

Universidad Autónoma de Querétaro
Facultad de Ingeniería
Maestría en Ciencias en Inteligencia Artificial

Algoritmos de clasificación de movimientos de señales mioeléctricas basados en técnicas Kernel

Opción de titulación

Tesis

Que como parte de los requisitos para obtener el Grado de
Maestro en Ciencias en Inteligencia Artificial

Presenta:

Jaime Fernández Girón

Dirigido por:

Dr. Marco Antonio Aceves Fernández

Dr. Marco Antonio Aceves Fernández

Presidente

Firma

Dr. Jesús Carlos Pedraza Ortega

Secretario

Firma

Dr. Saúl Tovar Arriaga

Vocal

Firma

Dr. Juan Manuel Ramos Arreguín

Suplente

Firma

Dr. Efrén Gorrostieta Hurtado

Suplente

Firma

Centro Universitario, Querétaro, Qro.
Diciembre 2020

RESUMEN

Hoy en día, la necesidad de incorporar la tecnología para facilitar y automatizar procesos ha dirigido las líneas de investigación a explorar los campos de la inteligencia artificial, así como mejorar la efectividad del uso de las señales mioeléctricas procedentes de una mano al ejecutar distintos movimientos. La incorporación de la inteligencia artificial en la clasificación de las señales mioeléctricas ha conseguido diversas aplicaciones en diferentes áreas, no solo en el ramo médico sino también lo es en el área de rehabilitación, medicina de deporte y bioprótesis, si no que el alcance de su aplicación podría bien ser de entretenimiento, como lo puede ser en el control de drones.

Los movimientos de una mano humana son considerablemente difíciles de clasificar debido a su estrecha similitud que existe entre cada uno de los diferentes movimientos que puede realizar, de todos los movimientos se seleccionaron 10 de los más importantes, que van desde la apertura y cierre del puño hasta movimientos finos como la pinza entre el dedo índice y el pulgar.

En esta investigación, la combinación de un método de inteligencia artificial basado en SVM (Support Vector Machine) y el uso de tres Kernel distintos, ha ampliado el panorama comparando el desempeño de los distintos Kernel, con la naturaleza de las señales mioeléctricas.

SUMMARY

Nowadays, the need to incorporate technology to facilitate and automate processes has directed research lines to explore the fields of artificial intelligence, as well as to improve the effectiveness of the use of myoelectric signals coming from a hand when executing different movements. The incorporation of artificial intelligence in the classification of myoelectric signals has achieved various applications in different areas, not only in the medical field but also in the area of rehabilitation, sports medicine and bioprosthesis, but the scope of its application could well be entertainment, as it can be in the control of drones.

The movements of a human hand are considerably difficult to classify due to the close similarity that exists between each of the different movements it can perform. Of all the movements, 10 of the most important ones were selected, ranging from the opening and closing of the fist to fine movements such as the grip between the index finger and thumb.

In this research, the combination of an artificial intelligence method based on SVM (Support Vector Machine) and the use of three different Kernels has broadened the picture by comparing the performance of the different Kernels with the nature of the myoelectrical signals.

Only knowledge that is used sticks in your mind.

Dale Carnegie

A mi familia: madre y padre, por acompañarme hasta el final.
Una dedicatoria especial al Dr. Marco Antonio Aceves Fernández por inspirarme y
guiarme por el camino de la investigación.

AGRADECIMIENTOS

Quiero manifestar mi más sincero agradecimiento a:

La Universidad Autónoma de Querétaro y al CONACYT por permitirme la oportunidad de terminar mis estudios de Maestría.

Mi director de tesis Marco Antonio Aceves Fernández, mis maestros Jesús Carlos Pedraza, Juan Manuel Ramos Arreguín, Saúl Tovar Arriaga, Arturo Gutiérrez González, Emilio Vargas Soto y Efrén Gorrostieta Hurtado, por su guía, paciencia y no negarme nunca un consejo.

Estefania Ruiz Guzmán, por alentarme, apoyarme y estar ahí a mi lado sin soltarme aun cuando los días se tornaban difíciles, siendo un fuerte pilar hasta el culmino de esta contribución.

Christian Murillo Ramírez por apoyarme desde el inicio hasta el final de esta etapa.

A mis compañeros de generación, por su apoyo cuando las cosas no iban bien para uno o para todos.

"No puedes llegar a ningún lado sentado en la banca"

Norman Schwarzkopf

Índice general

Índice	V
Índice de tablas	VII
Índice de figuras	VIII
1.. <i>Introducción</i>	1
1.1. Planteamiento del problema	1
1.2. Justificación	2
1.3. Antecedentes	3
1.4. Hipótesis	4
1.5. Objetivos	5
1.5.1. Objetivo general	5
1.5.2. Objetivos específicos	5
2.. <i>Marco teórico</i>	6
2.1. Inteligencia Artificial	6
2.2. Procesamiento digital	7
2.3. Bioelectricidad	8
2.4. Máquinas de Soporte de Vectores	13
3.. <i>Materiales y métodos</i>	17
3.1. Materiales	17
3.1.1. Base de datos	17
3.1.2. Software y Hardware	17
3.2. Métodos	17

3.2.1.	Normalización Min-Max	21
3.2.2.	Normalización Z-Score	22
3.2.3.	Normalización Escala Decimal	23
3.2.4.	Construcción del modelo	25
4.	<i>Resultados</i>	30
4.1.	Kernel Lineal	30
4.2.	Kernel Polinomial	42
4.2.1.	Kernel polinomial de orden 2	42
4.2.2.	Kernel polinomial de orden 3	52
4.2.3.	Kernel polinomial de orden 4	61
4.2.4.	Kernel polinomial de orden 5	70
4.2.5.	Kernel polinomial de orden 6	79
4.2.6.	Kernel polinomial de orden 7	88
4.3.	Kernel Gaussiano	99
5.	<i>Discusión de resultados y conclusiones</i>	108
	ANEXOS	115
5.1.	Artículo publicado	115
5.2.	Constancia de Lengua Extranjera	115

Índice de cuadros

1.1. Tabla de antecedentes.	4
4.1. Tabla con la exactitud y la varianza de los movimientos de la mano derecha e izquierda usando el Kernel lineal	41
4.2. Tabla con la exactitud y la varianza de los movimientos de la mano derecha utilizando el Kernel polinomial de orden cuatro	99
4.3. Tabla con la exactitud y la varianza de los movimientos de la mano derecha e izquierda usando el Kernel lineal	105

Dirección General de Bibliotecas UAQ

Índice de figuras

2.1.	Diagrama de origen del impulso mioeléctrico creado con BioRender.com	10
2.2.	Adquisición de señales mioeléctricas (Burbano et al., 2004)	11
2.3.	Procesamiento digital de señales (a) Señal filtrada con el pasa banda de 4Hz a 400Hz (b) Comparación entre la señal original y la filtrada (la que posee una amplitud menor se refiere a la filtrada) (Acevedo, Betancourt y Hurtado, 2013)	12
2.4.	Maquina de Soporte Vectorial (García et al., 2016)	14
2.5.	Ejemplos de hiperplanos separables en dos clases: (a) hiperplano con una sola separacion (b) hiperplano con varias separaciones. (Suárez, 2014)	15
2.6.	Hiperplano linealmente no separable (a) hiperplano en dimencion R^2 (b) hiperplano en dimencion R^3 (creación propia).	16
3.1.	Diagrama de los movimientos, canales y eventos generados por los 50 sujetos.	18
3.2.	Ejemplo de un movimiento realizado por el sujeto de prueba. La figura muestra el tiempo, la amplitud de la señal y cada uno de los canales almacenados para la extensión del movimiento de la muñeca (mano derecha)(creacion propia, basado en (Aceves-Fernandez et al., 2019)).	19
3.3.	. La figura muestra las señales mioeléctricas filtradas por la Transformada Wavelet (mano derecha) (Ramírez-Angeles, 2019).	19
3.4.	Muestra un ejemplo de señal EMG original antes de ser normalizada.	20
3.5.	Muestra la señal original normalizada por método de Min-Max.	21

3.6. Muestra la señal original normalizada por método de Z-Score.	22
3.7. Muestra la señal original normalizada por método de escala decimal. .	23
3.8. Muestra una comparativa en un grafico circular: (a) señal original (b) normalización por Min-Max (c) normalización por Z-Score (d) normalización por escala decimal (Creación propia).	24
3.9. Diagrama de la metodología propuesta.	25
3.10. Ejemplo de la selección aleatoria de los sujetos.	27
4.1. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	31
4.2. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	32
4.3. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	33
4.4. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	34
4.5. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	35
4.6. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	36
4.7. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	37

4.8. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	38
4.9. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	39
4.10. Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.	40
4.11. Maquina de Soporte Vectorial (García et al., 2016)	42
4.12. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	44
4.13. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	45
4.14. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	46
4.15. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	47

4.16. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	48
4.17. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	49
4.18. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	50
4.19. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.	51
4.20. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	53
4.21. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	54
4.22. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	55

4.23. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	56
4.24. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	57
4.25. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	58
4.26. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	59
4.27. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.	60
4.28. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	62
4.29. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	63

4.30. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	64
4.31. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	65
4.32. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	66
4.33. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	67
4.34. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	68
4.35. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.	69
4.36. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	71

4.37. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	72
4.38. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	73
4.39. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	74
4.40. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	75
4.41. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	76
4.42. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	77
4.43. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.	78

4.44. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	80
4.45. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	81
4.46. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	82
4.47. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	83
4.48. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	84
4.49. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	85
4.50. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	86

4.51. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.	87
4.52. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	89
4.53. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	90
4.54. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	91
4.55. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	92
4.56. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	93
4.57. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) Señaseñall de dos sujetos clasificados por Kernel polinomial de orden 7.	94

4.58. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	95
4.59. Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.	96
4.60. Comparación de la precisión del Kernel polinomial del orden 2D al orden 7D. La figura muestra la comparación en el comportamiento del orden 2D precisión del Kernel con el orden 7D (mano derecha). . .	97
4.61. Comparación de la precisión del Kernel polinomial del orden 2D al orden 7D. La figura muestra la comparación en el comportamiento del orden 2D precisión del Kernel con el orden 7D (mano izquierda). . .	98
4.62. Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.	100
4.63. Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.	101
4.64. Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.	102
4.65. Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.	103

4.66. Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano. 104

4.67. Maquina de Soporte Vectorial (García et al., 2016) 106

4.68. Comparación entre la exactitud de cada uno de los Kernel utilizados: (a) mano derecha. (b) mano izquierda. 107

Dirección General de Bibliotecas UAQ

1. INTRODUCCIÓN

1.1. *Planteamiento del problema*

Una electromiografía (EMG), es un conjunto de ondas senoidales con frecuencia de 2Hz a 500Hz y amplitud de $50\mu\text{V}$ a 5mV , las cuales dependen directamente de la acción del musculo al momento de hacer la adquisición de las muestras, durante la adquisición de las muestras es necesario tener en cuenta la fuerza, velocidad y fatiga del músculo y así evitar inconsistencias en las muestras (Joachims, 1998).

Las señales adquiridas de una electromiografía (EMG) suelen estar llenas de información y datos presentes durante la adquisición de las señales como lo son los ruidos parásitos y las frecuencias montadas de la misma red eléctrica, por lo cual llevan un proceso de filtrado, el cual se realiza de manera analógica por medio de dispositivos electrónicos (Martínez-Miguel et al., 2017).

Las señales EMG no cuentan con un patrón visible o fácilmente identificable, lo que dificulta distinguir el movimiento que se está realizando de otros movimientos, es necesaria una clasificación robusta que permita una mayor definición de la onda para su aplicación (Nougarou, Campeau, Islam, Massicotte y Gosselin, 2018).

Los avances en el procesamiento de señales ha permitido el diseño de algoritmos capaces de sintetizar y/o discriminar grandes cantidades de datos facilitando la interpretación de los mismos (Joachims, 1998). Entre estos algoritmos encontramos la Maquina de Vectores de Soporte, por sus siglas en inglés (SVM), constituye un

hiperplano óptimo en forma de superficie que mapea los puntos de entrada de modo que el margen de separación entre dos clases de datos se amplía al máximo, siendo empleado como una clasificación binaria o regresión (Betancourt, 2005).

Por lo antes expuesto, es necesario analizar en qué medida la implementación de los SVM en el procesamiento de las señales EMG, incide en la clasificación robusta de las señales producidas por los movimientos de una mano humana.

1.2. *Justificación*

El procesamiento de señales es una línea de investigación relativamente nueva, desde sus primeras implementaciones ha teniendo un enorme auge entre la comunidad científica gracias a la gran demanda de aplicaciones que involucran el procesamiento de señales.

La señal que se obtiene de un estudio de EMG, es un conjunto de pulsos eléctricos generados en el musculo durante la realización de movimientos voluntarios genera una señal aperiódica que contiene demasiada información y mucha de ella son artefactos de ruido, por lo que complica la aplicación de la señal.

Anteriormente, otros autores ya han tratado la señal EMG eliminando los artefactos de ruido, sin embargo, no se ha conseguido limpiar la señal afinándola en una sola señal limpia que conserve las principales características que implican al movimiento del músculo.

El procesamiento por métodos de inteligencia artificial basadas en técnicas Kernel impacta directamente en la linealización de una onda con parámetros dispersos a una señal limpia que contenga la menor cantidad posible de elementos dispersos. Y a su vez abre paso a una buena interpretación de la información por medio de un

sistema de prótesis de mano, esto al obtener señales finas y precisas de los principales movimientos de una mano humana. Será perfectamente aplicable a los movimientos de una prótesis de mano humana y como una herramienta adicional de rehabilitación.

1.3. Antecedentes

Desde el descubrimiento del procesamiento de señales, el campo de la inteligencia artificial ha ido mejorando las aplicaciones de estas técnicas tal y como lo hizo Sridhar P Arjunan con la máquina de vectores de doble soporte (TSVM del inglés Twin Support Vector Machines) con la que obtuvo una exactitud de 82 % a 95 % en 7 sujetos de estudio (Arjunan, Kumar y Naik, 2010).

Es posible implementar varias técnicas para un mismo fin, ya sean de distinto tipo o del mismo como lo hizo Bassam Moslem en el 2011 hizo una combinación de máquinas de vectores de soporte múltiple con la que consiguió una exactitud de 87 % utilizando 50 señales mioléctricas (Moslem, Khalil, Diab, Chkeir y Marque, 2011a). En ese mismo año Nayan M. Kakoty hizo un reconocimiento de dos tipos de agarre a través de los componentes principales de EMG basadas en la transformada discreta de Wavelet (DWT de el inglés Discrete Wavelet Transform) con lo que consiguió una exactitud del 97.5 % con un entrenamiento de 40 sujetos y 960 señales. (Kakoty, 2011). Así como Kakoty implemento el reconocimiento en el agarre de la mano Gonzalo Pomboza hizo también un reconocimiento del gesto de la mano utilizando SVM consiguiendo una exactitud del 79.36 % con un entrenamiento de 24,866 señales (Pomboza, 2016).

Los SVM son una técnica muy amigable que puede mezclarse con otras técnicas para mejorar sus propiedades, tal es el caso de Moataz Kilany quien combino la onda de agua y SVM para mejorar la clasificación de la señal para detectar tras-

tornos neurogenicos en los movimientos de la mano consiguiendo una exactitud de 91.44 % en 24 estudios (Kilany, Houssein y Hassanien, 2017). N. Sukumar al igual que Kilany fusiono los SVM con otra técnica, en este caso con el modo de descomposición variacional obteniendo 98.17 % de exactitud en 40 sujetos de estudio con un entrenamiento de 10,000 señales (Sukumar, Taran y Bajaj, 2018).

Actualmente el avance en la investigación e implementación de algoritmos basados en inteligencia artificial ha dirigido la investigación mejorar los métodos ya utilizados, el doctor Y Zeng ha implementado y mejorado el proceso gaussiano en evolución de aprendizaje basado en la autoregresión utilizando el método Kernel con el que consiguió una exactitud del 95 % en 6 sujetos de estudio (Zeng, Yang y Yin, 2019).

Tab. 1.1: Tabla de antecedentes.

Autor	año	Técnica	Exactitud	Datos
<i>Arjunan et al</i>	2010	TSVM	82 % a 95 %	7 Sujetos, 2 Señales
<i>Moslem et al</i>	2011	SVM	87 %	50 Señales
<i>Kakoty et al</i>	2012	NMF y Kernel	95 %	9 Sujetos
<i>Pomboza et al</i>	2016	SVM	79.36 %	24,866 Señales
<i>Kilany et al</i>	2017	WWO-SVM	91.44 %	24 Estudios
<i>Sukumar et al</i>	2018	VMD	98.17 %	4 Sujetos, 10,000 Señales
<i>Zeng et al</i>	2019	Gaussiano	95 %	6 Sujetos de estudio

1.4. Hipótesis

Es posible aumentar la exactitud de la onda mioeléctrica utilizando métodos de inteligencia artificial basados en técnicas Kernel para la correcta clasificación de señales

EMG a partir de los principales movimientos de una mano humana.

1.5. *Objetivos*

1.5.1. *Objetivo general*

Procesar señales mioeléctricas de tal manera que se permita una clasificación mediante las técnicas Kernel realizando una clasificación con una mayor exactitud con respecto a otros métodos, a partir de los movimientos de una mano humana.

1.5.2. *Objetivos específicos*

- Implementar algoritmos basados en inteligencia artificial que permitan el procesamiento y conjunción de varias señales mioeléctricas provenientes de una mano humana.
- Definir la técnica Kernel a utilizar para la clasificación de los movimientos.
- Implementar algoritmos SVM en la clasificación de las señales EMG de la mano humana.

2. MARCO TEÓRICO

2.1. *Inteligencia Artificial*

La inteligencia artificial hoy en día es la búsqueda constante comprendida por el ser humano para crear seres artificiales capaces de realizar tareas inteligentes, ya sea a su imagen y semejanza o simulando el comportamiento de otras formas de vida. Es por esto que la inteligencia artificial está rodeada de misticismo, teorías, creencias, promesas y esperanzas de que, con el apoyo de la ciencia y la tecnología, la humanidad podrá lograr un nuevo génesis de clones y autómatas artificiales (Banda, 2014).

Se considera que el origen de la IA se remonta a los intentos del hombre desde la antigüedad por incrementar sus potencialidades físicas e intelectuales, creando artefactos con automatismos y simulando la forma y las habilidades de los seres humanos (Romero et al., 2007).

La contribución del matemático inglés Alan Turing al mundo de la computación en general, y al de la IA en particular, fue muy considerable. Turing, participó en el diseño de uno de los primeros computadores que existieron, desarrollado para el ejército inglés, entre los años 40 y 50. Además de aportar los conceptos básicos de la arquitectura secuencial de los computadores (Turing, 2009).

2.2. *Procesamiento digital*

El procesamiento digital de señales tiene sus inicios en la década de los 50, cuando un grupo de ingenieros querían hacer la simulación de sus filtros analógicos, y así reducir los altos costos que implica hacer la construcción de un filtro analógico. La herramienta para hacer la simulación, era por supuesto el ordenador (Betancourt, 2005).

El procesamiento digital de las señales dependía directamente del procesador y de las matemáticas principalmente, así como de los algoritmos diseñados en los años 50. Una vez que los algoritmos fueron tomando forma los diseñadores del DSP (Digital Signal Processing) trabajaron en la estructura de los algoritmos para conseguir la meta de hacer procesamiento en tiempo real. (Betancourt, 2005)

Las señales pueden ser de una gran variedad de fenómenos físicos, como lo es la medición de la velocidad, el cambio de la temperatura o el tránsito de la sangre en el torrente sanguíneo (Maglogiannis, Karpouzis y Wallace, 2005).

Las señales se representan por medio de funciones matemáticas con una o varias variables independientes que determinan el comportamiento de las líneas graficadas. Estas señales se clasifican en discretas y continuas, siendo señales discretas las que son fácilmente agrupables en categorías, mientras que "las señales continuas son difíciles de agrupar y podrían ser cualquier numero entero que exista en el eje del tiempo" (Semmlow, 2008). De igual manera se dividen en 2 grandes tipos; las señales analógicas, las cuales se encuentran en el eje de las y continuo mientras las señales digitales corresponden al eje de las y discreto, generalmente estas últimas se ven representadas de manera binaria (Asefi, Moghimi y Kalani, 2016). A su vez las señales se dividen en periódicas y aperiódicas, donde las periódicas son las que repiten un patrón en la muestra y las aperiódicas no son tan fáciles de predecir ya que

no se repite el mismo patrón. También las podemos clasificar como determinísticas o aleatorias, en donde las determinísticas es posible que podamos predecir el comportamiento de la señal por medio de una función matemática mientras que la señal aleatoria no es fácil predecir su comportamiento futuro (Oppenheim, Willsky y Nawab, 1998).

2.3. Bioelectricidad

El estudio de células excitables y procesos bioeléctricos en el cuerpo humano ha generado un significativo impacto en el desarrollo de métodos clínicos tales como la Electroneurografía (ENG), Electromiografía (EMG), Electrocardiografía (ECG) y Electroencefalografía (EEG) en donde todos implican registrar e interpretar señales bioeléctricas que surgen de la propagación de potenciales de acción (Bronzino, 2015). La bioelectricidad tiene su origen en las diferencias de voltaje presentes entre el interior y el exterior de las células en donde el flujo de electricidad a lo largo de un nervio o dentro de un músculo ocurre pasivamente a través del medio conductor dentro de la célula (Bronzino, 2015). Matemáticamente, la corriente intracelular en una célula nerviosa unidimensional puede ser dado como se muestra en la ecuación 2.1 (Plonsey, 1988).

$$I_i = -\frac{1}{r_i} \frac{\partial \phi_i}{\partial z} \quad (2.1)$$

Donde ϕ , es el potencial intracelular, z es la coordenada axial, r_i es la resistencia intracelular longitudinal por unidad, y I_i es la corriente axial longitudinal por unidad de área de selección transversal.

Las neuronas mantienen una diferencia de potencial más o menos constante llamada potencial de reposo (-70 mV), siendo el interior de la célula negativo con respecto al exterior. El potencial de reposo se debe a las proteínas cargadas negativamente que no pueden atravesar la membrana y la distribución asimétrica de iones Sodio (Na) y Potasio (K) que se mantiene por bombas iónicas, transportadores e intercambiadores como se ve en la figura 2.1 (Larraza, 2006).

La membrana tiene poros los cuales constituyen la abertura por donde los iones atraviesan la membrana y sus propiedades proporcionan selectividad a ciertas especies iónicas, son los clasificados en canales de Na y de K . Los canales iónicos sufren cambios que provocan la apertura y el cierre del poro del canal en cuestión de milisegundos (estos cambios se llaman coloquialmente “gating”). La apertura de los canales está acoplada a mecanismos que detectan señales celulares como cambios en el voltaje, señales químicas (un neurotransmisor o ATP). Cuando los canales se abren, el movimiento de iones por su través genera las señales eléctricas de las neuronas y otras células excitables, como las células musculares (Larraza, 2006).

El potencial de acción se inicia con la activación de los canales de sodio activados por voltaje. La entrada de sodio al interior celular a favor de su gradiente electroquímico provoca una ligera despolarización que produce la apertura de más canales de sodio, lo que genera una mayor y más rápida entrada de sodio, que produce una mayor despolarización, y así sucesivamente. Se trata de un fenómeno de retroalimentación positiva. Así se genera la fase ascendente del potencial de acción. La salida de los iones de potasio hiperpolariza la célula llevándola al potencial de reposo. La repolarización provoca el cierre de los canales de potasio y permite a los canales de sodio inactivados pasar al estado original cerrado pero activable, listos para el siguiente potencial de acción. (Larraza, 2006).

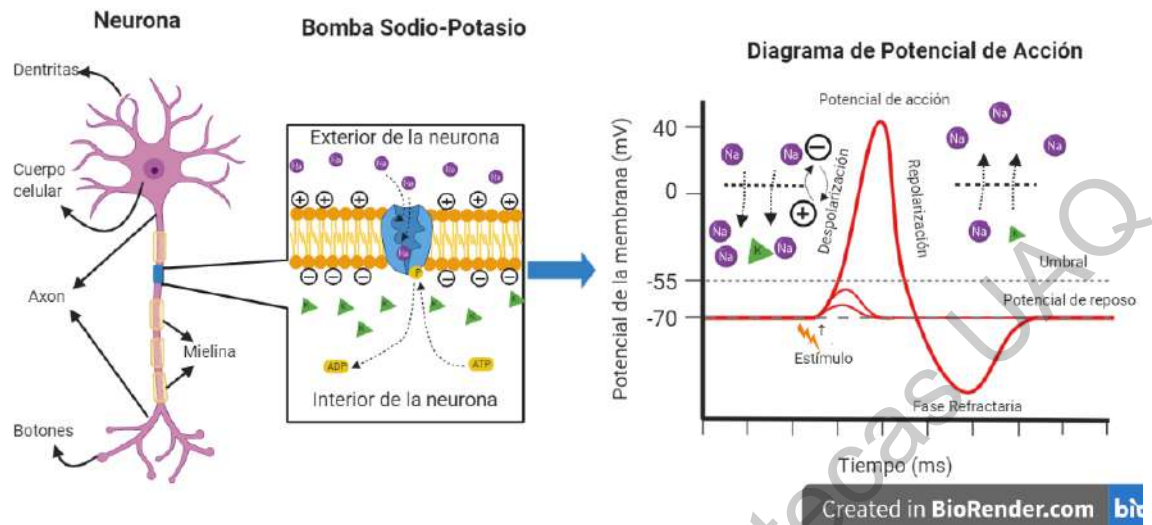


Fig. 2.1: Diagrama de origen del impulso mioeléctrico creado con BioRender.com

La señal analógica generada a partir del movimiento de los músculos es extraída a través de electrodos como se ve en la figura 2.2, además de un sistema de amplificación basado en filtros analógicos capaces de filtrar y amplificar la señal que va de los 50uV – 5mV con una frecuencia de 2 – 500Hz. Es necesario este primer tratamiento de la señal ya que con valores tan pequeños resulta casi imposible hacer un correcto procesamiento de la señal (Mesin et al., 2009). Una vez que la señal mioeléctrica es amplificada y filtrada, es posible hacer un tratamiento digital, lo que involucra el uso del procesamiento digital de señales como se aprecia en la figura 2.3 (Bronzino, 2015).

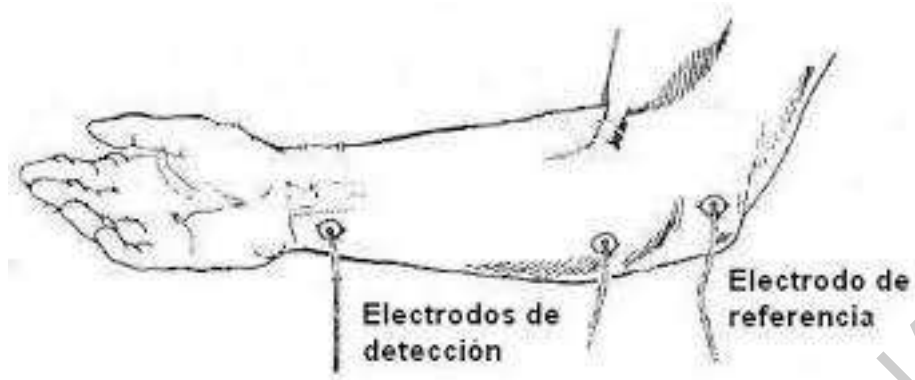
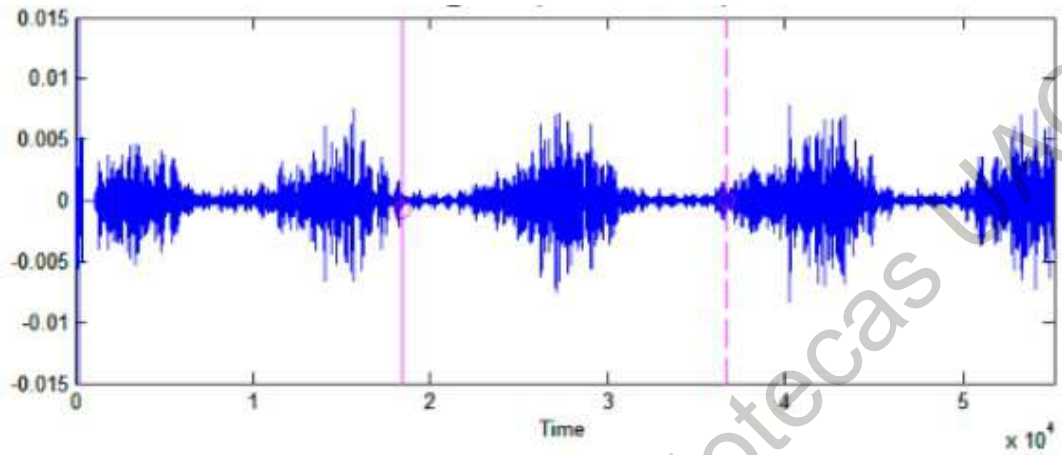
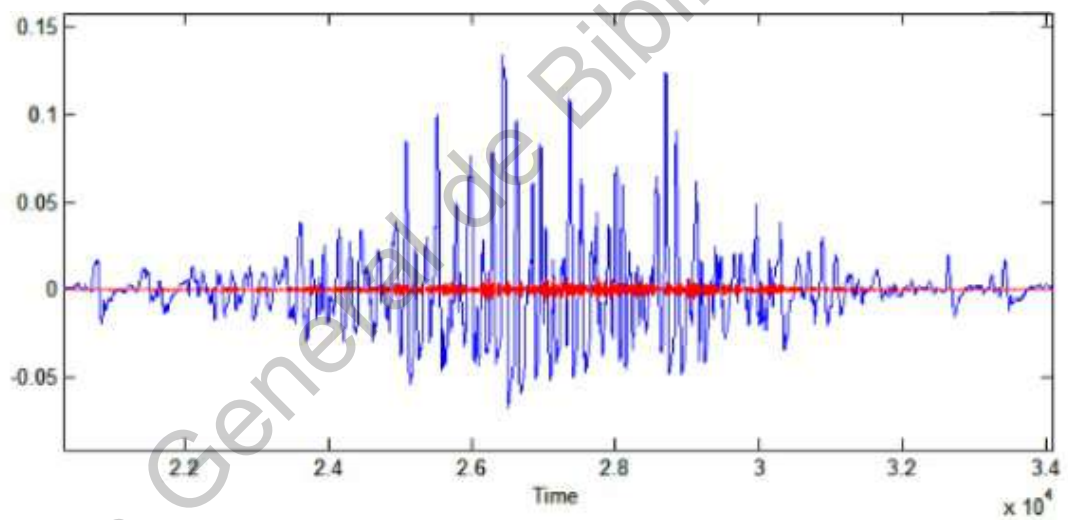


Fig. 2.2: Adquisición de señales mioeléctricas (Burbano et al., 2004)

La mano humana puede ejecutar tantos movimientos como lo permitan sus ejes y articulaciones. Es por ello que la clasificación de los movimientos de una mano mediante el procesamiento de las señales digitales, ha sido hoy en día un tema de interés para un gran sector de la población, que busca hacer una clasificación correcta de los movimientos de la mano y aplicarlo en la creación de prótesis y ortesis funcionales, en el área de rehabilitación, en medicina del deporte, domótica, realidad virtual y control de drones.



(a)



(b)

Fig. 2.3: Procesamiento digital de señales (a) Señal filtrada con el pasa banda de 4Hz a 400Hz (b) Comparación entre la señal original y la filtrada (la que posee una amplitud menor se refiere a la filtrada) (Acevedo, Betancourt y Hurtado, 2013)

2.4. Máquinas de Soporte de Vectores

Actualmente existe una gran cantidad de algoritmos capaces de hacer el tratamiento de la señal mioeléctrica, como lo puede ser el uso de técnicas de aprendizaje lineal (del inglés Linear Learning Machine) que involucra algoritmos como la clasificación lineal, la regresión lineal, discriminación multiclase, etc (Cruz and Herrera, 2010). Otra técnica aplicable al tratamiento de las señales mioeléctricas es por Kernel. También es posible aplicar técnicas como lo es la máquina de soporte de vectores (SVM por sus siglas en inglés Support Vector Machine) que deriva técnicas como la clasificación de vectores de soporte, regresión de vectores de soporte, procesos gaussianos, Kernel de regresión de cresta, clasificación de margen máximo, entre otros (Cristianini et al., 2000).

Las máquinas de soporte de vectores (SVM) fueron introducidas en la década de los 90s por Vapnik y sus colaboradores, fueron introducidas por sus trabajos de la teoría del aprendizaje automatizado, inicialmente fueron diseñadas para clasificación binaria (Vapnik, 1995).

SVM tiene aplicaciones en multitud de disciplinas, incluyendo la gestión de relaciones con los clientes (CRM), el reconocimiento facial y de otras imágenes, bioinformática, extracción de conceptos de minería de texto, detección de intrusiones, predicción de estructura de proteínas y reconocimiento de la voz (Sun, Fu, Wang, 2019).

Los SVM pertenecen a la categoría de clasificadores lineales donde su objetivo es diseñar un algoritmo computacionalmente eficiente de aprendizaje automatizado, separando hiperplanos en un espacio de características de alta dimensión, capaces de manejar tamaños de muestra del orden de las 100,000 instancias (Cristianini et al., 2000).

Son ideales para resolver problemas de clasificación y regresión, este método realiza un etiquetado de clases y entrenamiento para tener un modelo que determine a qué clase pertenece la nueva entrada. El objetivo de optimización de SVM es maximizar el margen, este se define como la distancia entre el hiperplano de la frontera de decisión y los llamados vectores de soporte (Betancourt, 2005), como se ve en la Figura 2.4.

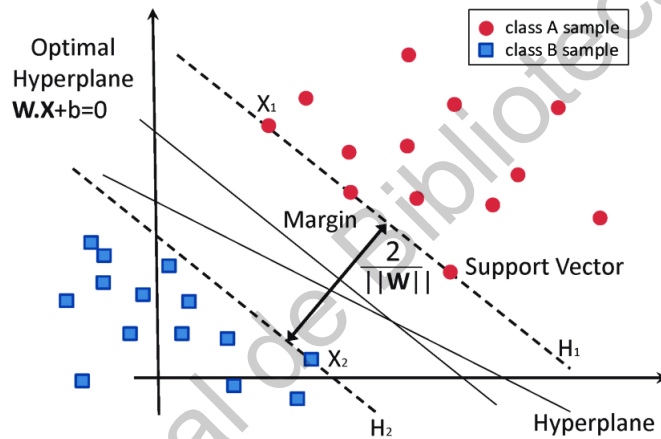


Fig. 2.4: Máquina de Soporte Vectorial (García et al., 2016)

Los SVM obtienen información principalmente de los puntos más cercanos al límite de decisión (vectores de apoyo). Se dice que tienen un margen duro cuando son perfectamente separables y tienen un margen débil cuando se permite una mezcla entre los distintos puntos.

Cada punto de entrenamiento que pertenece a alguna de las dos clases y se les ha asignado una etiqueta. Si tenemos un conjunto de puntos en un plano dado que $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$, donde $y_i \in \{+1, -1\}$ cuando $x_i \in \mathbb{R}^n$, se puede definir un hiperplano de separación como se aprecia en la figura 3.8.

$$N(x) = (w_1x_1 + \dots + w_nx_n) + b = \langle w, x \rangle + b \quad (2.2)$$

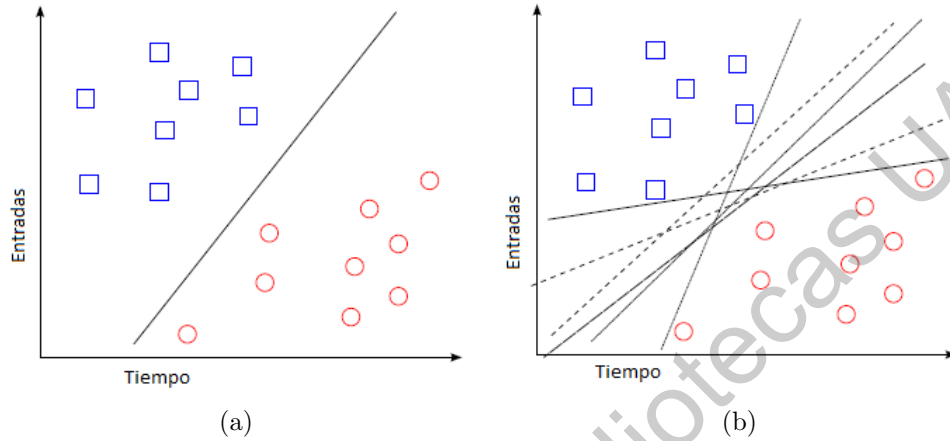


Fig. 2.5: Ejemplos de hiperplanos separables en dos clases: (a) hiperplano con una sola separacion (b) hiperplano con varias separaciones. (Suárez, 2014)

donde w y b son coeficientes reales. El hiperplano de separación cumplirá las siguientes restricciones para todo x_i del conjunto de ejemplos:

$$\begin{aligned} \langle w, x_i \rangle + b &\geq 0 & si & y_i = +1 \\ \langle w, x_i \rangle + b &\leq 0 & si & y_i = -1, i = 1, \dots, n \end{aligned} \quad (2.3)$$

Los SVM regularmente son capaces de obtener buenos resultados cuando el límite de separación entre sus clases se aproxima a la linealidad. Si no es el caso, su capacidad de separación se ve casi imposible, sin embargo, el hecho de que los grupos no sean linealmente separables no quiere decir que no lo sean. Una manera de enfrentarse a estos escenarios es expandiendo las dimensiones del espacio original como se muestra en la figura 2.6.

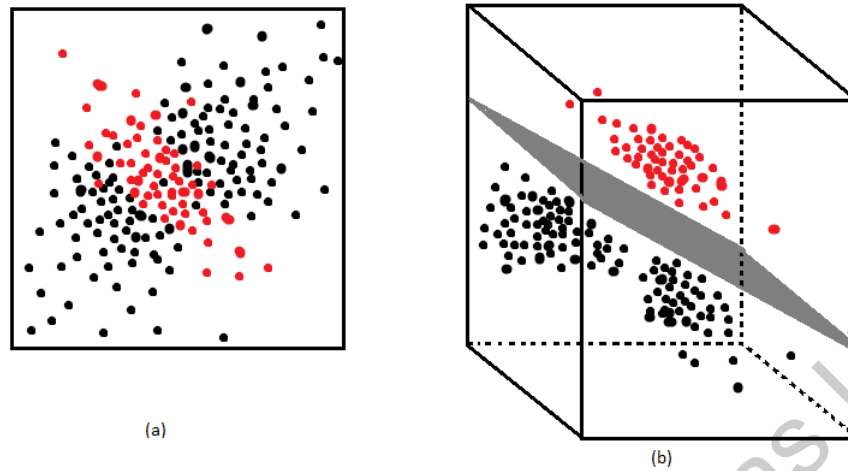


Fig. 2.6: Hiperplano linealmente no separable (a) hiperplano en dimencion R^2 (b) hiperplano en dimencion R^3 (creación propia).

Para aumentar la dimensión en el hiperplano de R^2 a R^3 se usa la siguiente función.

$$f(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) \quad (2.4)$$

3. MATERIALES Y MÉTODOS

3.1. *Materiales*

3.1.1. *Base de datos*

Para el desarrollo del presente trabajo se utilizó la base de datos con 50 sujetos (Ramírez-Ángeles and Aceves-Fernandez, 2018), utilizando un lector de señales multicanal (Ramírez-Angeles, 2019).

3.1.2. *Software y Hardware*

se utilizo el software Matlab version R2017b para el procesamiento de las señales mioeléctricas. El equipo de cómputo utilizado para el desarrollo del presente trabajo es una computadora portátil marca Dell modelo G3 con un procesador Intel Core i5-9300H a 2.4 GHz con 8 GB de memoria RAM bajo el sistema operativo Windows 10 de 64 bits.

3.2. *Métodos*

El modelo SVM se creó utilizando una base de datos de cincuenta sujetos, sin antecedentes de trastornos musculares (Ramírez-Angeles, 2019). En esta investigación, los cincuenta sujetos realizaron diez movimientos diferentes de mano tanto en la mano derecha como en la izquierda, con la ayuda de electrodos colocados en cada antebrazo (Aceves-Fernandez et al., 2019). Los datos se obtuvieron utilizando un brazalete con ocho electrodos colocados alrededor del antebrazo, con los cuales se

obtuvieron ocho canales que capturaron la actividad eléctrica de todos los músculos involucrados en el movimiento de la mano y cada movimiento fue repetido cinco veces obteniendo cinco eventos por movimiento como se muestra en la figura 3.1. La frecuencia de muestreo del hardware presentado es de 200 Hz y una duración de 16 segundos por cada canal (Ramírez-Ángeles et al., 2019). Por lo cual al momento de modelar el comportamiento del movimiento de la mano sea aun más desafiante, al tener la actividad de un solo movimiento en ocho electrodos almacenados simultáneamente como se muestra en el diagrama. 3.2, lo que también nos permite apreciar cual músculo está ejerciendo la mayor actividad durante la ejecución del movimiento, lo que genera un total de 40,000 señales y traducido en datos genera un total de 25,600,000 datos.

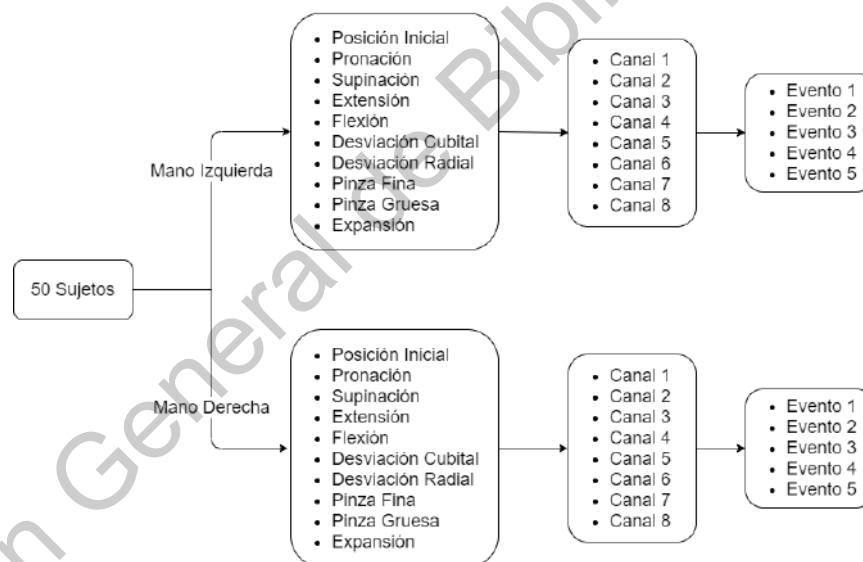


Fig. 3.1: Diagrama de los movimientos, canales y eventos generados por los 50 sujetos.

Las señales mioeléctricas tomadas de los cincuenta sujetos contienen demasiada información y mucha de ella son artefactos de ruido que alteran los resultados, por lo que se requirió de un preprocesamiento, en este caso se utilizó la Transformada Wavelet (Ramírez-Angeles, 2019).

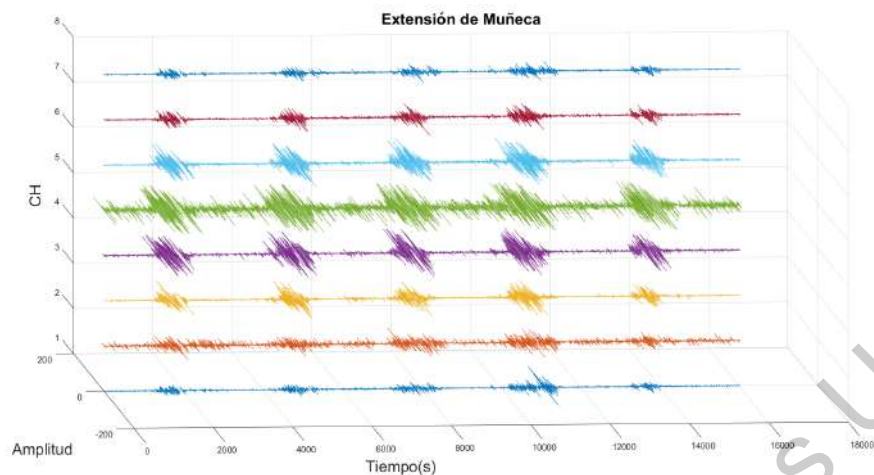


Fig. 3.2: Ejemplo de un movimiento realizado por el sujeto de prueba. La figura muestra el tiempo, la amplitud de la señal y cada uno de los canales almacenados para la extensión del movimiento de la muñeca (mano derecha)(creacion propia, basado en (Aceves-Fernandez et al., 2019)).

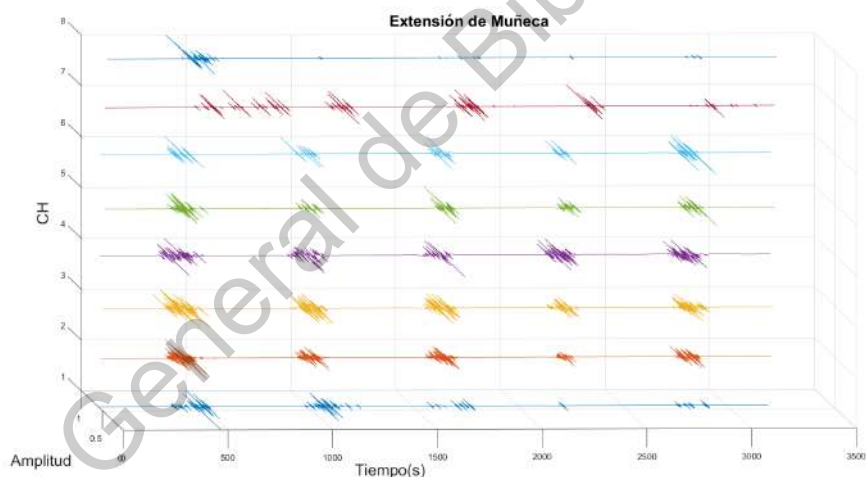


Fig. 3.3: . La figura muestra las señales mioeléctricas filtradas por la Transformada Wavelet (mano derecha) (Ramírez-Angeles, 2019).

En la figura 3.2 se muestran los ocho canales tomados de la extensión de la muñeca sin ser filtrados, mientras que en la figura 3.3, se muestran las mismas señales de la extensión de muñeca ya filtrados mediante la Transformada Wavelet, con la cual se consiguió eliminar los artefactos de ruido dejando la señal mioeléctrica lo más pura

posible.

Como otra forma de preprocesamiento para esta contribución se utilizó la normalización de los datos debido a que con el filtro por la Transformada Wavelet se filtran los datos, pero es necesaria la normalización de los datos, esto para ordenarlos y agruparlos dentro de un rango.

Para la normalización se realizaron pruebas con tres métodos diferentes, con la finalidad de identificar el método de normalización con el cual se obtengan mejores resultados, para esta contribución se utilizó normalización por Min-Max, Z-Score y escala decimal, para realizar las pruebas se utilizó un movimiento y canal de manera aleatoria, el cual se muestra en la figura 3.4 al cual se aplicaron los métodos de normalización antes mencionados.

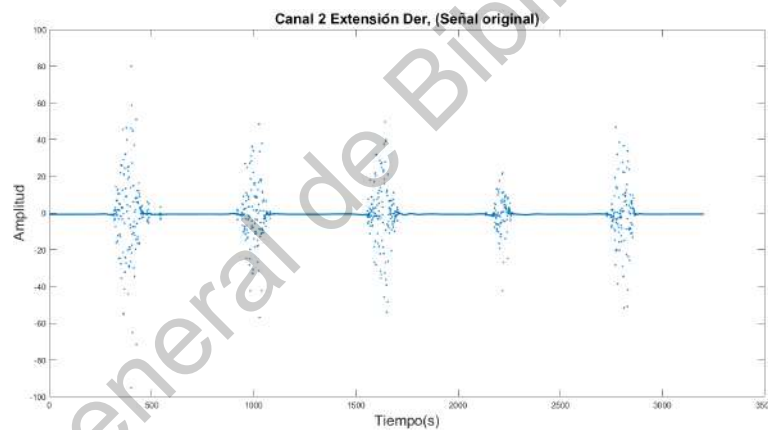


Fig. 3.4: Muestra un ejemplo de señal EMG original antes de ser normalizada.

3.2.1. Normalización Min-Max

La normalización Min-Max es una de las más conocidas maneras de normalizar los datos. El mínimo valor de los datos se transforma en 0 y el máximo valor se transforma en 1, de ahí viene su nombre, como se muestra en la figura 3.5, en donde se muestra el resultado de una comparación entre la señal original y la señal normalizada por Min-Max, y se representa por la formula:

$$X_{normalizada} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

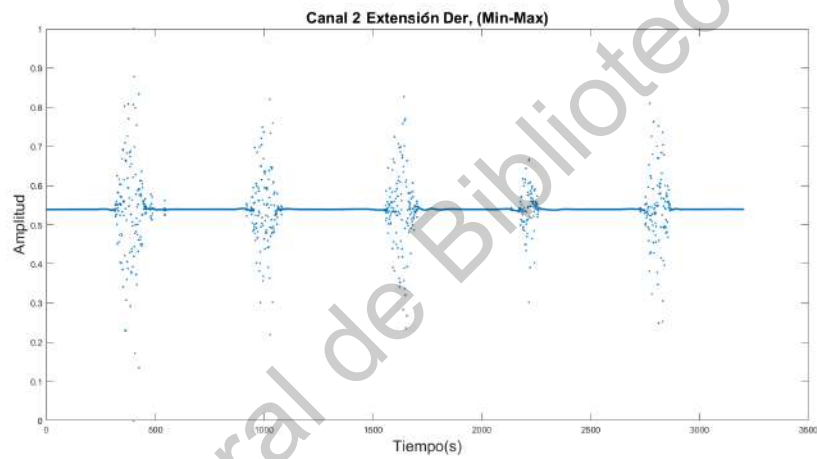


Fig. 3.5: Muestra la señal original normalizada por método de Min-Max.

3.2.2. Normalización Z-Score

La normalización Z-Score es una estrategia de normalización de datos en la que se pretende eliminar el problema de valores atípicos, como se muestra en la figura 3.9, en donde se muestra el resultado de una comparación entre la señal original y la señal normalizada por Z-Score, y se representa por la fórmula:

$$Z = \frac{X - \mu}{\sigma} \quad (3.2)$$

Donde μ es el promedio de los datos y σ es la desviación estándar. Si un dato es exactamente igual a la media de todos los datos, se normalizará a 0. Si está por debajo de la media, será un número negativo, y si está por encima de la media será un número positivo.

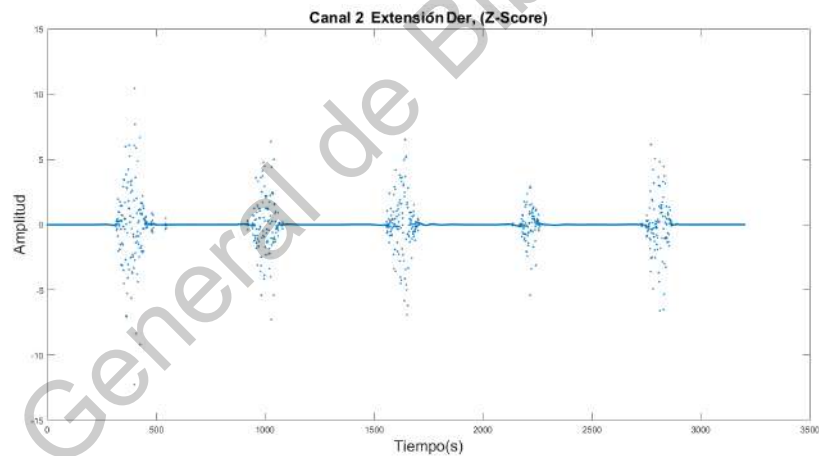


Fig. 3.6: Muestra la señal original normalizada por método de Z-Score.

3.2.3. Normalización Escala Decimal

La normalización en escala decimal se utiliza para escalar los datos de un atributo de un rango más grande a un rango más pequeño, como se muestra en la figura 3.9, en donde se muestra el resultado de una comparación entre la señal original y la señal normalizada por escala decimal y se representa por la fórmula:

$$X_{normalizada} = \frac{X}{10^j} \quad (3.3)$$

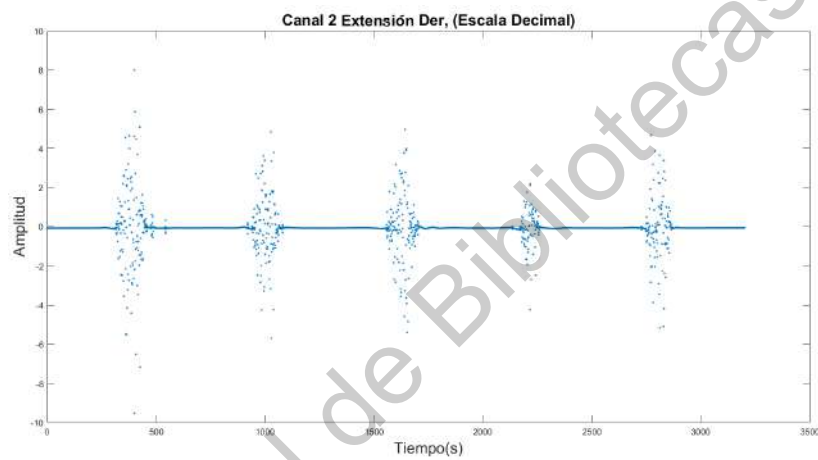


Fig. 3.7: Muestra la señal original normalizada por método de escala decimal.

Como se muestra en los gráficos anteriores de la normalización, desde un plano lineal la diferencia entre una grafica y otra no difiere en sus características ni altera el orden de lo datos, sin embargo, modifica el rango en el que se agrupan los datos, y desde un plano circular se aprecia que la normalización por Min-Max le agrega mejores características que favorecen el posterior procesamiento de la señal.

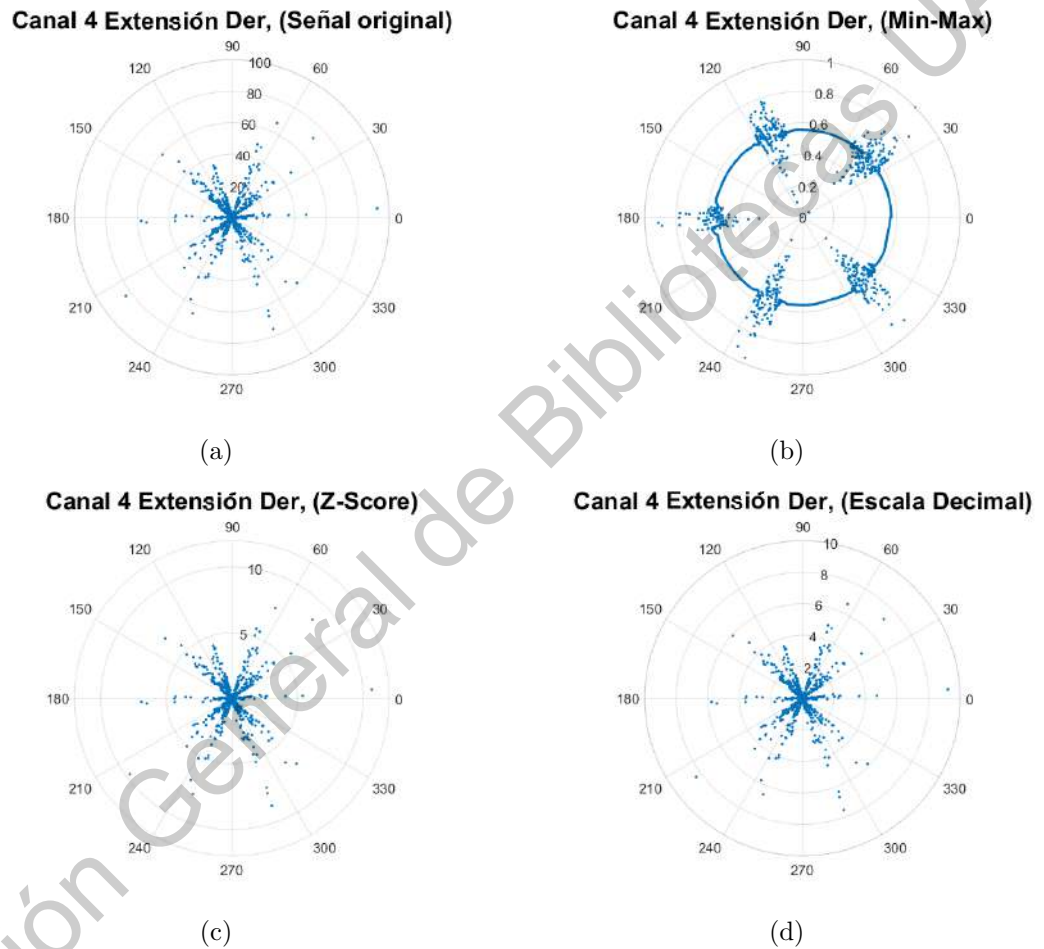


Fig. 3.8: Muestra una comparativa en un grafico circular: (a) señal original (b) normalización por Min-Max (c) normalización por Z-Score (d) normalización por escala decimal (Creación propia).

3.2.4. Construcción del modelo

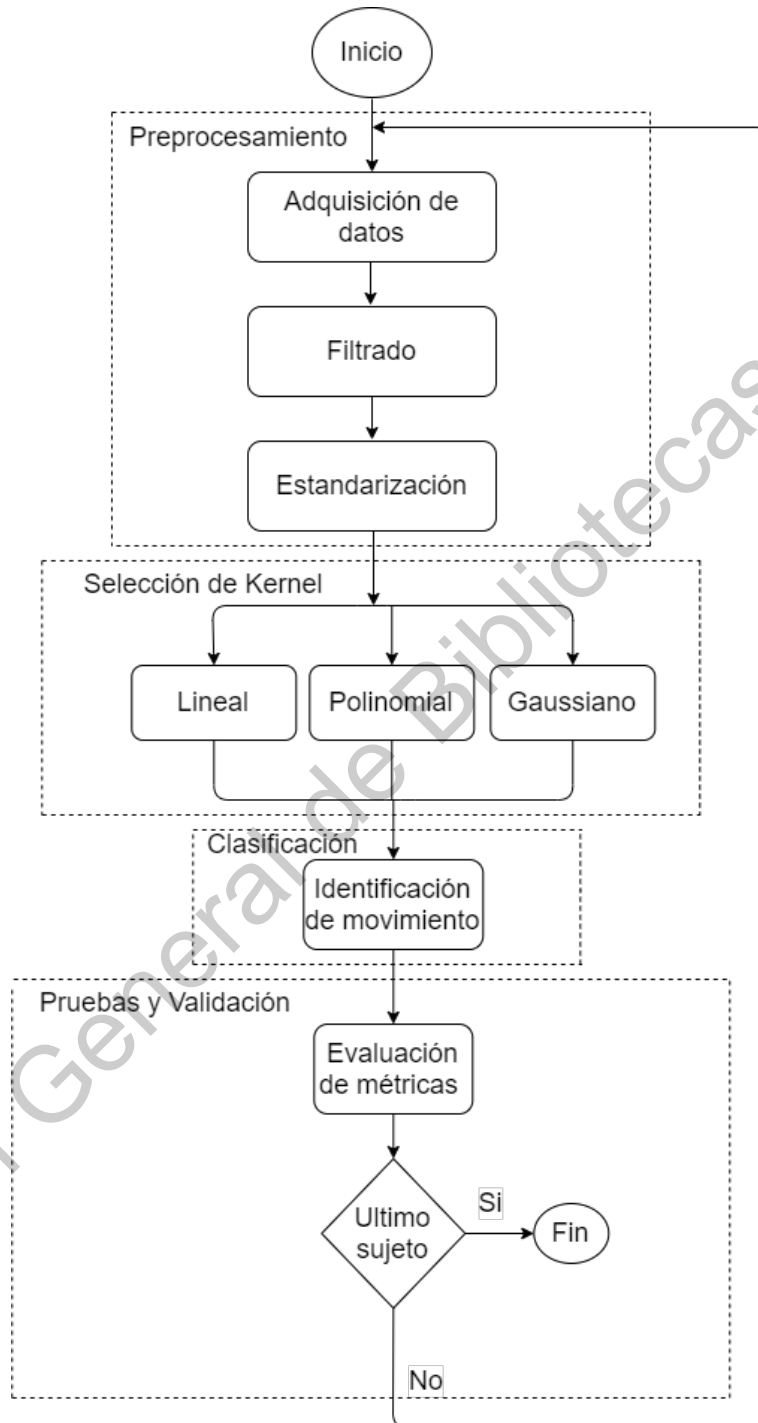


Fig. 3.9: Diagrama de la metodología propuesta.

La construcción del modelo se llevó a cabo utilizando una comparación en el desempeño de tres tipos de Kernel diferentes, solo se alcanzaron a implementar tres Kernel distintos debido a la complejidad de los mismos y al periodo de tiempo que se demora el entrenamiento de cada uno, los cuales se implementaron bajo las mismas condiciones en la distribución de los datos. La distribución de los datos se realizó de manera aleatoria, utilizando la base de datos de señales mioeléctricas, conformada por los 50 sujetos, en la que aleatoriamente se dividió la base de datos por individuos de tal manera que 80 % de los sujetos sea utilizado para hacer el entrenamiento del SVM, así mismo, se reservó el 20 % restante para hacer las pruebas y validación del modelo. Para realizar el entrenamiento del modelo se utilizaron cuarenta sujetos que representa el 80 % de los datos, estos datos fueron preparados de tal manera que aleatoriamente se seleccionen sin repetir los datos ya utilizados, esto fue posible generando una lista de todos los sujetos previamente enumerados desde el sujeto 1 hasta el sujeto 50, de los cuales se creo una lista de 40 números aleatorios como se muestra en la figura 3.10.

Cada uno de los 40 sujetos, contiene 10 movimientos por mano, 8 canales por movimiento y 5 eventos por canal, lo que se traduce en 800 señales mioeléctricas por sujeto de prueba, este modelo se construye enumerando los movimientos del 1 hasta el 20 tomando en cuenta los movimientos de la mano izquierda, tomando los de la mano derecha como los primeros 10 y los de la mano izquierda como los siguientes 10, una vez que se tienen los movimientos enumerados, se genera un ciclo de los movimientos del 1 al 20, al inicio del ciclo se toma el movimiento número 1 y se genera otro ciclo anidado para tomar los canales correspondientes al movimiento 1 y al inicio de este nuevo ciclo se selecciona el canal número 1, se vuelve a generar otro ciclo anidado para cada evento.

De esta manera nos aseguramos de utilizar las 800 señales mioeléctricas intercambiando aleatoriamente a los sujetos por cada evento, tomando el mismo movimiento,

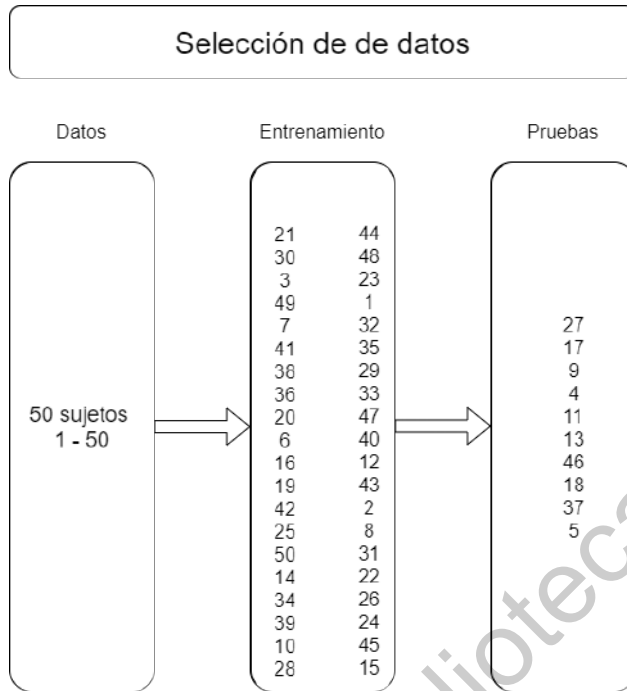


Fig. 3.10: Ejemplo de la selección aleatoria de los sujetos.

canal y eventos pero un sujeto distinto, limitándonos a los 40 sujetos que se tienen para el entrenamiento, lo que genera 32,000 señales solo para el entrenamiento.

La construcción de cada Kernel se implementó utilizando SVM, debido a que es una poderosa técnica de clasificación basada en la teoría del aprendizaje estadístico, lo que lo convierte principalmente en un clasificador de dos clases (Moslem et al., 2011b).

El criterio de optimización es el ancho del margen entre las clases, es decir, el área entre el margen de separación máximo de la clase A , hasta el margen de separación máximo de la clase B , dejando solo la mitad de la línea que separa ambas clases (el hiperplano de separación) definido por la distancia a los puntos de entrenamiento más cercanos (Kowalczyk, 2017). Estos puntos, llamados vectores de soporte, definen

la función de clasificación dada por la siguiente ecuación.

$$w^T \cdot w + b = 0 \quad (3.4)$$

Donde w^T representa la transpuesta de los datos, w el vector normal al hiperplano y b la constante de desplazamiento.

Mientras que la función Kernel contribuye a aumentar el alcance de clasificación del SVM, siendo una función que devuelve el resultado de un producto punto realizado en otro espacio. Más formalmente, podemos escribir:

Dada una función de mapeo $\phi : \chi \rightarrow \nu$, llamamos a la función $\kappa : \chi \rightarrow \mathfrak{R}$ definida por $\kappa(x, x^T) = \langle \phi(x), \phi(x^T) \rangle_\nu$, donde $\langle \cdot, \cdot \rangle_\nu$ denota un producto interno en ν , una función de Kernel (Kowalczyk, 2017).

La función Kernel se muestra a continuación (Hsu et al., 2003):

$$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \quad (3.5)$$

Kernel Lineal

El Kernel lineal es el kernel más simple, siendo un clasificador de vectores de soporte en un plano de 2 dimensiones. (Kowalczyk, 2017). Simplemente se define por:

$$K(x, x^T) = x \cdot x^T \quad (3.6)$$

Donde x y x^T son dos vectores.

Kernel Polinomial

Una función del kernel polinomial de grado d (López Díaz, 2018) se expresa como:

$$K(x, x^T) = (a \cdot b + c)^d \quad (3.7)$$

Donde a y b son dos vectores.

Tiene dos parámetros: donde c , representa un término constante y d , representa el grado del Kernel.

Kernel Gaussiano

Es un Kernel clasificador de vectores de soporte en infinitas dimensiones, se expresa como:

$$K(x, x^T) = e^{-\gamma(a-b)^2} \quad (3.8)$$

Por lo tanto:

$$e^{-\gamma(a-b)^2} = (s, s\sqrt{\frac{1}{1!}}a, s\sqrt{\frac{1}{2!}}a^2, \dots, s\sqrt{\frac{1}{\infty!}}a^\infty) \cdot (s, s\sqrt{\frac{1}{1!}}b, s\sqrt{\frac{1}{2!}}b^2, \dots, s\sqrt{\frac{1}{\infty!}}b^\infty) \quad (3.9)$$

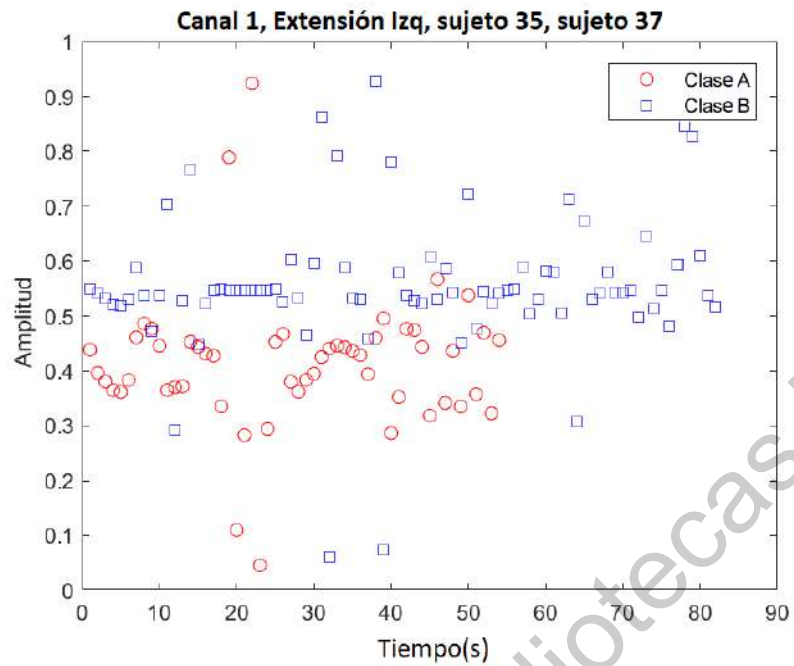
Donde a y b son dos vectores diferentes, y γ que es determinado por validación cruzada, escala la distancia cuadrada, por lo tanto, escala la influencia a las nuevas observaciones.

4. RESULTADOS

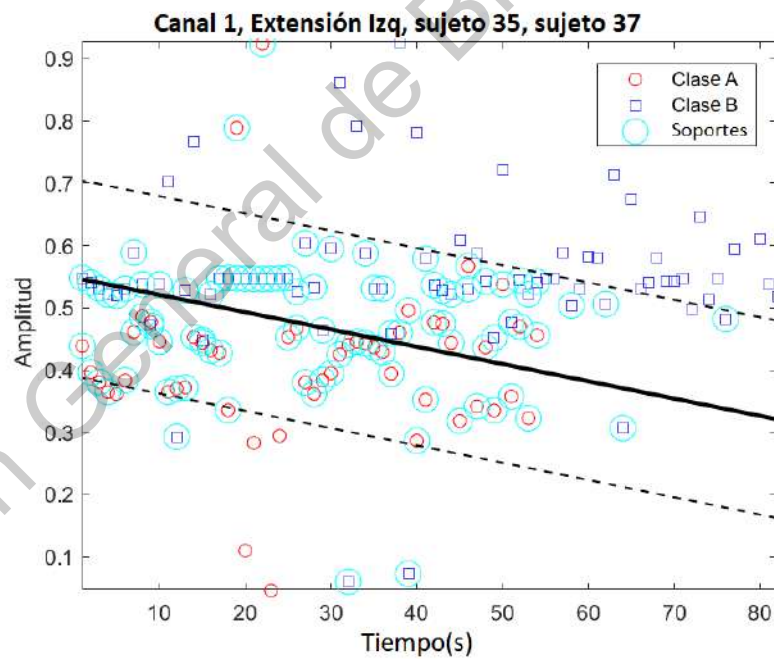
Los resultados presentados en esta investigación se separaron en tres experimentos, el primero consiste en generar un Kernel lineal, el segundo un Kernel polinomial y el tercero un Kernel Gaussiano, los cuales consisten en la clasificación de los movimientos de una mano con SVM y cada uno de los Kernel.

4.1. *Kernel Lineal*

A continuación, se visualizan algunas de las gráficas adquiridas con el Kernel lineal de varios sujetos. En cada gráfica se puede observar el comportamiento del Kernel lineal al ser implementado con distintos movimientos, en donde los canales más significativos muestran un mejor comportamiento, tanto del Kernel como del SVM.

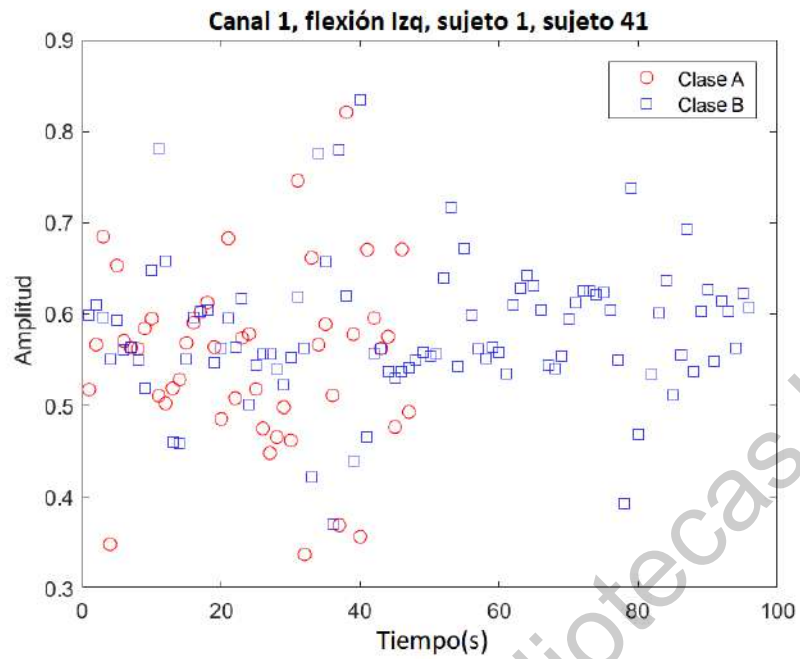


(a)

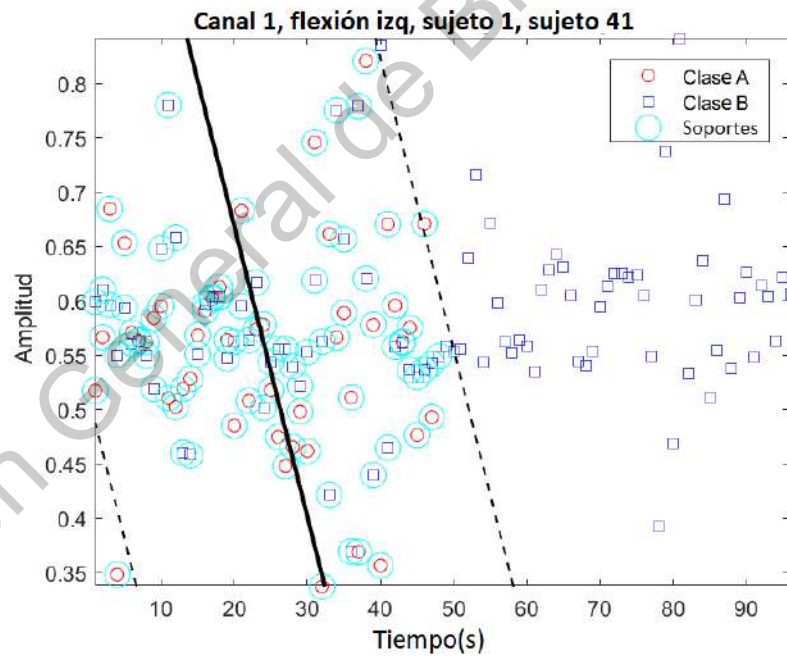


(b)

Fig. 4.1: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

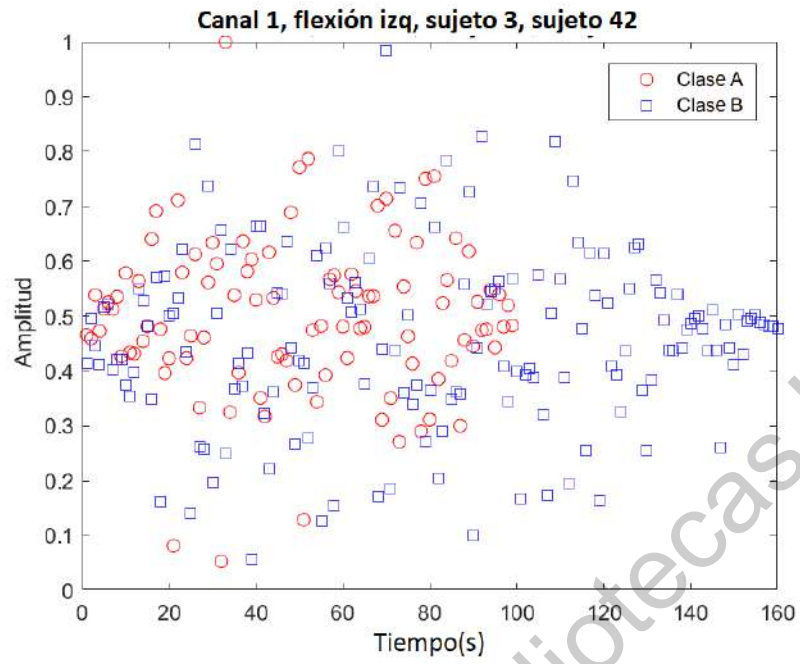


(a)

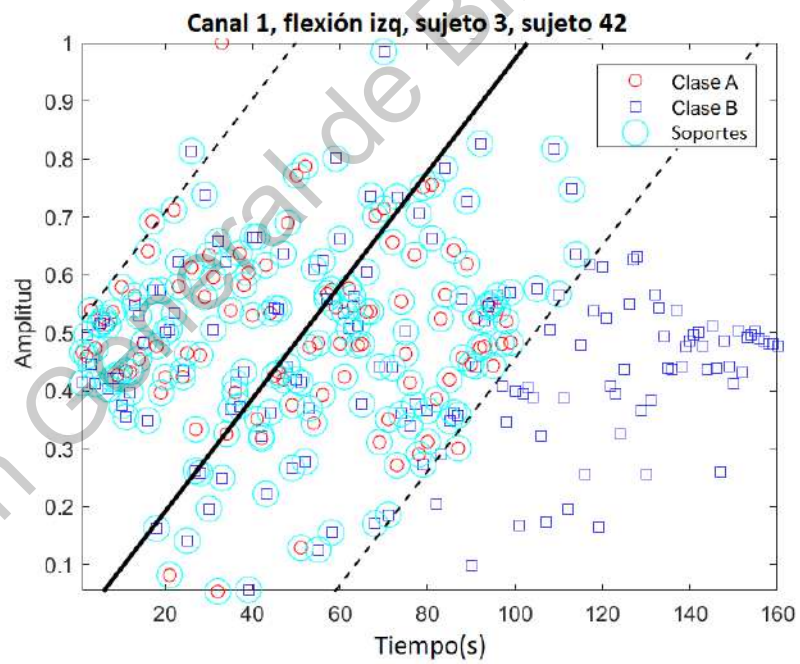


(b)

Fig. 4.2: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

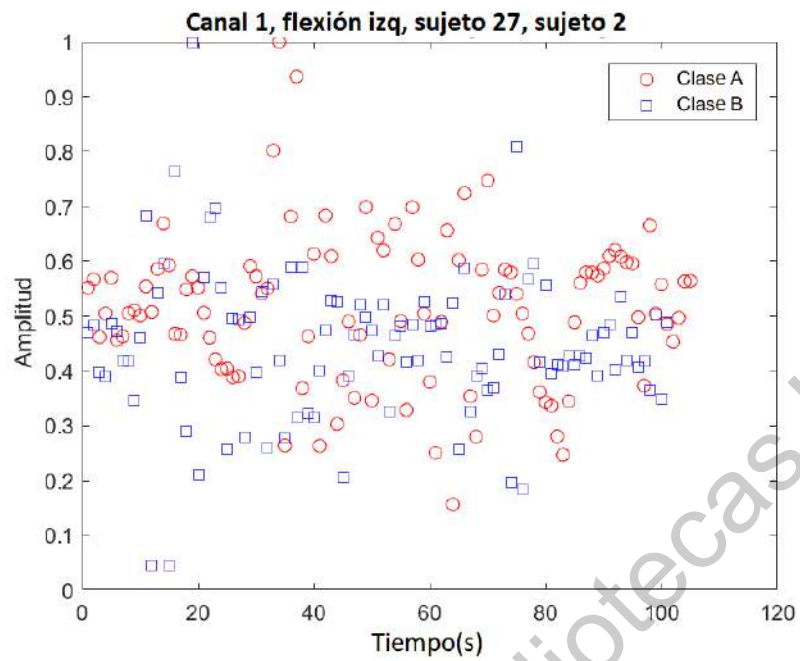


(a)

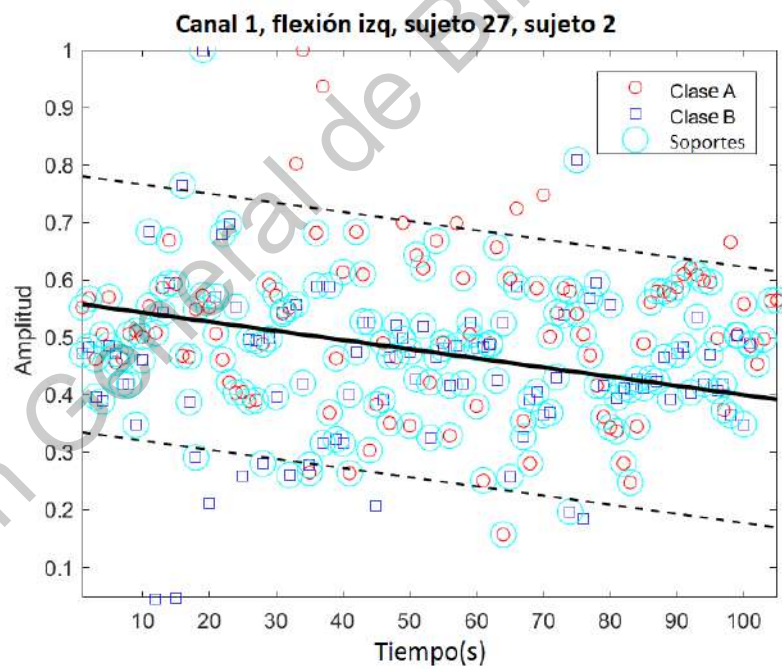


(b)

Fig. 4.3: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

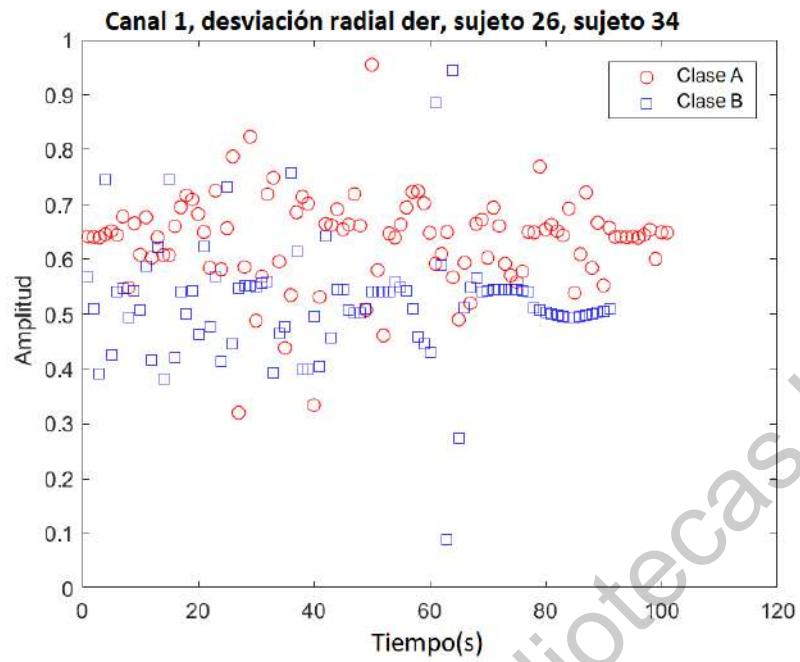


(a)

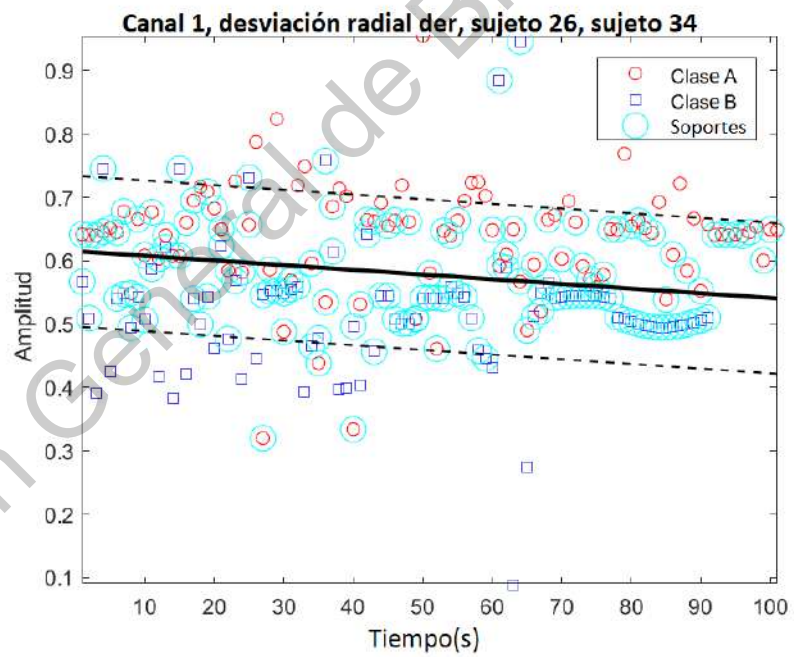


(b)

Fig. 4.4: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

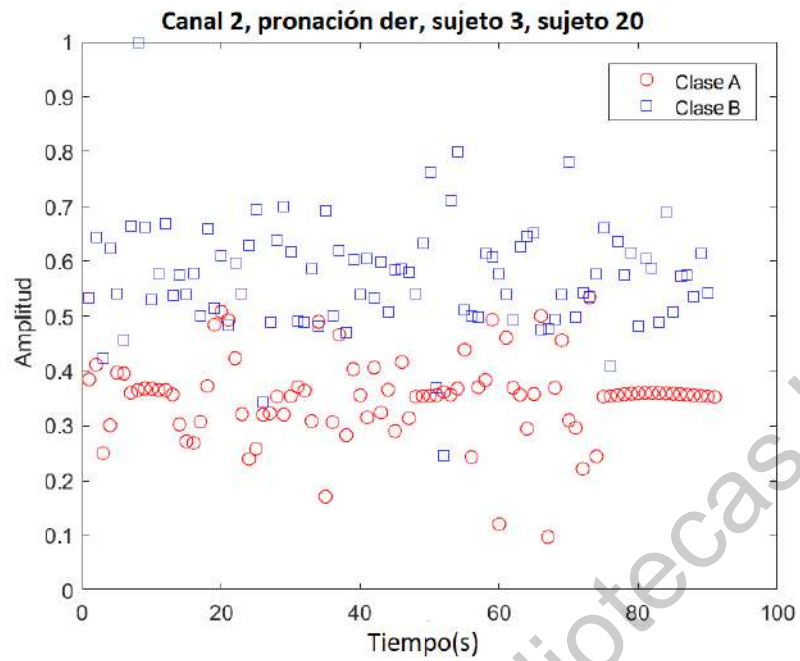


(a)

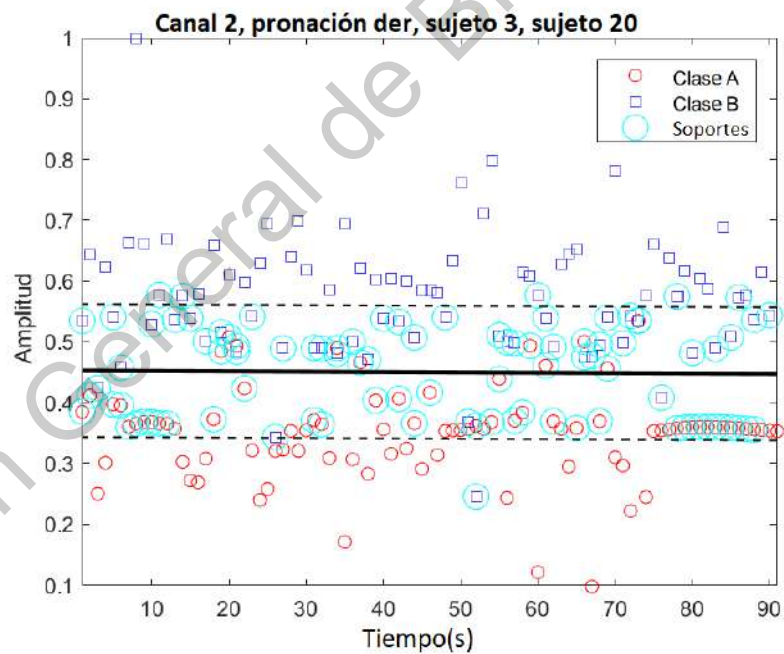


(b)

Fig. 4.5: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

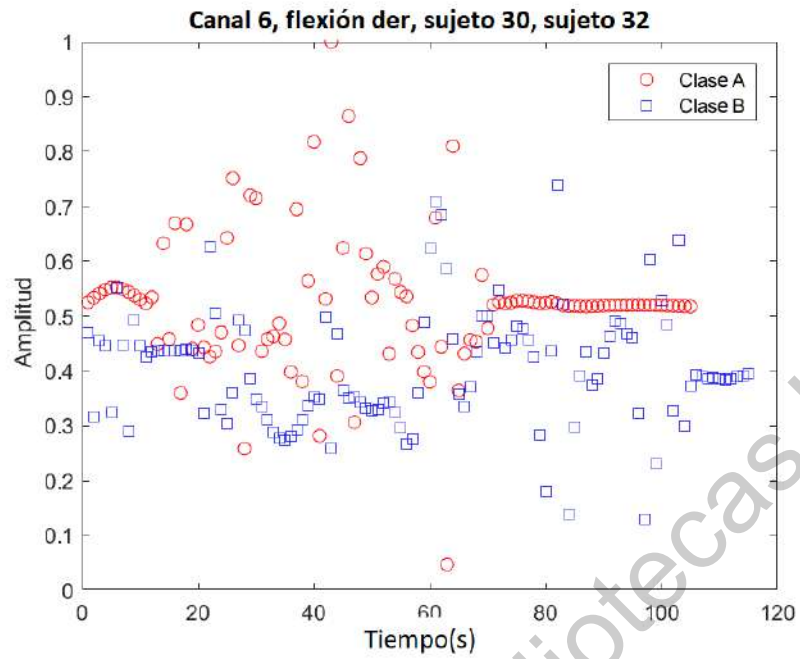


(a)

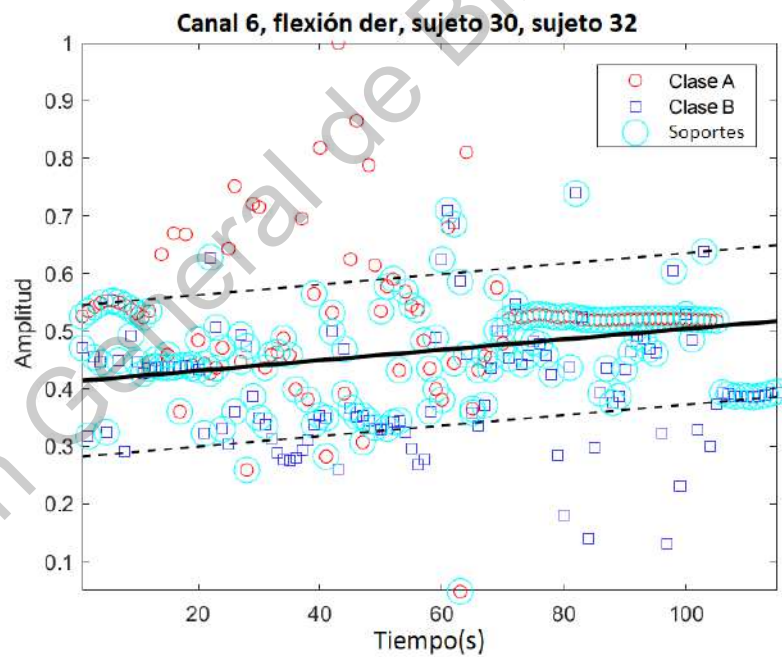


(b)

Fig. 4.6: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

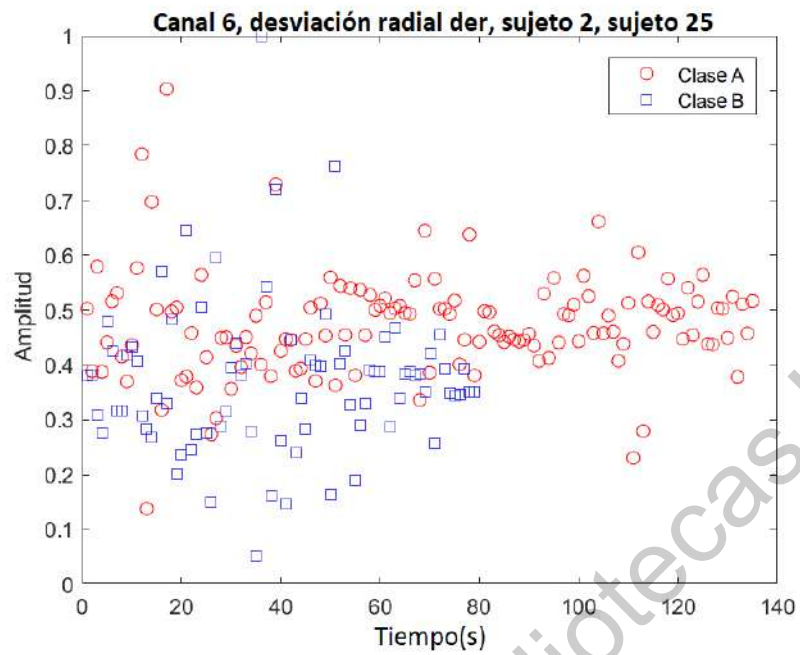


(a)

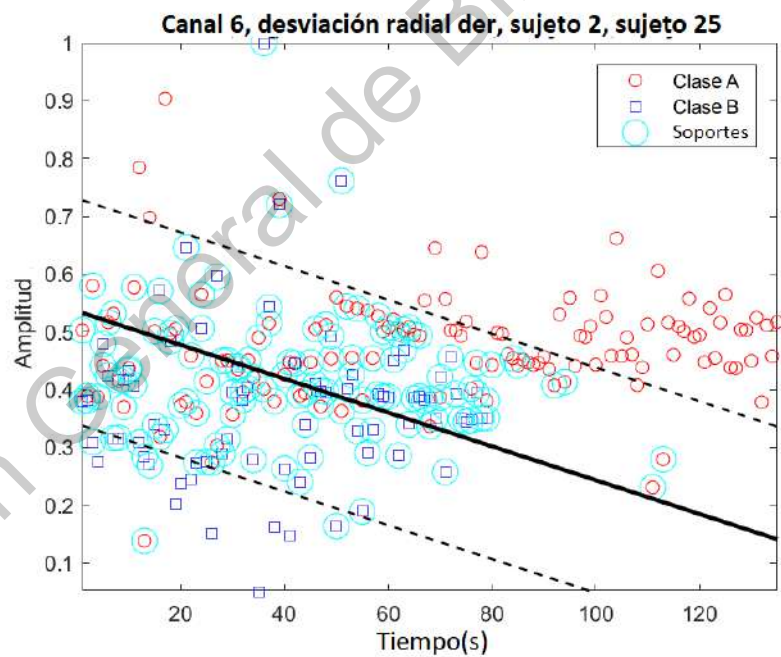


(b)

Fig. 4.7: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

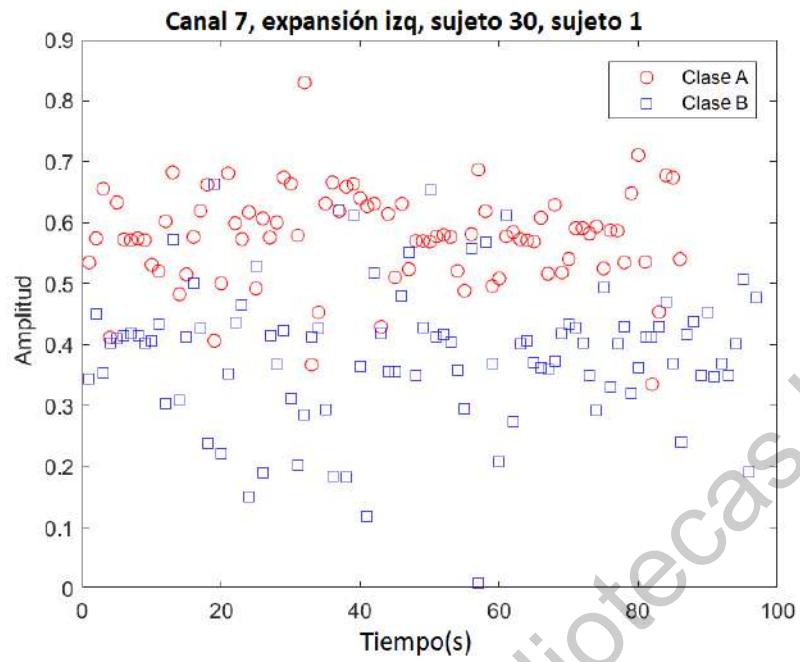


(a)

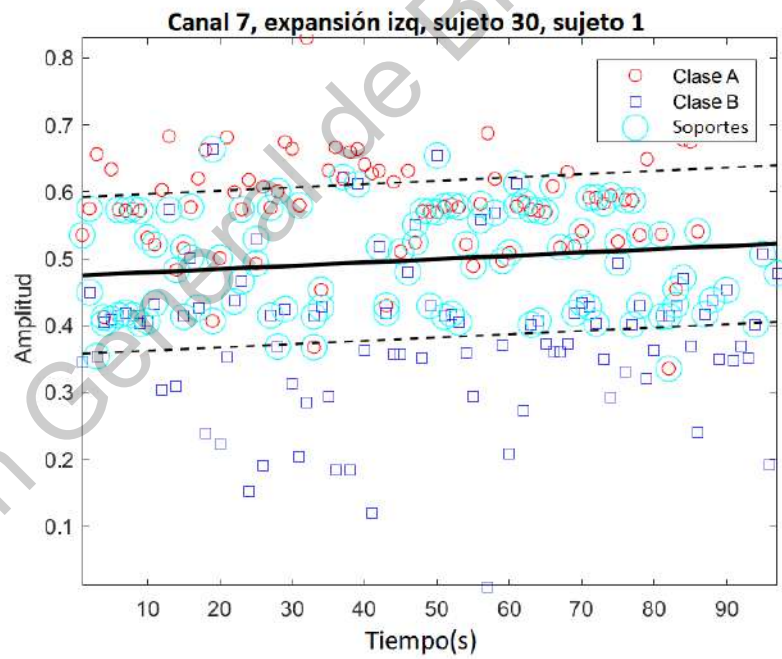


(b)

Fig. 4.8: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

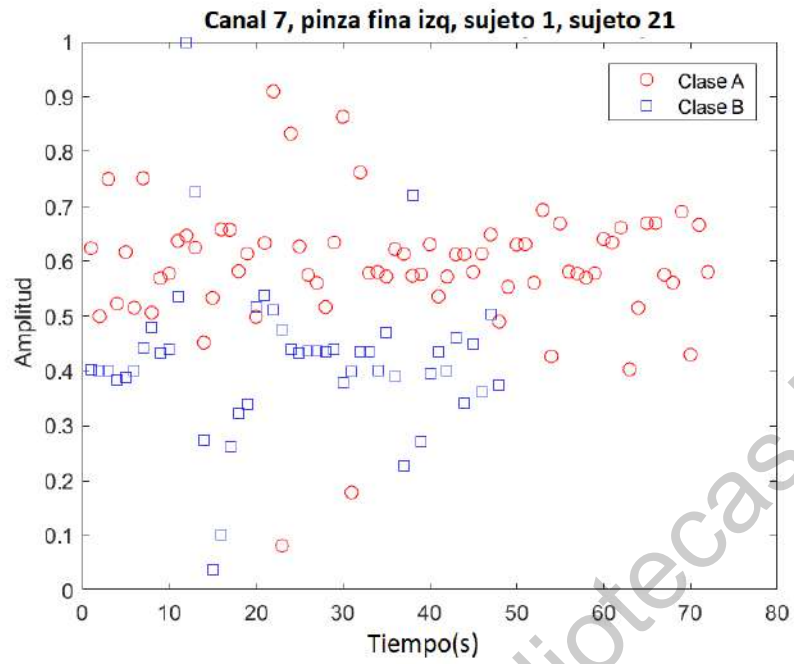


(a)

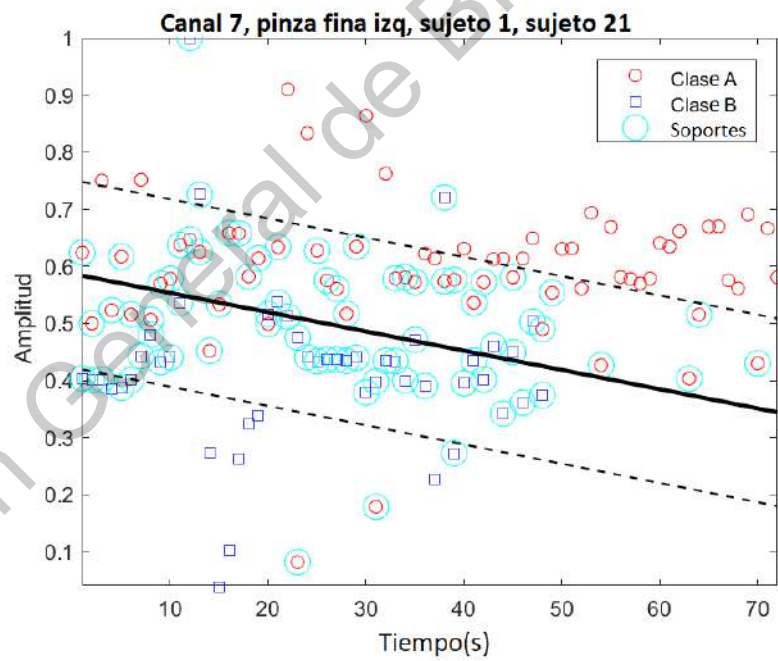


(b)

Fig. 4.9: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.



(a)



(b)

Fig. 4.10: Comparación entre la señal sin clasificar y la señal aplicando el Kernel lineal: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel lineal.

En la tabla 4.1, se muestra el comportamiento de exactitud y varianza, en cada uno de los movimientos utilizando el Kernel lineal, la tabla 4.1 muestra los movimientos y el comportamiento de la mano derecha en comparación con el comportamiento en la mano izquierda.

Tab. 4.1: Tabla con la exactitud y la varianza de los movimientos de la mano derecha e izquierda usando el Kernel lineal

Movimientos	Mano izquierda		Mano derecha	
	Exactitud	Varianza	Exactitud	Varianza
Posición Inicial	0.75864	0.14661	0.78427	0.15312
Pronación	0.83333	0.16667	0.75714	0.14048
Supinación	0.81735	0.16089	0.7624	0.14543
Extensión	0.75926	0.14417	0.77326	0.15643
Flexión	0.68519	0.12477	0.69833	0.12271
Desviación Cubital	0.78574	0.15237	0.70729	0.12768
Desviación Radial	0.77895	0.148	0.78428	0.15262
Pinza Fina	0.78769	0.15443	0.75674	0.14332
Pinza Gruesa	0.80556	0.16019	0.73307	0.14012
Expansión	0.83247	0.16632	0.7574	0.14777

En la figura 4.11, se muestra el comportamiento de exactitud, en cada uno de los movimientos utilizando el Kernel lineal, la figura 4.11 muestra los movimientos y el comportamiento de la mano derecha en comparación con el comportamiento en la mano izquierda.

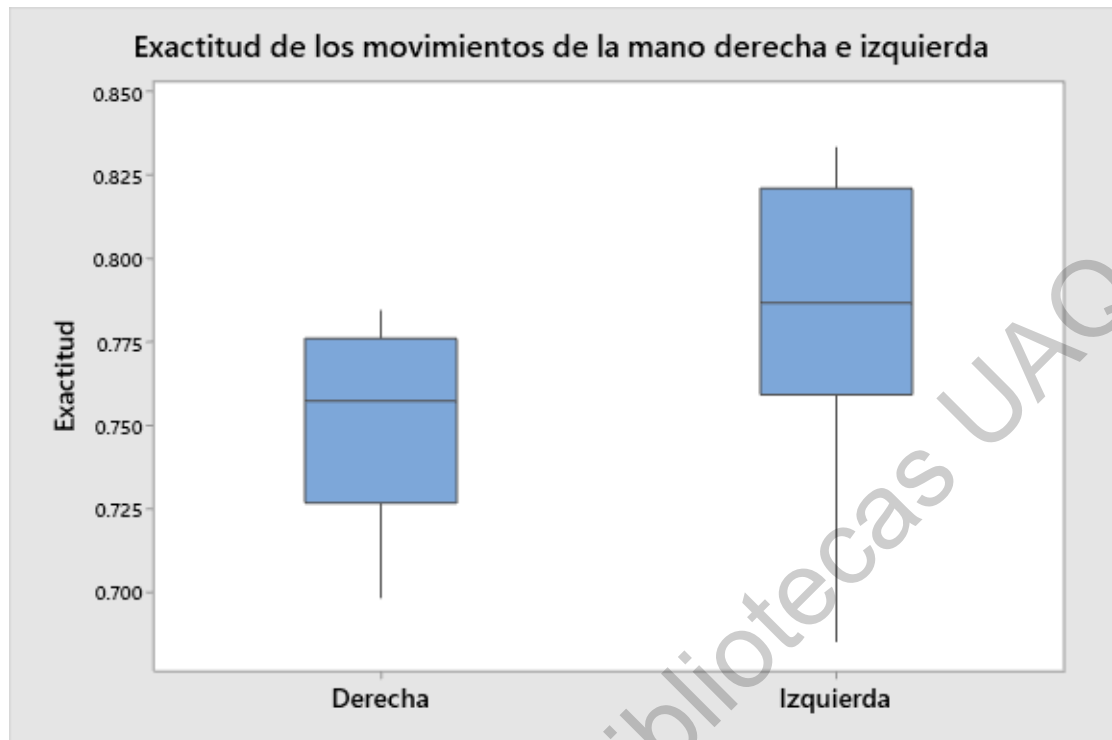


Fig. 4.11: Máquina de Soporte Vectorial (García et al., 2016)

4.2. Kernel Polinomial

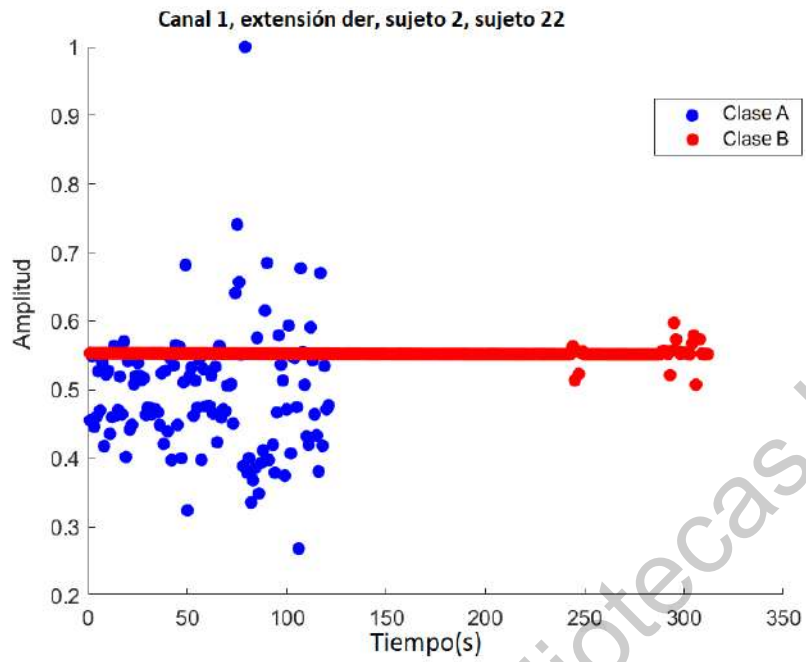
A continuación, se visualiza el Kernel polinomial, para este Kernel se llevó a cabo una serie de pruebas en la que consta de comparar el comportamiento de distintos Kernel polinomiales que van desde el orden 2 hasta el orden 7, se visualizan algunas de las gráficas adquiridas con cada Kernel polinomial de varios sujetos. En cada grafica se puede observar el comportamiento del Kernel polinomial al ser implementado con distintos movimientos, en donde los canales más significativos muestran un mejor comportamiento, tanto del Kernel como del SVM.

4.2.1. Kernel polinomial de orden 2

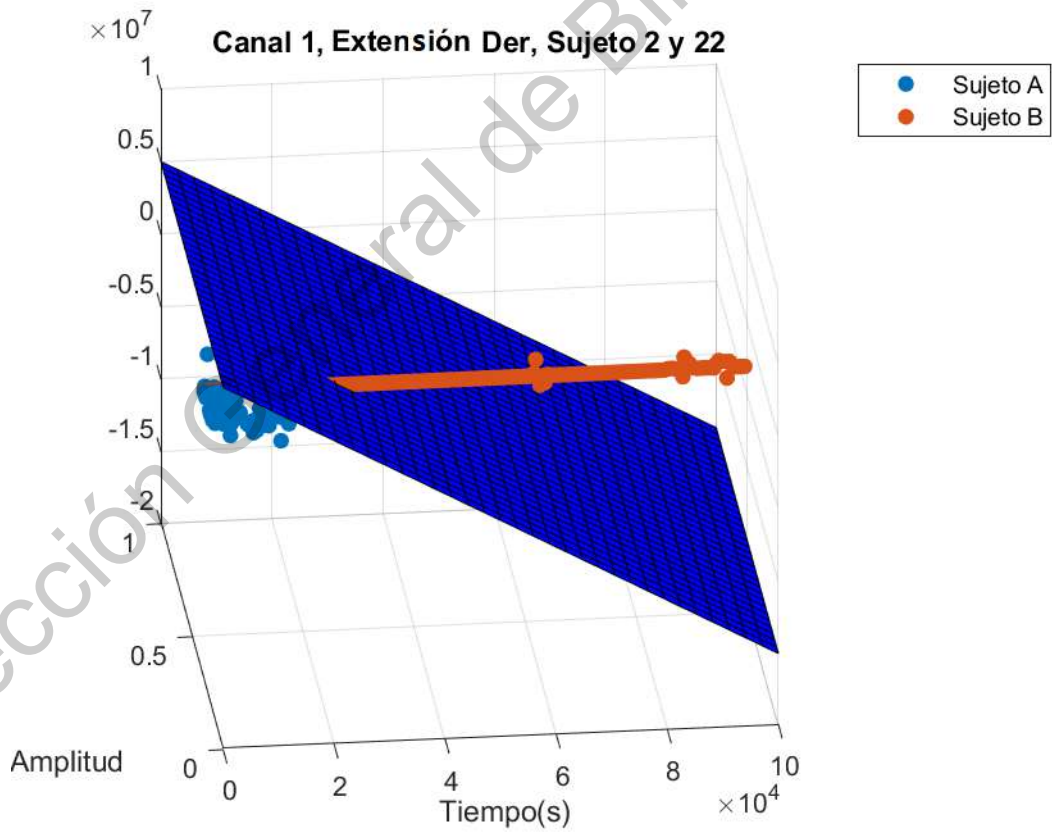
A continuación, se visualiza el Kernel polinomial de orden 2, para la obtención de estas graficas se utilizó el movimiento de extensión derecha con la finalidad de

estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

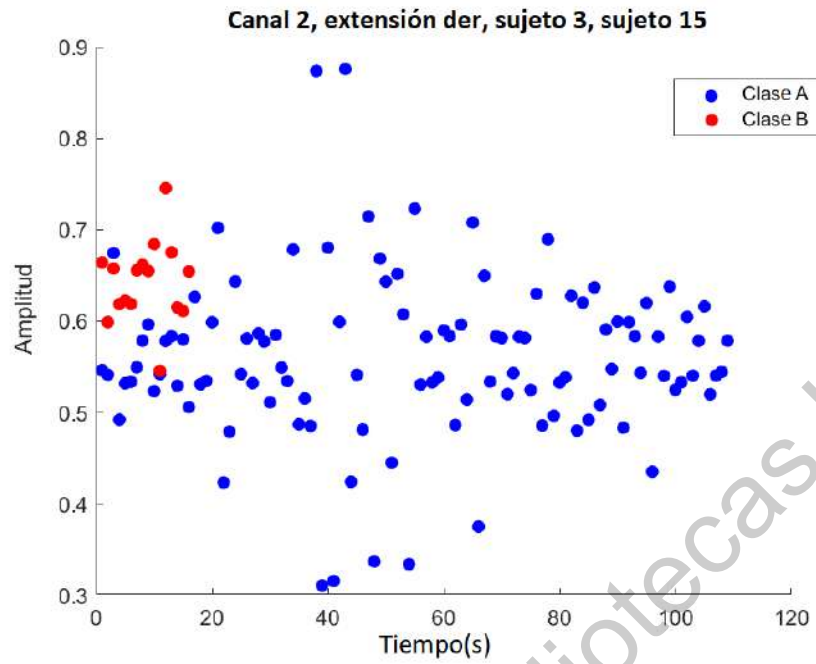


(a)

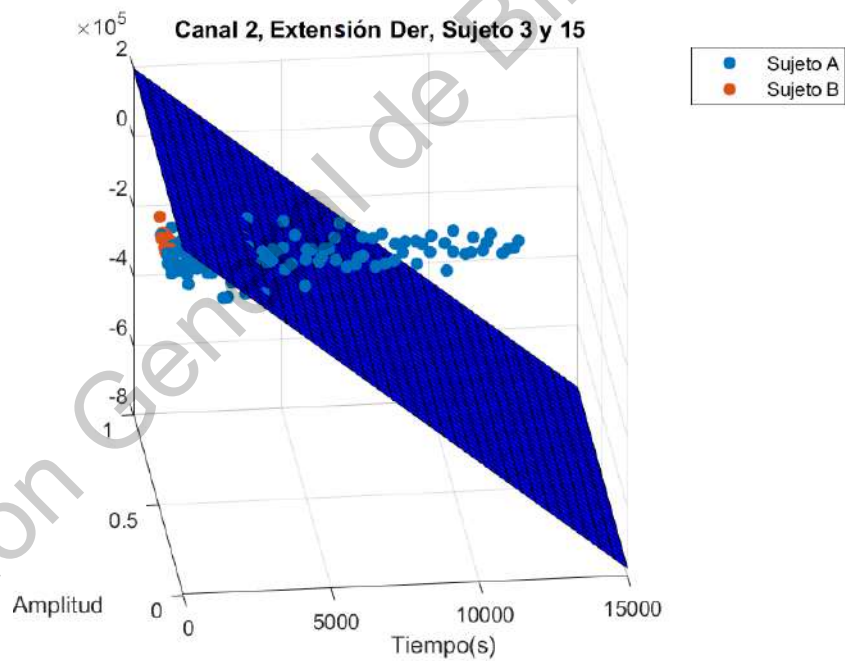


(b)

Fig. 4.12: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

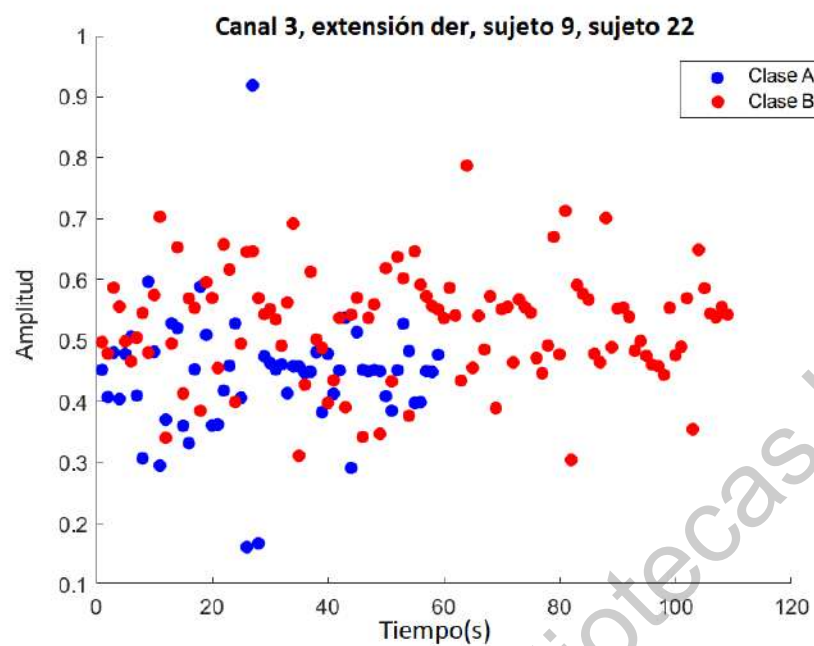


(a)

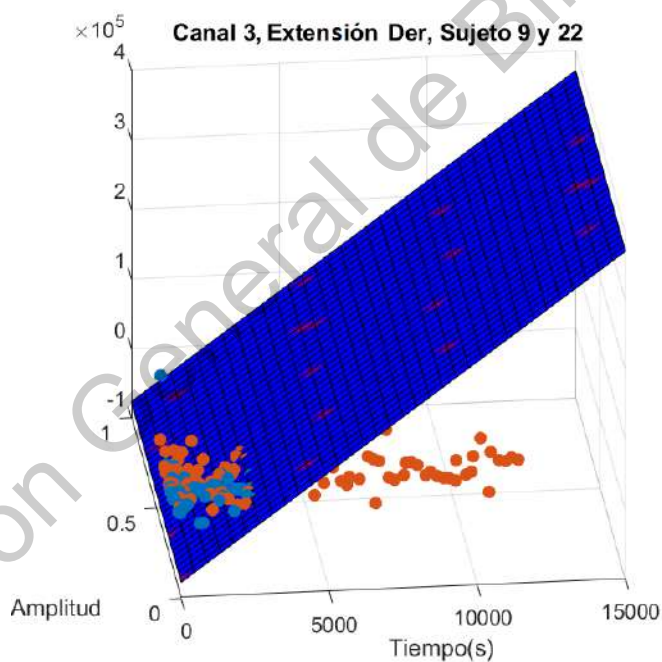


(b)

Fig. 4.13: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

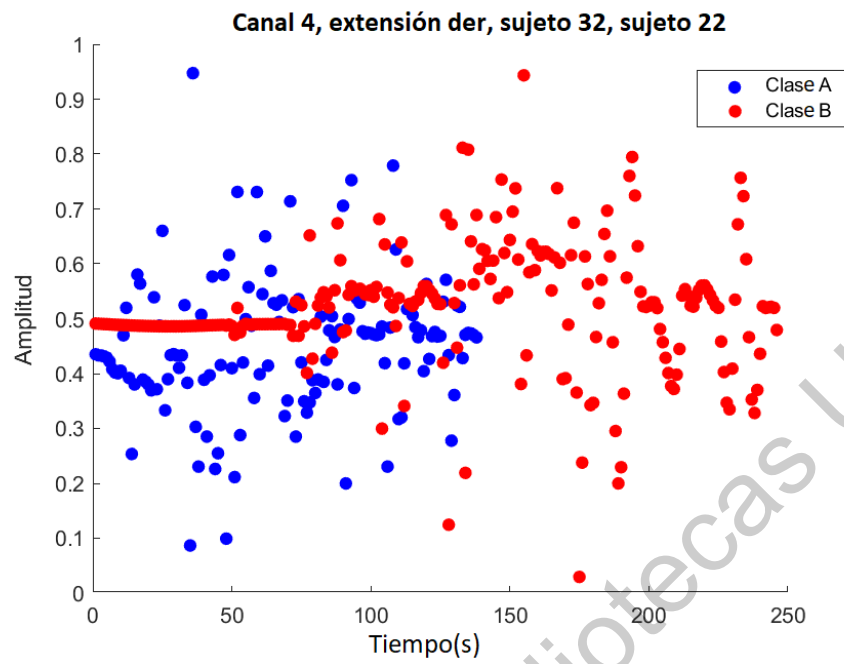


(a)

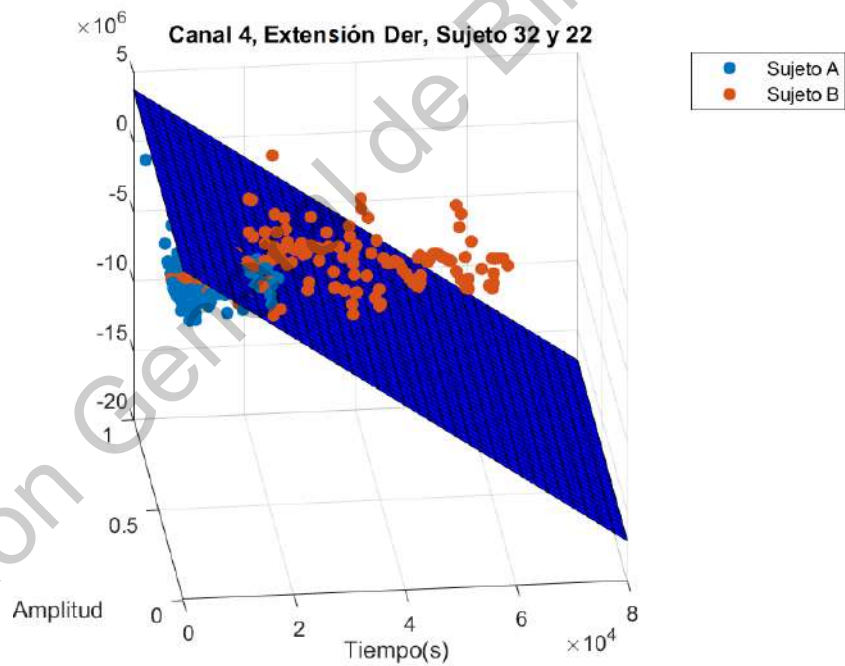


(b)

Fig. 4.14: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

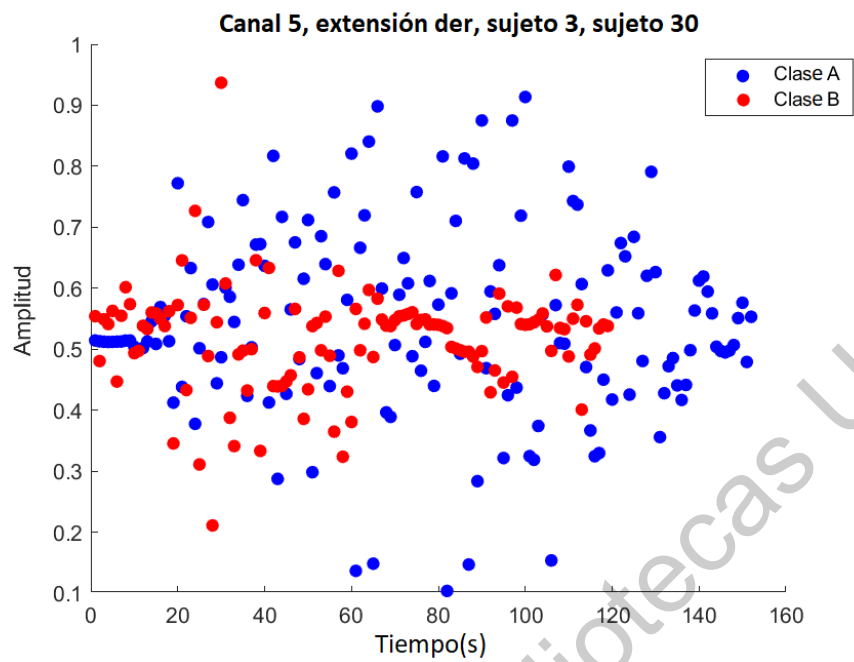


(a)

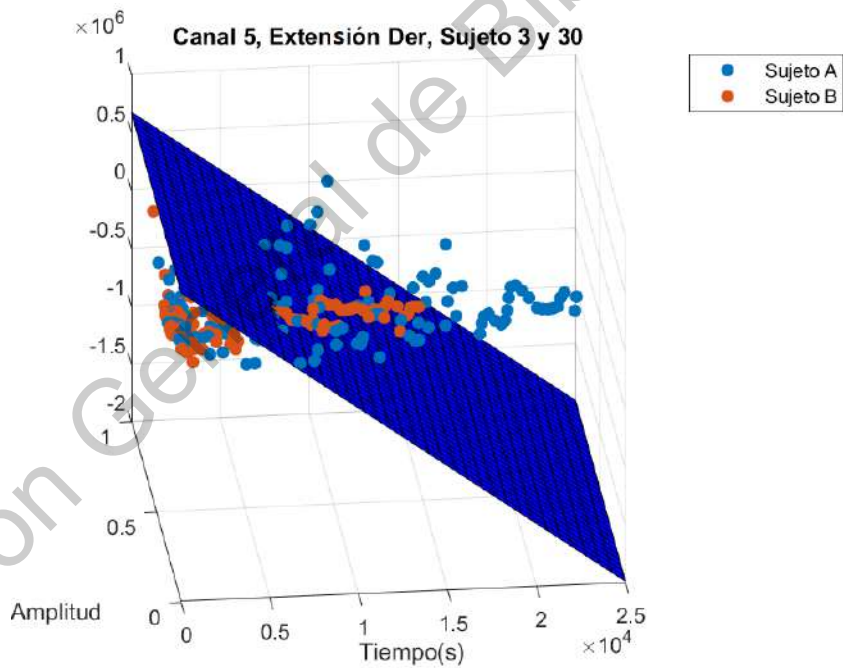


(b)

Fig. 4.15: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

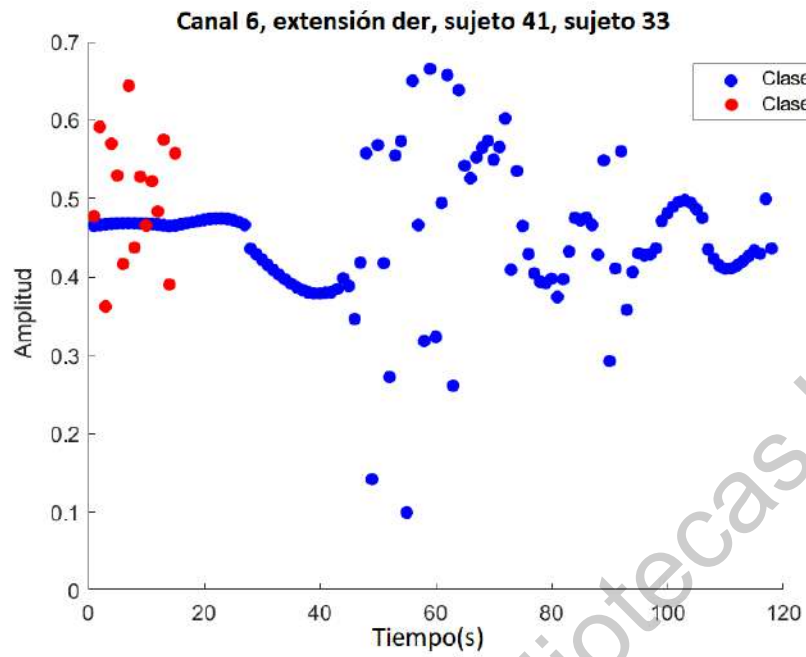


(a)

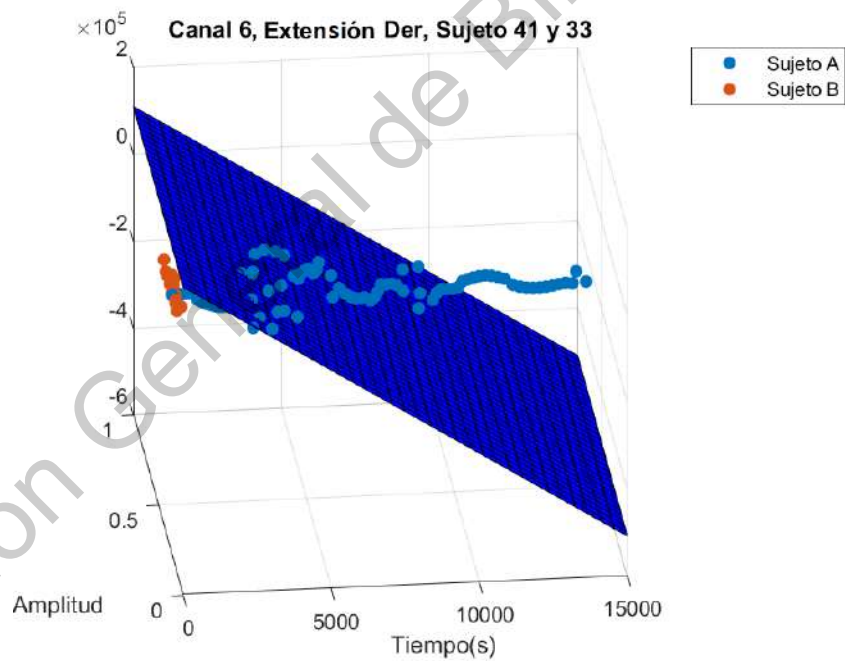


(b)

Fig. 4.16: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

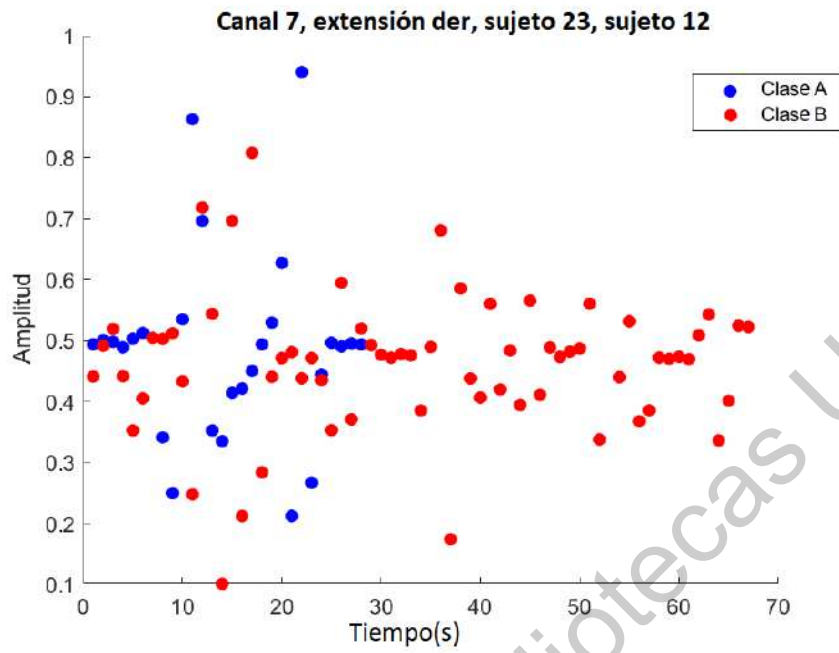


(a)

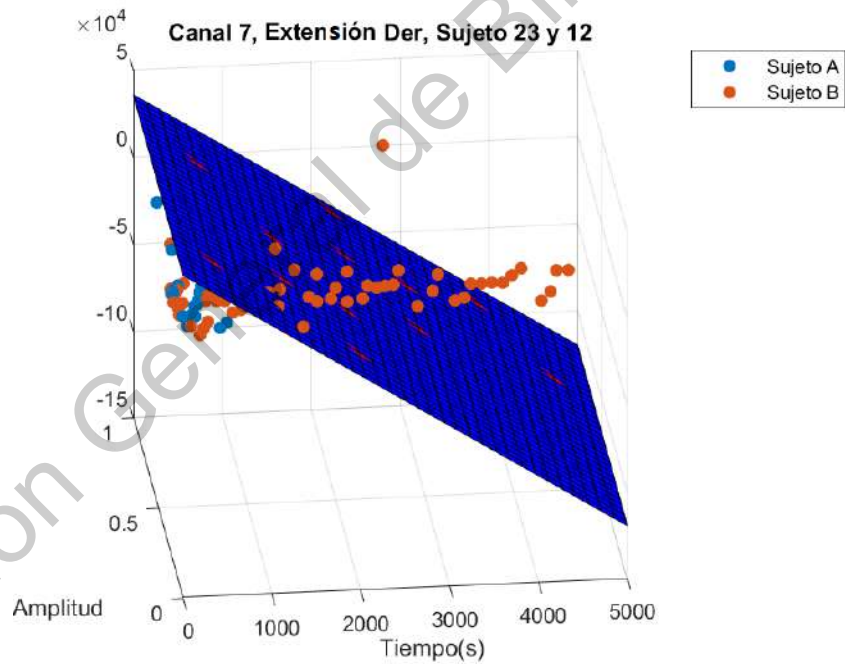


(b)

Fig. 4.17: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

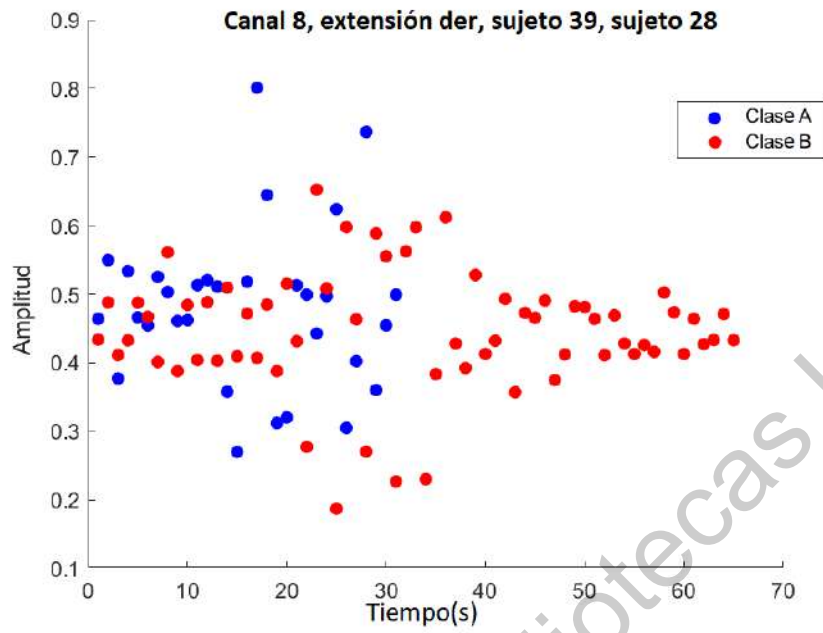


(a)

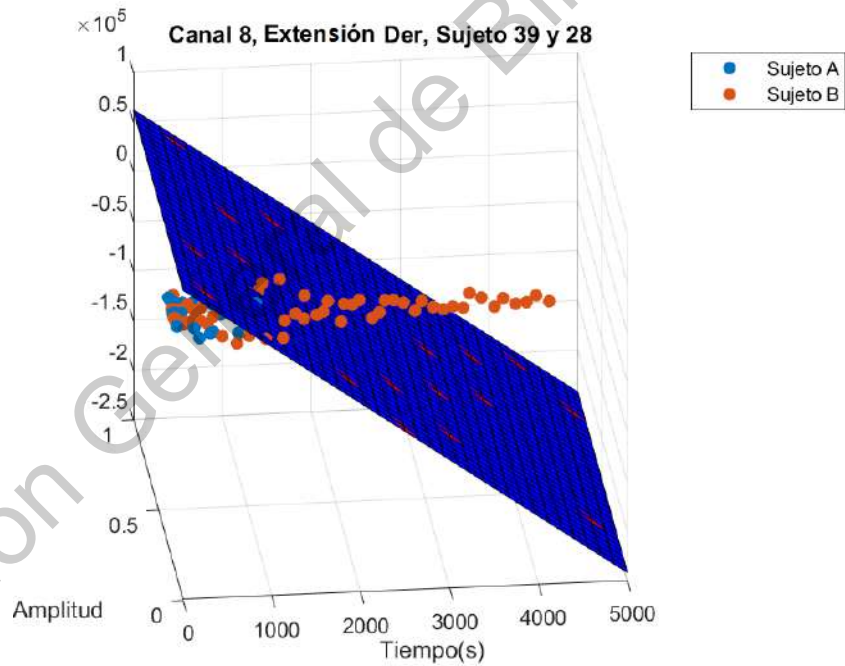


(b)

Fig. 4.18: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.



(a)



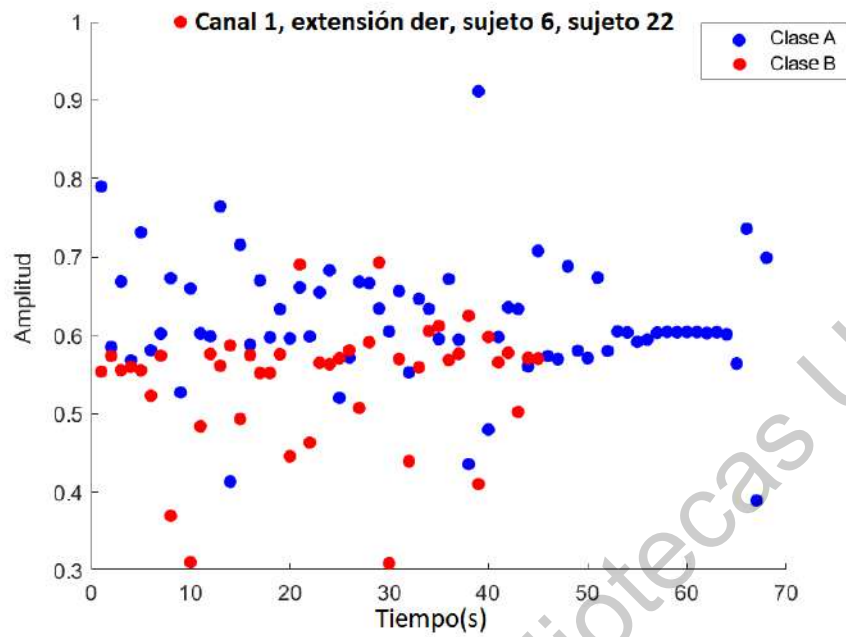
(b)

Fig. 4.19: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 2: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 2.

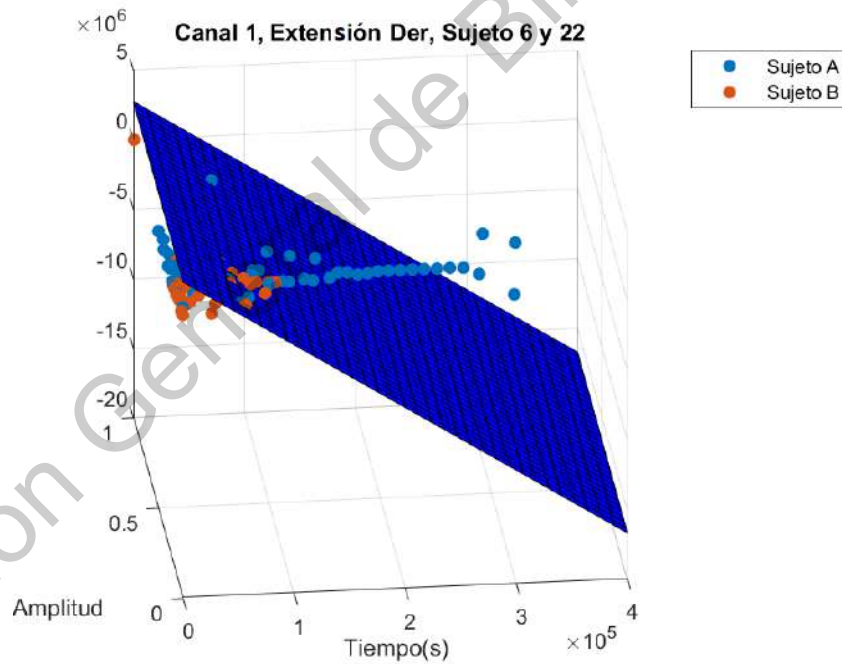
4.2.2. *Kernel polinomial de orden 3*

A continuación, se visualizan las gráficas correspondientes para el comportamiento de el Kernel polinomial de orden 3, para la obtención de estas gráficas se utilizó el movimiento de extensión derecha con la finalidad de estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

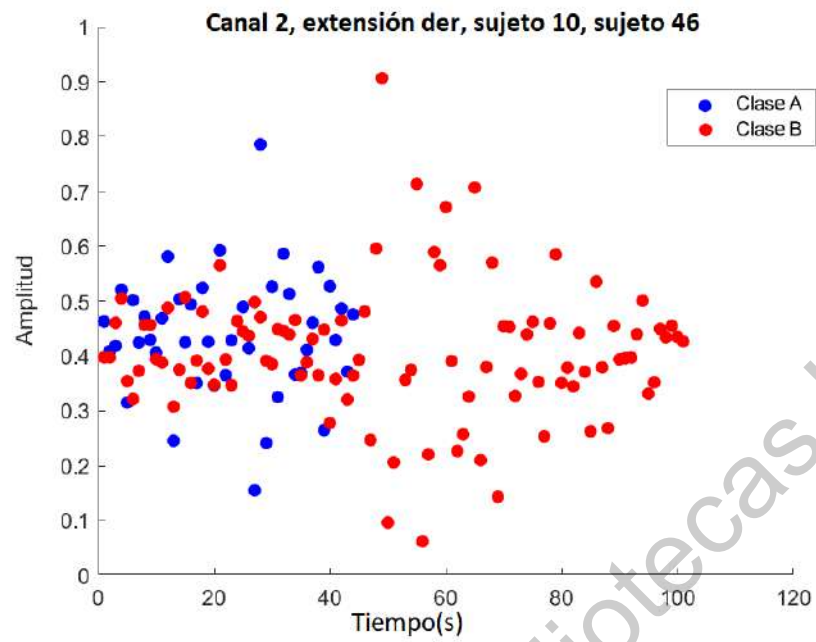


(a)

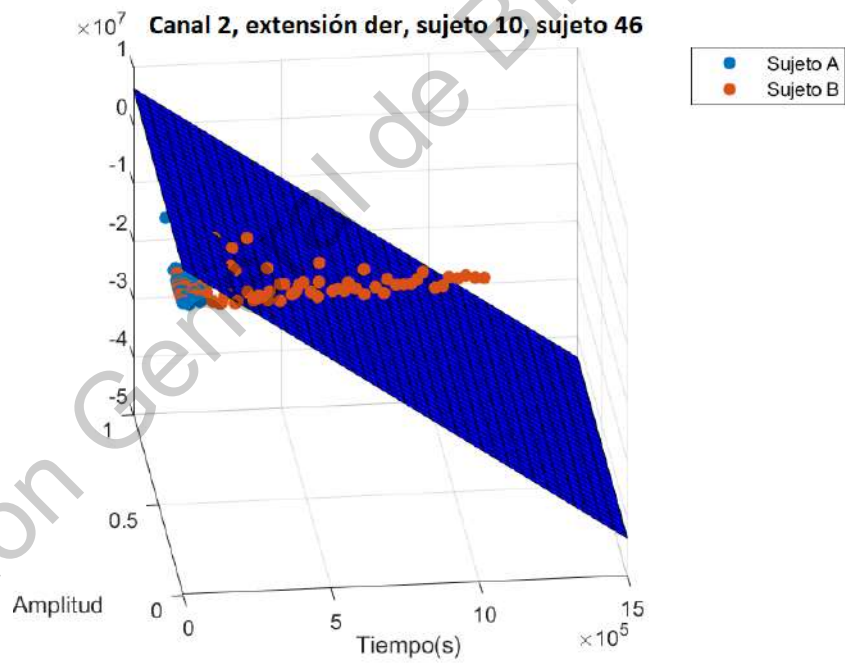


(b)

Fig. 4.20: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

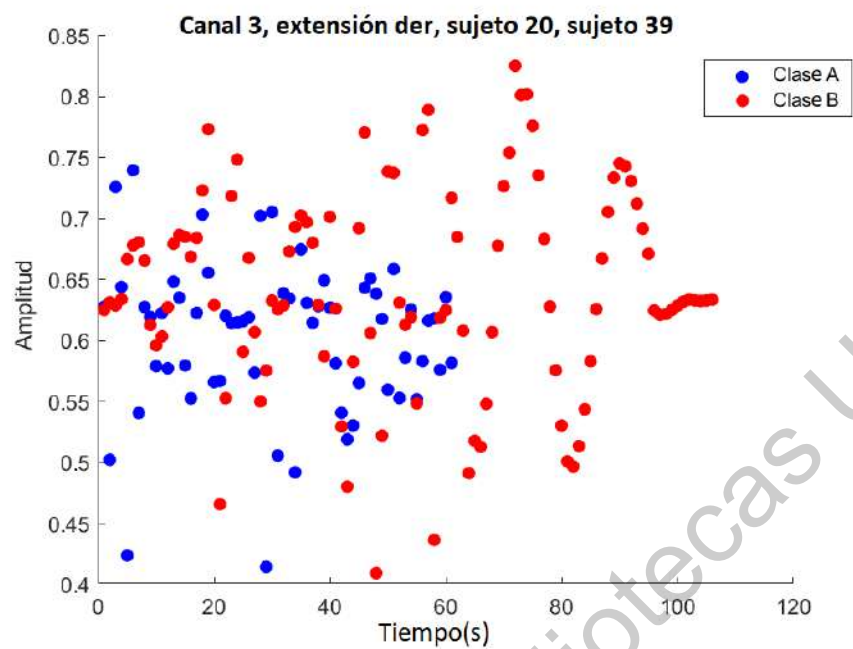


(a)

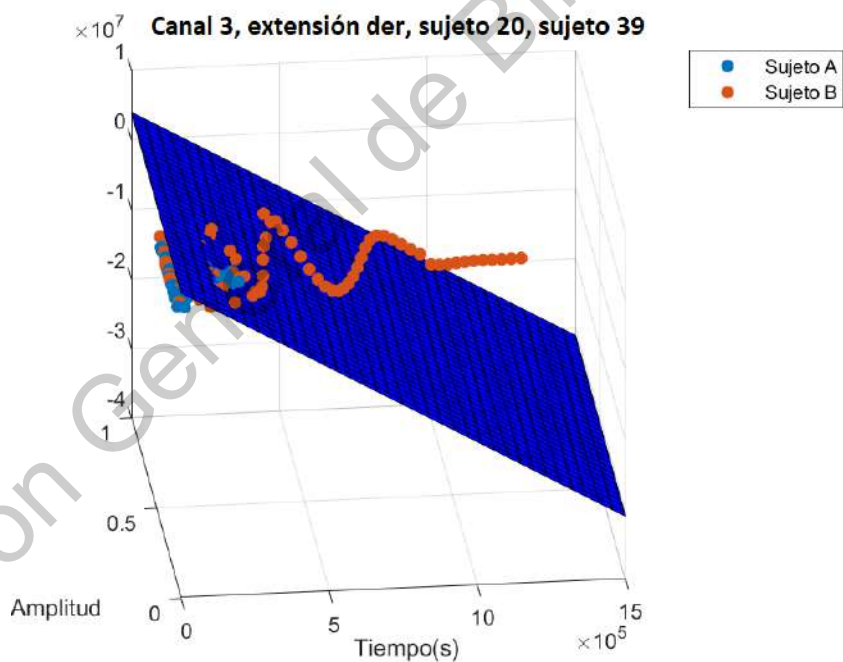


(b)

Fig. 4.21: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

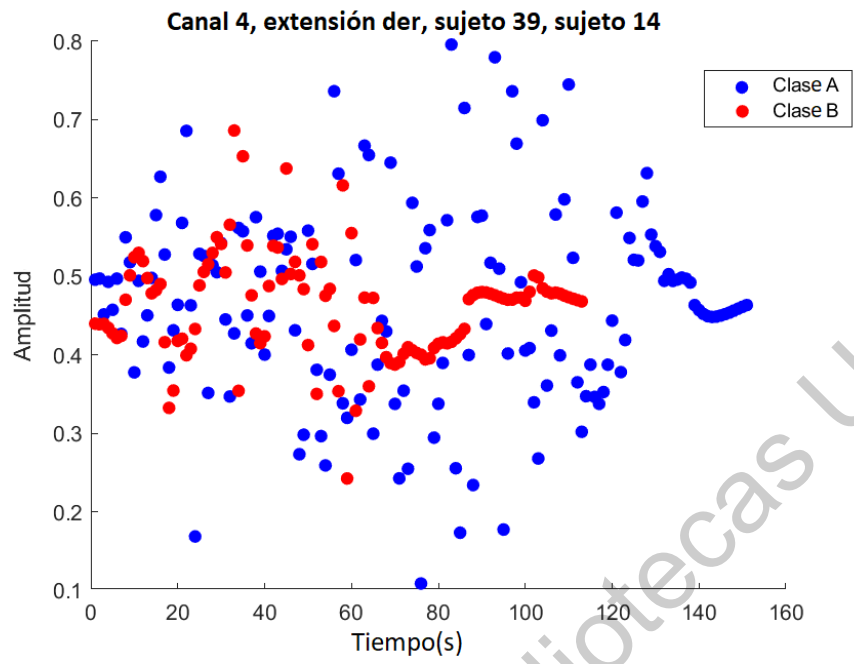


(a)

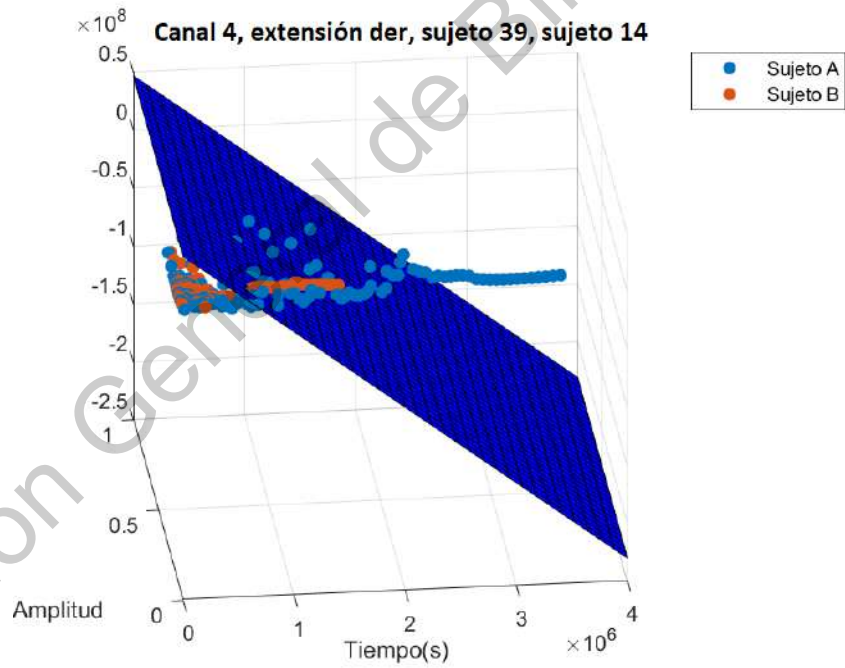


(b)

Fig. 4.22: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

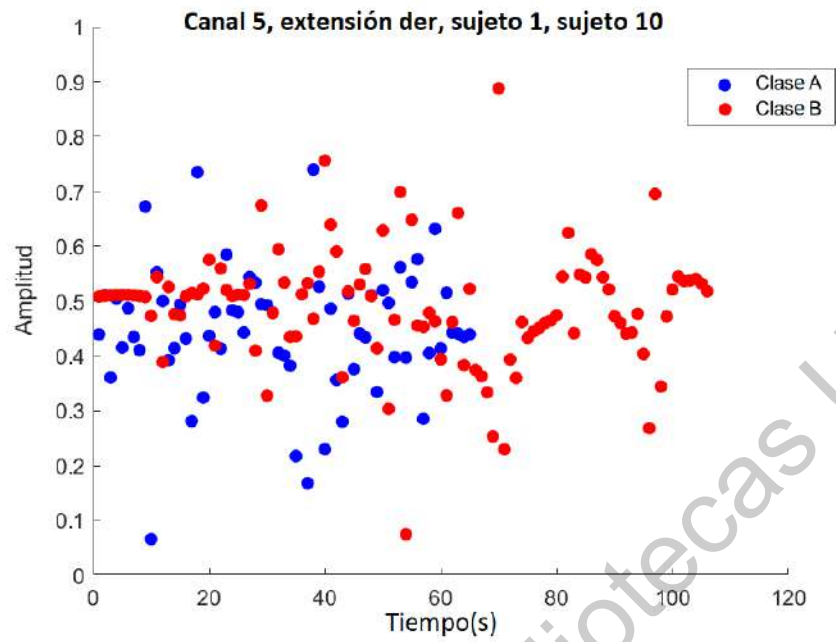


(a)

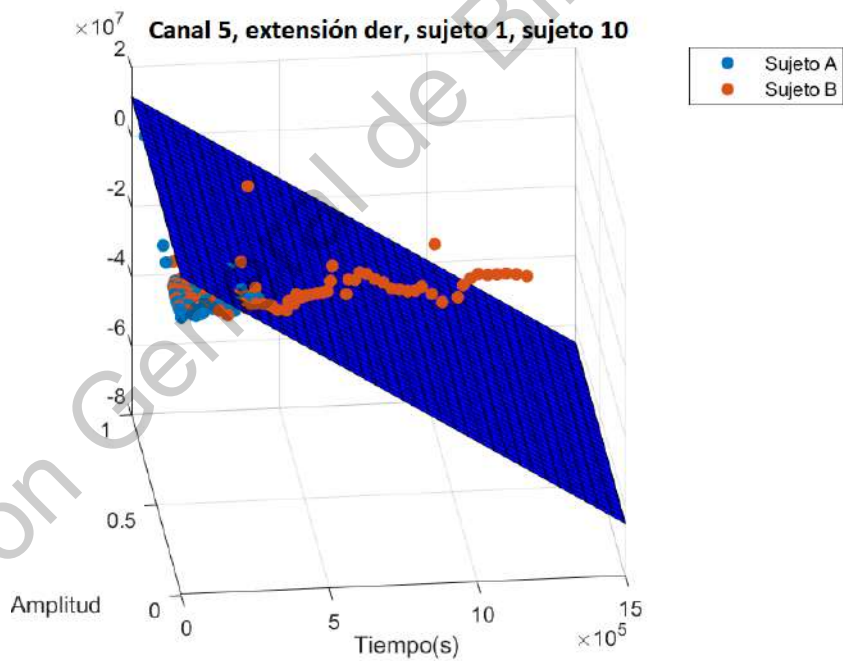


(b)

Fig. 4.23: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

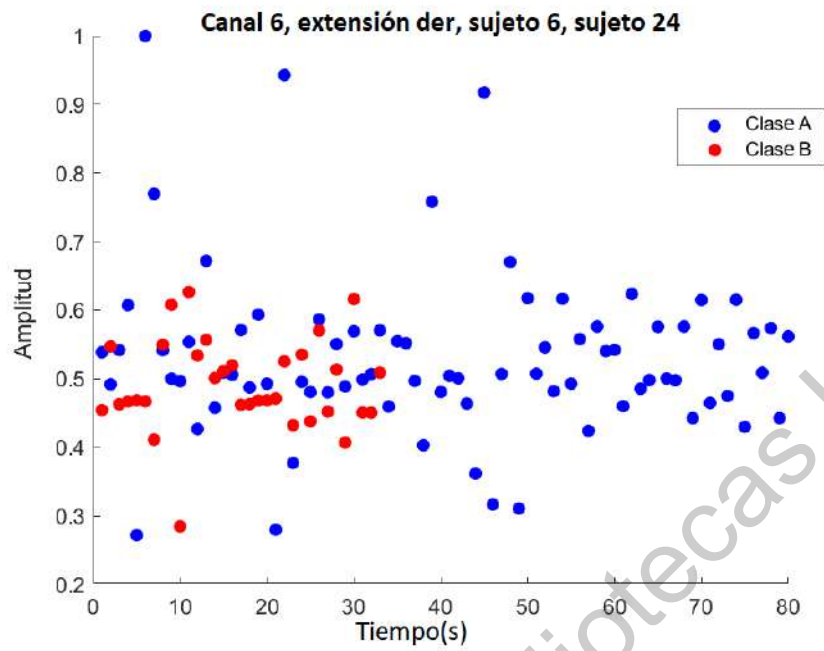


(a)

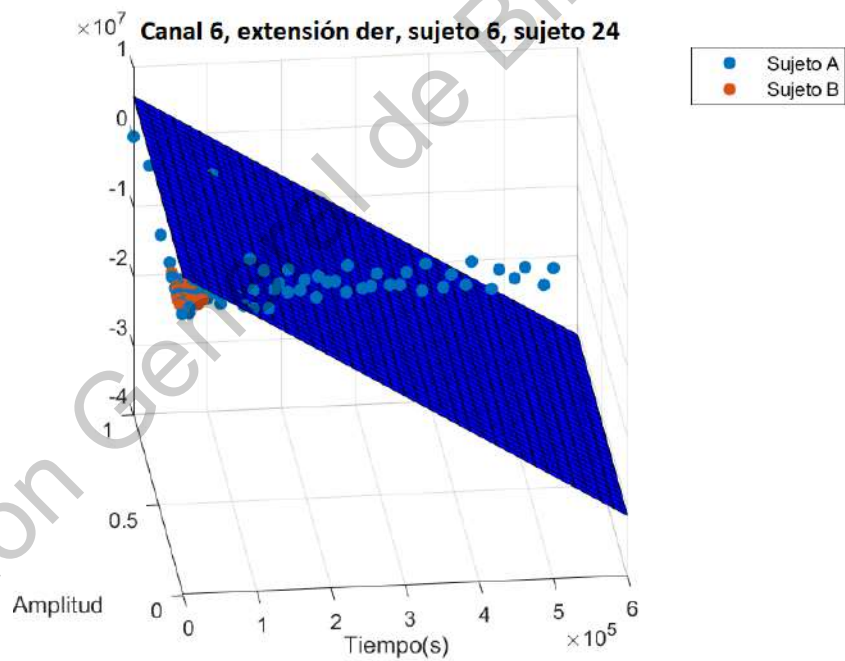


(b)

Fig. 4.24: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

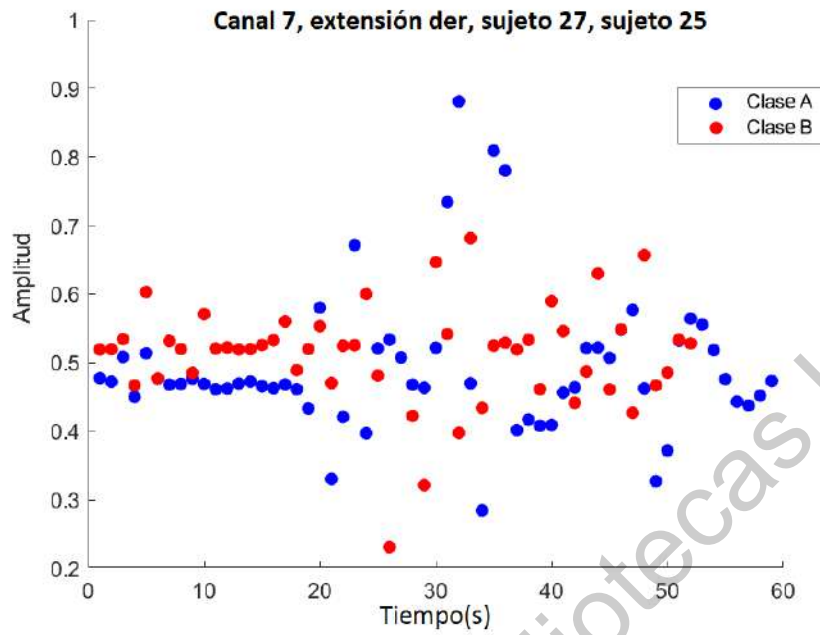


(a)

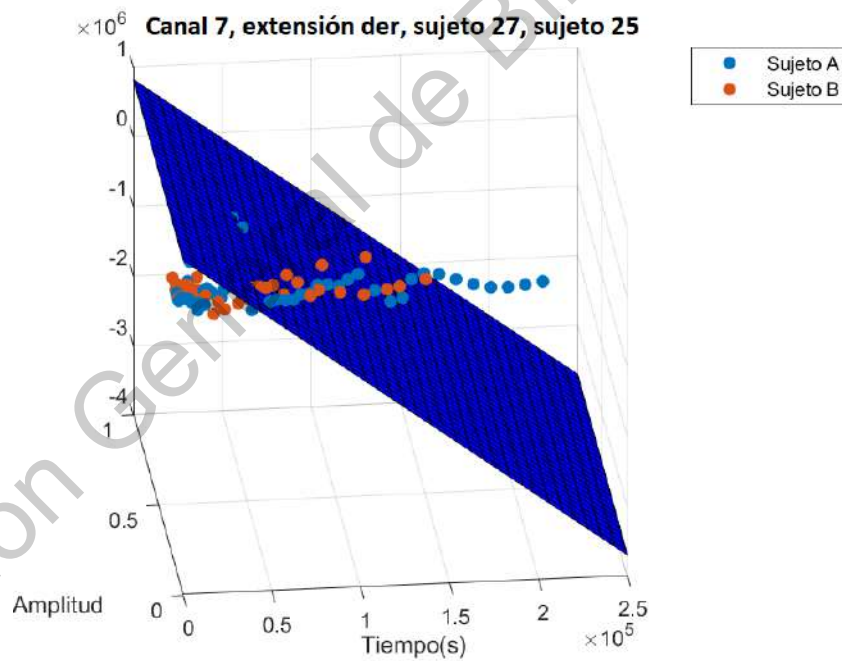


(b)

Fig. 4.25: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

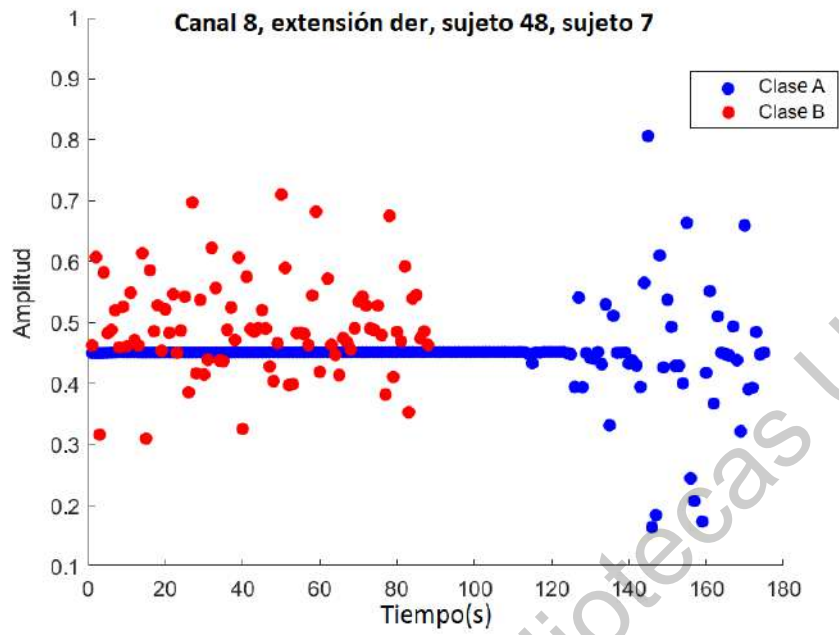


(a)

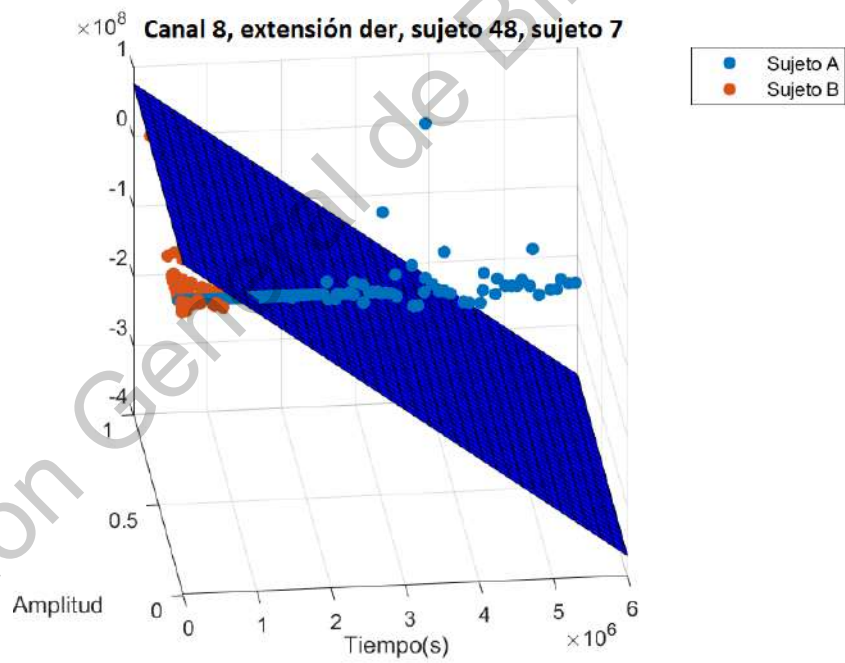


(b)

Fig. 4.26: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.



(a)



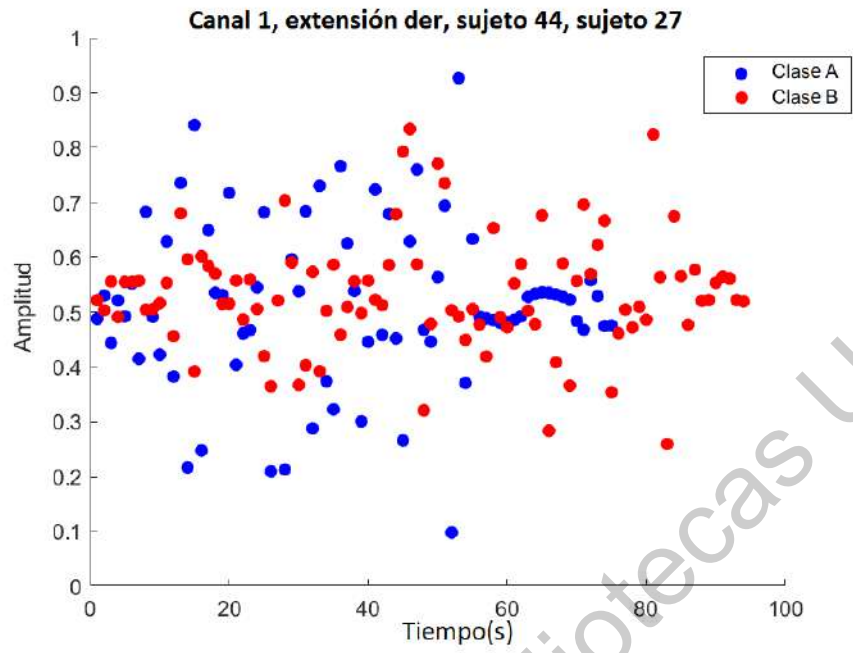
(b)

Fig. 4.27: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 3: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 3.

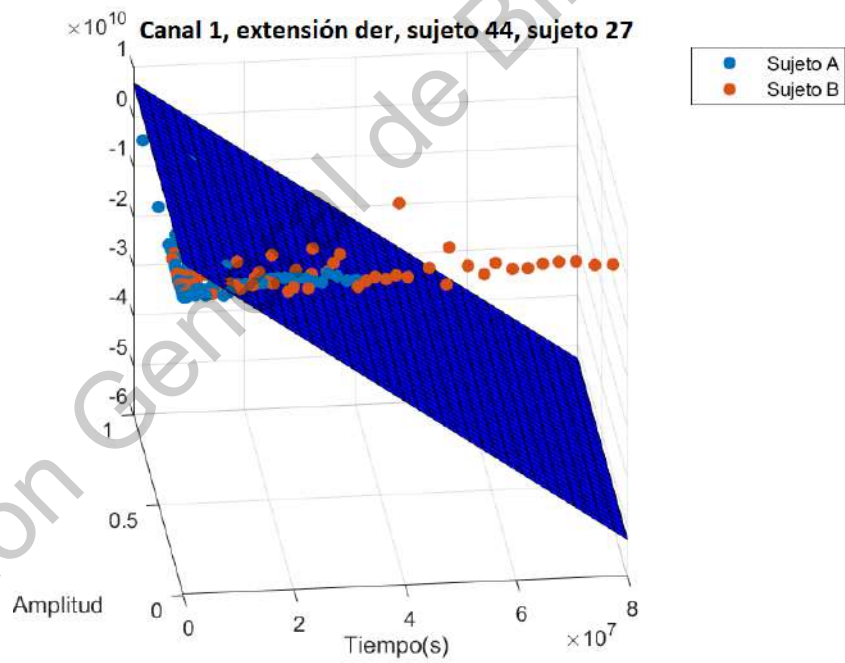
4.2.3. *Kernel polinomial de orden 4*

A continuación, se visualizan las gráficas correspondientes para el comportamiento de el Kernel polinomial de orden 4, para la obtención de estas gráficas se utilizó el movimiento de extensión derecha con la finalidad de estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

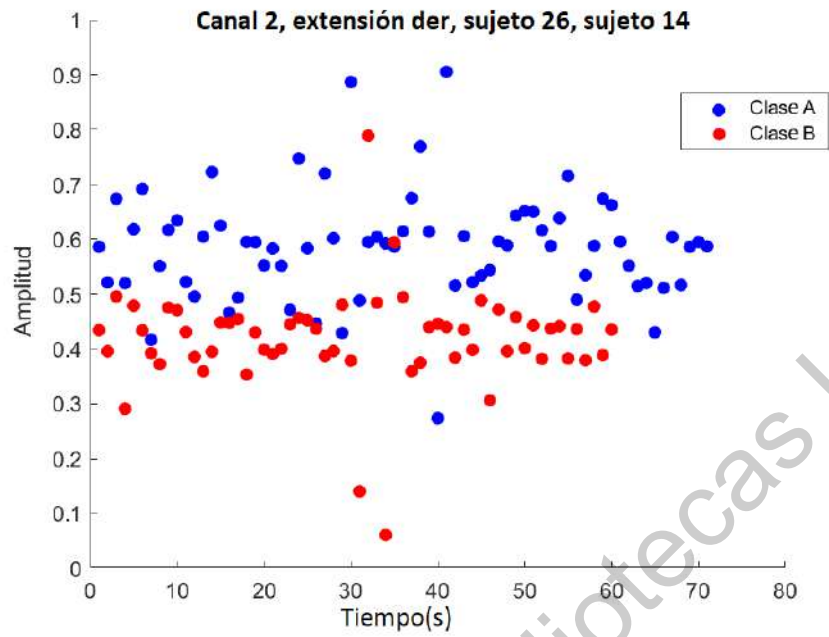


(a)

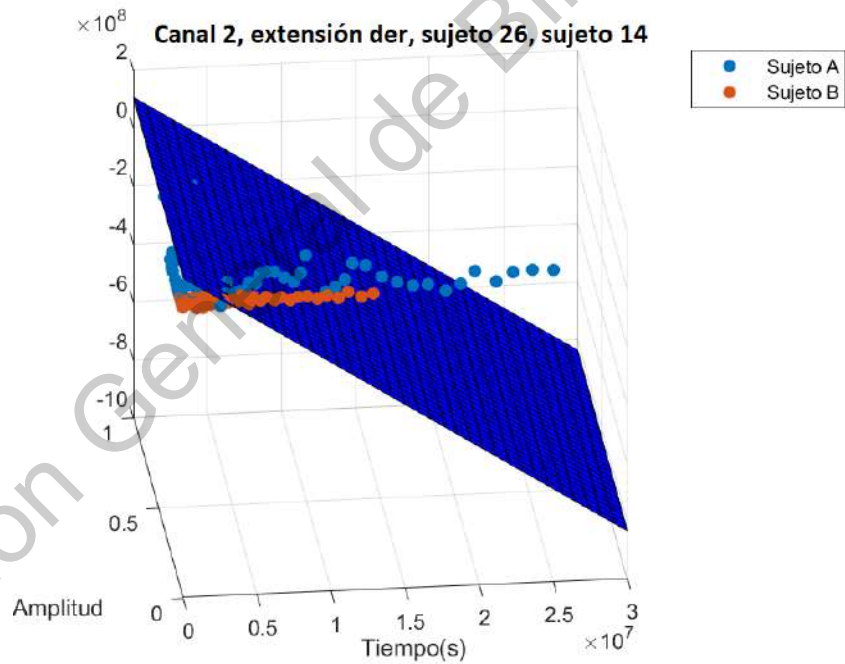


(b)

Fig. 4.28: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

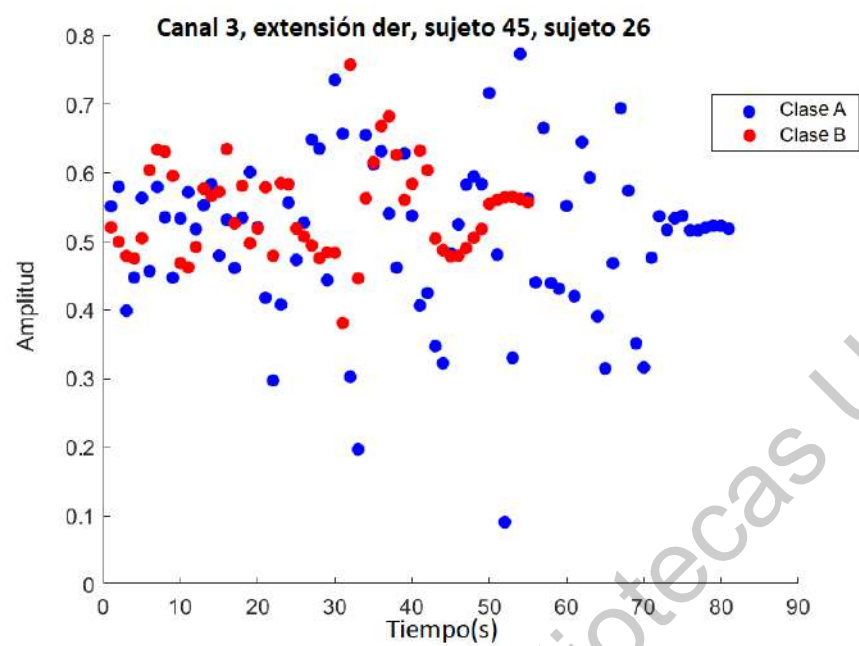


(a)

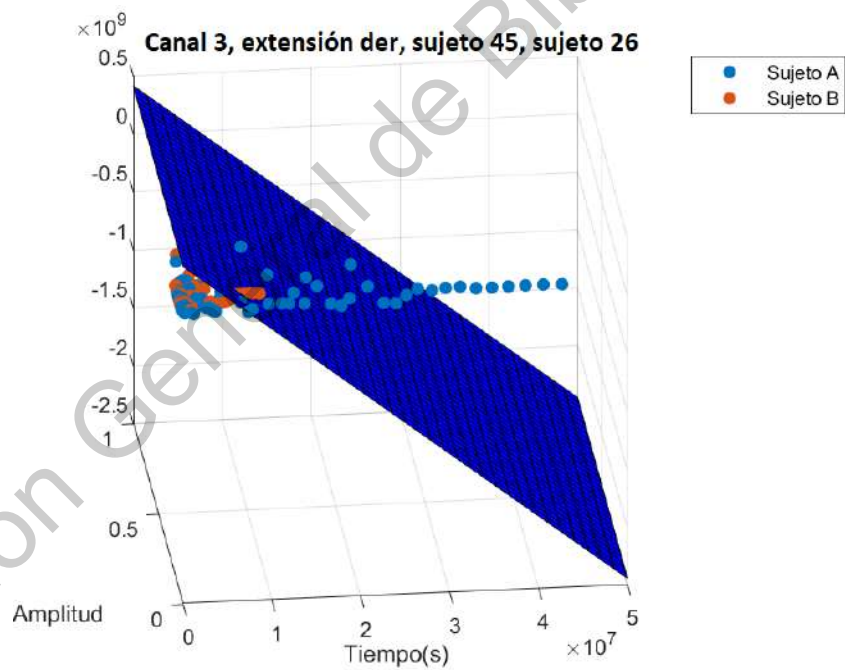


(b)

Fig. 4.29: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

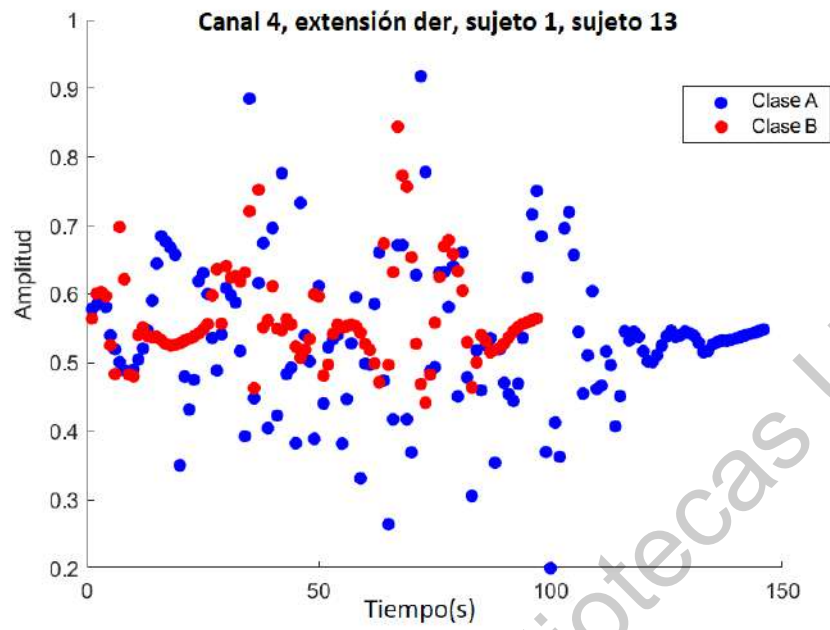


(a)

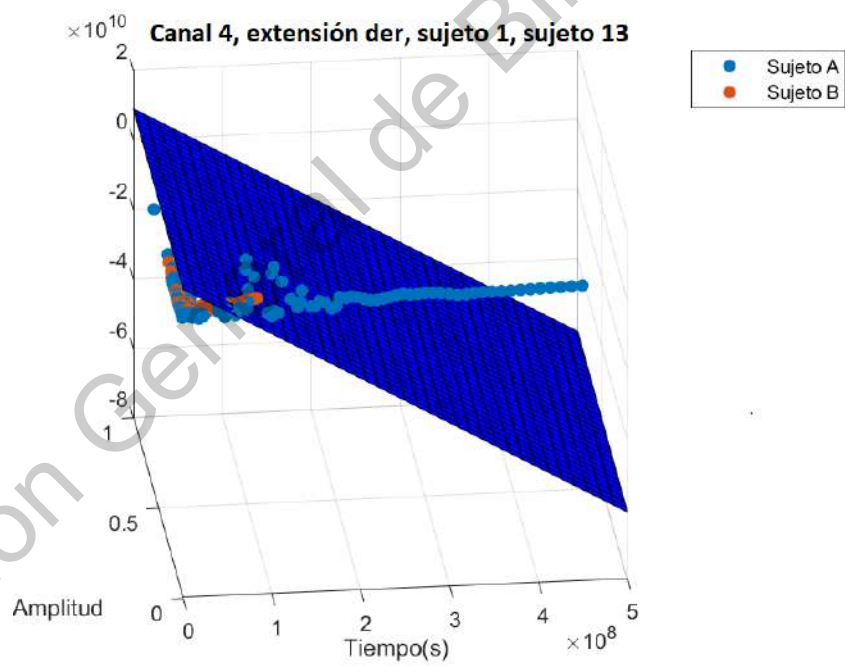


(b)

Fig. 4.30: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

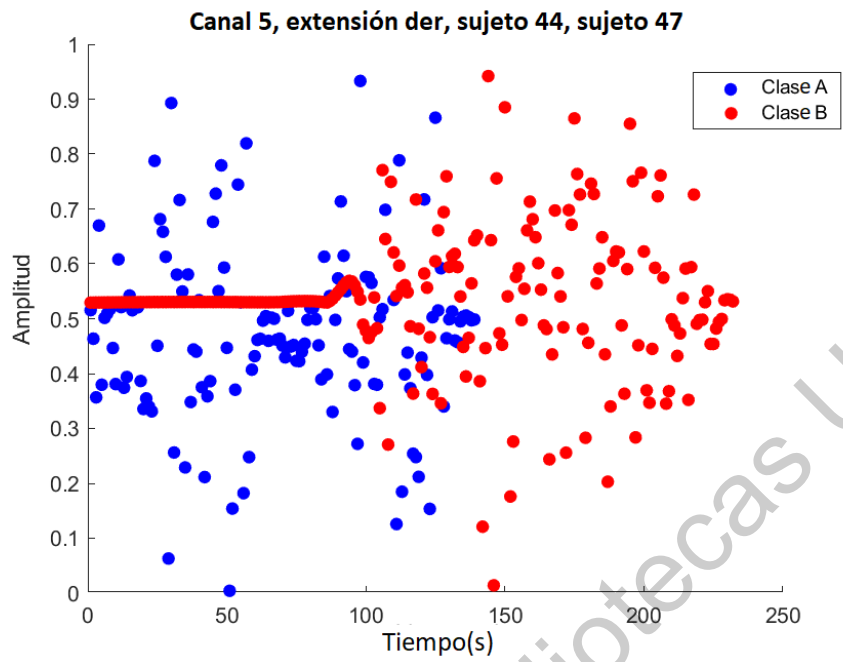


(a)

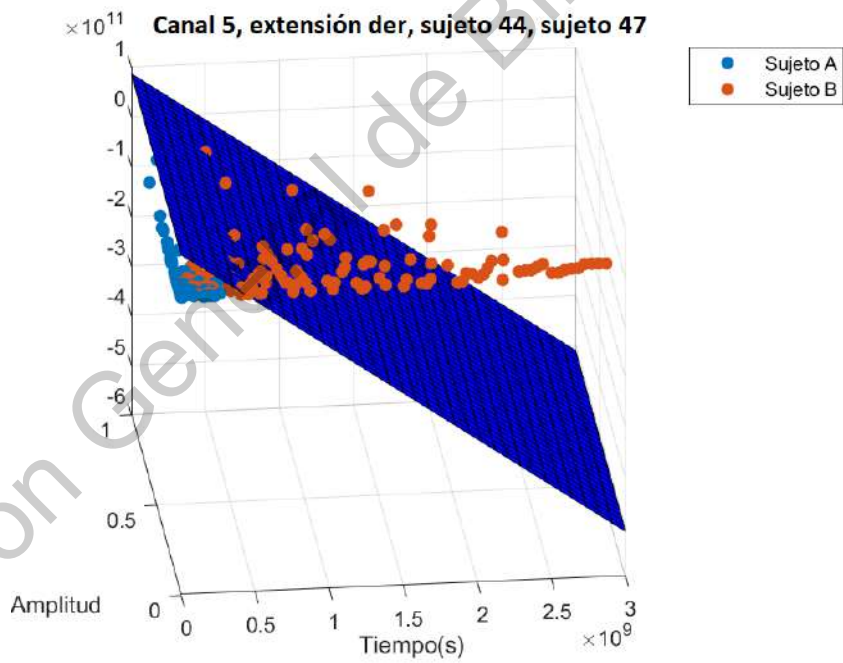


(b)

Fig. 4.31: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

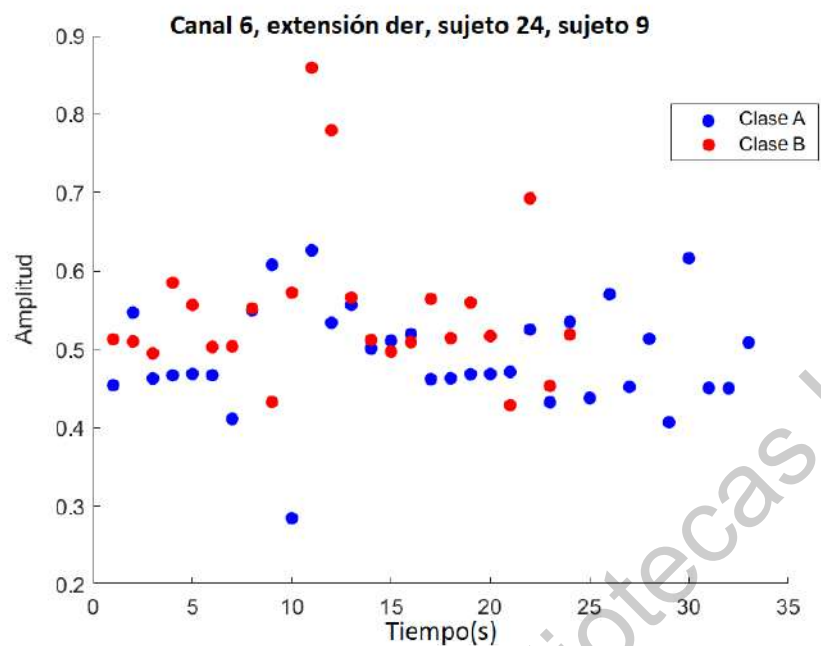


(a)

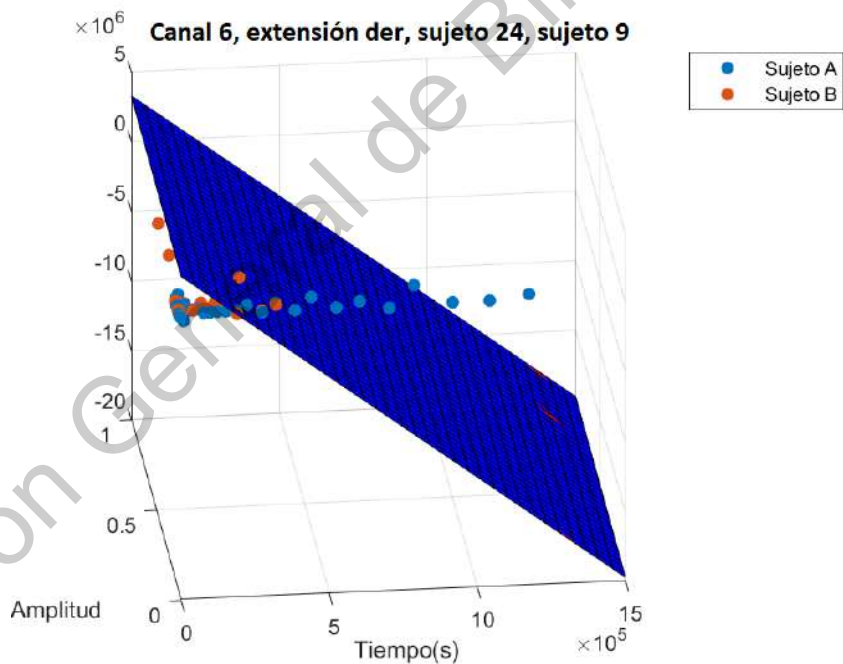


(b)

Fig. 4.32: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

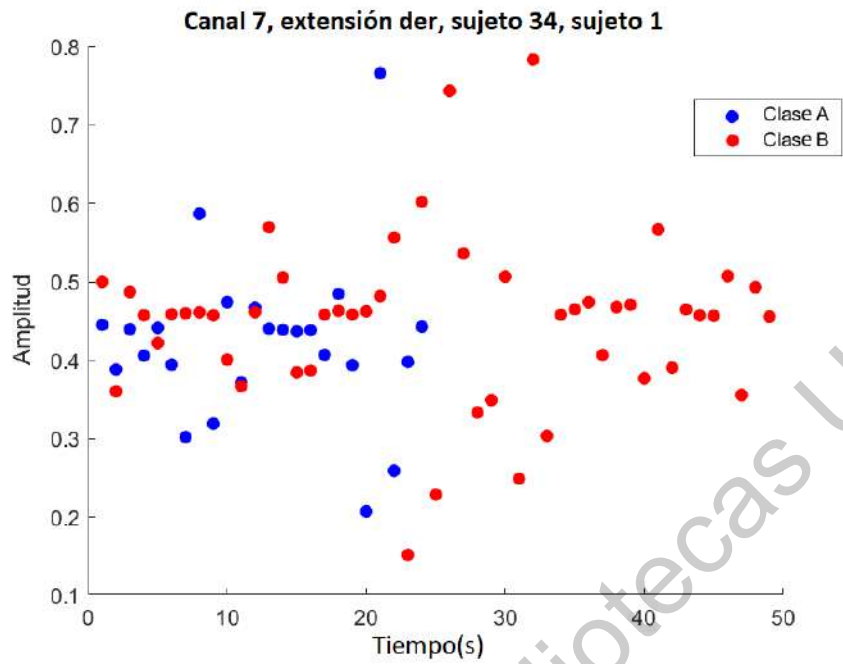


(a)

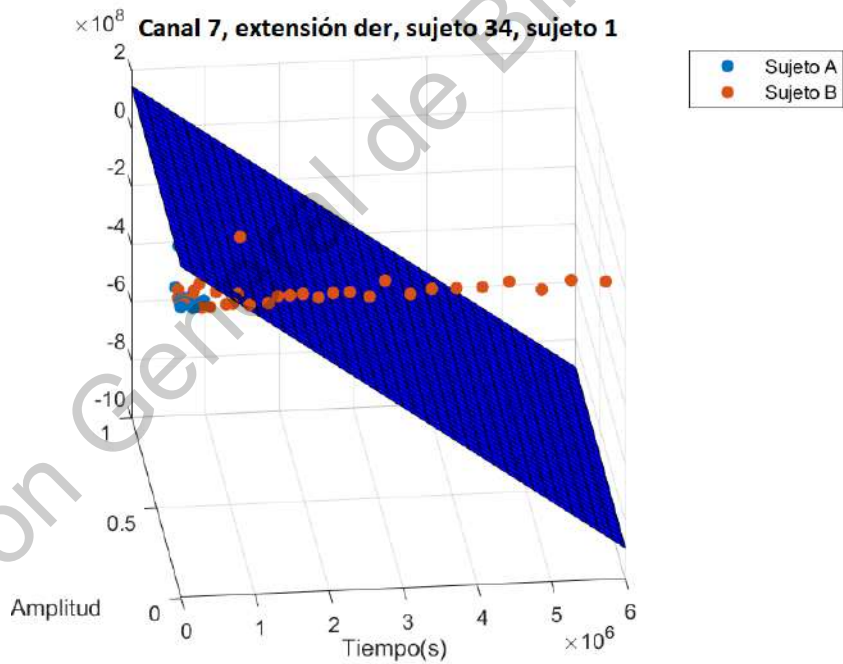


(b)

Fig. 4.33: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

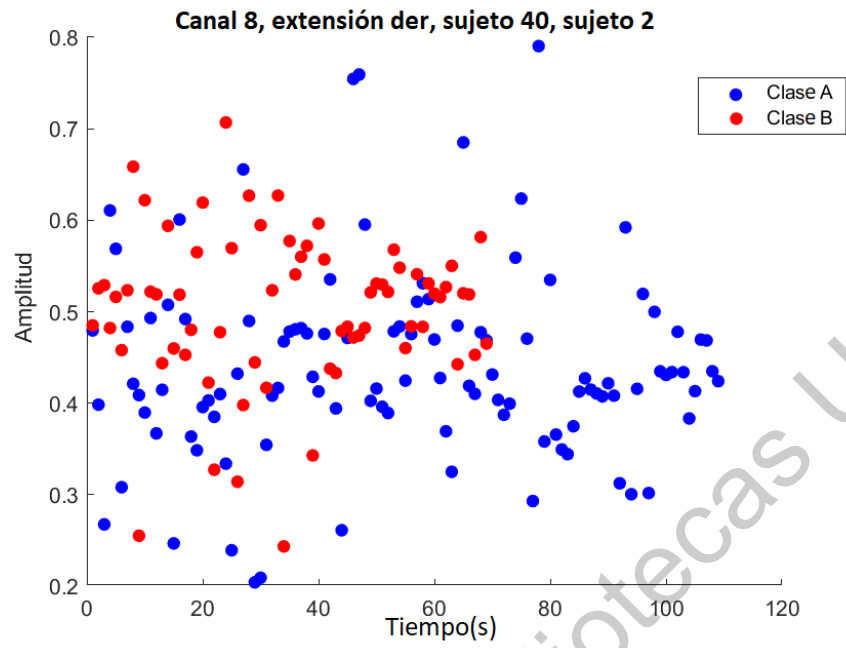


(a)

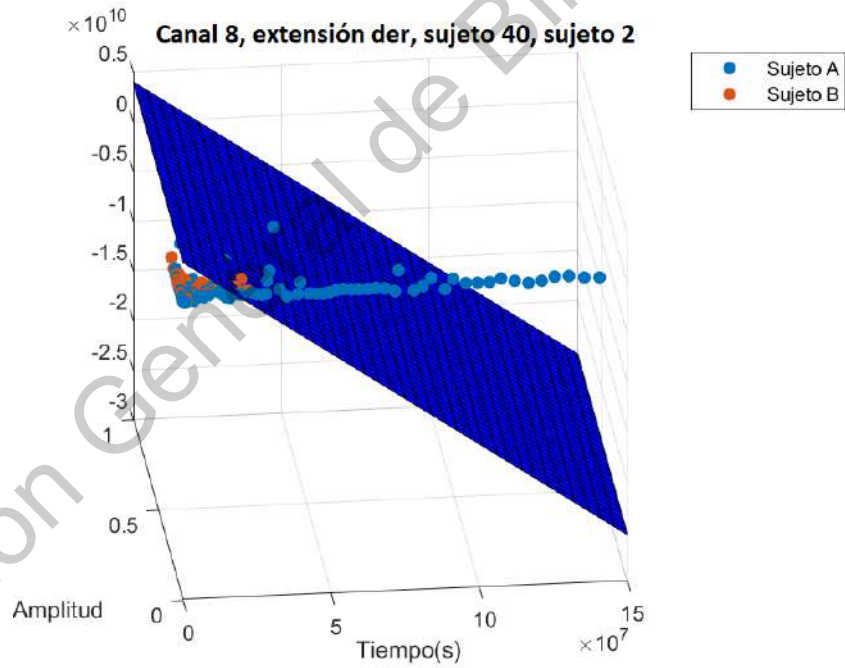


(b)

Fig. 4.34: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.



(a)



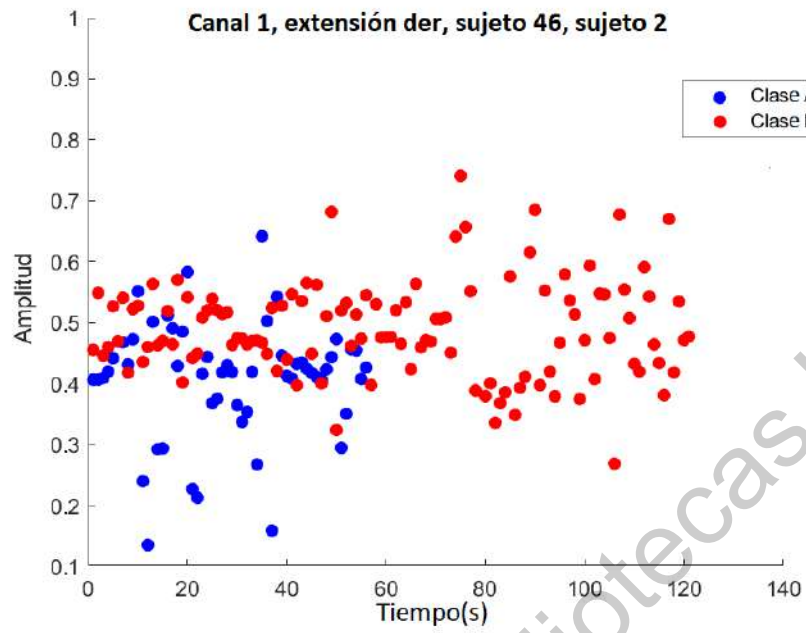
(b)

Fig. 4.35: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 4: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 4.

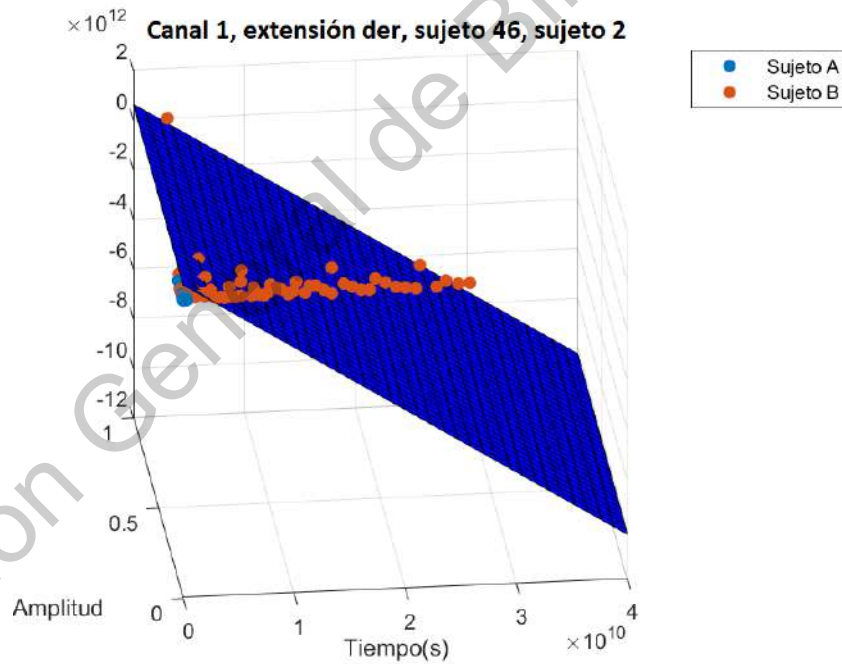
4.2.4. *Kernel polinomial de orden 5*

A continuación, se visualizan las gráficas correspondientes para el comportamiento de el Kernel polinomial de orden 5, para la obtención de estas gráficas se utilizó el movimiento de extensión derecha con la finalidad de estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

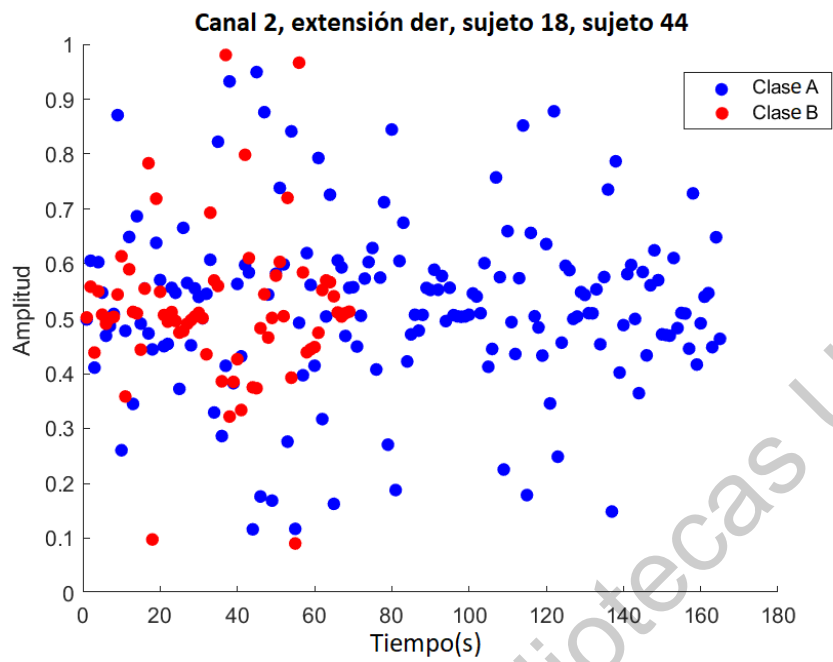


(a)

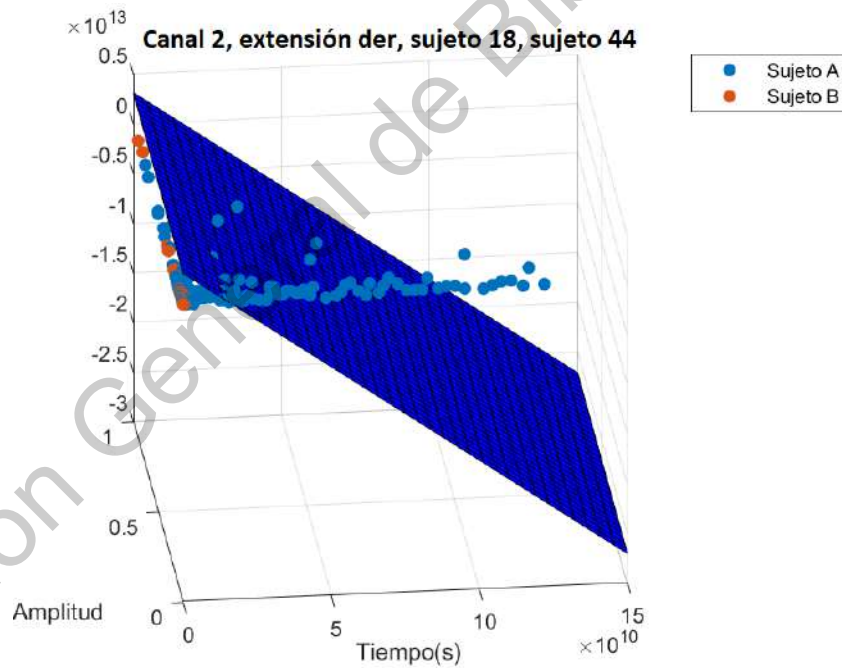


(b)

Fig. 4.36: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

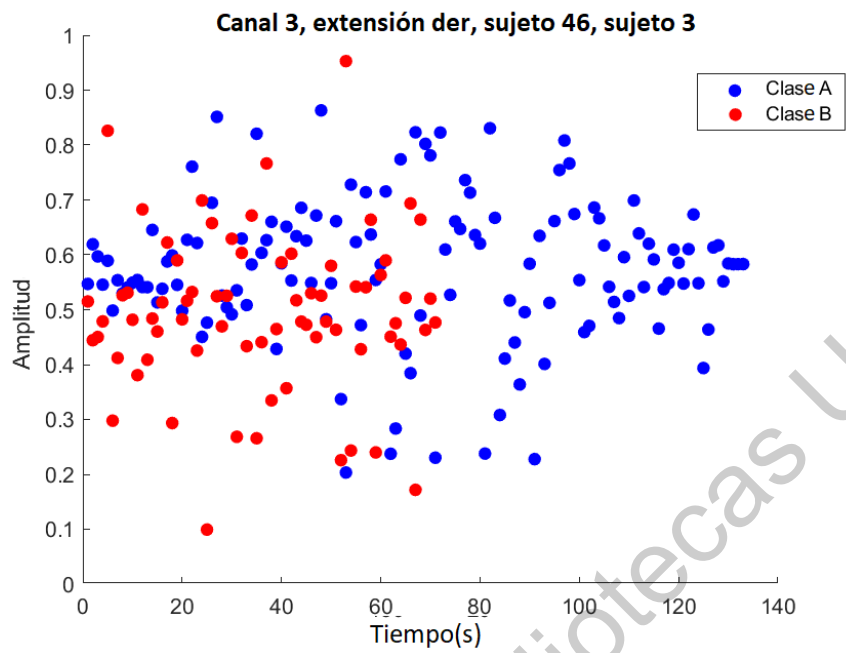


(a)

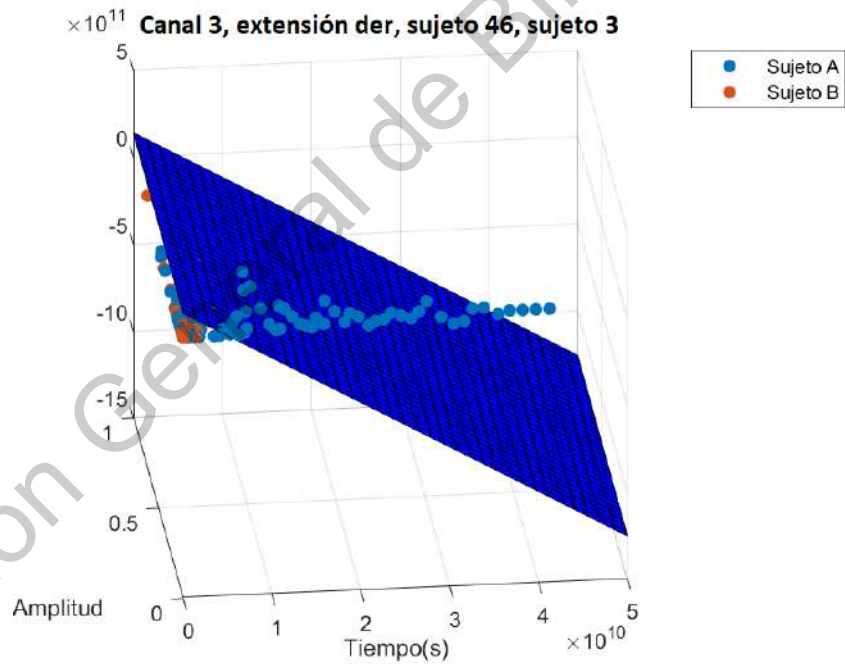


(b)

Fig. 4.37: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

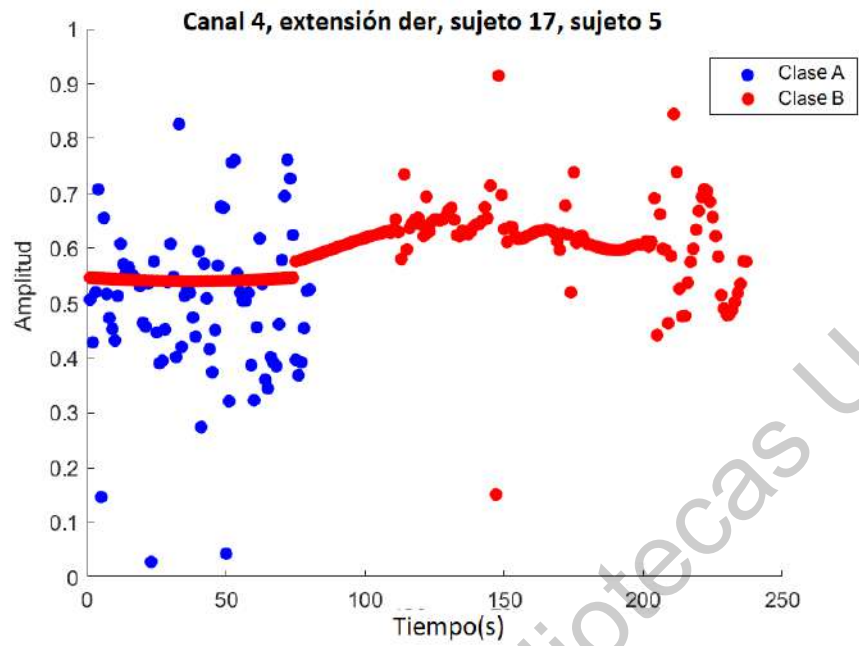


(a)

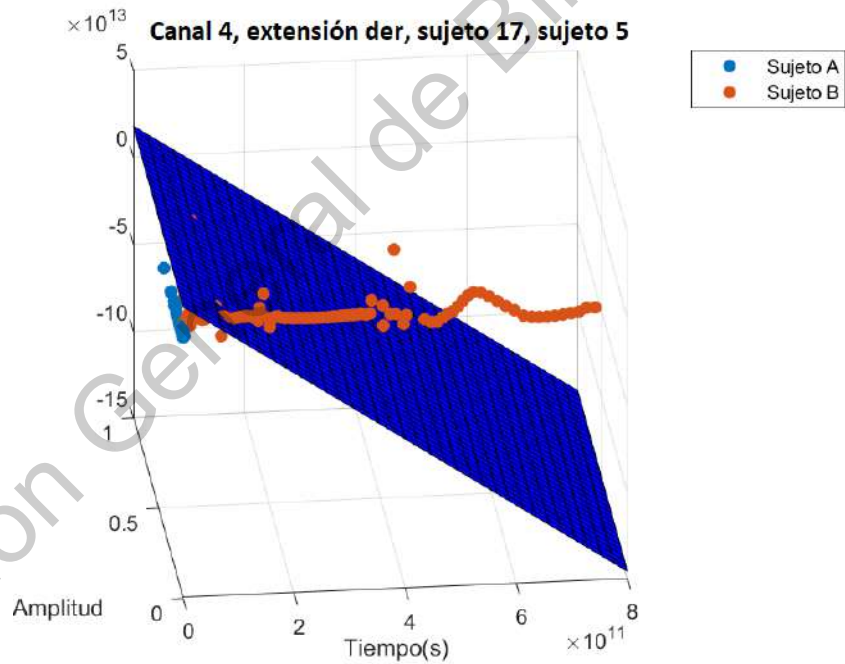


(b)

Fig. 4.38: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

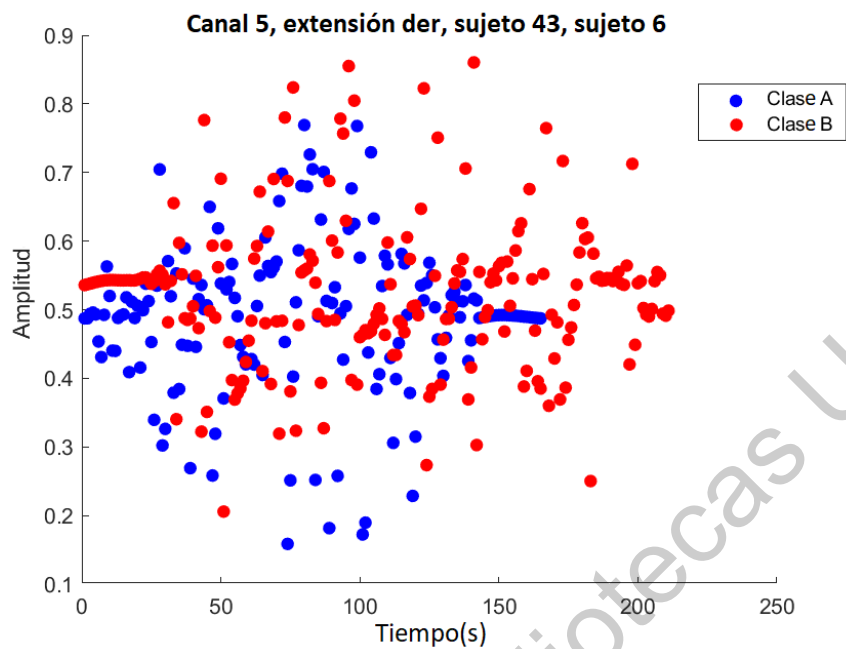


(a)

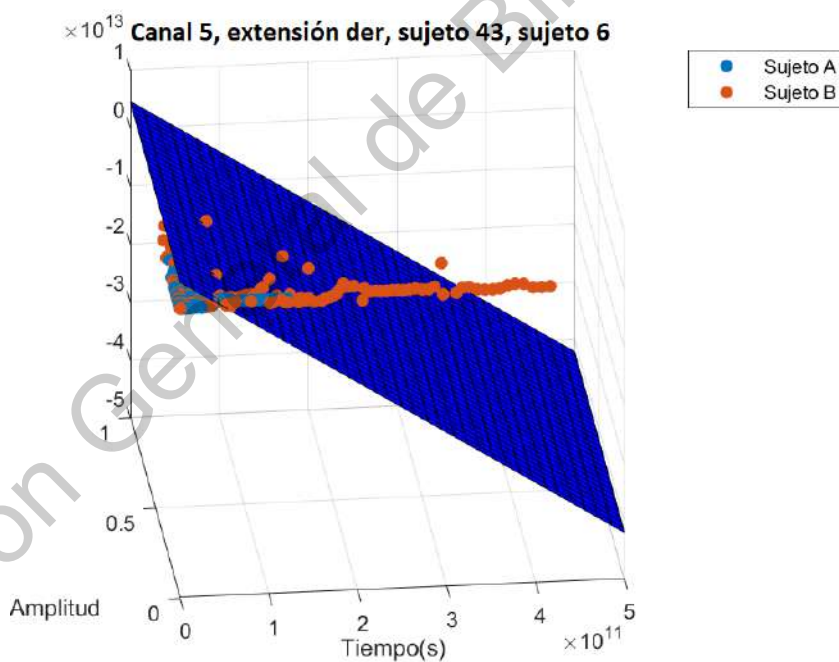


(b)

Fig. 4.39: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

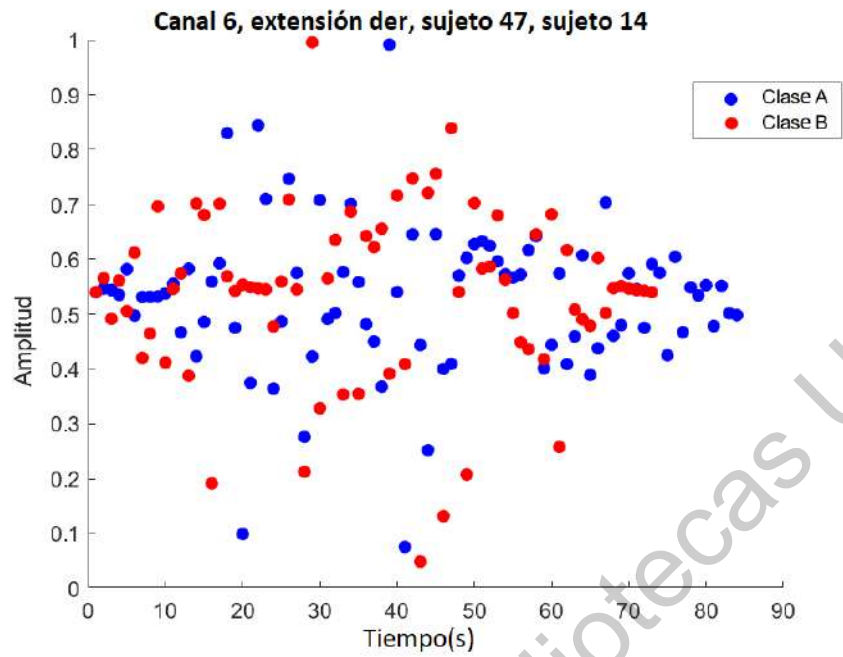


(a)

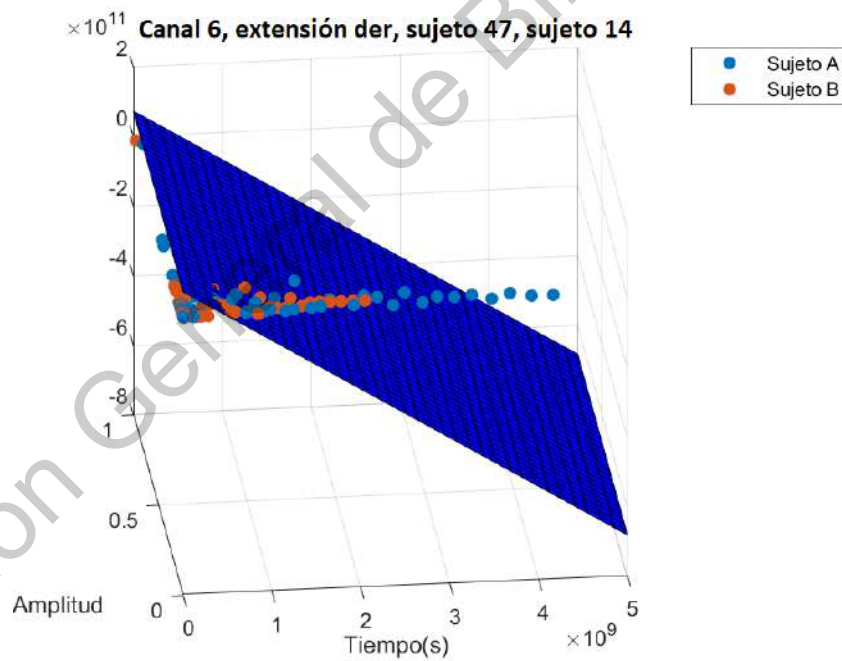


(b)

Fig. 4.40: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

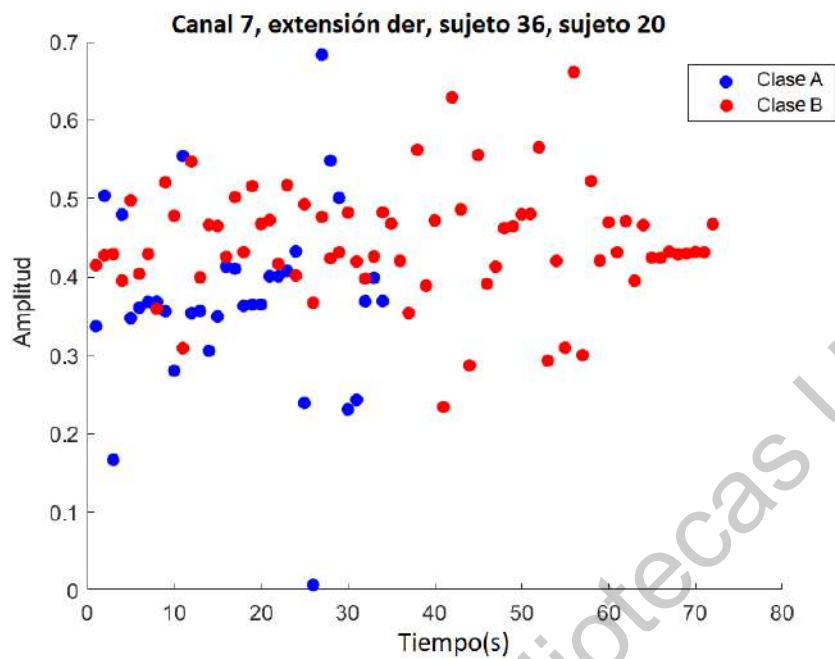


(a)

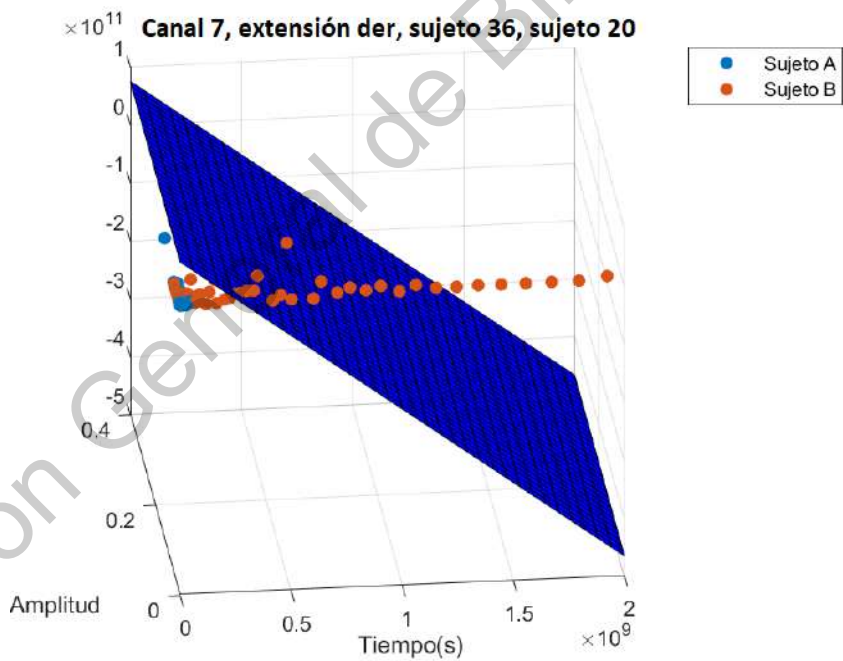


(b)

Fig. 4.41: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

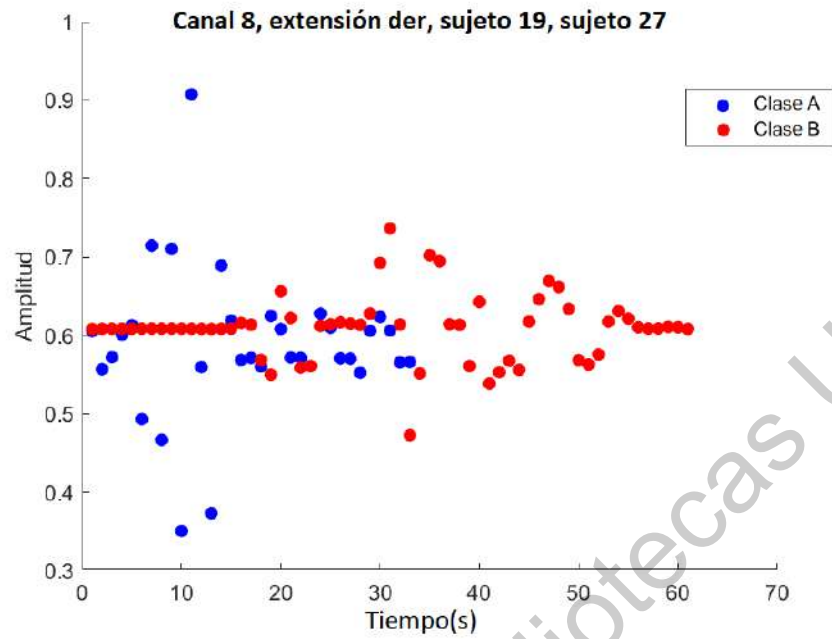


(a)

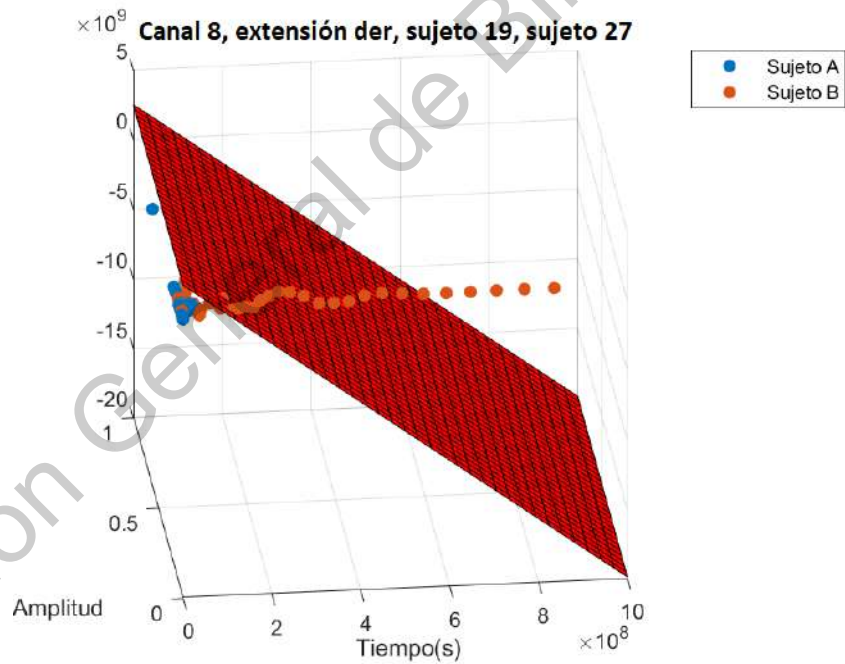


(b)

Fig. 4.42: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.



(a)



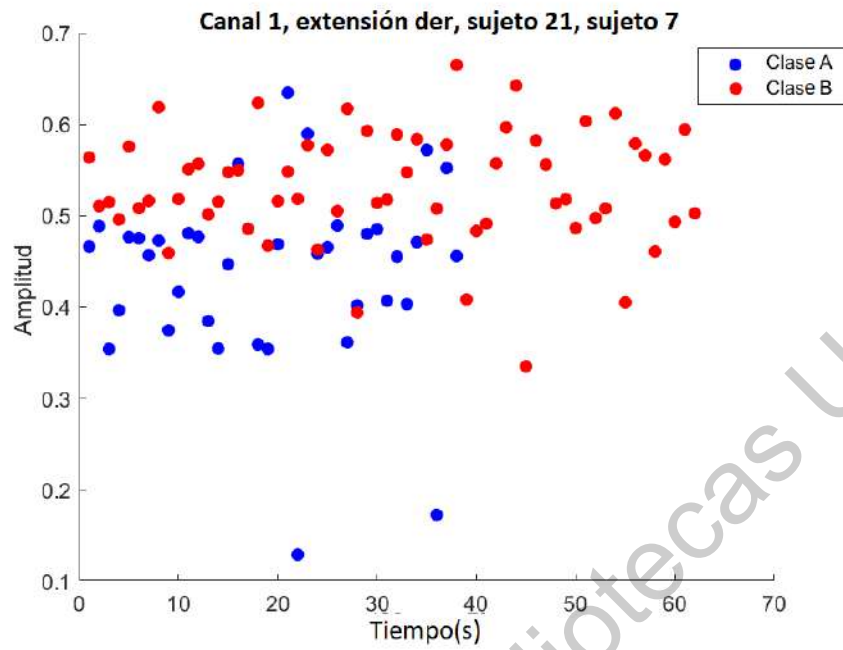
(b)

Fig. 4.43: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 5: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 5.

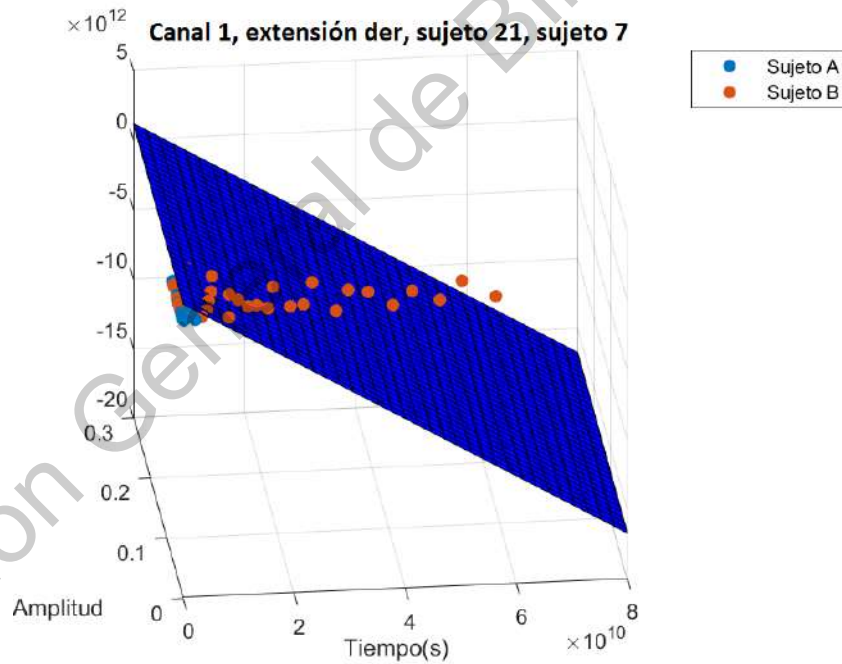
4.2.5. *Kernel polinomial de orden 6*

A continuación, se visualizan las gráficas correspondientes para el comportamiento de el Kernel polinomial de orden 6, para la obtención de estas gráficas se utilizó el movimiento de extensión derecha con la finalidad de estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

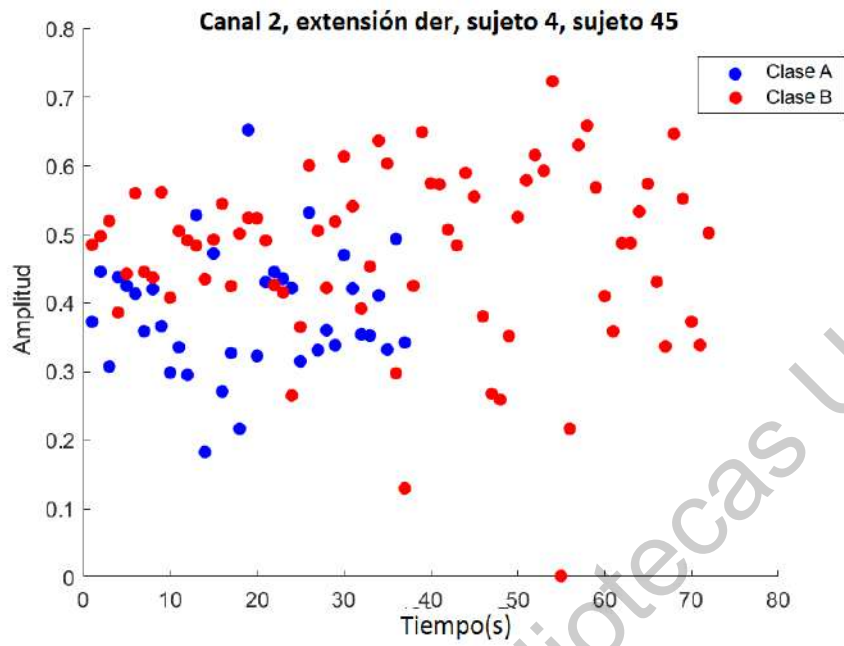


(a)

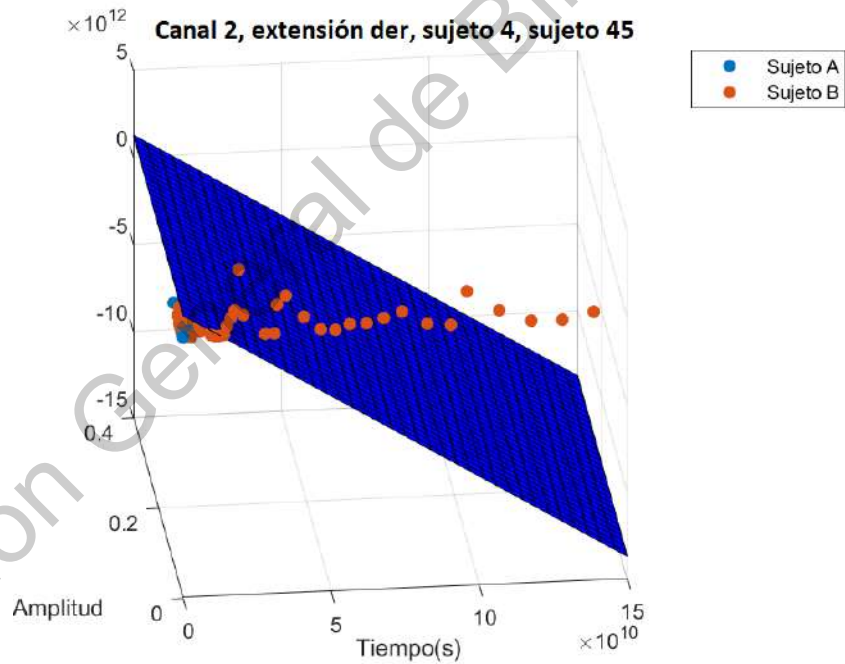


(b)

Fig. 4.44: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

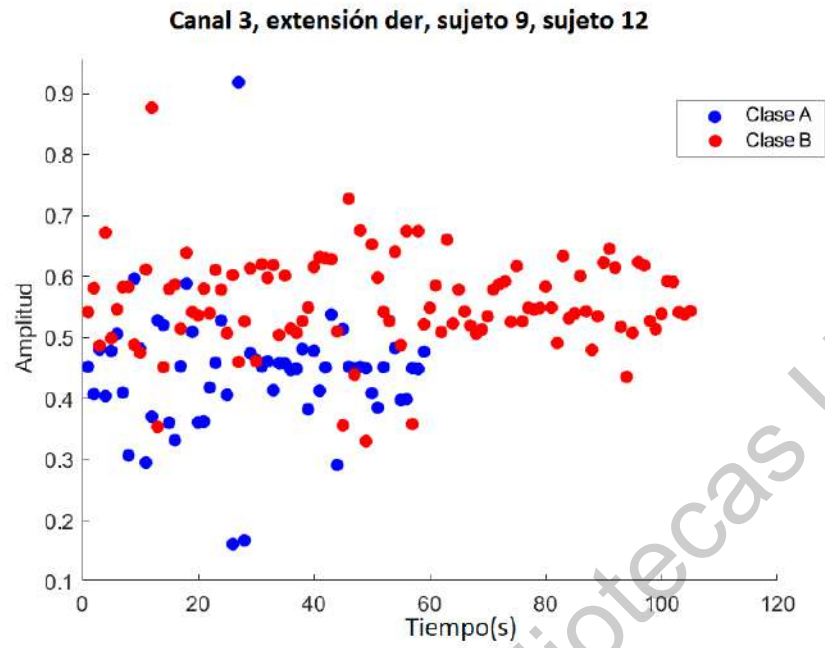


(a)

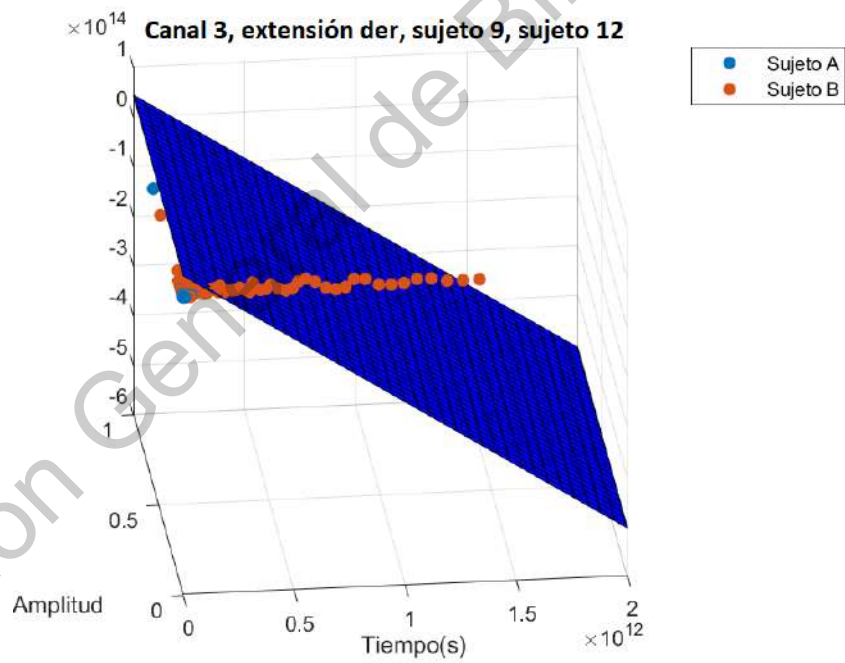


(b)

Fig. 4.45: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

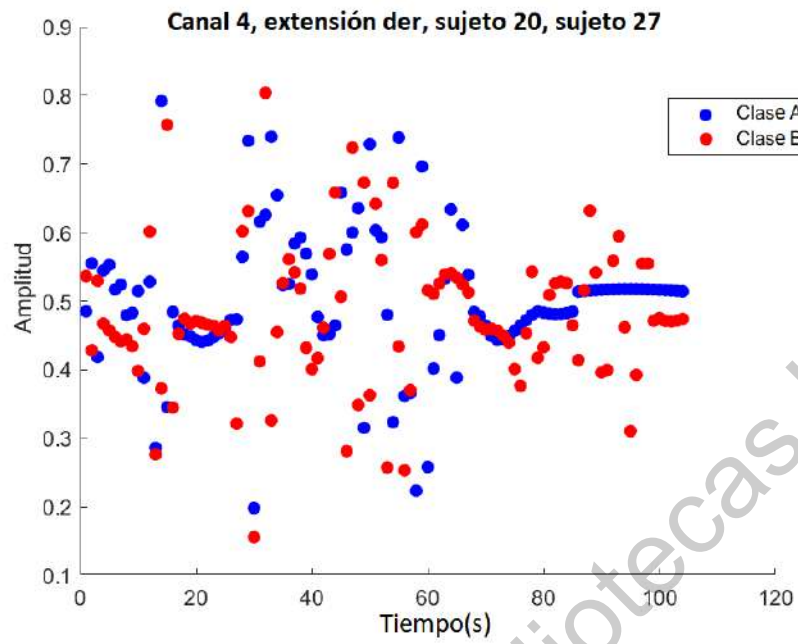


(a)

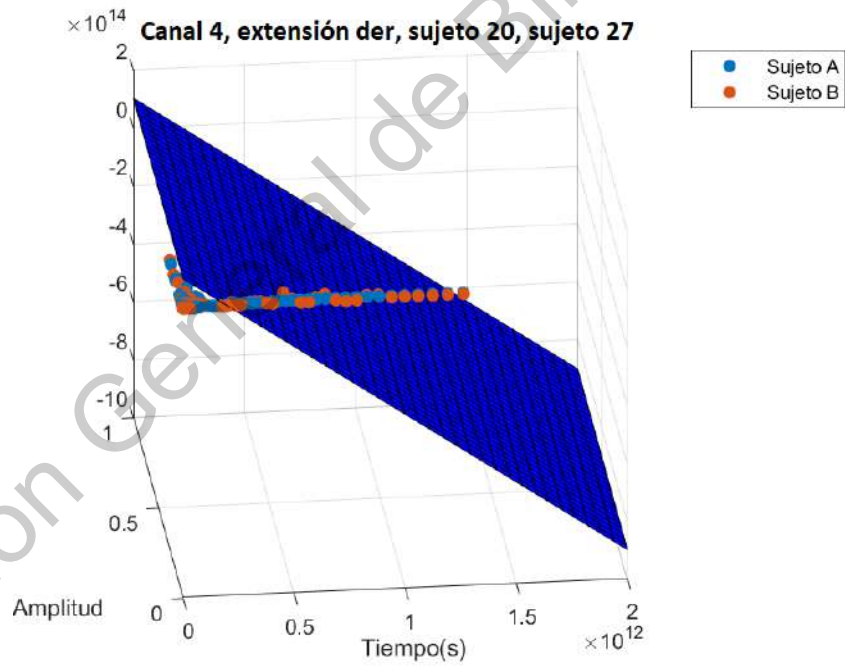


(b)

Fig. 4.46: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

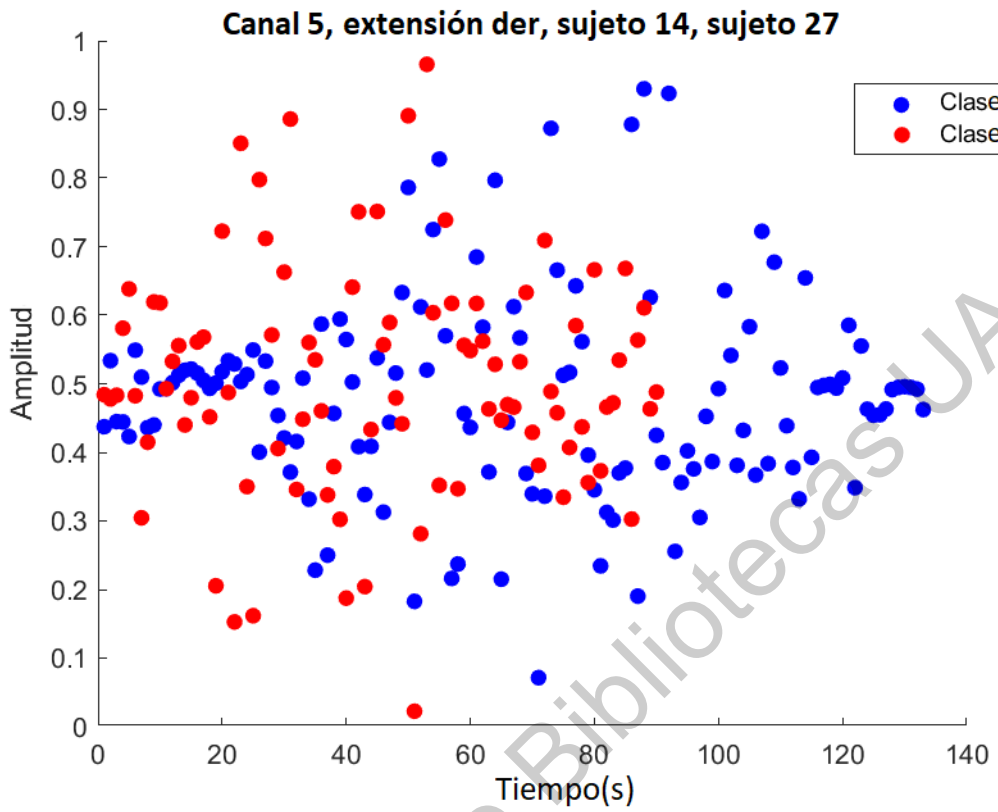


(a)

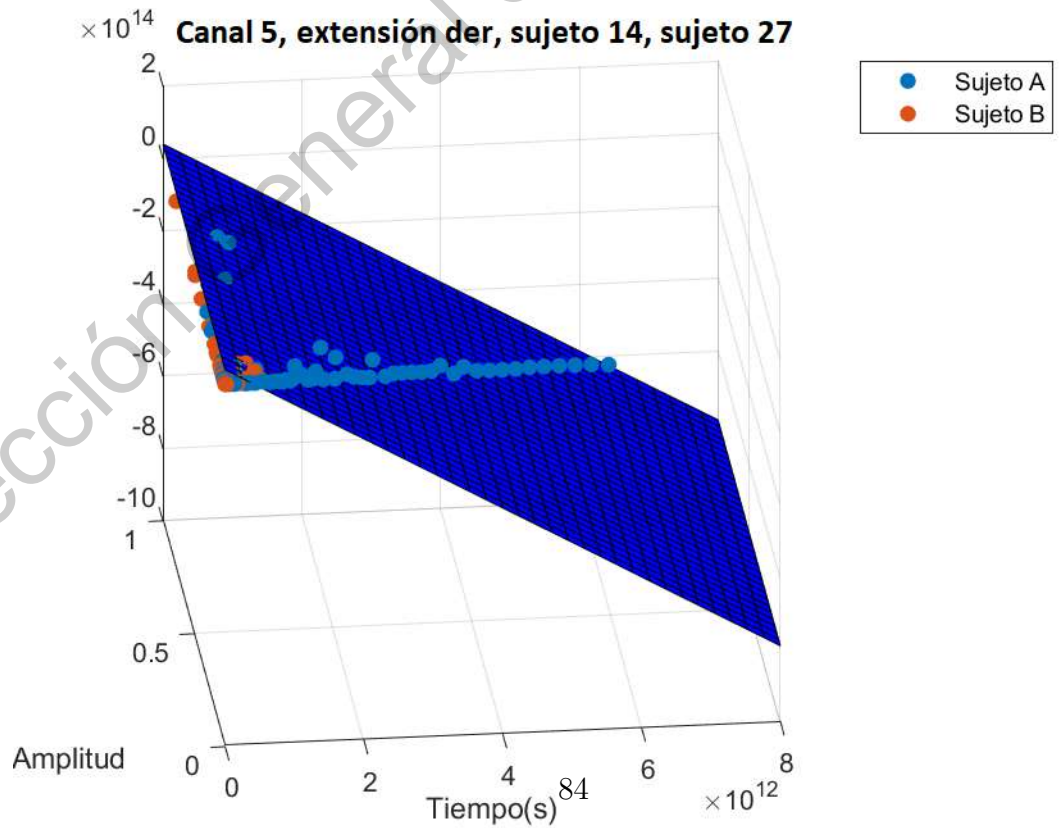


(b)

Fig. 4.47: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

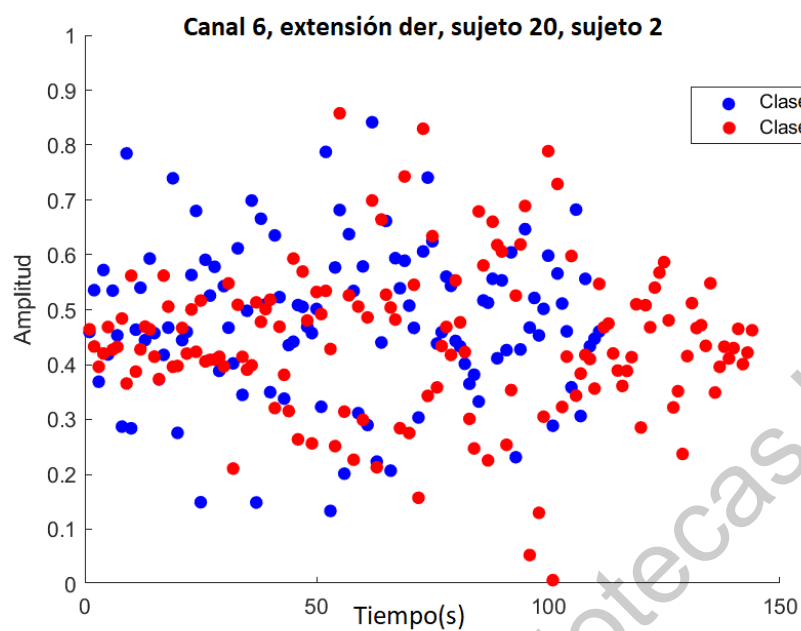


(a)

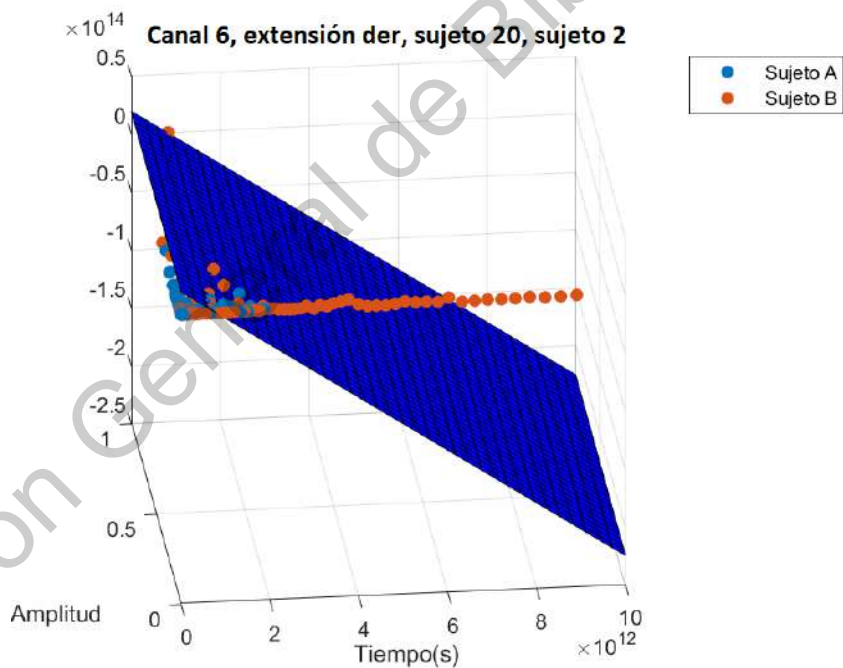


(b)

Fig. 4.48: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

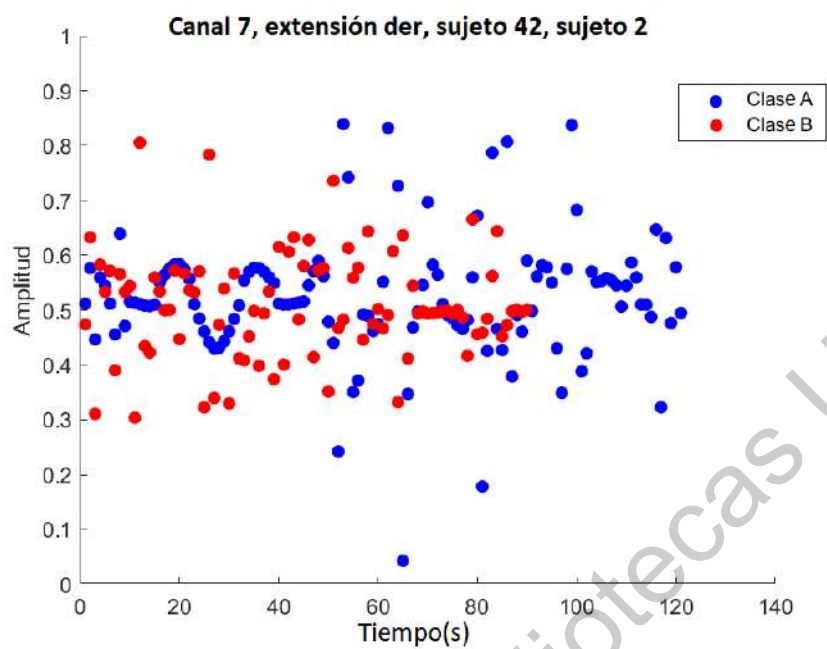


(a)

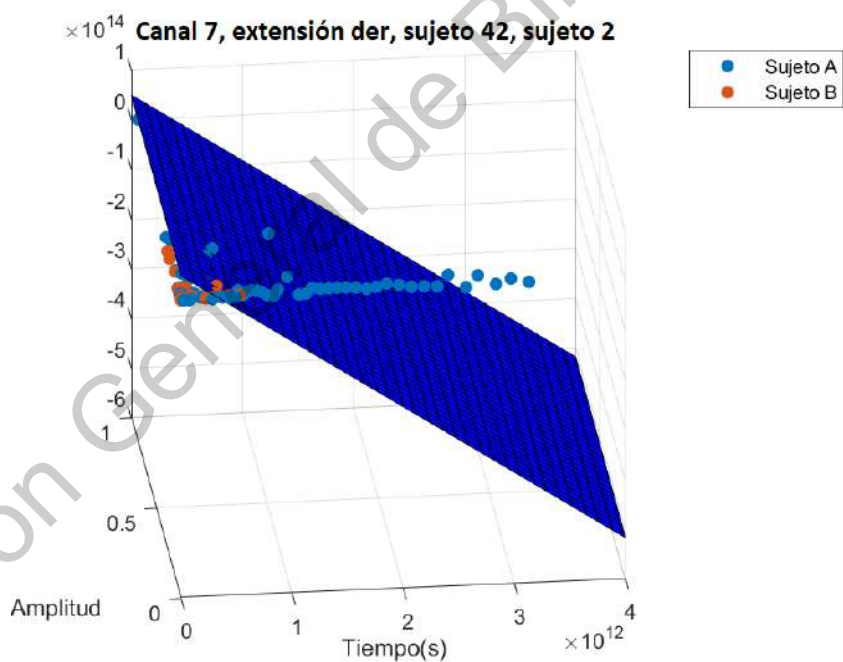


(b)

Fig. 4.49: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

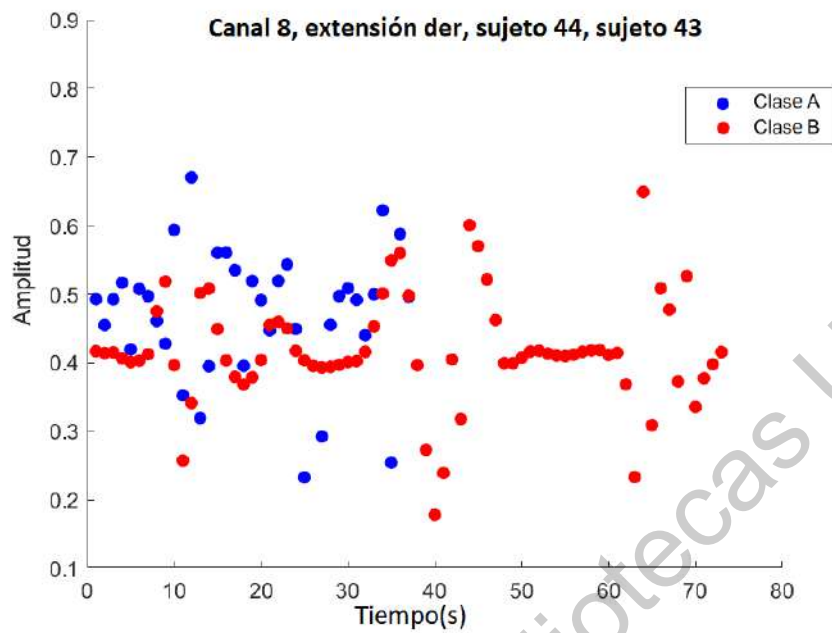


(a)

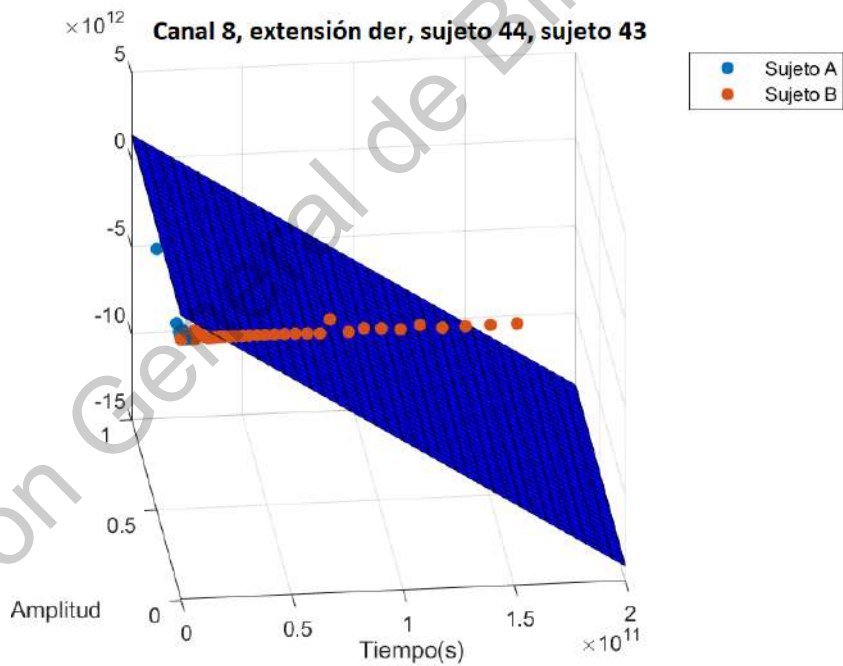


(b)

Fig. 4.50: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.



(a)



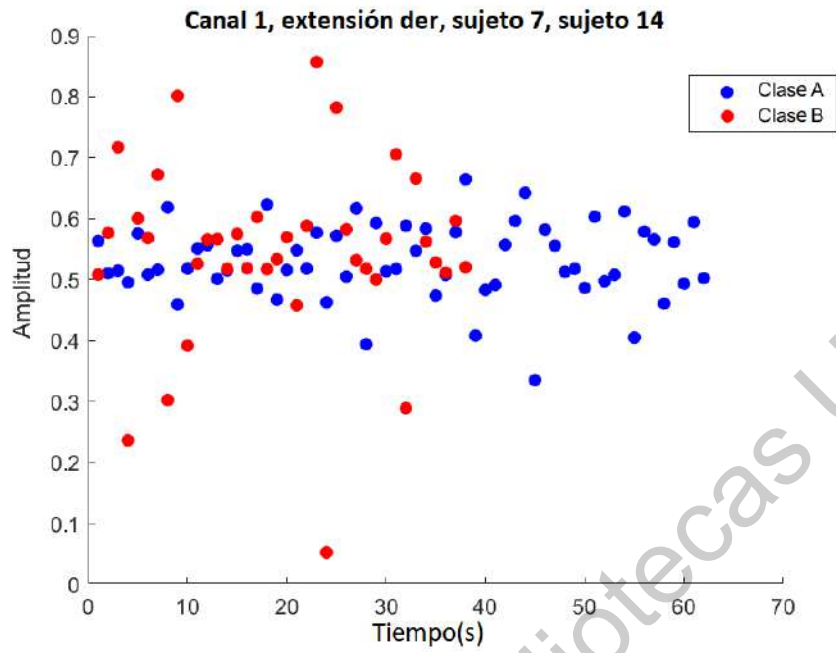
(b)

Fig. 4.51: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 6: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 6.

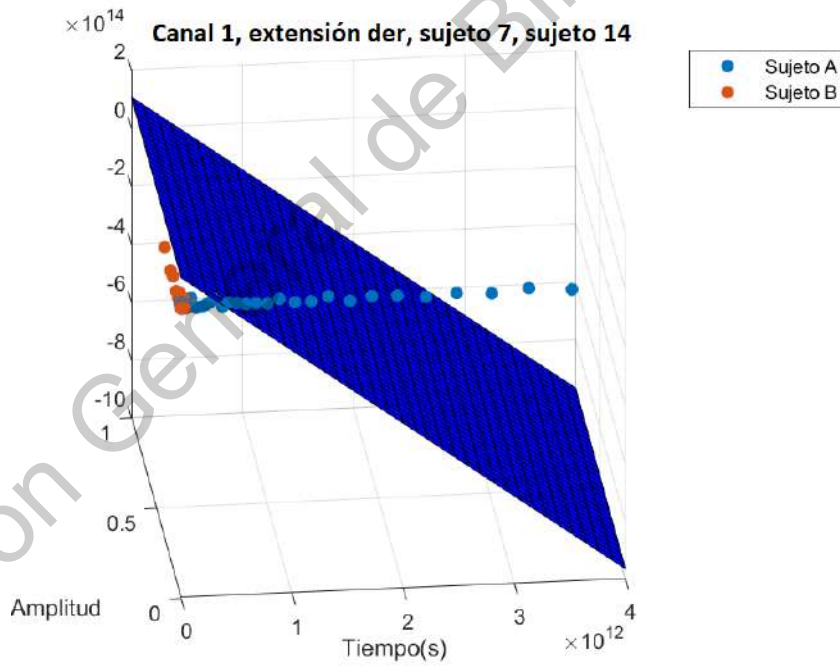
4.2.6. *Kernel polinomial de orden 7*

A continuación, se visualizan las gráficas correspondientes para el comportamiento de el Kernel polinomial de orden 7, para la obtención de estas gráficas se utilizó el movimiento de extensión derecha con la finalidad de estandarizar los resultados obtenidos.

Dirección General de Bibliotecas UAQ

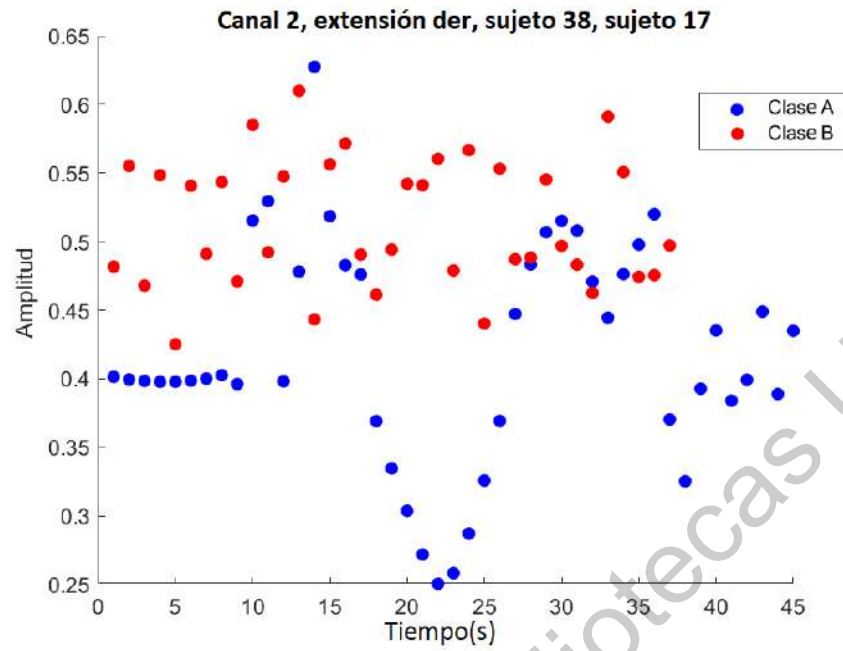


(a)

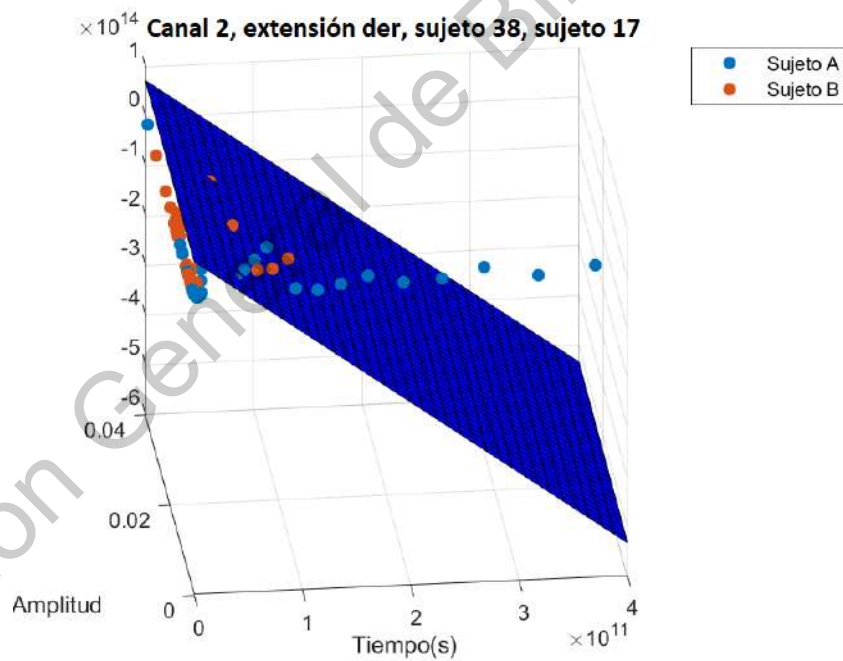


(b)

Fig. 4.52: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

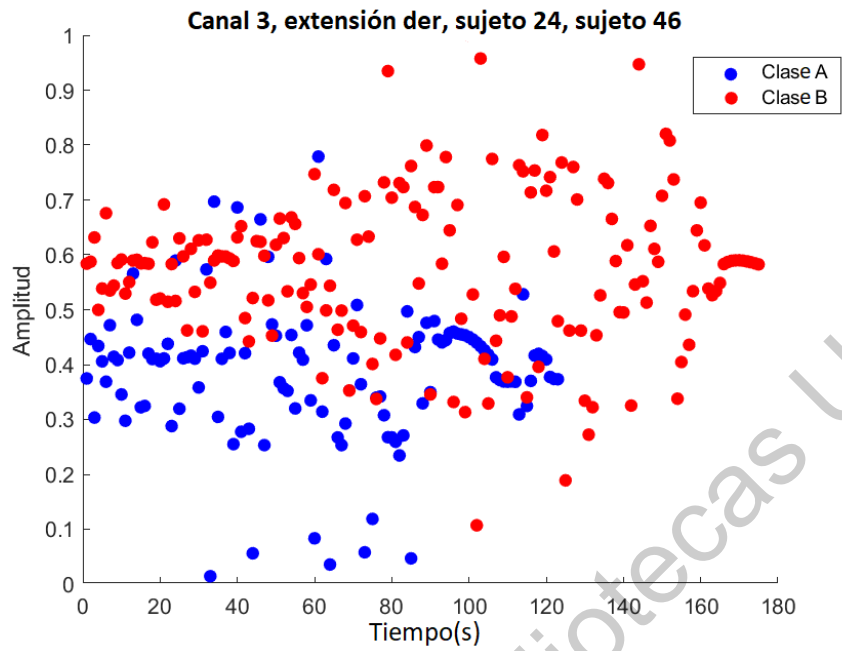


(a)

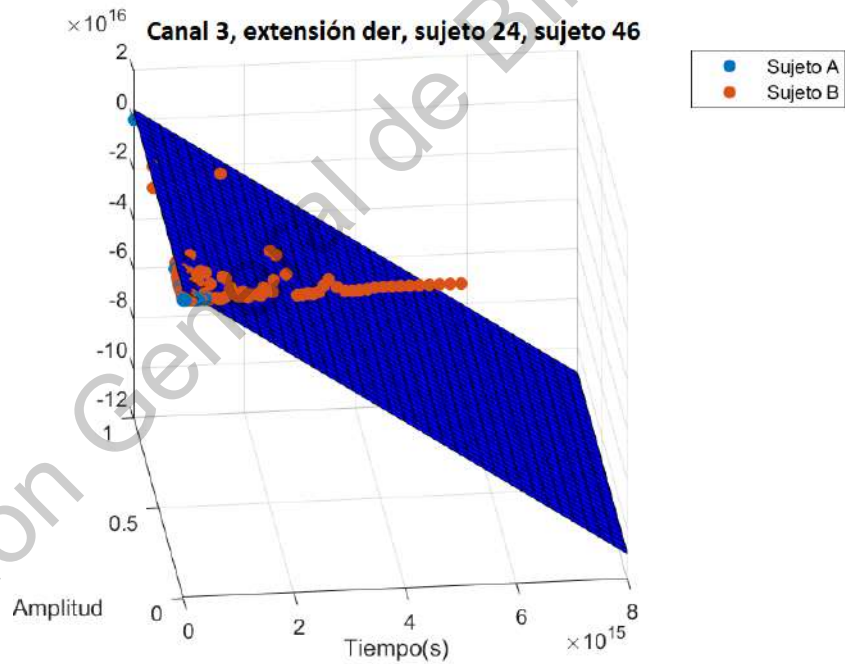


(b)

Fig. 4.53: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

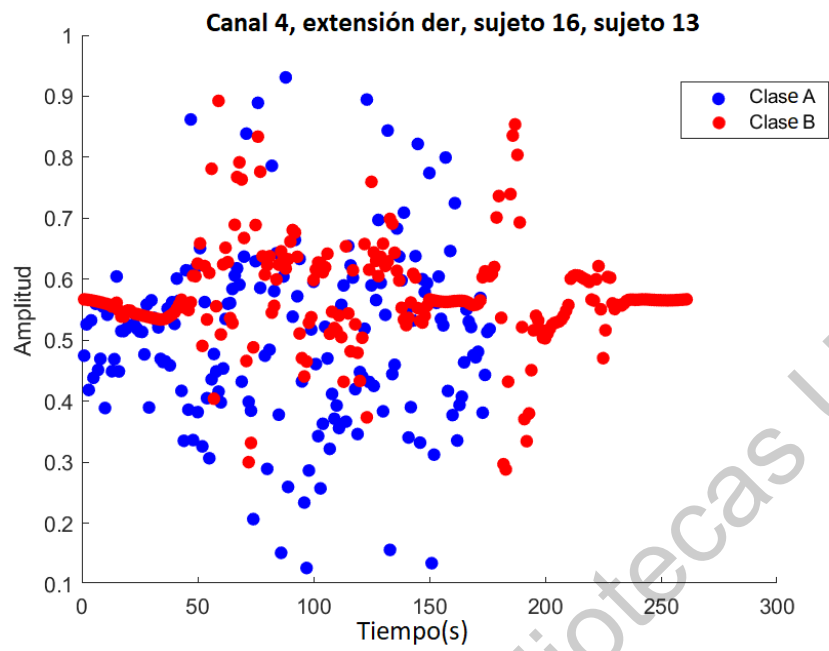


(a)

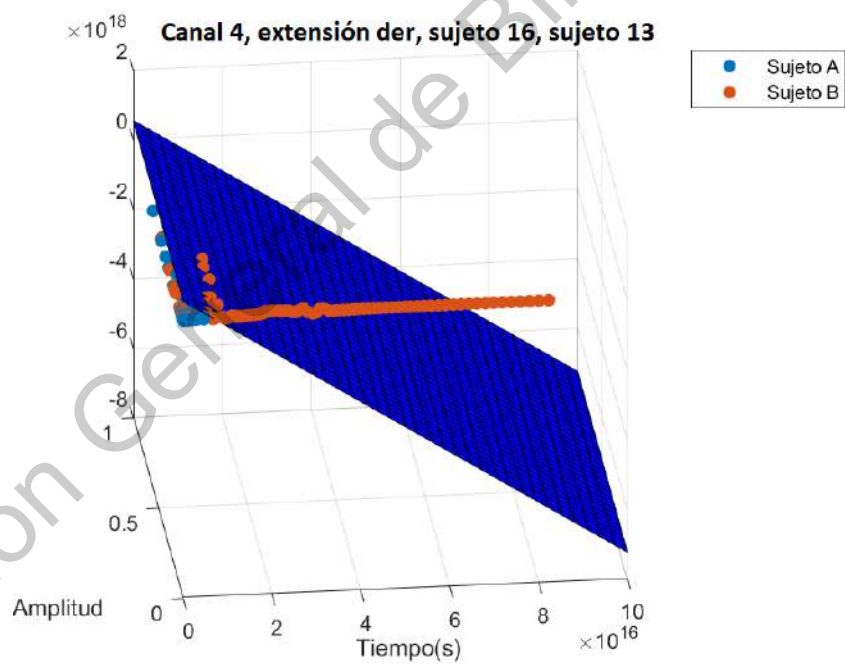


(b)

Fig. 4.54: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

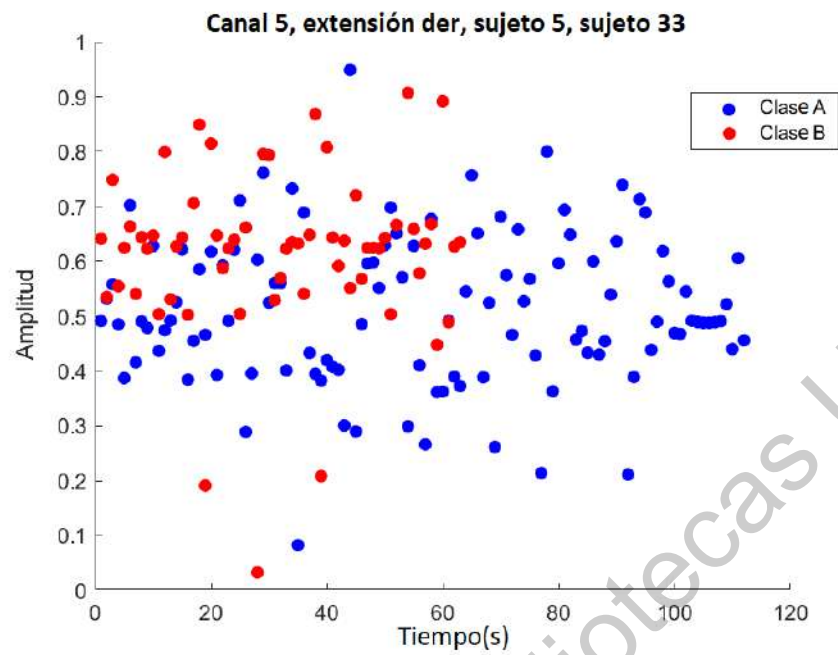


(a)

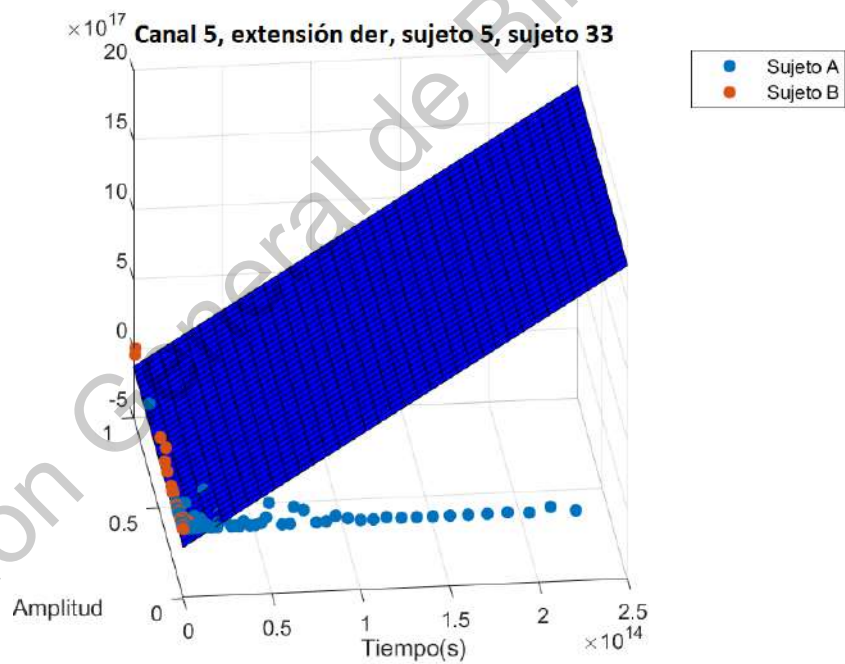


(b)

Fig. 4.55: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

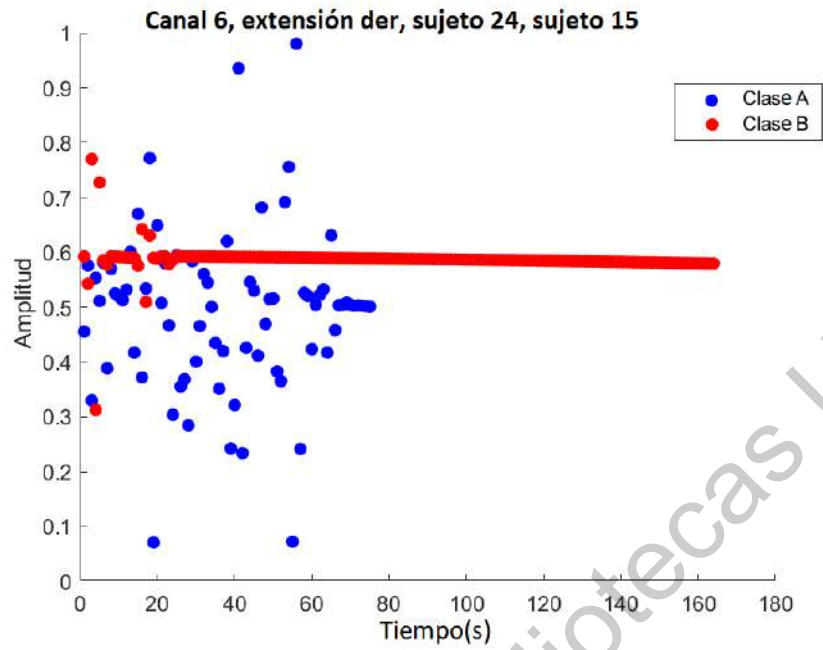


(a)

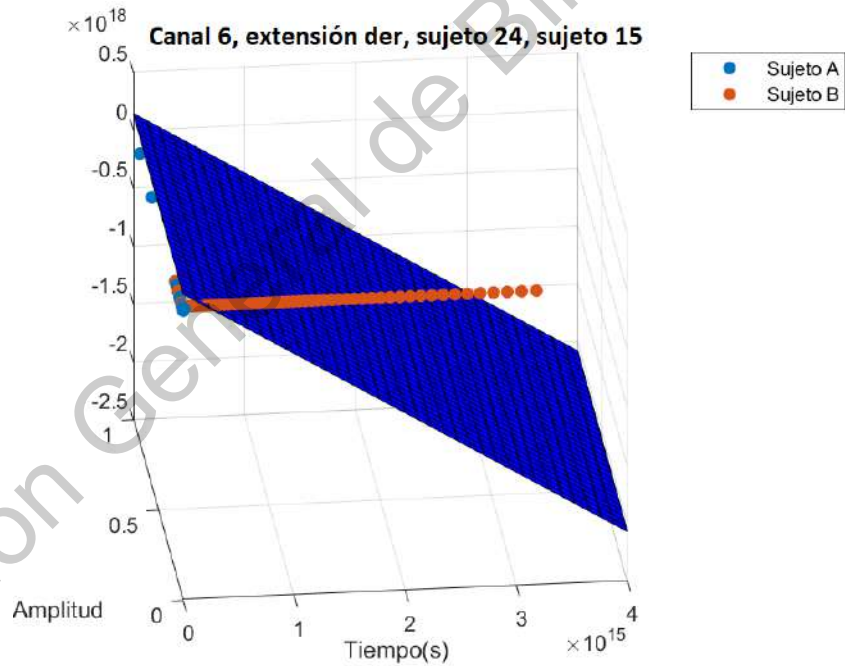


(b)

Fig. 4.56: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

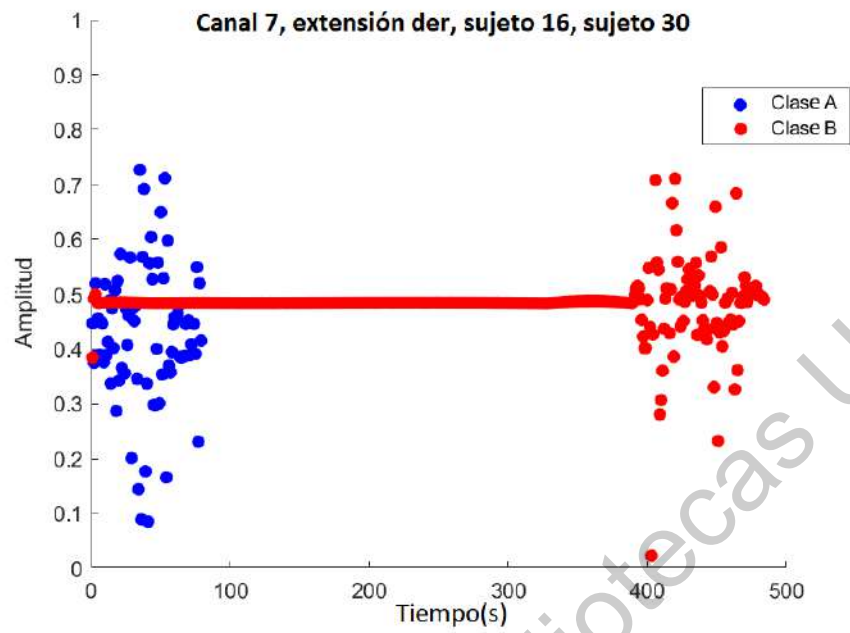


(a)

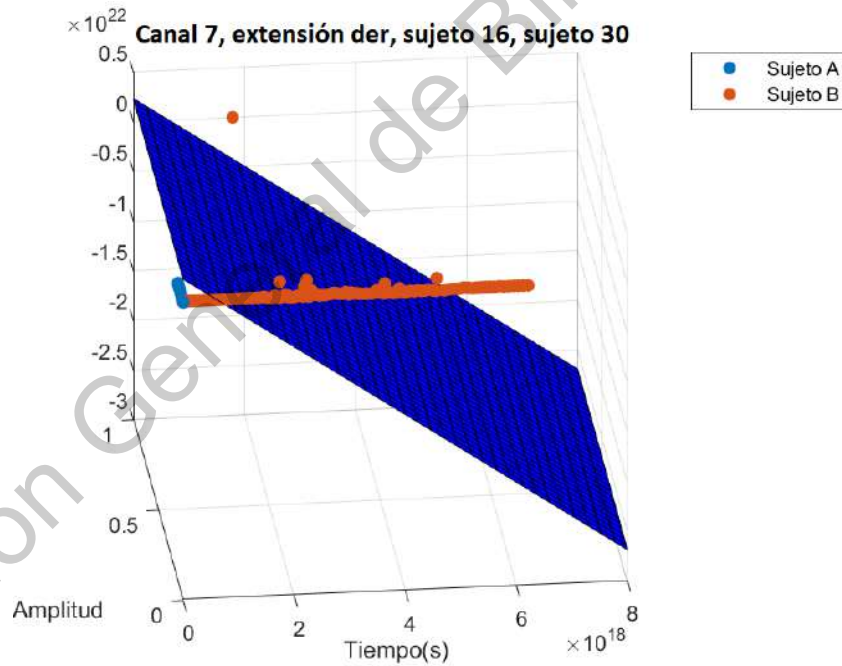


(b)

Fig. 4.57: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) Señaseñall de dos sujetos clasificados por Kernel polinomial de orden 7.

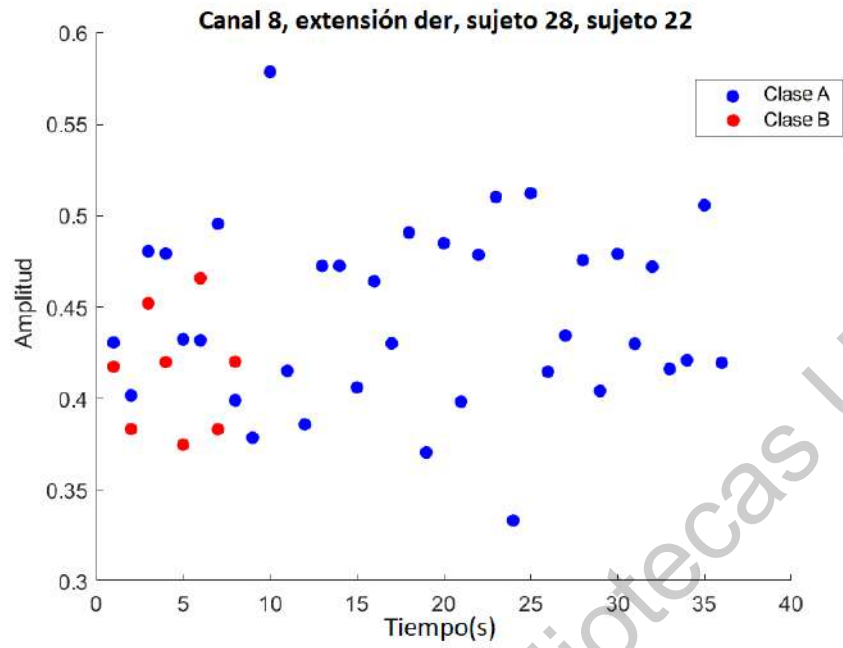


(a)

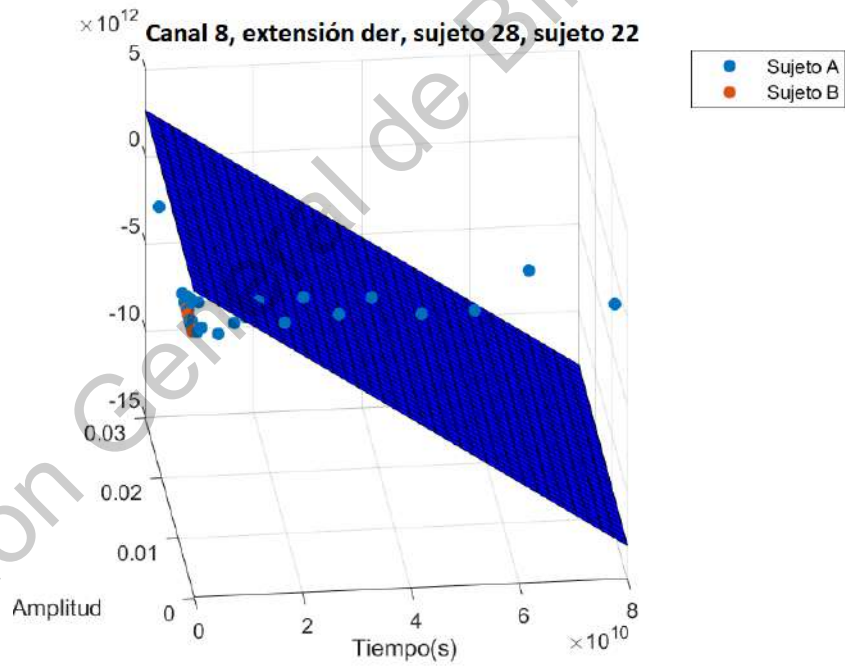


(b)

Fig. 4.58: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.



(a)



(b)

Fig. 4.59: Comparación entre la señal extensión derecha sin clasificar y aplicando el Kernel polinomial de orden 7: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel polinomial de orden 7.

A continuación, se visualiza una comparación para conocer el orden del Kernel con el que se obtienen los mejores resultados. Como se puede ver en la Fig. 4.60 el Kernel de orden cuatro es con el que obtiene los mejores resultados en términos de precisión con la que clasifica los datos, y este comportamiento con Kernel Order cuatro es el mismo para la mano derecha y la izquierda mano como se muestra en la Fig. 4.61.

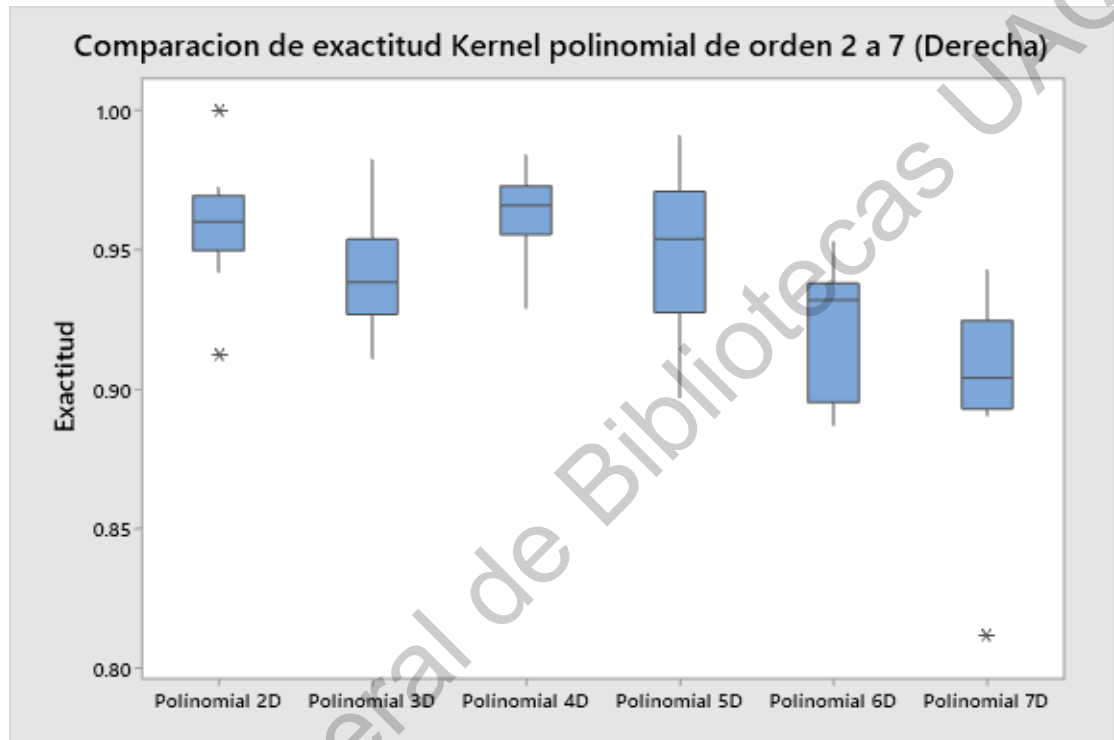


Fig. 4.60: Comparación de la precisión del Kernel polinomial del orden 2D al orden 7D. La figura muestra la comparación en el comportamiento del orden 2D precisión del Kernel con el orden 7D (mano derecha).

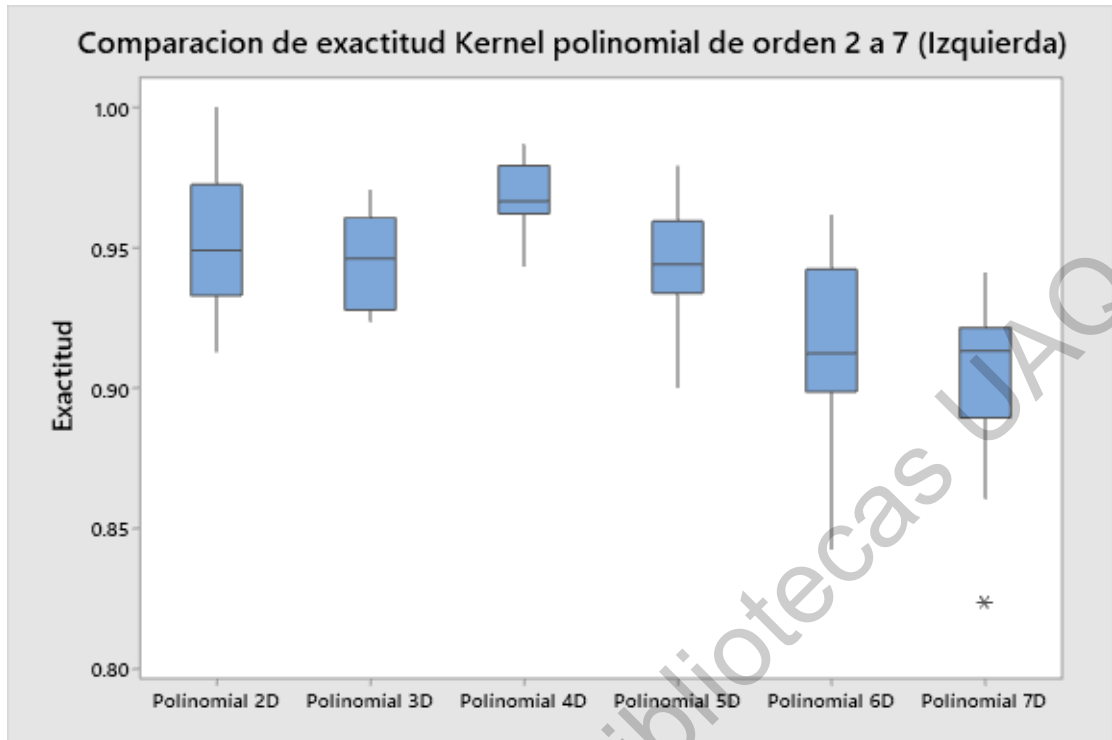


Fig. 4.61: Comparación de la precisión del Kernel polinomial del orden 2D al orden 7D. La figura muestra la comparación en el comportamiento del orden 2D precisión del Kernel con el orden 7D (mano izquierda).

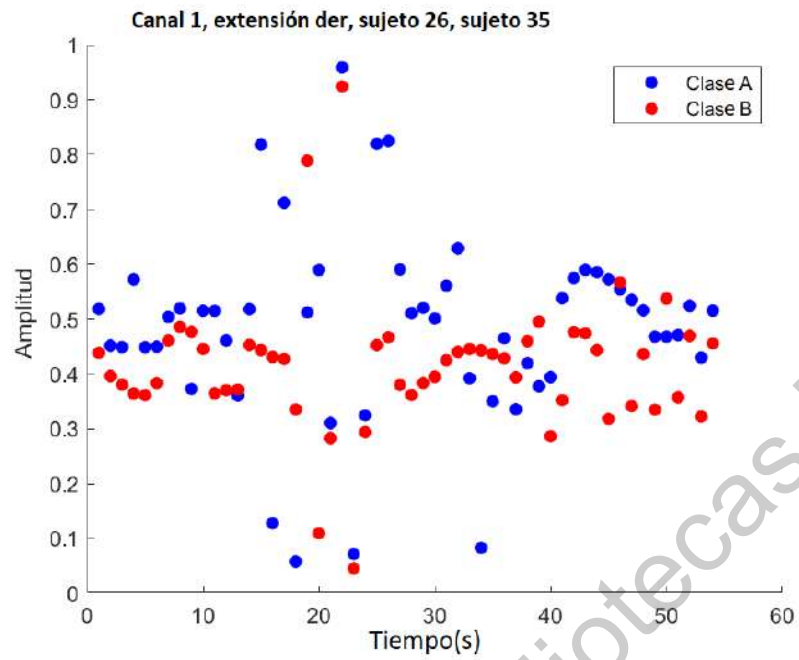
En la tabla 4.2, observamos el comportamiento de exactitud y varianza en cada uno de los movimientos utilizando el Kernel polinomial de orden cuatro, la tabla 4.3 muestra todos los movimientos y comportamiento tanto de la mano derecha como de la mano izquierda.

Tab. 4.2: Tabla con la exactitud y la varianza de los movimientos de la mano derecha utilizando el Kernel polinomial de orden cuatro

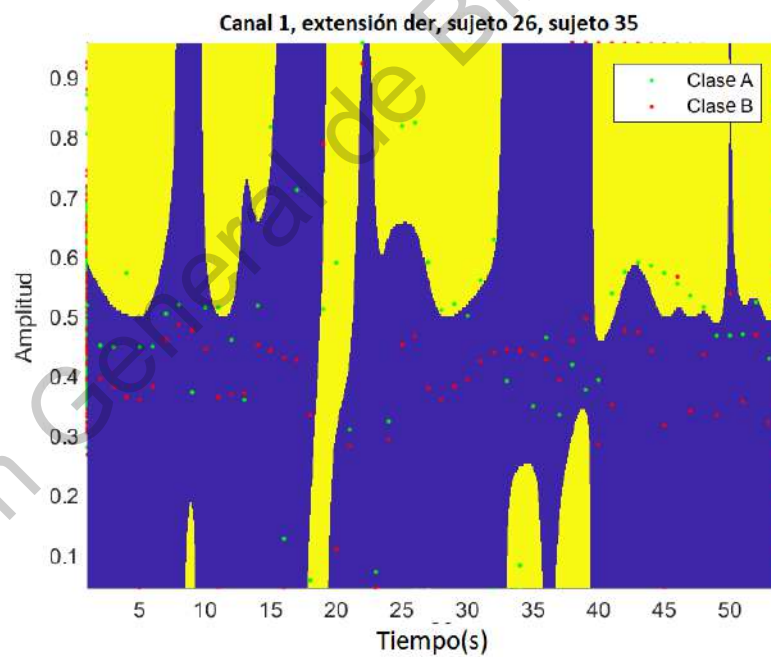
Movimientos	Mano Izquierda		Mano Derecha	
	Exactitud	Varianza	Exactitud	Varianza
Posición Inicial	0.95248	0.0099293	0.9293	0.013654
Pronación	0.97424	0.0039464	0.98405	0.0016066
Supinación	0.9676	0.0053768	0.94922	0.0076423
Extensión	0.94359	0.0075312	0.97013	0.0046315
Flexión	0.96586	0.0046785	0.96407	0.0044501
Desviación Cubital	0.96553	0.0061223	0.96802	0.0049454
Desviación Radial	0.98676	0.0015627	0.95754	0.0073397
Pinza Fina	0.98054	0.003843	0.96186	0.0051877
Pinza Gruesa	0.97906	0.0039901	0.96893	0.0055031
Expansión	0.96588	0.0063141	0.98132	0.0020278

4.3. Kernel Gaussiano

A continuación, se visualizan algunas de las gráficas adquiridas con el Kernel Gaussiano de varios sujetos. En cada grafica se puede observar el comportamiento del Kernel Gaussiano al ser implementado con distintos movimientos, en donde los canales más significativos muestran un mejor comportamiento, tanto del Kernel como del SVM.

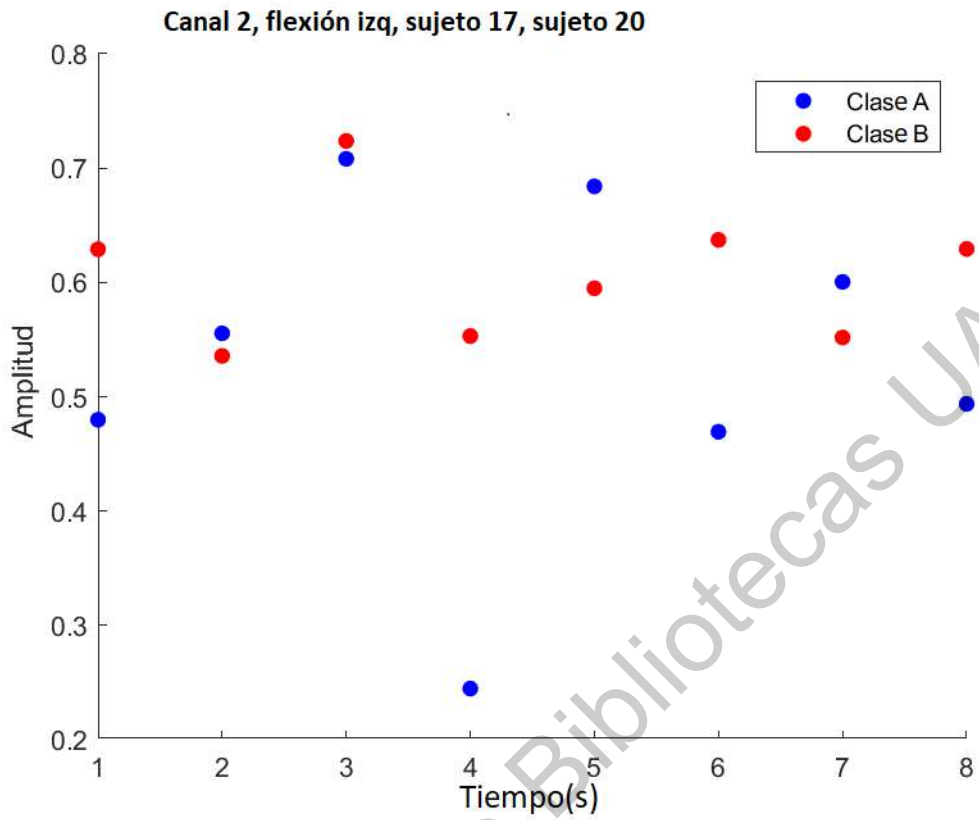


(a)

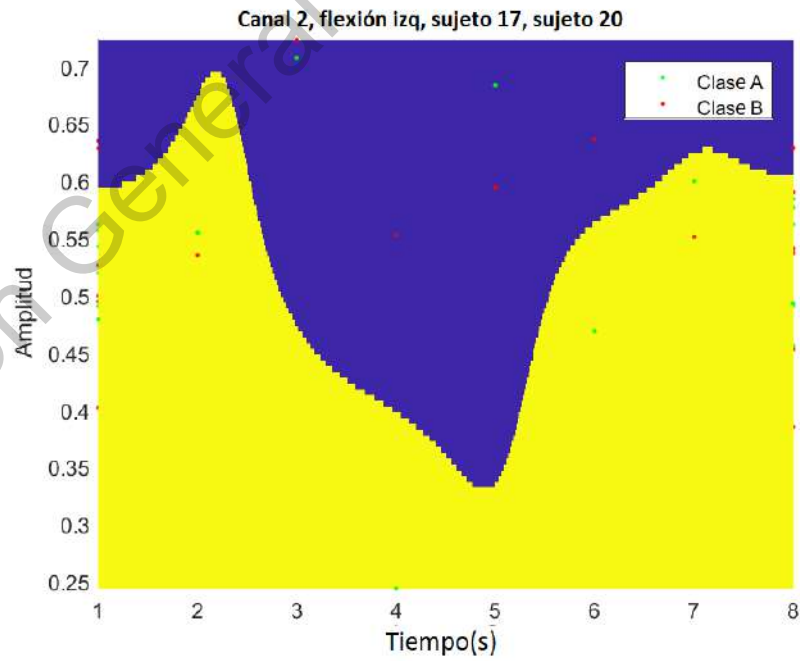


(b)

Fig. 4.62: Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.

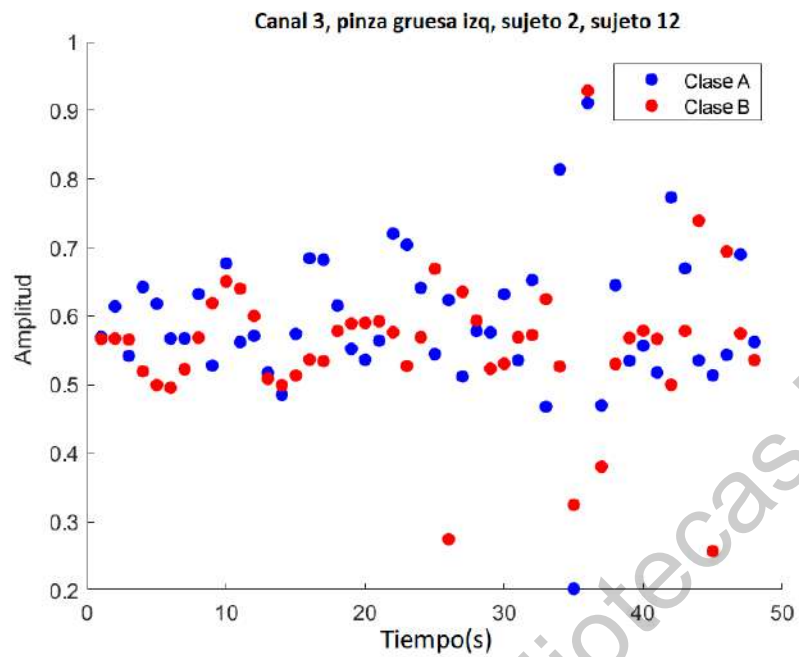


(a)

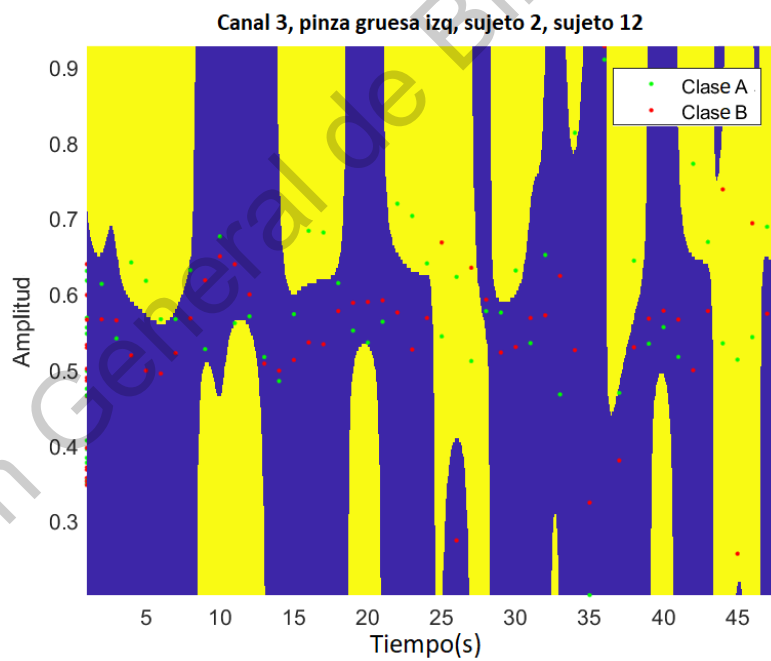


(b)

Fig. 4.63: Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.

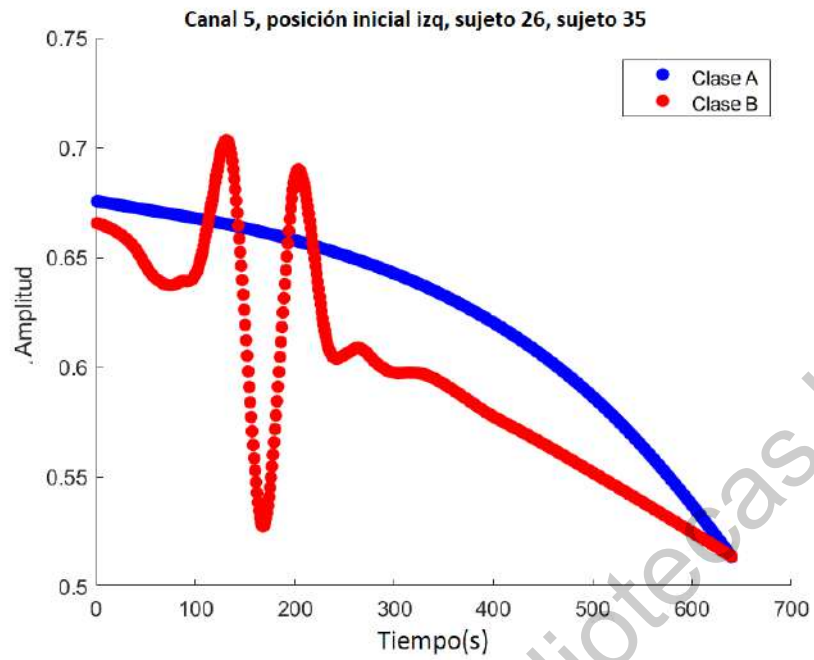


(a)

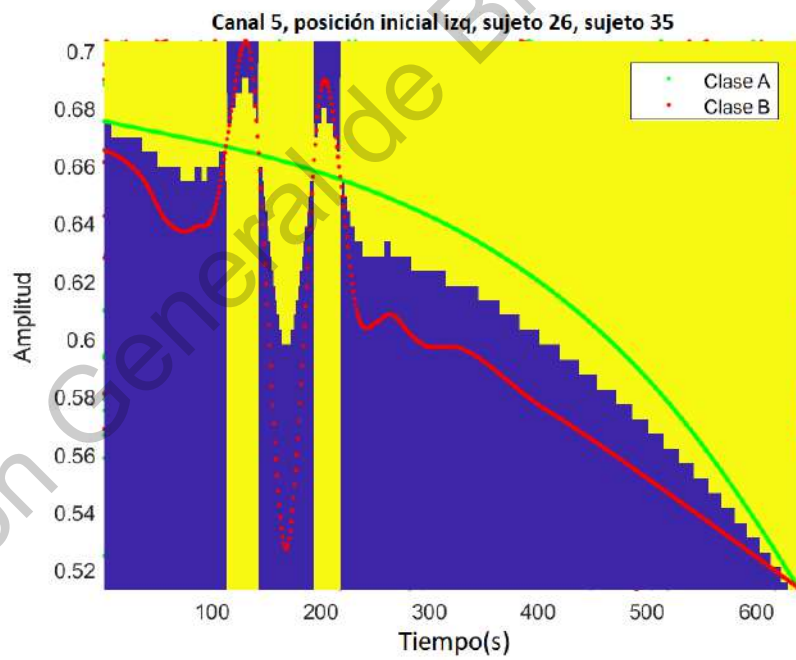


(b)

Fig. 4.64: Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.

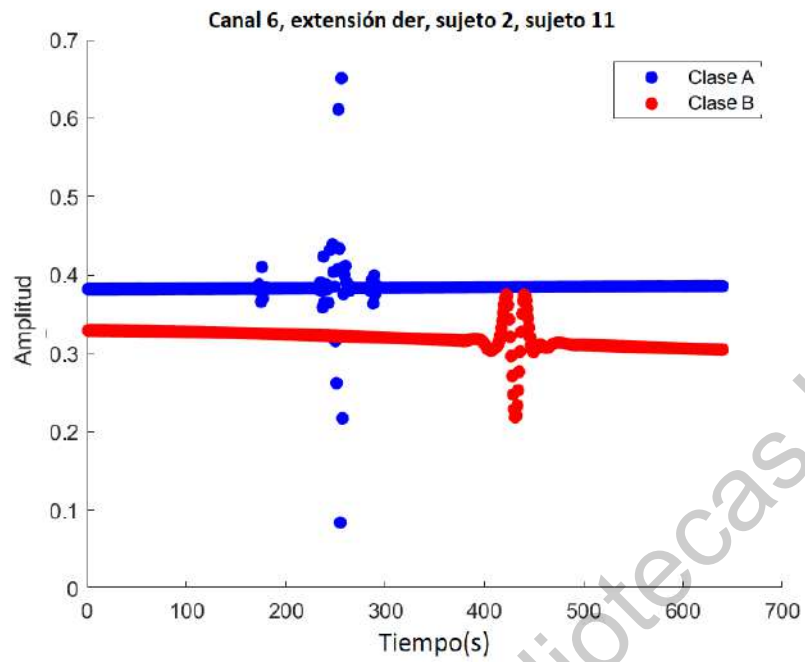


(a)

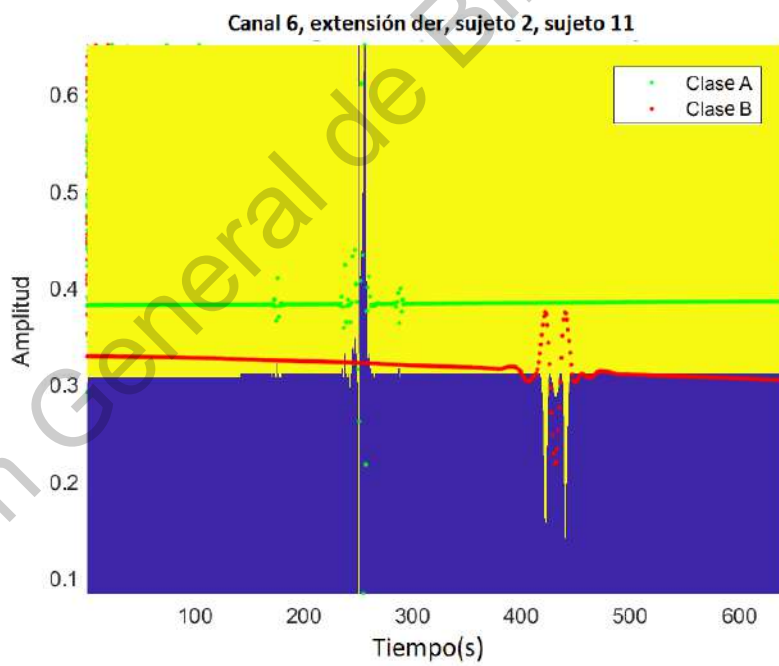


(b)

Fig. 4.65: Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.



(a)



(b)

Fig. 4.66: Comparación entre la señal sin clasificar y la señal aplicando el Kernel Gaussiano: (a) señal de dos sujetos sin clasificar. (b) señal de dos sujetos clasificados por Kernel Gaussiano.

En la tabla 4.3, se muestra el comportamiento de exactitud y varianza, en cada uno de los movimientos utilizando el Kernel Gaussiano, la tabla 4.3 muestra todos los movimientos y el comportamiento de la mano derecha y los movimientos y el comportamiento en la mano izquierda.

Tab. 4.3: Tabla con la exactitud y la varianza de los movimientos de la mano derecha e izquierda usando el Kernel lineal

Movimientos	Mano Derecha		Mano Izquierda	
	Exactitud	Varianza	Exactitud	Varianza
Posición Inicial	0.7776	0.12707	0.79179	0.11749
Pronación	0.87982	0.089312	0.84	0.14
Supinación	0.84	0.14	0.88523	0.066635
Extensión	0.95741	0.021782	0.96126	0.031745
Flexión	0.92308	0.073846	0.99238	0.0015081
Desviación Cubital	1	0	0.94009	0.048708
Desviación Radial	1	0	1	0
Pinza Fina	0.94537	0.029601	0.84615	0.13538
Pinza Gruesa	0.83942	0.11734	0.92232	0.054563
Expansión	1	0	0.97734	0.013345

En la figura 4.67, se muestra el comportamiento de exactitud, en cada uno de los movimientos utilizando el Kernel Gaussiano, la figura 4.67 muestra todos los movimientos y el comportamiento de la mano derecha y el comportamiento en la mano izquierda.

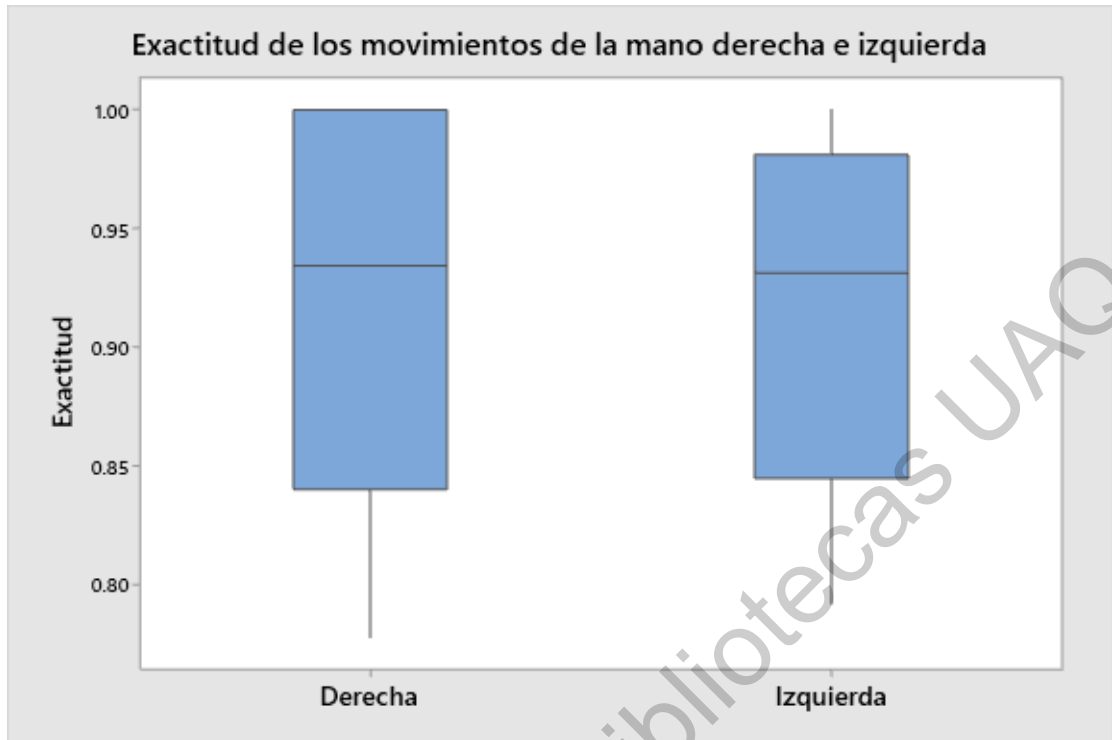
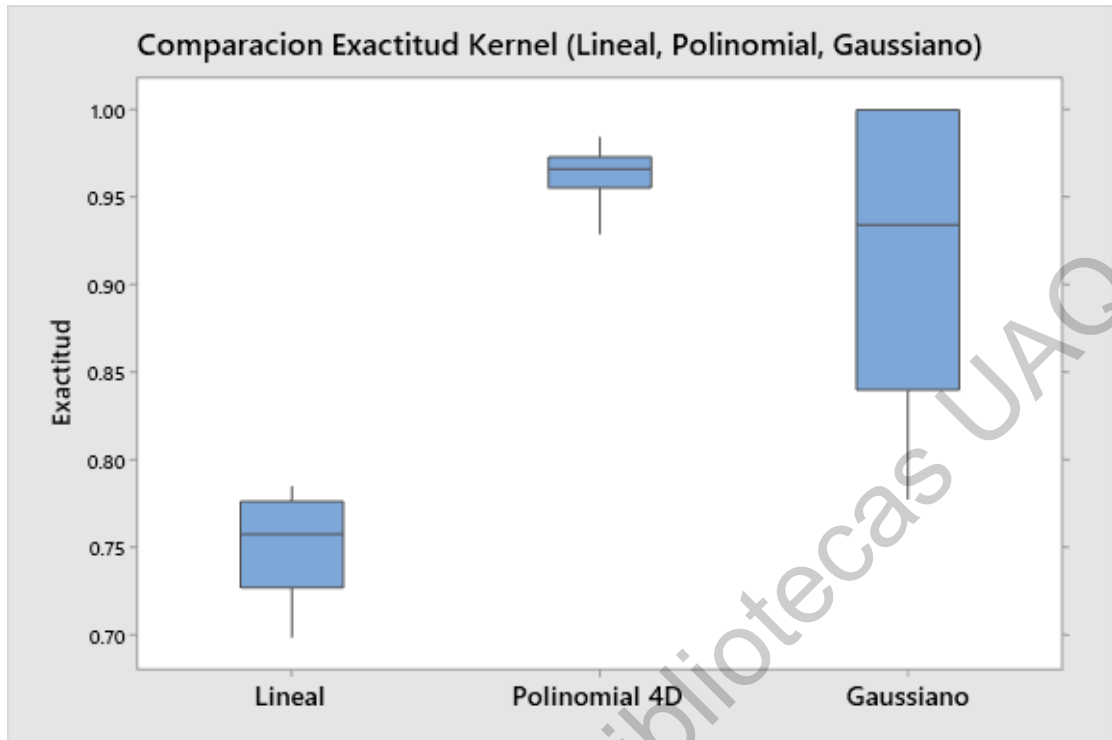
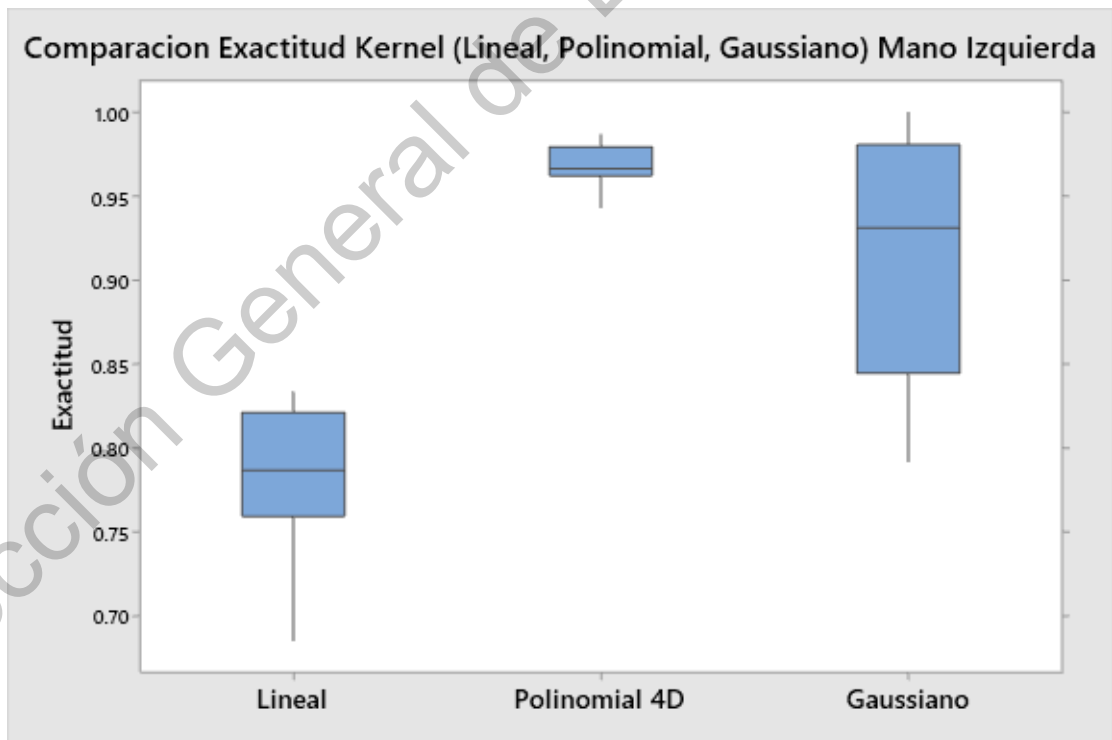


Fig. 4.67: Maquina de Soporte Vectorial (García et al., 2016)

En la figura 4.68, se muestra el comportamiento de la exactitud en ambas manos con cada de los Kernel utilizados.



(a)



(b)

Fig. 4.68: Comparación entre la exactitud de cada uno de los Kernel utilizados: (a) mano derecha. (b) mano izquierda.

5. DISCUSIÓN DE RESULTADOS Y CONCLUSIONES

En esta investigación se muestra el comportamiento de tres diferentes normalizaciones, de acuerdo con los resultados obtenidos, la normalización por Min-Max, en un gráfico circular, demuestra que genera una mejor estandarización de los datos con respecto a las otras dos normalizaciones utilizadas en esta investigación.

En esta investigación se muestra el comportamiento de la clasificación de las señales mioeléctricas utilizando un modelo SVM mediante una técnica Kernel. En los resultados se puede observar el comportamiento de tres tipos de Kernel diferente. El primer Kernel utilizado es el Kernel lineal, su comportamiento en la clasificación no obtuvo resultados favorables, mostrando una precisión de 69 % a 83 % dependiendo del movimiento, sin embargo, al ser el modelo de Kernel más simple, los resultados son bastante buenos, considerando que tiene poca variación.

El segundo Kernel que se aplicó es el Kernel polinomial, de acuerdo con los resultados obtenidos, el comportamiento del Kernel polinomial supera los resultados obtenidos con el Kernel lineal, este Kernel se implementó desde orden dos hasta orden siete, y como se observa en los resultados obtenidos, el Kernel polinomial de orden cuatro es que muestra el mejor comportamiento a la hora de clasificar las señales mioeléctricas. el cual mantuvo una precisión del 92 % al 98 % con una variación muy baja, lo que nos da un grado de confiabilidad mucho mayor que el resto de las pruebas para el mismo Kernel con orden diferente.

El tercer Kernel que se desarrolló es el Kernel Gaussiano, de acuerdo con los resultados obtenidos, el comportamiento del Kernel Gaussiano arrojó los mejores resultados, sin embargo, mostro una alta variabilidad en los resultados, teniendo una precisión de 70 % a 100 % lo que genera un grado de confiabilidad muy baja, con respecto al Kernel Polinomial de orden cuatro que en el que sus resultados se mantienen sobre el 90 % a diferencia del Kernel Gaussiano que consigue sus resultados de 100 % de exactitud y sus peores resultados de 70 %, lo que genera una baja precisión de clasificación.

Como conclusión, de acuerdo con los resultados obtenidos en esta investigación podemos afirmar que se cumple la hipótesis al aumentar la exactitud de la clasificación de las señales mioeléctricas utilizando técnicas Kernel, en la cual se demostró que el Kernel polinomial de orden cuatro supera los resultados obtenidos con el Kernel lineal y el Kernel Gaussiano.

Bibliografía

- Acevedo, J. A. D., Betancourt, I. D., and Hurtado, P. (2013). Diseño e implementación de un electromiógrafo mediante matlab. *ENGI Revista Electrónica de la Facultad de Ingeniería*, 1(2).
- Aceves-Fernandez, M., Ramos-Arreguin, J., Gorrostieta-Hurtado, E., and Pedraza-Ortega, J. (2019). Methodology proposal of emg hand movement classification based on cross recurrence plots. *Computational and Mathematical Methods in Medicine*, 2019.
- Arjunan, S. P., Kumar, D. K., and Naik, G. R. (2010). A machine learning based method for classification of fractal features of forearm semg using twin support vector machines. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 4821–4824. IEEE.
- Asefi, M., Moghimi, S., Kalani, H., and Moghimi, A. (2016). Dynamic modeling of semg–force relation in the presence of muscle fatigue during isometric contractions. *Biomedical Signal Processing and Control*, 28:41–49.
- Banda, H. (2014). *Inteligencia Artificial: Principios y Aplicaciones*.
- Betancourt, G. A. (2005). Las máquinas de soporte vectorial (svms). *Scientia et Technica*, 1(27).
- Bronzino, Joseph D Peterson, D. R. (2015). *The Biomedical Engineering Handbook: Four Volume Set*. CRC Press.

- Burbano, E. M., Paruma, O. H., and Florez, J. F. (2004). Aplicaciones de las señales mioeléctricas para el control de interfaces hombre-máquina. In *VI Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica TAAE*.
- Cristianini, N., Shawe-Taylor, J., et al. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.
- Cruz, P. P. and Herrera, A. (2010). *Inteligencia artificial con aplicaciones a la ingeniería*. Alfaomega.
- García, Fernández, Z., Garcia Nieto, P. J., Bernardo Sánchez, A., and Menéndez, M. (2016). Hard-rock stability analysis for span design in entry-type excavations with learning classifiers. *Materials*, 9:531.
- Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2003). A practical guide to support vector classification.
- Joachims, T. (1998). Making large-scale svm learning practical. Technical report, Technical report, SFB 475: Komplexitätsreduktion in Multivariaten
- Kakoty, Nayan M Hazarika, S. M. (2011). Recognition of grasp types through principal components of dwt based emg features. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–6. IEEE.
- Kilany, M., Houssein, E. H., Hassanien, A. E., and Badr, A. (2017). Hybrid water wave optimization and support vector machine to improve emg signal classification for neurogenic disorders. In *2017 12th International Conference on Computer Engineering and Systems (ICCES)*, pages 686–691. IEEE.
- Kowalczyk, A. (2017). Support vector machines succinctly. *Syncfusion Inc*.
- Larrazá, A. E. (2006). *Mecanismos moleculares implicados en la potenciación de canales de potasio KCNQ2/3*. PhD thesis, Universidad Autónoma de Madrid.

- López Díaz, A. (2018). *Fundamentos matemáticos de los métodos Kernel para aprendizaje supervisado*. PhD thesis, Universidad de Sevilla.
- Maglogiannis, I. G., Karpouzis, K., and Wallace, M. (2005). Image and signal processing for networked e-health applications. *Synthesis Lectures On Biomedical Engineering*, 1(1):1–108.
- Martínez-Miguel, A., Vargas-Pérez, S., Gómez-Merlín, E., Arias-Montiel, M., Lugo-González, E., and Miranda-Luna, R. (2017). Control de movimiento de una mano robótica mediante señales electromiográficas. In *Memorias del Congreso Nacional de Ingeniería Biomédica*, volume 3, pages 73–76. CNIB.
- Mesin, L., Merletti, R., and Rainoldi, A. (2009). Surface emg: the issue of electrode location. *Journal of Electromyography and Kinesiology*, 19(5):719–726.
- Moslem, B., Khalil, M., Diab, M. O., Chkeir, A., and Marque, C. (2011a). Combining multiple support vector machines for boosting the classification accuracy of uterine emg signals. In *2011 18th IEEE International Conference on Electronics, Circuits, and Systems*, pages 631–634. IEEE.
- Moslem, B., Khalil, M., Diab, M. O., Chkeir, A., and Marque, C. (2011b). Combining multiple support vector machines for boosting the classification accuracy of uterine emg signals. In *2011 18th IEEE International Conference on Electronics, Circuits, and Systems*, pages 631–634. IEEE.
- Nougarou, F., Campeau-Lecours, A., Islam, R., Massicotte, D., and Gosselin, B. (2018). Muscle activity distribution features extracted from hd semg to perform forearm pattern recognition. In *2018 IEEE Life Sciences Conference (LSC)*, pages 275–278. IEEE.
- Oppenheim, A. V., Willsky, A. S., and Nawab, S. H. (1998). *Señales y sistemas*. Pearson Educación.

- Plonsey, R. (1988). Bioelectric sources arising in excitable fibers (alza lecture). *Annals of biomedical engineering*, 16(6):519–546.
- Pomboza, Terriza, J. H. (2016). Hand gesture recognition based on semg signals using support vector machines. In *2016 IEEE 6th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, pages 174–178. IEEE.
- Ramírez-Angeles, I. (2019). *DETECCIÓN Y CLASIFICACIÓN DE SEÑALES MIOELÉCTRICAS EN EL BRAZO MEDIANTE EL USO DE ALGORITMOS BASADOS EN INTELIGENCIA ARTIFICIAL*. Thesis, Maestría en Ciencias en Inteligencia Artificial. Universidad Autónoma de Querétaro.
- Ramírez-Ángeles, I. and Aceves-Fernandez, M. (2018). Multi-channel electromyography signal acquisition of forearm. <http://dx.doi.org/10.17632/p77jn92bzg.1>.
- Ramírez-Ángeles, I., Razo-Lopez, N., Aceves-Fernandez, M., and Gorrostieta-Hurtado, E. (2019). Metodología para la adquisición de señales electromiográficas en el brazo utilizando un lector de señales multicanal. *La Mecatrónica en México*, page 22 – 36.
- Romero, J. J., Dafonte, C., Gómez, Á., and Penousal, F. J. (2007). Inteligencia artificial y computación avanzada. *Santiago de Compostela: Fundación Alfredo Brañas*, pages 10–15.
- Semmlow, J. L. (2008). *Biosignal and medical image processing*. CRC press.
- Suárez, E. J. C. (2014). Tutorial sobre máquinas de vectores soporte (svm). *Tutorial sobre Máquinas de Vectores Soporte (SVM)*.
- Sukumar, N., Taran, S., and Bajaj, V. (2018). Physical actions classification of surface emg signals using vmd. In *2018 International Conference on Communication and Signal Processing (ICCSP)*, pages 0705–0709. IEEE.

Sun, L., Fu, S., and Wang, F. (2019). Decision tree svm model with fisher feature selection for speech emotion recognition. *EURASIP Journal on Audio, Speech, and Music Processing*, 2019(1):2.

Turing, A. M. (2009). Computing machinery and intelligence. In *Parsing the turing test*, pages 23–65. Springer.

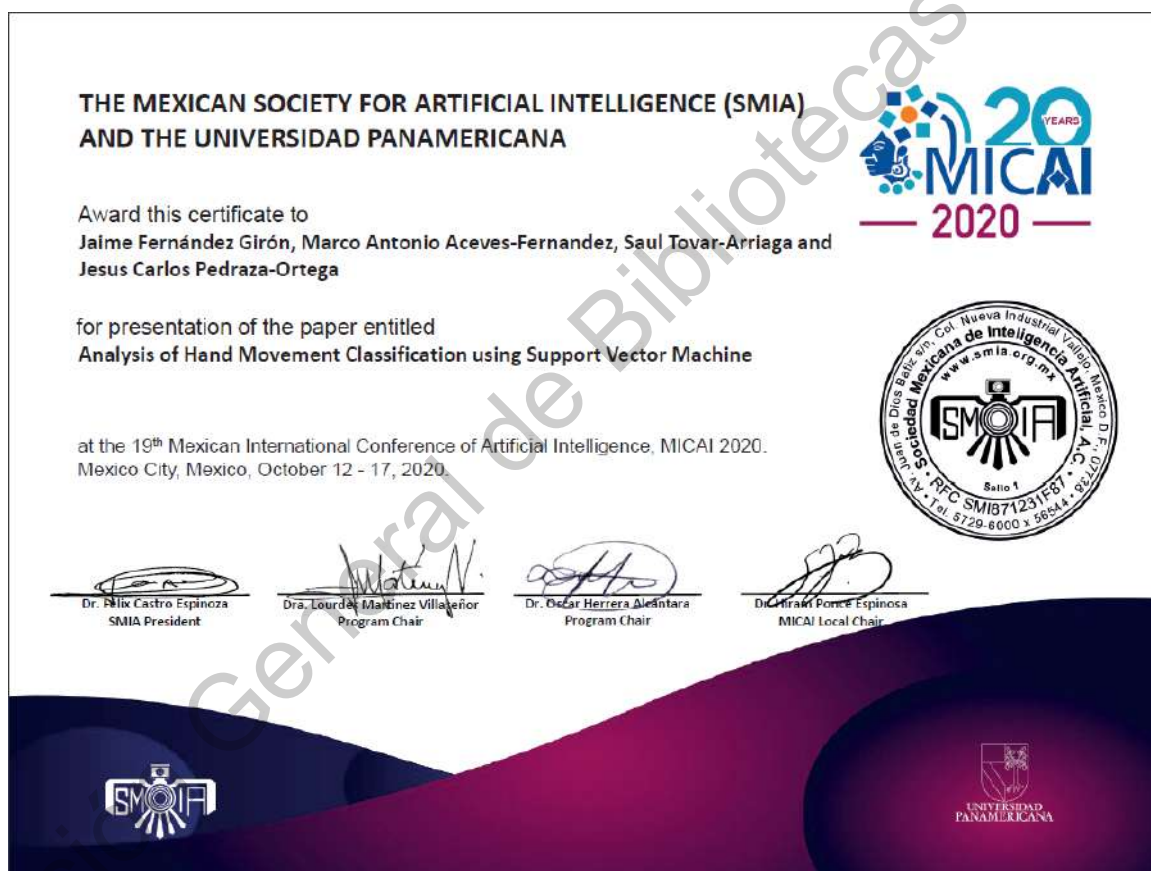
Vapnik, Cortes, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Zeng, Y., Yang, J., Peng, C., and Yin, Y. (2019). Evolving gaussian process auto-regression based learning of human motion intent using improved energy kernel method of emg. *IEEE Transactions on Biomedical Engineering*.

Dirección General de Bibliotecas UAQ

5. ANEXOS

5.1. Artículo publicado



5.2. Constancia de lengua extranjera



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE LENGUAS Y LETRAS



Campus Aeropuerto, 03 de diciembre de 2020.
FLL/SA/HCA/2020/1021

C. Fernández Girón, Jaime
Exp. 283985
Facultad de Ingeniería
Presente.

Me permito comunicar a usted que en Sesión Ordinaria del H. Consejo Académico de la Facultad de Lenguas y Letras, celebrada el **03 de diciembre de 2020**, se acordó **Aprobar** su solicitud de **convalidación** de la Constancia del Examen de Manejo de la Lengua (8+), expedida por el CELyC, por el Examen de Comprensión de Textos en Inglés

Sin más por el momento, me es grato quedar a sus órdenes para cualquier aclaración.

Atentamente,
"Enlazar Culturas por la Palabra"

Dra. Ma. de Lourdes Rico Cruz
Secretaría Académica



SECRETARÍA
ACADÉMICA

C.c. Expediente.
Archivo.

MURC/mip

SOMOS UAQ
EDUCAR CRECER CONSOLIDAR

Campus Aeropuerto, Anillo Vial Fray Junípero Serra S/N, Querétaro, Qro. C.P. 76140
Tel. 442 192 12 00 Dirección Ext. 61010, Secretaría Administrativa Ext. 61300, Posgrado Ext. 61140,
Licenciatura Ext. 61070, Centro de Lenguas Ext. 61050, Secretaría Académica Ext. 61100 y Planeación Ext. 61110

Analysis of Kernel for of Hand Movement Classification using Support Vector Machine

J. Fernández-Girón¹, M. A. Aceves-Fernández², S. Tovar-Arriaga³, and J. C. Pedraza-Ortega⁴

Faculty of Engineering, Universidad Autónoma de Querétaro, Cerro de las Campanas
S/N, 76000 Querétaro, Mexico
marco.aceves@gmail.com

Abstract. Nowadays, the need to use technology as an aid in our day-to-day life, has led us to want to use it with just a finger or a whole hand. The movements of a hand are considerably difficult to classify because of its close similarity between each of the different movements ranging from the opening of the fist to fine movements such as the tweezer between the index finger and the thumb. In this contribution, the combination of an artificial intelligence method based on SVM (Support Vector Machine) with different Kernel is intended to compare and demonstrate the classification capacity of each. The use of SVM with a Kernel technique increases the classification capacity in databases difficult to classify, as is the classification of movements of a hand, obtaining better results with a polynomial Kernel than with a linear Kernel or Gaussian Kernel.

Keywords: Mano · SVM · Kernel

1 Introduction

1.1 Electromyographic signal

The human hand is able to execute as many movements as its axes and joints allow, it is for them that the classification of the movements of a hand by means of the processing of the digital signals has nowadays been an issue of interest for a large sector of the population that seeks to make a correct classification of the movements of the hand and to apply it both in prostheses, orthosis and in the rehabilitation area [1].

The surface electromyographic (EMG) signal is the sum of the action potentials generated by the active motor units and detected over the skin [2], action potentials is a point process determined by the activation. of the motoneurons. [2].The use of EMG signals to identify hand movements uses a greater number of sensors, which generates a relatively high problem when electrical activity is very small and multiple muscles are simultaneously active and becomes a non-linear problem [3]. For this reason, in this contribution a model based on Support Vector Machine (SVM) is proposed to compare the classification accuracy of the movements of a hand using different Kernel.

1.2 Support Vector Machine

The Support Vector Machines (SVM) originate from the theory work of statistical learning by Vapnik and his collaborators [4]. Although originally designed to solve binary classification problems, they are currently used in various fields such as artificial intelligence, this thanks to their solid theoretical foundations [5]. The fundamental idea of the SVM is to find the hyperplane of separation that divides the data equally having of reference the closest points of each class and in this way obtain the maximum margin of separation on each side of the hyperplane. Where only points falling at the border of the margins will be taken into account. These points are called supporting vectors [6].

The SVM can be used as a classifier in a modular way, which involves the function of Kernel and depending on its purpose, domain and data separability, it can be hard-band, which implies that the data are linearly separable, or soft margin allowing a certain error percentage [7]. As mentioned above, being several electrodes simultaneously active, this generates a binary classification problem with examples not linearly separable and requires what is known as kernelized SVM [5].

1.3 Kernel

Due to its excellent approximation ability and affordable computational complexity, kernel method has been widely used in many machine learning algorithms [8]. There are different Kernel such as Gaussian kernel, polynomial kernel, sigmoid kernel, linear kernel, radial base kernel, etc. It is often not very clear which Kernel is best suited to the type of data being handled for a given task. and thus it is desirable for the kernel methods to use an optimized kernel function that adapts well to the data structure and type of the boundaries between classes. [9].

2 Materials and methods

2.1 Materials

The SVM model was built using a database of fifty subjects, with no history of muscle disorders (data previously processed by the Wavelet Transform [10]). In this contribution, fifty subjects performed ten different hand movements in both hands, having a total of twenty movements between both hands, with the help of electrodes placed on each arm [11]. The data were obtained using a bracelet with eight electrodes placed around the forearm, with which we obtained eight channels that captured the electrical activity of all the muscles involved in the movement of the hand, each movement was repeated five times generating five events per movement on each channel [10] as show on Fig. ?? which makes the task of modeling the behavior of hand movement more challenging, having

the activity of a single movement on eight electrodes recorded simultaneously as shown on Fig. ?? where the original signals were processed by the Wavelet Transform [10], which also allows us to appreciate that muscle is exercising the most activity during the execution of a certain movement. The model was validated using the same dataset as the test subjects. To measure the accuracy of the proposed method, a segment of subjects corresponding to 20% (randomly) of total subjects was omitted from the SVM model in order to validate how accurate is the model without overfitting the model or skewing the results.

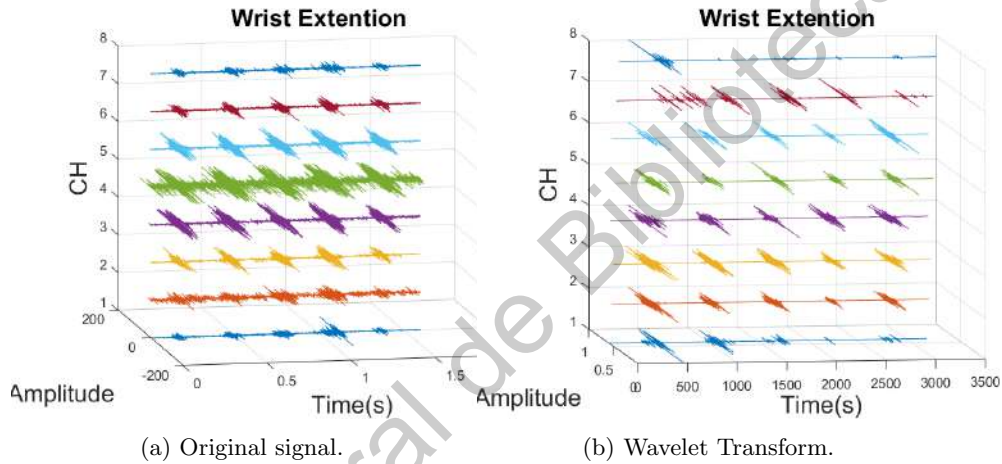


Fig. 1: Example of a movement performed by test subject. The Fig. ?? shows the pre-filter signal set, and the Fig. ?? shows the signal set previously processed by the Wavelet Transform [10] for the movement wrist extension (right hand).

2.2 Model construction

The construction of the model was carried out using a comparison between 2 different Kernel under the same conditions in the distribution of the data. To generate the SVM model with linear kernel was based on the model used by [12] y [13], as shown in the Fig. ?. For linear kernel training data was used with which the subject was randomly designated for the first class and also randomly the second subject was designated for the second class taking only the first movement and discretising the signal to get only the points.

SVM is a powerful classification technique based on the theory of statistical learning, which makes it primarily a two-class classifier [14].

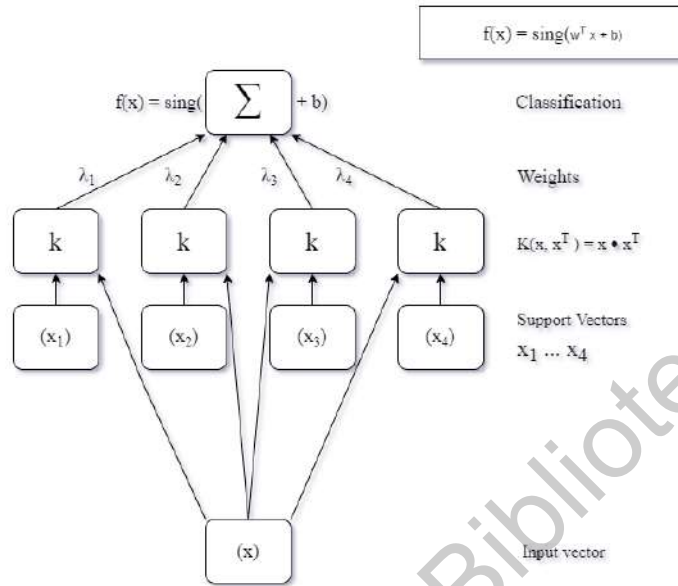


Fig. 2: Support vector machine architecture based on the diagram of [12]

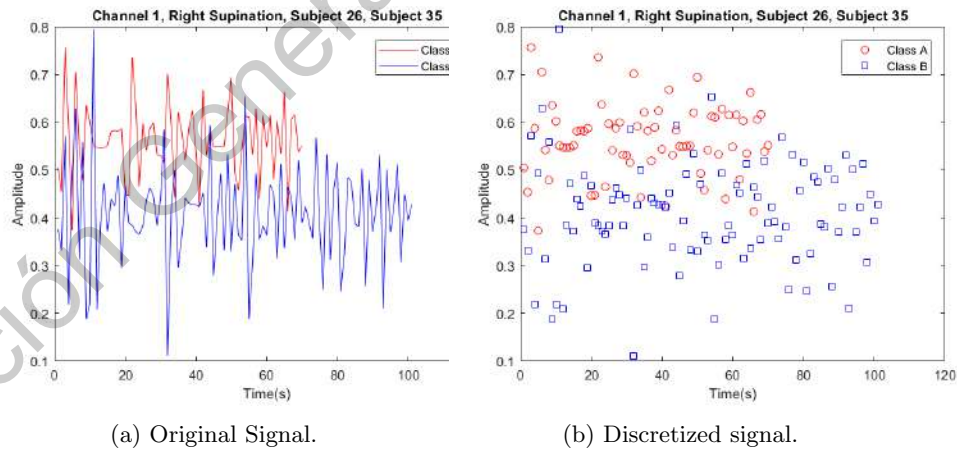


Fig. 3: Example of a movement made by two test subjects. The Fig. ?? shows the movement of Right supination for subjects 26 and 35 respectively, and the Fig. ?? shows the same signal Fig. ?? after discretization (right hand).

The optimization criterion is the width of the margin between the classes, is to say, the area between the maximum separation margin of class A, up to the maximum separation margin of class B, leaving just in half the line separating both classes (the hyperplane of separation) defined by the distance to the nearest training points. These points, called support vectors, finally define the classification function [14], [15].

$$w^T \cdot x + b = 0 \quad (1)$$

A Kernel is a function that returns the result of a dot product performed in another space. More formally, we can write:

Given a mapping function $\phi : \chi \rightarrow \nu$, we call the function $\kappa : \chi \rightarrow \mathfrak{R}$ defined by $\kappa(x, x^T) = \langle \phi(x), \phi(x^T) \rangle_\nu$, where $\langle \cdot, \cdot \rangle_\nu$ denotes an inner product in ν , a Kernel function [15].

The Kernel function is shown below [16]:

$$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \quad (2)$$

Linear kernel This is the simplest kernel [15]. It is simply defined by:

$$K(x, x^T) = x \cdot x^T \quad (3)$$

where x and x^T are two vectors.

Polynomial kernel A polynomial kernel function of degree d [17] is expressed as:

$$K(x, x^T) = (x \cdot x^T + c)^d \quad (4)$$

It has two parameters: c , which represents a constant term, and d , which represents the degree of the kernel.

Gaussian Kernel It is a Kernel classifier of support vectors in infinite dimensions, it is expressed as:

$$K(x, x^T) = e^{-\gamma(a-b)^2} \quad (5)$$

Thus:

$$e^{-\gamma(a-b)^2} = (s, s\sqrt{\frac{1}{1!}}a, s\sqrt{\frac{1}{2!}}a^2, \dots, s\sqrt{\frac{1}{\infty!}}a^\infty) \cdot (s, s\sqrt{\frac{1}{1!}}b, s\sqrt{\frac{1}{2!}}b^2, \dots, s\sqrt{\frac{1}{\infty!}}b^\infty) \quad (6)$$

Where a and b are two different vectors, and γ which is determined by cross validation, scales the square distance, therefore, it scales the influence to the new observations.

2.3 Proposed Methodology

The proposed methodology has been taken from [3]. This methodology provides the general steps to make a classification model of the movements of a hand using SVM. For this proposal the functions of the Linear Kernel and the Polynomial Kernel are used. The objective of this work is to make a comparison of the operation of these Kernel for the correct classification of the myoelectric signals in each case, generate the design, training and testing of the SVM with each Kernel, as shown in Fig. ??.

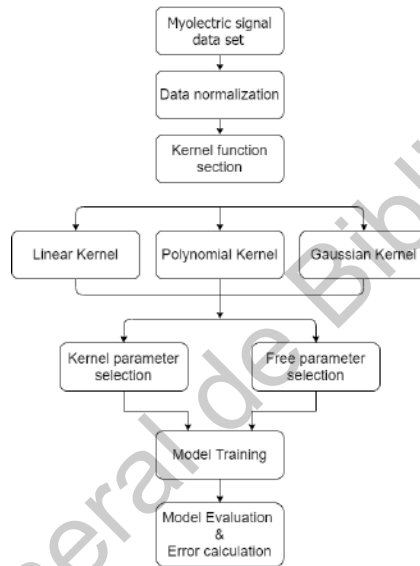


Fig. 4: Proposed methodology.

The steps based on the methodology are as follows:

- The data extracted from the 50 test subjects were previously processed using the Wavelet Transform [10], in order to eliminate as much noise as possible that makes it difficult to see the shape of the myoelectric wave.
- Preprocessing the input data by normalizing, scaling the data into a range of $[0, 1]$.
- The test subjects will be stopped to serve as training corresponding to 80% of the total and with the other 20% remaining reserve them until the end for the tests, the removal of subjects used for training shall be random in order

to avoid bias.

- Select the data that will be used to do the SVM training, using only the data of the subjects destined for the training, taking the channel and the movement in an orderly way, starting with the first movement of the right hand, followed by the next movement corresponding to the left hand until using the ten movements in both hands, and in the same way from channel one to channel eight, likewise for each event, that is, from event one to event five, randomly select the first subject and select the same parameters for the second subject, in such a way as to match the channel, movement and event for both subjects but that subjects are matched randomly.
- Plot both subjects in the same graph to proceed with the application of the SVM.
- Select the parameters of the kernel function, and in the case of the Polynomial the equation that will raise the data to another order.
- Validate the model obtained during training with the test data that were previously separated by taking the same parameters of the movements, that is, the channel, movement and event match each subject having only random test subjects.

3 Results

The results presented in this contribution were validated using 20% of the subjects in the database to reserve them at the end and generate the tests. And for the training 80% of the subjects were used with which randomly designated the subject corresponding to the first class and equally randomly designated the second subject for the second class. Two experiments were generated for this contribution, one consists of generating a linear kernel as shown on Fig. ??, that classifies the recorded EMG signals of the fifty subjects.

In Table ??, the behavior of accuracy and variance is observed in each of the movements of both hands using the linear Kernel, in the left side are shown all movements and behavior of the Left hand respectively, while on the right side shows the movements and behavior in the right hand.

The second experiment consists in generating a polynomial kernel like the one shown in Fig. ?? which, like the linear one, classifies the recorded EMG signals of the 50 subjects.

In this experiment, since it is a polynomial kernel and because it does not know the order of the kernel to which it generates the best results, it is executed from order two to order seven, and a comparison is made to know the order of the kernel with which the best results are obtained, as can be seen in Fig. ?? the Kernel of order four is with which you get the best results in terms of the

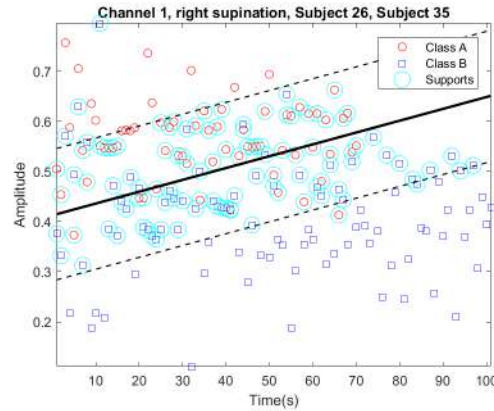


Fig. 5: Example of an SVM with linear kernel. The figure shows the model of a linear kernel, which classifies 2 subjects that execute the supination movement (right hand).

accuracy with which you classify the data, and this behavior with Kernel Order four is the same for both the right hand and the left hand as shown in Fig. ??.

In Table ??, the behavior of accuracy and variance is observed in each of the movements of both hands using the Kernel polynomial of order four, the table shows all movements and behavior of the left hand and behavior in the right hand.

Next, one of the graphs acquired with the Gaussian Kernel is visualized, its behavior can be observed when it is implemented with different movements, where the most significant channels show a better behavior, both of the Kernel and the SVM as shown on Fig. ??.

In Table ??, the behavior of accuracy and variance is observed in each of the movements of both hands using the Gaussian Kernel, in the left side are shown all movements and behavior of the Left hand respectively, while on the right side shows the movements and behavior in the right hand.

Previously it was observed in the graphs both the behavior of the linear Kernel, that of the polynomial Kernel and that of the Gaussian Kernel, in the following Fig. ?? it is observed in a single graph the behavior of both Kernel in the right hand and in Fig. ?? the behavior in the left hand.

Table 1: Table with the accuracy and variance of the movements of both hands using the linear kernel

Movement Left	Accuracy	Variance	Movement Right	Accuracy	Variance
Wrist in neutral	0.75864	0.14661	Wrist in neutral	0.78427	0.15312
Pronation	0.83333	0.16667	Pronation	0.75714	0.14048
Supination	0.81735	0.16089	Supination	0.7624	0.14543
Wrist extension	0.75926	0.14417	Wrist extension	0.77326	0.15643
Wrist flexion	0.68519	0.12477	Wrist flexion	0.69833	0.12271
Ulnar deviation	0.78574	0.15237	Ulnar deviation	0.70729	0.12768
Radial deviation	0.77895	0.148	Radial deviation	0.78428	0.15262
Fine pinch	0.78769	0.15443	Fine pinch	0.75674	0.14332
Thick grip	0.80556	0.16019	Thick grip	0.73307	0.14012
Hand open	0.83247	0.16632	Hand open	0.7574	0.14777

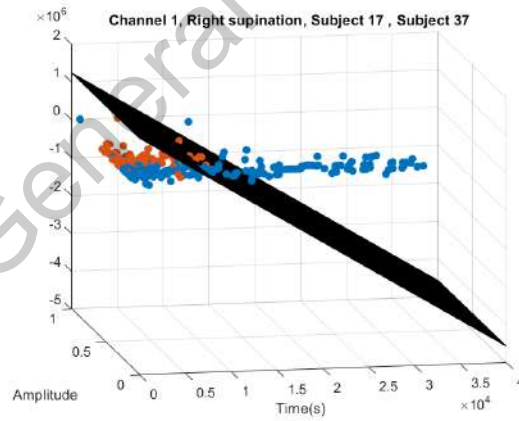


Fig. 6: Example of an SVM with polynomial kernel. The figure shows the model of a polynomial kernel, which classifies 2 subjects that execute the supination movement (right hand).

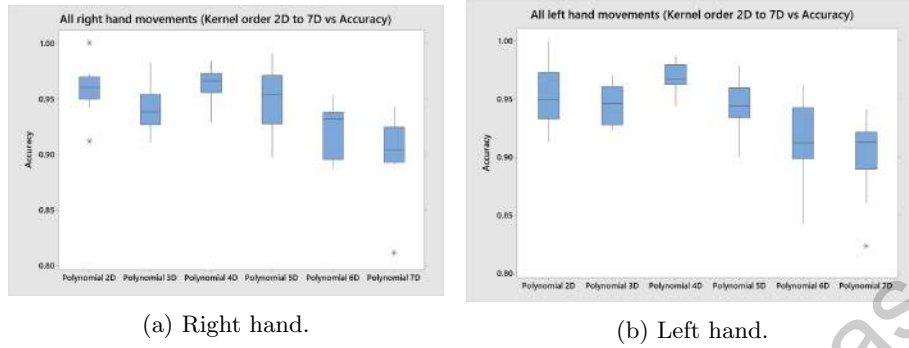


Fig. 7: Comparison of Polynomial Kernel accuracy of two order to seven order. The figure shows the comparison in the behavior of the accuracy of the polynomial kernel of order two up to order seven.

Table 2: Table with the accuracy and variance the movements of both hands using the polynomial Kernel of order four

Movement Left	Accuracy	Variance	Movement Right	Accuracy	Variance
Wrist in neutral	0.95248	0.0099293	Wrist in neutral	0.9293	0.013654
Pronation	0.97424	0.0039464	Pronation	0.98405	0.0016066
Supination	0.9676	0.0053768	Supination	0.94922	0.0076423
Wrist extension	0.94359	0.0075312	Wrist extension	0.97013	0.0046315
Wrist flexion	0.96586	0.0046785	Wrist flexion	0.96407	0.0044501
Ulnar deviation	0.96553	0.0061223	Ulnar deviation	0.96802	0.0049454
Radial deviation	0.98676	0.0015627	Radial deviation	0.95754	0.0073397
Fine pinch	0.98054	0.003843	Fine pinch	0.96186	0.0051877
Thick grip	0.97906	0.0039901	Thick grip	0.96893	0.0055031
Hand open	0.96588	0.0063141	Hand open	0.98132	0.0020278

Table 3: Table with the accuracy and variance of the movements of both hands using the Gaussian kernel

Movement Left	Accuracy	Variance	Movement Right	Accuracy	Variance
Wrist in neutral	0.7776	0.12707	Wrist in neutral	0.79179	0.11749
Pronation	0.87982	0.089312	Pronation	0.84	0.14
Supination	0.84	0.14	Supination	0.88523	0.066635
Wrist extension	0.95741	0.021782	Wrist extension	0.96126	0.031745
Wrist flexion	0.92308	0.073846	Wrist flexion	0.99238	0.0015081
Ulnar deviation	1	0	Ulnar deviation	0.94009	0.048708
Radial deviation	1	0	Radial deviation	1	0
Fine pinch	0.94537	0.029601	Fine pinch	0.84615	0.13538
Thick grip	0.83942	0.11734	Thick grip	0.92232	0.054563
Hand open	1	0	Hand open	0.97734	0.013345

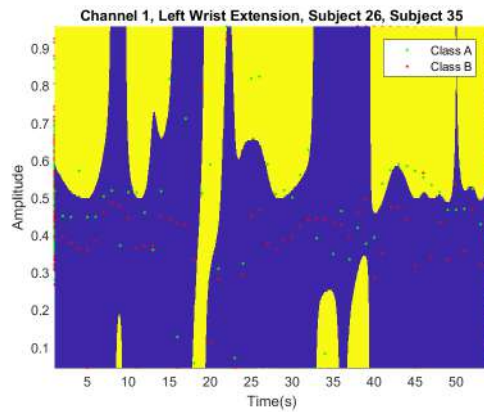


Fig. 8: Example of an SVM with Gaussian kernel. The figure shows the model of a Gaussian Kernel, which classifies 2 subjects that execute the Wrist Extension movement (Left hand).

4 Conclusions and future work

This paper shows the behavior of EMG signals when they are classified using an SVM model in conjunction with a kernel. The behavior of the linear Kernel was not as favorable as expected, it showed an accuracy of 69% to 83% depending on the movement, however, to be the simplest of the Kernel models, the results are quite good, considering that it has little variance. While the behavior of the polynomial kernel shows the best results with order four, which maintained an accuracy of 92% to 98% with a very low variance, which gives us a much higher degree of reliability than the rest of the tests for the same kernel and different order, also shows better performance for non-linear data classification than Gaussian Kernel which maintained an accuracy of 70% to 100% with a very high variance.

As future work, it may be convenient to explore other models such as the sigmoid Kernel or the radial-based Kernel, in order to find the Kernel that best performs the classification of EMG signals.

Conflict of interest

The authors declare that there is no conflict of interest at present regarding the publication of this paper.

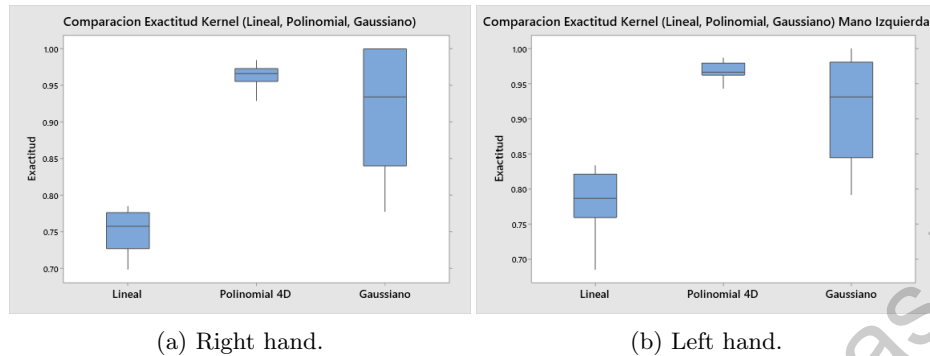


Fig. 9: Accuracy Comparison of Linear Kernel vs Gaussian Kernel vs Polynomial Kernel of order two up to order seven.

References

- [1] Martin Šavc, Vojko Glaser, Jernej Kranjec, Imre Cikajlo, Zlatko Matjačić, and Aleš Holobar. “Comparison of convolutive kernel compensation and non-negative matrix factorization of surface electromyograms”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.10 (2018), pp. 1935–1944.
- [2] Roberto Merletti and Dario Farina. “Surface EMG processing: Introduction to the special issue”. In: *Biomedical Signal Processing and Control* 3.2 (2008), pp. 115–117.
- [3] Sridhar P Arjunan, Dinesh K Kumar, and Ganesh R Naik. “A machine learning based method for classification of fractal features of forearm sEMG using twin support vector machines”. In: *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE. 2010, pp. 4821–4824.
- [4] Corinna Cortes and Vladimir Vapnik. “Support-vector networks”. In: *Machine learning* 20.3 (1995), pp. 273–297.
- [5] Enrique J Carmona Suárez. “Tutorial sobre máquinas de vectores soporte (sVM)”. In: *Tutorial sobre Máquinas de Vectores Soporte (SVM)* (2014), pp. 1–12.
- [6] Nello Cristianini, John Shawe-Taylor, et al. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.
- [7] Martin Hofmann. “Support vector machines-kernels and the kernel trick”. In: *Notes* 26.3 (2006).
- [8] Qitang Sun, Lujuan Dang, Wanli Wang, and Shiyuan Wang. “Kernel least mean square algorithm with mixed kernel”. In: *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*. IEEE. 2018, pp. 140–144.

- [9] Qingshan She, Zhizeng Luo, Ming Meng, and Ping Xu. “Multiple kernel learning SVM-based EMG pattern classification for lower limb control”. In: *2010 11th International Conference on Control Automation Robotics & Vision*. IEEE. 2010, pp. 2109–2113.
- [10] IJ Ramírez-Angeles. “Detección y clasificación de señales mioeléctricas en el brazo mediante el uso de algoritmos basados en inteligencia artificial”. Thesis. Maestría en Ciencias en Inteligencia Artificial. Universidad Autónoma de Querétaro, 2019.
- [11] MA Aceves-Fernandez, JM Ramos-Arreguin, E Gorrostieta-Hurtado, and JC Pedraza-Ortega. “Methodology Proposal of EMG Hand Movement Classification Based on Cross Recurrence Plots”. In: *Computational and Mathematical Methods in Medicine 2019* (2019).
- [12] Edgar David Monroy Jordan and Jairo Enrique Pérez Neira. “Máquinas de soporte vectorial (SVM)”. In: (2005).
- [13] Artemio Sotomayor-Olmedo, Marco A Aceves-Fernández, Efrén Gorrostieta-Hurtado, Carlos Pedraza-Ortega, Juan M Ramos-Arreguín, and J Emilio Vargas-Soto. “Forecast urban air pollution in Mexico City by using support vector machines: A kernel performance approach”. In: (2013).
- [14] Bassam Moslem, Mohamad Khalil, Mohamad O Diab, Aly Chkeir, and Catherine Marque. “Combining multiple support vector machines for boosting the classification accuracy of uterine EMG signals”. In: *2011 18th IEEE International Conference on Electronics, Circuits, and Systems*. IEEE. 2011, pp. 631–634.
- [15] Alexandre Kowalczyk. “Support vector machines succinctly”. In: *Synfusion Inc* (2017).
- [16] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. *A practical guide to support vector classification*. 2003.
- [17] Ana López Díaz. “Fundamentos matemáticos de los métodos Kernel para aprendizaje supervisado”. In: (2018).