



Universidad de Valladolid

Facultad de ciencias

GRADO EN ESTADÍSTICA

**DESARROLLO DE UN NUEVO MÉTODO DE CLASIFICACIÓN
PARA SEÑALES OSCILATORIAS MULTICANAL Y SU
APLICACIÓN EN LA DETECCIÓN DE SPIKES**

Alumno: César Merino Fidalgo

Tutor: Cristina Rueda Sabater



A mi madre

Agradecimientos

A mi tutora Cristina, por ofrecerme la oportunidad de realizar este trabajo sobre un tema apasionante y por apoyarme en todo momento. Gracias por guiarme todos estos meses.

A mi familia y amigos, por hacerme los días más fáciles y ayudarme en los momentos más difíciles. Gracias por estar conmigo siempre.

Pero más que a nadie, gracias a mi madre, por todo lo que has hecho por mí. Por quererme como sólo tú sabes, por apoyarme siempre en todo lo que hago, poniéndome siempre por delante. Gracias por todo tu cariño, por tus buenas palabras, por tus besos, tus risas y todos nuestros días juntos. Mamá, siempre me acompañas, y yo siempre te acompañaré.

Resumen

El estudio del cerebro es hoy en día uno de los mayores desafíos científicos debido a su enorme complejidad y a las limitaciones en la comprensión de su funcionamiento. Una de las herramientas clave para avanzar en este campo es el spike sorting, una técnica que permite clasificar los potenciales de acción —también conocidos como spikes— registrados mediante electrodos, identificando qué neurona los ha generado. Estos spikes son señales eléctricas breves y de forma oscilatoria que contienen información valiosa sobre la actividad neuronal.

Dado su carácter oscilatorio, modelar adecuadamente estas señales es fundamental para lograr una clasificación precisa. En este sentido, el modelo FMM (Frequency Modulated Möbius) ha demostrado ser eficaz al representar con precisión la forma oscilatoria de los spikes, lo que lo convierte en una herramienta prometedora para el spike sorting. Sin embargo, hasta ahora, este modelo solo se ha aplicado a datos de un único canal (un solo electrodo), lo que limita su utilidad práctica en entornos reales donde se utilizan múltiples electrodos para registrar actividad simultánea en varias regiones del cerebro.

Este trabajo propone un nuevo modelo, denominado Mix3DFMM, que extiende las capacidades del modelo FMM al contexto multicanal. Es decir, permite procesar simultáneamente señales oscilatorias provenientes de varios electrodos, aprovechando la información compartida entre canales para mejorar el rendimiento del proceso de clasificación.

Los resultados muestran que el modelo Mix3DFMM mejora significativamente la precisión del spike sorting, con incrementos del 8% al 12% respecto al modelo de un solo canal, demostrando su potencial de aplicación en estudios neurocientíficos complejos.

Palabras clave: spike sorting, FMM, señales oscilatorias, señales neuronales.

Abstract

The study of the brain is currently one of the greatest scientific challenges due to its enormous complexity and the limitations in understanding how it works. One of the key tools to advance in this field is spike sorting, a technique that allows classifying action potentials —also known as spikes— recorded by electrodes, identifying which neuron generated them. These spikes are brief electrical signals with oscillatory shapes, containing valuable information about neuronal activity.

Given their oscillatory nature, accurately modeling these signals is crucial for achieving precise classification. In this regard, the Frequency Modulated Möbius (FMM) model has proven to be effective in accurately representing the oscillatory shape of spikes, making it a promising tool for spike sorting. However, until now, this model has only been applied to single-channel data (a single electrode), limiting its practical utility in real-world environments where multiple electrodes are used to record simultaneous activity across various brain regions.

This work proposes a new model, called Mix3DFMM, which extends the capabilities of the FMM model to the multichannel context. That is, it allows simultaneous processing of oscillatory signals from multiple electrodes, leveraging the information across channels to enhance classification performance.

Results show that the Mix3DFMM model significantly improves spike sorting accuracy, with increases ranging from 8 % to 12 % compared to the single-channel model, demonstrating its potential for application in complex neuroscientific studies.

Keywords: spike sorting, FMM, oscillatory signals, neural signals.

Índice general

Agradecimientos	III
Resumen	V
Abstract	VII
Lista de figuras	XIII
Lista de tablas	XV
1. Introducción	1
1.1. Motivación	1
1.2. Objetivo	2
1.3. Estructura de la memoria	2
2. Contexto	5
2.1. La Neurona	5
2.1.1. Morfología	5
2.2. El impulso nervioso	6
2.3. Obtención de señales neuronales	7
2.4. Spike Sorting	8
2.4.1. Utilidades del Spike Sorting	8

2.4.2. Proceso de Spike Sorting	8
2.5. Estado del arte en el spike sorting	10
2.5.1. PCA	11
2.5.2. GMM	11
2.5.3. Kilosort	11
3. Metodología	13
3.1. Modelo FMM	13
3.1.1. Modelo 3DFMM	18
3.2. Modelo MixFMM	20
3.2.1. Estimación de máxima verosimilitud mediante el algoritmo EM	20
3.3. Modelo Mix3DFMM	24
3.3.1. Estimación de máxima verosimilitud para el modelo Mix3DFMM	24
3.3.2. Selección de canales	25
4. Diseño experimental	27
4.1. Conjunto de datos	27
4.1.1. Librería MEArec	27
4.1.2. Datos utilizados	28
4.2. Implementación monocanal	31
4.3. Implementación multicanal	32
5. Resultados	33
5.1. Resultados del modelo monocanal	33
5.1.1. Dataset 1	33
5.1.2. Dataset 2	34
5.2. Resultados del nuevo modelo multicanal	35
5.2.1. Dataset 1	35
5.2.2. Dataset 2	36

6. Conclusiones	39
6.1. Líneas de trabajo futuras	39
Bibliografía	41

Lista de Figuras

2.1. Estructura de una neurona	6
2.2. Señal de un impulso nervioso extraído de [9].	7
2.3. Ejemplo de filtrado de una señal con varios tipos de filtros. Extraído de [1].	9
2.4. Ejemplo de aplicación de un filtro a una señal neuronal. Extraído de [13].	9
2.5. Ejemplo de detección y recorte de spikes. Extraído de [14].	10
2.6. Flujo de trabajo del algoritmo Kilosort4. Extraído de [17].	12
3.1. Representación de la onda FMM con parámetros de la Tabla 3.1.	14
3.2. Variación de los parámetros de amplitud en la onda de la Figura 3.1.	15
3.3. Variación del parámetro α en la onda de la Figura 3.1.	15
3.4. Variación del parámetro β en la onda de la Figura 3.1.	16
3.5. Variación del parámetro ω en la onda de la Figura 3.1.	16
3.6. Ajuste de una onda FMM a una señal con 2 ondas.	17
3.7. Ajuste de una onda FMM ₂ a una señal con 2 ondas.	17
3.8. Ajuste y componentes del modelo 3DFMM a una señal con 4 canales.	19
3.9. Flujo de trabajo del algoritmo MixFMM. Extraído de [4].	23
4.1. Señales del dataset 1.	28
4.2. Señales del dataset 2.	29
4.3. Señales medias de los 16 primeros canales de las clases del dataset 1.	29

4.4. Señales medias ajustadas de los 16 primeros canales de las clases del dataset 1.	30
4.5. Señales medias por canal de las clases del dataset 2.	30
4.6. Señales medias ajustadas por canal de las clases del dataset 2.	31
5.1. Evolución de la verosimilitud y la precisión durante el entrenamiento del modelo multicanal para el dataset 1.	36
5.2. Evolución de la verosimilitud y la precisión durante el entrenamiento del modelo multicanal para el dataset 2.	37

Lista de Tablas

3.1. Parámetros del modelo base de una onda FMM.	14
4.1. Características de los conjuntos de datos utilizados.	28
5.1. Verosimilitud por canal obtenida del dataset 1.	34
5.2. Precisión obtenida por el canal 22 en el dataset 1.	34
5.3. Verosimilitud por canal para el Dataset 2.	34
5.4. Precisión obtenida por el canal 4 en el dataset 2.	35
5.5. Comparativa de los resultados obtenidos por ambos modelos para el dataset 1.	35
5.6. Resultados obtenidos por el modelo multicanal para el dataset 2.	36

Capítulo 1

Introducción

1.1. Motivación

En la actualidad, el estudio del cerebro es uno de los mayores desafíos de la ciencia dificultado por la gran complejidad de este órgano. Dentro de este ámbito, el análisis de la actividad eléctrica de las neuronas es una de las áreas fundamentales para abarcar esta cuestión. Como forma de estudiar las señales neuronales, surge el proceso de spike sorting[1], que es en una colección de técnicas que permite identificar y clasificar las señales eléctricas según la neurona que las ha generado. Este proceso realiza la clasificación de las señales en función de características de cada potencial de acción, también conocido como spike, consistente en una señal eléctrica breve y de carácter oscilatorio característica de cada neurona. La clasificación de estas señales puede resultar extremadamente útil para estudiar cuestiones como la conectividad entre neuronas o la respuesta de las neuronas a determinados estímulos. Sin embargo, el proceso de spike sorting es realmente complejo, ya que las señales capturadas por electrodos pueden ser muy similares entre sí y contener mucho ruido, lo cual dificulta la correcta identificación de los spikes.

Además del interés intrínseco del problema que plantea importantes desafíos desde el punto de vista estadístico, este trabajo ha estado fuertemente motivado por el desarrollo del modelo FMM[2] llevado a cabo por The FMM Project[3] en la Universidad de Valladolid. Este modelo ha demostrado ser capaz de ajustar con precisión señales oscilatorias, pero su aplicación en este ámbito no ha sido desarrollada de manera extensa. La investigación con este modelo llevada a cabo en el campo neurológico ha demostrado el buen funcionamiento en el proceso de spike sorting con un solo canal [4]. Sin embargo, al limitarlo al uso de un único canal, se reduce su aplicación práctica, ya que la mayoría de registros neuronales se realizan con múltiples electrodos. Por lo tanto, el desarrollo de un modelo que pueda ser capaz de trabajar con datos multicanal puede suponer un avance significativo en el desarrollo del modelo FMM.

Adicionalmente, el problema del spike sorting se lleva estudiando desde hace décadas y existe mucha literatura en la que se explica cómo se ha tratado de resolverlo con diferentes

enfoques. Por lo tanto, el desarrollo de esta técnica completamente novedosa puede aportar una nueva perspectiva al problema y abrir un nuevo enfoque capaz de mejorar los resultados obtenidos hasta ahora.

1.2. Objetivo

El objetivo principal de este proyecto es aportar una nueva solución al problema del spike sorting. Para ello, se va a crear un modelo completamente nuevo basado en el modelo FMM existente. Este nuevo modelo deberá ser capaz de procesar señales provenientes de varias clases de neuronas y utilizar la información de múltiples canales para realizar la clasificación de spikes. Para que el modelo sea verdaderamente útil, deberá ser capaz de obtener resultados mejores a los obtenidos por el modelo existente MixFMM[4] que solo utiliza un único canal, demostrando así la capacidad del nuevo modelo de aprovechar la información de múltiples canales para mejorar la clasificación de spikes.

1.3. Estructura de la memoria

La memoria del proyecto se ha estructurado de manera que permita entender el problema a tratar y cómo se ha llevado a cabo el trabajo. Para ello, se ha dividido en los siguientes capítulos:

- **Introducción:** en este capítulo se presenta la motivación del proyecto y los objetivos del mismo. Además, se acota el alcance del proyecto y se describe como se estructura el trabajo.
- **Contexto:** en este capítulo se presenta el contexto biológico más importante para entender el problema del spike sorting. En una primera parte se proporcionan nociones básicas sobre el funcionamiento de las neuronas y posteriormente se explica el proceso de spike sorting.
- **Metodología:** en este capítulo se describen los modelos utilizados en el proyecto. Concretamente, se detallan las características del modelo FMM, fundamental para el desarrollo del nuevo modelo, además de los modelos existentes 3DFMM[5] y MixFMM, que servirán de base para el nuevo modelo Mix3DFMM. Finalmente, se detalla el diseño y funcionamiento del nuevo modelo Mix3DFMM.
- **Diseño experimental:** en este capítulo se detallan los conjuntos de datos utilizados para las pruebas del modelo. Además, se explican las pruebas que se han realizado para realizar la comparación entre los modelos monocanal y multicanal.
- **Resultados:** en este capítulo se presentan los resultados obtenidos en las pruebas realizadas. Primero, se muestran los resultados obtenidos por el modelo monocanal MixFMM, y a continuación, se presentan los resultados del modelo multicanal, realizando una comparación de ambos modelos.

- **Conclusiones:** en el último capítulo se presentan las conclusiones obtenidas a partir de los resultados de la experimentación. Además, se proponen distintas líneas de trabajo futuras que pueden ayudar a mejorar el modelo y su funcionamiento.

Capítulo 2

Contexto

Para poder comprender correctamente el trabajo que se ha realizado, es necesario aportar un contexto que de sentido a la investigación. En la primera parte se dará contexto sobre las señales con las que se va a trabajar, y en la segunda parte se explicará detalladamente el problema del spike sorting.

2.1. La Neurona

La estructura y el funcionamiento básico de las neuronas ha sido descrito en múltiples ocasiones. A continuación, se exponen las características fundamentales de estas células basadas en [6][7].

Las neuronas son un tipo de célula muy especializada que constituye el componente fundamental del sistema nervioso. Estas células se encargan de transmitir la información a través de señales químicas y eléctricas.

2.1.1. Morfología

La morfología de las neuronas es variada, sin embargo, todas ellas comparten una serie de características comunes:

- **Cuerpo celular (Soma):** Es la parte central y principal de la neurona donde además se aloja el núcleo de la célula.
- **Dendritas:** Prolongaciones de la membrana celular que preceden al soma. Se encargan de recibir la información a través de señales eléctricas provenientes de otras neuronas.

- Axón: Del soma parte una única prolongación que una morfología diferente a las dendritas que se encarga de enviar el impulso nervioso a otras neuronas.

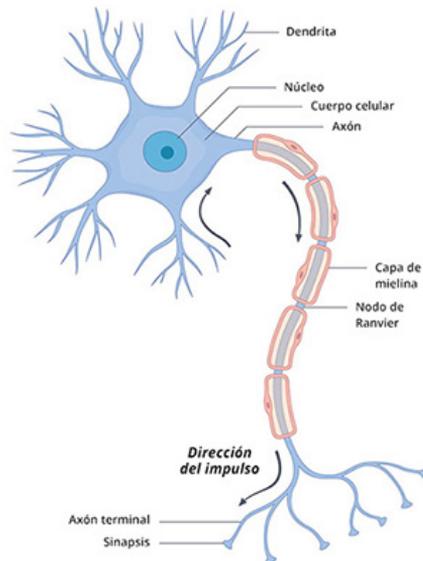


Figura 2.1: Estructura de una neurona

2.2. El impulso nervioso

El impulso nervioso o potencial de acción es el mecanismo por el que las neuronas transmiten información. Este impulso es una onda de descarga eléctrica que se propaga a lo largo del axón por la membrana celular modificando su distribución de carga eléctrica [8]. Este impulso se puede generar por otro tipo de células, como las células musculares.

Para la generación de un impulso nervioso, existe una diferencia de potencial entre el interior y el exterior de la membrana celular de unos $-60/-70$ mV. Cuando el potencial de membrana se despolariza por encima de un umbral crítico que puede estar entre -60 y -40 mV, el potencial de membrana se vuelve positivo, alcanzando el valor de $+30$ mV y propagando el impulso a lo largo del axón. Estas diferencias de potencial se deben a la presencia principalmente de iones de sodio y potasio en el exterior e interior de la célula respectivamente [8][9]. Los cambios de polaridad se deben al bombeo de estos iones a través de la membrana celular.

La señal que produce el impulso nervioso es lo que se conoce como spike debido al pico que se produce por la señal eléctrica generada. Este pico se puede observar en la figura 2.2, donde se muestra un ejemplo de un impulso nervioso en el que aparecen descritas las diferentes fases del potencial de acción. Esta señal o spike es la que se clasificará durante el proceso de spike sorting.

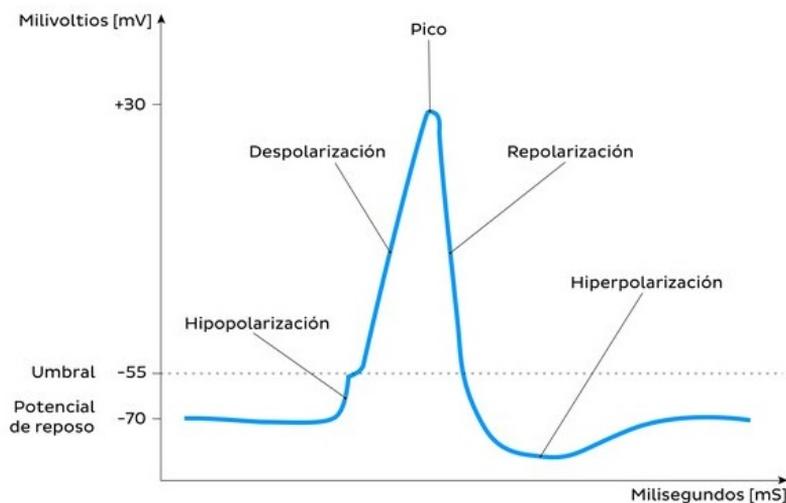


Figura 2.2: Señal de un impulso nervioso extraído de [9].

2.3. Obtención de señales neuronales

La actividad cerebral y las señales neuronales se pueden obtener de diferentes maneras. Además, existen muchos tipos de electrodos capaces de captar las señales neuronales, y habitualmente se utilizan varios de estos electrodos para capturar señales desde diferentes posiciones. Dentro del estudio de spikes, se pueden diferenciar dos tipos de registros según el entorno en el que se realicen:

- **In-vitro:** este tipo de registro de señales se realiza en un entorno controlado, en laboratorio, donde se utilizan cultivos de células o partes del tejido de interés como el hipotálamo o la corteza cerebral para estudiar sus características [10]. Aunque este tipo de estudio es muy útil para los casos controlados, no dejan de ser unas condiciones artificiales que no representan al completo la realidad de la actividad cerebral.
- **In-vivo:** los registros in-vivo se realizan sobre organismos vivos, como pueden ser animales o humanos. Estas mediciones se pueden hacer de forma intracelular, utilizando microelectrodos que se insertan en el tejido cerebral directamente sobre las neuronas, o de manera extracelular, la cual utiliza electrodos colocados en la superficie del cráneo o cerebro para captar las señales neuronales. Aunque este último tipo de grabaciones son las menos invasivas, conllevan una serie de problemas al no captar las señales de cada neurona individualmente. Además, aunque las señales de las neuronas tengan una forma característica que será captada por los electrodos, estas señales son susceptibles de variar durante la grabación debido a factores como el movimiento del sujeto, pequeños desplazamientos de los electrodos o incluso pequeños movimientos de las propias

neuronas que cambiarán la forma de la señal detectada por el electrodo. Este problema se conoce como *drift*, y constituye uno de los principales desafíos del spike sorting [11].

2.4. Spike Sorting

El spike sorting es el proceso de agrupación de spikes en clusters basado en la similitud de características de los spikes. Cada neurona tiende a generar spikes con una forma determinada, de esta manera, el objetivo final será asociar cada spike con la neurona que la ha generado. La información presentada en esta sección aparece recogida en [1] y [12].

2.4.1. Utilidades del Spike Sorting

Las neuronas cercanas entre sí pueden dispararse en respuesta a estímulos diferentes, sin embargo, las señales que generan también son diferentes y es muy importante determinar qué neurona ha generado la señal [12]. El estudio de la actividad cerebral puede ayudar a diagnosticar enfermedades neurológicas como la epilepsia o el parkinson, y determinar si hay neuronas con un funcionamiento anómalo puede ayudar a dicho diagnóstico.

2.4.2. Proceso de Spike Sorting

El proceso de clasificación de spikes no solo consiste en el agrupamiento de las señales, sino que involucran otros pasos anteriores para ser capaces de realizar esta clasificación. Este preprocesamiento previo es muy importante y puede cambiar los resultados obtenidos en gran medida. En los siguientes puntos se describe el flujo de trabajo para llevar a cabo el spike sorting.

1. Filtrado: Una vez se ha obtenido la señal neuronal, se debe aplicar un filtro para limpiar dicha señal. El filtro principal es el filtro pasa banda o *band-pass filter* que se encarga de eliminar el ruido de la señal y dejar pasar las frecuencias de interés. La elección del rango de frecuencias que pasarán el filtro es crítica, ya que si se elige un rango estrecho para visualizar mejor las señales, se pueden perder características importantes de la spike. En la Figura 2.3 se puede observar el efecto en el filtrado de la señal y como la elección del tipo de filtro y el ancho de banda afectan a la señal final obtenida. Por otro lado, en la Figura 2.4 se puede observar el efecto completo de la aplicación de un filtro a una señal.

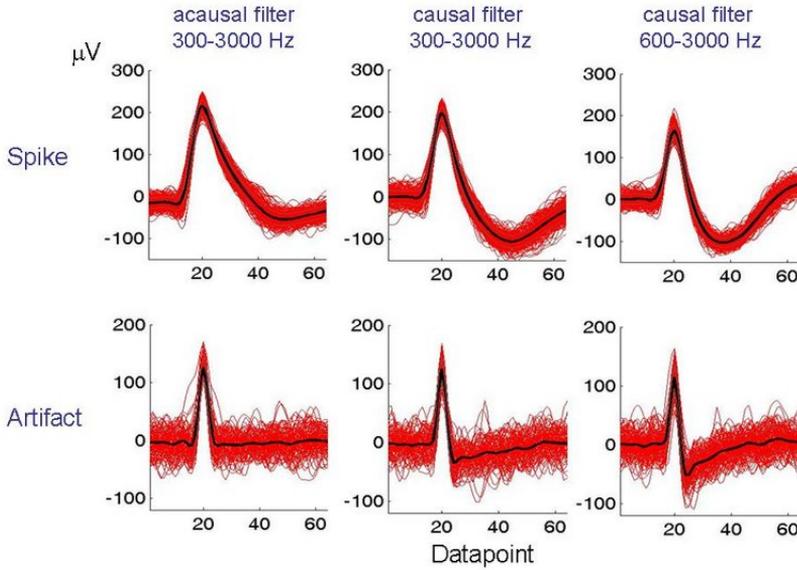


Figura 2.3: Ejemplo de filtrado de una señal con varios tipos de filtros. Extraído de [1].

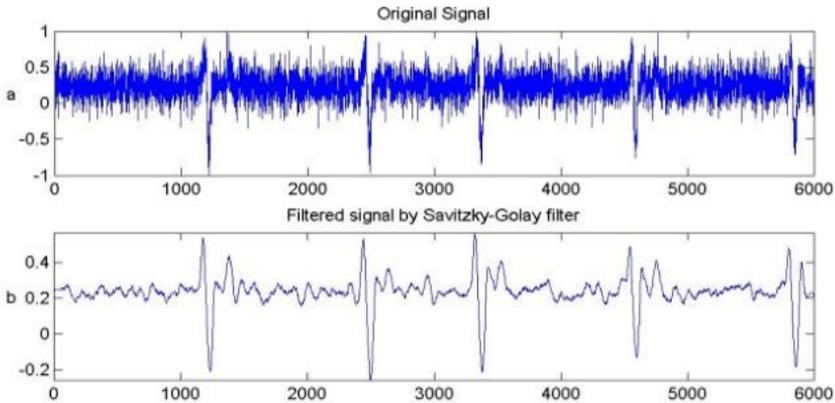


Figura 2.4: Ejemplo de aplicación de un filtro a una señal neuronal. Extraído de [13].

2. Detección: conocida como *spike detection*, es el proceso de detección de spikes. En este punto, se elige un *threshold* o umbral para detectar todas aquellas señales cuya amplitud supere este umbral. La elección puede ser manual o automática a partir de características de la señal. El valor del umbral es crítico, ya que si se elige un valor muy alto se perderán spikes, aumentando el error de tipo II, y si se elige un valor muy bajo, se pueden detectar falsos spikes cuando el ruido supere el umbral, aumentando el error de tipo I. Una vez se ha detectado el spike, esta debe ser recortada para poder

ser almacenada. Este proceso trata de seleccionar todos los puntos de la señal que corresponden a la spike detectada. En la Figura 2.5 se puede observar un ejemplo de spikes detectadas y recortadas.

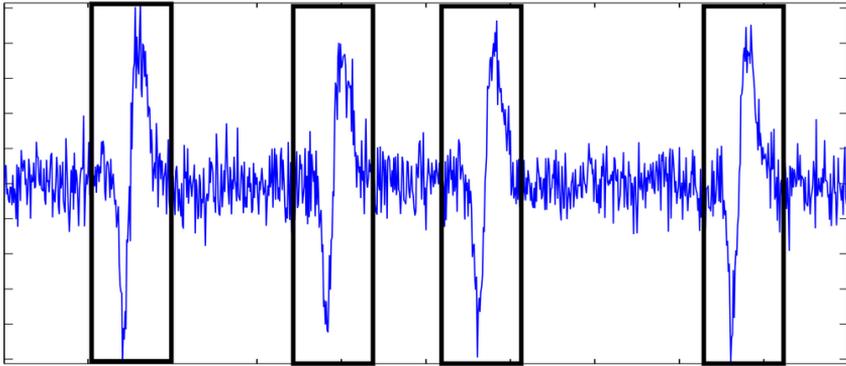


Figura 2.5: Ejemplo de detección y recorte de spikes. Extraído de [14].

3. Extracción de características: en este paso, se tratan de extraer las características más importantes para separar los spikes. Esto supone una operación de reducción de la dimensionalidad que puede ayudar a reducir el efecto del ruido y reducir el tiempo de computación. Uno de los métodos más utilizados es el análisis en componentes principales (PCA), en el que se eligen las direcciones en las que la varianza de los datos es máxima. Sin embargo, esta selección puede no ser la que mejor separe los spikes y por eso se han desarrollado otros métodos de *feature extraction*.
4. Agrupamiento: en el último paso de agrupamiento o *clustering*, se utilizan las características extraídas en el paso anterior para agrupar los spikes similares. Para lograr esto, se pueden usar algoritmos sencillos como *k-vecinos*, u otros más complejos como métodos bayesianos o algoritmos EM.

Una vez realizada la clasificación, se debe proceder a la validación de los resultados. La validación no es sencilla, ya que tener las etiquetas reales de a qué neurona pertenece cada spike es complicado si no se usan técnicas invasivas. Sin embargo, se pueden utilizar otros métodos de validación no tan precisos o utilizar datos simulados que imiten de la mejor forma posible las características de los datos reales.

2.5. Estado del arte en el spike sorting

Aunque el proceso de spike sorting se ha estudiado desde hace décadas, en los últimos años el desarrollo de nuevas técnicas y especialmente la aparición de ordenadores capaces de procesar grandes cantidades de datos con una capacidad de cómputo elevada ha permitido el desarrollo de nuevos métodos capaces de realizar este proceso de manera más rápida y

precisa. A continuación, se exponen algunos de los métodos más famosos y actuales para el spike sorting.

2.5.1. PCA

Hasta hace no mucho tiempo, el método más utilizado para la extracción de características era el uso de análisis en componentes principales o PCA [1]. Este método que consiste en la extracción de las direcciones de máxima varianza de los datos, reduciendo la dimensionalidad y permitiendo resumir los datos en un número reducido de variables [15]. La aplicación del PCA en el spike sorting consiste en la extracción de las 2 o 3 primeras componentes principales y utilizarlas para realizar posteriormente el agrupamiento de los spikes. Este método aunque resulta sencillo, ha permitido obtener buenos resultados en muchas ocasiones. Sin embargo, El análisis en componentes principales no siempre funciona bien y por ello se han desarrollado otros métodos más avanzados capaces de obtener mejores resultados.

2.5.2. GMM

Así como el PCA se utiliza en la fase de extracción de características para tratar de obtener la información más relevante de los spikes, los modelos de mixtura gaussiana o GMM[16] se utilizan en la fase de agrupamiento para tratar de separar correctamente los picos a partir de las características extraídas. Los GMM son modelos que asumen que los datos provienen de una combinación de k distribuciones gaussianas con medias y varianzas independientes. Por ejemplo, el flujo de trabajo de un algoritmo que usa GMMs para realizar spike sorting podría consistir en la extracción de características mediante PCA o alguna variación de este, y posteriormente utilizar esas características extraídas para calcular las probabilidades de pertenencia a cada uno de los clusters.

2.5.3. Kilosort

El spike sorting mediante Kilosort es un método que se lleva desarrollando varios años y en junio de 2024 publicó su último avance, Kilosort4[17]. En las primeras versiones, el principal avance que incorporaba Kilosort consistía en la combinación de los pasos de detección y clustering de spikes basado en el algoritmo de k -medias escaladas. Las versiones iniciales de este algoritmo requerían muchas correcciones y los resultados no eran del todo buenos. A medida que se publicaban nuevas versiones, se fueron solucionando los problemas principales del spike sorting como el *drift* de las ondas, lo cual produjo una gran mejora en los resultados obtenidos. Finalmente, en esta última versión de Kilosort4, se ha cambiado el proceso de clustering por uno basado en grafos que ha permitido una gran mejora en el desempeño del modelo. El algoritmo Kilosort4 ha mejorado prácticamente todas las precisiones de spike sorting en diferentes datasets y actualmente es de los métodos con mejor desempeño. En la Figura 2.6 se puede ver el flujo de trabajo del algoritmo Kilosort4.

2.5. ESTADO DEL ARTE EN EL SPIKE SORTING

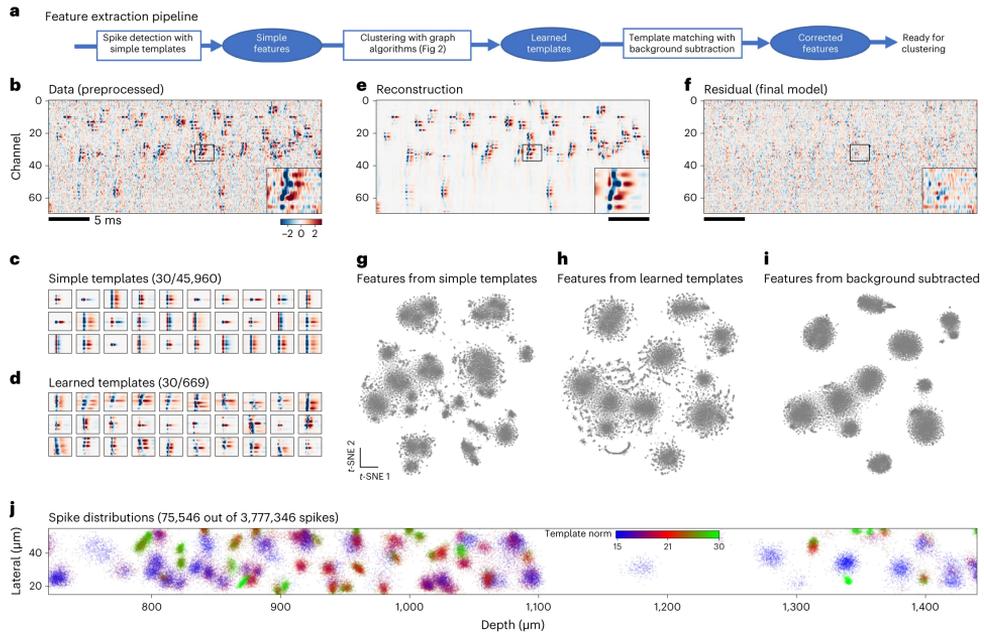


Figura 2.6: Flujo de trabajo del algoritmo Kilosort4. Extraído de [17].

Capítulo 3

Metodología

3.1. Modelo FMM

El modelo FMM ha sido descrito en múltiples ocasiones. La información presentada a continuación se recoge en los siguientes artículos: [2][4][18].

El modelo FMM o Frecuencia Modulada de Möbius es un procedimiento que permite el ajuste de señales oscilatorias combinando una formulación físicamente significativa con buenas propiedades estadísticas y computacionales. El modelo estadístico subyacente es un modelo de señal más error en el que la señal se describe paramétricamente y es formulada como una suma de componentes en el que cada una de las componentes es una señal FMM. Este enfoque paramétrico facilita la interpretación de los elementos esenciales de la señal.

Las principales ventajas del modelo FMM a la hora de ajustar señales oscilatorias frente a otros modelos son varias. Para empezar, las componentes del modelo describen procesos fisiológicos reales, y sus parámetros permiten caracterizar y reproducir las diferentes morfologías que se pueden observar en las ondas que componen la señal. Además, el algoritmo de ajuste proporciona unos parámetros del modelo robustos y precisos evitando problemas de sobreajuste.

Una onda FMM se define de la siguiente manera. Suponiendo que los tiempos de la señal están entre $[0, 2\pi)$, la onda FMM se define como:

$$W(t, v) = A \cos(\phi(t, \alpha, \beta, \omega)) \quad (3.1)$$

donde A es la amplitud y ϕ es la fase:

$$\phi(t, \alpha, \beta, \omega) = \beta + 2 \arctan\left(\omega \tan\left(\frac{t - \alpha}{2}\right)\right) \quad (3.2)$$

Los parámetros de la onda se pueden definir de la siguiente manera:

- A : parámetro de amplitud. Toma cualquier valor real.
- α : parámetro de fase. Varía entre 0 y 2π .
- β : parámetro de forma 1. Varía entre 0 y 2π .
- ω : parámetro de forma 2, semejante al apuntamiento. Varía entre 0 y 1.

Para entender estos parámetros de manera más intuitiva, en las figuras siguientes se presenta un modelo base con unos parámetros prefijados y variaciones sobre estos parámetros para observar como afectan a la onda. El modelo base toma los valores de la tabla Tabla 3.1 y se presenta en la Figura 3.1. En las figuras 3.2, 3.3, 3.4 y 3.5 se presentan variaciones de los parámetros del modelo.

Parámetro	Valor
A	0.5
α	0
β	π
ω	0.5

Tabla 3.1: Parámetros del modelo base de una onda FMM.

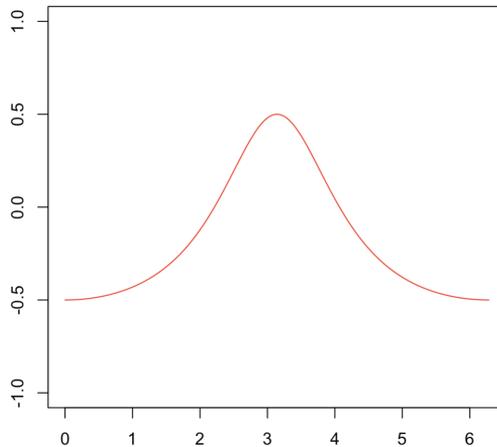
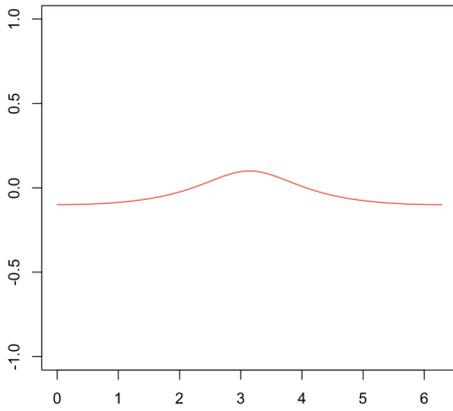


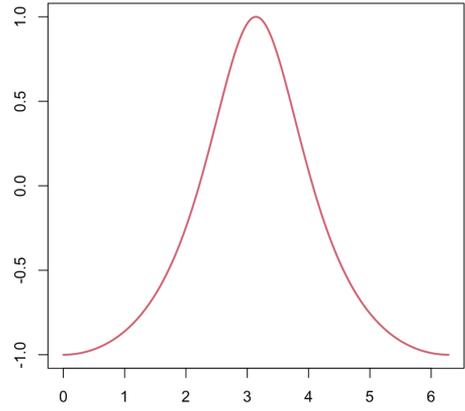
Figura 3.1: Representación de la onda FMM con parámetros de la Tabla 3.1.

En muchas ocasiones, la señal recogida por los electrodos está formada por los impulsos generados por varias neuronas. Para modelar este hecho, se presenta el modelo multicomponente FMM de orden m o FMM $_m$, el cual es un modelo gaussiano con media $\mu(\mathbf{t}; \boldsymbol{\theta})$ definida de la siguiente manera:

$$\mu(\mathbf{t}; \boldsymbol{\theta}) = M + \sum_{j=1}^m W(\mathbf{t}; A_j, \alpha_j, \beta_j, \omega_j) \quad (3.3)$$

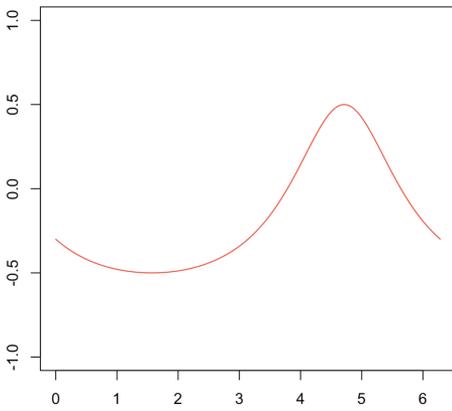


(a) $A = 0,1$

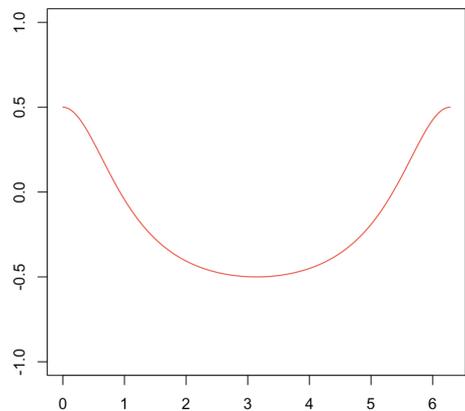


(b) $A = 1$

Figura 3.2: Variación de los parámetros de amplitud en la onda de la Figura 3.1.

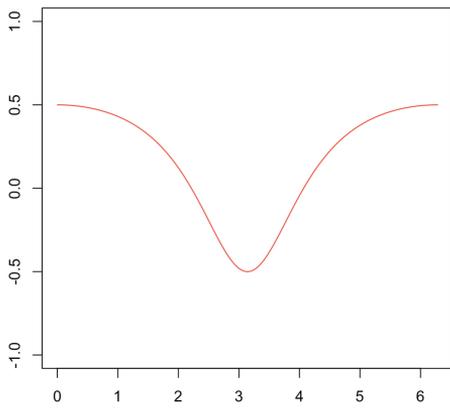


(a) $\alpha = \frac{\pi}{2}$

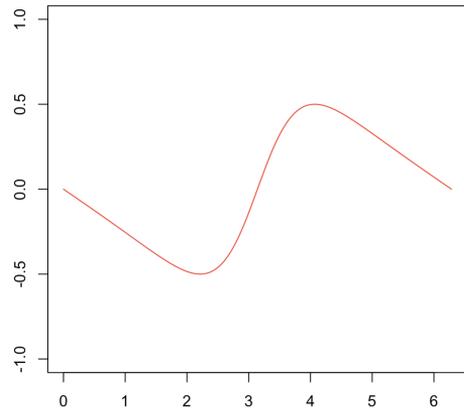


(b) $\alpha = \pi$

Figura 3.3: Variación del parámetro α en la onda de la Figura 3.1.

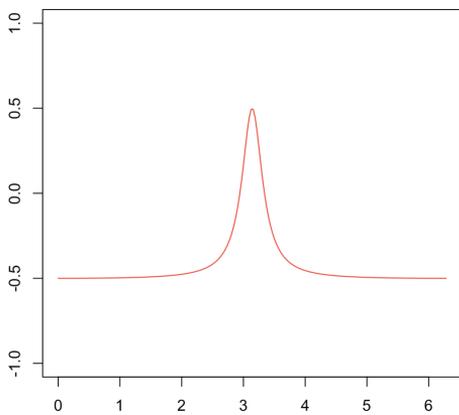


(a) $\beta = 0$

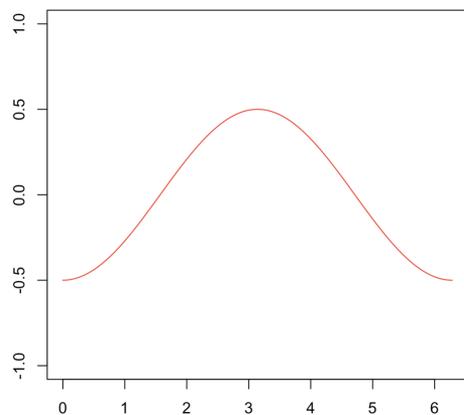


(b) $\beta = \frac{\pi}{2}$

Figura 3.4: Variación del parámetro β en la onda de la Figura 3.1.



(a) $\omega = 0,1$



(b) $\omega = 1$

Figura 3.5: Variación del parámetro ω en la onda de la Figura 3.1.

siendo θ el conjunto de todos los parámetros del modelo. Para asegurar la identificabilidad y unicidad del modelo, se imponen una serie de restricciones. Principalmente, $\alpha_1 \leq \dots \leq \alpha_m \leq \alpha_1$ y $A_1 = \max_{1 \leq j \leq m} A_j$.

Para observar el funcionamiento del ajuste de los modelos explicados, se presentan en las figuras 3.6 y 3.7 el ajuste de una onda con el modelo FMM y con el modelo FMM_m respectivamente. La señal final a ajustar esta compuesta por 2 ondas con parámetros propios, por lo que como se puede comprobar, solo el modelo FMM₂ es capaz de ajustarse a la señal y separarla correctamente.

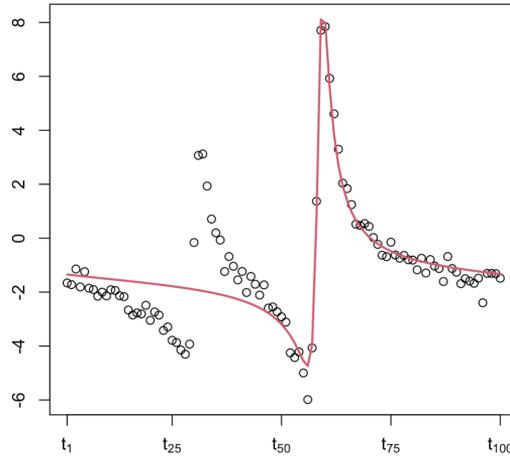
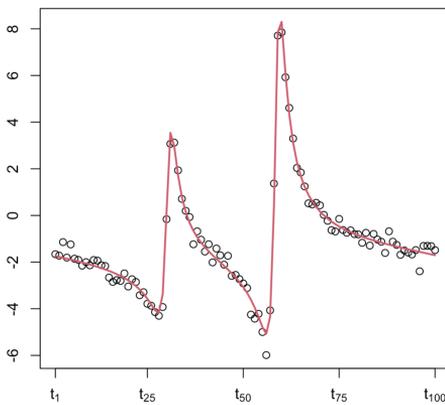
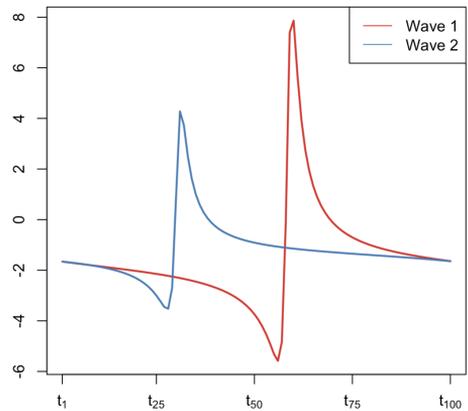


Figura 3.6: Ajuste de una onda FMM a una señal con 2 ondas.



(a) Onda final ajustada



(b) Detalle de las ondas ajustadas

Figura 3.7: Ajuste de una onda FMM₂ a una señal con 2 ondas.

3.1.1. Modelo 3DFMM

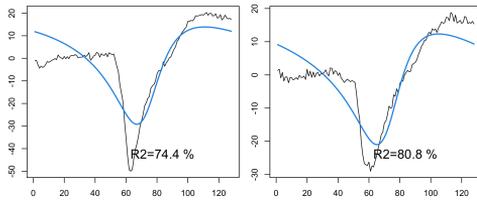
La presentación del modelo que se expone a continuación se recoge en los artículos [5][19].

A la hora de realizar la captura de señales neuronales es habitual contar con varios electrodos para recoger la actividad eléctrica de las neuronas desde diferentes posiciones. Los múltiples canales capturando información permiten representar los spikes desde diferentes perspectivas, y la información que aportan los electrodos o canales se puede aprovechar en el proceso de spike sorting. Sin embargo, el uso de múltiples canales también introduce una mayor complejidad en el procesamiento de la señal, ya que para poder aprovechar esta información se debe llevar a cabo un proceso de detección y alienación de spikes que tenga en cuenta todos los canales.

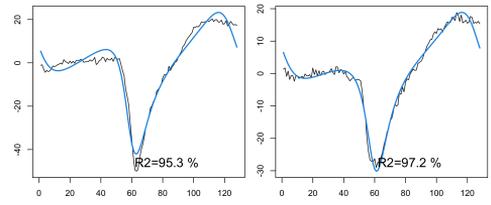
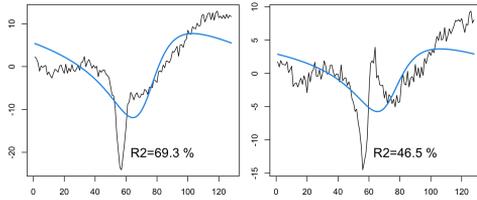
Al haber sido generadas por la misma neurona, las señales capturadas por los diferentes electrodos comparten una serie de características comunes. En la representación FMM de los spikes, los parámetros α y ω permanecen invariantes a la posición del electrodo, es decir, la representación de un mismo impulso en los diferentes canales compartirá los mismos parámetros α y ω [5]. Para modelizar globalmente una misma señal a través de varios canales se utiliza el modelo 3DFMM, el cual calcula la media de la señal en cada canal de la siguiente manera:

$$\mu(\mathbf{t}; \boldsymbol{\theta}_c) = M_c + \sum_{j=1}^m W(\mathbf{t}; A_{j,c}, \alpha_j, \beta_{j,c}, \omega_j); \quad c = 1, \dots, C \quad (3.4)$$

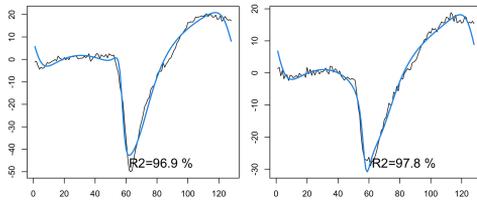
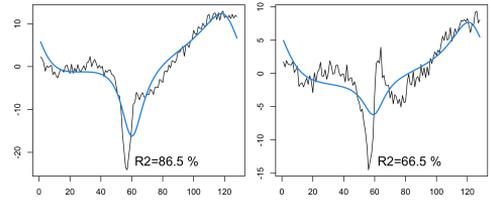
Para ejemplificar el funcionamiento del modelo 3DFMM, se presenta en la Figura 3.8a el ajuste del modelo $3DFMM_1$ a una señal de uno de los datasets que se van a utilizar en este proyecto. Además, para observar las diferencias en el número de componentes de la señal, en las figuras 3.8b y 3.8c se representa el ajuste del modelo 3DFMM de la misma señal con 2 y 3 componentes respectivamente. Finalmente, en la Figura 3.8d se presentan las 3 componentes ajustadas por el modelo correspondiente.



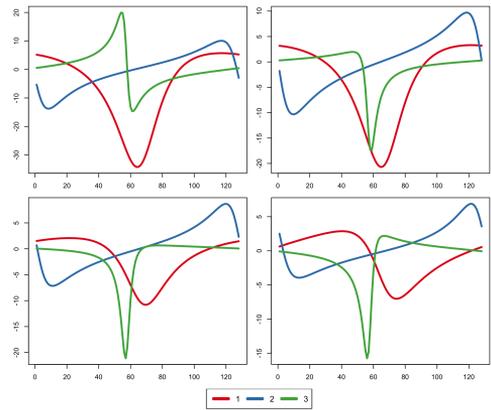
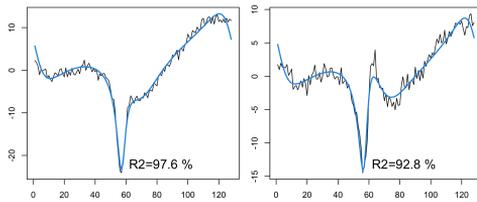
(a) Ajuste del modelo $3DFMM_1$.



(b) Ajuste $3DFMM_2$.



(c) Ajuste del modelo $3DFMM_3$.



(d) Componentes del modelo $3DFMM_3$ ajustado.

Figura 3.8: Ajuste y componentes del modelo $3DFMM$ a una señal con 4 canales.

Como se puede comprobar, el modelo con 3 componentes ajusta adecuadamente la señal y parece evidenciar que el uso del modelo $3DFMM_3$ puede ser adecuado para el spike sorting. Además, observando las componentes, se puede ver como estas mantienen las características de los parámetros α y ω de fase y apuntamiento en todos los canales.

3.2. Modelo MixFMM

Para llevar a cabo el proceso de clasificación de los spikes con un solo canal se va a utilizar el modelo MixFMM. El modelo de mixtura FMM o MixFMM es un modelo que asume que los datos provienen de una combinación de k distribuciones Gaussianas diferentes. La descripción detallada de este modelo a partir de la cual se realiza la exposición de esta sección se recoge en el artículo [4].

La densidad de los datos asumiendo este modelo con parámetros Ψ es la siguiente:

$$f(\mathbf{x}|\Psi) = \sum_{k=1}^K \gamma_k N(\mathbf{x}; \mu(\mathbf{t}; \boldsymbol{\theta}_k), \sigma_k^2 \mathbf{I}_p), \quad (3.5)$$

donde $\Psi = (\gamma_1, \dots, \gamma_K, \boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K, \sigma_1, \dots, \sigma_K)$, γ_k son las proporciones de la mixtura con $\gamma_k > 0$ y $\sum_{k=1}^K \gamma_k = 1$. $N(\dots)$ es la densidad de x asumiendo que sigue una distribución Normal de media $\mu(\mathbf{t}; \boldsymbol{\theta}_k)$ correspondiendo a la onda FMM _{m} descrita en la Ecuación 3.3 y varianza $\sigma_k^2 \mathbf{I}_p$, donde \mathbf{I}_p es la matriz identidad $p \times p$.

Para describir el funcionamiento del modelo MixFMM, a continuación se va a reproducir el artículo [4] donde se presenta este modelo por primera vez.

3.2.1. Estimación de máxima verosimilitud mediante el algoritmo EM

El algoritmo de esperanza-maximización o algoritmo EM es un mecanismo extremadamente útil a la hora de estimar parámetros en modelos probabilísticos con datos incompletos. Por este mismo motivo, es muy popular en los modelos biológicos, en los que se disponen de procedimientos avanzados para estimar parámetros a partir de observaciones, pero que suelen estar incompletos [20].

El algoritmo EM, como su nombre indica, consta de 2 pasos. Una vez se han inicializado los parámetros en la primera iteración del algoritmo, se procede al normal funcionamiento de este:

1. Esperanza: se calcula la esperanza de la log-verosimilitud con los datos observados y los parámetros de la iteración actual.
2. Maximización: se modifican los parámetros del modelo para tratar de maximizar el valor de la log-verosimilitud.

De esta manera, tras un determinado número de iteraciones se podría alcanzar el máximo global de la verosimilitud.

Este modelo será el utilizado para realizar la clasificación de picos. Los picos se modelarán como ondas FMM y se les asignará una probabilidad de pertenecer a cada grupo. La verosimilitud del conjunto de datos completo de este modelo MixFMM se define de la siguiente manera:

$$\log L_c(\Psi) = \sum_{i=1}^n \sum_{k=1}^K Z_{ik} \log(\gamma_k N(\mathbf{x}_i; \mu(\mathbf{t}; \boldsymbol{\theta}_k), \sigma_k^2 \mathbf{I}_p)) \quad (3.6)$$

donde Z_{ik} es una variable binaria que indica si la muestra x_i pertenece al modelo FMM k -ésimo. El proceso entonces comenzaría inicializando los parámetros del modelo, y se irían actualizando en cada iteración.

Paso E: Esperanza

En el paso E, se calcula la esperanza de la verosimilitud de todo el conjunto de datos con los parámetros $\Psi^{(q)}$:

$$Q(\Psi; \Psi^{(q)}) = \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} \log(\gamma_k^{(q)} N(\mathbf{x}_i; \mu(\mathbf{t}; \boldsymbol{\theta}_k), \sigma_k^2 \mathbf{I}_p)) \quad (3.7)$$

donde $\tau_{ik}^{(q)}$ es la probabilidad a posteriori de que la curva i pertenezca al modelo k -ésimo, y se define de la siguiente manera:

$$\tau_{ik}^{(q)} = P(Z_{ik} = 1 | x_i; \Psi^{(q)}) = \frac{\gamma_k^{(q)} N(\mathbf{x}_i; \mu(\mathbf{t}; \boldsymbol{\theta}_k), \sigma_k^2 \mathbf{I}_p)}{\sum_{h=1}^K \gamma_h^{(q)} N(\mathbf{x}_i; \mu(\mathbf{t}; \boldsymbol{\theta}_h), \sigma_h^2 \mathbf{I}_p)}, \forall i \in \{1, \dots, n\}, k = 1, \dots, K \quad (3.8)$$

Paso M: Maximización

Una vez se obtiene el valor de la log-verosimilitud, se procede a tratar de maximizar la Ecuación 3.7 actualizando los parámetros. La actualización de los parámetros se realiza de la siguiente manera:

$$\boldsymbol{\theta}_k^{(q+1)} = \arg \min_{\boldsymbol{\theta} \in \Theta} \sum_{j=1}^p \left(\bar{x}_k^{(q)}(t_j) - \mu(t_j; \boldsymbol{\theta}) \right)^2, \quad k = 1, \dots, K \quad (3.9)$$

siendo $\bar{x}_k^{(q)}(t_j)$ la onda media para el clúster k en la iteración q . Esta onda media por clúster se calcula como la media por puntos de todas las ondas del dataset ponderadas por la probabilidad de que pertenezcan al clúster k -ésimo. El cálculo exacto de la onda media se puede ver en la Ecuación 3.10.

$$\bar{x}_k^{(q)}(t_j) = \frac{\sum_{i=1}^n \tau_{ik}^{(q)} x_i(t_j)}{\sum_{i=1}^n \tau_{ik}^{(q)}}, \quad k = 1, \dots, K, \quad j = 1, \dots, p \quad (3.10)$$

Finalmente, se actualizan los estimadores de la desviación típica y las proporciones de la mezcla mediante las ecuaciones Ecuación 3.11 y Ecuación 3.13 respectivamente.

$$\sigma_k^{(q+1)} = \sqrt{\frac{\sum_{i=1}^n \tau_{ik}^{(q)} \|\mathbf{x}_i(\mathbf{t}) - \mu(\mathbf{t}; \boldsymbol{\theta}_k^{(q+1)})\|^2}{p \sum_{i=1}^n \tau_{ik}^{(q)}}}, \quad k = 1, \dots, K \quad (3.11)$$

La actualización de la desviación típica en la práctica es más sencilla, ya que se impone una restricción de homocedasticidad en la que $\sigma = \sigma_1 = \dots = \sigma_K$, por lo que todas las varianzas son iguales y por tanto dicha ecuación se reduce a una sola actualización:

$$\sigma^{(q+1)} = \sqrt{\frac{\sum_{k=1}^K \sum_{i=1}^n \tau_{ik}^{(q)} \|\mathbf{x}_i(\mathbf{t}) - \mu(\mathbf{t}; \boldsymbol{\theta}_k^{(q+1)})\|^2}{pN}} \quad (3.12)$$

$$\gamma_k^{(q+1)} = \frac{\sum_{i=1}^n \tau_{ik}^{(q)}}{n}, \quad k = 1, \dots, K \quad (3.13)$$

El flujo de trabajo completo puede verse resumido en la Figura 3.9. En dicha figura se observa el funcionamiento del algoritmo desde la obtención de los datos hasta la predicción final de los clústers.

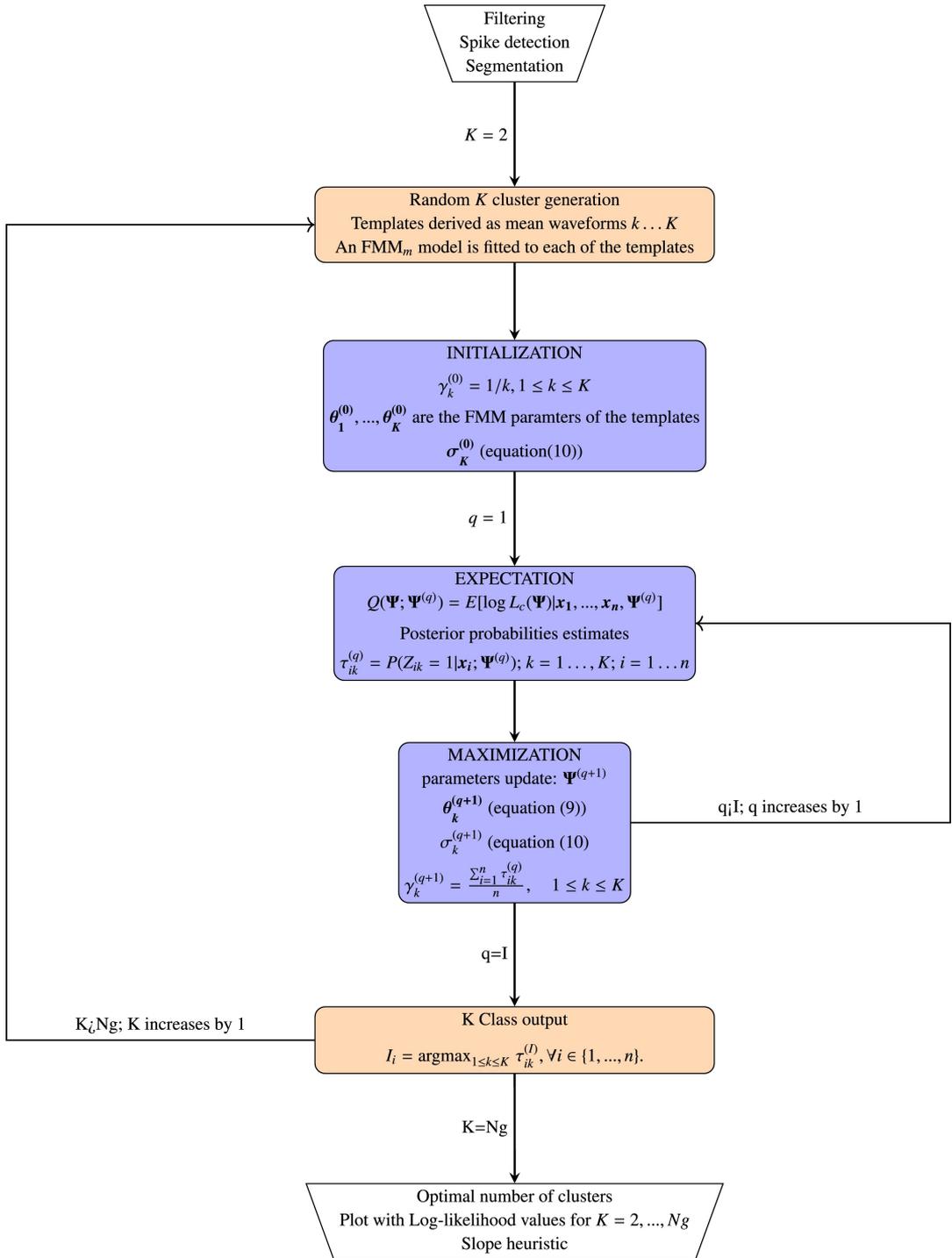


Figura 3.9: Flujo de trabajo del algoritmo MixFMM. Extraído de [4].

3.3. Modelo Mix3DFMM

Se presenta en esta sección la contribución más novedosa del trabajo: el modelo Mix3DFMM. El objetivo de este modelo es ser capaz de utilizar la información presente en múltiples canales para mejorar la clasificación de los spikes. La principal diferencia con el modelo MixFMM radica en la necesidad de trabajar con múltiples canales en vez de uno solo, y para ello se sustituye el ajuste del modelo FMM por el ajuste en todos los canales del modelo 3DFMM. A continuación, se describe en detalle el funcionamiento del modelo Mix3DFMM y cómo se adapta el algoritmo EM para trabajar con este nuevo modelo.

El primer cambio que se realiza es la definición de la función de densidad, la cual se modifica para adaptarse al nuevo modelo teniendo en cuenta todos los canales. Esta densidad se calcula de la siguiente manera:

$$f(\mathbf{x}|\Psi) = \sum_{k=1}^K \sum_{c=1}^C \gamma_k N(\mathbf{x}; \mu(\mathbf{t}; \boldsymbol{\theta}_{k,c}), \sigma_{k,c}^2 \mathbf{I}_p), \quad (3.14)$$

Una vez realizado este cambio, se procede a definir el funcionamiento del algoritmo EM para este nuevo modelo.

3.3.1. Estimación de máxima verosimilitud para el modelo Mix3DFMM

El objetivo del algoritmo es obtener la clasificación de los spikes con la mayor verosimilitud posible. Para maximizar la verosimilitud, se va a utilizar de nuevo el algoritmo EM presentado en la Subsección 3.2.1. Este algoritmo consta de dos pasos, el paso de esperanza y el paso de maximización, los cuales se van a adaptar al nuevo modelo.

Paso E: Esperanza

La esperanza de la log-verosimilitud se redefine teniendo en cuenta la nueva definición de la densidad del modelo. La fórmula permanece igual y lo que se modifica realmente son las probabilidades a posteriori de pertenecer a cada clúster. Esta verosimilitud se obtiene de la siguiente manera:

$$Q(\Psi; \Psi^{(q)}) = \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} \log(\gamma_k^{(q)} N(\mathbf{x}; \mu(\mathbf{t}; \boldsymbol{\theta}_{k,c}), \sigma_{k,c}^2 \mathbf{I}_p)) \quad (3.15)$$

Donde $\tau_{ik}^{(q)}$ es la probabilidad a posteriori de que la curva i pertenezca al modelo k -ésimo, y se redefine de la siguiente manera, agregando las verosimilitudes de todos los canales:

$$\tau_{ik}^{(q)} = \frac{\sum_{c=1}^C \gamma_k N(\mathbf{x}_{i,c}; \mu(\mathbf{t}; \boldsymbol{\theta}_{k,c}), \sigma_{k,c}^2 \mathbf{I}_p)}{\sum_{h=1}^K \sum_{c=1}^C \gamma_k N(\mathbf{x}_{i,c}; \mu(\mathbf{t}; \boldsymbol{\theta}_{h,c}), \sigma_{h,c}^2 \mathbf{I}_p)}, \forall i \in \{1, \dots, n\}, k = 1, \dots, K, c = 1, \dots, C \quad (3.16)$$

Paso M: Maximización

Posteriormente, la actualización de los parámetros se realiza mediante el ajuste del modelo 3DFMM sobre las ondas medias de cada cluster calculadas mediante la Ecuación 3.17. Finalmente, la actualización de las desviaciones típicas al ser los canales independientes, se realiza de manera individual para cada canal como si se estuviera en el caso del modelo con un solo canal.

$$\bar{x}_{k,c}^{(q)}(t_j) = \frac{\sum_{i=1}^n \tau_{ik}^{(q)} x_{i,c}(t_j)}{\sum_{i=1}^n \tau_{ik}^{(q)}}, \quad k = 1, \dots, K, \quad c = 1, \dots, C, \quad j = 1, \dots, p \quad (3.17)$$

$$\sigma_c^{(q+1)} = \sqrt{\frac{\sum_{k=1}^K \sum_{i=1}^n \tau_{ik}^{(q)} \|\mathbf{x}_{i,c}(\mathbf{t}) - \mu(\mathbf{t}; \boldsymbol{\theta}_{k,c}^{(q+1)})\|^2}{pN}}, \quad c = 1, \dots, C \quad (3.18)$$

3.3.2. Selección de canales

La mayoría de algoritmos GMM asumen que todas las características de las observaciones, en este caso canales, contribuyen de la misma manera al proceso de agrupamiento. Sin embargo, esto no suele ser cierto, y el uso de canales irrelevantes puede impedir al modelo realizar correctamente la clasificación. Este es un problema común que ya ha sido documentado en la literatura [21] [22]. Para tratar de evitarlo, es conveniente llevar a cabo una selección de canales para que el modelo utilice únicamente aquellos que contienen información relevante y serán más útiles para el proceso de clasificación. Esta selección se puede llevar a cabo de muchas maneras. A continuación, se presentan algunas de las estrategias seguidas en la literatura:

- **Penalización en la verosimilitud:** se sigue una aproximación similar a la de la regresión penalizada, introduciendo un término de penalización en la función de verosimilitud del modelo para regularizar la estimación de parámetros del algoritmo EM[23].
- **Índice de relevancia:** se calcula un índice de la relevancia de cada canal basado en cómo varían las probabilidades a posteriori de pertenecer a un clúster en función de la presencia o ausencia del canal. Aquellos con baja relevancia no se utilizarán [22].

- **Selección basada en la varianza de los clústers:** se define un canal como irrelevante si la varianza de sus observaciones en un clúster es cercana a la varianza global en todos los clústers. Si este es el caso, el canal no aporta información relevante para separar los clústers y no se utilizará en el proceso de clasificación [21].

Capítulo 4

Diseño experimental

En este capítulo se va a llevar a cabo el proceso de spike sorting multicanal sobre 2 conjuntos de datos diferentes. Además, previamente se realizará este proceso sobre un único canal de cada conjunto para comparar las precisiones obtenidas mediante ambos procesos. De esta manera, se podrá obtener una medida de comparación de ambas técnicas y ver así si el uso de un sistema multicanal mejora la precisión del proceso de spike sorting.

4.1. Conjunto de datos

Existen numerosos conjuntos de datos de distintas fuentes para realizar el proceso de spike sorting. Sin embargo, la mayoría de estos datasets contienen datos de una dimensionalidad muy alta, con un número de canales muy elevado o un número de grupos de neuronas muy grande. Además, muchos de estos conjuntos tienen los datos en crudo, es decir, no han sido procesados y requieren realizar un procedimiento de preprocesado en el que se elimine el ruido, se detecten los picos y se alineen los canales para poder trabajar adecuadamente con ellos. Por ello, se ha decidido trabajar sobre conjuntos de datos sintéticos que simulan el comportamiento de un sistema multicanal. Estos datos sintéticos se han creado con la librería MEArec y proporcionan los picos ya recortados y adecuadamente etiquetados. De esta manera, se podrá trabajar sin dificultad y se podrá centrar el trabajo de este proyecto en comprobar el funcionamiento del modelo FMM multicanal para el proceso de spike sorting.

4.1.1. Librería MEArec

La librería MEArec es un paquete de Python que permite generar grabaciones extracelulares sintéticas de neuronas con múltiples electrodos [24]. Esta librería permite generar las señales deseadas con facilidad y realismo. Para generar una grabación, se deben elegir unas plantillas o *templates* de los picos de las neuronas y el tipo de electrodos que se utilizarán.

Además, permite controlar el número de canales, el número de neuronas y otros parámetros. La ventaja de usar esta librería está en la facilidad de extracción de los picos y etiquetas de los mismos, ya que ofrece funciones para extraerlos directamente de manera que estén alineadas y procesadas.

4.1.2. Datos utilizados

Los datos han sido generados para el artículo original de la librería MEArec [24], y se han descargado desde [25]. Estos conjuntos de datos contienen grabaciones simuladas de señales extracelulares con diferentes características y dificultades, lo que permitirá comprobar correctamente el desempeño del modelo. Para poder utilizar estos datos en R se han tenido que cargar en Python y realizar una serie de operaciones poder exportarlos a un formato compatible y adecuado para el programa de R.

El primero de los datasets contiene información de 1721 spikes capturadas en una grabación de 60 segundos. Las señales han sido recogidas por 32 canales y generadas por 4 tipos de neuronas. Por otro lado, el segundo dataset contiene información de 1651 spikes capturadas en una grabación de 30 segundos. El número de grupos aumenta a 6 y el de canales se reduce a 4. En la Tabla 4.1 se presenta un resumen con las características de estos datasets..

Conjunto de Datos	Canales	Neuronas	Individuos
Dataset 1	32	4	1721
Dataset 2	4	6	1651

Tabla 4.1: Características de los conjuntos de datos utilizados.

Una muestra de los tipos de señales de cada tipo se puede ver en las figuras 4.1 y 4.2. En estas figuras se puede ver la señal media de cada tipo de neurona. Además, se puede apreciar como a en el dataset 1 las señales son muy similares entre sí, mientras que en el 2 hay más variabilidad entre señales, lo que le confiere ese nivel mayor de dificultad. Como existen diversos canales, las figuras muestran la señal del canal con mayor valor de amplitud.

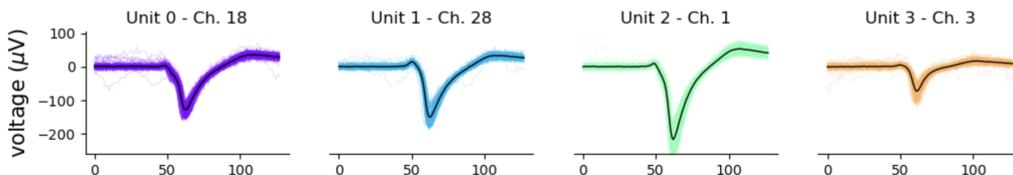


Figura 4.1: Señales del dataset 1.

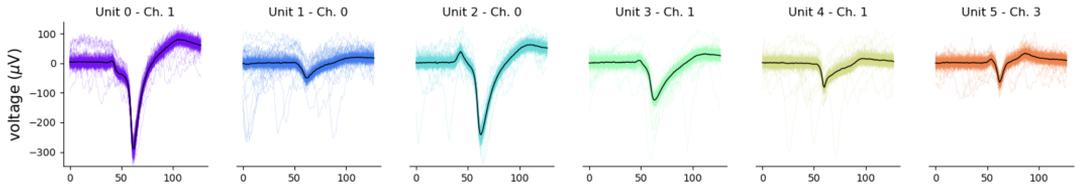


Figura 4.2: Señales del dataset 2.

Para observar como se muestran los diferentes canales de una misma grabación, en la Figura 4.3 se pueden ver las señales medias de los 16 primeros canales de cada grupo del dataset 1. En esta figura, se puede observar como la señal de cada canal es diferente, e incluso en alguno de ellos ese pico desaparece y simplemente se puede apreciar ruido. Además, en la Figura 4.4 se pueden ver las ondas 3DFMM ajustadas de los 16 primeros. Comparando estas 2 figuras, se puede comprobar como el modelo FMM es capaz de ajustar casi perfectamente las ondas de cada canal. Esta comparación se hace igualmente con el dataset 2 y se puede ver en las figuras 4.5 y 4.6.

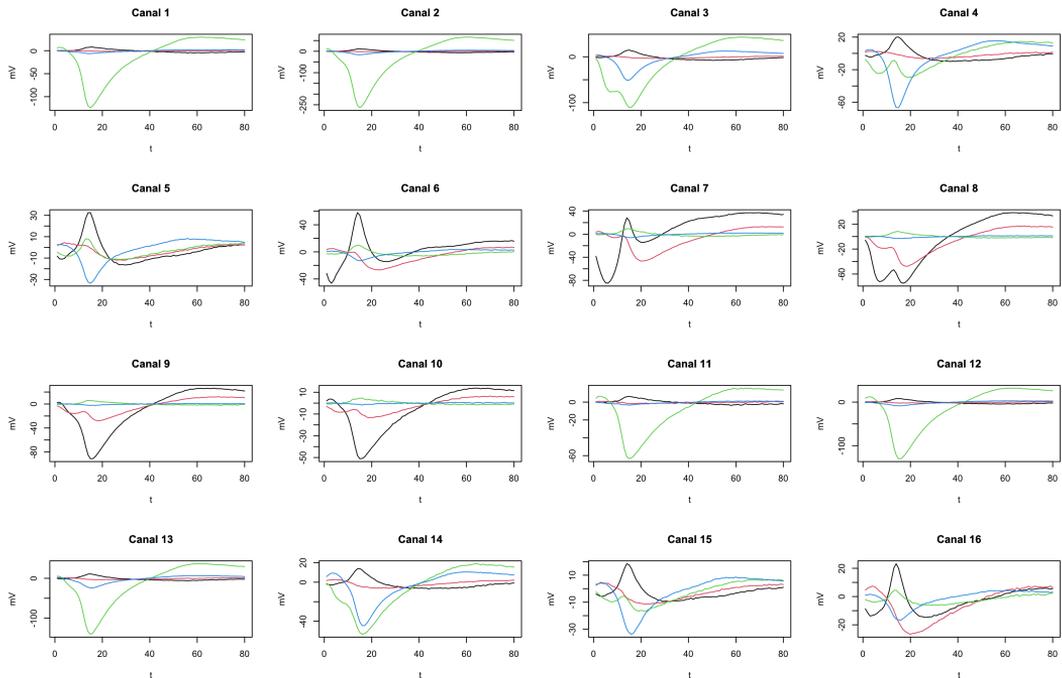


Figura 4.3: Señales medias de los 16 primeros canales de las clases del dataset 1.

4.1. CONJUNTO DE DATOS

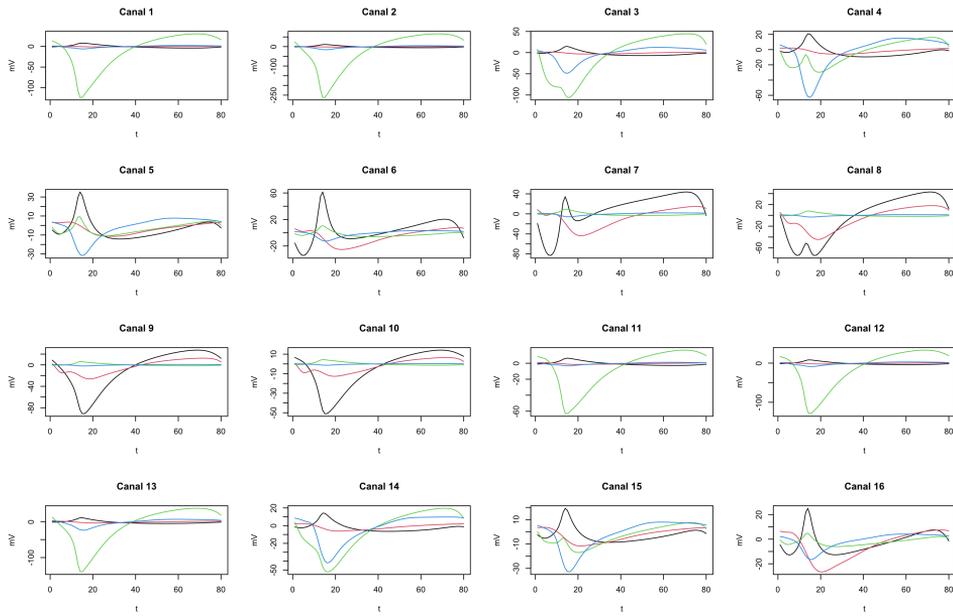


Figura 4.4: Señales medias ajustadas de los 16 primeros canales de las clases del dataset 1.

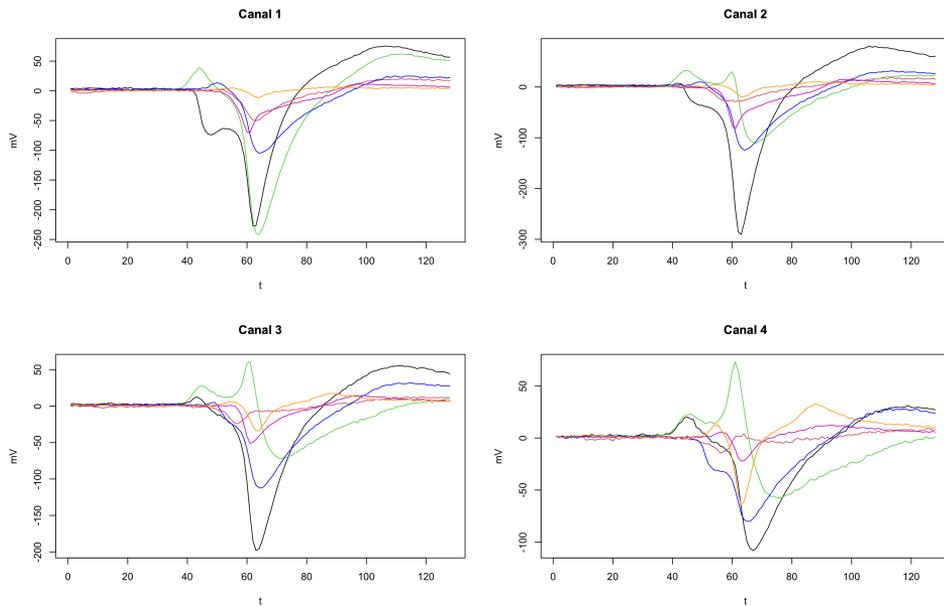


Figura 4.5: Señales medias por canal de las clases del dataset 2.

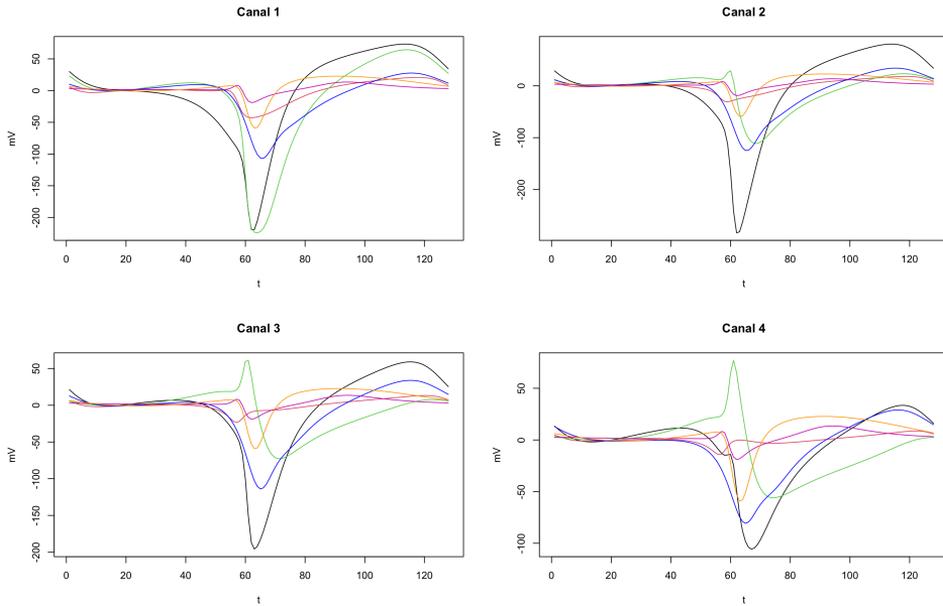


Figura 4.6: Señales medias ajustadas por canal de las clases del dataset 2.

4.2. Implementación monocanal

El primer paso en la ejecución del proceso de spike sorting de este trabajo será implementar el modelo MixFMM para un solo canal. Para ello, se han utilizado las funciones del paquete FMM [26] para ajustar las ondas y obtener los parámetros de las mismas. Además, se han implementado el resto de funciones descritas en la Sección 3.2 para poder realizar el proceso de clasificación. Finalmente, para simplificar el proceso, se ha decidido utilizar un número de clústers prefijado e igual al número real de grupos. Además, después de analizar las señales, se ha decidido utilizar un modelo con 3 componentes para modelar las señales adecuadamente, es decir, el modelo será MixFMM₃. La adecuación de este número de componentes se puede comprobar en la Figura 3.8.

La elección del canal óptimo para la comparación con el método multicanal se basará en el valor de la verosimilitud, es decir, se seleccionará el canal que consiga la mayor verosimilitud. Para obtener los valores de verosimilitud, se han realizado 10 repeticiones del proceso de spike sorting con el modelo MixFMM para cada uno de los canales de ambos datasets. De esta manera, se han obtenido los máximos valores de verosimilitud alcanzables por cada canal. Una vez obtenidos los resultados, se ha seleccionado aquel canal que ha obtenido la mayor verosimilitud de todos los canales y se ha calculado la precisión que se obtiene con la configuración final de dicho canal. Este valor de precisión será el que se utilice para comparar con el resultado del modelo multicanal.

4.3. Implementación multicanal

En esta sección se presenta la implementación del modelo Mix3DFMM. Para ello, se han definido las funciones descritas en la Sección 3.3. Además, para llevar a cabo el ajuste del modelo 3DFMM se ha utilizado el paquete 3DFMM [5] de R. De igual manera que en el caso monocal, el número de clústeres iniciales se fija de antemano al valor real. El modelo utilizado para el ajuste de las señales es Mix3DFMM₃, es decir, se utilizarán 3 componentes para modelar las señales.

Para seleccionar el mejor resultado, se ha ejecutado el algoritmo 10 veces y se ha seleccionado el resultado con mayor verosimilitud. Una vez obtenida la configuración que obtiene mayor verosimilitud, se ha calculado su precisión y se ha comparado con la del caso monocal.

Capítulo 5

Resultados

En este capítulo, se van a presentar los resultados obtenidos a partir de los diferentes experimentos detallados en el capítulo anterior. Estos resultados permitirán comparar el rendimiento de ambos modelos para ver si la inclusión de más canales mejora los resultados de clasificación.

5.1. Resultados del modelo monocanal

Los resultados obtenidos con el modelo que utiliza un solo canal servirán de referencia para comparar con los obtenidos por el nuevo modelo. Estos resultados han sido buenos, aunque tienen margen de mejora para el modelo multicanal.

5.1.1. Dataset 1

El primer dataset era el que contenía datos de 32 canales y 4 tipos de señal. Los resultados obtenidos son los contenidos en la Tabla 5.1. Observando la tabla, se comprueba como los valores de verosimilitud son bastante diferentes entre sí y oscilan entre -515871.5 que obtiene el canal 2, y -421387.5, que es el valor obtenido por el canal 22. Por lo tanto, el canal 22 será el tomado como referencia para compararlo con los resultados del modelo Mix3DFMM. En la Tabla 5.2 se presenta el valor de precisión obtenido por este canal, que es del 81.87%. Este valor es el que se tratará de superar con el nuevo modelo.

Canal	Verosimilitud	Canal	Verosimilitud
1	-455972.6	17	-467291.1
2	-515871.5	18	-491469.2
3	-484922.8	19	-482977.9
4	-468959.2	20	-453009.5
5	-446129.4	21	-427707.0
6	-470463.8	22	-421387.5
7	-489662.9	23	-428698.6
8	-476491.3	24	-438104.7
9	-469170.4	25	-435832.0
10	-441171.3	26	-428847.2
11	-434484.8	27	-431855.4
12	-457583.2	28	-454349.8
13	-468341.3	29	-484436.5
14	-457670.8	30	-480559.2
15	-441975.0	31	-453919.5
16	-442972.1	32	-431922.6

Tabla 5.1: Verosimilitud por canal obtenida del dataset 1.

Canal	Verosmilitud	Precisión
22	-421387.5	0.8187

Tabla 5.2: Precisión obtenida por el canal 22 en el dataset 1.

5.1.2. Dataset 2

El segundo dataset contiene información de 4 canales y 6 tipos de señales. Los resultados obtenidos se presentan en la Tabla 5.3. Como se puede observar, los valores de verosimilitud de nuevo son bastante diferentes entre sí, oscilando entre -906607.3 del canal 1 y -846844.7 del canal 4, que es el que se toma como referencia para comparar con el modelo multicanal. En la Tabla 5.4 se muestra el valor de precisión obtenido por este canal, que es del 86.86 %. Este valor será el que se tratará de superar con el modelo multicanal.

Canal	Verosimilitud
1	-906607.3
2	-897776.7
3	-868554.5
4	-846844.7

Tabla 5.3: Verosimilitud por canal para el Dataset 2.

Canal	Verosimilitud	Precisión
4	-846844.7	0.8686

Tabla 5.4: Precisión obtenida por el canal 4 en el dataset 2.

5.2. Resultados del nuevo modelo multicanal

Los resultados obtenidos mediante el modelo multicanal tratarán de mejorar los resultados obtenidos anteriormente. Para ver si este modelo tiene un buen funcionamiento con los datos multicanal, no solo se tendrán en cuenta los resultados de precisión, sino que también se comprobará que el algoritmo EM converge correctamente a la vez que la precisión mejora.

5.2.1. Dataset 1

Los resultados obtenidos por el modelo Mix3DFMM para el dataset 1 muestran como el modelo ha funcionado, siendo capaz de mejorar los resultados obtenidos por el modelo monocanal. En la Tabla 5.5 se pueden ver los valores obtenidos por el modelo junto a los resultados que obtenía el mejor canal individualmente. En este caso, el modelo multicanal ha conseguido una precisión final del 94.19%, lo que supone una mejora del 12.32% respecto al modelo monocanal. Estos resultados muestran que el modelo multicanal es capaz de usar la información de los diferentes canales para mejorar la precisión de la clasificación, ya que usando los canales independientemente, en ningún caso se conseguían valores similares al obtenido con este modelo.

	Precisión
Canal 22	0.8187
Completo	0.9419

Tabla 5.5: Comparativa de los resultados obtenidos por ambos modelos para el dataset 1.

En la Figura 5.1 se presentan los gráficos de la evolución de la verosimilitud y la precisión en cada iteración. Como se puede observar, la verosimilitud comienza en un valor muy bajo y va aumentando hasta converger en un valor que ronda los -415000. Por otro lado, la precisión evoluciona de una forma similar, con un escalón al principio y posteriormente convergiendo al valor óptimo que ronda el 94% de aciertos. Viendo estos resultados, se puede comprobar que el algoritmo ha funcionado correctamente, aumentando la verosimilitud hasta converger en un valor estable obteniendo entonces un buen valor de precisión final.

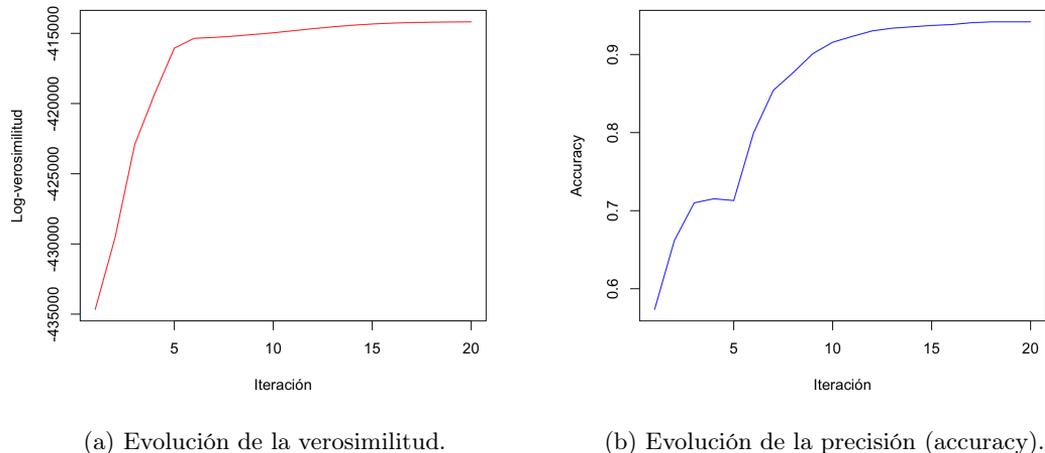


Figura 5.1: Evolución de la verosimilitud y la precisión durante el entrenamiento del modelo multicanal para el dataset 1.

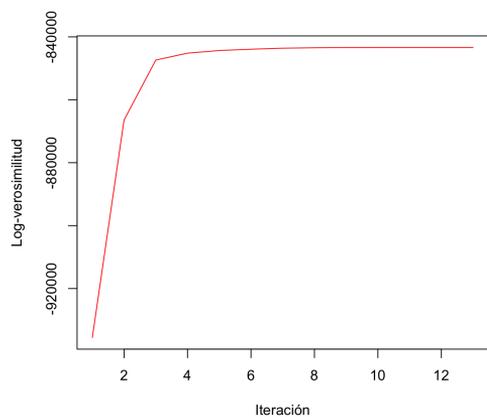
5.2.2. Dataset 2

Los resultados obtenidos por el modelo Mix3DFMM para el dataset 2 son muy similares a los del dataset 1. De nuevo, el modelo ha conseguido mejorar los resultados obtenidos por el modelo monocanal. En la Tabla 5.6 se presentan la comparativa de los resultados obtenidos por el modelo multicanal y el canal 4, que es el que mejor resultado obtenía de manera individual. En este caso, el modelo multicanal ha sido capaz de clasificar correctamente el 94.97 % de las señales, lo que supone una mejora del 8.11 % respecto al modelo monocanal. Los resultados obtenidos vuelven a demostrar como el modelo que agrega la información de los diferentes canales es capaz de mejorar la precisión de la clasificación.

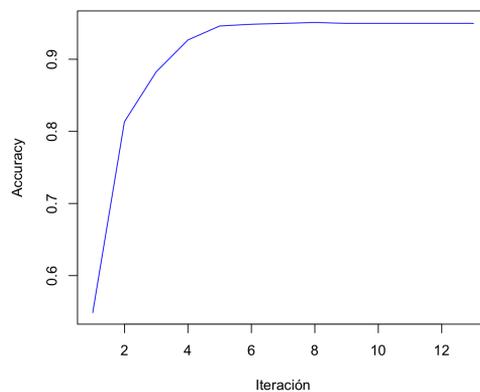
	Precisión
Canal 4	0.8686
Completo	0.9497

Tabla 5.6: Resultados obtenidos por el modelo multicanal para el dataset 2.

En la Figura 5.2 se presentan de nuevo los gráficos de la evolución de la verosimilitud y la precisión por iteración. En este caso, se puede ver una progresión muy uniforme. La verosimilitud y la precisión van aumentando de manera rápida hasta alcanzar el valor de convergencia final.



(a) Evolución de la verosimilitud.



(b) Evolución de la precisión (accuracy).

Figura 5.2: Evolución de la verosimilitud y la precisión durante el entrenamiento del modelo multicanal para el dataset 2.

Capítulo 6

Conclusiones

En este trabajo se ha desarrollado un novedoso método de clasificación de señales oscilatorias multicanal, orientado a su aplicación en el proceso de spike sorting. Inicialmente, se llevó a cabo una revisión de la literatura referente al spike sorting y al modelo FMM, estableciendo una base sólida sobre la que fundamentar el nuevo modelo. El desempeño del modelo Mix3DFMM se evaluó comparándolo con los resultados conseguidos por el modelo MixFMM monocanal.

Los resultados obtenidos evidencian que el modelo multicanal funciona y es capaz de aprovechar la información de los diferentes canales para mejorar la precisión de la clasificación. En los dos conjuntos de prueba utilizados, el modelo multicanal superó al monocanal, alcanzando mejoras de hasta un 12% de precisión. Además, este avance se logró sin necesidad de realizar una selección de canales, lo que sugiere que en contextos más complejos, la incorporación de técnicas que realicen este proceso podrían mejorar aún más los resultados.

A pesar de que los resultados obtenidos demuestran el potencial del modelo Mix3DFMM, no se ha podido comparar con otros algoritmos debido a las restricciones temporales. Para validar completamente la metodología, es necesario realizar pruebas comparativas con otros modelos en el estado del arte del spike sorting como Kilosort4 [17] o GEMSort[27]. Esta comparativa será realizada previsiblemente en un futuro y permitirá valorar al modelo de forma objetiva frente a otros métodos consolidados. Se espera que el modelo sea muy competitivo a la vista de los resultados obtenidos anteriormente por el modelo MixFMM frente a sus competidores.

6.1. Líneas de trabajo futuras

Ante los resultados obtenidos, queda demostrado el prometedor funcionamiento del modelo Mix3DFMM. Aun así, se puede seguir profundizando en el estudio de este modelo y desarrollando y puliendo sus características. Para ello, se proponen las siguientes mejoras para implementar en el futuro:

- **Pruebas con un mayor número de datasets, incluidos datos reales:** aunque se han realizado pruebas para comprobar el funcionamiento del modelo, es conveniente seguir estudiando el funcionamiento sobre más datasets de diferente procedencia. Además, el uso de datos reales y con resultados más heterogéneos también proporcionará una mejor idea del desempeño de este modelo.
- **Comparativa con otros modelos:** para validar completamente el modelo Mix3DFMM, es necesario realizar una comparativa frente a otros modelos multicanal, lo cual proporcionará una idea real de su utilidad y posible aplicación.
- **Selección de canales:** los conjuntos de datos sobre los que se ha trabajado no tenían demasiados canales ni eran excesivamente ruidosos, sin embargo, para trabajar con otros datasets con estas características, se podría implementar un sistema de selección de canales que permita elegir los más adecuados para distinguir cada tipo de neurona de manera dinámica en cada iteración. Esto puede ayudar a mejorar la precisión en gran medida, ya que se eliminan los factores que aportan ruido y solo se trabaja sobre las señales más relevantes de cada canal.
- **Optimización del modelo:** si bien el modelo es funcional y no tiene un gran coste computacional, a medida que aumenta el número de canales y especialmente el número de grupos, el tiempo de ejecución se dispara. Por ello, sería conveniente optimizar el flujo de trabajo del modelo, paralelizando las operaciones posibles y simplificando otras.

Bibliografía

- [1] R. Q. Quiroga, «Spike sorting,» *Scholarpedia*, vol. 2, n.º 12, pág. 3583, 2007, revision #137442. DOI: 10.4249/scholarpedia.3583.
- [2] C. Rueda, Y. Larriba y S. D. Peddada, «Frequency Modulated Möbius Model Accurately Predicts Rhythmic Signals in Biological and Physical Sciences,» *Scientific Reports*, vol. 9, n.º 1, pág. 18701, dic. de 2019, ISSN: 2045-2322. DOI: 10.1038/s41598-019-54569-1. dirección: <https://doi.org/10.1038/s41598-019-54569-1>.
- [3] *The FMM Project*, <https://www.eio.uva.es/the-fmm-project/>, [Accessed 11-06-2025].
- [4] C. Rueda y A. Rodríguez-Collado, «Functional clustering of neuronal signals with FMM mixture models,» *Heliyon*, vol. 9, n.º 10, oct. de 2023, ISSN: 2405-8440. DOI: 10.1016/j.heliyon.2023.e20639. dirección: <https://doi.org/10.1016/j.heliyon.2023.e20639>.
- [5] *GitHub - FMMGroupVa/FMM-Extensions: Code to perform FMM multichannel model and parameter inferences — github.com*, <https://github.com/FMMGroupVa/FMM-Extensions>, [Accessed 28-05-2025].
- [6] R. D. Andreatta, «Basic Structure and Function of Neurons,» *Neuroscience Fundamentals for Communication Sciences and Disorders*, pág. 13, 2022.
- [7] *Qué es la neurona. Diccionario médico. Clínica U. Navarra — cun.es*, <https://www.cun.es/diccionario-medico/terminos/neurona>, [Accessed 29-04-2025].
- [8] J. M. Berg, L. Stryer y J. L. Tymoczko, *Bioquímica*. Reverté, 2007.
- [9] *Potencial de acción — kenhub.com*, <https://www.kenhub.com/es/library/fisiologia/potencial-de-accion>, [Accessed 30-04-2025].
- [10] H. Kang e Y. Nam, «In Vitro Neural Recording by Microelectrode Arrays,» en *Stretchable Bioelectronics for Medical Devices and Systems*, J. A. Rogers, R. Ghaffari y D.-H. Kim, eds. Cham: Springer International Publishing, 2016, págs. 275-291, ISBN: 978-3-319-28694-5. DOI: 10.1007/978-3-319-28694-5_14. dirección: https://doi.org/10.1007/978-3-319-28694-5_14.
- [11] J. Boussard, C. Windolf, C. Hurwitz et al., «DARTsort: A modular drift tracking spike sorter for high-density multi-electrode probes,» *bioRxiv*, 2023. DOI: 10.1101/2023.08.11.553023. eprint: <https://www.biorxiv.org/content/early/2023/08/14/2023.08.11.553023.full.pdf>. dirección: <https://www.biorxiv.org/content/early/2023/08/14/2023.08.11.553023>.

- [12] H. G. Rey, C. Pedreira y R. Quian Quiroga, «Past, present and future of spike sorting techniques,» *Brain Research Bulletin*, vol. 119, págs. 106-117, 2015, Advances in electrophysiological data analysis, ISSN: 0361-9230. DOI: <https://doi.org/10.1016/j.brainresbull.2015.04.007>. dirección: <https://www.sciencedirect.com/science/article/pii/S0361923015000684>.
- [13] H. Azami y S. Sanei, «Three Novel Spike Detection Approaches for Noisy Neuronal Data,» oct. de 2012. DOI: 10.1109/ICCKE.2012.6395350.
- [14] W.-J. Hwang, S.-H. Wang e Y.-T. Hsu, «Spike Detection Based on Normalized Correlation with Automatic Template Generation,» *Sensors*, vol. 14, n.º 6, págs. 11 049-11 069, 2014, ISSN: 1424-8220. DOI: 10.3390/s140611049. dirección: <https://www.mdpi.com/1424-8220/14/6/11049>.
- [15] «Análisis de componentes principales - Wikipedia, la enciclopedia libre.» (), dirección: https://es.wikipedia.org/wiki/An%C3%A1lisis_de_componentes_principales (visitado 27-05-2025).
- [16] B. C. Souza, V. Lopes-dos-Santos, J. Bacelo y A. B. L. Tort, «Spike sorting with Gaussian mixture models,» *Scientific Reports*, vol. 9, n.º 1, pág. 3627, mar. de 2019, ISSN: 2045-2322. DOI: 10.1038/s41598-019-39986-6. dirección: <https://doi.org/10.1038/s41598-019-39986-6>.
- [17] M. Pachitariu, S. Sridhar, J. Pennington y C. Stringer, «Spike sorting with Kilosort4,» *Nature Methods*, vol. 21, n.º 5, págs. 914-921, mayo de 2024, ISSN: 1548-7105. DOI: 10.1038/s41592-024-02232-7. dirección: <https://doi.org/10.1038/s41592-024-02232-7>.
- [18] C. Rueda, I. Fernández, Y. Larriba y A. Rodríguez-Collado, «The FMM Approach to Analyze Biomedical Signals: Theory, Software, Applications and Future,» *Mathematics*, vol. 9, n.º 10, 2021, ISSN: 2227-7390. DOI: 10.3390/math9101145. dirección: <https://www.mdpi.com/2227-7390/9/10/1145>.
- [19] C. Rueda, A. Rodríguez-Collado, I. Fernández, C. Canedo, M. D. Ugarte e Y. Larriba, «A unique cardiac electrocardiographic 3D model. Toward interpretable AI diagnosis,» *iScience*, vol. 25, n.º 12, pág. 105 617, 2022, ISSN: 2589-0042. DOI: <https://doi.org/10.1016/j.isci.2022.105617>. dirección: <https://www.sciencedirect.com/science/article/pii/S2589004222018892>.
- [20] C. B. Do y S. Batzoglou, «What is the expectation maximization algorithm?» *Nature Biotechnology*, vol. 26, n.º 8, págs. 897-899, ago. de 2008, ISSN: 1546-1696. DOI: 10.1038/nbt1406. dirección: <https://doi.org/10.1038/nbt1406>.
- [21] H. Zeng e Y.-M. Cheung, «A new feature selection method for Gaussian mixture clustering,» *Pattern Recognition*, vol. 42, n.º 2, págs. 243-250, 2009, Learning Semantics from Multimedia Content, ISSN: 0031-3203. DOI: <https://doi.org/10.1016/j.patcog.2008.05.030>. dirección: <https://www.sciencedirect.com/science/article/pii/S0031320308002148>.
- [22] Y. Fu, X. Liu, S. Sarkar y T. Wu, «Gaussian mixture model with feature selection: An embedded approach,» *Computers & Industrial Engineering*, vol. 152, pág. 107 000, 2021, ISSN: 0360-8352. DOI: <https://doi.org/10.1016/j.cie.2020.107000>. dirección: <https://www.sciencedirect.com/science/article/pii/S0360835220306707>.

- [23] W. Pan y X. Shen, «Penalized model-based clustering with application to variable selection.,» *Journal of machine learning research*, vol. 8, n.º 5, 2007.
- [24] A. Buccino y G. Einevoll, «MEArc: A Fast and Customizable Testbench Simulator for Ground-truth Extracellular Spiking Activity,» *Neuroinformatics*, vol. 19, ene. de 2021. DOI: 10.1007/s12021-020-09467-7.
- [25] *MEArc: a fast and customizable testbench simulator for ground-truth extracellular spiking activity*. dirección: <https://zenodo.org/records/3696926>.
- [26] *FMM: Rhythmic Patterns Modeling by FMM Models — CRAN.R-project.org*, <https://CRAN.R-project.org/package=FMM>, [Accessed 28-05-2025].
- [27] Z. Mohammadi, D. J. Denman, A. Klug y T. C. Lei, «A fully automatic multichannel neural spike sorting algorithm with spike reduction and positional feature,» *Journal of Neural Engineering*, vol. 21, n.º 4, pág. 046 039, ago. de 2024. DOI: 10.1088/1741-2552/ad647d. dirección: <https://dx.doi.org/10.1088/1741-2552/ad647d>.