente de ruido externo, ya que éste podría ser interpretado como otra señal independiente, lo cual podría generar interpretaciones incorrectas.

- **Los datos deben estar centrados**

Los datos ingresados al método ICA deben ser centrados, esto es, su media debe ser cero. En general, esta condición no se tiene, para solucionar esto, es necesario hacer un preproceso a los datos que se conoce como *blanqueo*. En la siguiente sección explicaremos dicho proceso.

- **Función de densidad de probabilidad no Gaussiana**

Las señales fuente independientes deben tener función de densidad de probabilidad no Gaussiana, de esta forma garantizamos que las componentes independientes efectivamente se pueden separar. Veamos un ejemplo de esto:

La *figura 1-10* muestra una combinación lineal de dos señales aleatorias con función de densidad de probabilidad no Gaussiana, en ésta se observan claramente las aristas que determinan la combinación lineal de los datos, es posible entonces separar las señales independientes originales.

Si la condición no se cumple, es decir si las señales tienen función de densidad Gaussiana, los datos estarán agrupados cerca a la media estadística de cada señal y no será posible separar las señales independientes. La *figura 1-11* ilustra esto.

1.4. Construcción del método ICA

1.4.1. Blanqueo de datos

[12]El preproceso de blanqueo de datos antes del algoritmo del método ICA, ayuda a garantizar la convergencia de este y también facilita algunas condiciones para la construcción del

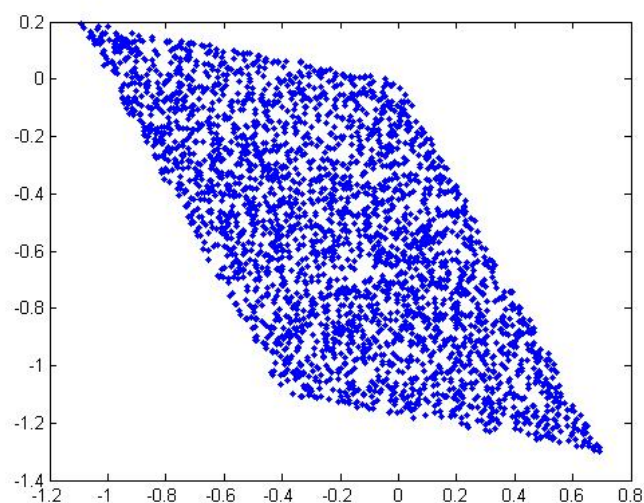


Figura 1-10: Combinación de señales con densidad no Gaussiana.

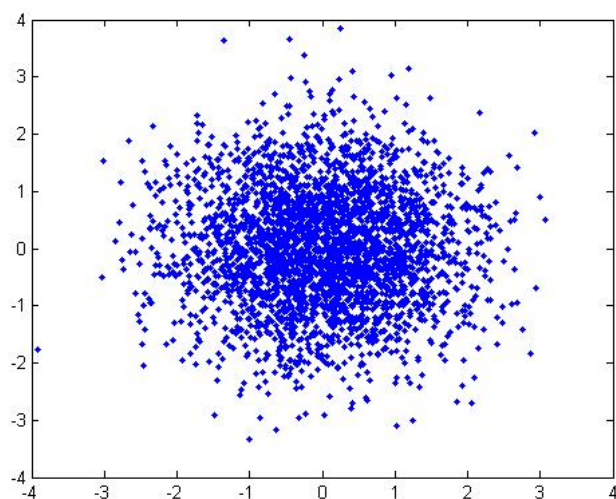


Figura 1-11: Combinación de señales con densidad Gaussiana.

algoritmo.

El primer paso para el blanqueo consiste en centrar los datos del vector de mezclas \mathbf{x} , para esto, restamos la media de cada una de las componentes observadas. Después de esto, a partir de la matriz de covarianza de los datos observados, eliminamos la correlación entre cada una de las señales observadas. A continuación explicamos, como se usa la matriz de covarianza.

La matriz de covarianza: Sea M una matriz $n \times l$, tomamos el promedio de cada columna x_j de M notado por \bar{x}_j ; con estos promedios formamos un vector que denotamos por \bar{X} . Se calcula cada entrada de la matriz de covarianza Q de M como:

$$q_{jk} = \frac{1}{n} \sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ki} - \bar{x}_k)$$

donde x_{ij} es cada una de las entradas de M , y n es el número de componentes en cada columna. En términos de cada columna X_j de M y del vector promedio de los datos, podemos escribir la matriz de covarianza Q como:

$$Q = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})^T$$

El resultado es una matriz Q cuadrada $n \times n$ simétrica que contiene sobre su diagonal la varianza de cada una de las columnas de la matriz y en las entradas q_{ij} contiene la covarianza entre la columna i y la columna j . El valor de covarianza entre dos variables, nos da un parámetro de correlación entre éstas; si la covarianza es mayor que 0 entonces se dice que las variables están directamente correlacionadas, esto quiere decir que si una de las variables aumenta, la otra también lo hace. Si la covarianza es menor que 0 entonces se dice que las variables están inversamente correlacionadas, esto quiere decir que si una de las variables aumenta, la otra disminuye. Si la covarianza es cero, las dos variables no tienen una relación lineal.

Para nuestro caso, el cálculo de la matriz de covarianza se reduce a:

$$Q = Cov(X) = \frac{XX^T}{n}$$

porque se trabaja con datos centrados, por lo tanto el vector promedio de cada señal es nulo. Realizamos ahora la descomposición espectral de la matriz de covarianza para obtener $Q = EDE^T$, eliminamos la covarianza entre las señales mezcladas calculando la inversa de la raíz cuadrada de la matriz de covarianza,

$$\tilde{Q} = Q^{-1/2} = ED^{-1/2}E^T$$

Al multiplicar nuestro vector de señales mezcladas por la matriz $D^{-\frac{1}{2}}E^T$, es decir $\tilde{X} = D^{-\frac{1}{2}}E^T X$ eliminamos la covarianza entre cada par de señales que tenemos; si calculamos la matriz de covarianza de los datos blanqueados obtenemos una matriz diagonal:

$$Cov(\tilde{X}) = \frac{\tilde{X}\tilde{X}^T}{n}$$

$$Cov(\tilde{X}) = \frac{D^{-\frac{1}{2}}E^T XX^T ED^{-\frac{1}{2}}}{n}$$

$$\begin{aligned}
&= D^{-\frac{1}{2}} E^T \frac{X X^T}{n} E D^{-\frac{1}{2}} \\
&= D^{-\frac{1}{2}} E^T \text{Cov}(X^T) E D^{-\frac{1}{2}} \\
&= D^{-\frac{1}{2}} E^T E D E^T E D^{-\frac{1}{2}} \\
&= I_n
\end{aligned}$$

Sabemos que la covarianza es una medida de correlación, con lo anterior se garantiza que cada una de las señales observadas está no correlacionada con las demás. Aplicando el proceso de blanqueo a los datos de la *figura1 – 9* obtenemos el siguiente resultado:

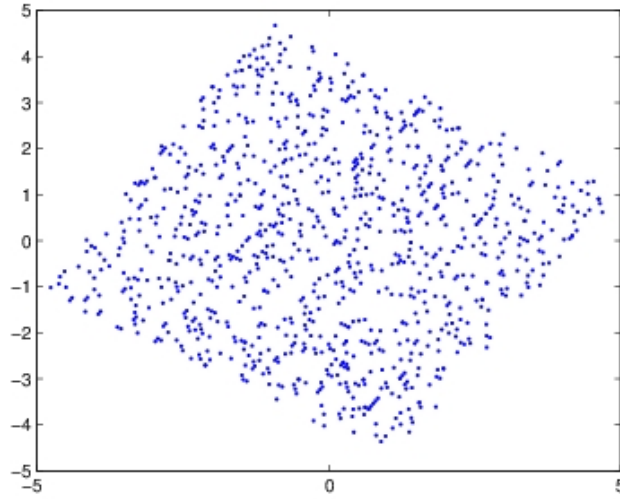


Figura 1-12: Datos blanqueados.

1.4.2. Teorema del límite central

[14] Como mencionamos anteriormente, la idea principal para el método ICA es la búsqueda de independencia estadística entre las variables, el siguiente teorema nos da una relación entre la medida de gaussianidad para variables aleatorias y la independencia estadística.

Teorema del Limite Central: Sea x_1, \dots, x_n una muestra aleatoria de tamaño n , es decir, una secuencia de variables aleatorias independientes e idénticamente distribuidas con valores esperados dados por μ y varianzas finitas dadas por σ^2 . Supongamos que estamos interesados en el promedio de la muestra

$$S_n = \frac{(x_1 + \dots + x_n)}{n}$$

de estas variables aleatorias. Por la ley de los grandes números, los promedios de la muestra convergen en probabilidad y casi seguramente al valor esperado μ cuando n tiende a infinito.

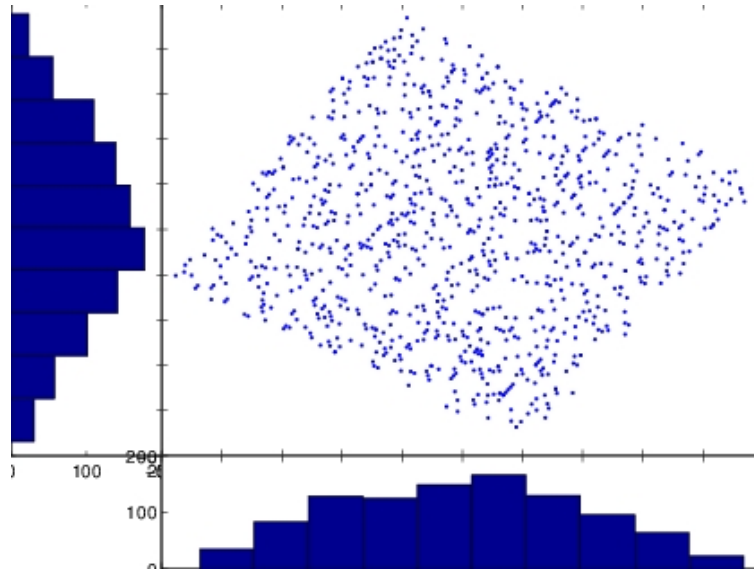


Figura 1-13: Funciones de densidad de las señales mezcladas.

El teorema del límite central afirma que a medida que n se hace más grande, la distribución de la diferencia entre la media de la muestra S_n y su límite μ , cuando n es muy grande, se aproxima a la distribución normal con media 0 y varianza σ^2 .

Por el teorema del límite central, nuestras señales mezcladas \mathbf{x} siempre van a ser más gaussianas que las señales independientes originales \mathbf{s} , la *figura 1-13* muestra las funciones de densidad de probabilidad para los datos de la *figura 1-12*, podemos observar la forma de campana gaussiana que tienen estas funciones de densidad (*figura 1-14*).

El trabajo del método consistirá entonces en tratar de reducir la gaussianidad que presentan las señales mezcladas para encontrar las señales independientes, geométricamente, es casi como "girar" los datos hacia los ejes que determinan las componentes independientes y de esta forma minimizar la gaussianidad entre las señales.

1.4.3. Negentropía y medida de Nogaussianidad

¿Cómo logramos entonces reducir la gaussianidad?, la medida de Gaussianidad está relacionada con la independencia estadística entre las variables, entre mayor sea la Gaussianidad [3], mayor información comparten las variables y son menos independientes entre sí. Definamos primero el concepto de Entropía H para un vector aleatorio $\mathbf{y} = [y_1, \dots, y_n]$:

$$H(\mathbf{y}) = - \int f(\mathbf{y}) \log f(\mathbf{y}) d\mathbf{y}$$

La entropía se puede considerar como una medida de cantidad de información mutua que contienen los elementos del vector. En base a la entropía H podemos definir la función de negentropía J como:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y})$$

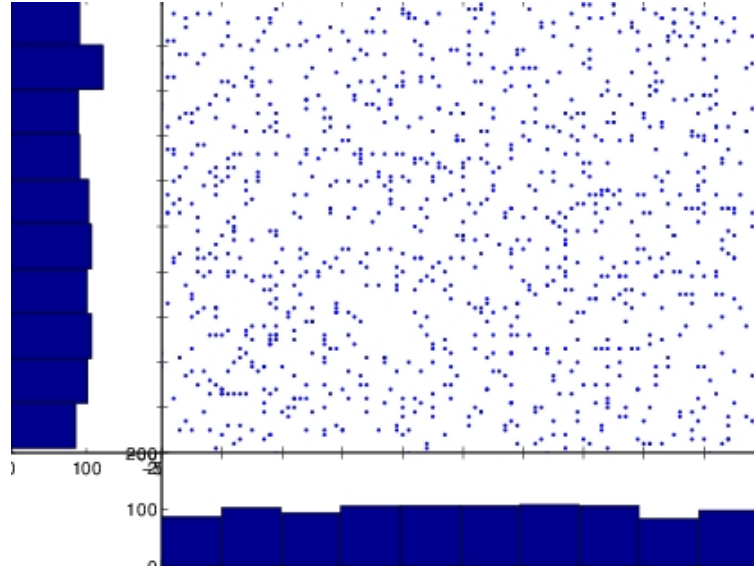


Figura 1-14: Funciones de densidad de las señales.

donde \mathbf{y}_{gauss} es un vector aleatorio Gaussiano con la misma matriz de covarianza de \mathbf{y} . Si la entropía mide la cantidad de información mutua en el vector, la negentropía determina la cantidad de información no mutua entre los elementos del vector, es decir a mayor negentropía, menor es la Gaussianidad de los elementos del vector. La negentropía la podemos considerar como una medida de Nogaussianidad.

Dado que la esperanza de una variable aleatoria se define como: $E(X) = \int x f(x) dx$, podemos usar una aproximación de la negentropía a través de la esperanza del vector aleatorio de la siguiente forma[3]:

$$J(y_i) \approx c[E\{G(y_i)\} - E\{G(V)\}]$$

donde G es una función llamada función de contraste, c es una constante irrelevante y v es una variable Gaussiana con media cero y varianza uno. Esta aproximación de la función de negentropía nos ayuda a definir el objetivo del método ICA, si hacemos $y_i = w^T x$, donde x es una de las componentes mezcladas y w^T es el vector de desmezclado para este x , nuestra meta será maximizar la función J dada por:

$$J(w) = E\{G(w^T x)\} - E\{G(v)\}$$

donde w es el vector $m - dimensional$ de mezcla de forma que $E\{(w^T x)^2\} = 1$. Esta ultima condición es simplemente de escala, para que los vectores no crezcan arbitrariamente.

Ahora podemos generalizar la función anterior para toda la matriz de desmezclado \mathbf{W} , teniendo en cuenta que se maximizan cada una de las negentropías cuando la suma de todas

ellas es máxima. El problema final de maximización será el siguiente:

$$\begin{aligned} \max \quad & \sum_{i=1}^m J(w_i) \\ \text{s.a.} \quad & E[(w_k^T x)(w_j^T x)] = \delta_{jk} \end{aligned}$$

La restricción del problema de maximización garantiza que cada una de las componentes de desmezclado w_j no esté correlacionada con las demás. Dos vectores aleatorios y_i, y_j están no correlacionados si $E[(y_i)(y_j)] - E(y_i)E(y_j) = 0$; Dado el preproceso de blanqueo de los datos, la media de cada vector es cero, por consiguiente $E(y_i) = 0$ y la condición de no correlación se reduce a

$$E[(y_i)(y_j)] = \delta_{jk}$$

donde

$$\delta_{jk} = \begin{cases} 1 & \text{si } j = k \\ 0 & \text{si } j \neq k \end{cases}$$

1.4.4. Algoritmo en una dimensión

Observemos que el óptimo de $J(w)$ se obtiene en un óptimo de $E\{G(w^T x)\}$. Aplicando el metodo de los multiplicadores de Lagrange, el óptimo de $E\{G(w^T x)\}$ bajo las condiciones $E\{G(w^T x)^2\} = 1, \|w\|^2 = 1$ se obtiene cuando

$$\nabla(E\{G(w^T x)\}) = \beta \nabla(\|w\|^2)$$

$$E\{xg(w^T x)\} - \beta w = 0$$

donde $g = G'$, β será entonces:

$$\beta = w_0^T E\{xg(w_0^T x)\}$$

$$\beta = E\{w_0^T xg(w_0^T x)\}$$

con w_0 el valor óptimo de w .

Para solucionar la ecuación $E\{xg(w^T x)\} - \beta w = 0$ y encontrar el optimo W_0 , aplicaremos el Método de Newton, si notamos el lado izquierdo de la ecuación anterior como F , obtenemos el Jacobiano JF como

$$JF(w) = E\{xx^T g'(w^T x)\} - \beta I$$

Para simplificar la expresión, aproximamos el término $E\{xx^T g'(w^T x)\}$, como $E\{xx^T\}E\{g'(w^T x)\} \simeq E\{g'(w^T x)\}I$, esto debido a que se garantiza independencia estadística entre los datos. Ahora la matriz Jacobiana es diagonal y puede representarse como:

$$JF(w) = E\{g'(w^T x)\}I - \beta I$$

$$= (E\{g'(w^T x)\} - \beta)I$$

lo anterior permite evitar el calculo de la inversa de la matriz Jacobiana al aplicar el método de Newton, porque la inversa de JF se reduce a:

$$JF(w) = \frac{1}{E\{g'(w^T x)\} - \beta} I$$

Aplicando el método de Newton se obtiene:

$$w^+ = w - [E\{xg(w^T x)\} - \beta w] / [E\{g'(w^T x)\} - \beta]$$

$$w^* = w^+ / \|w^+\|$$

donde w^* denota el nuevo valor de w , podemos simplificar el algoritmo multiplicando la ecuación anterior por $\beta - E\{g'(w^T x)\}$.

$$w^+ = E\{xg(w^T x)\} - E\{g'(w^T x)\}w$$

$$w^* = w^+ / \|w^+\|$$

1.4.5. Algoritmo en varias dimensiones

El algoritmo anterior, puede ser usado para construir un sistema de m vectores para m señales, así se obtienen los vectores de desmezclado w_1, \dots, w_n . Para ello se aplica el algoritmo anterior encontrando cada componente independiente una por una. Si ya se han encontrado p componentes independientes, podemos encontrar el $p + 1$ vector de mezcla de la siguiente forma:

1. Partiendo de un vector aleatorio w_{p+1} se realiza la resta de las proyecciones de los vectores w_1, w_2, \dots, w_p sobre el nuevo vector de la forma:

$$w_{p+1} = w_{p+1} - \sum_{j=1}^p w_{p+1}^T C w_j w_j$$

$$w_{p+1} = w_{p+1} / \sqrt{w_{p+1}^T C w_{p+1}}$$

donde C es la matriz de correlación.

2. El procedimiento anterior garantiza que el nuevo vector aleatorio que se usará no tenga relación con los vectores de desmezclado que ya se han encontrado. Con este nuevo vector se procede a aplicar nuevamente en algoritmo planteado en la sección anterior para encontrar una nueva componente independiente.

1.4.6. Elección de la función de contraste G

[2] Como en el proceso anterior se realiza una aproximación de la entropía de la variable aleatoria y en términos de la esperanza, es necesario usar una función adicional que llamaremos función de contraste G .

Recordemos que para una variable aleatoria y la entropía $H(y)$ se calcula como:

$$H(y) = - \int f(y) \log f(y) dy$$

donde $f(y)$ es la función de probabilidad de y .

Ahora la esperanza de una función cualquiera aplicada a la variable aleatoria se calcula como:

$$E[G(y)] = \int f(y) G(y) dy$$

Luego la función apropiada será $G(y) = -\log f(y)$, sin embargo dado que los datos que se tienen son mezclas de señales independientes, no conocemos la función de probabilidad $f(y)$, por ello es necesario aproximarla.

Según el teorema del límite central, las señales mezcladas de las señales independientes tienen distribución de probabilidad Gaussiana, por lo tanto aproximaremos la función de probabilidad con una función que aproxime una distribución Gaussiana, la elección más sencilla que cumpla esta condición será:

$$f(y) = k_1 e^{-k_2 y^\alpha}$$

con α un número par y k_1, k_2 constantes necesarias para que $f(y)$ sea función de probabilidad de varianza uno. Por practicidad se escogera $\alpha = 4$, $k_1 = 1$ y $k_2 = \frac{1}{4}$. De esta forma obtenemos:

$$\begin{aligned} G(y) &= -\log f(y) \\ &= -\log(e^{-\frac{1}{4}y^4}) \\ &= \frac{1}{4}y^4 \end{aligned}$$

Al calcular $G' = g$ se obtiene $g = y^3$, y $g' = 3y^2$, luego al reemplazar esta función en la iteración del método de Newton que habíamos obtenido para el cálculo de los vectores de desmezclado se obtiene:

$$w^+ = E\{x(w^T x)^3\} - 3E\{(w^T x)^2\}w$$

Debido a que nuestros datos fueron blanqueados y que la varianza de los mismos ahora es uno, lo anterior se reduce a:

$$\begin{aligned} w^+ &= E\{x(w^T x)^3\} - 3w \\ w^* &= w^+ / \|w^+\| \end{aligned}$$

1.5. Algoritmo en Matlab

El siguiente algoritmo desarrollado en matlab realiza primero un preprocesamiento de blanqueo de datos y después ingresa los datos mezclados para encontrar las componentes independientes a través del algoritmo de método ICA.

```
function [ica,A,W]=icadiego(senal)
covarianza=cov(senal', 1);
epsilon=0.0001;
    calculo de eigenvalores e eigenvectores
[E, D] = eig (covarianza);
blanqueo=inv (sqrt (D)) * E';
desblanqueo = E * sqrt (D);
blanqueo de la señal
nuevasenal= blanqueo*senal;
tamaño de la señal
[numcomponentes,nummuestras] = size(nuevasenal);
X=nuevasenal;
matriz para ir guardando
B = zeros(numcomponentes);
iteracion para la cantidad de componentes
for k=1:numcomponentes
    w = randn (numcomponentes, 1);
    w = w - B * B' * w;
    w = w / norm(w);
    w2=ones(numcomponentes,1);
    i=0;
    while norm(-w-w2)>epsilon
        w2=w;
        función para la iteracion
        w = (X * ((X' * w) .^ 3)) / nummuestras - 3 * w;
        normalizar el vector
        w = w / norm(w);
        guardo el resultado
        B(:, k) = w;
        A(:,k) = desblanqueo * w;
        W(k,:) = w' * blanqueo;
        i=i+1;
    end
end
```

i

```
end  
ica =W * senal;
```

2 ICA aplicado a la actividad cerebral

El deseo del ser humano de entender todo lo existente abarca también la necesidad humana de conocerse a sí mismo, entender el cuerpo y como funciona. Siempre se ha buscado entender como funciona cada parte del cuerpo humano, una en especial el cerebro, por ser el órgano principal y fuente de control de todo el cuerpo humano, existe particular interés en determinar de forma exacta cómo funciona, cómo se relaciona con el resto del cuerpo, cómo responde a los estímulos externos y sensaciones que percibimos.

Las herramientas médicas actuales, acompañadas de una interconexión directa con otras ciencias tales como matemáticas, física, ingeniería y ciencias sociales permiten avanzar a grandes pasos en la tarea de conocer el cerebro humano en totalidad. El diseño de algoritmos matemáticos, la creación de equipos medicos, la generación y mejora de terapias de recuperación son algunos de los avances que esta interconexión disciplinaria brinda a la sociedad.

A continuación describiremos como podemos aplicar el método de Análisis de Componentes Independientes (ICA) a los resultados recibidos al realizar un electroencefalograma a una persona, ésta herramienta matemática servirá de base para nuevos estudios que ayuden a perfeccionar el conocimiento que tenemos del cerebro humano.

2.1. El electroencefalograma

La electroencefalografía [5] es una rama de la medicina que estudia los potenciales eléctricos generados por el cerebro y que son percibidos mediante electrodos que se ubican sobre la superficie del cuero cabelludo. Un electroencefalograma (EEG) es el registro de dicha actividad eléctrica recibida por los electrodos [7]. Dichos registros varían mucho de persona a persona y por lo tanto es difícil asegurar qué parte del cerebro responde a cierto estímulo.

La historia del encefalograma [17] se remite a 1875, año en el cual el médico Británico Richard Birmick Caton presentó estudios sobre fenómenos bioeléctricos en el cerebro. Sin embargo, fue en 1928 cuando Hans Berger registro el primer electroencefalograma que llamó electrocerebrograma y con el que detectó ciertos potenciales eléctricos que se conocieron como el ritmo de Berger. Desde ese momento el tema cobró gran importancia en congresos y reuniones de medicina, en donde se discutía la veracidad del método, de los resultados y cómo

poder diagnosticar utilizando EEG.

Para un EEG se ubican electrodos en diferentes partes del cerebro, tratando de abarcar la mayor superficie posible, entre más electrodos se ubiquen sobre el encéfalo, más mediciones y más información se obtiene, en la práctica se ubican 32, 64 o 128 electrodos aproximadamente sobre la superficie del cuero cabelludo.

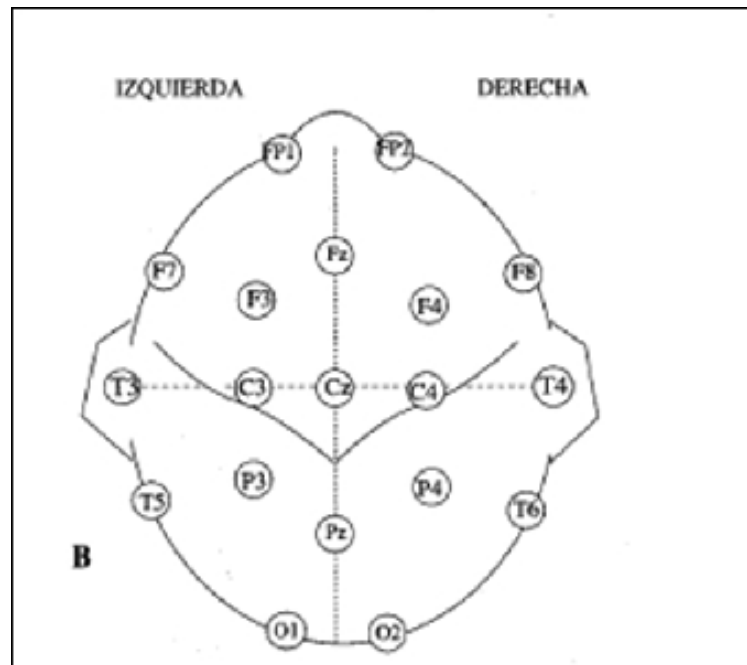


Figura 2-1: Ubicación de electrodos.

En general, no se necesita un tiempo determinado para la realización de un EEG, la observación puede realizarse desde unos pocos minutos hasta algunas horas, esto depende del estudio que se quiera realizar, los impulsos eléctricos recibidos por los electrodos viajan hacia un equipo que guarda los datos y los representa gráficamente. Generalmente se toman 1000 muestras de los impulsos eléctricos por segundo.

2.2. ICA aplicado a la actividad cerebral

Como dijimos anteriormente, el electroencefalograma muestra los potenciales eléctricos captados por cada uno de los electrodos ubicados en la superficie del cráneo, sin embargo la señal recibida por los electrodos no precede exactamente del lugar donde se ubica éste, sino que recibe la combinación de múltiples señales de diferentes zonas del cerebro que son las que se activan y responden al estímulo dado a la persona. El trabajo consiste entonces en lograr determinar cuales son las zonas del cerebro que efectivamente se activan, para esto

utilizaremos el método ICA.

2.2.1. El estudio

Para el siguiente ejercicio mostraremos un EEG de un sujeto de prueba, los datos fueron tomados de la página <http://sccn.ucsd.edu/~arno/indexeeg.html> [6] del Swartz Center for Computational Neuroscience de la Universidad de California, para poder visualizar y modelar los datos, se utiliza un toolbox para Matlab llamado EEGLAB Toolbox desarrollado por Arnaud Delorme y Scott Makeig, EEGLAB es una interfaz gráfica que permite visualizar, analizar y modificar los resultados obtenidos al realizar un EEG. Este Toolbox es de uso gratuito y también está disponible para descarga en la página de SCCN.

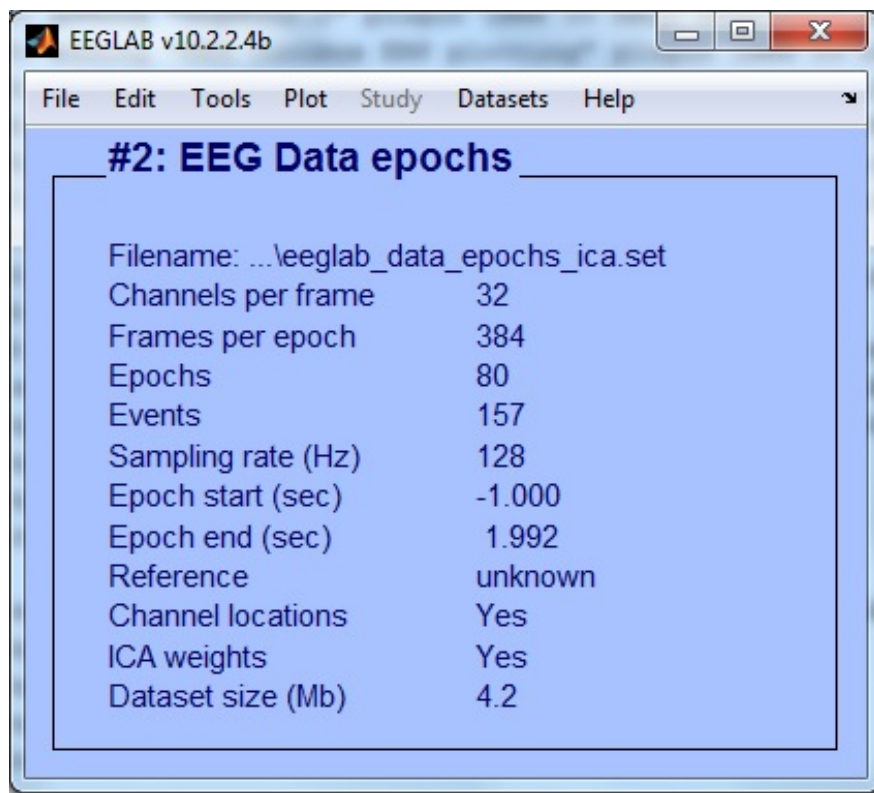


Figura 2-2: EEGLAB Toolbox.

Los datos descargados proceden de un experimento en el que se tomaron electroencefalogramas a 109 sujetos de prueba a los cuales se les muestrearon los impulsos cerebrales mientras observaban imágenes en una pantalla, el experimento se realizó durante dos días seguidos, dos veces por día. La *figura 2-3* muestra los resultados de un EEG realizado a una de las personas en el experimento. Para este EEG se ubicaron 32 electrodos sobre el cuero cabelludo como se muestra en la *figura 2-4*, se midieron los impulsos eléctricos durante 4 minutos

aproximadamente.

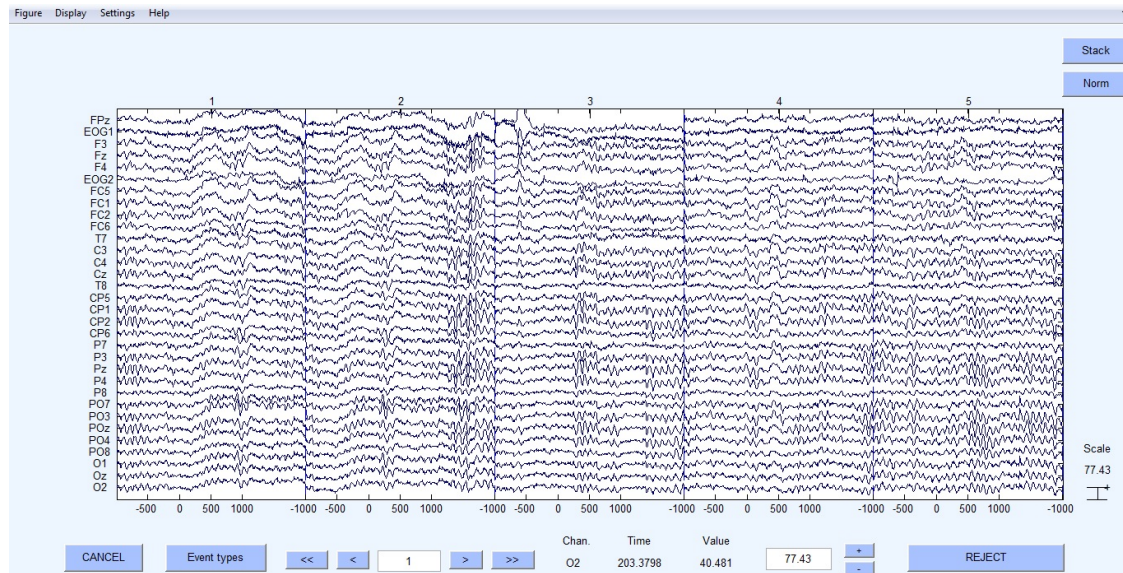


Figura 2-3: Imagen de EEG.

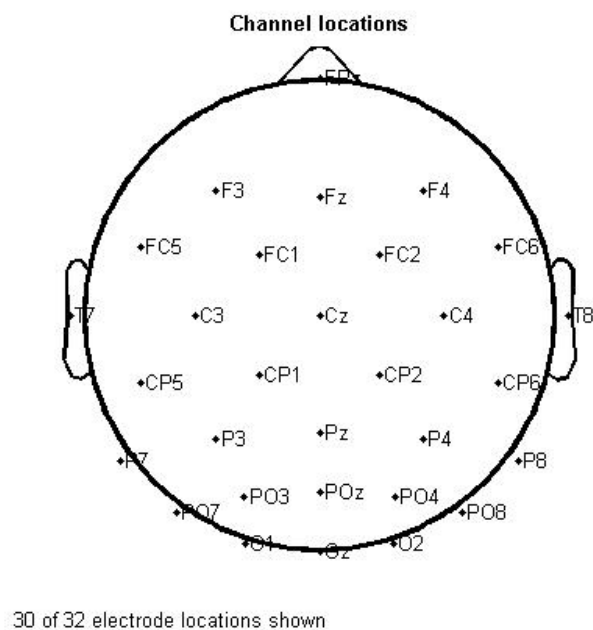


Figura 2-4: Ubicación de electrodos.

Si asumimos que los datos recibidos por cada electrodo son las zonas del cerebro en donde se genera el impulso eléctrico, obtendríamos un esquema de generación de impulsos eléctricos como el que se muestra en la *figura 2-5*:

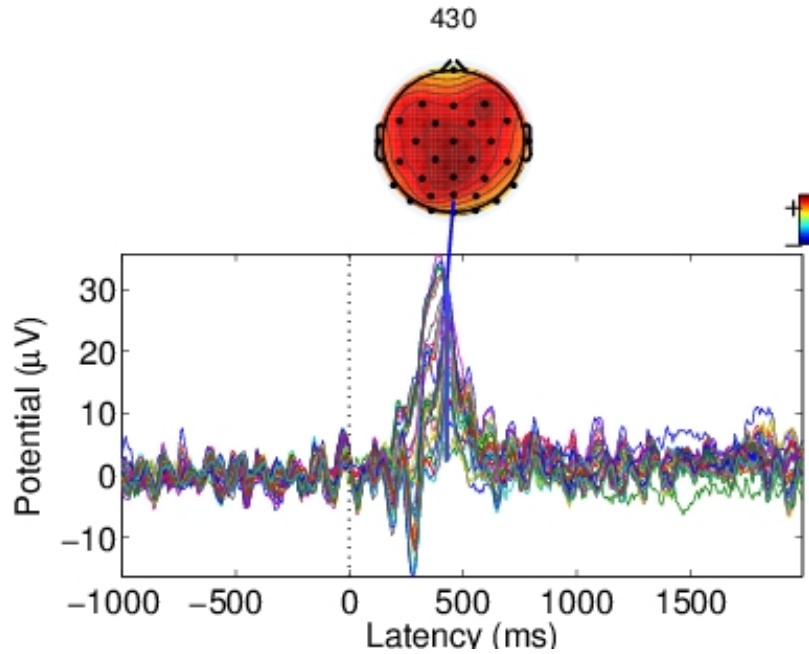


Figura 2-5: Imagen de EEG.

2.2.2. Análisis ICA

Sin embargo, los electrodos perciben una combinación de las señales eléctricas independientes que se generan en el encéfalo, podemos entonces aplicar el método ICA para encontrar las componentes independientes que dan origen a las combinaciones que reciben los electrodos. Si notamos como x_i cada una de las señales recogidas por cada electrodo, podemos conformar el vector de mezclas \mathbf{X} que ingresaremos al algoritmo ICA de la siguiente manera:

$$\mathbf{X} = [x_1, x_2, \dots, x_{32}]$$

El algoritmo ICA genera tres elementos: las señales independientes \mathbf{S} , la matriz de mezcla \mathbf{A} y la matriz de desmezclado \mathbf{W} , de tal forma que $\mathbf{AS} = \mathbf{X}$. Las componentes independientes encontradas son las siguientes:

¿Qué significan estas componentes independientes?

Para este caso, la información más importante, no son los resultados de las componentes independientes que se obtuvieron, ya que no podemos localizar éstas componentes de forma clara en algún lugar del cerebro, esto sucede dado que el método ICA busca las componentes independientes de manera aleatoria, sin embargo podemos completar nuestra tarea utilizando la matriz de desmezclado \mathbf{W} resultante, las entradas de esta matriz representan el peso que tiene cada componente independiente en el impulso eléctrico recibido por cada uno de los electrodos. Podemos ver entonces cada fila W_i de la matriz de desmezclado como la cantidad de información que aportan los electrodos a la i -ésima componente independiente, por lo

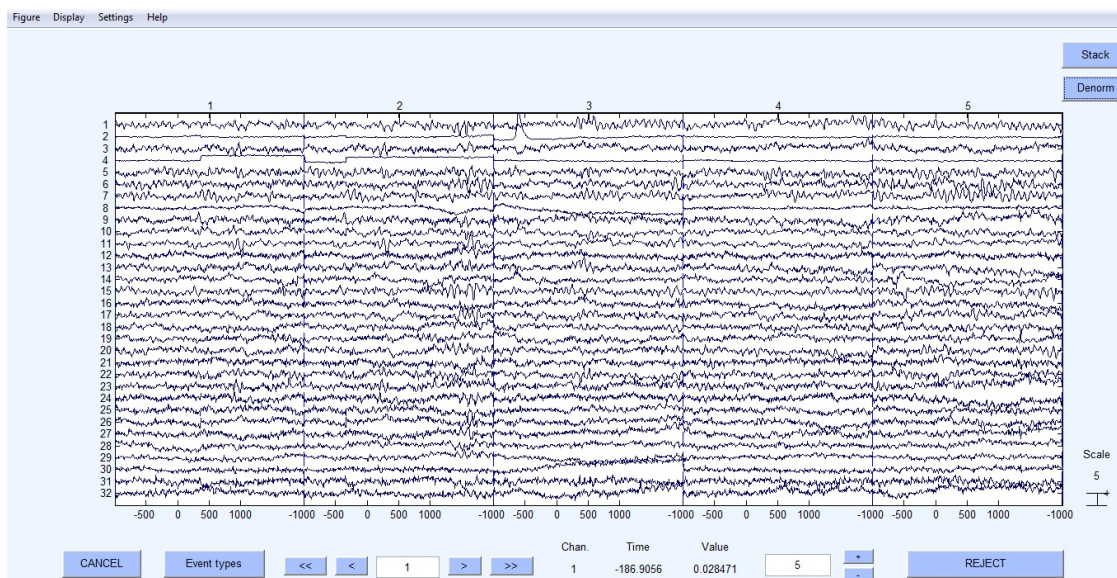


Figura 2-6: Señales independientes.

tanto, los mayores valores en cada una de las filas W_i . serán lo electrodos que más aportan a las componentes independientes.

Para el ejemplo anterior, las regiones del cerebro que más se activaron serán aquellas en donde se tenga mayor peso en cada una de las filas de la matriz de desmezclado. Si \mathbf{W} es la matriz de desmezclado entonces el vector T formado por los elementos

$$t_i = \max(\text{abs}(W_i))$$

representará los mayores coeficientes de peso para obtener las componentes independientes y las ubicaciones de estos coeficientes dentro de cada fila serán los electrodos que más información aportan a las componentes independientes, utilizando el comando $[\mathbf{t}, \mathbf{b}] = \max(\text{abs}(\mathbf{W}'))$ obtenemos el vector t que contiene los coeficientes más altos de cada fila y el vector b que contiene las ubicaciones de estos coeficientes dentro de cada fila de W .

$\mathbf{b} =$

Columns 1 through 18

3	31	27	32	7	31	31	19	17	19	12	28
8	30	14	19	28	3						

Columns 19 through 32

31 17 30 22 27 30 16 17 31 16 31 31
16 27

Podemos observar que el electrodo que más peso tiene dentro de las componentes independientes es el número 31 correspondiente a la region occipital central, ésta es entonces la región del cerebro que responde al estímulo dado. La siguiente imagen generada por EEGLAB ilustra los resultados obtenidos:

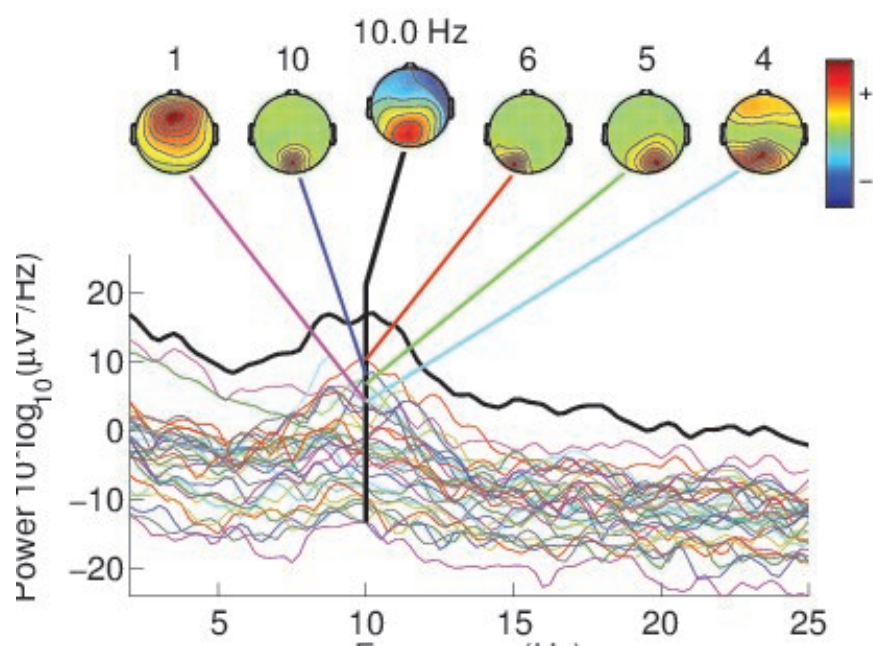


Figura 2-7: Resultados.

3 Wavelets

Ahora que conocemos y podemos aplicar el método ICA a la electroencefalografía, se buscará hacer más eficiente el tiempo de cómputo del método ICA cuando se trabaja sobre datos de electroencefalogramas. Para esto se realizara una compresión de los datos iniciales que se tienen de el EEG. Se busca usar una teoría que permita reducir significativamente la cantidad de los datos sin afectar en mayor medida los resultados que se obtienen con ICA. La teoría elegida para esto son las **Wavelets** o Ondeletas (Onditas), a continuacion se explicará detalladamente de dónde preceden las Wavelets y la compresión de datos via Wavelets.

3.1. Transformada de Fourier

La transformada de Fourier de una función de dominio real $f(t)$ se define como:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt$$

Y su transformada inversa como:

$$F^{-1}\{F(\omega)\} = f(t) = \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} dt$$

La transformada de Fourier obtiene una representación de la información de la señal pero en el dominio de la frecuencia y no en el dominio del tiempo.

3.1.1. Transformada Discreta (DFT)

Cuando no se tiene explícitamente una función, sino solamente un conjunto de datos discretos que muestrean dicha función se emplea la transformada discreta de Fourier. sea $x(t)$ una señal en el tiempo $x = [x_1, x_2, \dots, x_N]$ con N =cantidad de muestras, se define la transformada discreta de Fourier como:

$$F_n(\Omega) = \sum_{k=0}^{N-1} x_k e^{\frac{i2\pi kn}{N}}$$

con $n, k = 0, 1, 2, \dots, N - 1$.

En el siguiente ejemplo se muestra una señal discreta que muestrea la función:

$$f(t) = \sin(2\pi * 250t) + \sin(2\pi * 500t)$$

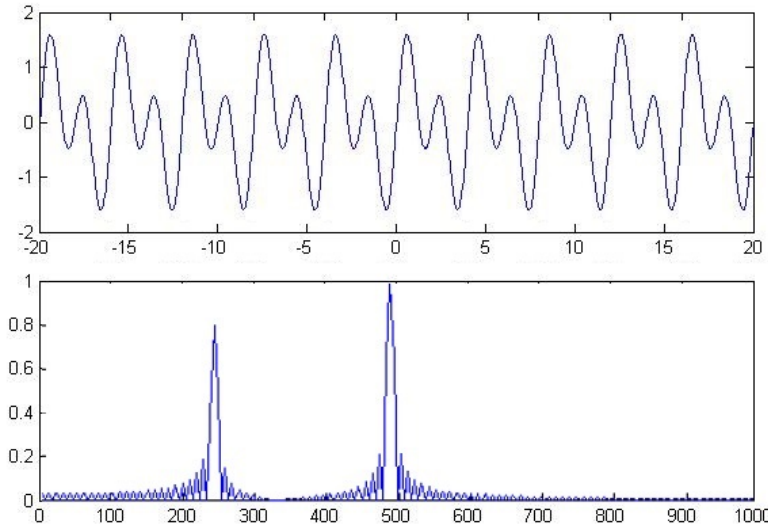


Figura 3-1: (*Arriba*) Representación discreta de la señal. (*Abaajo*) Espectro de frecuencia mediante DFT.

en el intervalo $[-20, 20]$ y su espectro de frecuencia al aplicar la TDF:

Se puede observar que la transformada DFT determina adecuadamente la resolución de frecuencias de la señal. Sin embargo, el análisis que realiza la DFT no es siempre confiable, supongamos ahora la siguiente función a trozos:

$$g(t) = \begin{cases} 0,8 \sin(2\pi * 250t) & \text{si } -20 \leq x < 0 \\ \sin(2\pi * 500t) & \text{si } 0 \leq x \leq 20 \end{cases}$$

A continuación se muestra la representación del muestreo de la señal $g(t)$ y el espectro de la DFT (*Figura 3-2*).

Para los dos ejemplos anteriores se observa que el espectro en el dominio de la frecuencia es muy similar para ambas señales, entonces la DFT determina con precisión las frecuencias altas de la señal, pero no logra determinar el lugar en donde se presentan en el dominio de la muestra.

3.1.2. Transformada de Fourier por ventanas STFT

[9] Para tratar de solucionar el problema de analizar las frecuencias y el segmento del dominio donde se presentan, se genera un proceso llamado *ventaneamiento*, que busca realizar una partición del dominio del tiempo de la función y realizar la transformada discreta DFT para cada uno de los segmentos que quedan. La forma de realizar esta división es usando una función auxiliar $h(t)$ que llamaremos función de ventana. Esta función se traslada en el dominio hasta un punto τ y allí se calcula la DFT, después se avanza otra sección, sin

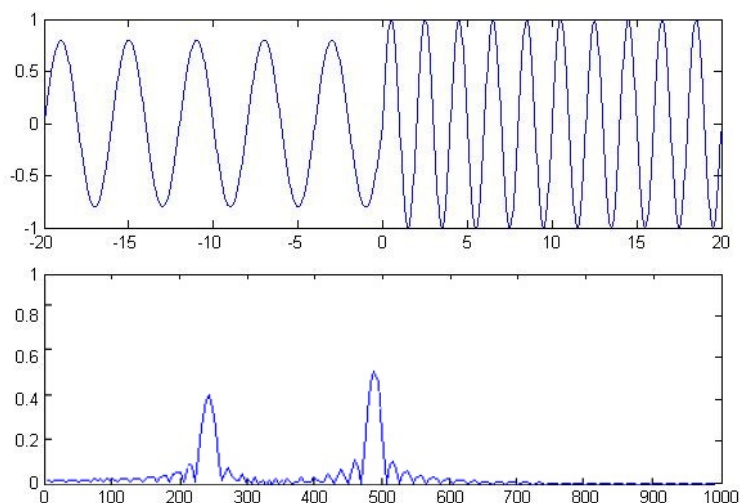


Figura 3-2: (arriba) señal $g(t)$. (abajo) Espectro usando DFT

sobreponer la ventana con la anterior y se repite el proceso hasta que se ha cubierto toda la señal.

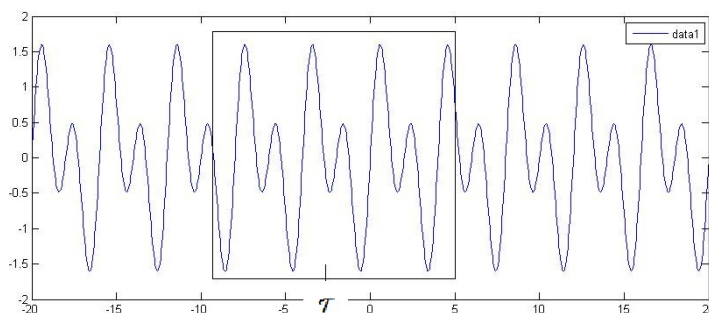


Figura 3-3: Función de ventana $h(t)$

Lo anterior se expresa formalmente mediante:

$$STFT(t, \omega) = \int_{-\infty}^{\infty} x(t)h(t - \tau)e^{-i\Omega t}dt$$

La transformada STFT permite entonces realizar un análisis más profundo de la señal, al realizar transformada de Fourier, no sobre toda el dominio de la señal, sino sobre ventanas más pequeñas.

Sin embargo, surge ahora la pregunta ¿qué tan anchas o angostas tienen que ser las ventanas? de tal forma que se realice la mejor búsqueda de frecuencias en la señal pero sin que sea demasiado costoso realizar el procedimiento.

3.2. Transformada Wavelet (CWT)

La transformada continua Wavelet constituye un procedimiento para el análisis local de una señal, al igual que la transformada STFT, la transformada wavelet realiza un análisis por ventanas de la señal. La diferencia reside en que la CWT realiza un recorrido dinámico del dominio de la señal (*Figura 3-3*).

La CWT expresa la señal $x(t)$ mediante coeficientes generados por el producto interno entre la señal y escalamientos y traslaciones de una función base $\psi(t)$ llamada *Wavelet Madre*. Se define la Transformada Continua Wavelet como[1]:

$$CWT(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi\left(\frac{t-b}{a}\right) dt$$

Se observa que en la CWT aparecen dos variables a y b , éstas corresponden respectivamente al escalamiento y la traslación que se va realizando de la wavelet madre $\psi(t)$.

La variable b se encarga de controlar la ubicación de la ventana dentro del dominio de la señal y la variable a determina el ancho de cada ventana. Tanto la variable de traslación b como

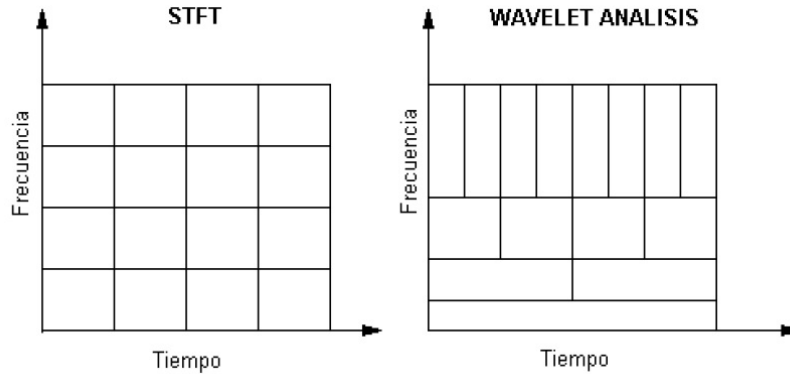


Figura 3-4: Comparación STFT Vs CWT

la de escalamiento a varían de forma continua para generar todo un sistema de funciones de la forma:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)$$

3.2.1. Características del análisis wavelet

Existen diversas wavelet madre y cada una genera todo un sistema de coeficientes wavelet, sin embargo todas en general satisfacen las siguientes propiedades:

1. Un sistema wavelet permite describir una señal de forma sencilla como combinación lineal en base a coeficientes $d_{j,k}$ ligados a la wavelet madre de la siguiente manera:

$$f(t) = \sum_k \sum_j d_{j,k} \psi_{j,k}$$

para algún conjunto de coeficientes $d_{j,k}$ con $j, k \in \mathbf{Z}$.

2. Los coeficientes wavelet se pueden calcular mediante un algoritmo en forma de arbol, es decir los coeficientes de mas baja resolución se calculan usando los coeficientes que ya han sido encontrados.
3. El tamaño de los coeficientes disminuye rapidamente a medida que se calculan.
4. Para la mayoría de wavelets, el cálculo de los coeficientes se reduce a sumas y multiplicaciones, esto hace que sea facil determinarlos y de poco costo computacional.

3.2.2. Wavelet Haar

[16]Esta wavelet es la mas usada ya que se reduce a calcular promedios (sumas) y cambio (diferencias) entre los datos. Toda wavelet madre tiene asociada una función de escala φ , para el caso de la wavelet Haar, la función de escala será:

$$\varphi_{u,v}(t) = \begin{cases} 1 & \text{si } u \leq x < v \\ 0 & \text{en otro caso} \end{cases}$$

La función de escala madre se definirá como:

$$\varphi_{0,1} = \varphi_{0,\frac{1}{2}} + \varphi_{\frac{1}{2},1}$$

usando esta misma funcion de escala se define la función wavelet Haar basica como:

$$\psi_{0,1} = \varphi_{0,\frac{1}{2}} - \varphi_{\frac{1}{2},1}$$

Estas dos definiciones anteriores multiplicadas por el coeficiente $\frac{1}{2}$ permiten generar los coeficientes wavelet con los datos de nuestra muestra de manera sencilla usando solamente promedios y diferencias entre los datos. Si se tiene un vector s de m datos con $m = 2^n$, se calcula el primer nivel de coeficientes wavelet de la forma

$$\begin{aligned} c^1 &= \left[\frac{s_1 + s_2}{2}, \frac{s_3 + s_4}{2}, \dots, \frac{s_{m-1} + s_m}{2}, \frac{s_1 - s_2}{2}, \frac{s_3 - s_4}{2}, \dots, \frac{s_{m-1} - s_m}{2} \right] \\ &= \frac{1}{2} [s_1 + s_2, s_3 + s_4, \dots, s_{m-1} + s_m, s_1 - s_2, s_3 - s_4, \dots, s_{m-1} - s_m] \end{aligned}$$

Observemos que el vector c^1 está dividido en dos partes, la primera parte corresponde únicamente a los promedios dos a dos de los elementos de s y la segunda sección corresponde a las diferencias dos a dos de los elementos de s .

Para el siguiente nivel wavelet se toma la primera seccion del vector c^1 y se repite el procedimiento anterior, la segunda parte del vector solamente se multiplica por factor de escala; es decir, si notamos los elementos de c^1 como

$$c^1 = [c_1^+, c_2^+, \dots, c_m^+, c_1^-, c_2^-, \dots, c_m^-]$$

el siguiente nivel de coeficientes wavelet será:

$$c^2 = \frac{1}{2}[c_1^+ + c_2^+, c_3^+ + c_4^+, \dots, c_{m-1}^+ + c_m^+, c_1^-, c_2^-, \dots, c_m^-]$$

el vector c^2 contiene en el primer cuarto, los promedios de los promedios de los elementos de s , en el segundo cuarto las diferencias de los promedios de s y en la segunda mitad solamente las diferencias de los elementos de s multiplicadas por $\frac{1}{4}$. El procedimiento se puede repetir hasta n veces, que es la cantidad de subdivisiones a la mitad que se pueden realizar del vector s , de tal forma que en cada nivel siguiente se calculan sumas y diferencias de cada subvector de promedios obtenido anteriormente.

A continuación se presenta el algoritmo para el cálculo de los coeficientes de la wavelet Haar.

.DATOS:

. s (vector de datos)
 . n (longitud de s)
 . t (cantidad de niveles wavelet)

.ITERACIÓN:

. $b = n$.para $k = 1$ hasta t
 . $p = 1$
 . $q = 1$
 . mientras $p < b$
 . $r = p + 1$
 . $resta_q = (s_p - s_r)$
 . $suma_q = (s_p + s_r)$
 . $p = p + 2$
 . $q = q + 1$
 . fin mientras
 . $x_{1:b} = 1/2 * [sumas, restas]$
 . $b = b/2$
 .fin para
 .SALIDA
 . x (vector de coeficientes wavelet nivel t)

3.3. Wavelet Haar Inversa

Al realizar el cálculo de los coeficientes wavelet hasta algun nivel t , es posible devolverse realizando de nuevo operaciones sencillas de sumas y restas, en términos de la función wavelet madre y función de escala madre, el proceso para obtener de nuevo los datos originales se

puede expresar mediante:

$$\frac{1}{2}(\varphi_{0,1} + \psi_{0,1}) = \varphi_{0,\frac{1}{2}}$$

$$\frac{1}{2}(\varphi_{0,1} - \psi_{0,1}) = \varphi_{\frac{1}{2},0}$$

En terminos de los datos que se generan al realizar un nivel wavelet, digamos c^1 , se obtuvieron los datos $c_1^+ = \frac{s_1+s_2}{2}$ y $c_1^- = \frac{s_1-s_2}{2}$. Se recuperan los datos originales con las operaciones:

$$s_1 = c_1^+ + c_1^-$$

$$s_2 = c_1^+ - c_1^-$$

Este proceso se puede repetir con todos los datos de un vector c^t hasta recuperar todos los datos del nivel wavelet anterior y también se repite tantas veces como t niveles wavelet se calcularon con la transformada wavelet. A continuación se muestra el algoritmo para el cálculo de los elementos originales de s cuando se tiene un vector de coeficientes wavelet c^t

.DATOS:

. x (vector de coeficientes wavelet nivel t)

. n (longitud de s)

. t (cantidad de niveles wavelet)

.ITERACIÓN:

. $b = n/(2^t)$.para $k = 1$ hasta t

. $p = 1$

. $q = 1$

. $resta_{1:b} = x_{1:b}$

. $suma_{1:b} = x_{b+1:2b}$

. mientras $p < b$

. $s_{q+1} = resta_p - suma_p$

. $s_q = resta_p + suma_p$

. $p = p + 1$

. $q = q + 2$

. fin mientras

. $b = 2b$

.fin para

.SALIDA

. s (vector de datos)

3.4. Compresión de datos vía Wavelet

Ahora que se conoce la forma sencilla como la wavelet Haar permite calcular coeficientes wavelet, se puede ahora realizar una compresion de los datos sin perder información relevante de los mismos [8]. Como se explicó en la seccion 4.1, la wavelet Haar va realizando sumas y restas. A medida que se calculan varios niveles c^t de coeficientes wavelet, la energía de la señal s se va concentrando en la primera sección del vector c^t y hacia la izquierda del vector, los coeficientes decaen acercándose a cero. La *figura 3-5* muestra el resultado del vector c^8 (8 niveles wavelet) para la señal recibida por un electrodo durante el EEG mostrado en el capítulo 3. Dado que la mayor parte de la información de la señal recibida se encuentra en

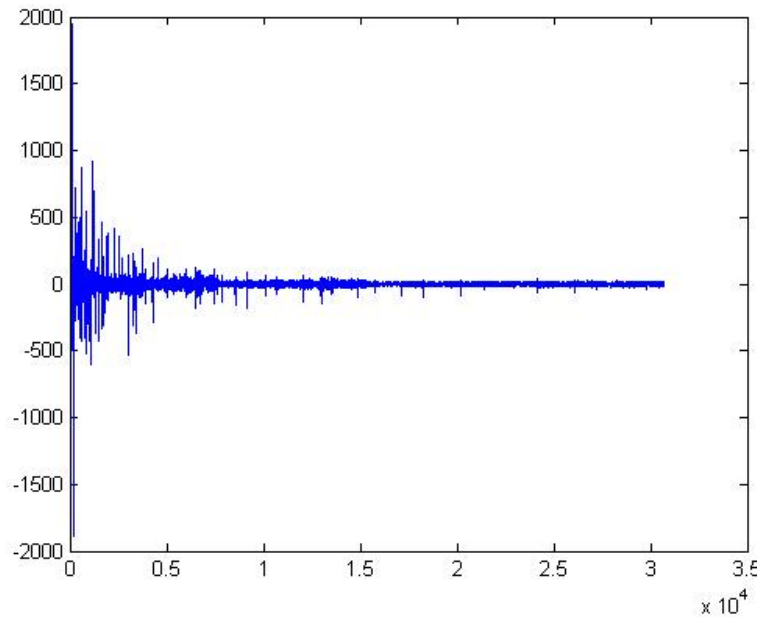


Figura 3-5: Coeficientes Haar nivel 8

la primera parte del vector c^t y dado que los coeficientes wavelet decrecen rapidamente, se puede reducir la cantidad de datos en potencias de 2, cortando el vector a la mitad u otra potencia de $\frac{1}{2}$ y tomando solo la primera parte de los coeficientes (donde se concentra la información).

Para obtener una aproximación de los datos originales, realizamos la transformada wavelet inversa, teniendo en cuenta que ya no podemos devolvernos tantos niveles como originalmente se calcularon, debido a la eliminación de coeficientes que se realizó; es decir, si se computaron en principio t niveles wavelet y el vector de coeficientes se recortó por ejemplo a un cuarto, es decir la longitud del nuevo vector es $\left(\frac{1}{2}\right)^2$, la cantidad de veces que se debe realizar el cálculo de coeficientes de wavelet inversa es $t - 2$ veces.

4 Analisis wavelet aplicado a EEG

Ahora que conocemos el proceso wavelet y cómo comprimir datos, tratando de evitar la pérdida de información, aplicaremos estos resultados para reducir la cantidad de datos que se tienen en los muestreos de los electroencefalogramas, para después aplicar el método ICA a los datos obtenidos.

La longitud del vector de datos de cada una de las señales recibidas por los electrodos ubicados en el craneo para el EEG es de 30720, adicionalmente se tienen 32 electrodos, lo cual generará entonces una matriz de $32 * 30720$ datos, buscamos aplicar la wavelet Haar para ir reduciendo los datos en un multiplo de dos varias veces.

4.1. Precisión al reducir datos con wavelet Haar

Como se mencionó en la sección 2.2.2., para nuestro trabajo se necesitan los electrodos que más se activan, respecto a las componentes independientes, después de aplicar el método de ICA. Se realizaron varias pruebas tomando algunos niveles wavelet para comprimir los datos y comparando los electrodos de mayor activación respecto a los resultados originales obtenidos en el capítulo 2.

Se realizaron pruebas usando 5, 8, y 11 niveles de wavelet Haar, 11 es la cantidad máxima de niveles que se pueden calcular para nuestros datos ya que la cantidad de datos se puede expresar como $30720 = 2^11 * 3 * 5$. Adicional a esto se realizó análisis ICA a los promedios de los datos originales. En la *figura 4-1* se muestran los resultados obtenidos.

Se puede observar en los resultados obtenidos que el método menos efectivo para la compresión de los datos es cálculo de promedios dos a dos de los datos. Aplicar las wavelets mejora notablemente la precisión cuando se reduce la cantidad de datos significativamente, por ejemplo al reducir la cantidad de datos a un dieciseisavo de la cantidad original, la wavelet Haar con 8 niveles obtiene una precisión del 69 %. Es decir con 6 % de los datos originales se obtiene un 69 % de precisión.

Se realizó otra prueba con otro conjunto de datos de electroencefalograma, la *figura 4-2* muestra los resultados obtenidos.

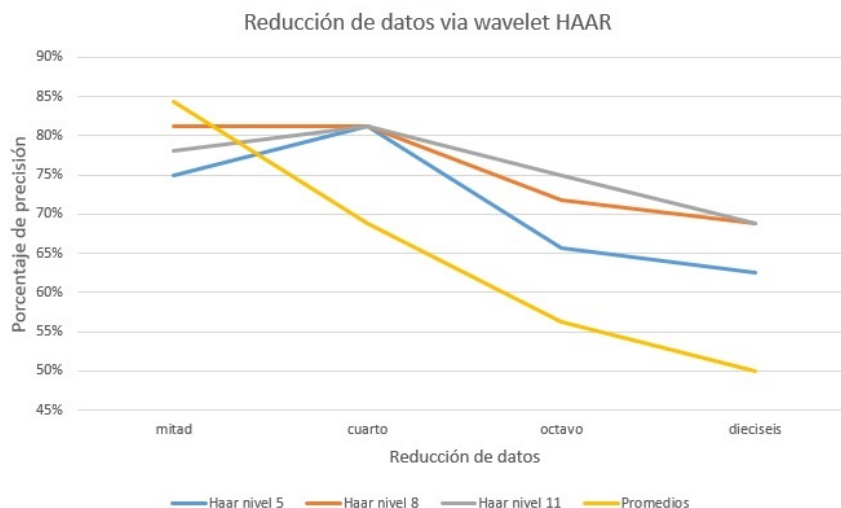


Figura 4-1: Reducción de datos vía wavelet Haar. Ejemplo 1

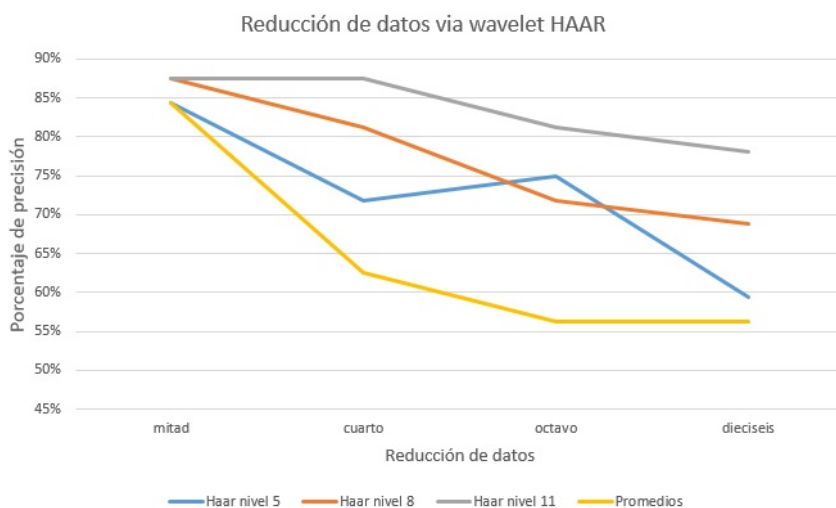


Figura 4-2: Reducción de datos vía wavelet Haar. Ejemplo 2

De nuevo se puede determinar que el cálculo de promedios genera malos resultados de precisión, se puede observar también en las *figuras 4-1 y 4-2*, que a medida que se aumentan los niveles wavelet se mejora la precisión, esto se debe a que entre mayor sea la cantidad de niveles, más información de los datos se condensa en los datos que no son eliminados.

4.2. Análisis del costo computacional

Al calcular la cantidad de operaciones necesarias para realizar el algoritmo del método ICA, se determina que es del orden $60m^3n^3$ donde n es la cantidad de señales recibidas y m la

cantidad de muestras de cada señal recibida. Para nuestro caso, $n = 32$ y $m = 30720$.

La cantidad de operaciones necesarias para el proceso de reducción de datos usando wavelet Haar es del orden $mn(4t - \frac{1}{2}2^t)$, en donde t es la cantidad de niveles wavelet procesados. Se realizó el cálculo de la cantidad de operaciones totales para el análisis wavelet pero resulta despreciable comparado con las operaciones necesarias para ICA.

La siguiente tabla Muestra la cantidad de operaciones necesarias según la cantidad de datos y el porcentaje que éstas representan respecto a la cantidad de datos totales.

Datos	Operaciones	Porcentaje
Datos iniciales	5,69987E+19	100%
mitad	7,12484E+18	12,500%
cuarto	8,90604E+17	1,5625%
octavo	1,11326E+17	0,1953%
dieciséis	1,39157E+16	0,02441%

Figura 4-3: Costo computacional

Estos resultados dejan ver el potencial del análisis wavelet para generar buenos resultados, reduciendo significativamente el costo computacional. Cuando se tiene sólo un dieciseisavo de los datos totales, la cantidad de operaciones se reduce al 0,02% de la cantidad original y como ya dijimos se logra obtener una precisión de hasta 78% respecto a los resultados de ICA originales.

5 Conclusiones y recomendaciones

5.1. Conclusiones

El método ICA es una herramienta muy poderosa para la separación ciega de fuentes, durante la elaboración de la presente tesis probé el método con diversos ejemplos y siempre se obtenían resultados óptimos. En los últimos años el método ICA ha ido evolucionando y mejorando de tal forma que ahora constituye un método fundamental cuando se requiere hablar de separación de señales.

Dado que actualmente las ciencias y el conocimiento del ser humano respecto a sí mismo en su parte fisiológica avanzan rápidamente y se apoyan en otras ciencias para generar un trabajo conjunto que ayude a fundamentar las aplicaciones y nuevas teorías que se generen con el fin de ayudar al ser humano a tener una vida digna. La aplicación de ICA a los EEGs es un aporte valioso que ayuda a determinar y poder comprender como funciona nuestro cerebro y de esta manera ayuda a generar procesos médicos que ayuden a evitar o contrarestar enfermedades que afectan el cerebro del ser humano.

La comprensión de todo el proceso necesario para la elaboración del método de ICA requiere un conocimiento matemático extenso, ya que aborda elementos de diversas áreas de la matemática, sobretodo de estadística.

Actualmente, uno de los procesos más usados para comprimir datos sin perdida importante de información es el análisis wavelet. Ésta teoría es ampliamente usada, en mayor medida usualmente se aplica al estudio de imagenes y a la reducción del peso de éstas sin perdida aparente de calidad. Sin embargo, la aplicación aquí dada es un claro ejemplo de la versatilidad del análisis Wavelet y la aplicación directa que se le puede dar para cualquier tipo de datos.

5.2. Recomendaciones

Aunque en este trabajo se realizó un desglose exhaustivo del método ICA, es posible realizar trabajo adicional sobre éste para realizar cambios y determinar una posible mejoría en el costo computacional. Por ejemplo en el algoritmo se escogió como función de contraste a

$G(y) = \frac{1}{4}y^4$, es posible escoger una función distinta que cumpla las mismas propiedades y analizar los resultados obtenidos.

Se puede continuar el trabajo de aplicación descrito en esta tesis, para aplicarlo no sólo a electroencefalogramas, sino a otras imágenes diagnósticas, para realizar una sectorización más precisa de las regiones del cerebro que responden a un estímulo dado.

Al realizar la compresión de datos mediante Wavelets, se escogió la wavelet Haar por ser la más sencilla computacionalmente. Sin embargo existen muchos y muy diversos sistemas wavelet que pueden ser usados y comparar sus resultados con los obtenidos en el presente trabajo.

Bibliografía

- [1] Introducción a la trasformada wavelet / Universidad nacional del centro de la provincia de Buenos Aires. 2006. – Informe de Investigación. – 39 p.
- [2] A., Hyvarinen: One-unit contrast functions for independent component analysis: a statistical analysis / Helsinki University of Technology. 1999. – Informe de Investigación. – 12 p.
- [3] A., Hyvarinen: New approximations of differential Entropy for Independent Component Analysis / Helsinki University of Technology. 2001. – Informe de Investigación. – 7 p.
- [4] ALIVISATOS P., George M. Church Ralph J. Roukes L. Yuste R.: The Brain Activity Map Project and the Challenge of Functional Connectomics / University of California. 2012. – Informe de Investigación. – 5 p.
- [5] BAREA NAVARRO, R.: Electroencefalografía / Universidad de Alcalá. 2006. – Informe de Investigación. – 26 p.
- [6] DELORME, A.: *fecha de consulta: 6 Agosto 2013.* recuperado de: <http://sccn.ucsd.edu/~arno/indexeeg.html> : Base de datos de electroencefalograma
- [7] DELORME, Kothe C. Bigdely N. Vankov A. Oostenveld R. Makeig S.: Matlab Tools for BCI Research. In "human-computer interaction and brain-computer interfaces".
- [8] DEVORE R., Popov V.: Compression of wavelet decompositions.
- [9] FAUNDEZ, A.: *Procesamiento Digital de señales acústicas utilizando Wavelets*, Instituto de Matemáticas UACH, Tesis de Doctorado, 2011
- [10] GRANDCHAMP, Delorme A.: NeuroTRIP: a Neurofeedback experiment design platform using Independent Component Analysis.
- [11] HAYKIN, S. ; CHEN, z.: The Cocktail Party Problem / Adaptive Systems Lab, McMaster University, Hamilton, Ontario, Canada L8S 4K1. 2005. – Informe de Investigación. – 1875–1902 p.
- [12] HYVARINEN A., Oja E.: A fastfixed-point algorithm for independent component analysis.

-
- [13] LANGLOYS D., Gosselin D.: An introduction to independent component analysis: infomax and fastica algorithms.
 - [14] MARTÍNEZ G. MONICA, Marí B. M.: Utilización Práctica del Teorema del Límite Central / Universidad Politecnica de Valencia. 2008. – Informe de Investigación. – 6 p.
 - [15] McDERMOTT, J: The cocktail party problem, 2009.
 - [16] NIEVERGELT, Yves: *Wavelets Made Easy*. Boston : Birkhäuser, 1999
 - [17] PALACIOS SÁNCHEZ, L.: Breve historia de la electroencefalografía / Universidad del Rosario. 2002. – Informe de Investigación. – 4 p.