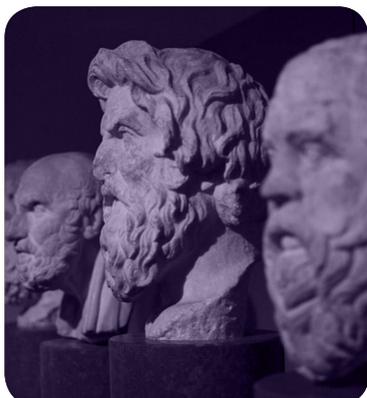
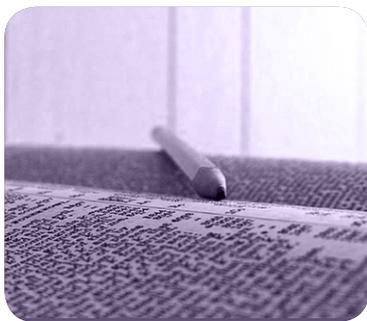




MÁSTERES de la UAM

Facultad de Psicología
/ 16-17

Metodología
de las Ciencias
del Comportamiento
y de la Salud



**Aplicación de un
método de
clústering a una
Región de Interés
en datos de
Resonancia
Magnética
Funcional en estado
de reposo**

Luis Marcos Vidal

Aplicación de un método de clústering a una Región de Interés en datos de Resonancia Magnética Funcional en estado de reposo

Trabajo de Fin de Máster (Modalidad B)

Luis Marcos Vidal

Tutora Académica: Carmen Ximénez

RESUMEN

El análisis de resonancia magnética funcional (RMf) es uno de los métodos más utilizados en la actualidad para estudiar el cerebro. En su modalidad en estado de reposo es muy frecuente medir la sincronización de la actividad de dos o más regiones. Para ello es necesario obtener una medida de dicha actividad. En la práctica, esto se lleva a cabo calculando el promedio de la actividad de sus unidades, lo que no está exento de problemas. El objetivo de este trabajo es proponer una forma de obtener varias señales de una región que la representen mejor y sean más estables que el promedio. Para ello, dividiremos la región en subunidades mediante un método de clústering para luego obtener una señal promedio de cada subunidad. Para dividir la región, se propone el uso de un método de clústering que incluye varias técnicas estadísticas para asegurar que las subunidades regionales que se obtienen son estables. Se llevaron a cabo dos estudios: uno de simulación y otro empírico. Los resultados obtenidos al aplicar el método a datos simulados muestran que la división que se hace de la muestra es adecuada y que las señales obtenidas de las subunidades son más estables y representan mejor a las regiones. Sin embargo, al aplicar el método a datos reales de RMf (en concreto, sobre cuatro regiones de 94 sujetos), el modelo de un clúster ajustó mejor que el dividido, en la mayoría de los casos. Además, en los sujetos en los que se encontraron subunidades regionales, la estabilidad y la representatividad no mejoraron.

ÍNDICE

RESUMEN	1
ÍNDICE	2
LISTADO DE ABREVIATURAS	4
1. INTRODUCCIÓN	5
1.1. Conectividad funcional	5
1.1.1. <i>Resonancia Magnética Funcional en Reposo</i>	<i>5</i>
1.1.2. <i>Conectividad funcional</i>	<i>7</i>
1.1.3. <i>Redes de conectividad funcional</i>	<i>7</i>
1.2. La correlación de señales promedio como medida de conectividad funcional	8
1.3. Análisis Clúster	10
1.3.1. <i>K-medias.....</i>	<i>12</i>
1.3.2. <i>X-medias y Máxima Verosimilitud.....</i>	<i>14</i>
1.3.3. <i>Modelos de Mezcla de Gaussianas y Expectación-Maximización.....</i>	<i>16</i>
1.3.4. <i>Clustering Jerárquico Aglomerativo.....</i>	<i>19</i>
1.4. Objetivo e hipótesis	20
1.4.1. <i>Hipótesis</i>	<i>20</i>
2. MATERIAL Y MÉTODOS	21
2.1. Diseño enventanado.....	21
2.2. Método empleado	22
2.2.1. <i>Obtención de Clústers estables: Circular block bootstrap</i>	<i>22</i>
2.2.2. <i>X-GMM</i>	<i>24</i>
2.2.3. <i>Visión general del método.....</i>	<i>26</i>
2.3. Comparación entre modelos.....	27
2.4. Software.....	28
3. RESULTADOS	29

3.1. Estudio en datos simulados.....	29
3.1.1. <i>Modelo de simulación.....</i>	29
3.1.2. <i>Relación de los parámetros con los resultados.....</i>	30
3.1.3. <i>Evaluación del gold standard.....</i>	32
3.1.4. <i>Evaluación del método.....</i>	33
3.1.5. <i>Evaluación de las medidas de estabilidad y representatividad.....</i>	34
3.2. Estudio en datos reales.....	35
3.2.1. <i>Participantes.....</i>	35
3.2.2. <i>Parámetros de adquisición.....</i>	35
3.2.3. <i>Preprocesado.....</i>	36
3.2.4. <i>Selección de las Regiones de Interés.....</i>	36
3.2.1. <i>Resultados del estudio en datos reales.....</i>	37
4. DISCUSIÓN	39
4.1. Evaluación del método: resultados en datos simulados	39
4.2. Aplicación del método en datos reales.	40
4.3. Posibles discrepancias entre los resultados obtenidos en datos simulados y en los reales	41
4.4. Limitaciones	42
REFERENCIAS.....	43
ANEXOS.....	47
ANEXO 1. Ejemplo de la aplicación del método de clústering a una región de un sujeto	49
ANEXO 2. Código del algoritmo <i>X-GMM</i>	51
ANEXO 3. Técnicas de preprocesado de imágenes de RMf	54

LISTADO DE ABREVIATURAS

ACP: Análisis de Componentes Principales.

BIC: Criterio de Información Bayesiano (*Bayesian Information Criterion*)

CBB: *Circular Block Bootstrap*.

CCP: Córtex Cingulado Posterior.

CP: Componentes Principales.

CPFm: Córtex Prefrontal medial.

E-M: Expectación-Maximización.

GMM: Modelos de Mezcla de Gaussianas (*Gaussian Mixture Models*)

Md: Corteza Motora derecha.

Mi: Corteza Motora izquierda.

MRC: Modelo de la Región Completa.

MRD: Modelo de la Región Dividida.

PVMC: Porcentaje de Vóxeles Mal Clasificados.

RM: Resonancia Magnética.

RMf: Resonancia Magnética funcional

ROI: Región de Interés (*Region Of Interest*).

r-RMf: Resonancia Magnética funcional en estado de reposo.

SNR: Relación señal-ruido (*Signal to Noise Ratio*).

1. INTRODUCCIÓN

El presente trabajo se enmarca dentro del campo de la resonancia magnética funcional y aborda un problema metodológico existente con aplicaciones tanto en investigación básica como clínica. La pregunta en cuestión es cómo extraer una medida de la activación de una región del cerebro, que sea tanto representativa de las unidades que la componen como estable a lo largo del tiempo. Esta medida se utiliza en varias técnicas de análisis de neuroimagen funcional (como el estudio de la conectividad funcional de regiones de interés (*ROIs* por sus siglas en inglés), el análisis de semillas o el análisis de redes funcionales).

1.1. Conectividad funcional

1.1.1. *Resonancia Magnética Funcional en Reposo*

La resonancia magnética (RM) es una de las técnicas de neuroimagen más utilizadas en la actualidad, tanto en su modalidad estructural como funcional. Dicha técnica se basa en el fenómeno conocido como resonancia magnética nuclear para obtener información de la composición del tejido de determinadas estructuras u órganos. La RM, en su modalidad funcional (en adelante, RMf), permite evaluar la activación de las áreas cerebrales en base a los niveles de oxihemoglobina y desoxihemoglobina (ambas con propiedades magnéticas diferenciadas), ya que las zonas con mayor activación requerirán un mayor consumo de oxígeno. Esta actividad queda registrada como la activación de multitud de vóxeles (las unidades espaciales 3D de la imagen o píxeles de volumen cerebral) que conforman una imagen del tejido cerebral a lo largo del tiempo. En la Figura 1 se puede ver una representación de la división de un cerebro en vóxeles mediante una imagen de RM.

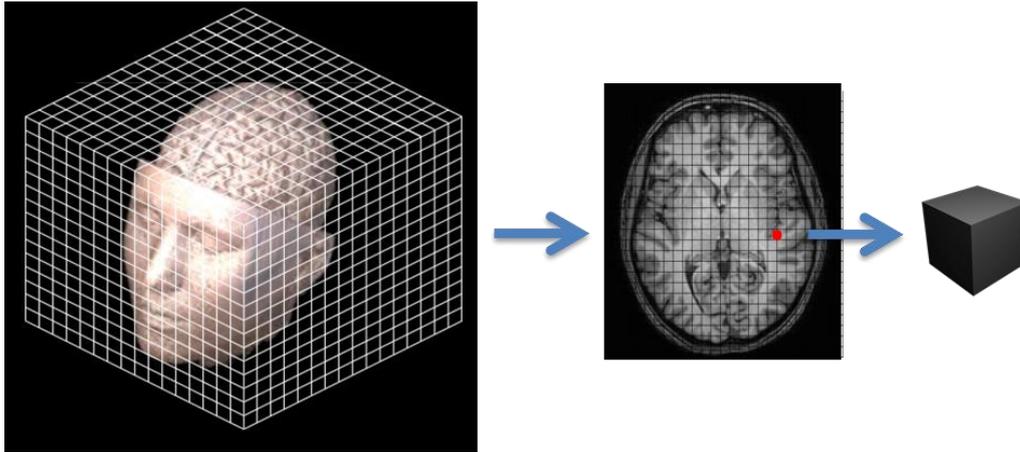


Figura 1. Representación tridimensional de la división en vóxeles de una imagen de Resonancia Magnética.

Una de las secuencias más utilizadas en RMf es la secuencia eco-planar (*Echoplanar Image, EPI*, en inglés). Ésta técnica de adquisición de RM tiene una elevada resolución temporal (a expensas de reducir la resolución espacial), lo cual es idóneo para RMf ya que nos permite obtener muchos volúmenes en poco tiempo.

Si durante la adquisición de la RMf el individuo no realiza ninguna actividad estaríamos hablando de RMf en reposo (r-RMf). Con este tipo de adquisición se posibilita evaluar la sincronización de fase en las señales de RM de distintas áreas cerebrales, asumiendo que regiones conectadas funcionalmente entre sí constituyen una red funcional ya que se comportan de la misma manera.

Esta técnica se empezó a utilizar a mediados de los años 90 para estudiar conectividad funcional (Biswal, Yetkin, Haughton, & Hyde, 1995; Biswal, Van Kylen, & Hyde, 1997; Lowe, Mock, & Sorenson, 1998). Con los avances en las metodologías de análisis de datos, a partir de 2007 hubo un resurgimiento de la r-RMf, y actualmente el número de estudios que se valen de esta técnica va en rápido aumento. Ha sido utilizada para el estudio de varios trastornos, tales como la enfermedad de Alzheimer (Krajcovicova, Marecek, Mikl, & Rektorova, 2014), trastorno depresivo mayor (Dichter, Gibbs, & Smoski, 2014), trastornos de ansiedad (Peterson, Thome, Frewen, & Lanius, 2014), enfermedad de Parkinson (Szewczyk-Krolkowski et al., 2014) y el trastorno bipolar (Wang et al., 2015).

1.2. Conectividad funcional

En un principio, cuando las investigaciones se centraban en el estudio de la activación del cerebro durante la realización de tareas, las fluctuaciones observadas en un cerebro en reposo fueron consideradas ruido (Mitra & Raichle, 2016). Sin embargo existe cada vez más evidencia de que las regiones cuya activación está sincronizada en reposo tienden a activarse juntas durante la realización de tareas (Biswal et al., 1995). A esta coactivación se le denomina conectividad funcional, y al darse simultáneamente entre múltiples regiones cerebrales genera estructuras jerárquicas que denominamos redes.

1.2.1. Redes de conectividad funcional

En personas sanas, el estudio de la conectividad funcional en estado de reposo mediante r-RMf ha identificado la existencia de redes funcionales a gran escala que interactúan dinámicamente. Éstas sirven de base fisiológica para explicar las zonas de activación cerebral durante la realización de tareas. Las redes de conectividad funcional cortical más comúnmente descritas hasta ahora son siete: la red visual, la motora, la atencional dorsal, la atencional ventral, la límbica, la frontoparietal (ejecutiva) y la de modo por defecto (Yeo et al., 2011). Después de esta parcelación, se realizaron dos investigaciones que dividieron en las mismas redes el cerebelo (Buckner, Krienen, Castellanos, Diaz, & Yeo, 2011) y el estriado (Choi, Yeo, & Buckner, 2012).

Existen múltiples maneras de estudiar las redes de conectividad funcional. Una de las más extendidas es mediante el uso de regiones de interés realizando una valoración de la correlación de la actividad entre estas regiones a lo largo del tiempo de adquisición. A partir de estos resultados es posible aplicar la teoría matemática de grafos, que consiste básicamente en el estudio de la organización de diversas regiones o nodos interconectados entre sí. A estas conexiones (valores de conectividad funcional) se les denomina puentes, mientras un conjunto de nodos y puentes forma una red. Según esta aproximación, se pueden obtener múltiples medidas tanto de los nodos (e.g. grado de conectividad, grado de centralidad, etc) como de las redes (segregación, integración, etc). Para una revisión más detallada ver (Bullmore & Sporns, 2009).

1.3. La correlación de señales promedio como medida de conectividad funcional

Como hemos mencionado anteriormente, la conectividad funcional está definida como la sincronización en la activación. Una de las medidas más empleadas en la literatura para definir matemáticamente la conectividad funcional entre dos señales temporales es el coeficiente de correlación de *Pearson*. Pero cuando una de las señales pertenece a una región (en vez de a un vóxel) no podemos obtener esta medida directamente, ya que cada región cuenta con varios vóxeles y cada uno de ellos tiene su propia señal temporal (véase figura 2). Hasta el momento, la forma más común de solucionar este problema consiste en obtener una señal promedio de la región, obteniendo la media aritmética de la activación de los vóxeles de ésta en cada instante temporal.

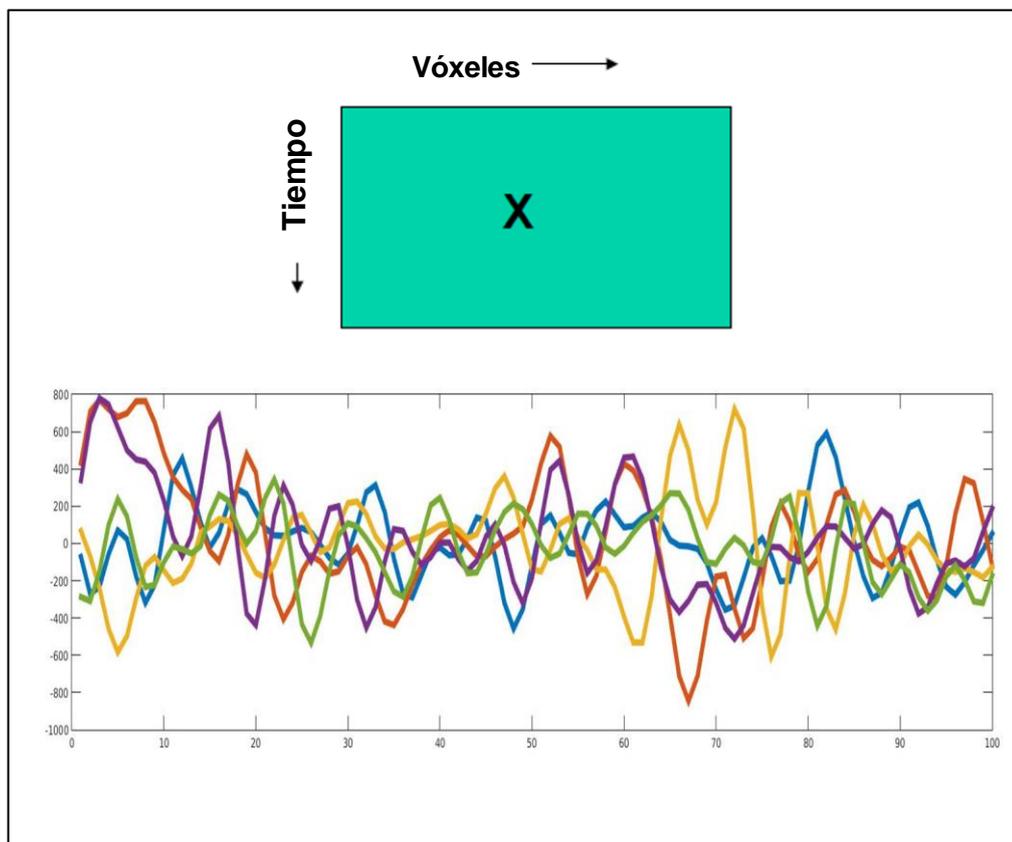


Figura 2. Arriba, matriz que representa una región, cuyas columnas se corresponden con cada uno de los vóxeles y las filas con los instantes temporales. Abajo la representación de esos 5 vóxeles en un gráfico.

Sin embargo, en determinadas circunstancias la media no tiene por qué ser un buen estimador sobre el cual resumir la información temporal de una región. Desde un punto de vista sustantivo, al utilizar la media de la región estamos asumiendo que tan solo hay una señal representativa de la región y, por tanto, que todos los vóxeles están involucrados en la misma tarea. En el caso de que esto no se cumpla, al realizar un promedio total estamos atenuando la variabilidad de las verdaderas señales representativas (Figura 3). Además de esto, como dichas señales pueden oscilar a distintas frecuencias, es posible encontrarse con que la señal promedio no es estable a lo largo del tiempo o que ocurran efectos de cancelación. Ello tendría como consecuencia a su vez que la conectividad funcional obtenida mediante esta señal y todas las técnicas de análisis derivadas de la conectividad entre dos regiones (como el análisis de redes) sean igual de inestables. Por último, esta metodología también asume que en todos los sujetos las regiones se dividen de la misma forma, mientras que se ha observado que en datos de r-RMf existe una gran variabilidad intersujeto (especialmente a pequeña escala) y cómo delimitar las regiones tiene efecto en los resultados (Sohn et al., 2015).

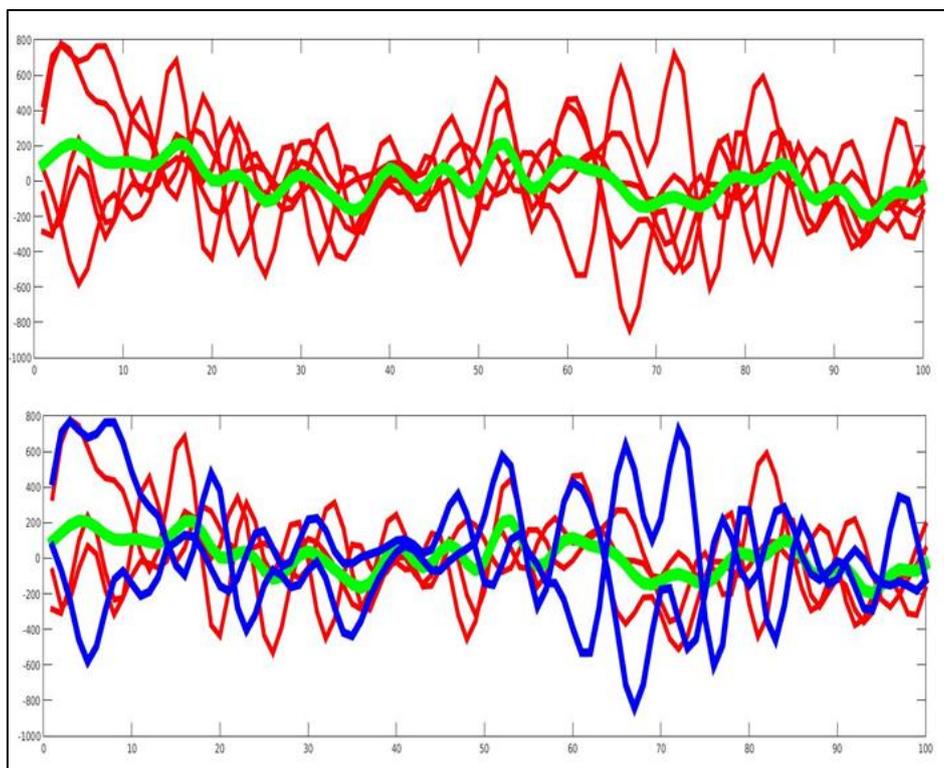


Figura 3. Arriba, la media de los 5 vóxeles de la Figura 2. Como se puede ver, la señal promedio ha perdido gran parte de la variabilidad que presentan los vóxeles por separado. Abajo, en azul, están representados dos vóxeles que muestran una correlación negativa ($r = -0.7$) y contribuyen por lo tanto a esta pérdida de variabilidad.

A modo de ejemplo, supongamos que una región está a su vez dividida en dos subregiones: en una de ella las neuronas están sincronizadas con la corteza motora y en la otra con la corteza visual. Al obtener una señal promedio total estamos atenuando (o incluso perdiendo) esta variabilidad, así como la de cada una de las verdaderas señales “promedio”. En la Figura 4 se muestran las señales correspondientes a los 10 vóxeles de una subregión, y en morado su señal promedio al incluir otras 10 señales con activación opuesta (a), no correlada (b) y sincronizada (c) respectivamente.

Por todo lo mencionado anteriormente, sería interesante evaluar si la región de la que queremos extraer la señal se divide en subunidades regionales, o lo que es lo mismo, si los vóxeles que la componen forman varios grupos con un señal promedio independiente cada uno. Para realizar estas divisiones puede emplearse el análisis clúster, también denominado análisis de conglomerados. A continuación se resumen estos procedimientos, deteniéndonos específicamente en aquellos que consideramos más relevantes para el presente trabajo.

1.4. Análisis Clúster

El análisis clúster es un conjunto de técnicas multivariantes empleadas para separar una muestra en subgrupos o clases en base a sus características. Su función es parecida a la del análisis discriminante y otras técnicas de clasificación, salvo porque en éstas las categorías de cada elemento se conocen previamente y por tanto buscan el algoritmo que mejor distinga unas de otras. Por el contrario, en el análisis clúster dichas categorías no se conocen *a priori* por lo que se trata de técnicas con un carácter predominantemente exploratorio. Por lo tanto, el análisis clúster incluye toda técnica cuya función sea agrupar un conjunto de objetos en un número de conglomerados o grupos según el parecido o similitud entre ellos (Everitt, Landau, Leese, & Stahl, 2011).

Existen multitud de algoritmos que producen conglomerados y cada uno ofrece resultados distintos dependiendo de su base matemática. Los más conocidos son las basadas en estimación de densidades, en jerarquías y en la teoría de grafos.

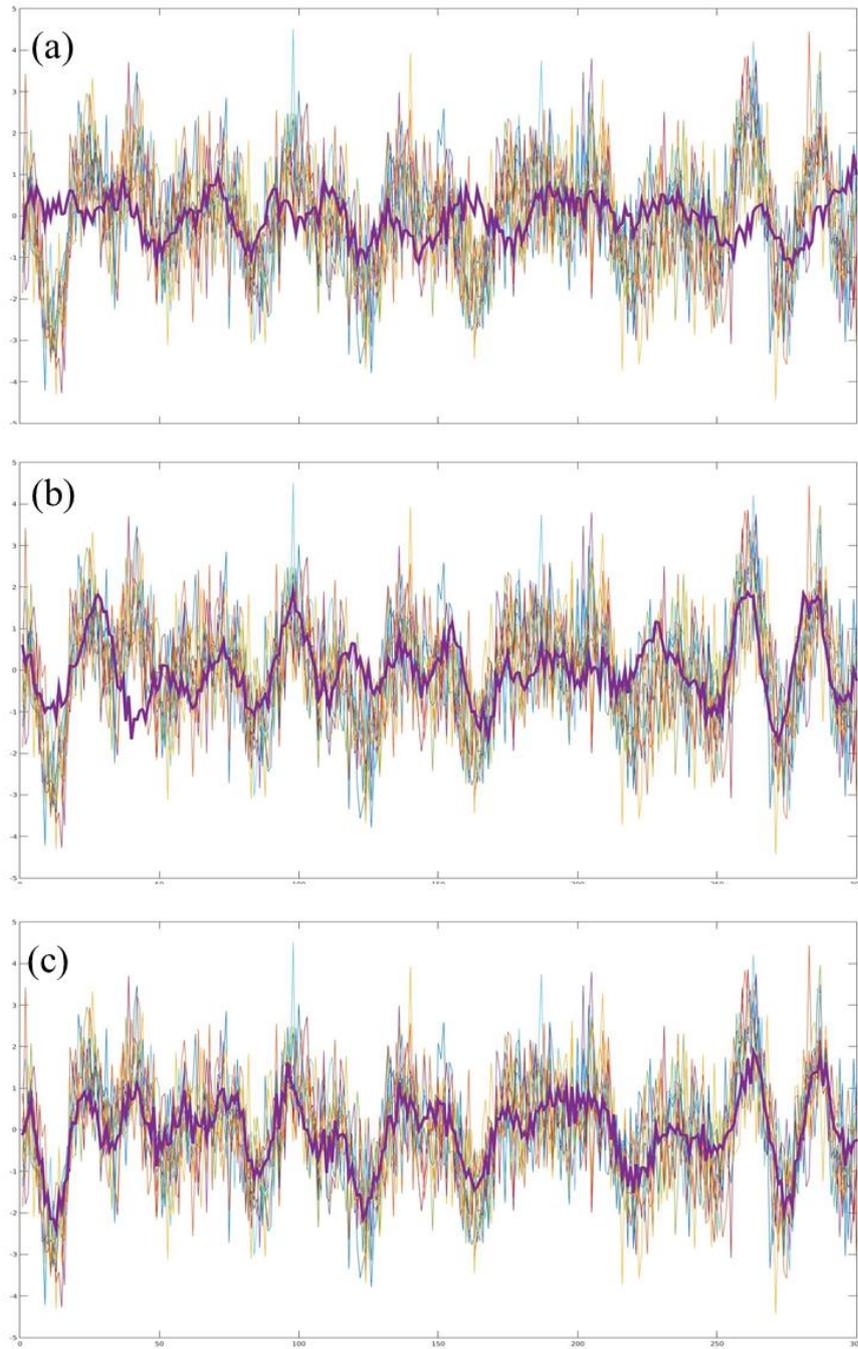


Figura 4. Ejemplo de una muestra de 10 señales simuladas al añadir ruido blanco a una señal original (promedio). En cada caso la señal principal (morado) es la media de estas 10 señales con otras 10 originadas por distintas señales promedio. En (a) han sido producidas por una señal correlacionada negativamente ($r=-0,69$) con la original. En (b) fueron generadas por una señal no correlacionada con la original ($r=-0,03$). En (c) la señal principal es el promedio de las 10 señales con otras 10 que han sido generadas a partir de una señal correlacionada positivamente con la original ($r=0,73$).

1.4.1. K-medias

La técnica K-medias (Lloyd, 1982) es un algoritmo iterativo de clústering no probabilístico que permite dividir una muestra en un número de particiones definido previamente utilizando una medida de distancia (normalmente euclidiana). Supongamos que tenemos un conjunto de datos multidimensional $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \dots \mathbf{x}_N\}$ de N elementos. Nuestro objetivo es obtener subgrupos con valores parecidos en estas variables, esto es, que su distancia a otros miembros del mismo grupo sea menor que su distancia a miembros de otros grupos. Para conseguirlo se tomará como referencia los centros de dichos grupos, obtenidos como su punto medio ($\boldsymbol{\mu}_k$) en el espacio D-dimensional.

La inicialización del algoritmo consiste en la asignación aleatoria del vector de centros $\boldsymbol{\mu}_k$ (donde k es el número predefinido de conglomerados) para luego proceder en dos pasos: primero asignando cada elemento al centro con una distancia menor y después recalculando los centros de cada grupo. Estos dos pasos se realizarán iterativamente hasta que no se produzca un cambio en las asignaciones de los elementos. En la Figura 5 se muestra la aplicación de este algoritmo iterativo a los datos *Old Faithful*¹.

Algoritmo:

1. Inicialización: Selección aleatoria de los centros $\boldsymbol{\mu}_k$.
2. Asignación de los elementos al centro más cercano.
3. Estimación de los nuevos centros $\boldsymbol{\mu}_k$ como el punto medio de cada clúster.
4. Evaluación del error, si cumple un criterio de parada predefinido terminar el algoritmo, de lo contrario realizar de nuevo los pasos 2 y 3.

Para formalizar esta definición utilizaremos como ejemplo la distancia euclidiana. Además del conjunto de datos \mathbf{x}_n , también tenemos un conjunto de variables binarias r_{nk} con valores 0 y 1, que describen a qué clúster k está asignado el elemento \mathbf{x}_n , por lo que $r_{nk} = 1$ y $r_{nj} = 0$ para $j \neq k$. El algoritmo trata de minimizar las distancias de cada punto \mathbf{x}_n a la media $\boldsymbol{\mu}_k$ de su grupo de pertenencia, y a esta función de dicha distancia se le denomina función de distorsión:

¹ La base de datos *Old Faithful* contiene información de la duración de las erupciones y del tiempo entre erupciones de los géiseres del tipo *Old Faithful* del Parque Nacional de Yellowstone, Wyoming, USA. Para más información ir a www.geyserstudy.org.

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x_n - \mu_k\|^2$$

Empleando la expresión anterior, primero se define μ_k aleatoriamente para después minimizar J en función de r_{nk} . Una vez realizado este paso se minimiza nuevamente la función de distorsión pero esta vez para estimar la nueva media μ_k , manteniendo fijo r_{nk} . Estos dos pasos se repetirían iterativamente hasta alcanzar el criterio de convergencia fijado.

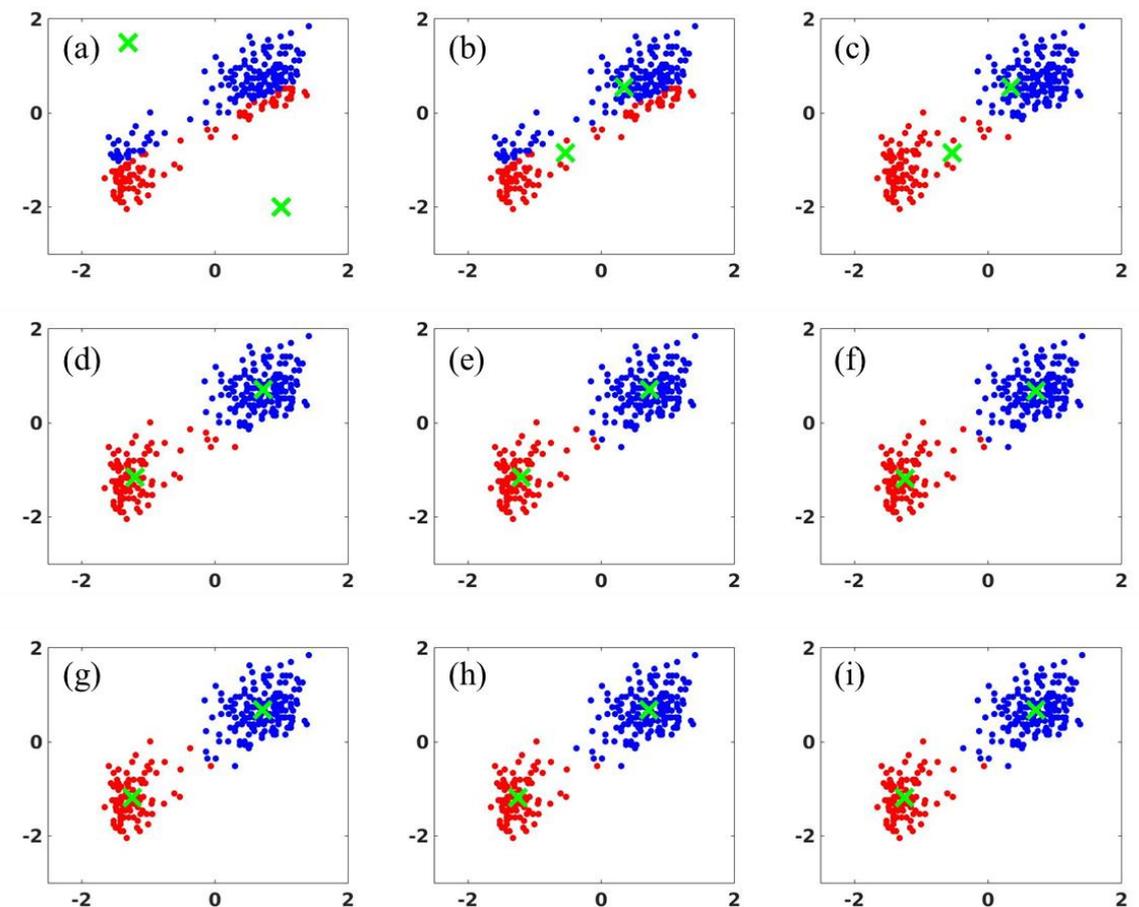


Figura 5. Ejemplo de aplicación del algoritmo K-medias aplicado al conjunto de datos *Old Faithful* en un espacio bidimensional. (a), (c), (e), (g) e (i) corresponden con los pasos de asignación, mientras que (b), (d),(f) y (h) el cálculo de los nuevos centros μ_k . Como en el paso (i) no se produce ningún cambio en la asignación de los elementos se termina el algoritmo.

K-medias es posiblemente la técnica de análisis clúster más famosa y la más utilizada, motivo por el que ha dado lugar a múltiples variantes. Algunas de las más utilizadas son k-medias++ (Arthur & Vassilvitskii, 2007), c-medias (Dunn, 1973) y X-medias (Pelleg & Moore, 2000), cada una diseñada para superar una carencia distinta del procedimiento K-medias. K-medias ++ es un algoritmo esencialmente igual que K-medias pero que incluye una inicialización más eficaz al utilizar como μ_k elementos del conjunto de datos original, tales que $\mu \in X$. Por su parte, c-medias es una versión probabilística del algoritmo original, por lo que los conjuntos de regiones dejan de ser excluyentes entre sí. Finalmente, X-medias es un algoritmo diseñado para obtener de manera automática el número óptimo de clúster k comparando criterios de información. En concreto, se compara el Criterio de Información Bayesiano (*BIC*, por sus siglas en inglés, (Schwarz, 1978), para el cual se requiere la función de verosimilitud. Dado el interés de este último procedimiento para nuestro trabajo, en el siguiente apartado se describe en más detalle.

1.4.2. X-medias y Máxima Verosimilitud

X-medias es también un algoritmo iterativo de dos pasos: uno en el cual optimiza los parámetros y otro en el que optimiza la estructura. La fase de optimización de parámetros consiste en correr el algoritmo K-medias para un valor de k predefinido, comenzando con $k = 1$. Una vez alcanzada la convergencia, se optimiza la estructura dividiendo cada clúster en dos y se comprueba si el ajuste del nuevo modelo es mejor que el del anterior. Estos dos pasos se realizan iterativamente hasta que se ha rechazado la subdivisión de todos los clúster existentes (Figura 6).

1. Estimación mediante K-medias para una k predefinida.
2. Dividir cada centro μ_k en dos y estimar 2-medias para los clúster obtenidos en el paso 1 por separado inicializados en los nuevos centros.
3. Comparar cada par de modelos; si la subdivisión ajusta peor en todos, terminar el algoritmo. En caso contrario, repetir los pasos 2 y 3.

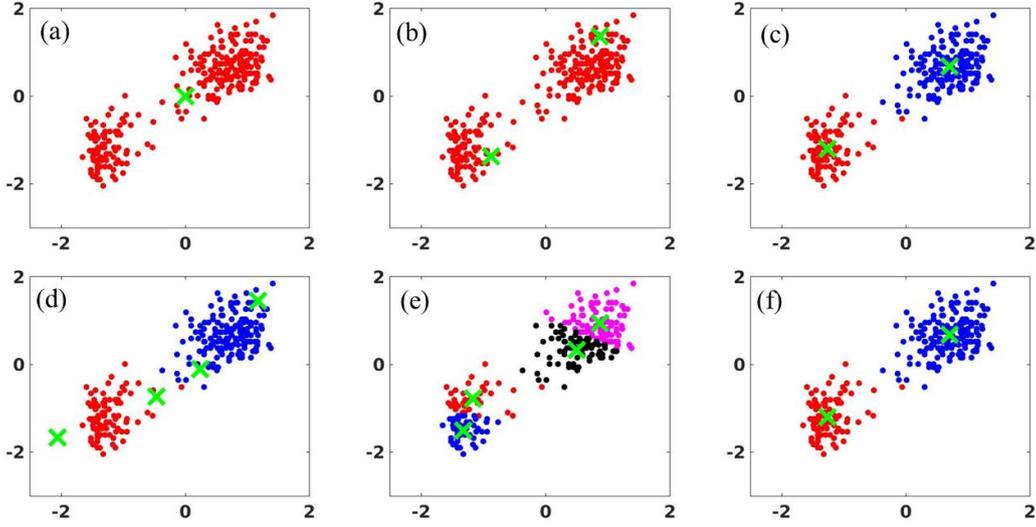


Figura 6. Ejemplo de la aplicación del algoritmo X-medias a la base de datos *Old Faithful*. En (a) comienza el algoritmo estimando K-medias para $k = 1$. En (b) se divide el centro en dos y en (c) se estima K-medias para $k = 2$ inicializado en esos centros. En los siguientes pasos se realiza el mismo procedimiento en cada uno de los clústers: en (d) la división del centro y en (e) la estimación para $k = 2$. Como en ambos clúster ajusta mejor el modelo de un clúster se rechazan las dos divisiones y en (f) se muestra el modelo definitivo.

En la fase de optimización de estructura se realiza la división de cada clúster y para eso se corre el algoritmo 2-medias (K-medias con $k = 2$) en sus elementos. Para acelerar la velocidad de cómputo los nuevos centros $\boldsymbol{\mu}_k$ se inicializan como los puntos del vector \vec{v} que pasa por el centro actual $\boldsymbol{\mu}_k$ y por el punto \boldsymbol{x}_{nk} más alejado a éste perteneciente al clúster k ($\max \text{dist}(\boldsymbol{x}_{nk}, \boldsymbol{\mu}_k)$) situados a $2/3$ de dicha distancia. Una vez ha convergido el algoritmo, se compara su ajuste con el del modelo antiguo de un solo clúster mediante el índice *BIC*, aunque esto requiere calcular la función de verosimilitud, ya que:

$$BIC(M_j) = \hat{l}_j(D) - \frac{p_j}{2} (\ln N)$$

Donde $\hat{l}_j(D)$ es el logaritmo de la función de verosimilitud del modelo j y p_j es el número de parámetros del modelo M_j . Para calcular la función de verosimilitud debemos considerar el resultado de k-medias como una mezcla de distribuciones normales (en la que cada clúster corresponde con una de estas gaussianas) con media $\boldsymbol{\mu}_k$ y matriz de varianzas $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}$ (ya que la minimización de errores da como resultado una distribución normal esférica y se asume que todas las distribuciones tienen la misma varianza, lo cual se conoce como esfericidad idéntica). El estimador máximo verosímil de esta varianza es:

$$\hat{\sigma}^2 = \frac{1}{N - K} \sum_n (\mathbf{x}_n - \boldsymbol{\mu}_k)^2$$

Las probabilidades de cada elemento \mathbf{x}_n corresponden al producto de la probabilidad a priori de que un elemento cualquiera pertenezca al clúster k y la probabilidad condicionada de x sabiendo que pertenece a k :

$$\hat{P}(\mathbf{x}_n) = P(k_{(n)}) \cdot P(\mathbf{x}_n | k_{(n)}) = \frac{N_{(k)}}{N} \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma})$$

Por tanto la función de verosimilitud es simplemente el producto de las probabilidades de todos los elementos (función de probabilidad conjunta) y la de log-verosimilitud es su logaritmo:

$$\hat{l}_j(D) = \ln \prod_n P(\mathbf{x}_n) = \sum_n \ln P(\mathbf{x}_n)$$

Teniendo la función de verosimilitud, podemos calcular el índice *BIC* y comparar por tanto los distintos modelos, corrigiendo por el tamaño muestral y el número de parámetros, priorizando modelos más parsimoniosos.

1.4.3. Modelos de Mezcla de Gaussianas y Expectación-Maximización

Como hemos visto al calcular la función de verosimilitud, el resultado que proporciona el procedimiento K-medias y sus variantes, puede ser visto como una mezcla de gaussianas bajo la asunción de esfericidad idéntica, y su forma de estimación puede también desarrollarse como maximización de probabilidades en vez de minimización del error. Como consecuencia, deshacernos de la asunción de esfericidad idéntica nos permitirá obtener modelos más flexibles que permitirían ajustarse mejor a los datos. Y si a esto le añadimos unas asignaciones suaves (probabilísticas en vez de categóricas) de los elementos a los clústers nos encontramos con lo que se consideran Modelos de Mezcla de Gaussianas (*GMM* por sus siglas en inglés, (Bishop, 2006)). El algoritmo K-medias, como veremos a continuación, no es sino la estimación de un caso concreto de este tipo de modelos. La distribución de mezcla de gaussianas viene dada por:

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Donde π_k es la probabilidad a priori $p(k)$ de que un elemento pertenezca al clúster k y viene dada por:

$$\pi_k = \frac{N_{(k)}}{N}$$

La otra parte de la fórmula de la función de densidad corresponde con la probabilidad condicionada de \mathbf{x} sabiendo que pertenece al clúster k . Por lo tanto, con el producto estamos en realidad calculando la probabilidad conjunta $p(\mathbf{x}, \mathbf{z})$, exactamente igual que hicimos con K-medias. Finalmente, el sumatorio se utiliza para obtener de la distribución condicionada la probabilidad marginal de \mathbf{x}_n y así obtener su función de densidad. Por lo tanto la función de log-verosimilitud es:

$$\begin{aligned} \hat{l}(D) &= \ln p(\mathbf{X} | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \ln \left\{ \prod_{n=1}^N \sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\} \\ &= \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\} \end{aligned}$$

Pero no se puede maximizar esta función de forma directa debido al sumatorio en el interior del logaritmo. Otro concepto importante que hay que mencionar es la *responsibility* de cada clúster, o su capacidad de explicar los datos ($p(\mathbf{z} | \mathbf{x})$) cuyo valor se obtiene mediante el Teorema de Bayes:

$$\gamma(z_k) = p(z_k | \mathbf{x}) = \frac{p(z_k) \cdot p(\mathbf{x} | z_k)}{p(\mathbf{x})} = \frac{\pi_k \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

Esta *responsibility* tiene un valor para cada elemento que aparece de forma natural al derivar la función de verosimilitud, estando implicada en la estimación máximo-verosímil

de la media y la varianza. Asimismo, se trata del equivalente a las asignaciones en K-medias, pero en forma probabilística.

$$\gamma(z_{nk}) = p(z_{nk}|\mathbf{x}) = \frac{\pi_k \cdot \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

Para maximizar la función de verosimilitud se puede realizar un algoritmo iterativo en dos pasos, estimando primero las *responsibilities* con los parámetros de las distribuciones existentes y recalculando después dichos parámetros a partir de las nuevas asignaciones. Estos pasos se repetirían hasta que el cambio en la función de verosimilitud alcance un criterio especificado. Este algoritmo se llama *Expectación-Maximización*, E-M (puesto que son las dos fases es en las que consiste), aunque en realidad es un ejemplo de su aplicación a *GMM* (Figura 7).

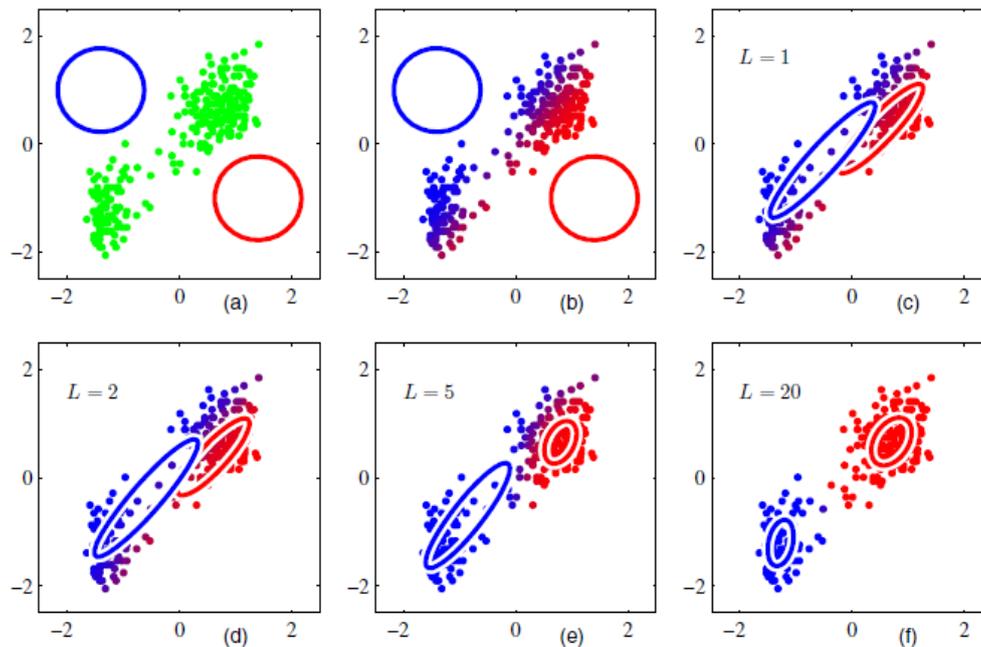


Figura 7. Ejemplo de la aplicación del algoritmo de EM en la base de datos *Old Faithful* (Bishop, 2006, pág. 437). al que se aplicó k-medias. (b) corresponde con el paso de Expectación, que es el paso de las asignaciones probabilísticas. Como se puede ver, en K-medias solo existían dos colores, mientras que aquí hay colores intermedios en función de las probabilidades de pertenecer a uno y otro clúster. A continuación, (c) se corresponde al resultado de la Maximización, que obtiene las medias y la matriz de covarianzas de las distribuciones de cada clúster. (d), (e) y (f) corresponden al resultado del algoritmo tras L iteraciones.

Como se puede ver, la aplicación de E-M a *GMM* es muy similar al procedimiento K-medias, y es que éste último es un caso concreto de aplicación de dicho algoritmo cuando todas las distribuciones son esféricas idénticas y $\sigma^2 \rightarrow 0$. En estas circunstancias, la probabilidad de un elemento de pertenecer a un clúster tiende a infinito para el de mínima distancia (Para una explicación más detallada ver (Bishop, 2006, págs. 423-459).

1.4.4. Clustering Jerárquico Aglomerativo

El clustering jerárquico comprende un grupo de técnicas que pretenden dividir los datos en una estructura por niveles basándose en una matriz de distancias (o similitud) entre los elementos. En el tipo aglomerativo se comienza considerando a cada elemento como un clúster y después se van uniendo los dos con distancia menor hasta que queda un número de clústeres predefinido. La información se representa en un dendrograma, que permite inspeccionar la jerarquía existente en los datos, así como la distancia acumulada al juntar clústeres (Figura 8). Este tipo de algoritmos de clustering ha sido ampliamente utilizados en datos de RM para clasificar las regiones del cerebro en redes funcionales (Bellec, Rosa-Neto, Lyttelton, Benali, & Evans, 2010). En RMf la matriz de similitud a la que se suele aplicar el algoritmo es la matriz de conectividad que representa la correlación entre la señal de cada par de vóxeles.

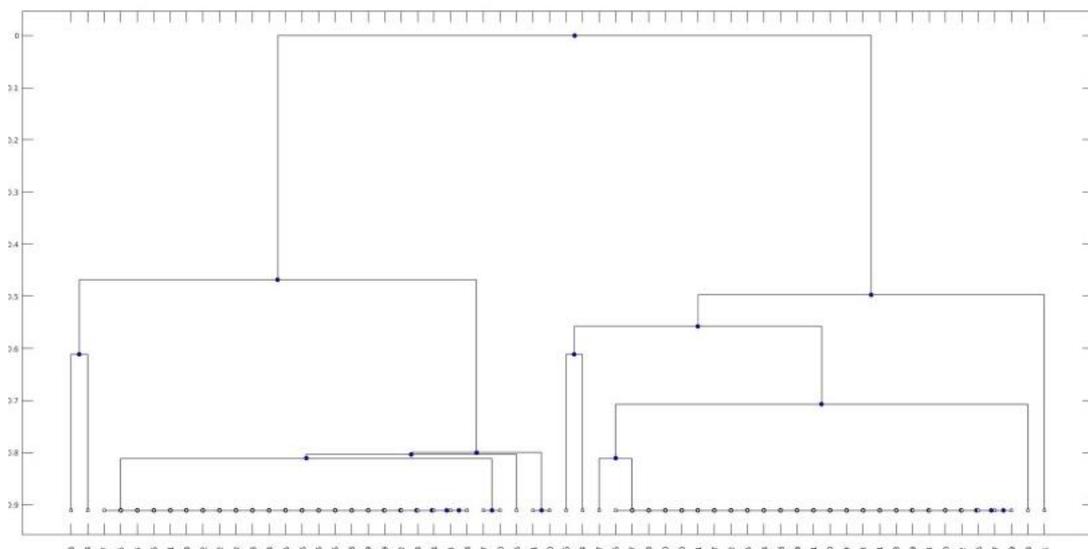


Figura 8. Ejemplo de un dendrograma que representa el error acumulado al unir cada par de clústers

1.5. Objetivo e hipótesis

El objetivo principal del presente trabajo es mejorar la medida que existe para resumir las señales de los vóxeles de una región en datos de r-RMf. Para ello se dividirá la región en subunidades regionales y se obtendrá una señal promedio de cada una. Este objetivo principal se desglosa en dos sub-objetivos:

1. Diseño e implementación de un método de clústering que realice de forma adecuada y estable la división de las regiones.
2. Comprobar si las señales promedio obtenidas del Modelo de la Región Dividida (MRD) son más estables y representan mejor la información de los vóxeles que la del Modelo de la Región Completa (MRC).

1.5.1. Hipótesis

El método propuesto se basa en la hipótesis general de que en una región del cerebro no todas las neuronas están encargadas de una misma tarea. Por el contrario, podemos dividir las subpoblaciones y obtener por tanto varias señales que representen la activación y desactivación de estas neuronas. Adicionalmente, se plantean dos hipótesis:

1. En el caso de que una región esté subdividida funcionalmente, la representatividad de la señal promedio será más baja. Esto se debe a que la media de dos variables que no están correlacionadas no recoge bien la variabilidad de éstas.
2. Además, esta representatividad será más estable. Si calculamos una señal promedio de una región que en realidad está funcionalmente subdividida, dicha señal contendrá la variabilidad de todos los subconjuntos y su relación con ellos será más inestable.

En un principio los análisis se realizarán en datos simulados para luego aplicar el método diseñado a datos reales y así evaluar la hipótesis principal de que las regiones están divididas.

2. MATERIAL Y MÉTODOS

En esta sección se presentará primeramente el “diseño enventanado” que nos permitirá obtener la medida de representatividad a lo largo del tiempo y será por tanto la clave al evaluar la estabilidad. A continuación se explicará el método propuesto para la división de las regiones de interés. Después se definirán matemáticamente las medidas de representatividad y estabilidad para evaluar el modelo de la región completa (MRC) y el modelo de la región dividida (MRD).

2.1. Diseño enventanado

El diseño enventanado permite obtener varias medidas de una señal a lo largo del tiempo. En datos de r-RMf se ha utilizado para ver cómo varía la conectividad funcional lo largo del tiempo (Hindriks et al., 2016), y nosotros lo vamos a utilizar para calcular la estabilidad de la señal. El objetivo del diseño es dividir una señal en secciones (o ventanas), que pueden estar o no solapadas. En la figura 9 se muestra un ejemplo de un diseño enventanado aplicado a una de nuestras señales de 1200 instantes temporales. En este tipo de diseño se selecciona una longitud z de la ventana w y un espacio a entre una ventana y la siguiente. La ventana w_1 comprendería el trozo de señal entre 1 y $(1 + z)$. La ventana w_2 por su parte constaría de los elementos de la señal entre $(1 + a)$ y $(1 + a + z)$. Y así sucesivamente hasta que el final de una ventana se encuentra en el último punto de la señal.

Tanto los datos simulados como los reales constan de 1200 instantes temporales, y hemos seleccionado $a = 40$ y $z = 600$, con lo que nos queda un total de 15 ventanas. El método que se describirá a continuación se aplicará a cada una de las ventanas, y esto nos permitirá obtener un valor de representatividad para cada vóxel en cada instante temporal. La estabilidad de un vóxel se calculará como la desviación típica de la representatividad en estos 15 instantes temporales.

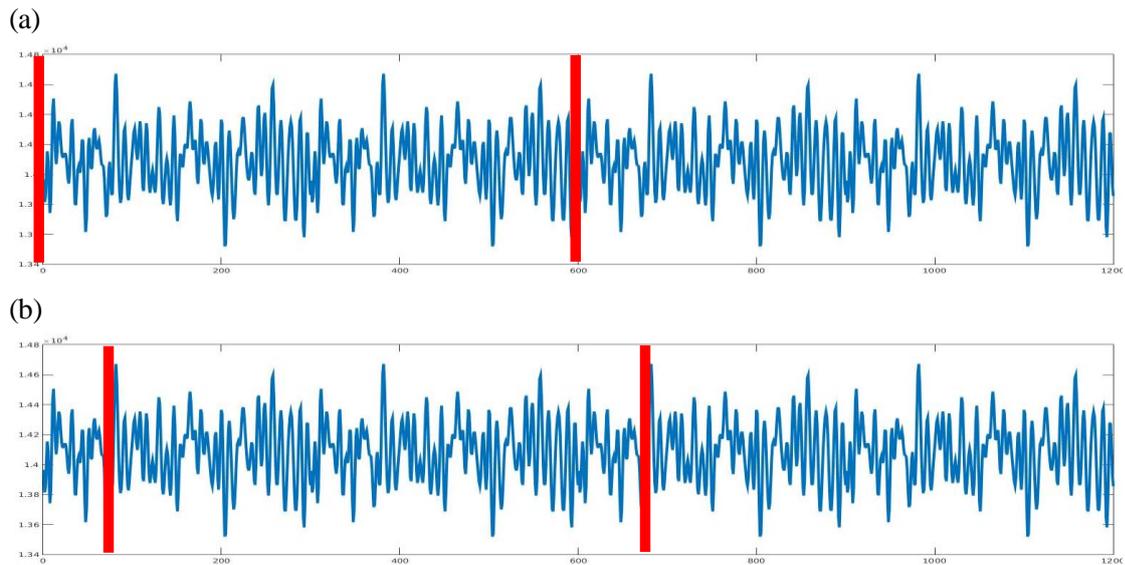


Figura 9. Ejemplo de un diseño enventanado en una de las señales del estudio para $z = 600$ y $\alpha = 40$. En (a) se muestra la primera ventana que empieza en el punto 1 y termina en el 601. En (b) se muestra la ventana que empieza en el punto 41 y termina en el 641.

2.2. Método empleado

El método empleado para el clústering de las regiones comprende el uso de varias técnicas que serán explicadas a continuación. Esta es la estructura general.

1. Obtención de varias muestras mediante un método de *bootstrap*.
2. Aplicación un algoritmo de clústering (*X-GMM*) a cada una de ellas para obtener una matriz de estabilidad.
3. Aplicación de un algoritmo de clústering jerárquico aglomerativo a la matriz de estabilidad para obtener los clúster definitivos.

2.2.1. Obtención de Clústers estables: *Circular block bootstrap*

Como hemos mencionado los datos de r-RMf suelen ser muy ruidosos, lo que hace que los resultados puedan ser inestables y dificulta su reproducibilidad. Una de las técnicas empleadas para obtener mayor precisión en las medidas y por tanto estabilidad es el remuestreo con reposición o *bootstrapping*. El *bootstrap* es una técnica no paramétrica

(con multitud de variantes) que consiste en la creación de nuevas muestras a partir de los datos para así obtener medidas más precisas. Sus variantes están creadas para adaptarse a distintos tipos de datos, y en el caso de series temporales (como es el caso de los datos de RMf) la más indicada es el *circular block bootstrap* (CBB, Efron, 1994).

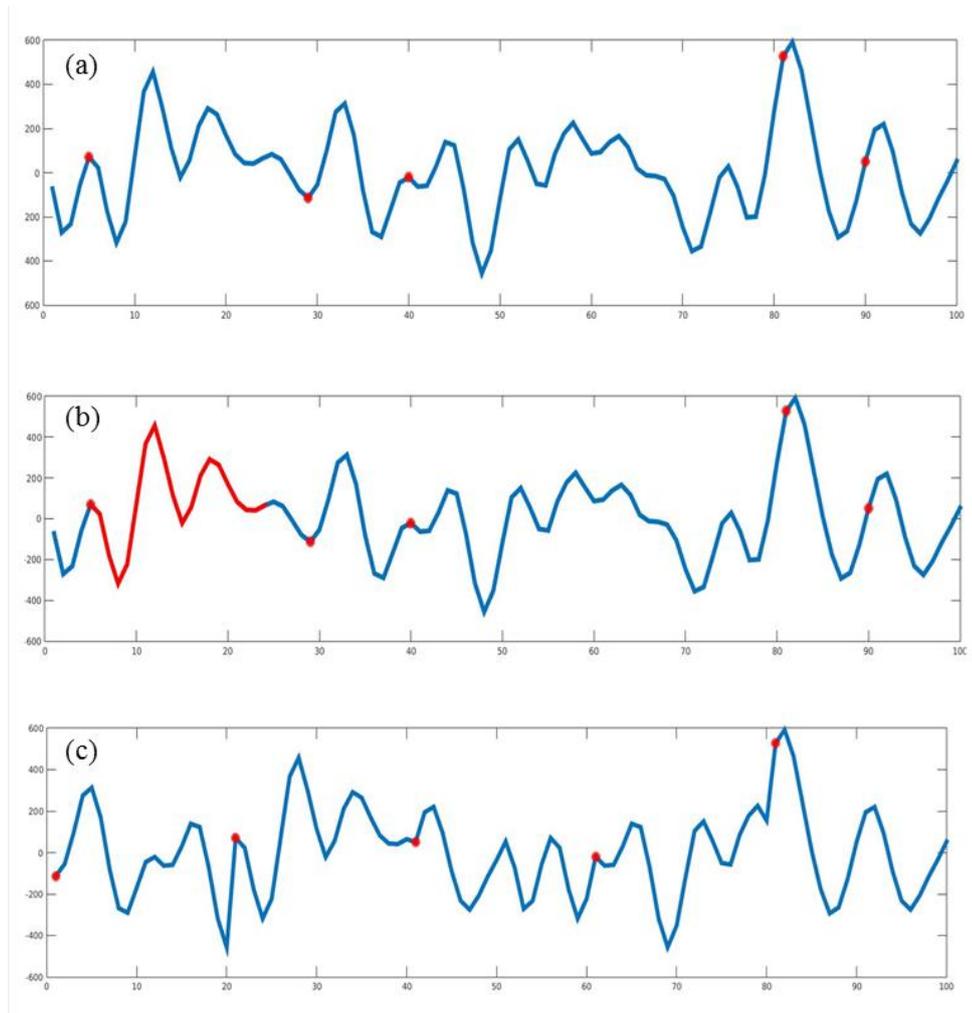


Figura 10. En (a) se muestra la señal original con los t puntos seleccionados aleatoriamente. En (b) se muestra la misma imagen con la sección en rojo que corresponde al bloque de tamaño l con inicio en t_1 . En (c) la nueva señal compuesta de las secciones de tamaño l comenzando desde los t puntos concatenadas. El orden de los puntos en la nueva señal es t_2, t_1, t_5, t_3, t_4 .

Dicha técnica respeta las dependencias temporales de los datos, y al ser realizada en todos los vóxeles por igual también las espaciales, algo fundamental en RMf (Bullmore et al., 2000). Si tenemos una señal de tamaño N , El CBB obtiene nuevas muestras temporales independientes de la misma dimensión compuestas de b bloques de tamaño l .

Dichos bloques corresponden al trozo de la señal desde el punto t obtenido aleatoriamente hasta $(t + l)$. En el caso de que $(t + l) > N$, la muestra se completará con la sección correspondiente del inicio de la señal (por eso es circular). En la figura 10 se muestra un ejemplo de cómo crear una señal temporal mediante esta técnica.

Aunque en el ejemplo de la Figura 10 solo aparece una señal, los inicios t y el orden de los bloques es el mismo para las señales de todos los vóxeles para una muestra b . Después se aplicará un algoritmo de clústering a cada una de las b muestras y se obtendrá como resultado una matriz de estabilidad $n \times n$ (donde n es el número de vóxeles) que representa la probabilidad de que cada par de vóxeles pertenezca al mismo clúster (Bellec et al., 2010).

2.2.2. *X-GMM*

El algoritmo de *clústering* que se va a aplicar a las muestras obtenidas mediante *CBB* para obtener las matrices de estabilidad está basado en dos de los explicados en la introducción. Sigue básicamente la misma estructura que *X-medias*, pero ofrece una mayor flexibilidad al realizar las particiones de los clúster con *GMM* en vez de con *k-medias*. Como se explica en la introducción, *X-medias* permite determinar automáticamente el número de clústeres comparando los *BIC* de un modelo simple y otro más complejo. Además de esto, se asegura una convergencia rápida al ir ramificando los clúster (dividiendo cada uno en dos sucesivamente) en vez de estimar directamente modelos completos. Al utilizar *GMM* en vez de *k-medias* para cada uno de los modelos se asegura un mejor ajuste a los datos en casos en los que existan covarianzas o diferencias en las varianzas de las dimensiones de los clúster.

El problema a la hora de estimar *GMM* utilizando como características las señales es que la dimensionalidad es muy alta, tanto como instantes temporales tengamos (que en nuestro caso son 600) y esto nos impedirá aplicarlo a regiones con menor número de vóxeles. Además esta alta dimensionalidad unido a las dependencias espaciales genera problemas de colinealidad, lo cual complica aún más la estimación y sugiere que podríamos reducir el número de variables. Para esto realizaremos un Análisis de Componentes Principales y extraeremos los componentes que sumen un 80% de varianza

explicada, aunque este valor puede ser menor en caso de que para llegar a él se requiera algún componente que explique menos del 1% de varianza. Las variables que se introducirán en el algoritmo *X-GMM* son los componentes extraídos con estos criterios (Ding & He, 2004).

Una vez se ha aplicado el algoritmo a las b muestras obtenidas mediante *CBB*, se calcula la matriz de estabilidad S como:

$$S_{ij} = \frac{k_{ij}}{b}$$

Donde k_{ij} corresponde al número de veces que los vóxeles i y j han sido asignados al mismo clúster. Una vez tenemos la matriz de estabilidad podemos aplicar en ella un algoritmo de clúster jerárquico aglomerativo. La decisión de aplicar este tipo de algoritmo se debe a que ha sido ampliamente utilizado en datos de r-RMf para encontrar redes funcionales (ya que estas obedecen a una estructura jerárquica) en todo el cerebro (Bellec et al., 2010). Y aunque en nuestro caso no estemos aplicándolo a todo el cerebro, las estructuras funcionales a pequeña escala en el cerebro son similares a las que se generan a gran escala (Downes et al., 2012).

Debido al diseño inventanado nos encontramos con que una misma región tiene 15 matrices de estabilidad distintas. Para calcular el número definitivo de clústers debemos aplicar el clústering jerárquico aglomerativo a una sola matriz de estabilidad, por lo que calcularemos una matriz \bar{S} que consistirá en el promedio de las 15 matrices de estabilidad. A ésta será a la que apliquemos el algoritmo de clústering aglomerativo que definirá los clústers.

Como el *clústering* jerárquico aglomerativo no proporciona el número de clústers de forma automática, debemos establecer un criterio para elegirlo. Utilizaremos el criterio de siluetas (Silhouette, (Rousseeuw, 1987)) que compara la distancia media de cada elemento a todos los de su mismo clúster con la distancia media de esos mismos elementos a los del clúster más cercano.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Donde $a(i)$ es la distancia media de un punto a su propio clúster y $b(i)$ es la distancia media de un punto al clúster más cercano. $s(i)$ está definido en el intervalo $[-1,1]$ y valores altos indican una buena separación entre cústeres. El criterio de siluetas del modelo completo consiste simplemente en la media aritmética de su valor en todos los elementos. El mayor inconveniente de este método es no poder evaluar el modelo de un solo clúster, ya que está basado en la comparación de distancias de cada punto con dos clúster. Sin embargo, consideraremos que este modelo proporciona mejores resultados si el valor mínimo en la matriz de estabilidad es de 0.75. Hemos basado este umbral en los resultados en los datos simulados ya que a partir de ese valor el clústering jerárquico aglomerativo comienza a fallar y la estabilidad no cambia (Ver resultados del estudio de simulación, apartado 3.1.2).

2.2.3. *Visión general del método*

El objetivo del método propuesto es dividir una región del cerebro en subregiones más pequeñas en base a sus señales funcionales. Como las señales de r-RMf son ruidosas oscilan en una combinación de frecuencia, es posible que esas divisiones cambien ligeramente a lo largo del tiempo. Por lo tanto debemos intentar obtener clústeres estables de cada región, para lo que realizaremos *CBB* y obtendremos 1000 muestras. Después, aplicaremos el algoritmo de clustering *X-GMM* a los componentes principales de cada una (por separado). Como resultado obtendremos la matriz de estabilidad \mathbf{S} , que contiene las probabilidades de que cada par de vóxeles pertenezca al mismo clúster según los resultados obtenidos al clusterizar las 100 muestras. Finalmente, a estas matrices de estabilidad les aplicaremos un clúster jerárquico aglomerativo para obtener las subregiones definitivas. En la Figura 11 se muestra un esquema de todos los pasos seguidos en el método propuesto, mientras que en el ANEXO 1 se puede ver un ejemplo de la aplicación y los resultados que ofrece cada uno de estos pasos en una región de un sujeto.

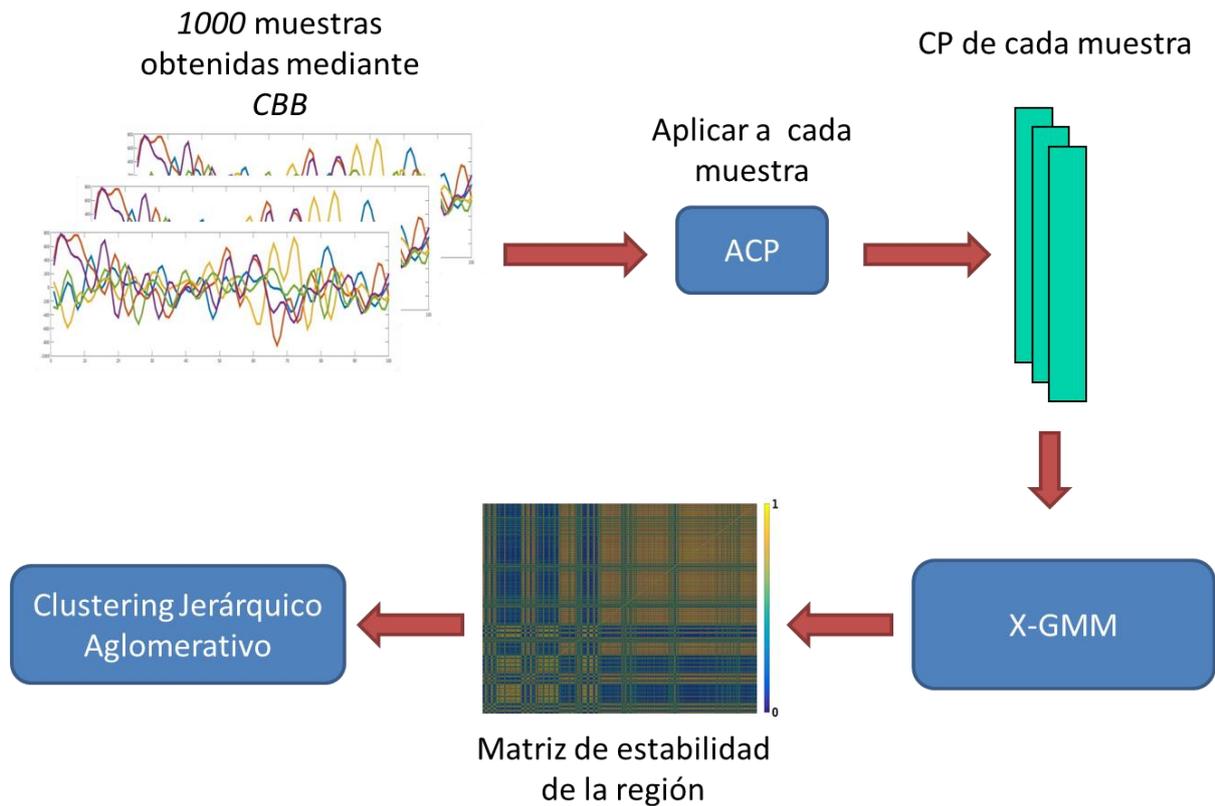


Figura 11. Visión general del método empleado para obtener las subregiones de una ROI.

2.3. Comparación entre modelos

Una vez tenemos las señales obtenidas para la *ROI* completa y para las subregiones pasaremos a medir la representatividad y su estabilidad a lo largo del tiempo. Supongamos que tenemos el modelo de la región completa x y el de la región dividida y . La representatividad la hemos definido matemáticamente como la correlación de la señal de cada vóxel con la señal promedio. Definiremos r_{ix} como la representatividad de un vóxel i en el modelo x , y la calcularemos como su valor medio a lo largo del tiempo (la media de su valor en las 15 ventanas). Por tanto \mathbf{r}_x será el vector que contiene los valores de representatividad del modelo x . Para comparar los dos modelos realizaremos una *Prueba t para muestras relacionadas*. También se calculará el tamaño del efecto *d de Cohen* ya que contamos con una muestra con un gran número de vóxeles (340 vóxeles en los datos simulados y 343 en los reales) y podemos obtener un resultado estadísticamente significativo incluso con una diferencia pequeña.

A continuación compararemos la estabilidad de la representatividad, esto es, si los vóxeles son representados por la señal en la misma medida a lo largo del tiempo. La estabilidad s_{ix} del vóxel i en el modelo x se representará a través de la desviación típica del valor de representatividad de un vóxel a lo largo del tiempo. En este caso la variable \mathbf{s}_x contendrá los valores de estabilidad del modelo x . En este caso comparamos desviaciones típicas, que no se distribuyen normalmente. Por esto para compararlas utilizaremos la *Prueba de Wilcoxon* que es la prueba no paramétrica utilizada para comparar dos muestras relacionadas. El tamaño del efecto lo calcularemos como *PS* (Grissom & Kim, 2012), la probabilidad de que un elemento de y (MRD) tenga un valor menor que uno de x (MRC).

$$PS = \frac{n_{y-}}{N}$$

2.4. Software

Para todos los análisis estadísticos así como la simulación de datos, se han utilizado funciones de elaboración propia en el software *MATLAB*². El del algoritmo desarrollado *X-GMM* puede encontrarse en el ANEXO 2. A continuación se incluye un listado de las funciones principales, que pueden encontrarse para consulta o uso en el CD entregado en secretaría:

- Carga de la *ROI* y creación del inventariado para el clústerizado. *clustering_rois.mat*.
- Circular Block Bootstrap. *cbb_clustering.mat*.
- *X-GMM*. *x_gmm_clustering.mat*.
- Clústering jerárquico aglomerativo. *hierarchical_clust.mat*.
- Simulación de datos con análisis paralelizado. *simulation_from_vars.mat*.
- Obtención de las matrices de representatividad de las regiones y sus subunidades regionales. *representativity_roi.mat*.

² *MATLAB Release 2015b, The Mathworks, Inc., Massachusetts, United States.*

3. RESULTADOS

3.1. Estudio en datos simulados

3.1.1. Modelo de simulación

El algoritmo se aplicó primero a datos completamente sintéticos para comprobar su adecuado funcionamiento, sus limitaciones y qué parámetros influyen más en sus resultados. Para ello se simuló una región de 340 vóxeles, de los cuales 170 estaban generados a partir de una señal promedio y otros 170 por una señal promedio distinta. La correlación entre estas señales promedio r_{xy} puede afectar tanto a la eficacia del algoritmo de clústering como a las medidas de representatividad y estabilidad, por lo que será un parámetro a variar en la simulación. La dimensionalidad temporal elegida fue la misma que en los datos reales, 1200 instantes temporales a los que se les aplicó el diseño enventanado con un tamaño de ventana de 600 y 40 puntos entre el inicio de una ventana y el de la siguiente. A las señales promedio se les añadió ruido simulando las diferencias en las señales de los vóxeles que pertenecen a la misma subregión (variabilidad intraclúster). Este ruido temporal presenta una distribución normal con $\boldsymbol{\mu} = 0$ y matriz de varianzas $\boldsymbol{\Sigma} = \mathbf{I}v\sigma^2$, donde σ^2 es la varianza de la señal promedio y v es un parámetro libre de escala. Este parámetro v es el que modificaremos para simular una mayor o menor similitud intraclúster. Una vez están simuladas las señales de todos los vóxeles se les añadió ruido espacial simulando el existente en un volumen de RM. Dicho ruido no sigue una distribución normal sino de Rice (Gudbjartsson & Patz, 1995), y varía en función de la relación señal-ruido (*SNR* por sus siglas en inglés). Los parámetros que se variaron fueron los siguientes (Tabla 1):

- La correlación entre las señales promedio r_{xy} , en un rango de lo que se suele obtener en datos de r-RMf.

- La relación señal ruido SNR , con el rango de valores que se suele encontrar en RM (Edelstein, Glover, Hardy, & Redington, 1986).
- El parámetro libre de escala v que regula la variabilidad intraclúster.

Parámetro	Rango	Intervalo
r_{xy}	(-0.6 - 0.7)	0.1
SNR	(5 - 30)	5
\sqrt{v}	(0.25 - 2.5)	0.25

Tabla 1. Valores de los parámetros de simulación.

Una de las ventajas de utilizar datos simulados es que conocemos a qué clúster pertenece cada vóxel. Para examinar el funcionamiento del método lo primero que haremos será evaluar el Porcentaje de Vóxeles Mal Clasificados ($PVMC$) y qué parámetros están relacionados con él. Después también comprobaremos si los parámetros tienen alguna relación con las medidas del tamaño del efecto de los contrastes. Recordemos que en el caso de la representatividad utilizaremos la d de Cohen mientras que en la estabilidad nuestra medida será PS .

Para medir estas relaciones ajustaremos tres modelos lineales: uno con $PVMC$, otro con d de Cohen y el tercero con PS como variable dependiente. Como regresores se incluirán los tres parámetros.

3.1.2. Relación de los parámetros con los resultados

El $PVMC$ es uno de los valores más importantes en datos simulados porque nos permite hacernos una idea de los errores que comete el método desarrollado. Al ser un método nuevo, antes de responder a nuestras hipótesis es necesario comprobar que el método funciona adecuadamente, así como bajo qué condiciones empieza a fallar. En la

Tabla 2 se puede ver que el porcentaje de errores de clasificación que comete el algoritmo aumenta cuando las variables de origen tienen una correlación más alta. Otra forma de decir esto es que el algoritmo comete más errores cuando la variabilidad interclúster es menor. El otro parámetro que está relacionado con los errores del método es ν , que representa la variabilidad intraclúster. Cuanto mayor es ν mayor es la variabilidad intraclúster y, según los resultados obtenidos, mayor es *PVMC*. La relación señal-ruido no parece estar relacionada con los errores cometidos por el método.

<i>PVMC</i>			
Variable	<i>t</i>	<i>p</i>	<i>r</i>²
<i>SNR</i>	-0.630	0.529	0
<i>r_{xy}</i>	6.366	<0.0001	0.058
<i>ν</i>	7.948	<0.0001	0.088

Tabla 2. Relación entre los parámetros y el *Porcentaje de Vóxeles Mal Clasificados*.

Los resultados de la Tabla 3 muestran que todos los parámetros están relacionados con la representatividad. Más concretamente, se puede observar que al aumentar la relación señal-ruido, las diferencias de representatividad entre MRD y el MRC también aumentan. Sin embargo, al aumentar la correlación entre las variables originales (r_{xy}) o al aumentar la variabilidad intraclúster (ν) se produce el efecto contrario.

Es importante destacar que el efecto de *SNR* es bastante menor estadísticamente hablando que el del resto de variables, de hecho ni siquiera explica el 1% de la varianza de

Representatividad			
Variable	<i>t</i>	<i>p</i>	<i>r</i>²
<i>SNR</i>	2.410	0.016	0.008
<i>r_{xy}</i>	-13.261	<0.0001	0.211
<i>ν</i>	-29.841	<0.0001	0.577

Tabla 3. Relación entre los parámetros y la diferencia en representatividad. la diferencia en representatividad.

En el caso de la estabilidad (Tabla 4) encontramos el mismo patrón que en representatividad, con la excepción de que la *SNR* en este caso no tiene un efecto estadísticamente significativo. Como este parámetro no está relacionado con los errores que comete el método ni con la diferencia en estabilidad los resultados que se mostrarán a continuación tan solo incluirán una $SNR = 20$, que es aproximadamente el valor que tienen nuestras imágenes de datos reales (Marcus et al., 2013). Además, aunque su relación con la representatividad sea estadísticamente significativa el porcentaje de varianza explicada no llega al 1%.

Estabilidad			
Variable	<i>t</i>	<i>p</i>	r^2
<i>SNR</i>	0.699	0.485	0
r_{xy}	-9.928	<0.0001	0.131
<i>v</i>	-68.798	<0.0001	0.878

Tabla 4. Relación entre los parámetros y la diferencia en estabilidad.

3.1.3. Evaluación del gold standard

El *gold standard* es un modelo con los “parámetros ideales” ($r_{xy} = 0$, $v = 0.25$) que utilizaremos para comprobar si, en caso de que la región efectivamente esté subdividida, la representatividad y la estabilidad son buenos indicadores.

En la Tabla 5 se puede ver que el método no cometió ningún error en la clasificación y que se produjo una mejora en representatividad y estabilidad. La correlación media de variables con la señal promedio (representatividad) es 59.31 desviaciones típicas superior en el *MRD* que en el *MRC*. Por otra parte, la estabilidad se muestra que la probabilidad de que la representatividad de un elemento a lo largo del tiempo varíe menos en el modelo dividido que en el completo es de 1.

<i>PVMC</i>	Representatividad (<i>d</i>)	Estabilidad (<i>PS</i>)
0%	59.31	1

Tabla 5. Resultados obtenidos en nuestro modelo *gold standard* que incluye los parámetros: $r_{xy} = 0$ y $v = 0.25$.

3.1.4. Evaluación del método

El *PVMC* muestra que el método empieza a cometer errores cuando existe una gran variabilidad intraclúster (para $v > 2$), en todos los demás casos ningún elemento se clasificó erróneamente. Y aun con valores altos de v si la correlación entre las variables es inferior a 0.5, el *PVMC* es muy bajo, de entre el 0 y el 5%. Sin embargo, si la correlación entre las variables supera este valor el porcentaje sube hasta un 21.76% en el caso de $v = 2.25$ y 30.59% para $v = 2.5$ (Figura 12). Por lo tanto, para que el método falle no basta con que haya una gran variabilidad intraclúster o una baja variabilidad interclúster, sino que es requisito indispensable que se den ambas condiciones.

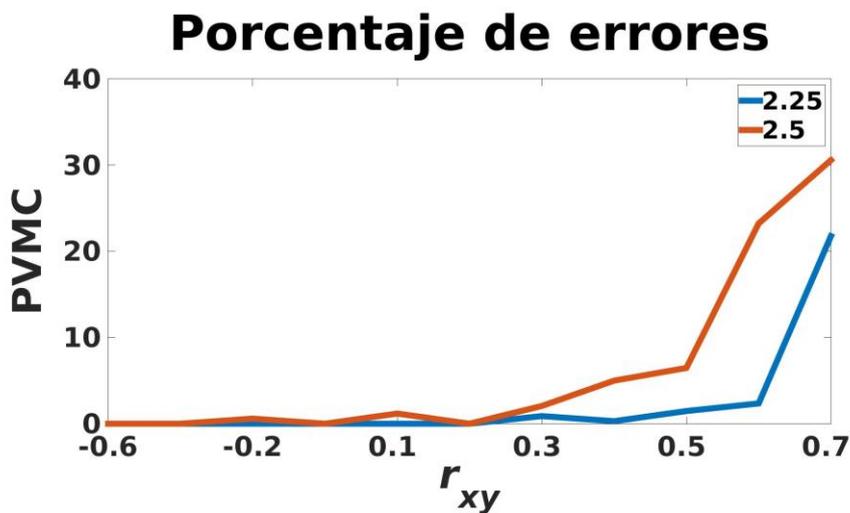


Figura 12. Variación del *PVMC* en función de la variabilidad interclúster para $v = 2.25$ y $v = 2.5$

3.1.5. Evaluación de las medidas de estabilidad y representatividad

La Figura 13 muestra cómo varían los resultados al modificar los parámetros de la simulación. Es importante destacar qué valores de los parámetros hacen que las diferencias de estabilidad entre MRD y MRC desaparezcan. Para $\nu = 2.25$, $\nu = 2.5$ el valor de PS está fluctuando alrededor de 0.5 (e incluso en $\nu = 2$; $\overline{PS} = 0.524$), lo cual nos indica que no hay cambio en la estabilidad y, por tanto, que en estos casos las dos medidas son igual de fiables. Para estos valores de ν aparecen matrices de estabilidad con todas las probabilidades iguales a 1, señal de que $X\text{-GMM}$ ha reconocido en las 1000 muestras que el modelo de 1 clúster es el que mejor ajustaba. En los casos en los que la matriz tenía valores distintos, el mínimo valor de probabilidad oscilaba entre 0.75-0.975, por lo que todos los elementos tenían una alta probabilidad de pertenecer al mismo clúster.

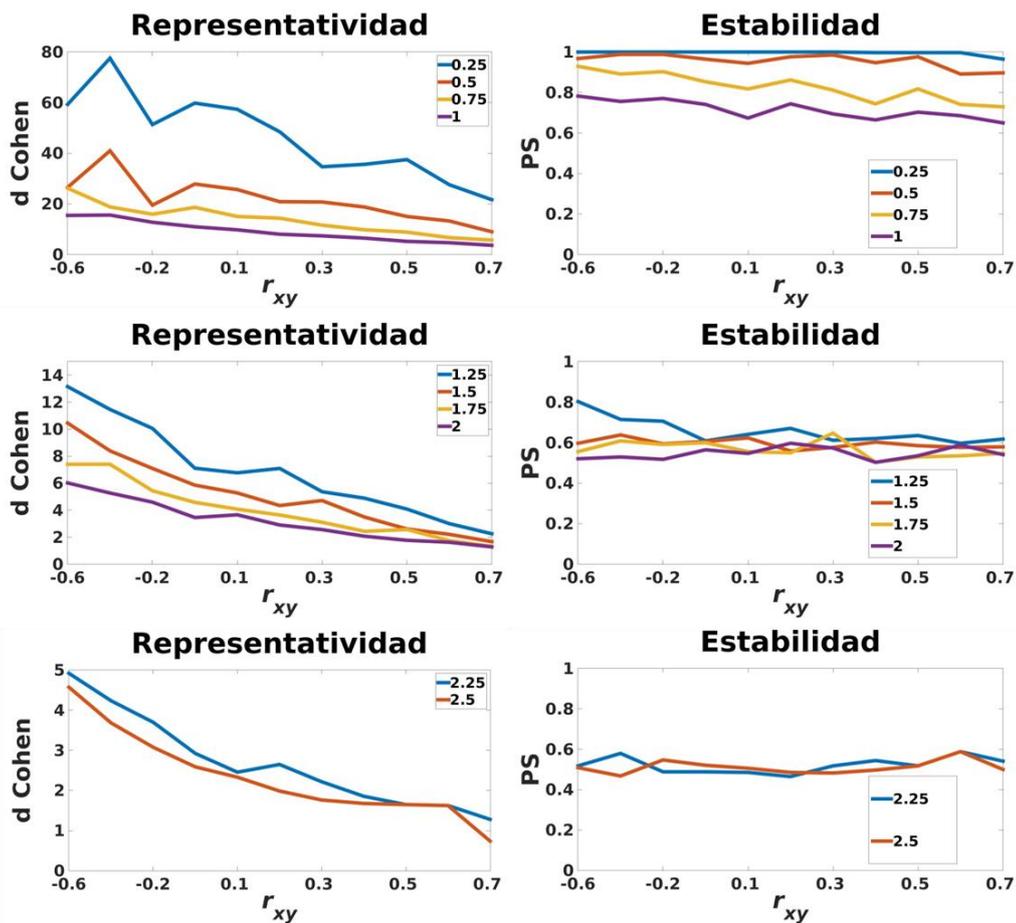


Figura 13. Resultados de representatividad y estabilidad en función de los parámetros. En el eje y se representa la correspondiente medida del tamaño del efecto y en el eje x la variabilidad interclúster. Las diferentes líneas de las gráficas representan las regiones con los distintos valores de ν que aparecen en la leyenda.

A partir de estos resultados se ha establecido el criterio de considerar un solo clúster cuando el valor mínimo en la matriz de probabilidad sea igual a 0.75. Los gráficos de la Figura 13 reflejan que la variabilidad intraclúster es lo que más influye en los resultados, llegando al extremo de anular el cambio en estabilidad y hacer el cambio en representatividad más de 20 veces menor.

3.2. Estudio en datos reales

3.2.1. Participantes

La muestra utilizada consta de 100 sujetos elegidos al azar de la base de datos abierta *Human Connectome Project* (Van Essen et al., 2013). www.humanconnectome.org), pero 6 de ellos tuvieron que ser retirados de los análisis posteriores (uno por presentar una anomalía intracraneal y otros cinco por distorsión en la imagen). El rango de edad es de entre 20 y 35 años.

3.2.2. Parámetros de adquisición

Todos los sujetos fueron escaneados en una Siemens 3T “*Connectome Skyra*” personalizada. En el caso de las imágenes funcionales se utilizó una secuencia multibanda de corte acelerado para poder adquirir más imágenes en menos tiempo. Esto permite que mientras el tiempo de repetición (TR, tiempo entre que se adquiere un volumen y el siguiente) suele oscilar entre 2-3 segundos, en estas imágenes es de 0.7 segundos. El tamaño del vóxel de estas imágenes es de 1x1x1 mm y se adquirieron un total de 1200 volúmenes por sujeto. Las imágenes estructurales fueron adquiridas en el mismo escáner, y su tamaño del vóxel es de 1x1x1 mm.

3.2.3. Preprocesado

A las imágenes de r-RMf se les realizó el procesado estándar, una primera parte se llevó a cabo por los investigadores que realizan el *Human Connectome Project* (Glasser et al., 2013) y otra en el Laboratorio de Imagen Médica. La primera parte constó de la corrección de un gradiente no lineal, el realineado y una normalización a espacio MNI, que transforma todas las imágenes a un mismo espacio con tamaño de vóxel 2x2x2 mm. La segunda parte del preprocesado incluyó el correregistro, un *Detrending*, un filtro pasa-banda, la normalización de intensidad media y un modelo lineal general en el que se incluyeron como regresores los 6 parámetros de movimiento, la señal media de sustancia blanca, la de líquido cefalorraquídeo y la de todo el cerebro para eliminar la variabilidad debida a estas variables. Las señales de cada vóxel son por tanto los residuos de este modelo.

3.2.4. Selección de las Regiones de Interés

Las regiones del cerebro tienen distintas características relacionadas con la función que desempeñan. Dependiendo de si las regiones pertenecen a redes sensoriales (con funciones primarias) o a redes asociadas a funciones más complejas (red de modo por defecto, ejecutiva...) se encuentran diferencias en la conectividad en sí mismas y con otras regiones (Gu et al., 2015). Por lo tanto la elección de las regiones de interés puede ser un factor que condicione las subregiones de división. En nuestro caso hemos elegido dos regiones motoras (izquierda y derecha) asociadas a funciones primarias y dos de la red de modo por defecto (Córtex Cingulado Posterior o CCP y Córtex Prefrontal medial o CPFm), que están asociadas con funciones complejas (Figura 14). Las características de las regiones quedaron resumidas en la Tabla 5. Las coordenadas que aparecen corresponden a las de máximo valor Z obtenido al buscar las regiones en un metaanálisis de estudios de RMf (www.neurosynth.com) y son, a su vez, el centro de nuestras regiones.

ROI	Coordenadas (x,y,z)	Tamaño	Número vóxeles
CCP	(0,-52,25)	14x14x14 mm	343
CPFm	(0,51,-4)	14x14x14 mm	343
Mi	(-35,-22,59)	14x14x14 mm	343
Md	(34,,23,57)	14x14x14 mm	343

Tabla 5. Características de las Regiones de Interés (ROIs).

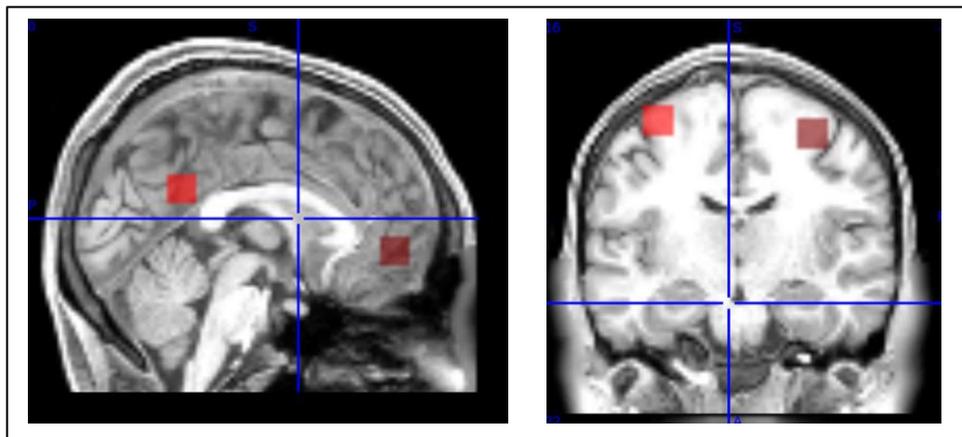


Figura 14. A la izquierda, las dos regiones de la red de modo por defecto: cíngulo posterior y córtex prefrontal medial (de izquierda a derecha). A la derecha, las dos regiones de la red sensoriomotora: la corteza motora izquierda y derecha.

3.2.1. Resultados del estudio en datos reales

Lo primero que cuantificaremos en los datos reales es en cuántas regiones se han detectado subunidades regionales. La Tabla 6 muestra que las regiones CPFm y Mi son las que en mayor número de sujetos se subdividen, en 41 (43.62%) y 44 (46.8%) casos respectivamente, mientras que CCP 16 (17.02%) y Md en tan solo 8 sujetos (8.56%).

A diferencia de los resultados en datos simulados, en los datos reales no cuantificaremos el tamaño del efecto sino que observaremos los porcentajes de contrastes

estadísticamente significativos. Las pruebas *t* y *Wilcoxon*, descritas en detalle en la sección de métodos (Apartado 2.3) se utilizaron para comparar las diferencias en estabilidad y representatividad entre el MRD y el MRC para cada una de las regiones que resultó dividida. Por lo tanto en sus respectivas columnas podemos encontrar en qué porcentaje de esas regiones se rechazó la hipótesis nula de igualdad de medias, y entre paréntesis el porcentaje respecto al total. La dirección de todos los contrastes fue la esperada (mayor representatividad y estabilidad en MRD que en MRC).

Región	Número (Porcentaje del total)	Representatividad (Porcentaje del total)	Estabilidad (Porcentaje del total)
CCP	16 (17.02%)	100% (17.02%)	62.5% (10.64%)
CPFm	41 (43.62%)	97.65% (42.59%)	24.39% (10.64%)
Mi	44 (46.8%)	100% (46.8%)	25% (11.7%)
Md	8 (8.56%)	100% (8.56%)	37.5% (3.21%)

Tabla 6. Resultados obtenidos en datos reales. Las columnas de representatividad y estabilidad representan el porcentaje de regiones en los que se encontraron diferencias significativas de las que fueron divididas.

4. DISCUSIÓN

La obtención de una señal que represente de manera estable la actividad de una región del cerebro constituye un vacío en la literatura de RMf. Hasta la fecha el indicador más utilizado es la media. El objetivo principal de este trabajo es proponer un método alternativo que presente mayor representatividad y estabilidad. Principalmente, se dividió las *ROIs* y se obtuvo la señal promedio de estas subunidades regionales. Se esperaba que dichas señales promedio fuesen más representativas de los vóxeles y que esta representatividad fuese más estable a lo largo del tiempo que la señal de la región completa. Para la división de la región se ha desarrollado un método de clústering diseñado para obtener clústers estables, que previamente fue puesto a prueba con datos simulados.

4.1. Evaluación del método: resultados en datos simulados

Este método ha proporcionado buenos resultados en los datos simulados. Fue capaz de identificar las regiones sin errores bajo diversas condiciones y se ha mantenido robusto ante las variaciones en *SNR*. Sin embargo, cuando la variabilidad intraclúster es muy alta y la interclúster muy baja es cuando el *PVMC* aumentó, pero en estas circunstancias la división de la región puede perder su sentido. De hecho no solo han aumentado los errores de clasificación, sino que la mejora en estabilidad deja de existir. Esto nos lleva a pensar que el modelo de un solo clúster es mejor que el dividido en estos casos (lo cual es también coherente con el criterio de parsimonia).

Tal como se esperaba, la división de las regiones de datos simulados vino acompañada de un aumento tanto de la estabilidad como de la representatividad. La mejora en representatividad fue de hecho ubicua, aunque disminuyó al aumentar la variabilidad intraclúster y disminuir la interclúster, especialmente cuando el *PVMC* fue mayor. En el caso de la estabilidad encontramos el mismo patrón, aunque tal como mencionamos

anteriormente con gran variabilidad intraclúster y baja interclúster se pierde a diferencia *MRD* y *MRC*.

Además, el correcto funcionamiento del método invita a probarlo en otros datos de RM. Dentro del campo de r-RMf podría utilizarse en la detección de redes funcionales a gran escala y comparar sus resultados con los obtenidos mediante otros algoritmos de clústering (Bellec et al., 2010; Yeo et al., 2011) o análisis de componentes independientes (Kiviniemi et al., 2009), ya que el estudio de dichas redes funcionales tiene aplicaciones en múltiples patologías (Coutinho et al., 2016; Mingoia et al., 2012; McLeod, Langevin, Dewey, & Goodyear, 2016). También es posible aplicar el método a la detección de redes en datos de RM estructural, adaptando el *bootstrap CBB a datos no temporales* (Efron, 1994). Además de la aplicación del método completo, también se puede emplear la matriz de estabilidad en los diversos métodos de teoría de grafos (Sepulcre, Sabuncu, Yeo, Liu, & Johnson, 2012; Shen, Papademetris, & Constable, 2010) y comparar los resultados con los obtenidos al utilizar la matriz de correlación. Nosotros en este trabajo lo hemos aplicado a la detección de subunidades regionales dentro de *ROIs* funcionales, un campo poco explorado por el momento.

4.2. Aplicación del método en datos reales.

Según los resultados obtenidos, las regiones no presentan un alto porcentaje de división. La región en la que más porcentaje de sujetos presentó subunidades regionales fue la Mi, con menos de un 50% de sujetos mostrando dicha subdivisión.

Además, las regiones en las que este porcentaje fue más alto no tienen una función parecida ya que una pertenece a la red motora y la otra a la red de modo por defecto. Esto descarta que la división refleje puramente medidas como la homogeneidad regional (Zang, Jiang, Lu, He, & Tian, 2004; Lopez-Larson, Anderson, Ferguson, & Yurgelun-Todd, 2011) y apunta a otros factores implicados que se discuten en el siguiente apartado.

Respecto a si las señales obtenidas en el MRD son más representativas y estables que la del MRC, los resultados son contradictorios. En el caso de la representatividad, sí

que hemos encontrado un aumento significativo en prácticamente todas las regiones que fueron divididas. Por el contrario, en la mayoría de los casos la estabilidad no fue significativamente mejor en MRD que en MRC. Este hecho va en contra de nuestras hipótesis iniciales y contradice los resultados obtenidos en datos simulados, lo cual puede deberse a varias causas que se presentan a continuación.

4.3. Posibles discrepancias entre los resultados obtenidos en datos simulados y en los reales

La simplicidad requerida por los datos simulados para evaluar el método hace difícil la transición a los complejos datos de r-RMf. Los datos de r-RMf incluyen, entre otras fuentes de ruido, el movimiento del sujeto durante la adquisición (Power et al., 2014) o el ruido fisiológico (frecuencia cardíaca y cardíaco o la frecuencia respiratoria principalmente, (Kruger & Glover, 2001)). Al utilizar las señales de RMf para el clústering (o sus componentes principales en nuestro caso) estamos también clusterizando en función de ese ruido. Esta puede ser la causa además de que los resultados en datos sintéticos tengan un tamaño del efecto tan grande. .

Para evitar estas fuentes de ruido, en futuros estudios, proponemos seleccionar otras características de la señal en vez de sus componentes principales. Como posibles alternativas se puede incluir la descomposición de la señal en frecuencias mediante *wavelets*, las coordenadas de los vóxeles o características de la distribución de cada vóxel.

A pesar de todo esto existe cierta evidencia de que la conectividad funcional entre regiones no es estable a lo largo del tiempo (Hindriks et al., 2016). Nuestra hipótesis sugería que estas variaciones podían deberse a que la señal obtenida para representar las regiones no era estable por la existencia de subunidades regionales. Sin embargo, aunque nuestros resultados apuntan a que ambos hechos no están relacionados, no podemos olvidar que nuestro método de evaluación de representatividad y estabilidad se basa en la conectividad funcional. En nuestro caso las unidades de estudio son vóxeles en vez de regiones, pero muy probablemente, los cambios que suceden en el cerebro a gran escala

también sucedan a nivel de voxel. Lo mismo sucede con las fuentes de ruido, que pueden contaminar tanto el procedimiento de clústering como la evaluación de sus resultados, por lo que no podemos determinar con certeza si la estructura de subunidades regionales es errónea.

4.4. Limitaciones

Algunas de las limitaciones del estudio ya se han mencionado anteriormente: las numerosas fuentes de ruido en r-RMf y el desconocimiento de patrones claros de conectividad funcional. Además, hay muchos más parámetros que se podrían añadir a la simulación, como son otras fuentes de ruido, un modelo auto regresivo para simular las señales de r-RMf, distintos tamaños de la región o modelos de correlación con más variables promedio. Otros parámetros a variar en los datos reales son los relativos al procesado y los tamaños de la región.

Este trabajo es el primero realizado que intenta proponer una alternativa metodológica a la señal promedio de una región. Esto nos ha obligado a centrarnos en la parte más básica de elección y desarrollo del método. En el presente proyecto de fin de máster proponemos una primera aproximación a un campo en el que apenas hay literatura al respecto, por lo que la selección y optimización de la técnica adecuada está por determinar.

REFERENCIAS

- Arthur, D., & Vassilvitskii, S. (2007). *k-Means++: the advantages of careful seeding*. Paper presented at the Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.
- Bellec, P., Rosa-Neto, P., Lyttelton, O. C., Benali, H., & Evans, A. C. (2010). Multi-level bootstrap analysis of stable clusters in resting-state fMRI. *Neuroimage*, *51*(3), 1126-1139. doi:10.1016/j.neuroimage.2010.02.082
- Bishop, C. M. (2006). *Pattern Recognition & Machine Learning*: Springer.
- Biswal, B., Yetkin, F. Z., Haughton, V. M., & Hyde, J. S. (1995). Functional connectivity in the motor cortex of resting human brain using echo-planar MRI. *Magn Reson Med*, *34*(4), 537-541.
- Biswal, B. B., Van Kylen, J., & Hyde, J. S. (1997). Simultaneous assessment of flow and BOLD signals in resting-state functional connectivity maps. *NMR Biomed*, *10*(4-5), 165-170.
- Buckner, R. L., Krienen, F. M., Castellanos, A., Diaz, J. C., & Yeo, B. T. (2011). The organization of the human cerebellum estimated by intrinsic functional connectivity. *J Neurophysiol*, *106*(5), 2322-2345. doi:10.1152/jn.00339.2011
- Bullmore, E., Horwitz, B., Honey, G., Brammer, M., Williams, S., & Sharma, T. (2000). How good is good enough in path analysis of fMRI data? *Neuroimage*, *11*(4), 289-301. doi:10.1006/nimg.2000.0544
- Bullmore, E., & Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nat Rev Neurosci*, *10*(3), 186-198. doi:10.1038/nrn2575
- Choi, E. Y., Yeo, B. T., & Buckner, R. L. (2012). The organization of the human striatum estimated by intrinsic functional connectivity. *J Neurophysiol*, *108*(8), 2242-2263. doi:10.1152/jn.00270.2012
- Coutinho, J. F., Fernandes, S. V., Soares, J. M., Maia, L., Goncalves, O. F., & Sampaio, A. (2016). Default mode network dissociation in depressive and anxiety states. *Brain Imaging Behav*, *10*(1), 147-157. doi:10.1007/s11682-015-9375-7
- Dichter, G. S., Gibbs, D., & Smoski, M. J. (2014). A systematic review of relations between resting-state functional-MRI and treatment response in major depressive disorder. *J Affect Disord*, *172C*, 8-17. doi:10.1016/j.jad.2014.09.028

- Ding, C., & He, X. (2004). *K-means clustering via principal component analysis*. Paper presented at the Proceedings of the twenty-first international conference on Machine Learning.
- Downes, J. H., Hammond, M. W., Xydias, D., Spencer, M. C., Becerra, V. M., Warwick, K., . . . Nasuto, S. J. (2012). Emergence of a small-world functional network in cultured neurons. *PLoS Comput Biol*, 8(5), e1002522. doi:10.1371/journal.pcbi.1002522
- Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3, 32-57.
- Edelstein, W. A., Glover, G. H., Hardy, C. J., & Redington, R. W. (1986). The intrinsic signal-to-noise ratio in NMR imaging. *Magn Reson Med*, 3(4), 604-618.
- Efron, B. T. R. J. (1994). *An Introduction to the Bootstrap*: Chapman & Hall/CRC.
- Everitt, B. S., Landau, S., Leese, M., & Stahl, D. (2011). *Cluster Analysis* (5th ed.): Wiley.
- Glasser, M. F., Sotiropoulos, S. N., Wilson, J. A., Coalson, T. S., Fischl, B., Andersson, J. L., . . . Consortium, W. U.-M. H. (2013). The minimal preprocessing pipelines for the Human Connectome Project. *Neuroimage*, 80, 105-124. doi:10.1016/j.neuroimage.2013.04.127
- Grissom, R. J., & Kim, J. J. (2012). *Effect sizes for research: Univariate and multivariate applications*: Routledge.
- Gu, S., Satterthwaite, T. D., Medaglia, J. D., Yang, M., Gur, R. E., Gur, R. C., & Bassett, D. S. (2015). Emergence of system roles in normative neurodevelopment. *Proc Natl Acad Sci U S A*, 112(44), 13681-13686. doi:10.1073/pnas.1502829112
- Gudbjartsson, H., & Patz, S. (1995). Simultaneous calculation of flow and diffusion sensitivity in steady-state free precession imaging. *Magn Reson Med*, 34(4), 567-579.
- Hindriks, R., Adhikari, M. H., Murayama, Y., Ganzetti, M., Mantini, D., Logothetis, N. K., & Deco, G. (2016). Can sliding-window correlations reveal dynamic functional connectivity in resting-state fMRI? *Neuroimage*, 127, 242-256. doi:10.1016/j.neuroimage.2015.11.055
- Kiviniemi, V., Starck, T., Remes, J., Long, X., Nikkinen, J., Haapea, M., . . . Tervonen, O. (2009). Functional segmentation of the brain cortex using high model order group PICA. *Hum Brain Mapp*, 30(12), 3865-3886. doi:10.1002/hbm.20813

- Krajcovicova, L., Marecek, R., Mikl, M., & Rektorova, I. (2014). Disruption of resting functional connectivity in Alzheimer's patients and at-risk subjects. *Curr Neurol Neurosci Rep*, *14*(10), 491. doi:10.1007/s11910-014-0491-3
- Kruger, G., & Glover, G. H. (2001). Physiological noise in oxygenation-sensitive magnetic resonance imaging. *Magn Reson Med*, *46*(4), 631-637.
- Lloyd, S. P. (1982). Least Squares Quantization in PCM. *IEEE Transactions Information Theory*, *28*, 129-137.
- Lopez-Larson, M. P., Anderson, J. S., Ferguson, M. A., & Yurgelun-Todd, D. (2011). Local brain connectivity and associations with gender and age. *Dev Cogn Neurosci*, *1*(2), 187-197. doi:10.1016/j.dcn.2010.10.001
- Lowe, M. J., Mock, B. J., & Sorenson, J. A. (1998). Functional connectivity in single and multislice echoplanar imaging using resting-state fluctuations. *Neuroimage*, *7*(2), 119-132. doi:10.1006/nimg.1997.0315
- Marcus, D. S., Harms, M. P., Snyder, A. Z., Jenkinson, M., Wilson, J. A., Glasser, M. F., . . . Consortium, W. U.-M. H. (2013). Human Connectome Project informatics: quality control, database services, and data visualization. *Neuroimage*, *80*, 202-219. doi:10.1016/j.neuroimage.2013.05.077
- McLeod, K. R., Langevin, L. M., Dewey, D., & Goodyear, B. G. (2016). Atypical within- and between-hemisphere motor network functional connections in children with developmental coordination disorder and attention-deficit/hyperactivity disorder. *Neuroimage Clin*, *12*, 157-164. doi:10.1016/j.nicl.2016.06.019
- Mingoia, G., Wagner, G., Langbein, K., Maitra, R., Smesny, S., Dietzek, M., . . . Nenadic, I. (2012). Default mode network activity in schizophrenia studied at resting state using probabilistic ICA. *Schizophr Res*, *138*(2-3), 143-149. doi:10.1016/j.schres.2012.01.036
- Mitra, A., & Raichle, M. E. (2016). How networks communicate: propagation patterns in spontaneous brain activity. *Philos Trans R Soc Lond B Biol Sci*, *371*(1705). doi:10.1098/rstb.2015.0546
- Pelleg, D., & Moore, A. (2000, June 29 - July 02). *X-means: Extending K-means with Efficient Estimation of the Number of Clusters*. Paper presented at the Proceedings of the Seventeenth International Conference on Machine Learning.
- Peterson, A., Thome, J., Frewen, P., & Lanius, R. A. (2014). Resting-state neuroimaging studies: a new way of identifying differences and similarities among the anxiety disorders? *Can J Psychiatry*, *59*(6), 294-300.

- Power, J. D., Mitra, A., Laumann, T. O., Snyder, A. Z., Schlaggar, B. L., & Petersen, S. E. (2014). Methods to detect, characterize, and remove motion artifact in resting state fMRI. *Neuroimage*, *84*, 320-341. doi:10.1016/j.neuroimage.2013.08.048
- Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53-65.
- Schwarz, G. (1978). Estimating the Dimension of a Model. *Annals of Statistics*, *6*, 461-464.
- Sepulcre, J., Sabuncu, M. R., Yeo, T. B., Liu, H., & Johnson, K. A. (2012). Stepwise connectivity of the modal cortex reveals the multimodal organization of the human brain. *J Neurosci*, *32*(31), 10649-10661. doi:10.1523/JNEUROSCI.0759-12.2012
- Shen, X., Papademetris, X., & Constable, R. T. (2010). Graph-theory based parcellation of functional subunits in the brain from resting-state fMRI data. *Neuroimage*, *50*(3), 1027-1035. doi:10.1016/j.neuroimage.2009.12.119
- Sohn, W. S., Yoo, K., Lee, Y. B., Seo, S. W., Na, D. L., & Jeong, Y. (2015). Influence of ROI selection on resting state functional connectivity: an individualized approach for resting state fMRI analysis. *Front Neurosci*, *9*, 280. doi:10.3389/fnins.2015.00280
- Szewczyk-Krolikowski, K., Menke, R. A., Rolinski, M., Duff, E., Salimi-Khorshidi, G., Filippini, N., . . . Mackay, C. E. (2014). Functional connectivity in the basal ganglia network differentiates PD patients from controls. *Neurology*, *83*(3), 208-214. doi:10.1212/WNL.0000000000000592
- Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E., Yacoub, E., Ugurbil, K., & Consortium, W. U.-M. H. (2013). The WU-Minn Human Connectome Project: an overview. *Neuroimage*, *80*, 62-79. doi:10.1016/j.neuroimage.2013.05.041
- Wang, Y., Zhong, S., Jia, Y., Zhou, Z., Zhou, Q., & Huang, L. (2015). Reduced interhemispheric resting-state functional connectivity in unmedicated bipolar II disorder. *Acta Psychiatr Scand*. doi:10.1111/acps.12429
- Yeo, B. T., Krienen, F. M., Sepulcre, J., Sabuncu, M. R., Lashkari, D., Hollinshead, M., . . . Buckner, R. L. (2011). The organization of the human cerebral cortex estimated by intrinsic functional connectivity. *J Neurophysiol*, *106*(3), 1125-1165. doi:10.1152/jn.00338.2011
- Zang, Y., Jiang, T., Lu, Y., He, Y., & Tian, L. (2004). Regional homogeneity approach to fMRI data analysis. *Neuroimage*, *22*(1), 394-400. doi:10.1016/j.neuroimage.2003.12.030

ANEXOS

ANEXO 1. Ejemplo de la aplicación del método de clústering a una región de un sujeto

La información que tenemos de una región es una matriz de 1200×343 que representa los valores de activación de los 343 vóxeles en los 1200 instantes temporales. Lo primero que hacemos con esta matriz es realizar el diseño enventanado, que no es parte del método de clústering en sí mismo sino que se utiliza para medir la estabilidad. Una vez realizado este diseño contamos con 15 matrices de 600×343 que representan la activación de los 343 vóxeles en una ventana de 600 instantes temporales pertenecientes a la señal original (recordemos que las ventanas estaban solapadas).

El método de clústering propiamente dicho se aplicará a cada una de estas matrices, pero para ilustrarlo se explicarán solo los resultados que se obtienen en una. El primer paso es aplicar *CBB* a la matriz. Al realizarlo obtenemos 1000 matrices de 600×343 que se corresponden con muestras independientes obtenidas a partir de la matriz a la que se aplica el método. Al igual que con el diseño enventanado, a continuación se va a explicar la aplicación en una de estas muestras pero habría que replicar el procedimiento en las 1000.

En esta muestra aplicamos un análisis de componentes principales para reducir la dimensionalidad de los datos, puesto no es posible clusterizar los 343 vóxeles a partir de 600 variables (al menos no mediante *GMM*). En la Tabla 8 se muestra el porcentaje de varianza explicada por los 30 primeros componentes, así como el de varianza acumulada. Los criterios para extraer los componentes que utilizaremos en el clústering son que sumen un 80% de varianza explicada y que ninguno de los componentes explique menos del 1% (aunque esto suponga quedarse por debajo del umbral del 80%). Siguiendo estos criterios en nuestro caso extraeríamos los 20 primeros componentes explicando un 80.899% de la varianza de nuestra matriz original.

A continuación aplicaremos el algoritmo *X-GMM* a la matriz 343×20 que contiene los pesos de los 20 componentes utilizados en los 343 vóxeles. El resultado de este algoritmo es un vector que indica a qué clúster pertenece cada vóxel. Al aplicar este procedimiento a las 1000 muestras independientes obtenidas mediante *CBB* obtendríamos el mismo número de vectores de asignaciones. A partir de ellos creamos una matriz de

estabilidad S de dimensiones 343×343 que incluye la probabilidad de cada par de vóxeles de pertenecer al mismo clúster obtenida de forma frecuentista (la suma del número de veces que han pertenecido al mismo clúster dividida entre 1000). Podemos ver un ejemplo de esta matriz en la Figura 15.

Debido al diseño inventariado obtenemos 15 matrices de estabilidad, y para realizar el clústering final obtendremos una matriz de estabilidad media \bar{S} . Finalmente aplicamos el algoritmo de clústering jerárquico aglomerativo a esta matriz y obtenemos el número de clústers definitivo.

N	% V. E.	% V. A.	N2	% V.E.	% V.A.	N3	% V.E.2	% V.A.
1	21.65	21.65	11	2.45	64.43	21	1.37	82.27
2	7.59	29.24	12	2.29	66.72	22	1.33	83.60
3	6.49	35.73	13	2.08	68.79	23	1.32	84.92
4	5.27	41.00	14	2.02	70.81	24	1.17	86.09
5	4.73	45.73	15	1.98	72.79	25	1.08	87.16
6	3.91	49.64	16	1.80	74.59	26	1.05	88.21
7	3.53	53.17	17	1.67	76.25	27	1.02	89.24
8	3.12	56.29	18	1.63	77.88	28	0.97	90.20
9	2.94	59.23	19	1.58	79.46	29	0.95	91.15
10	2.75	61.97	20	1.44	80.90	30	0.87	92.01

Tabla 8. En esta tabla se muestran los 30 primeros componentes (identificados en la columna N) y sus porcentajes de varianza explicada (% V. E.) y de varianza acumulada (% V. A.).

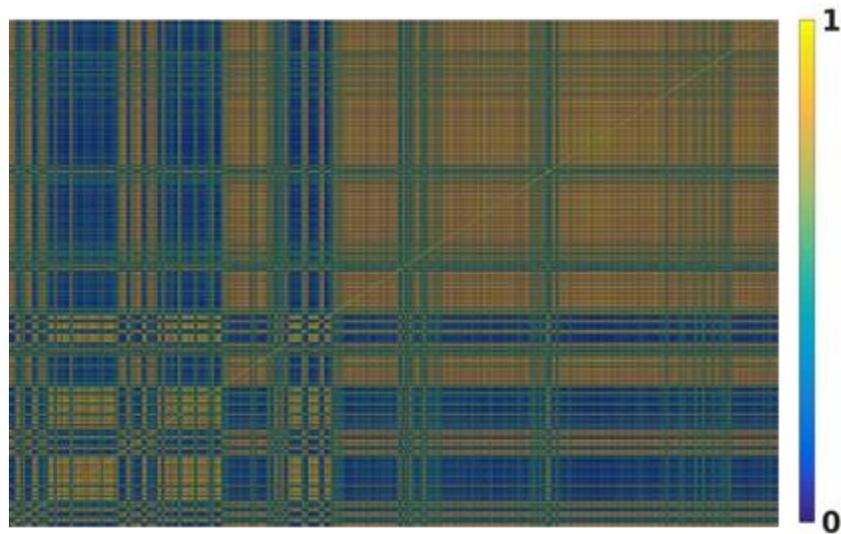


Figura 15. Matriz de estabilidad representando la probabilidad de que cada par de vóxeles pertenezca al mismo clúster.

ANEXO 2. Código del algoritmo X-GMM

```
1 function [memb,p] = x_gmm_clustering(varargin)
2
3 warning('off','all');
4
5 y = varargin{1};
6 if length(varargin)==2;
7     k = varargin{2};
8 else
9     k = ceil(size(y,1)/10);
10 end
11 try
12     gm = fitgmdist(y,1);
13 catch
14     gm = fitgmdist(y,1,'RegularizationValue',0.001);
15 end
16 bic1 = gm.BIC;
17 cent = gm.mu;
18 mu = gm.mu;
19 sigma = gm.Sigma;
20
21 dist = sum((y-repmat(cent,size(y,1),1)).^2,2).^5;
22 [~,ind] = max(dist);
23 dist = (y(ind,:)-cent)/3;
24 cent = [cent+dist;cent-dist];
25 centdist(:,1) = sum((y-repmat(cent(1,:),size(y,1),1)).^2,2).^5;
26 centdist(:,2) = sum((y-repmat(cent(2,:),size(y,1),1)).^2,2).^5;
27 [~,ind] = min(centdist,[],2);
28
29 stable = zeros(1,size(y,2));
30 flag = 0;
31
32 for i=2:k
33     if i==2;
34         try
35             gm = fitgmdist(y,2,'Start',ind);
36         catch
37             gm = fitgmdist(y,2,'Start',ind,'RegularizationValue',0.001);
38         end
39         bic2 = gm.BIC;
40
41         if bic1<bic2
42             break
43         else
44             mu = gm.mu;
45             sigma = gm.Sigma;
46             p = posterior(gm,y);
47             [~,ids] = max(p,[],2);
48         end
49     else
50         for j=1:i-1
51             if any(stable==j);
52                 continue
53             end
54             x = y(ids==j,:);
55             gm = fitgmdist(x,1);
```

```

59 - bic1 = gm.BIC;
60 - cent = gm.mu;
61 - dist = sum((x-repmat(cent,size(x,1),1)).^2,2).^5;
62 - [~,ind] = max(dist);
63
64 - dist = (x(ind,:)-cent)/3;
65 - cent = [cent+dist;cent-dist];
66 - clear centdist
67 - centdist(:,1) = sum((x-repmat(cent(1,:),size(x,1),1)).^2,2).^5;
68 - centdist(:,2) = sum((x-repmat(cent(2,:),size(x,1),1)).^2,2).^5;
69 - [~,ind] = min(centdist,[],2);
70
71 - try
72 -     gm = fitgmdist(x,2,'Start',ind);
73 - catch
74 -     gm = fitgmdist(x,2,'Start',ind,'RegularizationValue',0.001);
75 - end
76 - bic2 = gm.BIC;
77
78 - if bic2<bic1
79
80 -     d.mu = mu;
81 -     d.Sigma = sigma;
82 -     try
83 -         gm2 = fitgmdist(y,size(mu,1),'Start',d);
84 -     catch
85 -         gm2 = fitgmdist(y,size(mu,1),'Start',d,'RegularizationValue',0.001);
86 -     end
87 -     bic1= gm2.BIC;
88
89 -     mu2=mu;
90 -     mu2(j,:) = gm.mu(1,:);
91 -     mu2 = [mu2;gm.mu(2,:)];
92 -     sigma2 = sigma;
93 -     sigma2(:,j) = gm.Sigma(:,1);
94 -     sigma2 = cat(3,sigma2,gm.Sigma(:,2));
95 -     d.mu = mu2;
96 -     d.Sigma = sigma2;
97 -     try
98 -         gm2 = fitgmdist(y,size(mu2,1),'Start',d);
99 -     catch
100 -         gm2 = fitgmdist(y,size(mu2,1),'Start',d,'RegularizationValue',0.001);
101 -     end
102 -     bic2 = gm2.BIC;
103
104 -     if bic2<bic1
105 -         p = posterior(gm2,y);
106 -         [~,ids] = max(p,[],2);
107 -         if length(find(ids==j))<floor(size(y,1)/50) || ...
108 -             length(find(ids==i))<floor(size(y,1)/50)
109 -             if j==i-1
110 -                 flag = 1;
111 -                 break
112 -             end
113 -             stable = 1:j;
114 -             continue
115 -         end
116 -         mu = mu2;
117 -         sigma = sigma2;
118
119 -     else
120 -         if j==i-1
121 -             flag = 1;
122 -             break
123 -         end
124 -         stable = 1:j;
125 -     end

```

```

126 -         else
127 -             if j==i-1
128 -                 flag = 1;
129 -                 break
130 -             end
131 -             stable = 1:j;
132 -         end
133 -
134 -     end
135 - end
136 - if flag == 1;
137 -     break
138 - end
139 - end
140 -
141 - d.mu = mu;
142 - d.Sigma = sigma;
143 - try
144 -     gm = fitgmdist(y,size(mu,1),'Start',d);
145 - catch
146 -     gm = fitgmdist(y,size(mu,1),'Start',d,'RegularizationValue',0.001);
147 - end
148 - p = posterior(gm,y);
149 - [~,memb] = max(p,[],2);
150 - warning('on','all');

```

ANEXO 3. Técnicas de preprocesado de imágenes de RMf

A continuación se realiza una breve descripción de las etapas del preprocesado de las imágenes:

- **Corrección de movimiento:** Se realiza una minimización de las diferencias en la posición de la cabeza entre los volúmenes que componen la secuencia funcional. Estas diferencias se deben a pequeños movimientos involuntarios que pueda realizar el sujeto durante la adquisición. Para este proceso se utilizó la herramienta “motion correction” del programa SPM12.
- **Corregistro:** Se realizó un registro lineal de la imagen funcional (EPI) a la estructural (T1), obteniendo 6 parámetros de movimiento (3 de traslación y 3 de rotación) empleando la herramienta “coregister” incluida en SPM12.
- **Detrending:** Proceso que se emplea para eliminar los artefactos de *drift* temporal producidos por la sobrecarga del escáner. Se eliminó una tendencia cúbica mediante la herramienta “*detrend*” de AFNI.
- **Filtro pasa-banda (0,005-0,1 Hz):** Se utiliza para conservar tan solo las frecuencias dentro del espectro especificado, en nuestro caso las fluctuaciones de la señal que nos interesan para este tipo de análisis se encuentran el rango seleccionado (Fox & Raichle, 2007). Realizado con la herramienta “3DBandpass” de AFNI.
- **Normalización de la intensidad media global:** Se utiliza para que las señales de cada serie temporal estén en la misma escala. Realizado mediante la herramienta “*fslmaths*” del software FSL.