

DISEÑO DE UNA INTERFAZ ELECTRÓNICA PARA EL
RECONOCIMIENTO DE PATRONES EMG PARA PRÓTESIS DE
MANO

FABIAN MAURICIO LEON VARGAS
JHONATAN CAMACHO NAVARRO

ESCUELA DE INGENIERIAS ELECTRICA, ELECTRONICA Y
TELECOMUNICACIONES
UNIVERSIDAD INDUSTRIAL DE SANTANDER
BUCARAMANGA
2008

DISEÑO DE UNA INTERFAZ ELECTRÓNICA PARA EL
RECONOCIMIENTO DE PATRONES EMG PARA PRÓTESIS DE
MANO

FABIAN MAURICIO LEON VARGAS
JHONATAN CAMACHO NAVARRO

Trabajo de grado para optar al título de Ingeniero Electrónico

Director

JAIME GUILLERMO BARRERO PÉREZ
MPE Ingeniero Electricista

Codirector

CAMILO LEONARDO SANDOVAL
Magíster (c) Ingeniero Electrónico

ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
UNIVERSIDAD INDUSTRIAL DE SANTANDER
BUCARAMANGA

2008

*Al padre todopoderoso que guió cada desición tomada
A mi madre Luz Mila por ser el ejemplo de mi vida
A mi padre Hermes por apoyarme y ser mi guia
A mi hermanita Katherine que quiero tanto*

Fabian Mauricio Leon Vargas

*A mi Diosito por permitirme alcanzar las metas
A mis padres por apoyarme siempre que lo necesito
A mi novia que quiero tanto por brindarme compañía
A mis amigos, profesores y familiares por participar en este triunfo*

Jhonatan Camacho Navarro

AGRADECIMIENTOS

Los autores expresan sus agradecimientos:

Al Magíster Jaime G. Barrero, director del proyecto, por supervisar y orientar el trabajo, promoviendo calidad, coherencia, buena redacción y el uso de la terminología apropiada

Al Magíster Cesar A. Duarte, por sus aportes oportunos y su firme convicción en la investigación.

AL Ing. Carlos Andres Niño, por su gran colaboración y apoyo logístico.

Al Dr Rodolfo Villamizar y el Dr Homero Ortega por las sugerencias y grán interes.

RESUMEN

TÍTULO:

DISEÑO DE UNA INTERFAZ ELECTRONICA PARA EL RECONOIMIENTO DE PATRONES EMG PARA PROTESIS DE MANO^{*}

AUTORES:

FABIAN MAURICIO LEON VARGAS
JHONATAN CAMACHO NAVARRO^{**}

PALABRAS CLAVE:

Electromiográfico, Prótesis, Reconocimiento de Patrones, Análisis de Componentes Principales, Transformada *Wavelet* Discreta, Maquinas de Soporte Vectorial.

DESCRIPCION:

Implementar sistemas para control de prótesis con alta funcionalidad demanda reconocer a partir de impulsos eléctricos los movimientos ejecutados. Este trabajo representa los resultados de la investigación realizada con el objetivo de identificar automáticamente, en pacientes sanos, los movimientos básicos de la mano: Apertura, Cierre, Flexión, Extensión, Pronación y Supinación, incluyendo la condición de reposo.

El proceso de discriminación se desarrolló como un problema de reconocimiento de patrones teniendo en cuenta la metodología modular presentada en los trabajos realizados por **Zecca**[†] y los doctores **Englehart**, **Parker** y **Hudgins**,[‡] donde se describe el esquema para abordar la clasificación de movimientos a partir del estado estable de señales electromiográficas superficiales (*EMG's*). Para lograr el objetivo se construyó una *interface hardware - software* que permitió crear registros de datos y clasificar *off line* cada 256 ms de señal adquirida. La *interface* constituye un sistema de adquisición para señales *EMG's* de dos canales diferenciales con algoritmos en *MATLAB*[®] y *LABVIEW*[®] que realizan el procesamiento digital. Mediante la Transformada *Wavelet* Discreta se obtiene información en tiempo y frecuencia para describir la actividad eléctrica muscular generada al realizar un movimiento, además realizando Análisis de Componentes Principales (*PCA*) se organiza el patrón *EMG* (electromiográfico) para la discriminación a través de métodos de inteligencia artificial, en este caso se usaron Maquinas de Soporte Vectorial.

Como resultado se definieron modelos que permiten realizar el análisis de forma automática; mostrando que utilizar *Symlet4*, cuarto nivel de resolución, valor *RMS* de los coeficientes de detalle y aproximación, descomposición *PCA* normalizada y criterio de reducción estándar formulado por los autores; permite identificar movimientos con tasas de acierto que alcanzan el 99.25%. También se especificaron las indicaciones para realizar el entrenamiento y validación del sistema de inteligencia artificial. Además, los resultados obtenidos son validos para personas de diferente género, edad y complexión física.

^{*} Proyecto de Investigación

^{**} Facultad de Ingenierías Físico Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Jaime G. Barrero P.

[†] ZECCA, S. MICERA, M. C. CARROZZA, & P. DARIO. Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal. ARTS Lab, Scuola Superiore Sant'Anna, Pontedera, Italy, 2002.

[‡] ENGLEHART K.; HUDGINS B.; PARKER P. A Wavelet-Based Continuous Classification Scheme for Multifunction Myoelectric Control. IEEE transactions on biomedical engineering, vol. 48, no. 3, March 2001.

ABSTRACT

ABSTRACT

TITLE:

DESIGN OF AN ELECTRONIC INTERFACE FOR EMG PATTERN RECOGNITION FOR PROSTHETIC HAND^{*}

AUTHORS:

FABIAN MAURICIO LEON VARGAS
JHONATAN CAMACHO NAVARRO^{**}

KEYWORD:

Electromyography, Prosthetic hand, Pattern Recognition, Principal Component Analysis, Discrete Wavelet Transform, Support Vector Machines.

DESCRIPTION:

Implement systems to control prostheses with high functionality demand recognize from electrical impulses movements executed. This work represents the results of research carried out in order to identify automatically, in healthy patients, the basic movements of the hand: Opening, Closing, Flexion, Extension, Pronation and Supination, including the condition of none movement.

The discrimination process was developed as a problem of pattern recognition taking into account the modular methodology presented in work performed by **Zecca**[†] and the doctors **Englehart**, **Hudgins** and **Parker**[‡] which describes the scheme to address the classification of movements from the state stable signal surface electromyography (*EMG*'s). To achieve the goal was constructed an interface hardware - software that created data records and classify off line every 256 ms signal gained. The interface is a system for acquisition signals *EMG*'s two-channel differential and algorithms in *MATLAB*[®] and *LabVIEW*[®] performing digital processing. By Discrete Wavelet Transform information is obtained in time and frequency to describe the electrical activity generated muscle to make a movement, also by Principal Component Analysis (*PCA*) is organized pattern *EMG* (electromyographic) for discrimination through intelligence methods artificial, in this case were used Support Vector Machines.

As a result defined models that allow the analysis automatically; showing that use *Symlet4*, fourth level of resolution, RMS value of the coefficients of detail and approach, *PCA* decomposition standardized approach and reduction standard formulated by the authors; identifies movements with success rates reaching 99.25%. It also specified the indications for conducting training and validation system of artificial intelligence. In addition, the results are valid for people of different gender, age and physical complexion.

^{*} Research Project

^{**} Faculty of Mechanical Engineering Physicist. School of Electrical Engineering, Electronics and Telecommunications. Jaime G. Barrero P.

[†] ZECCA, S. MICERA, M. C. CARROZZA, & P. DARIO. Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal. ARTS Lab, Scuola Superiore Sant'Anna, Pontedera, Italy, 2002.

[‡] ENGLEHART K.; HUDGINS B.; PARKER P. A Wavelet-Based Continuous Classification Scheme for Multifunction Myoelectric Control. IEEE transactions on biomedical engineering, vol. 48, no. 3, March 2001.

Índice general

	pág.
1. Introducción	1
1.1. Estado del Arte	2
2. Metodología desarrollada	4
3. Sistema de Adquisición de señales EMG	12
3.1. Etapa 1	13
3.2. Etapa 2	14
3.3. Etapa 3	16
3.3.1. Tarjeta de Adquisición	16
3.3.2. Interfaz gráfica del Electromiógrafo	17
3.4. Elección de los componentes de la tarjeta	19
3.5. Adquisición de señales EMG con equipo BIOPAC	20
4. Interfaz Investigación de Patrones EMG	22
4.1. Fase de Entrenamiento	22
4.2. Fase de Prueba	24
4.3. Consideraciones	25
5. Resultados	27
5.1. Metodología desarrollada en la prueba	27
5.2. Pruebas realizadas	27
5.2.1. Prueba Preliminar	27
5.2.1.1. Resultados Prueba Preliminar de MSV's	29

INDICE GENERAL

5.2.2.	Prueba de Validación	32
5.3.	Discusión	38
5.4.	Interfaz práctica	39
6.	Conclusiones, Aportes y Trabajos Futuros	42
6.1.	Conclusiones	42
6.2.	Aportes	44
6.3.	Trabajo Futuro	45
	Bibliografía	46
A.	Generación de biopotenciales	50
A.1.	Bioelectricidad	50
A.2.	Potenciales de reposo y acción	50
A.3.	Electromiograma (EMG)	53
B.	Fisiología del antebrazo	55
B.1.	Músculos del antebrazo	55
C.	La transformada Wavelet	61
C.1.	Introducción	61
C.2.	Generalidades	62
C.2.1.	Espacios de Hilbert	62
C.2.2.	Ortogonalidad Bases Ortonormales	63
C.2.3.	Análisis de Fourier	63
	▪ Coeficientes y Series de Fourier	64
	▪ Transformada de Fourier	65
	▪ Limitaciones del análisis de Fourier	66
	▪ Transformada de Fourier con ventana	67
C.2.4.	Bases ortonormales	68

INDICE GENERAL

▪ Bases de la función escala	68
C.2.5. Análisis multiresolución	70
C.2.6. Bases Wavelet	70
C.3. Transformada Wavelet	71
C.3.1. Wavelets ortonormales y discretas	72
C.3.2. Relación dos – escala	74
C.3.3. Algoritmo piramidal - Descomposición wavelet	75
C.3.4. Algoritmo piramidal – Reconstrucción	78
C.4. Análisis tiempo – frecuencia	80
C.4.1. Escala y frecuencia	82
C.4.2. La naturaleza de escala	82
C.4.3. Número de operaciones	83
D. Selección de la wavelet	84
D.1. Introducción	84
D.2. Antecedentes de Metodologías planteadas	85
D.3. metodologías propuestas por los autores	85
D.3.1. Uso de la correlación cruzada	86
D.3.2. Uso de la Discret Wavelet Transform para elegir wavelet madre	87
D.3.3. Observaciones	88
E. Análisis de Componentes Principales	89
E.1. Introducción	89
E.2. Procedimiento	91
E.2.1. Obtener los datos	91
E.2.2. Estandarizar las variables originales	92
E.2.3. Calcular la matriz de covarianza	93
E.2.4. Calcular los valores y vectores propios.	94
E.2.5. Elegir componentes	95

INDICE GENERAL

E.2.6. Derivar el nuevo conjunto de datos	99
E.2.7. Volviendo al conjunto de datos originales	100
F. Maquinas de Soporte Vectorial	101
F.1. Introducción	101
F.2. Técnica de Aprendizaje en las MSV lineales	102
F.3. Datos Linealmente Separables	103
F.4. Hiperplano de separación optimo	104
F.5. Maximización del margen	104
F.6. Tolerancia al ruido de las MSVs	108
F.7. MSVs no lineales – Una generalización	110
F.8. Multi clasificación con MSV	113
F.8.1. Maquinas multi clasificadoras	114
F.8.2. Maquinas bi-clasificadoras generalizadas	114
F.9. Arquitecturas de descomposición normalizadas	115
F.9.1. Uno contra el resto (1-V-r)	115
F.9.2. Uno contra uno (O-V-O)	116
F.10. Métodos de reconstrucción	118
F.10.1. Clasificación por parejas	121
F.10.2. Salidas Numéricas Probabilísticas	122
F.10.3. Función decisión final	122
F.11. Validación cruzada en la MSV	122
G. Descripción interna de Interfaz Investigación de Patrones EMG	124
G.1. Formato de datos	124
G.2. Funciones	125
G.2.1. ProcesW	125
G.2.2. PCAmodelo	125
G.2.3. procesPCA	126

INDICE GENERAL

G.2.4. clasificarMSV	126
G.3. Formato en estructuras internas	126
G.3.1. Entrenamiento	127
G.3.2. Prueba	128
G.4. Instalación de InterfazEJECUCION	130
H. Tablas de resultado en fase de entrenamiento de prueba preliminar	131
H.1. Introducción	131
H.2. Tablas de resultados	131
H.3. Discusion	150
H.4. Tablas de selección final de la wavelet	150

Lista de tablas

	pág.
2.1. Parámetros para el diseño de los módulos en la identificación de movimientos	6
2.2. Rango de frecuencias analizadas en cada nivel de resolución	8
2.3. Asignación de etiquetas a cada modelo generado	11
3.1. Resumen de especificaciones para la Tarjeta de adquisición LAB-PC1200	17
3.2. Características del amplificador diferencial AD620	19
3.3. Características del amplificador operacional OP200	19
4.1. Orden en el que se deben almacenar los movimientos en el registro total de acuerdo a las clases etiquetadas	26
5.1. Rangos de variación de los parámetros de las MSVs en entrenamiento	28
5.2. Porcentajes de error en clasificación para fase de prueba con la Wavelet Coiflet 3	29
5.3. Porcentajes de error en clasificación para fase de prueba con la Wavelet Coiflet 4	30
5.4. Porcentajes de error en clasificación para fase de prueba con la wavelet Daubechies 4	30
5.5. Porcentajes de error en clasificación para fase de prueba con la Wavelet Daubechies 5	31
5.6. Porcentajes de error en clasificación para fase de prueba Con la Wavelet Symlet 4	31

5.7.	Porcentajes de error en clasificación para fase de prueba con la <i>Wavelet Symlet 6</i>	32
5.8.	Porcentaje de errores Total y por cada movimiento usando <i>Coiflet 3</i>	33
5.9.	Porcentaje de errores Total y por cada movimiento usando <i>Coiflet 4</i>	34
5.10.	Porcentaje de errores Total y por cada movimiento usando <i>Daubechies 4</i>	35
5.11.	Porcentaje de errores Total y por cada movimiento usando <i>Daubechies 5</i>	36
5.12.	Porcentaje de errores Total y por cada movimiento usando <i>Symlet 4</i>	37
5.13.	Porcentaje de errores Total y por cada movimiento usando <i>Symlet 6</i>	38
5.14.	Parámetros del modelo con mejor desempeño en la investigación ...	39
5.15.	Parámetros de entrenamiento de la MSV recomendados	40
5.16.	Instrucciones de uso como Interfaz Práctica	41
D.1.	Rango de frecuencias analizadas en cada nivel de resolución	87
G.1.	Formato de señales EMG	124
G.2.	Estructura de configuración en modelamiento	127
G.3.	Estructura de configuración de cada modeloW de entrada	128
G.4.	Estructura de configuración del modelo MSV	129
G.5.	Estructura de configuración de modeloPCA	130
H.1.	Resultados de Wavelet <i>Daubechies 4</i> con resolución hasta el cuarto nivel usando <i>kernel Polinomico</i>	132
H.2.	Resultados de Wavelet <i>Daubechies 4</i> con resolución hasta el cuarto nivel usando <i>kernel RBF</i>	132
H.3.	Resultados de Wavelet <i>Daubechies 4</i> con resolución hasta el cuarto nivel usando <i>kernel Sigmoidal</i>	133
H.4.	Resultados de Wavelet <i>Daubechies 4</i> con resolución	

LISTA DE TABLAS

	<u>hasta el quinto nivel usando <i>kernel</i> Polinómico.....</u>	133
H.5.	<u>Resultados de Wavelet Daubechies 4 con resolución hasta el quinto nivel usando <i>kernel</i> RBF.....</u>	134
H.6.	<u>Resultados de Wavelet Daubechies 4 con resolución hasta el quinto nivel usando <i>kernel</i> Sigmoidal.....</u>	134
H.7.	<u>Resultados de Wavelet Daubechies 5 con resolución hasta cuarto nivel usando <i>kernel</i> Polinomico.....</u>	135
H.8.	<u>Resultados de Wavelet Daubechies 5 con resolución hasta cuarto nivel usando <i>kernel</i> RBF.....</u>	135
H.9.	<u>Resultados de Wavelet Daubechies 5 con resolución Hasta cuarto nivel usando <i>kernel</i> Sigmoidal.....</u>	136
H.10.	<u>Resultados de Wavelet Daubechies 5 con resolución hasta quinto nivel usando <i>kernel</i> Polinomico.....</u>	136
H.11.	<u>Resultados de Wavelet Daubechies 5 con resolución hasta quinto nivel usando <i>kernel</i> RBF.....</u>	137
H.12.	<u>Resultados de Wavelet Daubechies 5 con resolución hasta quinto nivel Usando <i>kernel</i> Sigmoidal.....</u>	137
H.13.	<u>Resultados de Wavelet Coiflet 3 con resolución hasta cuarto nivel usando <i>kernel</i> Polinomico.....</u>	138
H.14.	<u>Resultados de Wavelet Coiflet 3 con resolución hasta cuarto nivel usando <i>kernel</i> RBF.....</u>	138
H.15.	<u>Resultados de Wavelet Coiflet 3 con resolución Hasta cuarto nivel usando <i>kernel</i> Sigmoidal.....</u>	139
H.16.	<u>Resultados de Wavelet Coiflet 3 con resolución hasta quinto nivel usando <i>kernel</i> Polinomico.....</u>	139
H.17.	<u>Resultados de Wavelet Coiflet 3 con resolución hasta quinto nivel usando <i>kernel</i> RBF.....</u>	140
H.18.	<u>Resultados de Wavelet Coiflet 3 con resolución hasta quinto nivel usando <i>kernel</i> Sigmoidal.....</u>	140
H.19.	<u>Resultados de Wavelet Coiflet 4 con resolución hasta cuarto nivel usando <i>kernel</i> Polinomico.....</u>	141
H.20.	<u>Resultados de Wavelet Coiflet 4 con resolución hasta cuarto nivel usando <i>kernel</i> RBF.....</u>	141
H.21.	<u>Resultados de Wavelet Coiflet 4 con resolución</u>	

LISTA DE TABLAS

[hasta cuarto nivel usando *kernel* Sigmoidal](#).....142

H.22. [Resultados de Wavelet Coiflet 4 con resolución hasta el quinto nivel usando *kernel* Polinomico](#)142

H.23. [Resultados de Wavelet Coiflet 4 con resolución hasta quinto nivel usando *kernel* RBF](#)..... 143

H.24. [Resultados de Wavelet Coiflet 4 con resolución hasta quinto nivel usando *kernel* Sigmoidal](#)143

H.25. [Resultados de Wavelet Symlet 4 con resolución hasta cuarto nivel *kernel* Polinomico](#)..... 144

H.26. [Resultados de Wavelet Symlet 4 con resolución hasta cuarto nivel usando *kernel* RBF](#)..... 144

H.27. [Resultados de wavelet Symlet 4 con resolución hasta cuarto nivel usando *kernel* Sigmoidal](#)..... 145

H.28. [Resultados de Wavelet Symlet 4 con resolución hasta quinto nivel *kernel* Polinomico](#)145

H.29. [Resultados de wavelet Symlet 4 con resolución hasta quinto nivel usando *kernel* RBF](#)146

H.30. [Resultados de Wavelet Symlet 4 con resolución hasta quinto nivel usando *kernel* Sigmoidal](#)146

H.31. [Resultados de Wavelet Symlet 6 con resolución hasta cuarto nivel *kernel* Polinomico](#)147

H.32. [Resultados de Wavelet Symlet 6 con resolución hasta cuarto nivel usando *kernel* RBF](#)147

H.33. [Resultados de Wavelet Symlet 6 con resolución hasta cuarto nivel usando *kernel* Sigmoidal](#)148

H.34. [Resultados de Wavelet Symlet 6 con resolución hasta quinto nivel *kernel* Polinomico](#)..... 148

H.35. [Resultados de Wavelet Symlet 6 con resolución hasta quinto nivel usando *kernel* RBF](#)149

H.36. [Resultados de Wavelet Symlet 6 con resolución hasta quinto nivel usando *kernel* Sigmoidal](#)149

H.37. [Maquinas seleccionadas para fase de prueba Coiflet 3](#)151

H.38. [Maquinas seleccionadas para fase de prueba Coiflet 4](#)151

H.39. [Maquinas seleccionadas para fase de prueba Symlet 4](#)152

LISTA DE TABLAS

H.40. Maquinas seleccionadas para fase de prueba Symlet 6.	152
H.41. Maquinas seleccionadas para fase de prueba Daubechies 4.	153
H.42. Maquinas seleccionadas para fase de prueba Daubechies 5.	153

Lista de figuras

	pág.
2.1. Grados de libertad identificados	4
2.2. Módulos para identificación de movimientos	5
2.3. Pirámide de variación de parámetros.	7
2.4. Parámetros evaluados para una Wavelet y un Nivel de resolución específico	10
3.1. Esquema general por etapas del diseño implementado.	12
3.2. Circuito implementado en la adquisición de ECG adaptado a EMG	13
3.3. Filtro Butterworth pasa-alta de 20Hz. Configuración Sallen-Key.	14
3.4. Filtro Butterworth pasa-baja de 500 Hz. Configuración Sallen-Key.	15
3.5. Filtro ranura de 60 Hz	15
3.6. Tarjeta de adquisición LAB-PC 1200.	16
3.7. Tarjeta de acondicionamiento de señales EMG.	17
3.8. Diagrama de bloques del electromiógrafo.	18
3.9. Interfaz gráfica del electromiógrafo.	18
3.10. Señal EMG	20
3.11. Equipo BIOPAC (a) Hardware. (b) Software.	21
3.12. Disposición de los electrodos en región anterior y posterior del antebrazo	21
4.1. Interfaz gráfica usada en el reconocimiento de patrones EMG	23
4.2. Interfaz gráfica usada para evaluar la cantidad de modelos iniciales en el reconocimiento de patrones EMG	24

4.3.	<u>Clasificación de las señales de prueba con uno de los pacientes.....</u>	25
A.1.	<u>Célula polarizada con su potencial de reposo.</u>	51
A.2.	<u>Despolarización de una célula</u>	52
A.3.	<u>Célula despolarizada durante un potencial de acción</u>	52
A.4.	<u>Forma de onda del potencial de acción</u>	53
A.5.	<u>Espectro de frecuencia de la señal EMG</u>	54
B.1.	<u>Músculos <i>pronador terete</i> y <i>pronador cuadrado</i></u>	56
B.2.	<u>Músculos <i>flexor radial del carpo</i>, <i>palmar largo</i>, <i>flexor ulnar del carpo</i>.....</u>	56
B.3.	<u>Músculo <i>flexor superficial de los dedos</i>.....</u>	57
B.4.	<u>Músculos <i>flexor profundo de los dedos</i>, <i>flexor largo del pulgar</i></u>	57
B.5.	<u>Músculo <i>braquiorradial</i></u>	58
B.6.	<u>Músculos <i>extensor radial largo</i>, <i>extensor radial breve</i>, <i>extensor ulnar</i>.....</u>	59
B.7.	<u>Músculos <i>extensores de los dedos</i>, <i>extensor del dedo mínimo</i>.</u>	59
B.8.	<u>Músculo <i>supinador</i></u>	60
C.1.	<u>Proceso de análisis de Fourier.</u>	63
C.2.	<u>Forma de onda de ecuación Gaussiana.</u>	67
C.3.	<u>Proceso de análisis de Fourier de tiempo corto.</u>	68
C.4.	<u>Proceso de análisis Wavelet con filtros <i>pasa-bajo</i> y <i>pasa-alto</i>.</u>	77
C.5.	<u>Esquema de descomposición en Series Wavelet, <i>algoritmo piramidal</i>.</u>	78
C.6.	<u>Banco de filtros</u>	79
C.7.	<u>Esquema de la reconstrucción Wavelet.</u>	79
C.8.	<u>Esquemas de los distintos espacios de operación para análisis.</u>	80
C.9.	<u>Calculo de <i>C</i></u>	80
C.10.	<u>Desplazamiento</u>	81

C.11.	Dilatación	81
C.12.	Representación de transformada Wavelet en el espacio escala-tiempo	81
C.13.	Escalamiento de la función wavelet	82
E.1.	Formato de datos para realizar análisis de componentes principales	90
E.2.	Vectores para realizar análisis de componentes principales	91
E.3.	Matriz de datos centrados	93
E.4.	Matriz de covarianza	94
E.5.	Gráfica de codo para seleccionar el número de componentes	98
F.1.	Hiperplanos que separan correctamente los datos	103
F.2.	Margen de separación entre dos conjuntos de datos	104
F.3.	Hiper plano lineal clasificador para el caso no separable	109
F.4.	Transformación de los datos originales	111
F.5.	Diagrama de multi clasificación	114
F.6.	Módulos del proceso de multi clasificación	115
F.7.	Esquema de votación por parejas	121

Capítulo 1

Introducción

Con el reciente avance tecnológico en el área de la bioingeniería se ha logrado dar solución a muchos de los problemas que afectan la integridad física del hombre, tal es el caso de los sistemas protésicos, dispositivos tecnológicos con la capacidad de simular en cierto grado la complejidad del miembro ausente. En particular, las prótesis eléctricas para miembro superior evidencian la necesidad de satisfacer mucho más el rehabilitamiento de las funciones motoras perdidas, que por lo general a nivel comercial cubren un solo grado de libertad (DoF, *Degree of Freedom*) limitando el desempeño de la persona en sus actividades diarias.

Las señales empleadas para el control de prótesis con mas DoFs que no requieren una intervención directa con los músculos encargados de los movimientos son adquiridas a través de la superficie del antebrazo por medio de electrodos ubicados sobre la piel que convierten impulsos nerviosos en señales eléctricas, llamadas señales EMG (electromiográficas), ver [Anexo A](#).

El enfoque presentado en este proyecto de grado toma como referencia los trabajos realizados por *Englehart et al* [18], y centra su atención en la implementación de la **Transformada Wavelet Discreta (DWT, *Discret Wavelet Transform*)** como técnica para extraer información relevante de las señales EMG, así como el uso del **Análisis de Componentes Principales (PCA, *Principal Components Analysis*)** para reducir la dimensión del vector característico (patrón) que será ingresado al algoritmo de inteligencia artificial, **Maquinas de Soporte Vectorial (SVM, *Support Vector Machine*)**, para su clasificación.

Es necesario que el lector este familiarizado con las herramientas matemáticas usadas en este proyecto de grado, por tal razón, en los anexos [C](#), [E](#) y [F](#) se encuentra la fundamentación para entender cada una de ellas.

1.1. Estado del arte

Las señales electromiográficas (EMG), recogidas en la superficie de la piel, se han utilizado para el control de prótesis para extremidad superior desde 1948, porque proporcionan acceso fácil y no invasivo a los procesos fisiológicos que causan la contracción de los músculos. En la actualidad, el procesamiento de señales EMG es el método más común usado para el control de prótesis de mano activas. Sin embargo, la señal mioeléctrica permite el control de no más de uno o dos DoFs. Limitaciones en la mecánica de los dispositivos de prótesis y en el procesamiento de datos EMG han hecho imposible el control de más [\[17\]](#).

En las últimas décadas, y especialmente durante los últimos años, muchos esfuerzos se han llevado a cabo con el fin de aplicar un control efectivo con base en algoritmos para el procesamiento de señales EMG. A partir de los primeros intentos de los fines de la década de 1940, varios algoritmos basados en señales EMG se han desarrollado y utilizado para mejorar la funcionalidad y facilidad de uso de prótesis de manos [\[17\]](#).

El esquema formal para la adquisición y análisis de la señal EMG para el control de la prótesis se compone de varios módulos:

1. Acondicionamiento y preprocesamiento de la señal.
2. Extracción de características.
3. Reducción de dimensionalidad.
4. Reconocimiento de patrones.

El primer módulo preprocesa la señal EMG con el fin de reducir el ruido electromagnético y acondiciona los componentes espectrales que contienen información importante para el análisis de datos.

Durante la fase de extracción de características (módulo 2), la señal EMG es procesada con el fin de enfatizar las estructuras pertinentes de los datos, mientras se rechaza el ruido y los datos irrelevantes, produciendo el llamado "conjunto de características originales".

Una reducción de la dimensionalidad (módulo 3) es necesaria para simplificar la tarea del clasificador.

En este caso, un algoritmo de reconocimiento de patrones (módulo 4) utiliza el conjunto reducido de características, y la señal medida es clasificada en el espacio de salida [\[17\]](#).

En cada uno de estos módulos es posible encontrar variaciones según el interés de la investigación.

Investigaciones realizadas por universidades a nivel mundial como la *University of New Brunswick*, han desempeñado un rol trascendental en la solución de este problema para tres DoFs. *Englehart, Hudgins, Parker*, entre otros, profundizaron en cada uno de los módulos propuestos y establecieron métodos para lograr una exactitud superior al 90% y en algunos casos muy cercana al 100%.

Hasta el día de hoy se siguen realizando investigaciones que buscan establecer métodos que requieran menores prestaciones tanto en *Hardware* como en *Software* manteniendo una exactitud pertinente en el proceso de clasificación.

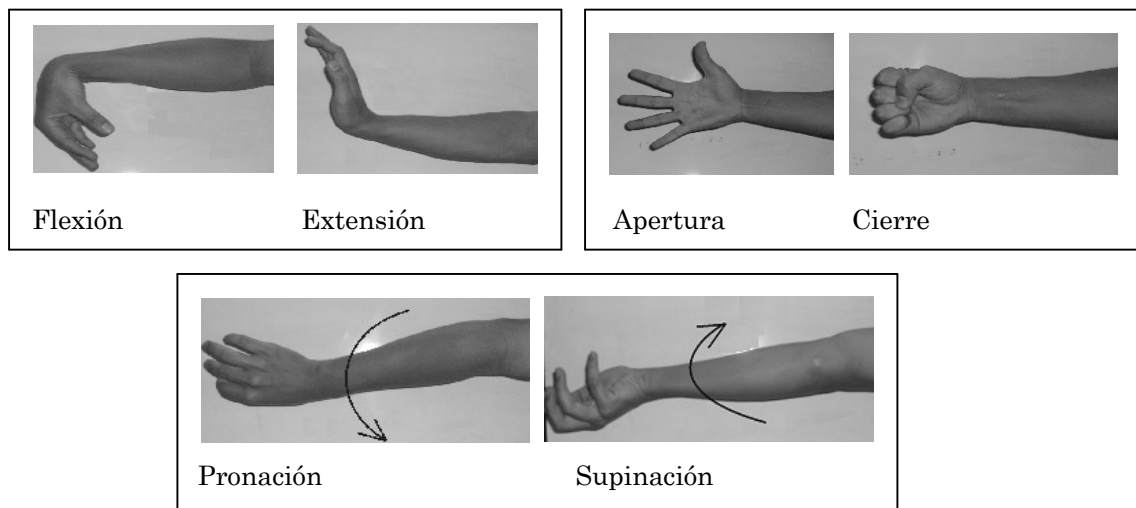
Capítulo 2

Metodología desarrollada

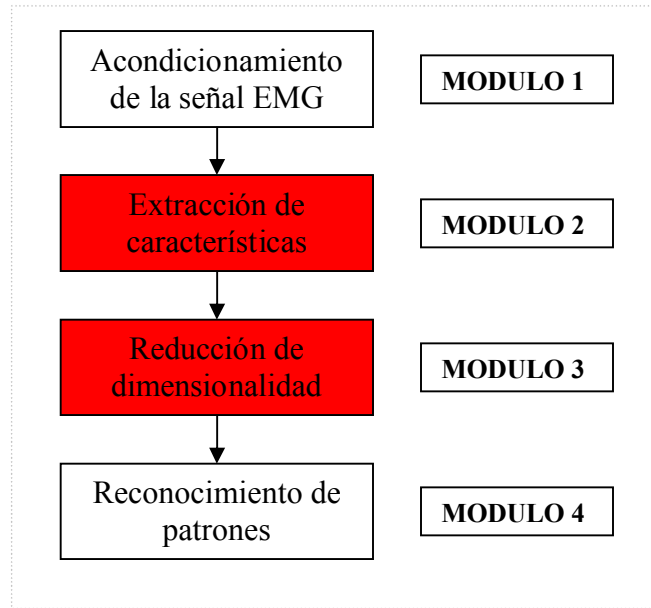
A partir del estado del arte descrito en el [capítulo 1](#), se elige trabajar con una metodología por módulos. En la [figura 2.1](#) se presentan los tres DoFs identificados en la presente investigación.

La [figura 2.2](#) muestra el contenido en cada uno de los módulos y resalta con fondo rojo aquellos basados en la metodología diseñada por *Englehart et al* [\[18\]](#). La [tabla 2.1](#) detalla los parámetros tenidos en cuenta en cada módulo.

Figura 2.1. Grados de libertad identificados



Fuente autores

Figura 2.2. Módulos para identificación de movimientos

Fuente autores

El modulo 1 fue desarrollado con dos canales de adquisición para simplificar la instrumentación y reducir la cantidad de cálculos en los módulos posteriores. Cabe señalar, que esta configuración de sólo dos canales no ha sido utilizada para identificar seis movimientos en conjunto con los módulos restantes. La tarjeta de adquisición diseñada en la presente investigación se describe en el [capítulo 3](#). El módulo 4 utiliza MSV como algoritmo de inteligencia artificial para la clasificación de los movimientos. La fase de entrenamiento fue desarrollada por completo en la *toolbox* **MSVtoolbox10** diseñada en la Universidad Industrial de Santander por [\[6\]](#) y [\[23\]](#). Sin embargo, los módulos 2 y 3 diseñados por [\[18\]](#) no especifican los parámetros relevantes para cada uno. En [\[18\]](#) no se detalla hasta que nivel de resolución fue calculada la DWT ni los estadísticos calculados en cada uno de ellos, tampoco detalla el tipo de análisis realizado a los datos con PCA para su respectiva reducción de dimensionalidad ni el criterio usado para establecer la cantidad de componentes que forman el patrón final.

Tabla 2.1. Parámetros para el diseño de los módulos en la identificación de movimientos

MODULO	DESCRIPCIÓN	RAZON
1	Dos canales	Reduce la instrumentación y los cálculos realizados en los módulos posteriores.
	Ancho de banda entre 20-500Hz	Es el rango de frecuencias que contiene información relevante de las señales EMG [15], [18].
	Frecuencia de muestreo 1000Hz	Permite extraer información en frecuencia hasta los 500Hz según <i>Nyquist</i> .
	Electrodos de superficie desechables	Fáciles de conseguir, económicos, y cumplen las recomendaciones impuestas en [1] para la adquisición de señales EMG.
	Ubicación de los electrodos longitudinal y en la zona medial del músculo	Recomendada por [1]. En el anexo B se presenta la fisiología del antebrazo que se tiene en cuenta para la ubicación precisa de los electrodos sobre el antebrazo.
	Recolección de la señal EMG en estado estable	De acuerdo con [18] produce la mejor exactitud en clasificación, permite descartar la detección del inicio del movimiento.
	Adquisición diferencial de la señal EMG	Permite obtener un mejor desempeño en la clasificación según pruebas realizadas en [18]
2	Transformada <i>Wavelet</i> discreta	En [18] se afirma que de todas las herramientas matemáticas probadas por ellos, la DWT y la transformada <i>Wavelet Packet</i> fueron las que mejores resultados arrojaron en clasificación, DWT requiere menos complejidad de calculo que <i>Wavelet Packet</i> .
	Calculo de la DWT a registros cuya dimensión es de 256 muestras.	Pruebas realizadas en [18] establecen que longitudes de 256 muestras son idóneas para realizar los cálculos de la DWT.
3	Análisis de componentes principales	Pruebas realizadas por [18] establecieron que reducción de dimensionalidad con PCA presenta mejores resultados en clasificación.
4	Maquinas de soporte vectorial	Alternativa de inteligencia artificial que no ha sido probado con los módulos precedentes.

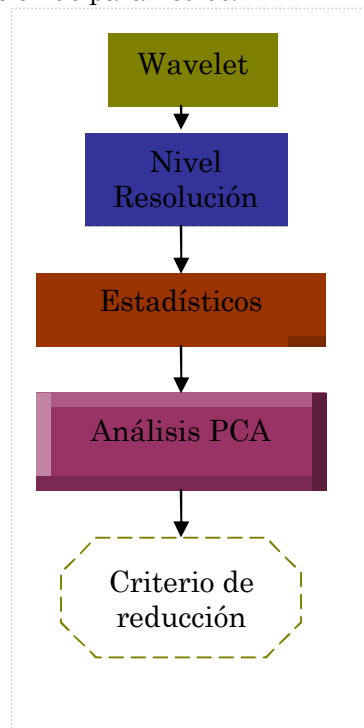
Fuente autores

En la [figura 2.3](#) se muestra la pirámide de parámetros determinados para arrojar resultados pertinentes en la clasificación. Tales parámetros se describen a continuación:

- **Wavelet:** No todas las *Wavelet* extraen la información de mayor relevancia para la clasificación.

- **Nivel de resolución de la DWT:** El nivel de resolución hasta el cual se calcula la DWT puede contener información redundante o poca información para el proceso de clasificación.
- **Estadísticos:** Representan la información contenida en la DWT para cada nivel de resolución calculado, forman el conjunto de características originales.
- **Análisis PCA:** De acuerdo con la forma que es analizada la información de los datos originales se normaliza o no la variabilidad de los estadísticos calculados para convertir el conjunto de características originales al espacio de salida de PCA.
- **Criterio de reducción:** Establece cuantas componentes se deben tomar luego de realizar PCA.

Figura 2.3. Pirámide de variación de parámetros.



Fuente autores

La selección de las *Wavelets* madre que fueron contempladas en la búsqueda de aquella con mejor desempeño para la clasificación, tuvo en cuenta dos métodos propuestos en la presente investigación, ver [anexo E](#), que intentaron medir

cuantitativamente el parecido de la señal EMG con diferentes *Wavelet* madre*. A pesar de ello no fue posible obtener resultados confiables, razón por la cual el grupo de *Wavelets* madre contemplado se realizó mediante una prueba piloto descrita en el [capítulo 5](#).

Para establecer cual nivel de resolución de la DWT tomar se contemplaron dos alternativas. La primera fue calcular la DWT hasta el cuarto nivel de resolución teniendo en cuenta que la energía dominante de la señal EMG se encuentra en el rango de 50 Hz a 150 Hz, ver [anexo A](#). La segunda consistió en tomar un nivel de resolución más, es decir, calcular la DWT hasta el quinto nivel de resolución en caso tal que hasta el cuarto nivel no fuera suficiente. La [tabla 2.2](#) muestra el rango de frecuencias que es analizado en cada nivel de resolución calculado por la DWT.

Tabla 2.2. Rango de frecuencias analizadas en cada nivel de resolución.

Nivel de resolución	Rango de frecuencias en Hz
1er Nivel	250←→500
2do Nivel	125←→250
3er Nivel	62.5←→125
4to Nivel	31.25←→62.5
5to Nivel	15.625←→31.25

Fuente autores

Los estadísticos calculados en cada nivel de resolución para la extracción de características son:

- Valor RMS
- Valor medio
- Varianza
- Desviación estándar

Adicionalmente se realizó todo el proceso de extracción de características con sólo el valor RMS. La razón se basó en determinar si era posible clasificar pertinentemente los movimientos utilizando tan sólo un estadístico.

* Esta fue la observación propuesta por [\[18\]](#) para explicar el buen desempeño de algunas *Wavelet* en el proceso de clasificación.

Teniendo en cuenta que el análisis PCA se contempla de dos formas diferentes, normalizado* o sin normalizar†, ambas alternativas se calculan para convertir la información del conjunto de características originales al espacio de salida del PCA.

Finalmente para determinar el criterio que establece la cantidad idónea de componentes que conforman el patrón final ingresado a la MSV, se trabajaron dos criterios, kaiser y estándar, ambos criterios se explican en el [anexo E](#).

La [figura 2.4](#) muestra el árbol de variaciones que se genera para una Wavelet y un nivel de resolución de la DWT particular teniendo en cuenta los parámetros descritos previamente.

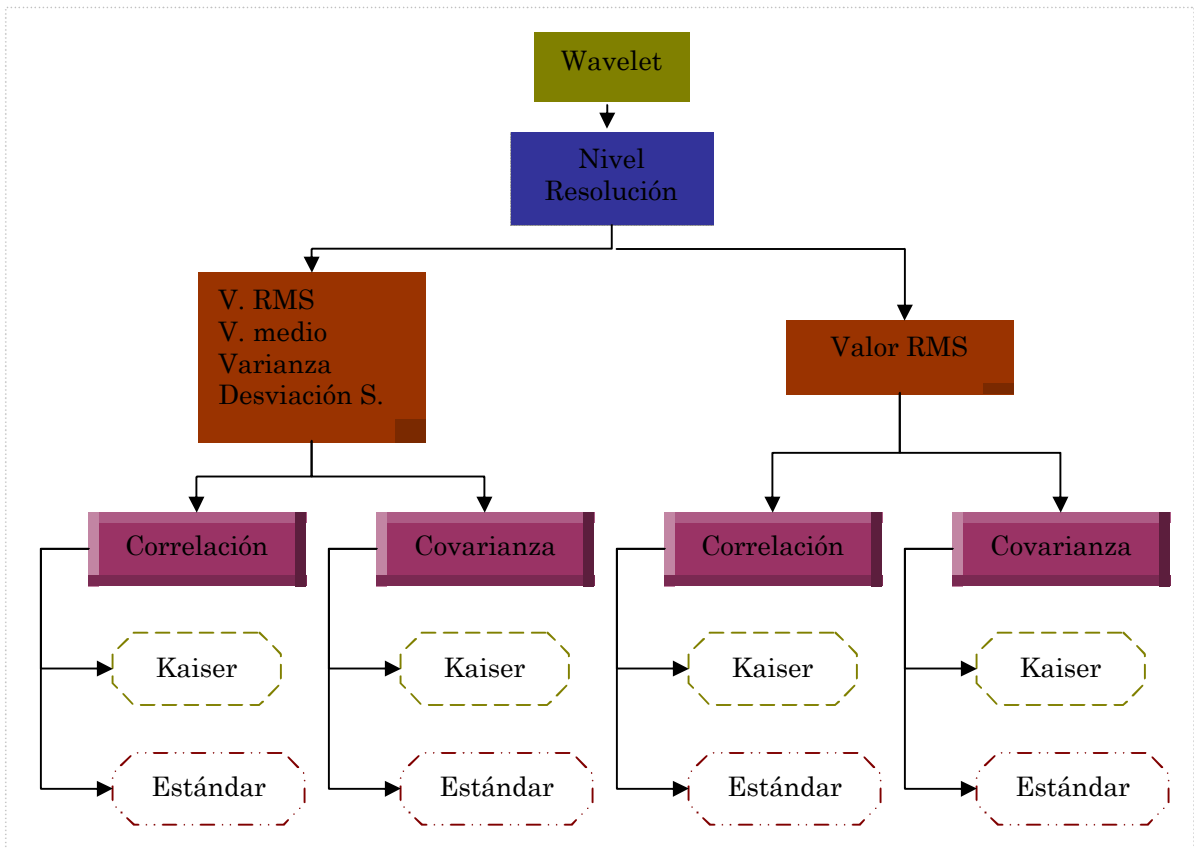
La cantidad de modelos que resultan al establecer una *Wavelet* y un nivel de resolución de la DWT en particular, finalmente son 8, tal como se aprecia en la [figura 2.4](#). Para identificar cada modelo generado se asigna una etiqueta, la [tabla 2.3](#) muestra la asignación.

Debido a la gran cantidad de cálculos y la complejidad que implica una realización confiable de los mismos, se diseñó una herramienta *software* descrita en el [capítulo 4](#), con la flexibilidad para realizar variaciones a los parámetros recién descritos, no detallados en [\[18\]](#).

* También conocido como análisis PCA por correlación, [ver anexo E](#).

† También conocido como análisis PCA por covarianza, [ver anexo E](#).

Figura 2.4. Árbol de parámetros evaluados para una Wavelet y hasta un Nivel de resolución de la DWT específico.



Fuente autores

Tabla 2.3. Asignación de etiquetas a cada modelo generado

Etiqueta	Características	PCA	Criterio
Modelo 1	Valor RMS	Correlación	Kaiser
Modelo 2	Valor RMS	Correlación	Estándar
Modelo 3	Valor RMS	Covarianza	Kaiser
Modelo 4	Valor RMS	Covarianza	Estándar
Modelo 5	Valor RMS Valor medio Varianza Desviación estándar	Correlación	Kaiser
Modelo 6	Valor RMS Valor medio Varianza Desviación estándar	Correlación	Estándar
Modelo 7	Valor RMS Valor medio Varianza Desviación estándar	Covarianza	Kaiser
Modelo 8	Valor RMS Valor medio Varianza Desviación estándar	Covarianza	Estándar

Fuente Autores

Capítulo 3

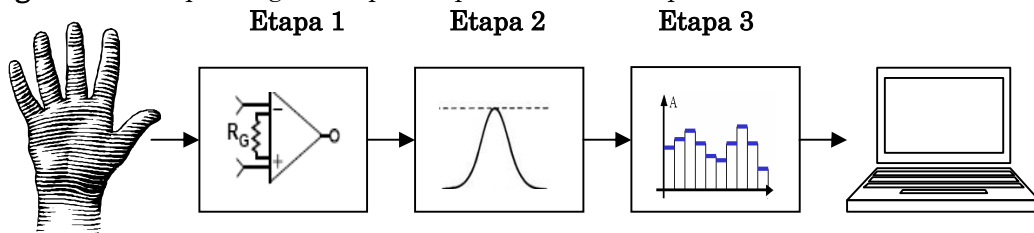
Sistema de adquisición de señales EMG

El diseño de la tarjeta para la adquisición de las señales EMG ([Figura 3.1](#)), necesaria para el desarrollo del módulo 1 descrito en [capítulo 2](#), esta basado en el funcionamiento del electrocardiógrafo común que se encuentra ampliamente documentado en la literatura a nivel mundial [\[4\]](#).

Para adaptar el modelo del ECG a señales EMG, es fundamental conocer las características tanto en amplitud como en frecuencia de las mismas. En estudios previos [\[16\]](#) se ha determinado que el ancho de banda que ocupan las señales EMG se encuentra en el rango de 20 a 500Hz, ver [anexo A](#), diferencia de gran relevancia cuando se compara con señales de tipo ECG. Adicionalmente, la diferencia que existe en amplitud, de microvoltios en señales EMG comparada con amplitudes de milivoltios en señales ECG, implica que la señal debe ser amplificada con una ganancia muy superior tal que sea apreciable en la adquisición [\[1\]](#).

Dado que la naturaleza de las señales es la misma, variando solo por su ancho de banda y amplitud, el problema puede ser tratado de igual forma, teniendo en cuenta las consideraciones previamente mencionadas.

Figura 3.1 Esquema general por etapas del diseño implementado.



Fuente Autores

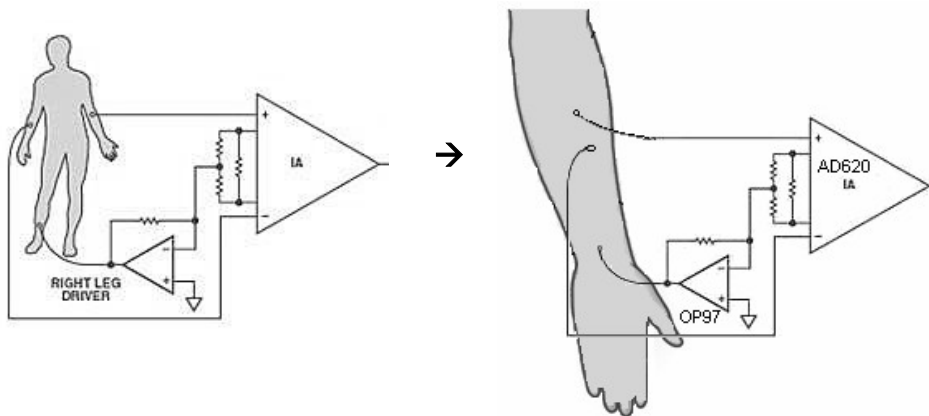
3.1. Etapa 1

La primera etapa o etapa de entrada esta constituida por el amplificador de instrumentación AD620 y el amplificador operacional OP97. Fue diseñada para acondicionar la señal de entrada en modo diferencial eliminando interferencia por ruido electromagnético y señales en modo común aumentando la relación señal a ruido [16].

La ganancia total implementada en la tarjeta es de 2000V/V, distribuida en dos circuitos diferentes. El AD620 amplifica la señal diferencialmente en 20V/V, relativamente baja para la capacidad del mismo, sin embargo es necesario ya que en ocasiones las condiciones de la piel o el desgaste de los electrodos por movimientos constantes producen pequeñas diferencias de potencial estático entre los mismos, saturando la salida a tensiones de alimentación siempre que su ganancia sea potencialmente superior [16].

El circuito de entrada se muestra en la [figura 3.2](#):

Figura 3.2 Circuito implementado en la adquisición de ECG adaptado a EMG*.



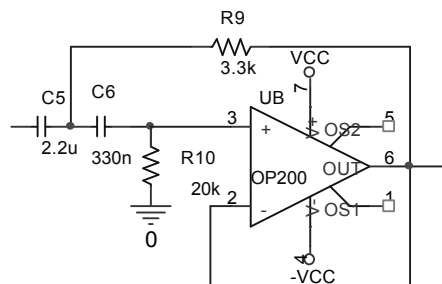
Fuente autores

* La derivación de pierna derecha es análoga al electrodo de referencia y los electrodos de los brazos corresponden a los ubicados sobre los músculos de interés en el antebrazo.

3.2. Etapa 2

La segunda etapa corresponde al filtrado necesario para aislar señales que no pertenecen al rango de las frecuencias de interés. En primer lugar se implementa un filtro pasa-altos de segundo orden y ganancia unitaria cuya frecuencia de corte es 20Hz, encargado de atenuar la señal de entrada en el rango de 0–20Hz, y de eliminar tensión DC presente en la señal. Los componentes de frecuencia presentes entre 0 Hz y 20 Hz son particularmente inestables, ya que se ven afectados por el carácter cuasi-aleatorio de la tasa de disparos en unidades musculares, ver [anexo A](#). Debido a la naturaleza inestable de estos componentes de señal, es aconsejable considerarlos como no deseados y eliminar su efecto. Otra razón de importancia para implementar este filtro se debe a dos fuentes principales de ruido ocasionado por movimiento: una en la interfaz debido al movimiento entre la superficie de detección del electrodo y la piel, otra debida al movimiento del cable de conexión del electrodo al amplificador. Las señales eléctricas de las dos fuentes de ruido tienen la mayor parte de su energía en la gama de frecuencias de 0Hz a 20 Hz. Ambas fuentes son esencialmente reducidas en el diseño adecuado del filtro de orden 2 o superior y 20 dB/dec de atenuación [\[16\]](#). El circuito correspondiente se muestra en la [figura 3.3](#).

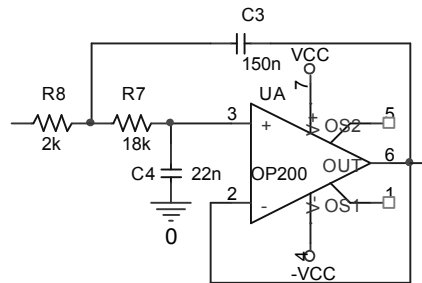
Figura 3.3 Filtro Butterworth pasa-alta de 20Hz. Configuración Sallen-Key.



Fuente Autores

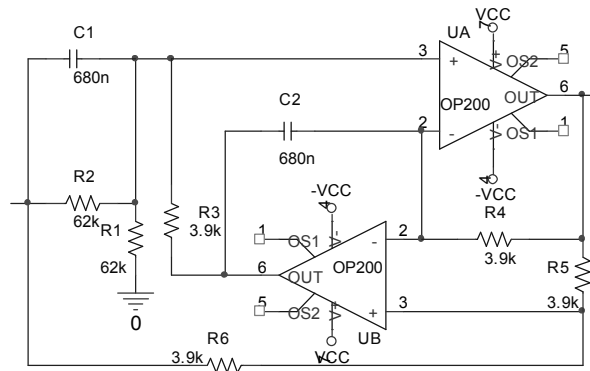
En esta misma etapa se implementa un filtro pasa-bajos de segundo orden, atenuación de 20dB/dec, ganancia unitaria y frecuencia de corte 500Hz, encargado de atenuar amplitudes de la señal a frecuencias superiores dado que el aporte significativo de energía de la señal electromiográfica se encuentra por debajo de dicha frecuencia [\[16\]](#).

El circuito correspondiente se muestra en la [figura 3.4](#).

Figura 3.4 Filtro Butterworth pasa-baja de 500 Hz. Configuración Sallen-Key.

Fuente Autores

Para reducir los efectos restantes de la red eléctrica que se hubiesen filtrado a través del AD620, se implementa un filtro ranura de 60Hz, [figura 3.5](#), diseñado de primer orden con $Q=10$ y ganancia unitaria, que reduce la señal de 60Hz en 23dB.

Figura 3.5 Filtro ranura de 60 Hz*.

Fuente Autores

Para la amplificación final de la señal EMG se implementa un circuito amplificador no inversor a la salida del filtro ranura con una ganancia de 100, lo que resulta en una ganancia total de la tarjeta de 2000V/V.

* Diseñado en FilterPro™ V.2.0.

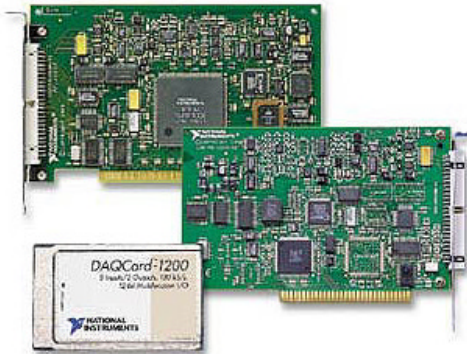
3.3. Etapa 3

Esta corresponde a la digitalización, adquisición y almacenamiento de la señal EMG filtrada y amplificada, para ello se utilizó una tarjeta de adquisición en conjunto con la herramienta software LABVIEW que permitió diseñar la interfaz gráfica del electromiógrafo.

3.3.1. Tarjeta de Adquisición

Para el proceso de digitalización, muestreo y transmisión de la señal EMG se utilizó la tarjeta de adquisición LAB-PC 1200 de National Instruments®, Ver [figura 3.6](#), cuyas características se muestran en la [tabla 3.1](#).

Figura 3.6. Tarjeta de adquisición LAB-PC 1200.



Fuente www.ni.com

Para la adquisición se utilizan los canales 0 y 1 en el modo de entrada RSE, que consiste en referenciar todas las señales a una tierra común (SE).

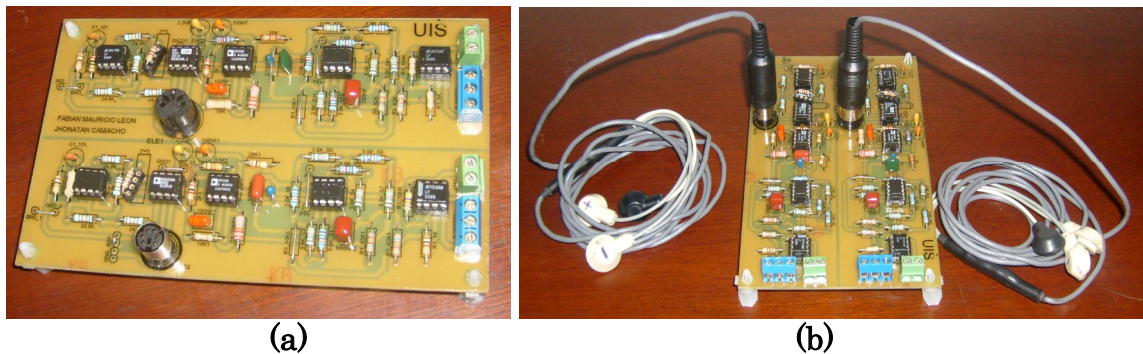
En la [figura 3.7. \(a\)](#) se observa la tarjeta realizada y en la [figura 3.7 \(b\)](#) una vista más amplia de la misma junto con los cables* que conectan los electrodos.

* Estos cables cumplen con el apantallamiento necesario para aislar ruido electromagnético del ambiente.

Tabla 3.1 Resumen de especificaciones para la Tarjeta de adquisición LAB-PC 1200

Especificación	Descripción	Detalle	Observación
General	Formato Físico	ISA	Comunicación
	Soporte para SO	Windows	no se especifica versión
Entrada Analógica	Número de canales de entrada analógica	8 SE/4DI	Señales de una sola terminal (SE) Entradas diferenciales (DI)
	Velocidad de muestreo de entrada analógica	100 kS/s	Velocidad Máxima
	Resolución de entrada analógica	12 bits	Especificación para el convertidor ADC
	Muestreo simultáneo	No	Se debe utilizar la opción de escaneo por intervalos
	Rango de voltaje máximo	-5...5 V	Especificación para el convertidor ADC
	Rango de voltaje mínimo	~50...50mV	
	Número de rangos	14	Maximizan el rango del ADC

Fuente www.ni.com

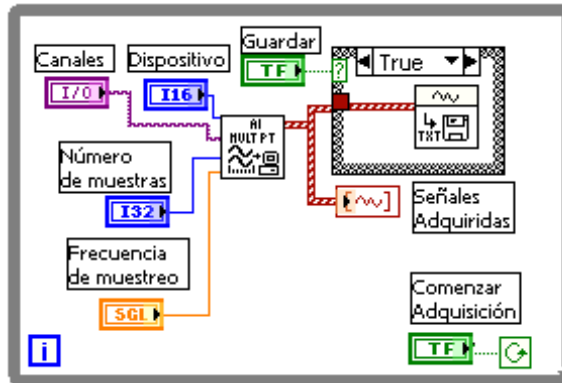
Figura 3.7. (a) Tarjeta de acondicionamiento de señales EMG. (b) Tarjeta junto con los cables para conectar los electrodos.

Fuente autores

3.3.2 Interfaz gráfica del electromiógrafo

Se utilizó el programa LABVIEW® versión 6i para elaborar la herramienta software usada para la adquisición y almacenamiento de la señal EMG. En las figuras [3.8](#) y [3.9](#) se muestra el diagrama de bloques y la interfaz gráfica del electromiografo.

Figura 3.8. Diagrama de bloques del electromyografo.



Fuente Autores

El electro miógrafo permite variar la frecuencia de muestreo y la cantidad de muestras por canal; las cuales son almacenadas en formato .txt.

Figura 3.9. Interfaz gráfica del electromyografo.



Fuente Autores

3.4. Elección de los componentes de la tarjeta

Amplificador de instrumentación: El AD620 es de propósito general, pero dado su alto desempeño tiene especial implementación en aplicaciones médicas. Las características relevantes que posee el AD620 en relación a los requerimientos mínimos* se muestran en la [tabla 3.2](#).

Tabla 3.2 Características del amplificador diferencial AD620

Índice	AD620	Esperado
CMRR	120 dB Típico	90 dB
Impedancia de Entrada Diferencial	10 GΩ 2 pF	10 GΩ 5 pF

Fuente Autores

Amplificadores del filtro: Amplificador operacional OP200. Este OpAmp cumple con los parámetros mostrados en la [tabla 3.3](#). aspectos:

Tabla 3.3. Características del amplificador operacional OP200

Índice	OP200	Esperado
CMRR	130 dB	110 dB
Impedancia de entrada	125 GΩ	10 GΩ

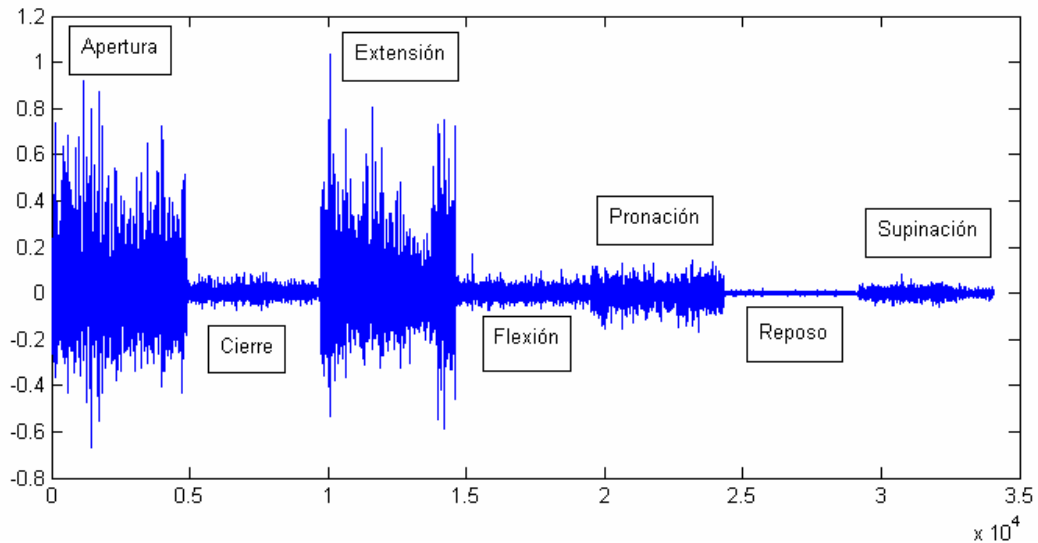
Fuente Autores

Si bien no es posible obtener una señal completamente libre de distorsión o ruido de la red eléctrica, la implementación de estos integrados cumple satisfactoriamente con las necesidades que implica la implementación de un electromiógrafo.

* Los requerimientos mínimos son extraídos directamente de www.delsys.com, empresa dedicada a la fabricación comercial de dispositivos para adquirir señales EMG.

En la [Figura 3.10](#) se muestra la señal EMG* correspondiente a los movimientos a identificar†. El registro de la señal es ordenado alfabéticamente con igual cantidad de muestras por movimiento‡.

Figura 3.10. Señal EMG



Fuente Autores

3.5. Adquisición de señales EMG con equipo BIOPAC

La tarjeta diseñada en la investigación para cumplir sin problemas los requerimientos del módulo 1 descrito en el [capítulo 2](#) fue una alternativa utilizada en las ocasiones que no se contaba con el equipo comercial especializado en esta tarea de recolección de señales EMG, es decir el BIOPAC.

Este equipo BIOPAC, ver [figura 3.11 \(a\)](#), permite adquirir señales de diferente naturaleza como ECG, EEG, EMG, entre otras. Posee una interfaz gráfica, [figura 3.11 \(b\)](#), en la cual se configuran digitalmente los filtros, la ganancia y parámetros como la frecuencia de muestreo y el tiempo de adquisición. Las señales son almacenadas con una resolución de 12 bits.

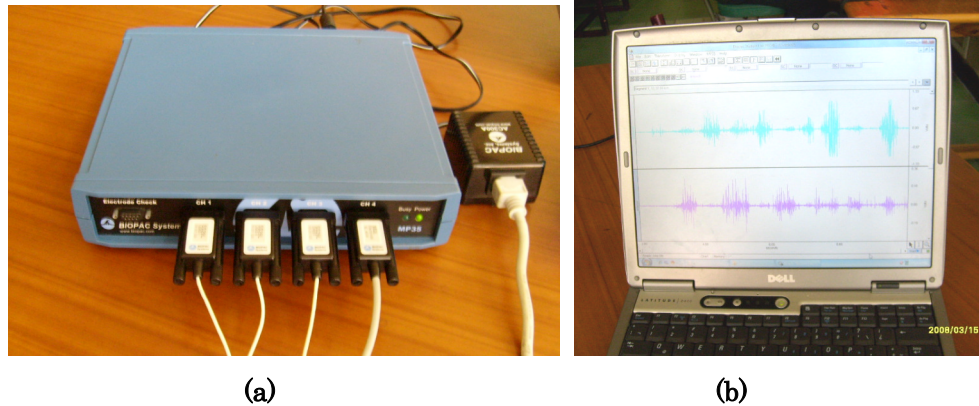
En la [figura 3.12](#) se muestra la disposición de los electrodos en el antebrazo de uno de los pacientes con quien se realizó la prueba

* La señal ha sido tomada utilizando el sistema de adquisición diseñado por los Autores.

† Apertura, cierre, extensión, flexión, pronación, reposo y supinación.

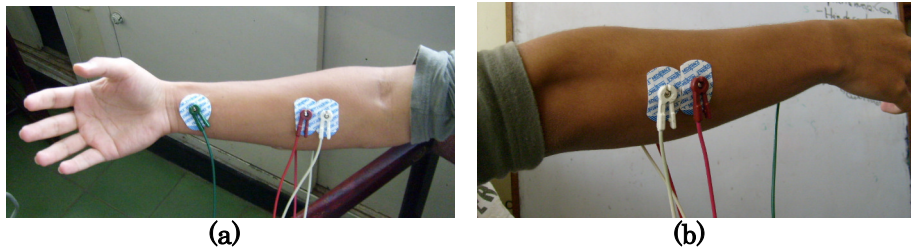
‡ El tiempo mostrado corresponde a 5 segundos por movimiento.

Figura 3.11. Equipo BIOPAC (a) Hardware. (b) Vista del Software.



Fuente autores

Figura 3.12. Disposición de los electrodos en (a) región anterior del antebrazo y (b) región posterior del antebrazo.



Fuente autores

El archivo que genera este equipo tiene extensión .TXT y es compatible con los formatos admisibles para importar en MATLAB, el cual debe ser posteriormente procesado para que cumpla con los requerimientos descritos en el [anexoG](#).

Capitulo 4

Interfaz gráfica para investigación de patrones EMG

Debido a que el procedimiento para encontrar los mejores parámetros que constituyen los módulos 2 y 3 requiere de un alto grado de cálculos matemáticos complejos, se diseñó una herramienta software que permite llevar a cabo todas las variaciones descritas en el [capítulo 2](#) en una única interfaz organizada y eficiente, [figura 4.1](#). Esta interfaz se compone de dos fases, entrenamiento y prueba, utilizadas en ese orden para determinar cual de todos los modelos* presenta mejor desempeño en la clasificación.

4.1. Fase de entrenamiento

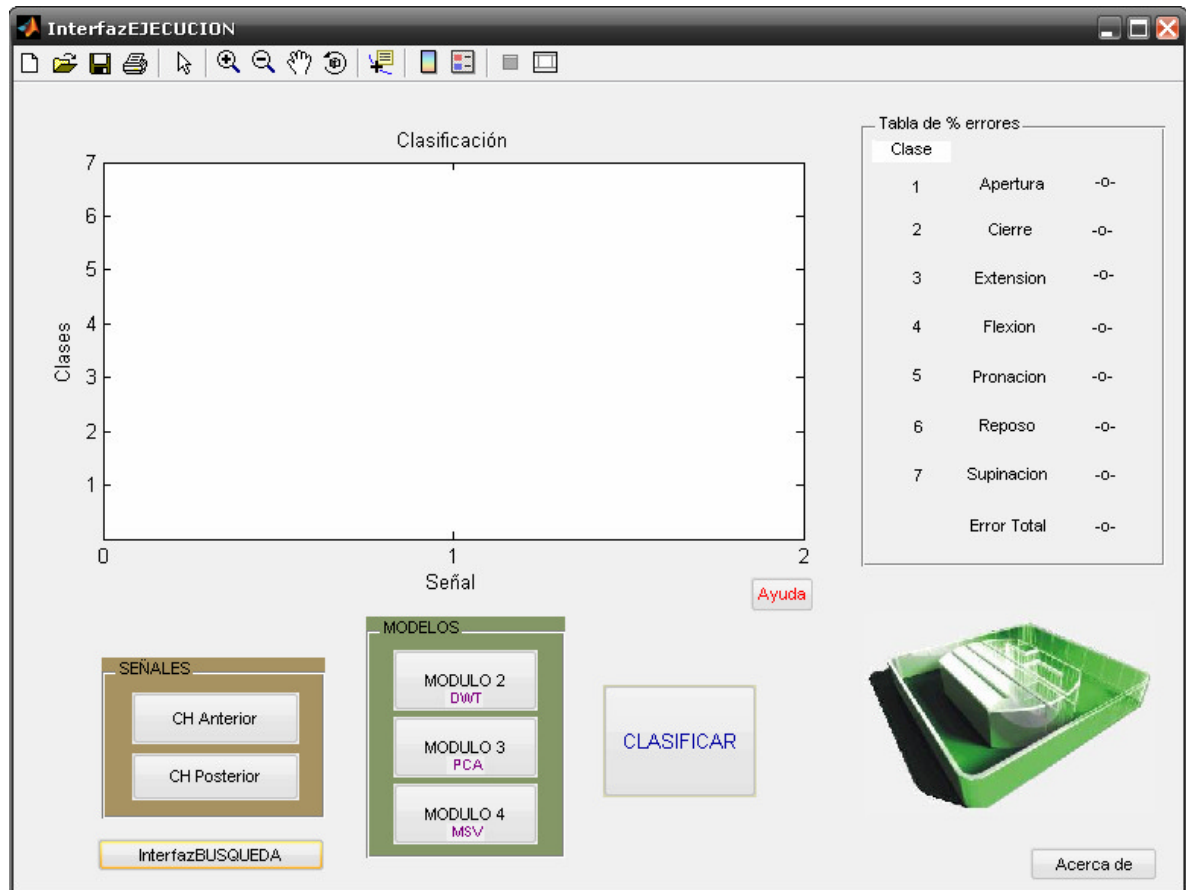
La interfaz principal se llama **InterfazEJECUCION**, la cual se debe correr inicialmente. En esta fase de entrenamiento se trabaja con **InterfazBUSQUEDA**, [figura 4.2](#), la cual se abre con el boton del mismo nombre loalizado justo debajo del contenedor de ingreso de señales de **InterfazEJECUCION** de la [figura 4.1](#). Los ocho modelos generados para una *Wavelet* y un nivel de resolución respectivo de la DWT de la [tabla 2.3](#) se ingresan como un archivo cluster que los contiene y cuya extensión es .MAT en el botón “*Modelos a evaluar*” ubicado en el contenedor llamado “*Calculo de modelos*”, ver [figura 4.2](#). La interfaz toma cada uno de los modelos almacenados, denominado en adelante como **modeloW**, y realiza los cálculos de acuerdo con la información contenida en cada uno de ellos. Los archivos de salida tienen extensión .MAT y se guardan en una dirección asignada por el usuario; los cuales se describen a continuación:

- **Matriz_Resultado:** Contiene el patrón final calculado en cada uno de los registros de 256 muestras que componen la señal original. Esta matriz es

* Estos modelos corresponden a los propuestos en el [capítulo 2](#).

la entrada de la *toolbox* **MSVtoolbox10** en la cual se realiza la fase de entrenamiento de la MSV.

Figura 4.1 Interfaz gráfica usada en el reconocimiento de patrones EMG



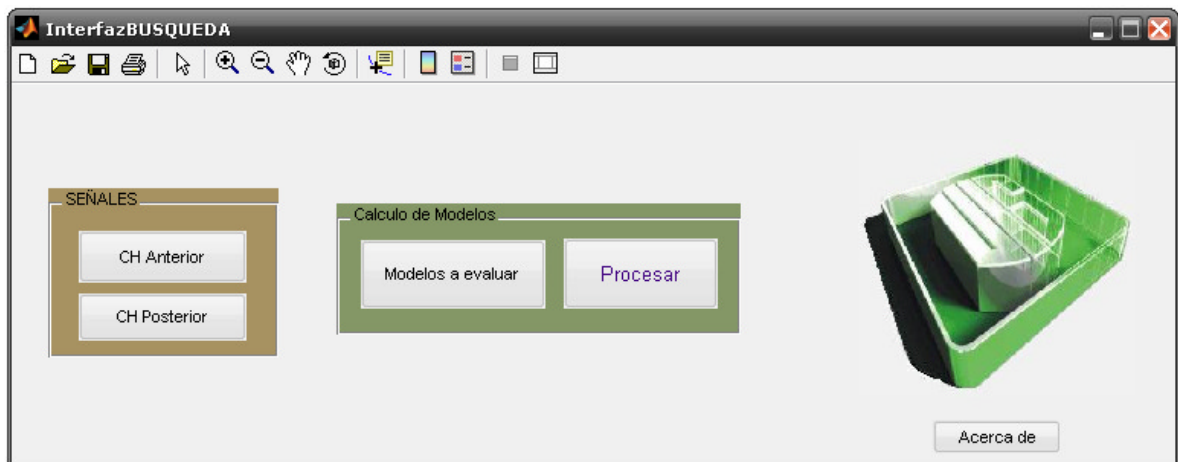
Fuente autores

- **Modelo_PCA:** Contiene los vectores que permiten calcular las componentes que conforman el patrón final a partir del conjunto de características originales. Este modelo es utilizado posteriormente en la fase de prueba.
- **Analisis_PCA:** Contiene información sobre la variabilidad de las componentes que fue utilizado para análisis de las mismas.

El entrenamiento de las MSVs se realizó por completo en **MSVtoolbox10**, la cual genera un archivo .TXT y un .MAT en el que se almacenan los parámetros propios de las MSVs entrenadas. Los parámetros que conforman el llamado

modeloMSV requerido en la fase de prueba son descritos en detalle en el [anexo H](#).

Figura 4.2 Interfaz gráfica usada para evaluar la cantidad de modelos iniciales en el reconocimiento de patrones EMG



Fuente autores

4.2. Fase de prueba

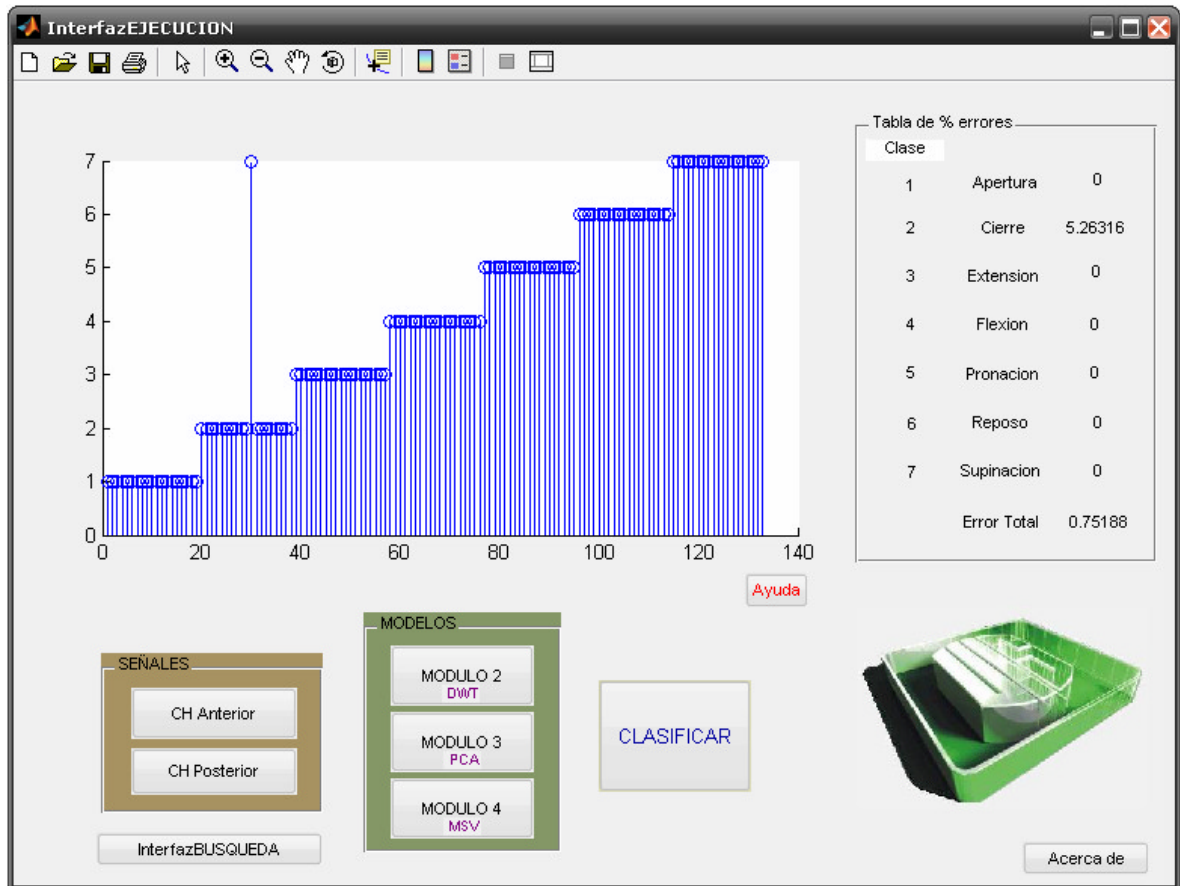
Los modelos generados por **interfazBUSQUEDA** en la fase de entrenamiento para cada **modeloW** son ingresados en la sección **MODELOS** de **InterfazEJECUCION** que se aprecia en la [figura 4.1](#). En los módulos 3 y 4 se ingresan respectivamente **modeloPCA** y **modeloMSV**; y en módulo 2 se ingresa el denominado **modeloW*** respectivo.

La interfaz arroja los porcentajes de error por movimiento y total correspondiente a los modelos ingresados, los cuales se aprecian en la parte superior derecha de **InterfazEJECUCION**, [figura 4.1](#), y muestra una gráfica con los movimientos clasificados e identificados con etiquetas de acuerdo con la [tabla 4.1](#). La [figura 4.3](#) es un ejemplo de la grafica generada luego de clasificar señales de prueba con uno de los pacientes.

Para ver detalles del funcionamiento de la Interfaz ver [Anexo H](#).

* Contiene información sobre alguno de los modelos de la [tabla 2.3](#) del [capítulo 2](#), para cada *Wavelet*.

Figura 4.3 Clasificación de las señales de prueba con uno de los pacientes



Fuente autores

4.3. Consideraciones

Las señales almacenadas usando la tarjeta descrita en el [capítulo 3](#) son registros continuos de los movimientos clasificados, incluyendo la condición de reposo.

Estos registros tienen una duración de 10 segundos por cada movimiento lo que resulta en un registro total de 70 segundos por persona; 35 de ellos se utilizan para la fase de entrenamiento y los restantes 35 segundos en la fase de prueba.

El orden en el que deben ser ingresadas las señales en “CH Anterior” y “CH Posterior” de la interfaz, tanto para fase de entrenamiento como para fase de prueba corresponde con el mostrado en la [tabla 4.1](#).

Tabla 4.1 Orden en el que se deben almacenar los movimientos en el registro total de acuerdo a las clases etiquetadas.

Clase	1	2	3	4	5	6	7
Movimiento	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación
Orden	Inicio.....Final						

Fuente autores

Los formatos que deben cumplir las señales EMG ingresadas a la interfaz, así como una descripción detallada de las funciones que la conforman se encuentran en el [anexo H](#).

El [capítulo 5](#) presenta los resultados de la prueba piloto y de la prueba de validación encontrados con esta interfaz.

Capítulo 5

Resultados

La determinación de los modelos que permitieron seleccionar los parámetros que conforman los módulos 2 y 3 para obtener el patrón mas probable en la identificación de los movimientos se realiza en este capítulo. Esto se logra comparando los resultados obtenidos de cada uno de los modelos descritos en el [capítulo 2](#) para una resolución de la DWT determinada con su respectiva *Wavelet* madre. La variación de cada parámetro permite observar el desempeño del patrón permitiendo discriminar cálculos que pueden ser innecesarios para una correcta clasificación de los movimientos.

5.1. Metodología desarrollada en la prueba

Se trabaja conforme al desarrollo descrito en las especificaciones del [capítulo 2](#) con la herramienta diseñada del [capítulo 4](#).

5.2. Pruebas realizadas

Teniendo en cuenta el [anexo D](#), se realizó una prueba preliminar para establecer cuales *Wavelet* madre pueden presentar un buen desempeño en la conformación del patrón característico. La segunda prueba realizada sobre las señales EMG a un segundo paciente, buscó determinar si los resultados obtenidos con las *Wavelet* madre de buen desempeño en el primer paciente pueden reflejarse en otro.

5.2.1. Prueba preliminar

Las *Wavelet* que son evaluadas en el proceso de identificación de movimientos están conformadas por aquellas que han arrojado buenos resultados en investigaciones precedentes como en [\[18\]](#) y algunas otras que presentaron buenos resultados en pruebas de carácter exploratorio hechas por los autores.

Las *Wavelet* escogidas para esta prueba preliminar son:
Daubechies de orden 4 y 5, Coiflet de orden 3 y 4, Symlet de orden 4 y 6.

Esta prueba preliminar busca además establecer si los parámetros de la toolbox MSVtoolbox10 [6] descrita en el [capítulo 3](#) influyen en la exactitud de la clasificación. Tales parámetros hacen referencia a tres aspectos:

- Cantidad de grupos en validación cruzada durante el proceso de entrenamiento de la MSV.
- Rango de valores en parámetros propios de cada tipo de *kernel*.
- Rango de valores en parámetro de penalización C .

Para el primer aspecto se establece una cantidad de 3 y 4 grupos para el proceso de validación cruzada en la maquina, que busca determinar si tales divisiones del conjunto de entrenamiento influyen en la clasificación final. Para el segundo y tercer aspecto se establece un rango de valores idénticos para todos los parámetros de los *kernels*, incluyendo el valor de C . Este rango es el resultado de realizar pruebas preliminares con la búsqueda en malla y hallar el intervalo de valores de mejor desempeño. La [tabla 5.1](#) muestra estos rangos de variación.

Tabla 5.1. Rangos de variación de los parámetros de las MSVs en entrenamiento.

Intervalo	Polinómico			Sigmoide			RBF	
	C	d	c	C	K	θ	C	σ
Inicio	10	0.5	0.5	10	0.5	0.5	10	0.5
Paso	100	0.5	0.5	100	0.5	0.5	100	0.5
Final	1010	1.5	1.5	1010	1.5	1.5	1010	1.5

Fuente autores

5.2.1.1. Resultados prueba preliminar de MSVs.

Las señales tomadas del paciente 1 fueron procesadas según las especificaciones de la [sección 4.3](#) en cada una de las *Wavelet* establecidas para el entrenamiento de las MSVs.

En el [anexo H](#) se encuentran las tablas que muestran los resultados encontrados para cada *Wavelet* durante la fase entrenamiento de las MSVs. Con base en esas tablas se seleccionan solo aquellos modelos de *Wavelets* con nivel de resolución hasta el cuarto nivel para realizar la fase de prueba. Los detalles sobre la selección de los modelos usados para la fase de prueba se discuten en el [anexo H](#).

En las [tablas 5.3](#) a [5.8](#) se muestran los porcentajes de error en clasificación de la fase de prueba para cada *Wavelet*.

Tabla 5.2. Porcentajes de error en clasificación para fase de prueba con la *Wavelet* Coiflet 3.

MODELOS	COIFLET 3		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	9.02	9.02	9.77
Modelo 2	6.76	6.76	9.02
Modelo 3	18.04	17.29	24.06
Modelo 4	12.03	7.51	20.30
Modelo 5	11.27	10.52	8.27
Modelo 6	12.03	9.77	10.52
Modelo 7	14.28	9.77	20.30
Modelo 8	13.53	12.78	21.80

Fuente autores

Tabla 5.3. Porcentajes de error en clasificación para fase de prueba con la *Wavelet* Coiflet 4.

MODELOS	COIFLET 4		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	8.27	9.02	6.76
Modelo 2	6.01	6.76	6.76
Modelo 3	11.27	9.02	20.30
Modelo 4	9.02	6.01	18.04
Modelo 5	9.02	9.02	12.03
Modelo 6	7.51	7.51	10.52
Modelo 7	9.02	9.02	18.04
Modelo 8	7.51	6.01	16.54

Fuente autores

Tabla 5.4. Porcentajes de error en clasificación para fase de prueba con la *Wavelet* Daubechies 4.

MODELOS	DAUBECHIES 4		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	9.77	9.02	9.02
Modelo 2	7.51	7.51	5.26
Modelo 3	12.78	11.27	19.54
Modelo 4	9.02	8.27	18.04
Modelo 5	12.03	11.27	9.77
Modelo 6	10.52	11.27	9.77
Modelo 7	12.03	9.02	18.04
Modelo 8	9.02	8.27	15.03

Fuente autores

Tabla 5.5. Porcentajes de error en clasificación para fase de prueba con la *Wavelet* Daubechies 5.

MODELOS	DAUBECHIES 5		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	10.52	13.53	12.03
Modelo 2	6.76	6.76	7.51
Modelo 3	18.79	14.28	20.06
Modelo 4	12.03	8.27	20.30
Modelo 5	11.27	10.52	8.27
Modelo 6	10.52	9.77	9.51
Modelo 7	9.02	8.27	17.29
Modelo 8	9.02	8.27	18.04

Fuente autores

Tabla 5.6. Porcentajes de error en clasificación para fase de prueba con la *Wavelet* Symlet 4.

MODELOS	SYMLET 4		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	9.02	8.27	9.02
Modelo 2	7.51	7.51	7.51
Modelo 3	15.03	15.78	18.79
Modelo 4	12.78	9.77	21.05
Modelo 5	9.02	9.02	9.02
Modelo 6	10.52	10.52	10.52
Modelo 7	10.52	9.02	18.79
Modelo 8	9.77	9.77	21.05

Fuente autores

Tabla 5.7. Porcentajes de error en clasificación para fase de prueba con la *Wavelet Symlet 6*.

MODELOS	SYMLET 6		
	Polinómico	RBF	Sigmoide
	ERROR %	ERROR %	ERROR %
Modelo 1	9.02	9.02	9.02
Modelo 2	9.02	9.02	9.02
Modelo 3	12.78	10.52	16.54
Modelo 4	11.27	9.02	19.54
Modelo 5	12.03	9.02	9.77
Modelo 6	10.52	11.27	9.02
Modelo 7	10.52	10.52	18.04
Modelo 8	10.52	11.27	18.04

Fuente autores

5.2.2. Prueba de validación

Con base en los resultados arrojados por la prueba preliminar (prueba piloto) se realizan las pruebas de validación a las señales EMG del segundo paciente con aquellos modelos que mostraron un desempeño en la clasificación menor al 10% de error*.

Los resultados de la fase de prueba se muestran en las tablas [5.9](#) a [5.13](#) y la discusión de estos resultados se encuentra detallada en la [sección 5.3](#).

* Estos modelos se identifican en las [tablas 5.3](#). a [5.8](#). con aquellas celdas sombreadas de color verde.

Tabla 5.8. Porcentaje de errores Total y por cada movimiento usando Coiflet 3.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
1	POLY	5.26	28.31	0	0	10.52	0	0	6.01
1	RBF	5.26	31.57	0	0	10.52	0	0	6.76
1	SIGMO	5.26	31.57	0	0	10.52	0	0	6.76
2	POLY	5.26	21.05	0	0	0	0	0	3.75
2	RBF	5.26	47.36	0	0	5.26	0	0	8.27
2	SIGMO	5.26	21.05	0	0	0	0	0	3.75
4	RBF	0	26.31	0	0	0	0	0	3.75
5	SIGMO	5.26	31.57	0	0	15.78	0	0	7.51
6	RBF	5.26	31.57	0	0	0	0	21.05	8.27
7	RBF	0	21.05	0	0	5.26	0	0	3.75

Fuente autores

Tabla 5.9. Porcentaje de errores Total y por cada movimiento usando Coiflet 4.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
1	POLY	10.52	21.05	0	0	0	0	0	4.51
1	RBF	10.52	21.05	0	0	5.26	0	26.31	5.26
1	SIGMO	31.57	31.57	0	0	0	0	26.31	12.78
2	POLY	10.52	21.05	0	0	0	0	0	4.51
2	RBF	10.52	21.05	0	0	5.26	0	0	5.26
2	SIGMO	10.52	31.57	0	0	0	0	0	6.01
4	POLY	10.52	21.05	0	0	0	0	0	4.51
4	RBF	10.52	26.31	0	0	0	0	0	5.26
5	POLY	5.26	21.05	0	0	15.78	0	0	6.01
5	RBF	5.26	21.05	0	0	10.52	0	0	5.26
6	POLY	5.26	26.31	0	0	0	0	5.26	5.26
6	RBF	5.26	36.84	0	0	15.78	0	36.84	13.53
7	POLY	10.52	21.05	0	0	5.26	0	26.31	9.02
7	RBF	10.52	10.52	0	0	0	0	0	3.00
8	RBF	10.52	15.78	0	0	0	0	0	3.75

Fuente autores

Tabla 5.10. Porcentaje de errores Total y por cada movimiento usando Daubechies 4.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
1	POLY	0	26.31	0	0	0	0	0	3.75
1	RBF	0	10.52	0	0	0	0	0	1.50
1	SIGMO	0	31.57	0	0	0	0	0	4.51
2	POLY	0	5.26	0	0	0	0	5.26	1.50
2	RBF	0	5.26	0	0	0	0	5.26	1.50
2	SIGMO	0	10.52	0	0	0	0	0	1.50
4	POLY	0	10.52	0	0	0	0	5.26	2.25
4	RBF	0	0	0	0	0	0	5.26	0.75
5	SIGMO	0	21.05	0	0	10.52	0	10.52	6.01
6	SIGMO	5.26	26.31	0	0	5.26	0	5.26	6.01
7	RBF	0	21.05	0	0	0	0	5.26	3.75
8	POLY	0	10.52	0	0	0	0	5.26	2.25
8	RBF	0	21.05	0	0	0	0	5.26	3.75

Fuente autores

Tabla 5.11. Porcentaje de errores Total y por cada movimiento usando Daubechies 5.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
2	POLY	5.26	5.26	0	0	0	0	5.26	2.25
2	RBF	0	21.05	0	0	0	0	21.05	6.01
2	SIGMO	5.26	15.78	0	0	0	0	5.26	3.75
4	RBF	0	5.26	0	0	0	0	5.26	1.50
5	SIGMO	5.26	31.57	0	0	0	0	21.05	8.27
6	RBF	0	31.57	5.26	0	0	0	15.78	5.26
6	SIGMO	0	31.57	5.26	0	0	0	47.36	12.03
7	POLY	0	15.78	0	0	5.26	0	15.78	5.26
7	RBF	5.26	5.26	0	0	5.26	0	5.26	3.00
8	POLY	0	5.26	0	0	0	0	5.26	1.50
8	RBF	0	15.78	0	0	0	0	5.26	3.00

Fuente autores

Tabla 5.12. Porcentaje de errores Total y por cada movimiento usando Symlet 4.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
1	POLY	5.26	21.05	0	0	0	0	0	3.75
1	RBF	5.26	26.31	0	0	0	0	0	4.51
1	SIGMO	10.52	21.05	0	0	10.52	0	0	4.51
2	POLY	0	5.26	0	0	0	0	0	0.75
2	RBF	0	10.52	0	0	0	0	0	1.50
2	SIGMO	0	10.52	0	0	0	0	0	1.50
5	POLY	15.78	21.05	0	0	15.78	0	5.26	8.27
5	RBF	21.05	15.78	0	0	10.52	0	5.26	7.51
5	SIGMO	10.52	10.52	0	0	10.52	0	10.52	6.01
7	RBF	0	15.78	0	0	0	0	0	3.00
8	RBF	5.26	21.05	0	0	0	0	0	3.75

Fuente autores

Tabla 5.13. Porcentaje de errores Total y por cada movimiento usando Symlet 6.

Modelo	Kernel usado en la MSV	Apertura	Cierre	Extensión	Flexión	Pronación	Reposo	Supinación	TOTAL %
1	POLY	10.52	26.31	0	0	0	0	0	5.26
1	RBF	5.26	26.31	0	0	0	0	0	4.51
1	SIGMO	10.52	26.31	0	0	0	0	0	5.26
2	POLY	0	26.31	0	0	0	0	0	3.75
2	RBF	0	26.31	0	0	0	0	0	3.75
2	SIGMO	0	26.31	0	0	0	0	0	3.75
4	RBF	0	21.05	0	0	0	0	0	3.00
5	RBF	5.26	21.05	0	0	31.57	0	5.26	8.27
5	SIGMO	21.05	26.31	0	0	21.05	0	0	9.77
6	SIGMO	21.05	36.84	0	0	10.52	0	42.10	15.78

Fuente autores

5.3. Discusión

Los resultados encontrados en las pruebas realizadas al segundo paciente evidencian claras tendencias de exactitud con respecto a las maquinas seleccionadas en el paciente preliminar (paciente 1).

De las 70 maquinas de la fase de prueba tan sólo 2 de ellas arrojaron resultados que estuvieron por fuera del rango del 10% de error en clasificación. Dicho de otra forma, el 97.14% de las maquinas presentaron un buen desempeño en la prueba al segundo paciente.

Otro aspecto no menos importante, son los porcentajes obtenidos con este segundo paciente en los cuales se presentan valores de error en promedio menor a los encontrados con el primer paciente. En particular es bastante interesante saber que los resultados mínimos de error fueron del 0.75% en relación al 5.26% del primer paciente, esto puede deberse a la plasticidad motora propia del mismo o quizás en la ejecución del movimiento este la clave para obtener errores tan bajos. Es pertinente realizar pruebas adicionales que esclarezcan la razón por la cual se obtienen resultados mucho mejores en ciertos pacientes, que establezca si la realización de un entrenamiento adecuado con el paciente en su ejecución de los movimientos pueda mejorar sustancialmente los porcentajes de exactitud en la clasificación.

5.4. Interfaz Práctica

Como consecuencia de los resultados encontrados se presenta una función mucho más práctica de la **InterfazEJECUCION** cuyo fin es usar solo aquellos parámetros que permiten calcular el patrón con mejor desempeño de acuerdo a las pruebas realizadas en esta investigación. Los parámetros que componen este modelo se presentan en la [tabla 5.15](#).

Además del modelo, se recomienda realizar el proceso de entrenamiento de la MSV con los parámetros descritos en la [tabla 5.16](#).

Tabla 5.14. Parámetros del modelo con mejor desempeño en la investigación

Parámetro	Descripción
Wavelet	Symlet4
Nivel de Resolución	4
Estadísticos calculados	RMS
Análisis PCA	CORRELACION
Criterio de selección	ESTANDAR

Fuente autores

Tabla 5.15. Parámetros de entrenamiento de la MSV recomendados

Parámetro	Descripción
Kernel	Polinómico
C	Inicio:10, Paso:100, Final:1010
d	Inicio:0.5, Paso:0.5, Final:1.5
c	Inicio:0.5, Paso:0.5, Final:1.5
Grupos en Validación cruzada	3
Método de optimización	SMO
Descomposición	O.v.O

Fuente autores

Para este propósito, dentro de la carpeta que contiene la interfaz se crea el archivo que contiene el modelo que mejores resultados de clasificación arrojó. Este modelo se identifica por el sufijo **mejor** en la carpeta **Mejor Modelo**.

De tal forma que un usuario estándar puede acceder a la interfaz y manipularla únicamente con el modelo de mejor desempeño; probarla siguiendo las instrucciones de la [tabla 5.16](#) para nuevas señales EMG que cumplan el formato descrito en el [anexo G](#), o con las señales EMG ya recopiladas en cuyo caso sólo se necesitan los tres últimos pasos de la tabla 5.16.

Tabla 5.16. Instrucciones de uso como Interfaz Práctica

Paso	Descripción
1	Instalar* y Correr InterfazEJECUCION
2	Ingresar las señales EMG de cada canal para la fase de entrenamiento según el formato descrito en el anexo G y las consideraciones del capítulo 4 .
3	Oprimir el boton “ <i>InterfazBUSQUEDA</i> ”
4	En la ventana que se abre al oprimir el botón “ <i>Modelos a evaluar</i> ” en la interfaz, buscar el archivo que se encuentra en la carpeta Mejor Modelo llamado Cluster_mejor ó Cluster_mejor2
5	Dar clic en el botón <i>Procesar</i> y seleccionar la carpeta de destino.
6	Abrir la <i>Toolbox MSVtoolbox10</i> y entrenar la MSV con el archivo Matriz_Resultado que generó la interfaz, siguiendo los parámetros recomendados en la tabla 5.16 . Guardar el modelo que genera en la misma carpeta de los archivos anteriores.
7	Ingresar las señales EMG de cada canal para la fase de prueba en InterfazEJECUCION según el formato descrito en el anexo G .
8	En InterfazEJECUCION dar clic en el botón <i>MODULO 3</i> e ingresar el archivo llamado modeloPCA generado en el paso 4. Dar clic en el botón <i>MODULO 2</i> e ingresar archivo llamado ModeloW_mejor [†] . Dar clic en el botón <i>MODULO 4</i> e ingresar el archivo generado en el paso 6.
8	Dar clic en el botón <i>CLASIFICAR</i>

Fuente autores

* El proceso de instalación de la InterfazEJECUCION se presenta al final del [anexo G](#).† Este archivo esta ubicado en la carpeta **Mejor Modelo**

Capítulo 6

Conclusiones, aportes y trabajos futuros

6.1. Conclusiones

- Se identificaron los movimientos de apertura, cierre, flexión, extensión, pronación y supinación de la mano, aplicando procesamiento digital a las señales EMG generadas voluntariamente por un paciente sano con tasas de acierto superiores al 90%.
- La tarjeta de acondicionamiento de señales electromiográficas garantizó la disponibilidad en todo momento de una herramienta hardware para recopilar datos de diferentes pacientes.
- A través del procesamiento digital de señales EMG mediante la Transformada Discreta *Wavelet*, Analisis de Componentes Principales y Maquinas de Soporte Vectorial se encontró una metodología que permite identificar los movimientos de interés.
- Es factible diseñar un modelo de identificación de patrones capaz de procesar información con el uso de poca instrumentación, fácil adaptabilidad para mejoras de funcionalidad y la posibilidad de implementación en un sistema embebido.
- Es posible usar el mismo tipo de *Wavelet* con su respectiva metodología de procesamiento en PCA y MSV a diferentes pacientes con resultados pertinentes.

- Una ejecución particular de los movimientos por parte del paciente es innecesaria para obtener resultados satisfactorios. No obstante, se observó que al realizar los movimientos con un umbral mínimo de fuerza estos presentaban un mejor desempeño.

Es posible utilizar como patrón de las señales EMG únicamente el valor RMS de los coeficientes obtenidos mediante transformada *Wavelet*.

- El criterio *estándar* propuesto por los autores para la selección de la cantidad de componentes que conforman el patrón final presenta mejores resultados en la clasificación que el criterio káiser.
- Niveles de descomposición *Wavelet* superiores al cuarto, no son necesarios para cumplir con los objetivos planteados.
- Los resultados de clasificación no presentan una tendencia al conformar 3 o 4 grupos en el proceso de validación cruzada. Por esta razón se recomienda probar con ambos y escoger la MSVs que requiere la menor cantidad de vectores soporte.
- Para el proceso de multclasificación en las MSVs no es viable utilizar métodos de optimización IRWLS, ni métodos de descomposición ECOC2 para datos obtenidos mediante señales EMG debido a la falta de convergencia en los resultados. En contraste, se recomienda el método de optimización SMO y la descomposición O-v-O por presentar tiempos de entrenamiento que no superan los 20 segundos*; esta metodología se prefiere sobre métodos de descomposición O-v-R, que a pesar de converger en los resultados, sus tiempos de entrenamiento son mayores.
- La metodología presentada para el proceso de multclasificación mostró que normalizar los datos, utilizar votación por parejas y utilizar función probabilística en el sistema de decisión hace posible obtener exactitudes superiores al 90%.

* Equipo usado Pentium 4 con 512 MB de RAM.

6.2. Aportes

- Se mostró que es posible lograr aciertos con una tasa de error menor al 10%, utilizando instrumentación que tan solo requiere dos canales de adquisición.
- Se mostró la eficiencia de la metodología propuesta para identificar seis tipos de movimientos incluyendo la condición de reposo.
- Se desarrolló una interfaz que permite realizar de manera conjunta la identificación de movimientos a partir de modelos estructurados en la misma.
- Se diseñó un sistema de acondicionamiento y tratamiento digital de señales EMG, que posibilita el uso de técnicas de procesamiento digital de señales. El sistema puede ser usado como modelo para realizar otras investigaciones afines.
- Se recopilaron señales EMG, las cuales pueden ser usadas para realizar análisis de diferente tipo en investigaciones posteriores.
- Se realizó la consulta y recopilación bibliográfica del estado del arte de la Transformada *Wavelet*, Análisis de Componentes Principales, Máquinas de Soporte Vectorial, que puede ser utilizada como guía base para entender rápidamente los conceptos, con el fin de proponer nuevas metodologías de solución en las mismas.
- Se realizó una investigación y recopilación bibliográfica de la fisiología del antebrazo, generación de bio-potenciales, electromiograma, así como el estado del arte para la adquisición adecuada de señales EMG para distintos propósitos.
- La implementación de un clasificador continuo no mejoró los resultados en la metodología propuesta en la presente investigación.

6.3. Trabajo Futuro

- El buen desempeño de la metodología propuesta incita a dar el siguiente paso en reconocer movimientos deseados por un paciente amputado.
- Desarrollar un sistema embebido que implemente la metodología aportada por los autores con el fin de interpretar on-line la clasificación de movimientos para controlar sistemas robóticos, una prótesis diseñada o prótesis comerciales.
- Plantear investigaciones que permitan realizar control de la velocidad y fuerza con la que se desea realizar un movimiento. O si es preciso de acuerdo a las necesidades presentes, estudiar la posibilidad de realizar la clasificación de movimientos conjuntos.
- Evaluar la viabilidad de adaptar el proceso de control independiente del grado de amputación en el antebrazo del paciente.
- Un estudio posterior puede corroborar si un entrenamiento en la ejecución de los movimientos por parte del usuario permite mejorar los resultados en la identificación.
- Realizar estudios para elegir a priori *Wavelets* madre que permitan un proceso confiable y rápido, identificando la influencia que puede darse en esta elección para medir el grado de representabilidad de una *Wavelet* madre de la señal EMG.
- Realizar estudios comparativos observando la dependencia de los resultados cuando el conjunto de movimientos a identificar es otro.
- Adecuar el sistema desarrollado para identificar señales que no pertenecen al grupo de movimientos de interés, con el objetivo de asegurar confiabilidad en el sistema de decisión protésico.

BIBLIOGRAFÍA

- [1] FRERIKS, B.; HERMENS, H. European Recommendations for Surface Electromyography, Results of the SENIAM project. 2002.
Disponible en: www.seniam.org
- [2] CROMWELL, L. Biomedical Instrumentation and measurements. Prentice-Hall Inc., Englewood Cliffs New Jersey, 1980.
- [3] ADRIAN DEL BOCA, DUNG C. PARK. “Myoelectric signal recognition using fuzzy clustering and artificial neuronal networks in real time”. IEEE transactions on Biomedical Engineer, Jun 1990.
- [4] BRONZINO JOSEPH. Biomedical Engineer Handbook, Second Edition, 2000.
- [5] GAMBA, P.; LANGE, R.; SACCOMANO, C. Trabajo Final “Estudio de la Aplicación Wavelet al Diagnóstico Asistido por Computadora de Mamografías”. UNICEN, 1999.
- [6] CAMPOS E.; SUÁREZ A. Clasificación automática de perturbaciones de señales de tensión o corriente utilizando máquinas de soporte vectorial (MSV). Universidad Industrial de Santander, 2007.
- [7] KECMAN, V. Learning and soft computing, support vector machines, neural networks, and fuzzy logic models. A Bradford Book, 2001.
- [8] M. VETTERLI, y J. KOVACEVIC. “Wavelets and Subband Coding”. Prentice Hall Signal Processing Series, 1995.
- [9] S. E. ZARANTONELLO. “Theory and application of Wavelets”. Santa Clara University, 1997.
- [10] DELGADO ALBERTO. Inteligencia Artificial y Minirobots. Universidad Nacional de Colombia, 1998. ISBN: 958-648-155-7.
- [11] CHANG C.-C, and Lin, C—J. 2001. LIBSVM: *a Library for support vector machines*. Software disponible en <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

- [12] CRAWFORD B., MILLER K., SHENOY P., RAO R. Real-Time Classification of Electromyographic Signals for Robotic Control, University of Washington, March 2005.
- [13] BONIFACIO MARTIN DEL BRIO, SANZ A. Redes Neuronales y Sistemas Difusos 2da edición Ampliada y Revisada, Universidad de Zaragoza, 2002.
- [14] LINDSAY I. SMITH. A tutorial on Principal Components Analysis. 2002. Disponible en:
http://csnet.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
- [15] CARLO J. DE LUCA, Surface electromyography: Detection and Recording, Delsys Inc, 2002. Disponible en:
http://www.delsys.com/Attachments_pdf/WP_SEMGintro.pdf
- [16] GIANLUCA DE LUCA, Fundamental Concepts in EMG Signal Acquisition, Delsys Inc, 2001. Disponible en:
http://www.delsys.com/Attachments_pdf/WP_Sampling1-4.pdf
- [17] ZECCA, S. MICERA, M. C. CARROZZA, & P. DARIO. Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal. ARTS Lab, Scuola Superiore Sant'Anna, Pontedera, Italy, 2002.
- [18] ENGLEHART K.; HUDGINS B.; PARKER P. A Wavelet-Based Continuous Classification Scheme for Multifunction Myoelectric Control. IEEE transactions on biomedical engineering, vol. 48, no. 3, March 2001.
- [19] CONTANDRIOPOULOS AP.; CHAMPAGNE F.; POTVIN L.; DENIS JL, BOYLE P. Preparar un proyecto de investigación. Barcelona: SG Editores, 1991.
- [20] RUEL CHURCHILL. "Series de FOURIER y problemas de contorno". McGraw-Hill, 1978.
- [21] DELGADO GARCIA A. Anatomía funcional y clínica. Universidad del valle, Cali, 1994.
- [22] BARREDA LUIS E. Electromiógrafo. Universidad Nacional del Mar del Plata, 2005.
- [23] MORALES G.; GOMEZ A. Estudio e implementación de una herramienta basada en Maquinas de Soporte Vectorial aplicada a la localización de

- fallas en sistemas de distribución. Universidad Industrial de Santander, 2005.
- [24] Y. SHENG, "The Transforms and Applications Handbook". CRC Press, 1996.
- [25] I. DAUBECHIES. "Ten Lectures on Wavelets". The Society for Industrial and Applied Mathematics, 1992.
- [26] M. MISITI; Y. MISITI; G OPPENHEIM; Y JM POGGI. The Mathworks-Wavelet Toolbox 4, 2007.
- [27] JOLLIFE I. T. Principal Component Analysis, Springer-Verlag. 1986.
- [28] KAISER H. F. The application of electronic computers to factor analysis, Educ. Psychol. Meas. (Vol. 20). Pp. 141-151. 1960.
- [29] DURAN ACEVEDO C. M. Diseño y optimización de los subsistemas de un sistema de olfato electrónico para aplicaciones agroalimentarias e industriales, Capitulo 2. URV, 2005. ISBN 84-689-8167-2.
Disponible en: <http://www.tdx.cat/TDX-0308106-085448/>
- [30] RAO, C.R. The use and interpretation of principal component analysis in applied research. Sankhya A 26, 329 – 358, 1964.
- [31] PEÑA D. Análisis de datos multivariantes, Capítulo 5. McGraw-Hill. ISBN: 8448136101. Disponible en:
http://www.mhe.es/universidad/ciencias_matematicas/pena/home/CAPITULO.PDF.
- [32] COLMENARES G. Analisis Multivariante y aplicaciones. Universidad de los Andes Mérida- Venezuela. Postgrado en economía.
Disponible en: <http://webdelprofesor.ula.ve/economia/gcolmen/postgrado2.html>
- [33] CHRISTOPHER J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition". Data Mining and Knowledge Discovery 2:121 - 167, 1998
- [34] VLADIMIR VAPNIK. The Nature of Statistical Learning Theory. Springer, NY, 1995.
- [35] HENAO R. Selección de Hiperparámetros en Maquinas de Soporte Vectorial, Universidad Nacional de Colombia, 2004.

- [36] ANGULO B. C. Aprendizaje con maquinas núcleo en entornos de multclasificación, Universitat Politècnica de Catalunya (UPC), Tesis doctoral, Vilanova i la Geltrú, abril 2001
- [37] PEDROZA B. J. G. Aplicación de las maquinas de soporte vectorial al reconocimiento de hablantes, Universidad Autónoma metropolitana, maestría en ciencias y tecnologías de la información, 22 junio 2007.

Anexo A

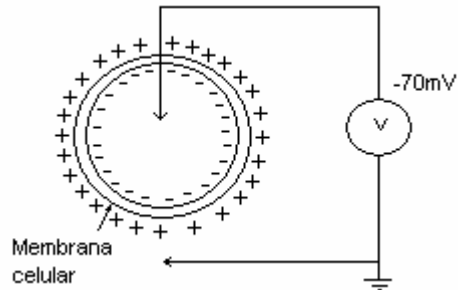
Generación de biopotenciales

A.1. Bioelectricidad [\[2\]](#)

Los potenciales bioeléctricos se asocian a las actividades musculares y nerviosas entre otras, tales potenciales bioeléctricos son realmente potenciales iónicos producidos como resultado de la actividad electroquímica de ciertos tipos especiales de células (musculares y nerviosas). Por medio de transductores capaces de convertir potenciales iónicos en tensiones eléctricas, se pueden medir estas señales de monitorización naturales para presentar los resultados de una forma comprensible y ajustable a diagnóstico y tratamiento de enfermedades, como también en el área investigativa para el desarrollo de prótesis.

A.2 Potenciales de reposo y acción [\[2\]](#)

Ciertos tipos de células del organismo, como las células musculares y nerviosas, están encerradas en una membrana semipermeable que permite el ingreso de algunas sustancias a través de ella y otras no. Alrededor de estas células se encuentran líquidos orgánicos o soluciones conductoras que contienen átomos cargados conocidos como iones. Los principales iones son sodio (Na^+), potasio (K^+) y cloruro (Cl^-). La membrana de las células excitables permite fácilmente la entrada de iones potasio y cloruro pero bloquea eficazmente la entrada a iones sodio, lo cual acarrea dos consecuencias principalmente debido al desequilibrio entre el interior y el exterior de la célula. En primer lugar, el exterior de la célula tiende a ser más positiva que el interior debido a que la concentración de iones sodio es mayor en el exterior. En segundo lugar, en un intento por equilibrar la carga eléctrica, entrarán en la célula iones potasio, produciendo una concentración de iones potasio mayor al interior que en el exterior de esta. Sin embargo, este equilibrio de cargas no se puede lograr debido al desequilibrio en la concentración de iones potasio. El equilibrio se alcanza con una diferencia de potencial a través de la membrana, negativo en el interior y positivo en el exterior.

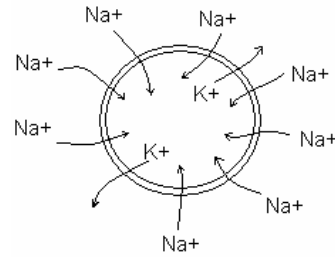
Fig A.1 Célula polarizada con su potencial de reposo.

Fuente [\[2\]](#)

Este potencial de membrana se denomina *potencial de reposo* de la célula y se mantiene hasta que una perturbación de algún tipo altera el equilibrio. Dado que la medida del potencial de la membrana se hace por lo general respecto a los líquidos orgánicos, el potencial de reposo de una célula viene dado como un valor negativo. Una célula en estado de reposo se dice que esta *polarizada*.

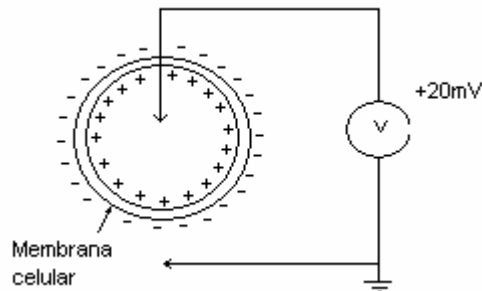
Cuando se excita una parte de la membrana celular mediante el flujo de corriente iónica o mediante algún tipo de energía aplicada externamente, la membrana cambia sus características y empieza a permitir la entrada de algunos iones sodio. Este movimiento de iones sodio hacia el interior de la célula constituye un flujo de corriente iónica que reduce más la barrera de la membrana a los iones de sodio. El resultado neto es un efecto de avalancha en el que los iones sodio se precipitan literalmente en el interior de la célula intentando alcanzar un equilibrio con los iones del exterior. Al mismo tiempo, los iones potasio, que estaban en mayor concentración en el interior de la célula durante el estado de reposo, intentan salir pero son incapaces de moverse tan rápido como los iones sodio. Como resultado de ello, la célula tiene un potencial ligeramente positivo en el interior debido al desequilibrio de iones potasio.

Este potencial se conoce como *potencial de acción*. Una célula que ha sido excitada y que presenta un potencial de acción se dice que esta despolarizada; el proceso de cambio desde el estado de reposo al potencial de acción se denomina despolarización.

Figura A.2 Despolarización de una célula

Fuente [2]

Una vez ha cesado la avalancha de iones sodio a través de la membrana celular (se ha alcanzado un nuevo estado de equilibrio), desaparecen las corrientes iónicas que reducían la barrea a los iones sodio y la membrana vuelve a la situación original de permeabilidad selectiva, bloqueándose de nuevo el paso de iones sodio desde el exterior al interior de la célula.

Figura A.3 Célula despolarizada durante un potencial de acción

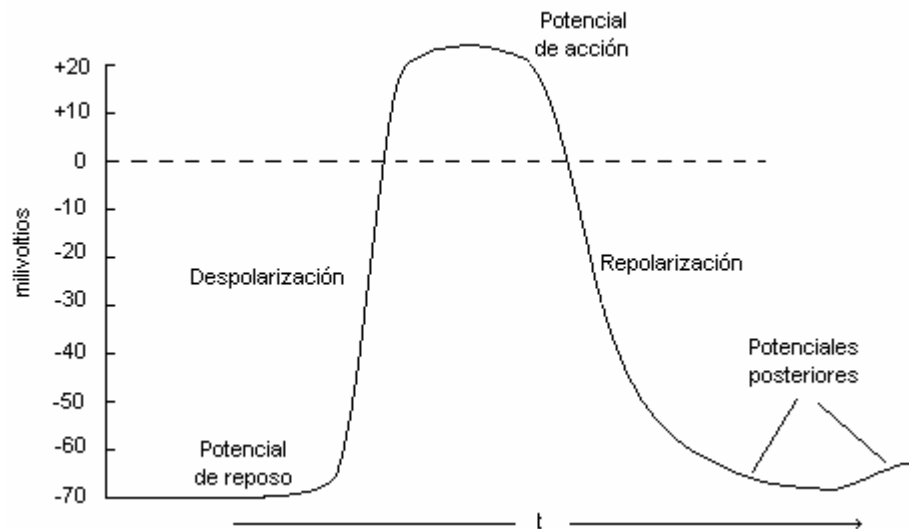
Fuente [2]

Mediante un proceso activo se logra retomar el estado de reposo en un tiempo mas corto, denominado **bomba de sodio**, los iones sodio son transportados rápidamente al exterior de la célula, y esta queda polarizada de nuevo adquiriendo su potencial de reposo. Este proceso se denomina repolarización. La velocidad de bombeo es directamente proporcional a la concentración de sodio en la célula.

La escala de tiempos para el potencial de acción depende del tipo de célula que produce el potencial. En las células nerviosas y musculares, la repolarización se produce después de la despolarización de forma tan rápida que el potencial de acción aparece como un impulso de una duración total de tan sólo un milisegundo.

Independiente del método por el que se excite una célula o de la intensidad del estímulo (suponiendo que sea suficiente para activar la célula), para una célula dada cualquiera, el potencial de acción es siempre el mismo. Esto se conoce como ley de todo o nada. La altura neta del potencial de acción se define como la diferencia entre el potencial de la membrana despolarizada en el pico del potencial de acción y el potencial de reposo.

Figura A.4 Forma de onda del potencial de acción



Fuente [\[2\]](#)

A.3. Electromiograma (EMG) [\[2\]](#) [\[15\]](#)

Los potenciales bioeléctricos asociados con la actividad muscular constituyen el *electromiograma* (**EMG**). Esos potenciales se pueden medir en la superficie del cuerpo cerca del músculo de interés o directamente en el músculo atravesando la piel con electrodos de aguja. Dado que la mayoría de las medidas EMG se proponen obtener más bien una indicación de la cantidad de actividad de un músculo determinado, o un grupo de músculos, que de una fibra muscular individual, la señal es generalmente una suma de los potenciales de acción individuales de las fibras que constituyen el músculo o músculos donde se mide. Al igual que el EEG, los electrodos de EMG recogen potenciales de todos los músculos dentro de su alcance. Esto significa que los potenciales de los músculos grandes cercanos pueden interferir con los intentos de medir el EMG de músculos pequeños, aún cuando los electrodos se coloquen directamente sobre los músculos pequeños.

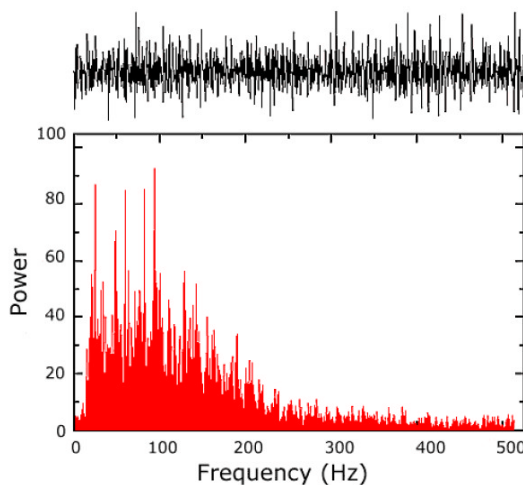
El potencial de acción de un músculo determinado (o fibra nerviosa) tiene una magnitud fija, independientemente de la intensidad del estímulo que genera la respuesta. Así, en un músculo, la intensidad con que actúa no incrementa la altura neta del impulso del potencial de acción sino que incrementa el ritmo con que se dispara cada fibra muscular y el número de fibras que se activan en un instante determinado. La amplitud de la forma de onda EMG medida es la suma instantánea de todos los potenciales generados en un instante determinado. Dado que esos potenciales de acción se producen tanto con polaridades positivas como negativas en un par de electrodos determinado, a veces se adicionan y a veces se cancelan.

De este modo la señal EMG se parece mucho a un ruido aleatorio, siendo la energía de la señal función de la cantidad de actividad muscular y de la situación de los electrodos, [figura A.5](#).

Las características más relevantes de las señales EMG son:

- La señal EMG es de naturaleza estocástica y puede ser razonablemente representada por una función de distribución Gausiana.
- La amplitud de la señal tiene un rango de 10mV (pico-pico).
- La energía útil de la señal está limitada al rango de 0-500 Hz. Con la energía dominante localizada en el rango de 50-150 Hz.

Figura A.5 Espectro de frecuencia de la señal EMG*.



Fuente www.delsys.com

* Señal medida en el músculo tibia anterior durante una contracción isométrica forzada al 50% del máximo voluntario.

Anexo B

Fisiología del antebrazo

B.1. Músculos del antebrazo [\[10\]](#)

Los músculos del antebrazo están dispuestos en dos grupos: anterior y posterior. Los del primer grupo son *flexores* y *pronadores*, los del segundo son *extensores* y *supinadores*.

El grupo anterior, situado por delante del esqueleto antebraquial, está constituido por ocho músculos, distribuidos entre dos planos: *superficial*, *medio* y *profundo*.

El plano superficial consta de cuatro músculos que son, en dirección lateromedial: pronador terete, flexor radial del carpo, palmar largo y flexor lunar del carpo; ordinariamente presentan un tendón común de origen en el epicóndilo medial.

El plano medio está formado por el flexor superficial de los dedos. En el plano profundo está constituido por el flexor profundo de los dedos, el flexor largo del pulgar y el pronador cuadrado.

El *músculo pronador terete* está situado en la parte superior del antebrazo. [Figura B.1.](#)

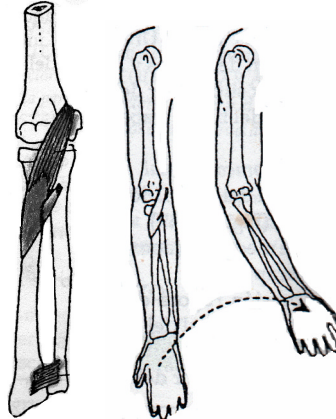
El *músculo pronador cuadrado* es plano y cuadrilátero, situado en la parte distal del antebrazo. [Figura B.1.](#)

El *músculo flexor radial del carpo* (tradicionalmente palmar mayor) está situado en la parte mediana de la cara anterior del antebrazo. Tiene acción como flexor y abductor de la mano. [Figura B.2 \(1\).](#)

El *músculo palmar largo* (tradicional palmar menor). Delgado y medial al músculo flexor radial del carpo. Tiene acción como flexor de la mano. [Figura B.2 \(2\).](#)

El *músculo flexor ulnar del carpo* (tradicional cubital anterior) es el más medial de los músculos superficiales del antebrazo. Tiene acción como flexor y aductor de la mano. [Figura B.2 \(3\)](#).

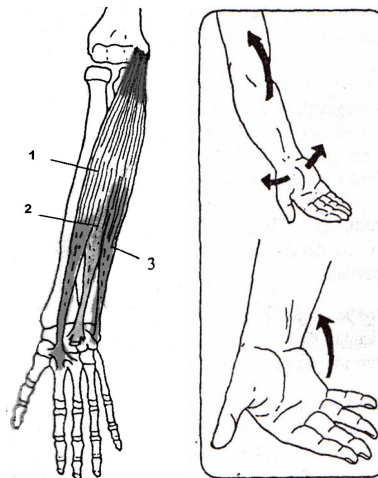
Figura B.1 Músculos *pronador terete* (parte superior) y *pronador cuadrado* (parte inferior).



Fuente [\[10\]](#)

El *músculo flexor superficial de los dedos* forma el plano medio. Tiene como acción flexor de la falange media de los dedos correspondientes. [Figura B.3](#).

Figura B.2 Músculos *flexor radial del carpo* (1), *palmar largo* (2), *flexor ulnar del carpo* (3).

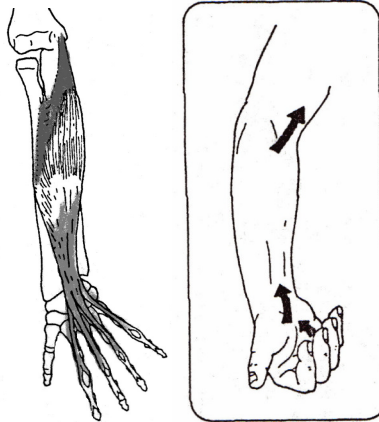


Fuente [\[10\]](#)

El *músculo flexor profundo de los dedos* (tradicionalmente flexor común profundo) cubre la ulna. Forma la mayor parte del relieve muscular que se

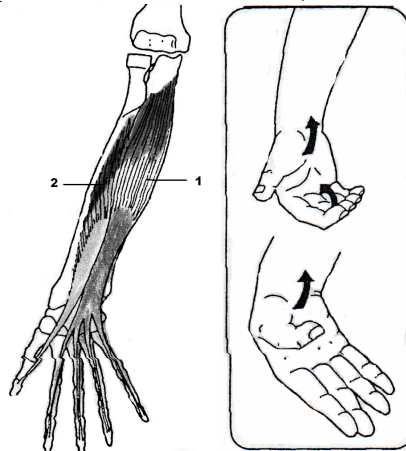
palpa en la parte posterior del antebrazo, medialmente al borde posterior de la ulna. Tiene como acción flexor de la falange distal de los dedos II a V, flexor de la mano. [Figura B.4 \(1\)](#).

Figura B.3 Músculo *flexor superficial de los dedos*.



Fuente [\[10\]](#)

Figura B.4 Músculos *flexor profundo de los dedos (1)*, *flexor largo del pulgar (2)*



Fuente [\[10\]](#)

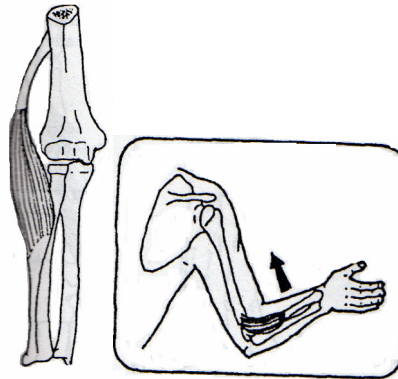
El grupo posterior está situado por detrás del esqueleto antebraquial. Los músculos están distribuidos en un plano superficial y otro profundo. El primero está constituido lateromedialmente por el braquiorradial, los dos extensores radiales del carpo, el extensor de los dedos, el extensor del dedo mínimo, el extensor lunar del carpo y el ancóneo. El plano profundo está formado por el supinador, abductor largo del pulgar, extensor breve del pulgar, extensor largo del pulgar y extensor del índice.

El *músculo braquirradial* (tradicional supinador largo), es el más superficial y lateral del antebrazo. Tiene como acción flexor del cúbito. [Figura B.5.](#)

El *músculo extensor radial largo del carpo* está situado por detrás del braquirradial. Tiene acción como extensor y abductor de la mano. [Figura B.6 \(1\).](#)

El *músculo extensor radial breve del carpo* está situado en la cara posterolateral del antebrazo. Tiene acción como extensor y abductor de la mano. [Figura B.6 \(2\).](#)

Figura B.5 Músculo *braquirradial*



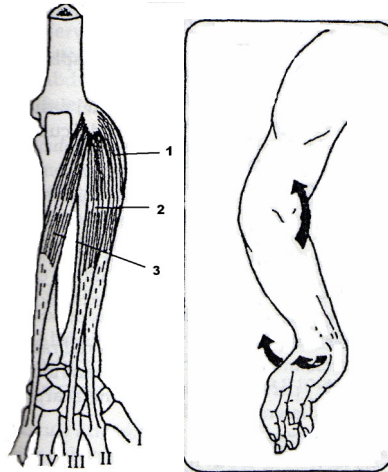
Fuente [\[10\]](#)

El *músculo extensor ulnar del carpo* cubre las caras posterior y medial de la ulna. Tiene acción como extensor y aductor de la mano. [Figura B.6 \(3\).](#)

El *músculo extensor de los dedos* está situado en la cara posterior del antebrazo. Tiene acción como extensor de los dedos (tiende a abducir los dedos índice anular y mínimo cuando los extiende, pero no tiene esta acción sobre el dedo medio), extensor de la mano. [Figura B.7 \(1\).](#)

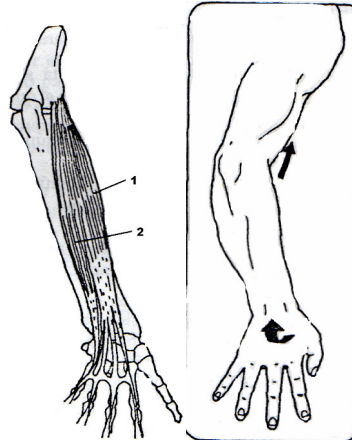
El *músculo extensor del dedo mínimo* está situado medialmente al extensor de los dedos, con el que está unido con frecuencia. Tiene como acción extensor y aductor de la mano. [Figura B.7 \(2\).](#)

Figura B.6 Músculos extensor radial largo del carpo (1), extensor radial breve del carpo (2), extensor ulnar del carpo (3).



Fuente [\[10\]](#)

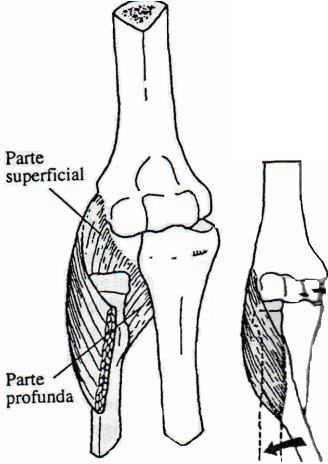
Figura B.7 Músculos extensores de los dedos (1), extensor del dedo mínimo (2).



Fuente [\[10\]](#)

El *músculo supinador* está enrollado sobre el tercio superior del radio. Tiene acción supinadora de antebrazo. [Figura B.8.](#)

Figura B.8 Músculo *supinador*



Fuente [\[10\]](#)

Anexo C

La transformada Wavelet

C.1. Introducción

Para comprender de forma mas clara en que consiste el análisis de señales mediante la transformada wavelet es necesario tener en cuenta la evolución a través del tiempo de herramientas matemáticas desarrolladas para tal fin.

En primer lugar se realiza un breve análisis de señales con la transformada de Fourier haciendo énfasis tanto en sus potencialidades como en sus limitaciones, dejando ver la necesidad de una herramienta que solucione estos inconvenientes, posteriormente la transformada de Fourier de tiempo corto (STFT) o transformada de Gabor, se presenta como alternativa de análisis al describir información de la señal tanto en tiempo como en frecuencia, usando ventanas de tiempo con un ancho definido.

La Transformada Wavelet, herramienta matemática desarrollada a mediados de los años '80, es eficiente para el análisis local de señales no estacionarias y de rápida transitoriedad y al igual que la Transformada de Fourier con Ventana (STFT), mapea la señal en una representación de tiempo-escala. El aspecto temporal de las señales es preservado. La diferencia está en que la Transformada Wavelet provee análisis de multiresolución con ventanas dilatadas. El análisis de las frecuencias de mayor rango se realiza usando ventanas angostas y el análisis de las frecuencias de menor rango se hace utilizando ventanas anchas [\[24\]](#).

Este capítulo se presenta como material base para entender el concepto y profundiza brevemente en la fundamentación matemática que sustenta el análisis de señales usando transformada wavelet, al presentar conceptos y definiciones útiles que facilitan su entendimiento.

Las Wavelets, funciones bases de la Transformada Wavelet, son generadas a partir de una función Wavelet básica, mediante traslaciones y dilataciones. Estas funciones permiten reconstruir la señal original a través de la

Transformada Wavelet inversa. La Transformada Wavelet no es solamente local en tiempo, sino también en frecuencia.

Dentro de los usos de esta poderosa herramienta podemos nombrar, además del análisis local de señales no estacionarias, el análisis de señales electrocardiográficas, sísmicas, de sonido, de radar, así como también es utilizada para la compresión y procesamiento de imágenes y reconocimiento de patrones.

C.2. Generalidades

Por razones de claridad se comienza definiendo el espacio métrico sobre el que se va a trabajar: el espacio $L^2 [-\infty, +\infty]$ de Hilbert.

C.2.1. Espacios de Hilbert

El espacio H de Hilbert es un espacio vectorial cuyos elementos pertenecen al plano complejo \mathbf{C} [8]. Sea \mathbf{H} el conjunto de elementos del espacio H . Los vectores complejos de este conjunto pueden ser sumados con las reglas usuales de la aritmética de vectores (propiedad aditiva) y multiplicados por escalares.

El espacio H posee una métrica y un producto interno. Considerando en particular el espacio H formado por funciones vectoriales fn . Si f y g son funciones del conjunto \mathbf{H} de H , el producto interno para este conjunto de funciones es un escalar definido por

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f^*(x)g(x)dx, \quad (\text{Ec. C.1})$$

Donde $f^*(x)$ es el complejo conjugado de $f(x)$ [9]. El producto escalar o interno de la función f con sí misma es un número real no negativo. En particular, si la función $f \in \mathbf{C}$, entonces satisface la condición:

$$\int_{-\infty}^{+\infty} |f(t)|^2 dt < \infty, \quad (\text{Ec. C.2})$$

Este espacio métrico recibe el nombre de Espacio de Hilbert $L^2 [-\infty, +\infty]$.

C.2.2. Ortogonalidad bases ortonormales

Se dice que dos vectores x e y son *ortogonales* en un Espacio Hilbert H si su producto interno es cero:

$$\langle X, y \rangle = 0$$

Se le llama *conjunto ortogonal* a aquel conjunto de vectores en el cual cualquier par de sus elementos es *ortogonal*. Además, este conjunto es *ortonormal* si la norma de los vectores es igual a uno:

$$\|x\| = \sqrt{\langle x, x \rangle} = 1$$

También se define a la *base ortonormal* de \mathbf{H} como un conjunto *ortonormal maximal* en \mathbf{H} si cualquier vector en \mathbf{H} puede ser representado como el límite de las combinaciones lineales de los elementos de una base ortonormal [9].

C.2.3. Análisis de Fourier

En 1807, Jean B. Fourier demostró que una función podía ser desarrollada en términos de series trigonométricas, y que se podían obtener, por integración, fórmulas para los coeficientes del desarrollo.

Un análisis de Fourier permite descomponer la señal original en sus componentes sinusoidales de diferentes frecuencias, lo cual puede ser interpretado como un cambio en la señal de su base de tiempo a una base en frecuencia (Figura C.1).

Figura C.1 Proceso de análisis de Fourier.



Fuente www.mathwoks.com

▪ C.2.3.1. Coeficientes y series de Fourier

Los desarrollos en Series de Fourier, [Ec. C.5](#), tienen dos aplicaciones fundamentales:

- (a) representar una función $f(x)$ definida en el intervalo $(-c, c)$, para valores de x en ese intervalo, o
- (b) representar una función periódica con período $2c$ para todos los valores de x .

La función $f(x)$ puede ser proyectada en una base ortonormal de funciones $\{\phi_k(x)\}$, de la siguiente forma:

$$F(x) = c_1\phi_1(x) + c_2\phi_2(x) + \dots + c_k\phi_k(x) + \dots \quad (-c < x < c) \quad (\text{Ec. C.3})$$
$$k = 1, 2, \dots$$

Se espera que el desarrollo de $f(x)$ converja a la función $f(x)$ [\[25\]](#).

Se puede demostrar que los coeficientes c_k de la suma son los coeficientes de Fourier de $f(x)$ con respecto a la base ortonormal $\{\phi_k(x)\}$ [\[20\]](#). Estos coeficientes pueden expresarse como:

$$c_k = \int_{-c}^c f(x)\phi_k^*(x)dx \quad k = 1, 2, \dots \quad (\text{Ec. C.4})$$

Siendo ϕ^* el complejo conjugado de ϕ .

La serie de la [Ec. C.3](#) con estos coeficientes es la *Serie de Fourier* genérica correspondiente a la función $f(x)$, y se define como:

$$f(x) = \sum_{k=1}^{\infty} c_k \phi_k(x) \quad (\text{Ec. C.5})$$

Si $f(x)$ está definida en el intervalo $(0, 2\pi)$ y determinada fuera de ese intervalo por $f(x+2\pi) = f(x)$, esto es, $f(x)$ tiene período 2π , la *serie de Fourier* que corresponde a $f(x)$ sobre la base ortogonal de senos y cosenos se define como:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sen kx), \quad (\text{Ec. C.6})$$

Donde los coeficientes de Fourier a_k y b_k se definen como:

$$\begin{cases} a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos kx \, dx \\ b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \operatorname{sen} kx \, dx, \end{cases} \quad \text{Con } k = 1, 2, \dots \quad (\text{Ec. C.7})$$

Puede observarse que los coeficientes de Fourier de la función transformada representan la contribución de cada función seno y coseno para cada frecuencia [\[24\]](#).

Usando la identidad de Euler: $e^{ikx} = \cos kx + i \operatorname{sen} kx$ podemos escribir la *serie de Fourier* de $f(x)$ como combinación lineal de funciones exponenciales complejas:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{ikx}, \quad (\text{Ec. C.8})$$

Donde las funciones $e_k(x) = \frac{e^{ikx}}{\sqrt{2\pi}}$ constituyen un conjunto ortonormal [\[9\]](#). Los coeficientes de Fourier de $f(x)$, respecto de esta base, pueden expresarse como:

$$c_k = \frac{1}{2\pi} \int_0^{2\pi} f(x) e^{-ikx} \, dx. \quad (\text{Ec. C.9})$$

Se puede notar que la *Serie de Fourier*, [Ec. C.5](#), no puede representar una función para todos los valores de x si la función no es periódica [\[5\]](#).

■ Transformada de Fourier

Para obtener una representación que pueda ser válida para todos los valores de x cuando $f(x)$ no es periódica, es natural intentar extender la representación anterior dejando que c tienda a infinito, lo que da lugar a la Transformada de Fourier [\[20\]](#).

La Transformada de Fourier de una función no periódica $f(x)$ está definida por

$$F(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(x) e^{-ikx} \, dx, \quad (\text{Ec. C.10})$$

Donde k es una variable real continua.

La función puede ser reconstruida a partir de sus componentes de Fourier, por medio de la transformada inversa de Fourier:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} F(k) e^{ikx} dk. \quad (\text{Ec. C.11})$$

La Transformada de Fourier en $\mathbf{L}^2[-\infty, +\infty]$ satisface las siguientes propiedades:

- Es una transformación de Fourier uno-a-uno de $\mathbf{L}^2[-\infty, +\infty]$ en sí mismo.
- Preserva la norma,

$$\int_{-\infty}^{+\infty} |f(x)|^2 dx = \int_{-\infty}^{+\infty} |F(k)|^2 dk. \quad (\text{Ec. C.12})$$

- Preserva el producto interno,

$$\int_{-\infty}^{+\infty} f(x)g^*(x) dx = \int_{-\infty}^{+\infty} F(k)G^*(k) dk. \quad (\text{Ec. C.13})$$

▪ Limitaciones del análisis de Fourier

La Transformada de Fourier es ampliamente utilizada en el procesamiento y análisis de señales con resultados satisfactorios en los casos en que estas señales son periódicas y lo suficientemente regulares, pero no ocurre lo mismo para el análisis de señales cuyo espectro varía con el tiempo (señales no estacionarias).

Tomando el caso en el que la función f a descomponer es una señal dependiente del tiempo, puede decirse que las funciones de la base de Fourier son de duración infinita en el tiempo, pero locales en frecuencia.

La Transformada de Fourier detecta la presencia de una determinada frecuencia pero no brinda información acerca de la evolución en el tiempo de las características espectrales de la señal. Muchos aspectos temporales de la señal, tales como el comienzo y el fin de una señal finita y el instante de aparición de una singularidad en una señal transitoria, no pueden ser analizados adecuadamente por el análisis de Fourier.

Para los casos de señales no estacionarias y transitorias se utiliza generalmente la *Transformada de Fourier con Ventana*.

▪ Transformada de Fourier con ventana

Una forma de analizar una señal no estacionaria es realizar un análisis espectral dependiente del tiempo. Una señal estacionaria es dividida en una secuencia de segmentos de tiempo en los cuales la señal puede ser considerada como cuasi-estacionaria y la Transformada de Fourier es aplicada a cada segmento local de la señal.

Gabor, en 1940, fue el primero en introducir la Transformada de Fourier de tiempo corto, conocida como la Transformada de Fourier con Ventana Deslizante, definida como:

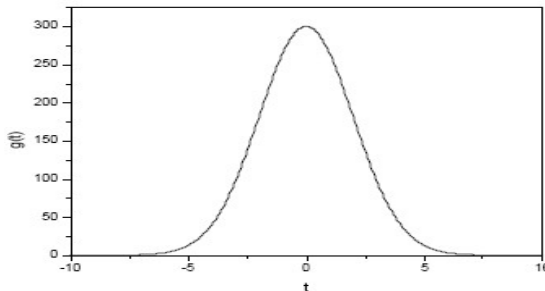
$$S_f(\omega, \tau) = \int f(t) g^*(t-\tau) \exp(-i\omega t) dt \quad (\text{Ec. C.14})$$

Donde $g(t)$ es una ventana deslizante, la cual tiene un ancho fijo y se traslada a lo largo del eje temporal por un factor τ [24]. Así, propuso a la función Gaussiana como la función ventana $g(t)$ y demostró que la Transformada de Fourier de una ventana Gaussiana continúa siendo Gaussiana [24].

La función está definida como (Figura C.2)

$$g(t) = \frac{1}{s} \exp\left(-\frac{\pi t^2}{s^2}\right) \quad (\text{Ec. C.15})$$

Figura C.2 Forma de onda de ecuación Gaussiana.



Fuente autores

Con la Transformada de Fourier con Ventana se logra una mejor localización de la aparición de una singularidad en una señal. Pero solo se conocerá en qué intervalo de tiempo se produce la singularidad, debido a que la localización depende del ancho elegido para la función ventana. Además, los eventos no

podrán ser observables si aparecen muy cerca unos de otros, ya que no será posible distinguirlos dentro de una misma amplitud de ventana ([Figura C.3](#)) [8].

Una herramienta matemática que permite resolver estos problemas es la Transformada Wavelet. Este tipo de transformada es capaz de concentrarse en fenómenos transitorios y de alta frecuencia mejor que la Transformada de Fourier con Ventana. Con esta última, una vez que el tamaño de la ventana es elegido, todas las frecuencias son analizadas con las mismas resoluciones de tiempo y frecuencia, distinto de lo que sucede en la Transformada Wavelet que tiene un tamaño de ventana adaptado a las frecuencias.

Figura C.3 Proceso de análisis de Fourier de tiempo corto.



Fuente www.mathwox.com

C.2.4. Bases Ortonormales

▪ Bases de la función escala

Las *funciones de escala* tienen el papel de funciones promedio. La correlación entre la función de escala y una función continua arbitraria produce la aproximación promediada de la última.

La función de escala básica $\phi(t)$, dilatada por un factor de escala 2^i , es desplazada con un parámetro de escala discreto de traslación k ,

$$\phi_{i,k}(t) = 2^{-i/2} \phi(2^{-i}t - k). \quad (\text{Ec. C.16})$$

Las funciones de escala básica $\phi(t)$ que se emplean satisfacen la condición de ortogonalidad, tal que las traslaciones discretas $\{\phi(t-k)\}$ con $k \in \mathbf{Z}$, forman un conjunto ortonormal [26]. La proyección de una función $f(t) \in \mathbf{L}^2(\mathbf{R})$ en la base ortonormal $\{\phi(t-k)\}$ es una correlación entre la función $f(t)$ original y la función de escala $\phi(t)$ muestreada a intervalos enteros.

Como resultado de la proyección de $f(t)$ en la base de la función de escala, se obtiene una aproximación menos detallada de $f(t)$. Todas las aproximaciones de $f(t)$ forman un subespacio $V_0 \in \mathbf{L}^2(\mathbf{R})$. El espacio vectorial V_0 puede ser interpretado como el conjunto de todas las posibles aproximaciones de la función en $\mathbf{L}^2(\mathbf{R})$ generado por el conjunto ortonormal $\{\phi(t-k)\}$.

Las funciones de escalas para todas las escalas $s = 2^i$ con $i \in \mathbf{Z}$, generadas a partir de la misma $\phi(t)$, son todas de forma similar. Debido a que la función de escala básica $\phi(t)$ genera la base ortonormal $\{\phi(t-k)\}$ de V_0 , con un paso de traslación entero, la función de escala dilatada $\phi(t/2)$ generará la base ortonormal $\{\phi(2^{-1}t-k)\}$ de V_1 con un paso de traslación igual a 2, y $\phi(t/4)$ generará la base ortonormal $\{\phi(2^{-2}t-k)\}$ de V_2 con un paso de traslación igual a 4, y así sucesivamente. Existe entonces un conjunto de bases ortogonales de las funciones de escala. Cada base de la función de escala es ortonormal en el espacio de la misma escala:

$$\langle \phi_{i,k}, \phi_{i,n} \rangle = \delta_{k,n} \quad (\text{Ec. C.17})$$

Para todo k y $n \in \mathbf{Z}$.

Las proyecciones en $\mathbf{L}^2(\mathbf{R})$ sobre el conjunto de bases ortonormales de la función de escala, forman un conjunto de subespacios V_i . Cada subespacio V_i es el conjunto de todas las posibles aproximaciones de la función en $\mathbf{L}^2(\mathbf{R})$ generado por la base ortonormal de la función de escala $\{\phi(2^{-i}t-k)\}$. El subespacio V_i es abarcado por la base ortonormal de la función de escala en el nivel de resolución i . Por lo tanto, la función de escala $\phi(t)$ genera los subespacios del análisis multiresolución [26].

Las aproximaciones de una función $f(t)$ en diferentes resoluciones deben ser similares, ya que son todas generadas por la misma función de escala con escalas diferentes. Los espacios de aproximación V_i pueden ser, entonces, deducidos unos de otros por simple dilatación:

$$f(t) \in V_i \Leftrightarrow f(2t) \in V_{i-1}. \quad (\text{Ec. C.18})$$

Toda la información útil para calcular la función de aproximación en el nivel de menor resolución i , está contenida en la función de aproximación en el nivel de mayor resolución $(i-1)$. Entonces, V_i es un subespacio de V_{i-1} [5].

C.2.5. ANALISIS MULTIRESOLUCION

El análisis multiresolución es una técnica que permite analizar señales en múltiples bandas de frecuencia. Consiste de una secuencia de subespacios cerrados V_i en $L^2(\mathbf{R})$:

$$\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots \subset L^2(\mathbf{R}) \quad (\text{Ec. C.19})$$

Cuando la resolución se incrementa con i tendiendo a $-\infty$, la función aproximada debería converger a la función original. Esto es:

$$\overline{\bigcup_i V_i} = L^2(\mathbf{R}), \quad (\text{Ec. C.20})$$

Por el contrario, cuando la resolución se decrementa a cero con i tendiendo a $+\infty$, las aproximaciones contienen cada vez menos información y convergen a cero:

$$\bigcap_i V_i = \{0\}, \quad (\text{Ec. C.21})$$

con $i \in \mathbf{Z}$ [24].

C.2.6. Bases Wavelet

Debido a que la proyección de una función sobre la base de la función de escala ortonormal es una aproximación menos detallada de la función en un nivel de resolución particular, se pierde algo de información en el proceso, esto significa que la función de escala ϕ no es completa a cualquier nivel de resolución. Por lo tanto, se usan las proyecciones sobre otras funciones, denominadas *wavelet ortonormales* (o simplemente *wavelets*), para obtener la información complementaria de los detalles de la función.

Como se verá más adelante, las wavelets son generadas a partir de la *wavelet madre* $\psi(t)$ por traslaciones y dilataciones discretas

$$\psi_{i,k}(t) = 2^{-i/2} \psi(2^{-i}t - k). \quad (\text{Ec. C.22})$$

Cuando la transformada de Fourier $\psi(w)$ de la wavelet madre satisface la condición de ortogonalidad [24], las traslaciones discretas de las wavelet madre

$\{\psi(2^i t - k)\}$ forman una base ortonormal para cada escala 2^i . Más aún, en el mismo nivel de resolución, el conjunto de traslaciones wavelet es ortogonal al conjunto de traslaciones de la función de escala en el espacio de la misma resolución

$$\langle \phi_{i,k}, \psi_{i,n} \rangle = 2^{-i} \int \phi_i(t-k) \psi_i(t-n) dt = 0 \quad (\text{Ec. C.23})$$

Para todo k y $n \in \mathbf{Z}$.

La proyección de $f(t)$ sobre las bases wavelet ortonormales es una correlación entre $f(t)$ y $\psi(t)$ muestreada a intervalos discretos. Las proyecciones de las funciones en $\mathbf{L}^2(\mathbf{R})$ sobre la base wavelet ortonormal $\{\psi(2^i t - k)\}$, forman un subespacio W_i . El subespacio W_i es abarcado por $\{\psi(2^i t - k)\}$.

Como la base wavelet $\{\psi(2^i t - k)\}$ es ortogonal a la base de función de escala $\{\phi(2^i t - k)\}$, dentro de la misma escala, el subespacio W_i es el complemento ortogonal del subespacio V_i :

$$W_i \perp V_i \quad (\text{Ec. C.24})$$

Tanto V_i como W_i son subespacios de V_{i-1} : $V_i, W_i \in V_{i-1}$ [24], y en razón de que W_i es el complemento ortogonal de V_i , el subespacio V_{i-1} es la suma directa de V_i y W_i :

$$V_{i-1} = V_i \oplus W_i. \quad (\text{Ec. C.25})$$

C.3. Transformada Wavelet

De manera muy general, la Transformada Wavelet de una función $f(t)$ es la descomposición de $f(t)$ en un conjunto de funciones $\psi_{s,\tau}(t)$, que forman una base y son llamadas las “Wavelets” [24]. La Transformada Wavelet se define como:

$$W_f(s, \tau) = \int f(t) \psi_{s,\tau}^*(t) dt. \quad (\text{Ec. C.26})$$

Las Wavelets son generadas a partir de la traslación y cambio de escala de una misma función wavelet $\psi(t)$, llamada la “Wavelet madre”, y se define como:

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right), \quad (\text{Ec. C.27})$$

Donde s es el factor de escala, y τ es el parámetro de traslación.

Las wavelets $\psi_{s,\tau}(t)$ generadas de la misma función wavelet madre $\psi(t)$ tienen diferente escala s y ubicación τ , pero tienen todas la misma forma. Se utilizan siempre factores de escala $s > 0$. Las Wavelets son dilatadas cuando la escala $s > 1$, y son contraídas cuando $s < 1$. Así, cambiando el valor de s se cubren rangos diferentes de frecuencias. Valores grandes del parámetro s corresponden a frecuencias de menor rango, o una escala grande de $\psi_{s,\tau}(t)$. Valores pequeños de s corresponden a frecuencias de mayor rango o una escala muy pequeña de $\psi_{s,\tau}(t)$ [\[25\]](#).

C.3.1. Wavelets Ortonormales y discretas

Cuando la función $f(t)$ es continua y las wavelets son continuas con factor de escala y traslación discretas, la Transformada Wavelet resulta en una serie de coeficientes wavelets, y es llamada la descomposición en Series Wavelet [\[24\]](#).

La función $f(t)$ puede ser reconstruida desde los coeficientes wavelets discretos $Wf(s,\tau)$, de la siguiente manera:

$$f(t) = A \sum_s \sum_\tau W_f(s,\tau) \psi_{s,\tau}(t), \quad (\text{Ec. C.28})$$

Donde A es una constante que no depende de $f(t)$.

A estas funciones wavelets continuas con factores de escala y traslación discretos se las denomina *Wavelets discretas* [\[24\]](#). Los factores de escala y traslación de las wavelets discretas pueden ser expresados como:

$$s = s_0^i \text{ y } \tau = k\tau_0 s_0^i \quad (\text{Ec. C.29})$$

Donde el exponente i y la constante k son enteros, y $s_0 > 1$ es un paso fijo de dilatación.

El parámetro de traslación τ depende del paso de dilatación s , [Ec. C.29](#). Entonces, a partir de la [Ec. C.27](#) y con la [Ec. C.29](#), las correspondientes wavelets discretas quedan expresadas como:

$$\psi_{i,k}(t) = s_0^{-i/2} \psi(s_0^{-i}(t - k\tau_0 s_0^i)) = s_0^{-i/2} \psi(s_0^{-i}t - k\tau_0) \quad (\text{Ec. C.30})$$

A través de la [Ec. C.26](#), la Transformada Wavelet de una función continua es realizada a frecuencias y tiempos discretos que corresponden a muestreos con distintas traslaciones (tiempo) y distintas dilataciones (o cambios de escala).

El paso de muestreo en tiempo es pequeño para el análisis utilizando wavelets de pequeña escala, mientras que es grande para el análisis con wavelets de gran escala. La posibilidad de variar el factor de escala s permite usar wavelets de escala muy pequeña para concentrar el análisis en singularidades de la señal. Cuando solo los detalles de la señal son de interés, unos pocos niveles de descomposición son necesarios. Por lo tanto el análisis wavelet provee una forma más eficiente de representar señales transitorias.

A modo de ejemplo, podemos hacer una analogía entre el análisis de Wavelet y el microscopio. Así, el factor de escala s_0^i corresponde al aumento o resolución del microscopio y el factor de traslación τ corresponde a la ubicación donde se hace la observación con el microscopio. Si queremos mirar detalles muy pequeños, el aumento y la resolución deben ser grandes, lo que se corresponde con un i grande y negativo. Esto da lugar a una función wavelet muy concentrada, y a pasos de traslación pequeños. Para un valor de i grande y positivo, la wavelet se extiende y los pasos de traslación son adaptados a esa amplitud [\[24\]](#).

Eligiendo adecuadamente $\psi(t)$ y los parámetros s_0, τ_0 , es posible lograr que las funciones $\psi_{s,\tau}(t)$ constituyan una base ortonormal de $\mathbf{L}^2(\mathbf{R})$. En particular si se elige $s_0 = 2$ y $\tau_0 = 1$, entonces existe $\psi(t)$, con buenas propiedades de localización tiempo–frecuencia, tal que $\psi_{s,\tau}(t)$ constituye una base ortonormal $\mathbf{L}^2(\mathbf{R})$ [\[25\]](#).

De esta forma, si las funciones wavelets discretas forman una base ortonormal, una función $f(t)$ de soporte finito puede ser reconstruida como una suma de los coeficientes wavelets discretos $W_f(s, \tau)$ multiplicados por las funciones de la base, como sigue:

$$f(t) = \sum_s \sum_\tau W_f(s, \tau) \psi_{s,\tau}(t). \quad (\text{Ec. C.31})$$

Una descomposición wavelet ortonormal no posee información redundante y representa la señal en forma unívoca. Una base wavelet ortonormal es posible con wavelets con factores de traslación y dilatación discretos [\[26\]](#). Por lo tanto,

para estas funciones wavelets discretas ortogonales, los productos internos son iguales a cero:

$$\int \psi_{i,k}^*(t) \psi_{m,n}(t) dt = \begin{cases} 1 & \text{si } i=m \text{ y } k=n \\ 0 & \text{en otro caso} \end{cases} \quad (\text{Ec. C.32})$$

En 1986 Meyer y Mallat [26] demostraron que la descomposición y reconstrucción wavelet ortonormal podrían ser implementadas en el marco del análisis multiresolución de señales.

C.3.2. Relación Dos-Escala

Con sus traslaciones discretas, las funciones de escala y las de wavelets forman dos bases ortonormales en cada nivel de resolución. Las funciones de escala y las wavelets en múltiples niveles de resolución son la versión dilatada de la función de escala básica y de la wavelet madre, respectivamente.

Sea $\phi(t)$ la función de escala básica cuyas traslaciones generan el subespacio V_0 . Entonces $\phi(t)$ puede ser expresada como combinación lineal de la suma ponderada del conjunto $\{\phi(2t-k)\}$ generado por $\phi(2t)$. Así las funciones de escala en dos niveles de resolución adyacentes satisfacen la relación dos-escala:

$$\phi(t) = \sum_k p(k) \phi(2t-k), \quad (\text{Ec. C.33})$$

Que puede ser considerada como la proyección de la función $\phi(t) \in V_0$ en el subespacio de mayor resolución V_{-1} . Esta relación es la ecuación fundamental en el análisis multiresolución.

La secuencia $p(k)$ es el *coeficiente interescala*, correspondiente a un filtro discreto pasobajo [24].

Sea $\psi(t) \in V_0$ la wavelet madre, la cual puede ser desarrollada en la base ortonormal de la función de escala $\{\phi(2t-k)\}$ en V_{-1} como:

$$\psi(t) = \sum_k q(k) \phi(2t-k), \quad (\text{Ec. C.34})$$

Donde la secuencia $q(k)$ es el coeficiente de interescala, correspondiente a un filtro discreto paso-alto [24]. Esta relación dos-escala permite generar las wavelets a partir de las funciones de escala.

En el lado izquierdo de las relaciones [C.33](#) y [C.34](#), $\phi(t)$ y $\psi(t)$ son continuas. En el lado derecho de las relaciones, los coeficientes interescala, $p(k)$ y $q(k)$, son discretos.

C.3.3. Algoritmo piramidal. Descomposicion wavelet

Sea la función $f(t) \in V_0$ que puede ser representada como la combinación lineal de las funciones de escala trasladadas $\phi(t-k)$ en V_0

$$f(t) = \sum_k c_0(k) \phi(t-k), \quad (\text{Ec. C.35})$$

Con los coeficientes

$$c_0(k) = \langle f, \phi_{0,k} \rangle = \int f(t) \phi(t-k) dt. \quad (\text{Ec. C.36})$$

La función a ser analizada pertenece a V_0 , el cual corresponde al nivel de digitalización inicial al comenzar la descomposición. En el siguiente nivel de menor resolución $i = 1$, existen dos subespacios mutuamente ortogonales $\{\phi_{i,k}(t)\}$ y $\{\psi_{i,k}(t)\}$, respectivamente. Debido a que V_0 es la suma directa de V_1 y W_1 , existe una única forma de expresar una función $f(t) \in V_0$, como combinación lineal de funciones v_1 y w_1 , donde $v_1 \in V_1$ y $w_1 \in W_1$. En particular, la función $f(t) \in V_0$ puede descomponerse en sus componentes a lo largo de V_1 y W_1 :

$$f = (P_1 + Q_1) f, \quad (\text{Ec. C.37})$$

Donde las dos componentes son las proyecciones ortonormales de $f(t)$ sobre V_1 y W_1 :

$$\begin{aligned} \text{(a)} \quad P_1 f &= \sum_n c_1(n) \phi_{1,n}, \\ \text{(b)} \quad Q_1 f &= \sum_n d_1(n) \psi_{1,n}. \end{aligned} \quad (\text{Ec. C.38})$$

Multiplicando ambos lados de la [Ec. C.37](#) por $\phi_{1,k}$ y calculando los productos internos, se obtiene:

$$\langle \phi_{1,k}, f \rangle = \langle \phi_{1,k}, P_1 f \rangle. \quad (\text{Ec. C.39})$$

Haciendo lo mismo en la [Ec. C.38 \(a\)](#) pero multiplicando por $\phi_{1,n}$ y usando la [Ec. C.35](#), se obtiene:

$$\begin{aligned}
c_1(k) &= \langle \phi_{1,k}, f \rangle = \langle \phi_{1,k}, P_1 f \rangle & (\text{Ec. C.40}) \\
&= \sum_n \langle \phi_{1,k}, \phi_{0,n} \rangle c_0(n)
\end{aligned}$$

Donde el producto interno entre los dos conjuntos de la función de escala $\{\phi_{1,k}\}$ y $\{\phi_{0,n}\}$ se puede calcular como

$$\begin{aligned}
\langle \phi_{1,k}, \phi_{0,n} \rangle &= 2^{-1/2} \int \phi\left(\frac{t}{2} - k\right) \phi(t - n) dt & (\text{Ec. C.41}) \\
&= 2^{1/2} \int \phi(t) \phi(2t - (n - 2k)) dt.
\end{aligned}$$

Sustituyendo $\phi(t)$ por la relación dos-escala en la [Ec. C.41](#) y usando la ortonormalidad del conjunto $\{\phi(2t)\}$ se obtiene

$$c_1(k) = 2^{-1/2} \sum_n p(n - 2k) c_0(n). \quad (\text{Ec. C.42})$$

La secuencia $c_1(k)$ o *tendencia* contiene los coeficientes del desarrollo de la función continua $f(t)$ en la base de la función de escala continua $\{\phi_{1,k}\}$ en V_1 . La secuencia $c_1(k)$ representa la versión suavizada de los datos originales $c_0(n)$.

Simultáneamente, multiplicando ambos lados de las [Ec. C.37](#) y [C.38 \(b\)](#) por la wavelet $\psi_{1,n}$ y calculando los productos internos, se obtiene:

$$\begin{aligned}
d_1(k) &= \langle \psi_{1,k}, Q_1 f \rangle = \langle \psi_{1,k}, f \rangle & (\text{Ec. C.43}) \\
&= \sum_n \langle \psi_{1,k}, \phi_{0,n} \rangle c_0(n),
\end{aligned}$$

Y siguiendo los pasos aplicados para la obtención de $c_1(k)$ se llega a que:

$$d_1(k) = 2^{-1/2} \sum_n q(n - 2k) c_0(n). \quad (\text{Ec. C.44})$$

De acuerdo con la [Ec. C.37](#), la proyección ortonormal $Q_1 f$ sobre W_1 es la información de *detalle* de $f(t)$. La secuencia $d_1(n)$ representa la diferencia entre la $f(t)$ original y la aproximación $P_1 f$, y se conoce como los *coeficientes wavelet discretos*.

La descomposición en aproximaciones suavizadas y detalles a menor resolución se puede continuar tanto como se desee. Generalizando,

$$\begin{aligned}
 P_{i-1}f &= P_i f + Q_i f = \sum_k c_i(k)\phi_{i,k} + \sum_k d_i(k)\psi_{i,k}, \\
 c_i(k) &= 2^{-1/2} \sum_n p(n-2k)c_{i-1}(n), \\
 d_i(k) &= 2^{-1/2} \sum_n q(n-2k)c_{i-1}(n).
 \end{aligned}
 \tag{Ec. C.45}$$

La secuencia $c_i(n)$ y $d_i(n)$ pueden ser calculadas a partir de $c_{i-1}(n)$ por filtrado iterativo.

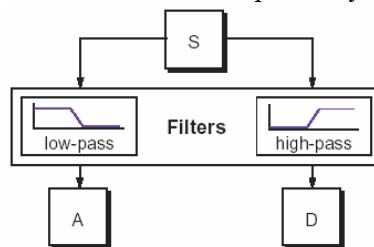
De esta manera, iterando hasta un nivel de resolución M , donde M toma un valor determinado, se puede representar la función original $f(t)$ por una serie de funciones detalle más una aproximación gruesa:

$$\begin{aligned}
 f(t) &= P_M f + Q_M f + Q_{M-1} f + \dots + Q_1 f, \\
 f(t) &= \sum_{k \in \mathbf{Z}} 2^{-M/2} c_M(k)\phi(2^{-M}t - k) + \sum_{i=1}^M \sum_{k \in \mathbf{Z}} 2^{-i/2} d_i(k)\psi(2^{-i}t - k).
 \end{aligned}
 \tag{Ec. C.46}$$

La [Ec. C.46](#) es la descomposición $f(t)$ en Series Wavelet [\[24\]](#). En esta descomposición wavelet las bases de la función de escala y las *bases wavelet* son todas continuas. Los coeficientes de aproximación $c_M(k)$ y los *coeficientes wavelet* $d_i(k)$ con $i = 1, 2, \dots, M$ y $k \in \mathbf{Z}$ son discretos.

Los coeficientes $c_1(n)$ y $d_1(n)$ se pueden calcular con un algoritmo discreto implementado por la aplicación recursiva de filtros discretos paso-alto y paso-bajo a las aplicaciones discretas $c_{i-1}(n)$ ([Figura C.4](#)).

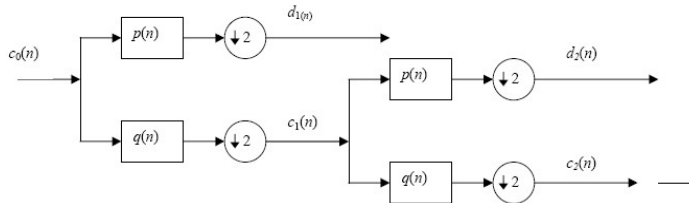
Figura C.4 Proceso de análisis Wavelet con filtros pasa-bajo y pasa-alto.



Fuente www.mathwooks.com

Este algoritmo es conocido como algoritmo piramidal o de Mallat. Los dos primeros pasos del algoritmo para calcular la descomposición wavelet se muestran en la [figura C.5](#) y en la [figura C.6](#) se muestra la representación en general como bancos de filtros.

Figura C.5 Esquema de la descomposición en Series Wavelet, con el algoritmo piramidal.

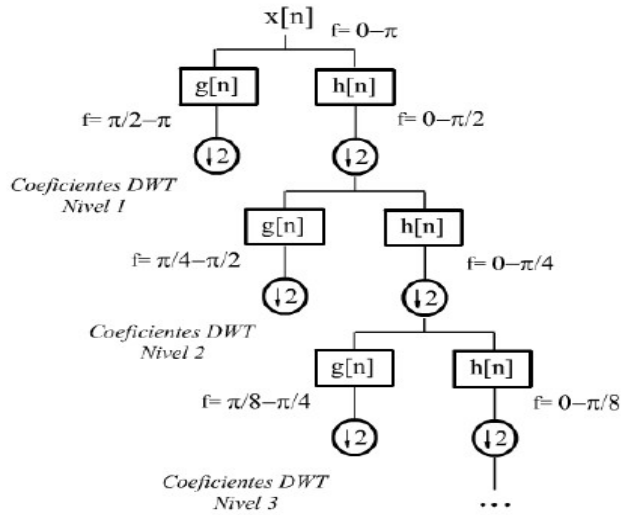


Fuente [\[24\]](#)

C.3.4. Algoritmo piramidal. Reconstrucción

La secuencia de la señal original $c_0(n)$ puede ser reconstruida a partir de las secuencias de coeficientes de aproximación $c_i(n)$ y de los *coeficientes wavelet* $d_i(n)$ con $0 < i \leq M$, donde $i = M$ es la menor resolución en la descomposición [\[24\]](#).

Figura C.6 Banco de filtros



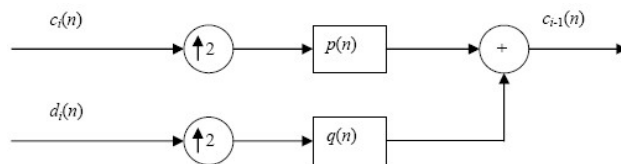
Fuente [24]

La aproximación discreta $c_{i-1}(n)$ en el próximo nivel de mayor resolución puede ser obtenida como la suma de dos convoluciones, una entre la aproximación discreta $c_i(n)$ y el filtro paso-bajo $p(n)$ y otra entre los *coeficientes wavelet* $d_i(n)$ y el filtro paso-alto $q(n)$,

$$\begin{aligned}
 c_{i-1}(n) &= \langle P_{i-1}f, \phi_{i-1,n} \rangle \\
 &= \sum_k c_i(k) \langle \phi_{i,k}, \phi_{i-1,n} \rangle + \sum_k d_i(k) \langle \psi_{i,k}, \phi_{i-1,n} \rangle \\
 &= 2^{-1/2} \sum_k c_i(k) p(n-2k) + 2^{-1/2} \sum_k d_i(k) q(n-2k),
 \end{aligned}
 \tag{Ec. C.47}$$

Como se muestra en la [Figura C.7](#). El proceso puede continuar hasta que la secuencia original $c_0(n)$ es reconstruida.

Figura C.7 Esquema de la reconstrucción Wavelet.

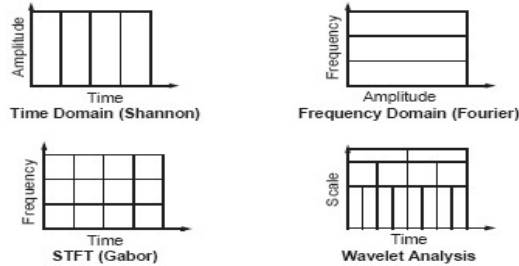


Fuente [24]

C.4. Análisis Tiempo-Frecuencia

El objetivo del análisis multiresolución es expandir una señal en una base de funciones cuyas propiedades tiempo-frecuencia se adapten a la estructura local de la señal. La Transformada Wavelet permite obtener el desarrollo de una señal en una base ortonormal de funciones wavelets, las cuales tienen propiedades de localización en tiempo y frecuencia similares a la Transformada de Fourier con Ventana ([Figura C.8](#)). En todos los casos anteriores, el patrón de descomposición del plano tiempo-frecuencia está predeterminado por la elección de las funciones de la base.

Figura C.8 Esquemas de los distintos espacios de operación para análisis.



Fuente www.mathwoks.com

El cálculo de la Transformada wavelet (CWT) se puede resumir en los pasos siguientes:

1. Tomar una wavelet y compararla –realizar un calculo– con la sección encontrada al inicio de la señal original.
2. Calcular un número, c , que representa cuan estrechamente correlacionada esta la wavelet con esta sección de la señal. Un alto valor de c indica una mayor similitud. En otras palabras, si la energía de la señal y la energía de la wavelet son iguales a uno, puede ser interpretado como un coeficiente de correlación.
3. Se debe tener en cuenta que los resultados dependen de la forma de la wavelet escogida.

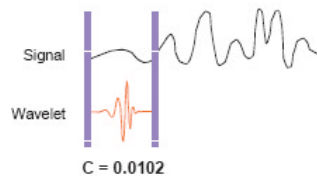


Figura C.9. Calculo de c . Fuente www.mathwoks.com

- Mover la wavelet a la derecha y repetir los pasos 1 y 2 hasta que halla cubierto por completo la señal.

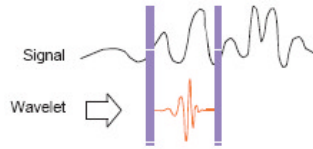


Figura C.10. Desplazamiento. Fuente www.mathwoks.com

- Escalar (dilatarse) la wavelet y repetir los pasos 1 a 3.

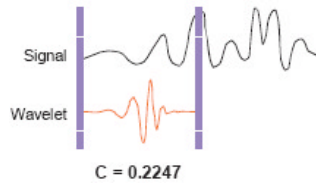
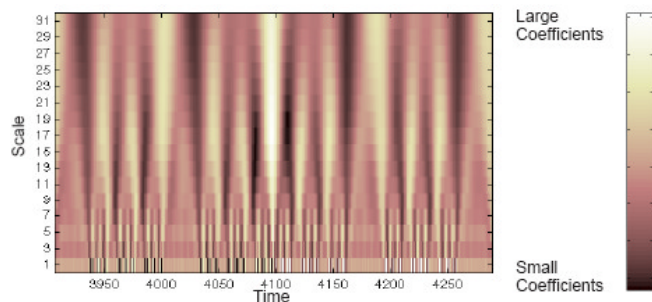


Figura C.11. Dilatación. Fuente www.mathwoks.com

- Repetir los pasos 1 a 4 para todas las escalas.

Una representación de la transformada wavelet se realiza tomando como eje x el tiempo y como eje y cada valor de escala escogido al realizar el cálculo, denominado plano escala-tiempo, y el valor de los coeficientes c es representado por un color de menor a mayor intensidad, tal como se observa en la [Figura C.12 \[26\]](#).

Figura C.12 Representación de la transformada Wavelet en el espacio escala-tiempo.



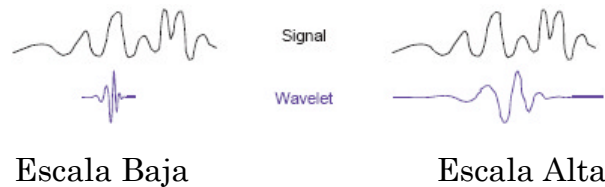
Fuente www.mathwoks.com

C.4.1. Escala y Frecuencia

Teniendo en cuenta que valores de escala grandes se identifican con formas de wavelet alargadas y viceversa, la relación entre la escala y la frecuencia es una relación inversa debido a la naturaleza de la información obtenida en cada escala, siendo evidente que una wavelet estrecha tiene la capacidad de identificar singularidades y detalles con mayor precisión, mientras que una wavelet estirada permite tener información de baja frecuencia (aproximaciones gruesas) en la señal original ([Figura C.13](#)) [26].

De esta forma, la relación correspondiente entre escala y frecuencia, en el análisis wavelet obtenido se relaciona así:

Figura C.13 Escalamiento de la función wavelet



Fuente www.mathwoks.com

- Baja escala $a \rightarrow$ Wavelet comprimida \rightarrow Detalles cambian rápidamente \rightarrow Alta frecuencia ω .
- Alta escala $a \rightarrow$ Wavelet alargada \rightarrow Cambios suaves, características gruesas \rightarrow Baja frecuencia ω .

C.4.2. La naturaleza de escala

Se debe tener en cuenta que aunque el análisis wavelet no provee una representación como tiempo-frecuencia, no significa que esto sea una debilidad de la técnica, al contrario es una fortaleza.

No solo el plano escala-tiempo es una forma diferente de ver los datos, es una manera natural de analizar un gran número de fenómenos derivados de la naturaleza, como es el caso de los fractales.

C.4.3. Número de operaciones

Consideremos ahora el número de operaciones requerido para la Transformada Wavelet ortonormal de un vector de datos. Sea L la longitud del vector de datos y N la longitud de los filtros $p(n)$ y $q(n)$. En la banda de frecuencia más alta, el primer paso de descomposición requiere $2NL$ multiplicaciones y sumas. En el algoritmo piramidal, en la siguiente banda de frecuencia más ancha la longitud del vector de aproximación discreta $c(n)$, está reducida a $N/2$. Por lo tanto, el próximo paso de descomposición requiere $2(NL/2)$ multiplicaciones y sumas. El total de operaciones de la descomposición wavelet ortonormal:

$$2\left(NL + \frac{NL}{2} + \frac{NL}{4} + \dots\right) = 2NL\left(1 + \frac{1}{2} + \frac{1}{4} + \dots\right) \cong 4NL. \quad (\text{Ec. C.48})$$

De esta forma la Transformada Wavelet ortonormal requiere sólo $O(L)$ operaciones. Esto es todavía más rápido que la Transformada de Fourier, la cual requiere $O(L \log_2 L)$ multiplicaciones y sumas.

Anexo D

Selección de la Wavelet Madre

D.1. Introducción

La transformada wavelet en el procesamiento digital de señales biológicas ha mostrado que su implementación promueve resultados favorables que no han sido posibles con otras técnicas de análisis espectral o en el dominio del tiempo; sin embargo, la Discrete Wavelet Transform (DWT) trae consigo la tarea de establecer metodologías que permitan elegir la Wavelet madre apropiada para realizar la representación mas adecuada en el espacio tiempo – frecuencia de la señal original. Esta desventaja, que no esta presente en la FT*, resulta de la gran diversidad de Wavelets madre que pueden ser utilizadas en análisis.

Encontrar metodologías para este fin no es tarea fácil, ya que la elección de la Wavelet madre depende en gran medida de los objetivos que se pretendan alcanzar con el análisis. Cada familia de Wavelets registrada en la literatura surge para cumplir propiedades específicas y cada una de ellas puede influir de manera positiva o negativa en los objetivos de una investigación.

Además, debido a la gran cantidad de Wavelets madre, no es práctico realizar pruebas con todas las familias, debido a la complejidad de realizar el proceso para algunas de ellas, a la dificultad de interpretar tantos resultados y al tiempo que demanda realizar los análisis. Debido a esto, muchas investigaciones proponen el uso de algunas Wavelets madre con carácter exploratorio, buscando mostrar resultados para una familia en particular, de tal forma que se elabore un registro para investigaciones futuras.

* La Transformada de Fourier solo utiliza señales exponenciales como bases para realizar la aproximación de cualquier tipo de señal, esto es una desventaja respecto a la Transformada Wavelet que permite elegir entre muchas bases para realizar la representación, sin embargo surge la ambigüedad para decidir cual de ellas debe ser utilizada.

D.2. Antecedentes de Metodologías planteadas

En el área de reconocimiento de patrones con señales EMG, *Englenhart et al* [18] plantearon la hipótesis que el buen funcionamiento de la transformada wavelet se debe principalmente al parecido de las wavelets madre con la señal original. Sin embargo, surge la inquietud de como medir de forma apropiada el parecido de una wavelet madre con la señal EMG.

No obstante, el Instituto Nacional de Astrofísica, Óptica y Electrónica se encuentra realizando una investigación titulada “OPTIMIZACIÓN DE TRANSFORMADA WAVELET APLICADA A BIOSEÑALES”^{*} que pretende mostrar metodologías que permitan medir el parecido de las wavelets madre con potenciales de acción generados artificialmente a partir de modelos matemáticos. Además, analizan el comportamiento de la transformada wavelet para procesos de filtrado y compresión de señales biológicas, las cuales son operaciones estándar en aplicaciones wavelet. Todo esto con el objetivo de explicar la superioridad de la transformada wavelet frente a otras técnicas de procesamiento de señales.

Para señales de tipo biológico en otras áreas como la electrocardiografía o la encefalografía, las metodologías existentes para elegir la wavelet madre se basan en resultados de investigaciones anteriores y tratan de explicar hipótesis sobre su buen funcionamiento a través de figuras de mérito que permitan medir ese buen comportamiento.

Otras metodologías buscan señales puras (construidas artificialmente) para agregarle ruido con distribución normal y evaluar el error medio encontrado después de aplicar el algoritmo. Este planteamiento es adecuado cuando se puede obtener una señal pura aceptable, lo cual no es posible con señales electro miográficas.

D.3. Metodologías propuestas por los autores

En vista que los estudios existentes para solucionar la ambigüedad al momento de elegir una wavelet madre se encuentran incompletos, los autores planteamos dos metodologías al respecto. Los esquemas enunciados a

^{*} El estudio se encuentra en curso y es realizado por los investigadores José Martínez-Bonetti, A. Z. Escudero, Adán-Martínez en el estado de Puebla, México.

continuación muestran la forma de implementación, resultados obtenidos, así como los inconvenientes y ventajas de cada metodología propuesta, esto con el fin que puedan ser utilizadas en otras investigaciones o en los trabajos futuros de la presente investigación.

D.3.1. Uso de la correlación cruzada

Con base en la idea planteada por el Instituto Nacional de Astrofísica, Óptica y Electrónica, una forma para medir el parecido de dos señales se realiza mediante la correlación cruzada normalizada, de esta forma el coeficiente de correlación es “1” cuando las dos señales son totalmente idénticas, “0” cuando no tienen ningún parecido y “-1” cuando las señales se encuentran invertidas.

MATLAB permite representar en el tiempo las wavelets madre mediante una serie de muestras obtenidas a partir del comando `wavefun` (), donde la cantidad de muestras que representan la Wavelet depende del número de iteraciones que se realice. Si mayor es la cantidad de iteraciones mejor representada se encuentra la wavelet madre, lo que indica una forma de modelar la frecuencia de muestreo.

La correlación cruzada de la wavelet madre con una señal EMG arroja un vector de coeficientes de correlación normalizados, que evalúan la semejanza para diferentes desplazamientos de la señal. El cálculo se realiza en MATLAB mediante el comando `XCORR` (), cuyo requerimiento es la equidad en el número de muestras de las dos señales.

Es posible, que las señales correspondientes a cada movimiento no se parezcan entre si, por esa razón es adecuado medir la similitud de la wavelet madre con las señales registradas por cada movimiento en particular, para seguidamente evaluar cuales wavelet madre se parece a la mayor cantidad de movimientos.

Después de implementar la metodología planteada se encontraron los siguientes resultados:

1. Los coeficientes de correlación obtenidos en el proceso dependen enormemente de la cantidad de muestras que se utilicen para realizar el cálculo, mostrando una gran variabilidad en los resultados al utilizar diferentes iteraciones para generar la wavelet madre.
2. El valor máximo de los coeficientes para diferentes variaciones de las pruebas no supera el valor de 0.1
3. Se considera conveniente generar funciones wavelets con iteraciones que no superen un orden “15” ya que el tiempo para construir funciones wavelets con mas iteraciones es muy elevado.

4. la cantidad de datos que se genera es muy elevada dificultando su análisis además que las variaciones que es necesario realizar elevan el tiempo de interpretación.
5. Los resultados son diferentes para las señales correspondientes a un mismo movimiento pero diferente canal.

De acuerdo a los resultados y los diversos resultados, se notó que es necesario realizar pruebas que evalúen la semejanza de la wavelet madre en diferentes segmentos de la señal que representa cada movimiento. Es necesario también observar la variabilidad de los resultados al cambiar de paciente.

D.3.2. Uso de la Discrete Wavelet Transform para elegir Wavelet Madre

Buscando practicidad en los análisis de tal forma que sea posible realizarlos automáticamente y tomar decisiones con el menor esfuerzo posible, los autores plantean un tercer criterio que permita elegir las wavelet madre.

Cuando se realiza un análisis con la DWT el rango de frecuencias procesado en cada nivel de resolución se muestra en la [tabla D.1](#).

Las señales EMG tienen su mayor contenido de información en la banda de frecuencias correspondiente a los 50 – 150 Hz. De acuerdo a esto y a la [tabla D.1](#). Los coeficientes del tercer nivel recopilan prácticamente toda esta información, lo que indica que si los coeficientes de detalle en el tercer nivel obtenidos para una wavelet madre en particular son altos, es porque la función wavelet es capaz de conservar adecuadamente esa información.

Tabla D.1. Rango de frecuencias analizadas en cada nivel de resolución.

Nivel de resolución	Rango de frecuencias en Hz
1er Nivel	250←→500
2do Nivel	125←→250
3er Nivel	62.5←→125
4to Nivel	31.25←→62.5
5to Nivel	15.625←→31.25

Fuente autores

Se propone el uso de una figura de mérito que permita evaluar cuáles wavelet madre representan mejor la señal EMG correspondiente a todos los movimientos. La figura de mérito es directamente el valor medio de los coeficientes de detalle en el tercer nivel de resolución.

Los resultados de la metodología propuesta presentan los mismos inconvenientes que el esquema propuesto en el [ítem D.3.2](#), por esa razón no es práctico utilizar estos esquemas propuestos hasta que se elabore un sistema de interpretación más compacto.

D.3.3. Observaciones

A pesar que se plantearon tres formas diferentes para decidir de forma práctica las wavelets madre que deben ser utilizadas en el proceso completo de identificación de movimientos, los resultados no fueron favorables, requiriendo esquematizar las propuestas de forma más compacta, con sistemas de interpretación más confiables y teniendo en cuenta las diversas variaciones del problema. Además, los tres métodos propuestos presentan la misma ambigüedad al tener que decidir sobre cual conjunto de familias wavelet deben ser aplicados, contando como único criterio aplicárselos a las familias que facilita MATLAB y las que menor tiempo de análisis demanden.

Como las investigaciones que pretenden dar solución a esta ambigüedad no se encuentran aún terminadas, y este no es el objetivo de la presente investigación, además que el tiempo para realizar esos análisis es elevado, se propone realizar pruebas pilotos que permitan identificar de modo rápido las wavelets madre que al ser utilizadas en el proceso completo de identificación presenten tasas de acierto mayores al 90%.

No obstante, para trabajos futuros se busca formalizar mejor las ideas aquí planteadas, así como otros métodos que permitan medir el grado de representabilidad que una wavelet madre tiene de la señal EMG*. Además, en busca de explicar el buen funcionamiento de la transformada wavelet para procesar señales biológicas es necesario estudiar la influencia de la función escala dentro del análisis.

* Una forma que se plantea para medir el grado de representabilidad, es comprimir la señal EMG mediante la DWT. Luego de comprimir la señal se evalúan parámetros como el porcentaje de energía retenido, el porcentaje de ceros y la diferencia entre la señal original y la acabada de procesar. Sin embargo esta metodología está aún en estudio por la complejidad de implementación.

Anexo E

Análisis de Componentes Principales*

E.1. Introducción [\[14\]](#), [\[29\]](#), [\[30\]](#), [\[31\]](#), [\[32\]](#)

En la práctica, cuando se recoge información correspondiente al fenómeno de la actividad eléctrica muscular, lo más frecuente es tomar el mayor número de características que describan adecuadamente el proceso. Esto sucede porque en general se quiere explorar su comportamiento. La señal EMG es entonces registrada y dividida en n segmentos, los cuales se representan mediante p descriptores (características calculadas) en el dominio del tiempo o la frecuencia para obtener n observaciones ([Figura E.1](#)).

Sin embargo, demasiadas características ($p > 3$) implican extensos cálculos que no permiten interpretar adecuadamente los datos; ya que las variables pueden estar fuertemente correlacionadas entre ellas y posiblemente brinden la misma información desde otro punto de vista. En este caso se tiene un exceso de información y muchos de los cálculos son innecesarios.

En su lugar, los esfuerzos se orientan sobre la posibilidad de reducir las variables sin pérdida significativa de información, de tal forma que se conserve el patrón que siguen los datos.

Los valores de $x_i(w_j)$ son datos o medidas cuantitativas (numéricas y de valor real).

* también conocido como descomposición de valores singulares o transformada de *Karhuemn Loeve*

Figura E.1 Formato de datos para realizar análisis de componentes principales

	X_1	X_2	...	X_i	...	X_p
W_1	$X_1(W_1)$	$X_i(W_1)$...	$X_p(W_1)$
W_2	.	medidas o datos			medidas o datos	
.	.	medidas		o	datos	
.	.	medidas		o	datos	
W_n	$X_1(W_n)$	$X_i(W_n)$...	$X_p(W_n)$

Vectores Característica u observaciones de cada segmento de la señal EMG

Fuente autores

El análisis de componentes principales es una técnica estadística y algebraica útil para la compresión de información y en la identificación de patrones sobre una muestra de datos. Más específicamente los objetivos del PCA son:

- Evaluar la semejanza entre las observaciones (muestras) a través de las medidas consideradas. Este objetivo en particular responde las siguientes inquietudes:
 1. ¿Existen grupos de muestras semejantes?
 2. ¿Se observa un patrón entre las observaciones?
- Evaluar la relación existente entre las características consideradas:
 1. ¿Existen grupos de variables correlacionadas entre ellas?
 2. ¿Se observa una tipología de variables?

Es importante resaltar que el objetivo principal del análisis de componentes principales, es detectar información redundante y re-organizarla de tal forma que sea fácil de interpretar. PCA permite reducir el número de variables manteniendo la mayor información posible.

Las variables originales son transformadas en otro conjunto de variables que no tienen relación entre si, midiendo el aporte de cada una de ellas sobre la

varianza total de los datos. Las nuevas variables no tienen redundancia de información y son combinaciones lineales de las variables originales.

De modo ideal, se buscan menos variables que las originales y almacenen la mayor parte de la información o variabilidad de los datos. EL concepto de mayor información en el contexto de PCA, se relaciona con mayor variabilidad o varianza sin realizar suposiciones sobre el comportamiento de las variables.

Si las variables originales no están inicialmente correlacionadas, no se justifica un análisis de componentes principales. De hacerlo, se obtendrían las mismas variables pero reordenadas de mayor a menor varianza. Para saber si X_1, \dots, X_p están correlacionadas, se puede calcular la matriz de correlaciones aplicándose posteriormente el test de esfericidad de *Barlett*.

E.2. Procedimiento [\[14\]](#), [\[32\]](#), [\[33\]](#)*

A continuación se presentan los pasos que deben realizarse para realizar un análisis de componentes principales:

E.2.1. Obtener los datos:

Considérese una serie de variables X_1, \dots, X_p que constituyen cada segmento de la señal EMG caracterizada ([Figura E.2](#)).

Figura E.2 Vectores de variables para realizar análisis de componentes principales

		Variables					
		X_1	X_2	...	X_i	...	X_p
muestras	W_1	$X_1(W_1)$	$X_i(W_1)$...	$X_p(W_1)$
	W_2	.	medidas			medidas	.
	.	.	o			o	.
	.	.	datos			datos	.
	W_J	$X_1(W_J)$	$X_i(W_J)$...	$X_p(W_J)$
.	.	medidas		o	datos	.	
.	
W_n	$X_1(W_n)$	$X_i(W_n)$...	$X_p(W_n)$	

Fuente autores

* El procedimiento presentado es expuesto en las referencias citadas.

Asumiendo n observaciones de cada variable X_i , se desea calcular a partir de ellas un nuevo conjunto de variables (y_1, y_2, \dots, y_p) sin correlación entre sí, cuyas varianzas vayan decreciendo progresivamente.

E.2.2. Estandarizar las variables originales:

Habitualmente, se calculan los componentes principales sobre variables originales estandarizadas, es decir, variables con media 0 y varianza 1.

Este proceso se realiza para evitar problemas de interpretación, como, ¿Qué significado tiene una variable artificial que ha sido creada como una mezcla de otras variables cuyas naturalezas pueden ser muy diferentes? ¿Qué nombre puede recibir por ejemplo, una variable conformada por un poco de edad, otro poco de peso, otro poco de ingresos, etc.? Teniendo en cuenta que las nuevas variables son combinación lineal de las variables originales.

Por otra parte, el peso de cada variable original, traducido fundamentalmente en variabilidad, puede ser muy diferente para cada una. Una variable muy dispersa puede contribuir enormemente a la varianza total, mientras que una variable más homogénea contribuye menos. Esto finalmente determina la participación de cada variable en la conformación de un factor.

Las variables estandarizadas no tienen nombre, son simplemente números sin unidades en las cuales se expresen las mediciones, la estandarización lleva todas las escalas a una escala común de media 0 y varianza 1, con lo cual se elimina el problema de medición y variabilidad diferente de las variables originales. El ACP realizado con variables originales estandarizadas se llama ACP normado.

Al estandarizar las variables originales, se obtiene la siguiente matriz (Z) de datos centrada, [figura E.3](#). Donde \bar{x}_j y S_j representan la media y la varianza de cada variable original respectivamente.

Es importante anotar que esta transformación no altera el comportamiento de las variables originales. Sin embargo, si se actúa de esta forma, se da igual importancia a todas las variables originales.

Figura E.3 Matriz de datos centrados*.

$$\begin{bmatrix}
 z_{11} & \cdots & z_{1j} & \cdots & z_{1p} \\
 \vdots & \vdots & \vdots & \vdots & \vdots \\
 z_{i1} = \frac{x_{i1} - \bar{x}_1}{s_1} & \cdots & z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j} & \cdots & z_{ip} = \frac{x_{ip} - \bar{x}_p}{s_p} \\
 \vdots & \vdots & \vdots & \vdots & \vdots \\
 z_{n1} & \cdots & z_{nj} & \cdots & z_{np}
 \end{bmatrix}$$

término general

Fuente autores

Resulta claro que el ACP normado debe ser la técnica a seguir en cualquier caso, a menos que se quieran explorar algunas otras posibilidades de tipo teórico o que se tengan variables muy similares tanto en su naturaleza como en su escala de medida.

E.2.3. Calcular la matriz de Covarianza.

Para realizar ACP corriente (variables originales sin estandarizar) se calcula la matriz de covarianzas sobre los datos originales y el [paso en E.2.4](#) se realiza sobre esta matriz.

Para el ACP normado se parte de la matriz de correlaciones de los datos originales y el [paso en E.2.4](#) se realiza sobre esta matriz.

Si las variables originales han sido estandarizadas, esto implica que la matriz de covarianzas es igual a la matriz de correlaciones. La correlación es la covarianza medida para valores estandarizados. La matriz de varianza – covarianza se muestra en la [figura E.4](#).

Donde S representa la varianza o la covarianza según corresponda. Se calculan sobre las variables x o sobre las variables z según se desee PCA corriente o PCA normado, respectivamente.

* media cero y varianza 1

Figura E.4 Matriz de covarianza

$$S = M = \begin{pmatrix} S_{z_1 z_1} & S_{z_1 z_2} & \cdots & S_{z_1 z_p} \\ S_{z_2 z_1} & S_{z_2 z_2} & \cdots & S_{z_2 z_p} \\ \vdots & \vdots & \ddots & \vdots \\ S_{z_p z_1} & S_{z_p z_2} & \cdots & S_{z_p z_p} \end{pmatrix}$$

Fuente autores

E.2.4. Calcular los valores y vectores propios:

Este paso resulta como solución a la condición de que las nuevas variables deben estar in-correlacionadas entre ellas mismas y deben ser ordenadas de mayor a menor varianza, lo que resulta en el habitual problema de maximizar una función de múltiples variables sujeta a restricciones, el cual queda completamente determinado por multiplicadores de *Lagrange* y el teorema de *Roché - Frobenius*.

En este paso se calculan los valores y vectores propios de la matriz de covarianza construida en el [paso E.2.3](#), estos son muy importantes dado que nos proporcionan información importante del conjunto de datos:

- Para PCA corriente, cada valor – propio corresponde a la varianza del componente y_i definido por el respectivo vector – propio, como se indica en la [Ec. E.1](#):

$$Var(y_i) = \lambda_i \quad Ec (E.1).$$

Si se suman todos los valores propios se obtiene la varianza total original, [Ec. E.2](#):

$$\sum_{i=1}^p Var(y_i) = \sum_{i=1}^p \lambda_i = \sum_{i=1}^p Var(X_i) \quad Ec (E.2).$$

La suma de las varianzas de las variables originales y la suma de las varianzas de las componentes, son iguales. Esto permite hablar del porcentaje de varianza total que recoge un componente principal:

$$\%Var_{tot}(PC_i) = \frac{\lambda_i}{\sum_{i=1}^p \lambda_i} \times 100$$

Ec (E.3).

Así, de la [Ec. E.3](#) también se podrá expresar el porcentaje de variabilidad recogido por los primeros m componentes, como se indica en la [Ec. E.4](#):

$$\%Var_{tot}(PC_m) = \frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^p Var(X_i)} \times 100 \quad m < p$$

Ec (E.4).

- Para PCA normado, en la matriz de correlaciones todos los elementos de la diagonal son iguales a 1, lo que implica que la suma de todos los autovalores es P, y la proporción de varianza recogida por el j-ésimo componente es :

$$\%Var_{tot}(PC_j) = \frac{\lambda_j}{P}$$

Ec (E.5).

Donde p es el número de variables que se están analizando.

E.2.5. Elegir componentes

Aquí es donde se hace evidente la noción de compresión de información y reducción de dimensión. De hecho, se tiene que el vector – propio con el más alto valor propio es la componente principal del conjunto de datos (porque es el que mayor porcentaje de varianza aporta sobre la varianza total).

Una vez teniendo los vectores propios de la matriz de covarianza, se procede a ordenarlos de mayor a menor valor propio. Esto nos da los componentes en orden de significancia.

En este punto, se puede decidir ignorar componentes de menor significancia, al hacer esto se pierde una pequeña parte de la información, pero si los valores propios son pequeños, la pérdida será tolerable. Si se ignoran algunas componentes, el conjunto de datos final, después del proceso de reconstrucción, tendrá dimensión menor al original.

Entonces, se necesita formar un vector característica, tomando los vectores propios que se deseen conservar y conformar una matriz con estos vectores propios como columnas:

$$\text{vector característica} = (eig_1, eig_2, \dots, eig_r) \quad Ec (E.6).$$

Se tienen 2 opciones:

- Se puede formar el vector característica con todos los vectores propios ($r = p$).
- Se pueden dejar fuera las componentes menos significantes para formar el vector característica ($r < p$).

Para PCA los vectores propios deben ser unitarios. Esto es particularmente útil, ya que los vectores propios deben mostrar el patrón que siguen los datos originales. Así por ejemplo, si los datos son bidimensionales ($p=2$), el vector propio con mayor valor propio, muestra la línea que mejor se ajusta a los datos (por eso se llama la componente principal), mientras el segundo vector propio mide que tanto se alejan los datos de esa línea (por eso se llama segunda componente principal). Este procedimiento se puede generalizar para dimensiones mayores, cuando se desea encontrar el hiperplano (patrón) que mejor se ajusta a los datos. Recordar que en dos dimensiones un hiperplano es una línea recta. Cada vector propio define una línea recta que mide la variación en cada dirección posible de los datos.

Si interesa PCA como método de compresión de información ($r < p$), para reducir la dimensión de los datos originales; entonces, ¿cómo saber cuáles son las componentes menos significantes? de tal forma que se pueda ignorar su información y no tenerlas en cuenta a la hora de formar el vector característica.

¿Como encontrar el valor de r que representa adecuadamente los datos originales?

Se pensaría que el mejor valor de r , es el menor y tan cercano a p como sea posible, y que además esas r componentes deberían ser las de mayor valor propio (porque contienen el mayor porcentaje de varianza). Una manera muy natural de encontrar el valor de r es conservar las componentes que mantengan un porcentaje adecuado de la varianza total, según el criterio de cada investigador y de acuerdo a la naturaleza del problema (un 70%, 80%, 90%, etc.). Sin embargo para determinar el valor de r existen criterios de optimización. A continuación se enuncian varios criterios:

- Según el criterio de **Káiser** [28] se deben conservar las componentes cuyo valor propio este por encima del promedio :

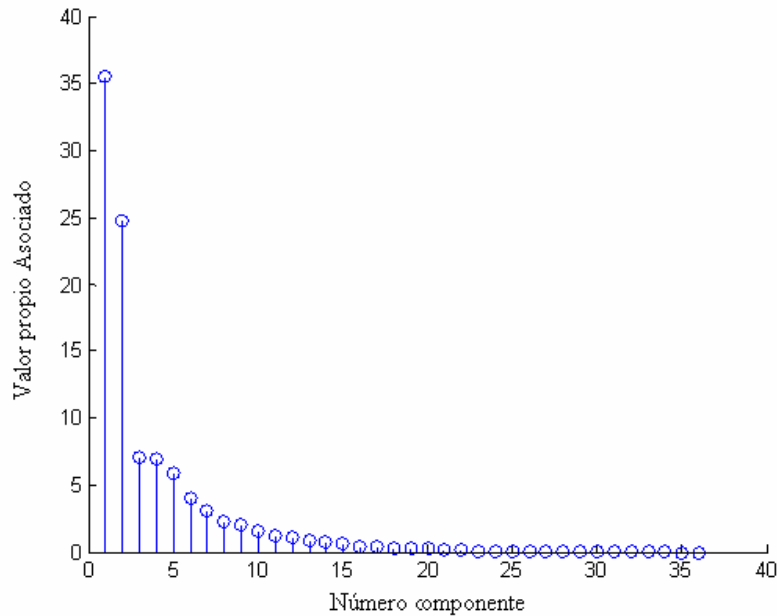
$$\lambda > \frac{V_T}{P} \quad Ec (E.7).$$

- Según el criterio de **Jolliffe** [27] se deben conservar las componentes cuyo valor propio sea mayor que 0.7:

$$\lambda > 0.7 \quad Ec (E.8).$$

- Calcular la correlación de las nuevas variables con las variables originales, y mantener aquellas componentes que tengan una correlación significativa de acuerdo a la variación de los datos. [32]
- Realizar un gráfico de componentes frente a valor propio asociado. Comenzar seleccionando componentes hasta que los restantes tengan aproximadamente el mismo valor propio. La idea es buscar un “codo” en el gráfico, es decir, un punto a partir del cual los valores propios son aproximadamente iguales y el criterio es mantener un número de componentes que excluya los asociados a valores pequeños y aproximadamente del mismo tamaño. La [figura E.5.](#) muestra la naturaleza de los componentes*.

* Por construcción, a medida que crece el orden del componente, su valor propio debe ir disminuyendo.

Figura E.5. Gráfica de codo para seleccionar el número de componentes

Fuente autores

En la [Figura E.5](#) se observa que a partir de la componente número 15 los valores propios son constantes y pequeños en comparación con su valor máximo.

No obstante, implementar este criterio requiere de supervisión visual, haciéndose necesaria una forma para aplicarlo automáticamente.

Para datos obtenidos a partir de señales EMG*, se realizaron pruebas pilotos y se encontró que la mayoría de las gráficas realizadas muestran que se deben conservar las componentes que tienen asociado un valor propio mayor al 2% del valor máximo†. De esta forma, es posible aplicar el criterio del “codo” automáticamente de acuerdo al comportamiento de los datos.

La manera de implementar automáticamente el criterio del “codo”, para efectos de la presente investigación, se denominó criterio “estándar” y es una alternativa propuesta por los autores; que consiste en retener aquellos componentes que tengan un valor propio mayor o igual al 2% del máximo correspondiente.

* Como los mostrados en la [figura E.5](#).

† Esto garantiza que se aplica correctamente el criterio del codo para conservar las componentes más significativas.

- Usar el criterio de validación cruzada, para determinar el número de componentes por usar y cuáles de ellas, así por ejemplo se pueden tomar una cantidad r , no precisamente las r primeras. Pueden ser de índices diferentes, esto acorde con el análisis recursivo de tomar un número de ellas, r , y alternar cuáles pueden dar mejor resultado para una aplicación concreta.

E.2.6. Derivar el nuevo conjunto de datos:

El paso final en PCA una vez que se han escogido las componentes (vectores propios), es tomar la transpuesta del vector característica y multiplicar por la izquierda del conjunto de datos estandarizados, como se muestra en la [Ec E.9](#):

$$DatosFinales = lineaVectorCaarcteristica \times lineaDatosAjustados \quad Ec (E.9).$$

En este punto lo que se obtiene son los datos estandarizados en términos de los vectores escogidos. Si se conservan todas las componentes, se obtienen los datos ajustados, rotados y sin pérdidas, con los vectores propios como nuevos ejes.

Básicamente se han transformado los datos, expresados en términos de los patrones hallados en ellos, donde los patrones son las líneas que mejor describen la relación entre ellos. Esto es particularmente útil porque se han expresado los datos como una combinación de las contribuciones de cada una de las líneas, removiendo las contribuciones menos significativas (en el caso que no se tengan todos los componentes).

Estos datos finales, con reducción de dimensionalidad son los que se utilizaran en el proceso de clasificación.

Puesto que uno de los objetivos que se persigue con el PCA es la representación de las observaciones en un espacio de pocas dimensiones, resulta interesante tener una medida de tal representación para cada muestra. Esa medida esta dada por la suma de cosenos cuadrados. Estos valores son los cuadrados de los cosenos de los ángulos formados por el vector que representa a cada muestra con los ejes del sistema de coordenadas factoriales. La suma de todos estos cosenos es igual a la unidad. Sin embargo, si se retienen r factores (componentes), la suma de los r cosenos mide el grado de representabilidad de cada muestra, siendo mejor representadas aquellas muestras para las cuales la suma de los r cosenos cuadrados este mas cerca de 1.

E.2.7. Volviendo al conjunto de datos Originales

Si se desea tener de vuelta el conjunto de datos original (transformada inversa), tras usar la transformación PCA para la compresión de información, hay que recordar que solo si se consideraron todos los componentes en la transformada, se tendrán de vuelta exactamente los datos originales. Sin embargo, si se ha reducido el número de componentes en la transformada final, entonces se recuperaran los datos con alguna pérdida de información.

$$\text{lineaDatosAjustados} = \text{lineaVectorCaracteristica}^T \times \text{DatosFinales} \quad \text{Ec. (E.10)}$$

Pero, para regresar al conjunto original, se debe multiplicar por la varianza y agregar la media de los datos originales (por el proceso de estandarización):

$$\text{DatosOriginal} = \text{varianza} \times (\text{VectorCaracteristica}^T \times \text{DatosFinales}) + \text{media} \quad \text{Ec (E.11)}.$$

La media y la varianza deben ser las de los datos originales. Lo anterior constituye la transformada PCA inversa.

Anexo F

Maquinas de Soporte Vectorial

F.1. Introduccion

En el área de reconocimiento de patrones y más específicamente en la parte de clasificación, las máquinas de soporte vectorial (SVM, *Support Vector Machine*), se han convertido en los últimos años en una de las técnicas más importantes sobre otras muy populares como K – ésimo vecino cercano (KNN), redes neuronales artificiales (ANN), árboles de clasificación (CART), las técnicas bayesianas y otras técnicas de aprendizaje heurístico. [\[23\]](#) , [\[35\]](#)

En contraste con el método bayesiano, las MSV no requieren ningún tipo de hipótesis sobre la densidad de probabilidad de los datos. Frente a las RNA, ofrece la ventaja de reducir la cantidad de parámetros que es necesario especificar, en las cuales se escoge el tipo de red a entrenar como primera medida, seguido de una función de activación, luego el numero de capas y a su vez el número de neuronas por capa, es decir, que la búsqueda se vuelve bastante compleja y su implementación requiere de gran esfuerzo. Además que su soporte matemático está fundamentado sobre bases muy sólidas que las hacen poseer múltiples ventajas sobre las otras técnicas mencionadas. [\[34\]](#)

Sin embargo, Lin presenta en perspectiva la posibilidad de hacer que las máquinas de soporte vectorial se conviertan en el principal método de clasificación argumentando que el problema de las SVM es el mal empleo que se les da probablemente por falta de conocimiento de la metodología. De acuerdo a esto, las redes neuronales y los árboles de clasificación permanecen como principales herramientas.

Lo que usualmente los usuarios desprevenidos están haciendo es convertir la información a clasificar al formato de algún programa SVM disponible, sin tener en cuenta en la mayoría de los casos las implicaciones del formato, escalamiento, etc., para luego tratar aleatoriamente con valores de parámetros y *kernels* indiscriminadamente sin hacer validación y sin saber de antemano que los parámetros por defecto en dichos programas son sorprendentemente

importantes y el hecho es que muchos de los usuarios obtienen como resultado valores de error y aprendizajes insatisfactorios. [23], [33], [35]

Lo mínimo que se espera que haga el usuario según Lin es escalar los datos para validación y entrenamiento, considerar diversos *kernels* y encontrar valores adecuados para cada parámetro involucrado en el entrenamiento de la máquina. Ahora, esto de encontrar “valores adecuados” a veces no es tarea fácil, sin mencionar que lo que se pretende no es encontrar valores adecuados sino los mejores valores para un caso dado. Hasta el momento, las técnicas de selección de parámetros o selección del modelo como también es llamado son las siguientes: búsqueda manual intuitiva, cotas para LOO (*leave one out*) o para riesgo empírico, búsqueda en dos sentidos y búsqueda en malla. [23]

Las SVM han sido utilizadas en el reconocimiento de dígitos manuscritos, reconocimiento de objetos, reconocimiento de voz, identificación de rostros en imágenes y categorización textual, así como en casos de estimación de regresiones. [34], [35], [37]

F.2. Técnica de aprendizaje en las msv lineales [6], [23]

Intuitivamente, dado un grupo de datos distribuidos en dos clases, una SVM lineal busca un hiper plano de tal manera que la mayor cantidad de puntos de la misma clase queden al mismo lado, mientras se maximiza la distancia de dichas clases al hiper plano. De acuerdo a *Vapnik* [34], este hiper plano minimiza el riesgo de clasificaciones erróneas en el grupo tomado para realizar el proceso de validación.

Para un grupo de entrenamiento compuesto de η pares *atributo-etiqueta* (\vec{x}_i, y_i) ,

$$\vec{x}_i \in \mathbb{R}^N \quad y \quad y_i \in \{+1, -1\} \\ 1 < i < \eta$$

Se desea obtener una función F tal que para una entrada en \mathbb{R}^N produzca una salida en $\{\pm 1\}$:

$$F: \mathbb{R}^N \rightarrow \{+1, -1\}$$

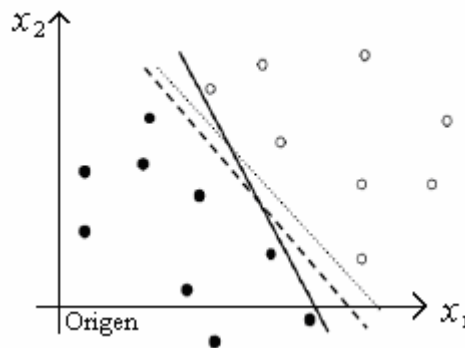
Para que así se pueda clasificar un nuevo dato (\vec{x}, y) . Téngase en cuenta que $y = F(\vec{x})$ para este nuevo dato y es generado con la misma probabilidad de los datos de entrenamiento.

Si ninguna restricción es impuesta a la clase de función F que se escoge, se pueden cometer errores en la estimación de una etiqueta para un nuevo atributo de entrada, pues aunque trabaje bien en los datos de entrenamiento, no necesariamente tiene una buena generalización para datos desconocidos, luego el aprendizaje es imposible y la minimización del error de *entrenamiento* no implica que se deba esperar un pequeño error de *test* ; sin embargo se pueden utilizar técnicas de optimización para minimizarlo.

F.3. Datos linealmente separables [6], [23]

Las maquinas de soporte vectorial están basadas en hiper planos que separan los datos de entrenamiento en dos subgrupos, de manera que aquellos datos con igual etiqueta queden al mismo lado del hiper plano. En un caso de dos dimensiones los hiper planos se representan por líneas rectas (figura F.1). Si existe este hiper plano se dice que los datos son linealmente separables.

Figura F.1. Hiperplanos que separan correctamente los datos



Fuente Autores

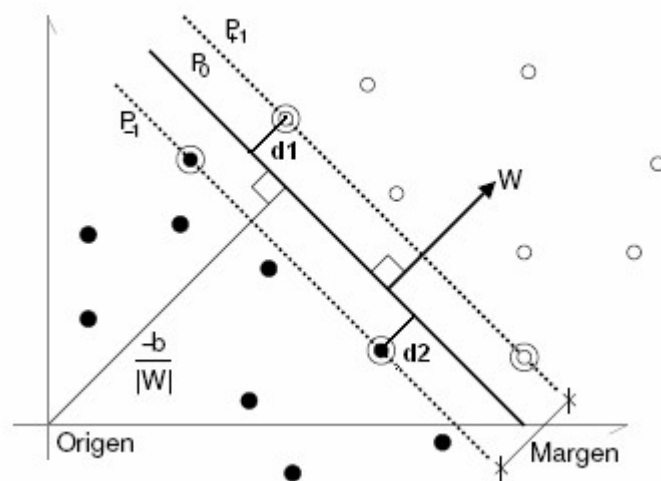
Un plano \mathbb{N} dimensional es de la forma $a_n x_n + a_{n-1} x_{n-1} + \dots + a_1 = 0$, que se puede volver a escribir en forma vectorial como $(\vec{w} \cdot \vec{x}_i) + b = 0$, donde $\vec{w} = [a_n, a_{n-1}, \dots, a_1]$ es un vector perpendicular al plano, $\vec{x}_i = [x_n, x_{n-1}, \dots, x_1]$ y $b = a_0$. La distancia del plano al origen es $\frac{|b|}{\|\vec{w}\|}$.

F.4. Hiperplano de Separación Óptimo [34]

Es claro de la [figura F.1](#) que si los datos son linealmente separables la existencia del hiperplano clasificador no es única. Cuando se elige el plano clasificador existe un único valor de b asociado a $\|\vec{w}\|$ e infinitud de vectores \vec{w} para definirlo. Entonces es necesario definir un criterio que permita establecer como encontrar \vec{w} y b para construir el plano clasificador. También es inmediato que si ambos conjuntos son separables, existe una distancia mínima entre ambos conjuntos denominada margen m , definida como la suma de las distancias d_1 y d_2 *

En medio de todos los posibles planos de separación entre las dos clases, existe un único hiperplano de separación óptimo (OSH), de forma que el margen sea máximo, con la intención de forzar el aprendizaje óptimo de la maquina

Figura F.2. Margen de separación entre dos conjuntos de datos



Fuente Autores

F.5. Maximización del margen [33], [34]

Para maximizar el margen, con el fin de encontrar los valores óptimos de \vec{w} y b , se proponen dos planos paralelos al plano OSH, representado por la ecuación $\vec{w} \cdot \vec{x} + b = 0$. Estos planos contienen los puntos más cercanos al OSH y

delimitan un margen m entre los conjuntos. Definiendo como P_{+1} y P_{-1} los planos que contienen los datos más cercanos al OSH de la clase 1 y la clase 2 respectivamente, las ecuaciones de dichos planos son:

$$\begin{aligned} P_{+1} &= \vec{w} \cdot \vec{x} + b = \varepsilon \\ P_{-1} &= \vec{w} \cdot \vec{x} + b = -\varepsilon \end{aligned}$$

Siendo el margen la distancia perpendicular entre P_{+1} y P_{-1} * se obtiene:

$$[(\vec{w} \cdot \vec{x}_{+1}) + b] - [(\vec{w} \cdot \vec{x}_{-1}) + b] = \varepsilon - (-\varepsilon)$$

$$\vec{w} \cdot (\vec{x}_{+1} - \vec{x}_{-1}) = 2\varepsilon$$

$$\frac{\vec{w} \cdot (\vec{x}_{+1} - \vec{x}_{-1})}{\|\vec{w}\|} = \frac{2\varepsilon}{\|\vec{w}\|}$$

$$m = \frac{2\varepsilon}{\|\vec{w}\|}$$

Se infiere que si se desea maximizar la magnitud del margen m , es necesario minimizar la magnitud de \vec{w} .

Debe notarse que entre los planos P_{+1} y P_{-1} no existen datos de entrenamiento y todos los datos deben cumplir:

$$\begin{aligned} (\vec{w} \cdot \vec{x}_i) + b &\geq +\varepsilon & \text{para } y_i = +1 \\ (\vec{w} \cdot \vec{x}_i) + b &\leq -\varepsilon & \text{para } y_i = -1 \end{aligned}$$

Normalizando las restricciones respecto a ε , se obtienen un nuevo \vec{w} y un nuevo b que permiten conseguir el hiper plano clasificador en forma canónica. Por lo tanto es común que se expresen las restricciones de la forma:

** Recordando que la distancia perpendicular de cualquier punto a un plano es

$$d = \frac{(\vec{w} \cdot \vec{x}_i) + b}{\|\vec{w}\|}$$

\vec{x}_{+1} y \vec{x}_{-1} son los datos más cercanos al OSH de la clase 1 y la clase 2 respectivamente

$$\begin{aligned} (\vec{w} \cdot \vec{x}_i) + b &\geq +1 & \text{para } y_i = +1 \\ (\vec{w} \cdot \vec{x}_i) + b &\leq -1 & \text{para } y_i = -1 \end{aligned}$$

La función decisión $f_{\vec{w}, b}(\vec{x}_i) = y_i$, se define como el signo que resulta de evaluar un dato en la ecuación del OSH:

$$f_{\vec{w}, b}(\vec{x}_i) = \text{sign}(g(\vec{x}_i)) = \text{sign}(\vec{w} \cdot \vec{x}_i + b)$$

Combinando las restricciones impuestas en P_{+1} y P_{-1} se obtiene:

$$\begin{aligned} y_i (\vec{w} \cdot \vec{x}_i + b) &\geq 1 \\ i &= 1, \dots, \eta \end{aligned}$$

Si existe un plano que satisfaga dicha restricción, se dice que los datos son linealmente separables. Para encontrar el OSH se debe maximizar el margen m teniendo en cuenta la restricción antes impuesta. Equivalente a resolver el siguiente problema:

$$\min \left\{ f(\vec{w}) = \frac{\|\vec{w}\|^2}{2} \right\}$$

Sujeto a

$$y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 \quad i = 1, \dots, \eta \quad b \in \mathfrak{R}$$

La función $f(\vec{w})$ es llamada función objetivo y junto con la restricción es llamado un problema de optimización cuadrático con restricciones. Por lo tanto este es un problema de optimización de tipo cuadrático sujeto a η restricciones.

Los problemas de este tipo son tratados introduciendo el método de multiplicadores de *Lagrange*.

Para esta clase de problemas con restricciones, se introducen los multiplicadores de *Lagrange* $\alpha_i > 0$ uno por cada restricción. Para las restricciones de la forma $r_i > 0$ cada restricción es multiplicada por α_i (un multiplicador de Lagrange positivo) y se restan de la función objetivo, para así formar la función de Lagrange.

$$L(\vec{w}, b, \alpha_i) = \frac{\|\vec{w}\|^2}{2} - \sum_{i=1}^{\eta} \alpha_i [y_i (\vec{w} \cdot \vec{x}_i + b) - 1]$$

La función de *Lagrange* debe ser minimizada con respecto a las variables primarias \vec{w} y b y maximizada sobre los α_i (en otras palabras encontrar el punto de silla). Para el caso de $y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1$, el correspondiente α_i debe ser cero, debido a que este es el valor de α_i que se maximiza en la función de Lagrange. Los α_i diferentes de cero son para el caso en que $y_i (\vec{w} \cdot \vec{x}_i + b) - 1 = 0$. (Estos son los patrones de entrenamiento que quedan sobre los planos paralelos al OSH).

Este último enunciado corresponde a las condiciones de *Karush-Kuhn-Tucker* (KKT, condiciones complementarias de optimización)

$$\begin{aligned} \alpha_i [y_i (\vec{w} \cdot \vec{x}_i + b) - 1] &= 0 \quad \forall i \\ \frac{\partial}{\partial \vec{w}} L(\vec{w}, b, \alpha) &= 0 \quad \Rightarrow \quad \vec{w} = \sum_{i=1}^{\eta} \alpha_i \vec{x}_i y_i \\ \frac{\partial}{\partial b} L(\vec{w}, b, \alpha) &= 0 \quad \Rightarrow \quad \sum_{i=1}^{\eta} \alpha_i y_i = 0 \end{aligned}$$

La solución de \vec{w} queda en función de un subconjunto de patrones de entrenamiento, aquellos cuyo multiplicador de *Lagrange* es diferente de cero, es decir, el soporte de \vec{w} esta en los patrones de entrenamiento más cercanos al OSH. De aquí el nombre de maquinas de soporte vectorial.

En la práctica se resuelve el problema de *optimización dual de wolfe*, que se obtiene al reemplazar las condiciones KKT en la función de *Lagrange*:

$$\begin{aligned} \max & \left[\sum_{i=1}^{\eta} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j (\vec{x}_i \cdot \vec{x}_j) \right] \\ \text{sujeto a} & \quad \alpha_i \neq 0 \quad \forall i \quad \text{y} \quad \sum_{i=1}^{\eta} \alpha_i y_i = 0 \end{aligned}$$

Este problema puede ser resuelto con métodos de programación cuadrática estándar.

Una vez obtenido el vector $\vec{\alpha}$ se pueden obtener el parámetro \vec{w} a partir de las condiciones KKT:

$$\vec{w} = \sum_{i=1}^{\eta} \alpha_i y_i \vec{x}_i$$

Dado un vector soporte \vec{x}_i el parámetro b puede ser obtenido a partir de las condiciones KKT como:

$$b = y_i - \vec{w} \cdot \vec{x}_i$$

Ahora la ecuación del OSH y la función de decisión puede ser escrita como:

$$g(\vec{x}) = \sum_{i=1}^{\eta} [\alpha_i y_i (\vec{x}_i \cdot \vec{x})] + b$$

$$f(\vec{x}) = \text{sign} \left(\sum_{i=1}^{\eta} [\alpha_i y_i (\vec{x}_i \cdot \vec{x})] + b \right)$$

Se trabaja con la teoría de Lagrange por dos razones fundamentales:

Las restricciones fueron reemplazadas y quedaron en términos de α_i .

En la reformulación del problema, los datos de entrenamiento \vec{x}_i solo aparecen en forma de productos punto entre ellos mismos, como se vera mas adelante esto es muy conveniente.

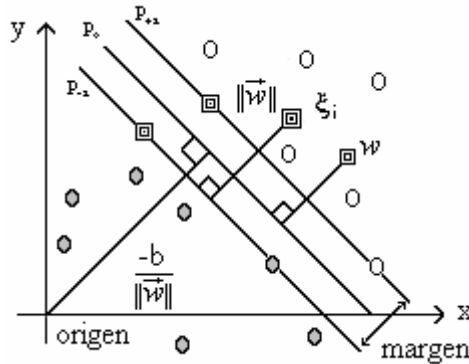
F.6. Tolerancia al ruido de las MSV [\[6\]](#), [\[23\]](#), [\[35\]](#)

Si los datos son no linealmente separables, buscar un OSH carece completamente de sentido. Con la finalidad de posibilitar las violaciones se introducen variables “*slack*” (de relajación).

$$(\xi_1, \dots, \xi_M), \text{ para } \xi_i \geq 0$$

El propósito de las variables ξ_i es permitir puntos erróneamente clasificados. De esta forma si existen datos erróneos, ruido o alto solapamiento de clases en los datos de entrenamiento, se busca el mejor hiper plano clasificador que pueda tolerar el ruido.

Figura F.3. Hiper plano lineal clasificador para el caso no separable



Fuente Autores

De esta manera se formula una nueva restricción:

$$y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i \quad \forall i$$

Luego el clasificador es hallado controlando tanto su capacidad de clasificación

con $\|\vec{w}\|$, como el limite superior del número de errores de entrenamiento $\sum_{i=1}^N \xi_i$. La forma de obtener el hiper plano clasificador óptimo con margen débil es minimizando la función:

$$\min_w \left[\frac{1}{2} (\vec{w} \cdot \vec{w}) + C \sum_{i=1}^N \xi_i \right]$$

$$\text{sujeto a } y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i, \quad \forall i$$

El primer término es minimizado para controlar la capacidad de aprendizaje del mismo modo que en el caso separable; el segundo término permite mantener bajo control el número de clasificaciones erróneas. El parámetro C es elegido por el usuario de manera que un valor grande es equivalente a asignar una alta penalización a los errores. En analogía con el caso separable, la utilización de multiplicadores de Lagrange deriva en el siguiente problema de optimización:

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_{i,j=0}^N \alpha_i \alpha_j y_i y_j x_i \cdot x_j$$

$$\text{sujeto a } \sum_i y_i \alpha_i = 0 \quad \text{y} \quad 0 \geq \alpha_i \geq C, \quad \forall i$$

Se puede notar que la única diferencia con el caso separable es que α_i , tiene una cota superior C.

La solución a este sistema queda:

$$\vec{w} = \sum_{i=1}^N \alpha_i y_i \vec{x}_i$$

$$b = y_i - \vec{w} \cdot \vec{x}_i$$

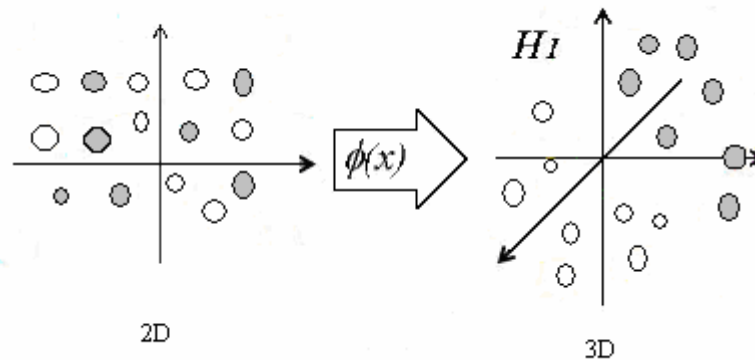
$$g(\vec{x}) = \sum_{i=1}^N [\alpha_i y_i \vec{x}_i \cdot \vec{x}] + b$$

$$f(\vec{x}) = \text{sign} \left(\sum_{i=1}^N [\alpha_i y_i \vec{x}_i \cdot \vec{x}] + b \right)$$

F.7. MSV no lineales – Una generalización [\[23\]](#), [\[34\]](#), [\[35\]](#)

El principio de SVM no lineal consiste en mapear un espacio de entrada \mathbb{R}^n a un espacio de representación mas alta (\mathbb{R}^m , llamado espacio característico \mathcal{H}) a través de una función no lineal ϕ elegida a priori.

$$\phi: \mathbb{R}^n \rightarrow \mathbb{R}^m, \text{ con } n, m \in \mathbb{N}, m \geq n \text{ o } m = \infty$$

Figura F.4. Transformación de los datos originales

Fuente Autores

El objetivo de realizar este mapeo, es que en la nueva dimensión los datos transformados sean separables linealmente o en su caso se minimice el error mediante la separación con un hiperplano, es decir que el número de vectores clasificados incorrectamente sea mínimo.

En este nuevo espacio se puede encontrar la solución mediante el problema de optimización cuadrática como hasta ahora. La solución obtenida es de mayor dimensión, si se desea obtener la solución en el espacio original se realiza el mapeo inverso de los vectores soporte, los cuales determinarán las fronteras que separarán los datos. Los vectores soporte son considerados como el modelo para cada clase.

No es posible determinar a priori, una función Φ que cumpla los objetivos descritos previamente, por lo que el procedimiento para determinarla no es constructivo. Por tanto el tratamiento para este caso está basado en la realización de ensayos con funciones Φ conocidas. Así la elección de la función Φ es de especial importancia para el éxito en la solución del problema de clasificación.

Del procedimiento mostrado para MSV lineales (tanto para el caso linealmente separable o con tolerancia al ruido) puede observarse que las operaciones están expresadas en términos del producto punto entre vectores (como se mencionó anteriormente) y a partir de la función producto punto y de la métrica definida en \mathbb{R}^n se realiza la clasificación de los datos:

$$f(\vec{x}) = \text{sign} \left(\sum_{i=1}^n [\alpha_i y_i \vec{x}_i \cdot \vec{x}] + b \right)$$

Tal norma y tal métrica es empleada también al separar los datos mapeados en el nuevo espacio característico \mathbb{R}^m , El problema análogo planteado en el nuevo espacio característico \mathbb{R}^m es:

$$\min \left\{ f(w) = \frac{1}{2\varepsilon} \|w\|^2, w \in \mathbb{R}^m \right\}$$

$$y_i (w \cdot \phi(x_i) + b) \geq \varepsilon$$

$$i = 1, \dots, \eta$$

$$\varepsilon > 0, b \in \mathfrak{R}$$

De la solución al problema análogo de optimización cuadrática por multiplicadores de *Lagrange* se obtiene:

$$\begin{aligned} \max \sum_i^N \alpha_i - \frac{1}{2} \sum_{i,j}^N \alpha_i \alpha_j y_i y_j \phi(x_i)' \phi(x_j) \\ \text{sujeto a } \sum_{i=1}^N y_i \alpha_i = 0 \quad y \quad \alpha_i \geq 0 \quad \forall i \end{aligned}$$

De lo anterior, el algoritmo de entrenamiento solo depende de los datos a través de los productos puntos en el espacio de representación, esto es funciones de la forma $\phi(x_i)' \phi(x_j)$, se infiere que la función ϕ modifica la norma y métrica del espacio original.

A la función definida por:

$$K(x_i, x_j) = \phi(x_i)' \cdot \phi(x_j)$$

Se le conoce como función núcleo (o función kernel) y su uso es más importante que el de la propia función ϕ , de la cual no se requiere su conocimiento en forma explícita, ya que el entrenamiento solo depende de $K(x_i, x_j)$.

Entonces dada una función kernel, es posible plantear el problema de *optimización dual de wolfe* incluyendo tolerancia de error en la MSV no lineal, para plantear el problema más general, como:

$$\max_{\alpha} \left[\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j}^N \alpha_i \alpha_j y_i y_j k(\vec{x}_i \cdot \vec{x}_j) \right]$$

$$\text{sujeto a } 0 \leq \alpha_i \leq C, \forall i \quad \text{y} \quad \sum_i^{\eta} \alpha_i y_i = 0$$

A partir de métodos de optimización cuadrática, se obtiene la función de decisión:

$$f(\vec{x}) = \text{sign} \left(\sum_{i=1}^{\eta} [\alpha_i y_i k(\vec{x}_i, \vec{x}) + b] \right)$$

La función kernel debe establecer las condiciones para que el mapeo exista, tales condiciones son aseguradas por las condiciones de **Mercer**:

$$K(x, y) = \sum_{i=1}^{\infty} \phi(x)_i \phi(y)_i$$

si y solo si, para algun $g \in L^2(C)$, tal que

$$\int_{C \times C} K(x, y) g(x) g(y) dx dy \geq 0$$

Los primeros Kernels investigados para reconocimiento de patrones (y los más utilizados) son los siguientes:

kernel polinomia : $K(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y} + c)^d$

funcion de base radial (RBF) :

$$K(\vec{x}, \vec{y}) = e^{\left(-\frac{\|\vec{x} - \vec{y}\|^2}{2\sigma^2} \right)}$$

sigmoide

$$K(\vec{x}, \vec{y}) = \tanh(k \times (\vec{x} \cdot \vec{y}) + \Theta)$$

F.8. Multclasificación con SVM [\[36\]](#)

Hasta el momento solo se ha tratado el problema de la bi clasificación, en los cuales las clases solo pueden tomar valores ± 1 ; sin embargo, es muy habitual cuando se trabaja con problemas de clasificación extraídos de la vida real encontrarse con situaciones de multi clasificación $y_i \in \{1, \dots, l\}$, $l > 2$. Para

resolver el problema de multi clasificación con MSV se admiten dos tipos de arquitectura.

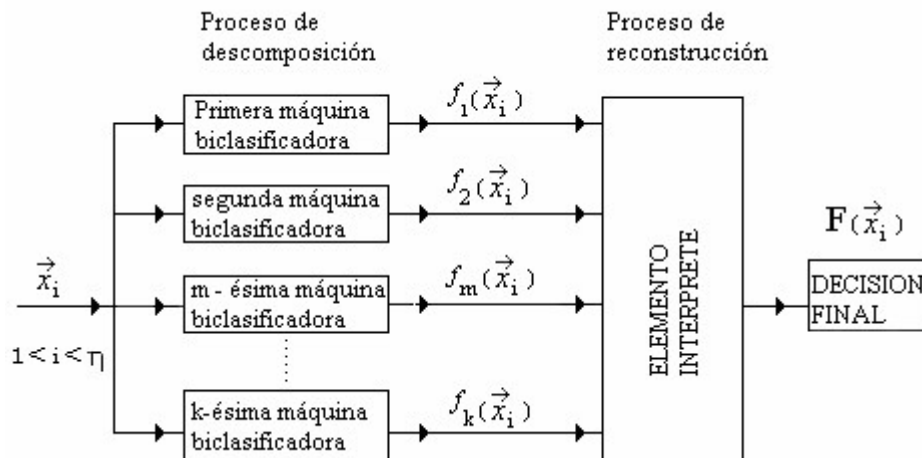
F.8.1. Maquinas multclasificadoras. [36]

Esta maquina construye una función clasificadora global directamente considerando todas las posibles clases a la vez. Este tipo de solución es lenta (alto costo computacional), y además no existe ninguna inclusión de técnicas que mejoren la robustez del sistema, ni algún estudio teórico sobre la cota de error. Esta maquina ya esta concebida para el trabajo con múltiples salidas, como en el caso de los MLP's. Existe un tipo de maquina de aprendizaje, como por ejemplo los árboles de decisión, que trabajan directamente con problemas multi clase aunque para ello necesiten nodos de decisión.

F.8.2. Maquinas bi clasificadoras generalizadas. [6][23][36]

En el esquema de descomposición estándar se construyen k maquinas bi clasificadoras en paralelo, que son entrenadas sobre modificaciones del conjunto de aprendizaje. El proceso de multi clasificación se visualiza más fácilmente mediante el siguiente diagrama:

Figura F.5. Diagrama de multclasificación



Fuente Autores

La cantidad de maquinas bi clasificadoras, el tipo o tipos de maquinas de aprendizaje para realizar la descomposición del espacio de clasificación en biparticiones y el acondicionamiento de las entradas multi clase para realizar

el entrenamiento y las clasificaciones binarias, quedan completamente determinados al elegir un método de descomposición.

El esquema de reconstrucción realiza la fusión e interpretación y/o agrupación de las predicciones parciales $f_m(\vec{x}_i)$, facilitando una respuesta $F(\vec{x}_i)$ al clasificador global. La respuesta final es la asignación de una de las l clases a la correspondiente entrada \vec{x}_i .

Figura F.6. Módulos del proceso de multi clasificación



Fuente Autores

F.9. Arquitecturas de descomposición normalizadas [\[36\]](#)

Para las arquitecturas estándar de descomposición se construyen k maquinas bi clasificadoras en paralelo, que son entrenadas sobre todo el conjunto de aprendizaje o sobre una porción del mismo, creándose una matriz de descomposición que depende de la arquitectura. Los elementos de unas clases son asignados a salidas positivas, los de otras a salidas negativas, y si es el caso, los restantes no son tenidos en consideraron en aquel clasificador en particular (nodo de dicotomía).

F.9.1. Uno contra el resto (1 – v – r). [\[6\]](#) [\[23\]](#) [\[36\]](#)

Conocido como 1 – v – r (*del ingles one-versus-rest*), este esquema se basa en la idea de que si existe un grupo de η datos de entrenamiento donde existen l clases, se pueden tener un grupo de k clasificadores binarios ($k = l$), cada uno entrenado para separar una clase del resto de clases existentes ($l - 1$). La arquitectura fue propuesta originalmente por Vapnik.

Puede observarse que el tiempo de entrenamiento de este método estándar es linealmente proporcional al número de clases (l). Cada maquina bi clasificadora es entrenada sobre todo el conjunto de aprendizaje, con el costo computacional

que esto implica pero con la ventaja de contar con toda la información en la fase de entrenamiento.

La descomposición se realiza de la siguiente manera:

Existe un grupo de datos (η_j) que pertenecen a la j -ésima clase $j \in \{1, \dots, l\}$, a los cuales se les dará una etiqueta positiva ($t_j = +1$) y al resto ($\eta_r = \eta - \eta_j$) se les dará una etiqueta negativa ($t_r = -1$) para el entrenamiento de la m -ésima MSV. Así se crea una matriz de descomposición (D_{1-v-r}) de k filas y l columnas:

$$D_{r,j} = \begin{cases} +1 & \text{si } \eta_i \in \eta_j \\ -1 & \text{si } \eta_i \in \eta_r \end{cases}$$

Por ejemplo, para una maquina de clasificación multi clase $(1-v-r)$ con $l = 5$, se obtiene $k = 5$. Entonces la correspondiente matriz de descomposición es la siguiente:

$$D_{1-v-r} = \begin{pmatrix} +1 & -1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 & -1 \\ -1 & -1 & +1 & -1 & -1 \\ -1 & -1 & -1 & +1 & -1 \\ -1 & -1 & -1 & -1 & +1 \end{pmatrix}$$

F.9.2. Uno contra uno (o-v-o) [\[6\]](#) [\[23\]](#) [\[36\]](#)

Conocido como $1-v-1$ (*del ingles one-versus-one*), se realiza implementando $k = \frac{l(l-1)}{2}$ clasificadores binarios en paralelo y entrenados cada uno con solo 2 de las l clases existentes en el grupo de η datos de entrenamiento.

A pesar que algunos autores defienden que el esquema asociado al $1-v-1$ es, en general, preferible al esquema $1-v-r$ como así muestran sus estudios empíricos, existen tres inconvenientes asociados al método de descomposición $1-v-1$, que ponen en entredicho esta afirmación [\[36\]](#):

1. La MSV $1-v-1$ en cada nodo de dicotomía solo es entrenada sobre un conjunto modificado del total de datos de aprendizaje, por lo que la fisonomía que adopte la función decisión en cada maquina bi

clasificadora, no tendrá en consideración la existencia de otros patrones de entrenamiento pertenecientes a clases distintas.

2. Si la respuesta final $F(x_i)$ dada por la maquina de clasificación multi clase global es por ejemplo $b \in \{1, \dots, l\}$, todas las predicciones dadas por los (p, j) ésimos nodos de dicotomía que $\in \{1, \dots, l\}$ con $p, j \neq b$, no deberían ser tomadas en cuenta, pues la respuesta a patrones de la clase b no pertenece a su tarea de clasificación.
3. El tamaño de la maquina de clasificación multi clase asociada a nodos de dicotomía $1-v-1$ puede crecer súper linealmente con l (numero de clases).

Mientras que el tercer ítem es una cuestión de tiempo, que no debe afectar las bondades de la respuesta final, los dos primeros ítems están relacionados al hecho que en cada nodo de dicotomía el conjunto de entrenamiento es una parte de todo el conjunto de aprendizaje. Sería oportuno construir una maquina que capture las propiedades positivas del esquema $1-v-1$ que lo hacen empíricamente preferible sobre $1-v-r$ y que por otro lado cada nodo de dicotomía sea entrenado sobre todo el conjunto de aprendizaje, para que sus respuestas sean consecuentes sea cual sea la entrada al nodo de dicotomía. Una forma de unificar ambos esquemas de descomposición dentro de un marco común más general es posible si se especifican mediante una matriz de descomposición D .

La validez de un esquema de descomposición, se expresa mediante la restricción que para cualquier par de columnas de D , debe existir al menos una fila para la cual los coeficientes en las dos columnas son +1 y -1.

De esta forma, en el esquema $1-v-1$ se le otorga etiqueta positiva ($t_j = +1$) a los datos (η_j) que pertenecen al subgrupo de datos de la clase j ($j \in \{1, \dots, l\}$) y etiqueta negativa ($t_i = -1$) a los datos (η_p) que pertenecen al subgrupo de datos de la clase p ($p \in \{1, \dots, l\}$ e $p \neq j$). Los demás datos ($\eta_r = \eta - \eta_p - \eta_j$) no se utilizan en el entrenamiento de la m -ésima MSV, por lo tanto son etiquetados con cero ($t_r = 0$), creándose la matriz de descomposición (D_{1-v-1}).

$$D_{p,j} = \begin{Bmatrix} +1 & \text{si } \eta_i \in \eta_j \\ -1 & \text{si } \eta_i \in \eta_p \\ 0 & \text{si } \eta_i \in \eta_r \end{Bmatrix}$$

Por ejemplo, para una maquina de clasificación multi clase $(1-v-1)$ con $l=5$, se obtiene $k=10$. Entonces la correspondiente matriz de descomposición es la siguiente:

$$D_{1-v-1} = \begin{pmatrix} +1 & -1 & 0 & 0 & 0 \\ +1 & 0 & -1 & 0 & 0 \\ +1 & 0 & 0 & -1 & 0 \\ +1 & 0 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & +1 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & -1 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & +1 & 0 & -1 \\ 0 & 0 & 0 & +1 & -1 \end{pmatrix}$$

F.10. Métodos de reconstrucción [\[23\]](#)

Durante el proceso de descomposición estándar, se construyen k nodos de dicotomía, que son entrenados sobre modificaciones del conjunto de aprendizaje; de forma que de acuerdo a la matriz de descomposición, los elementos de unas clases son asignados a salidas positivas, los de otra clase a salidas negativas, y los patrones restantes no son tenidos en cuenta en aquel clasificador en particular.

Cada maquina bi clasificadora entrenada emite una respuesta en forma numérica $Z_m = g_m(\vec{x})$ a una entrada \vec{x}_i .

La información más importante en esta respuesta, en principio se encuentra en el signo $s_m = f_m(\vec{x}_i) = \text{sign}[g_m(\vec{x}_i)]$ que adopta la función de decisión. Sin embargo en la determinación de la respuesta final facilitada por el método de reconstrucción de la maquina de aprendizaje multi clase, han de ser tomados en consideración los siguientes elementos:

1. Las predicciones numéricas parciales en los nodos de dicotomía $Z_m = g_m(\vec{x}_i)$.
2. El signo de las predicciones numéricas $s_m = f_m(\vec{x}_i) = \text{sign}[g_m(\vec{x}_i)]$.
3. Un elemento interprete de las predicciones numéricas y binarias $\theta(Z_m, S_m)$ con el fin de asignar o no, una o varas clases como posible respuesta de clasificación a una entrada \vec{x}_i .
4. Un elemento de combinación de las predicciones $\Psi(\theta(Z_1, S_1), \dots, \theta(Z_m, S_m), \dots, \theta(Z_k, S_k))$ que tenga o pueda tener en consideración las predicciones numéricas, sus signos y/o la clase o clases asignadas.

El esquema de votación es la forma de reconstrucción más habitual. En cada maquina bi clasificadora se asigna un voto (a favor, en contra o nulo) a las clases implicadas en el respectivo nodo de dicotomía, teniendo en consideración solo el signo de las predicciones parciales. Estos signos son interpretados en función del esquema de descomposición implementado.

Tras la interpretación de las predicciones parciales, el elemento de combinación Ψ realiza un recuento del número de clases votadas. A continuación se definen algunas de las posibilidades del esquema de reconstrucción por votación, de acuerdo al método de descomposición:

1. Votación por unanimidad: se determina como respuesta aquella única clase que haya obtenido todos los votos posibles en las predicciones.

En el caso $1-v-r$ significa que solo debe existir un voto valido, siendo los demás nulos, por lo tanto se asigna la clase que tenga un voto. Para el esquema $1-v-1$ significa que se asigna la clase con número de votos igual a $l-1$, que es el numero de nodos de dicotomía en los que la cada clase esta implicada. Las clases con menor número de votos son irrelevantes.

2. Votación por mayoría absoluta. Se determina como respuesta final aquella única clase que haya obtenido más de la mitad de los votos posibles.

En la metodología $1-v-r$, solo es posible un voto por clase, ya que cada clase esta implicada con etiqueta positiva en solo un nodo de dicotomía. Por lo que este tipo de votación es equivalente al voto por unanimidad.

Para el caso $1-v-1$ se elige la clase que tenga mayor número de votos que el resto y que además el número de votos sea mayor a $\frac{l-1}{2}$.

3. Votación por mayoría simple. Se determina como respuesta final aquella única clase que haya obtenido más votos que el resto de clases.

De nuevo en la metodología $1-v-r$ este esquema de votación es equivalente a los anteriores. En el caso $1-v-1$ se asigna la clase que tenga mayor cantidad de votos.

Si las maquinas bi clasificadoras son entrenadas todas sin error sobre el conjunto de entrenamiento, para el esquema $1-v-r$ la votación por unanimidad debería bastar como método de reconstrucción para obtener una respuesta adecuada, sin embargo en numerosas ocasiones la respuesta final será un empate, o un caso de ambigüedad extrema con todas las predicciones de signo negativo, por lo que la entrada se habrá quedado sin etiquetar.

La combinación de $1-v-1$ clasificadores binarios mediante un esquema por unanimidad, elimina la posibilidad de empate, pero en numerosas ocasiones deja entradas sin etiquetar debido a la necesidad de que todas las maquinas bi clasificadoras realicen una predicción correcta.

Una metodología para evitar la falta de etiqueta para una entrada de prueba, es evaluar en caso de empate, ambigüedades o falta de unanimidad el valor numérico entregado por cada bi clasificador y asignar a la entrada aquella clase para la que se obtuvo un mayor valor numérico.

Esta metodología denominada “el ganador se queda con todo”, es útil asumiendo que las respuestas numéricas de cada bi clasificador son comparables. Sin embargo en la mayoría de las situaciones ese no es el caso, por lo que el esquema de votación debe ser independiente del rango de salidas de cada bi clasificador.

Los intentos de mejora en la respuesta se centran en variaciones sobre el método de reconstrucción, en interpretaciones probabilísticas de las salidas numéricas de los bi clasificadores y en modificaciones en la agrupación de clases para realizar la dicotomía, denominada codificación.

F.10.1. Clasificación por parejas [\[6\]](#) [\[23\]](#) [\[36\]](#)

En el método de reconstrucción denominado “por parejas”, se interpretan las salidas de las dicotomías por medio de una especie de grado de pertenencia a la clase.

Si se renombra la salida $f_m^+(x_i)$ correspondiente a cada uno de los l bi clasificadores en un esquema $1-v-1$, como $[f_m^+(x_i)]^{p,j}$, para indicar que la m -ésima dicotomía separa las clases p y j , entonces es posible definir el elemento de combinación como la suma de votos recibidos a favor y en contra por la clase b en todos los nodos de dicotomía donde se encuentra implicada.

Se comprueba empíricamente que este método de reconstrucción produce mejores fronteras de decisión en el sentido de obtener espacios finales de ambigüedad mas reducidos, aunque el estudio se restringe a fronteras lineales con problemas de solo 3 clases.

Figura F.7. Esquema de votación por parejas

RESULTADOS PARA UNA ENTRADA EN PARTICULAR METODO DE DESCOMPOSICION O-V-O CON 4 CLASES						
	Máquina 1	Máquina 2	Máquina 3	Máquina 4	Máquina 5	Máquina 6
signo de la funcion decisión	Positivo	Negativo	Positivo	Positivo	Negativo	Negativo
voto clase 1	favor (+1)	contra (-1)	favor (+1)	nulo (0)	nulo (0)	nulo (0)
voto clase 2	contra (-1)	nulo (0)	nulo (0)	favor (+1)	contra (-1)	nulo (0)
voto clase 3	nulo (0)	favor (+1)	nulo (0)	contra (-1)	nulo (0)	contra (-1)
voto clase 4	nulo (0)	nulo (0)	contra (-1)	nulo (0)	favor (+1)	favor (+1)

total clase 1	1
total clase 2	-1
total clase 3	-1
total clase 4	1

Fuente Autores

F.10.2. Salidas numericas probabilisticas [\[6\]](#) [\[23\]](#) [\[36\]](#)

En la teoría de aprendizaje estadístico, una MSV es capaz de convertir la salida $f_m^+(x_i)$, en una salida probabilística. De esta manera la MSV calcula la

probabilidad de pertenencia de una entrada \vec{x}_i a una clase (i,j) . En ese intento se propone el uso de la función logística $p[f(\vec{x}_i)]$ y se asigna como etiqueta aquella clase que tenga mayor probabilidad:

$$p_+[f(\vec{x}_i)] = \frac{1}{1 + e^{-f(\vec{x}_i)}}$$

$$p_-[f(\vec{x}_i)] = 1 - p_+[f(\vec{x}_i)]$$

Se debe garantizar que cada nodo de dicotomía sea capaz de calcular una probabilidad, por ello se deben realizar modificaciones sobre las salidas numéricas $f(\vec{x}_i)$, normalizando por su valor máximo.

F.10.3. Función desición final

Para evitar cualquier tipo de ambigüedad (empates, no asignación de etiquetas, etc.), la función de decisión final $F(\vec{x}_i)$, dependerá de la máxima probabilidad de pertenencia a la clase utilizando la función logística y de la máxima cantidad de votos en cada nodo de dicotomía utilizando un esquema de votación por parejas.

F.11. Validacion Cruzada en la MSV [\[6\]](#) [\[23\]](#)

En el proceso de entrenamiento se debe considerar tanto el error de aprendizaje como el error de generalización para no incurrir en sobre aprendizajes o sobre ajustes (overfitting). Para el caso de las MSV este sobre ajuste se corrige gracias a la misma naturaleza de la técnica, en la cual no existen mínimos locales, por tratarse de un problema de optimización cuadrático.

Para minimizar la posibilidad de que exista una tendencia a memorizar los ejemplos, usualmente, de todo el conjunto de entrenamiento se emplea aproximadamente un 80% de los patrones para entrenar, reservándose un 20% como conjunto de test.

El proceso de validación cruzada es utilizado en MSV para buscar el punto óptimo entre el error de entrenamiento y el error de test. La situación que se

evidencia en problemas prácticos se observa en la figura, en la cual se aprecia que a medida que el error de entrenamiento disminuye con cada iteración, el error de test oscila. De esta forma, con la validación cruzada se optimiza el comportamiento del error en test, al realizar de manera conjunta tanto el entrenamiento de la MSV como su validación con el grupo de prueba hasta encontrar el punto óptimo con pocas iteraciones.

En suma la validación cruzada es una técnica que se utiliza para tratar problemas iterativos repartiendo la muestra en dos sistemas de datos, uno se utiliza para construir el modelo, y el otro para probarlo. Esta técnica incorporada en el entrenamiento de la MSV es usada para seleccionar los mejores parámetros del Kernel y un valor óptimo de penalización C para un grupo específico de datos, de tal forma que el error en test se encuentre dentro de un rango esperado.

Anexo G

Descripción interna de Interfaz para investigación de patrones EMG

G.1. Formato de los datos

Las restricciones en el formato que deben cumplir los datos de entrada (señales EMG) para ser admitidos por la interfaz, bien sea para fase de entrenamiento o para fase de prueba se detallan en la [tabla G.1](#).

Tabla G.1: Formato de señales EMG

Nombre [tipo de dato]	Descripción
Señal CH Anterior [.txt ó .MAT] (Botón de IG.*)	Datos sin procesar provenientes del canal ubicado en la región anterior del antebrazo
Señal CH Posterior [.txt ó .MAT] (Botón de IG.†)	Datos sin procesar provenientes del canal ubicado en la región posterior del antebrazo

Fuente autores

* IG: Interfaz Gráfica

† Ver figura XX

G.2. Funciones

Para conocer las funciones que componen la interfaz teclear en el *work* de MATLAB:

```
>> help InterfazEJECUCION
```

INTERFAZ PARA RECONOCIMIENTO DE PATRONES EMG

Funciones empleadas en fase de entrenamiento

procesW - Genera modelo Wavelet
PCAmodelo - Genera modelo PCA

Funciones empleadas en fase de prueba

procesW - Realiza proceso de análisis Wavelet
procesPCA - Realiza proceso de análisis PCA
clasificarMSV - Realiza proceso de clasificación con MSV

G.2.1. procesW

Realiza el proceso de descomposición wavelet a la señal EMG para que sea representada en el espacio de la DWT hasta un nivel indicado en el modelo de entrada de esta función*.

Sinopsis: $M_Carac = \text{procesW}(X, \text{modeloW})$

M_Carac corresponde a la matriz de características resultante de la matriz de entrada X según los cálculos especificados en modeloW .

G.2.2. PCAmodelo

Realiza el proceso PCA para generar un modelo correspondiente†.

Sinopsis: $[\text{MatrizT}, \text{cargas}, \text{variabilidadC}, \text{media}, \text{desviacion}] = \dots$

* Este modelo es aquel que establece los parámetros de los módulos 2 y 3 del [capítulo 2](#).

† Este modelo esta compuesto por los vectores que calculan directamente las componentes en el espacio de PCA.

... PCAmodelo(DATOS,modeloMi)

El modelo generado por esta función se almacena en estructuras para su posterior aplicación.

G.2.3. procesPCA

Esta función realiza el proceso PCA de acuerdo a un modelo PCA previo* que lo establece.

Sinopsis: $M = \text{procesPCA}(X, \text{modeloW}, \text{modeloPCA})$

La matriz M es el resultado de realizar el proceso PCA para la clasificación de los datos X , partiendo de un *modeloW* y un *modeloPCA*, estructuras donde se han almacenado que tipos de análisis hacer.

G.2.4. clasificarMSV

Esta función es una compilación de las funciones que hacen posible la clasificación por medio del algoritmo de Maquinas de Soporte Vectorial, implementado en los trabajos realizados por [\[6\]](#) y [\[23\]](#).

Sinopsis: $y = \text{clasificarMSV}(X, \text{modelo})$

En el vector y se almacenan los resultados de la función, a partir de la matriz de entrada X y la estructura *modelo*.

G.3. Formato en estructuras internas

La interfaz trabaja con arreglos de estructuras que permiten el almacenamiento organizado de diversos tipos de información en un solo archivo. Esto hace posible acceder a información de diferente naturaleza en forma práctica para realizar cálculos conforme a los requerimientos de cada caso.

De acuerdo con el proceso que realiza la interfaz, las estructuras cumplen un formato específico en cada caso.

* El modelo PCA previo es aquel obtenido de la fase de entrenamiento

G.3.1. Entrenamiento

En esta opción encontrada en la interfaz como “*Modelos a evaluar*”, ver [figura 4.1](#), se debe ingresar el archivo cluster de modelos y a continuación dar clic en “*Procesar*”.

Este archivo cluster de modelos debe cumplir el formato mostrado en la [tabla G.2](#).

Cluster_wavelet es entonces una estructura formada por la cantidad de modelos trabajados y cada uno esta conformado por la estructura mostrada en la [tabla G.3](#).

Tabla G.2: Estructura de configuración en modelamiento

Nombre [tipo de dato]	Descripción
Cluster_wavelet* [struct]	
.cantidad [int]	Parámetro que indica la cantidad de modelos que se procesaran.
.modelo† [struct]	Parámetro que corresponde al nombre que lleva cada modelo a procesar.

Fuente autores

El parámetro *.Características* de la [tabla G.3](#) es también una estructura. Cada característica evaluada sobre la transformada wavelet tiene asociado un nombre. Por ejemplo, si se evalúan las cuatro características disponibles, el parámetro *.Características* se vería así:

```
modelo5.Características  
  
ans =  
  
C1: 'rms'  
C2: 'varianza'  
C3: 'mean'  
C4: 'desviacion'
```

* El nombre de este cluster es opcional, sin embargo, para identificar con que wavelet se trabaja hemos colocado *cluster* seguido del nombre de la *wavelet* en todos los casos, ejemplo: cluster_db4

† Este nombre debe guardarse siempre igual, por ejemplo si cantidad es 2: modelo1 modelo2.

Tabla G.3: Estructura de configuración de cada modeloW de entrada

Nombre [tipo de dato]	Descripción
.modelo* * [struct]	
.Wavelet [string]	Parámetro que identifica con que wavelet se ha de realizar el análisis.
'db*'	Daubechies
'sym*'	Symlet
'coif*'	Coiflet
.NivelR [int]	Parámetro que indica hasta que nivel de descomposición tendrá lugar en la transformada Wavelet.
.enventanado [string]	Parámetro que indica cual tipo de enventanado realizar sobre la señal de entrada.
'disjunto'	Segmenta la señal en forma disjunta.
'continuo'	Segmenta la señal forma continua
.N_caracteristicas [int]	Parámetro que informa cuantas características serán tenidas en cuenta para la conformación del vector patrón.
.Caracteristicas [struct]	
.C* [string]	Parámetro que establece la característica calculada.
.tipoPCA [string]	Parámetro que indica cual método de PCA será realizado en el modelo.
'corre'	PCA con correlación
'covar'	PCA con covarianza
.ComponentesPCA [string]	Parámetro que indica cual criterio será tenido en cuenta para la selección del número de características del vector patrón reducido con PCA.
'kaiser'	Kaiser
'estandar'	Estándar (criterio propuesto)

Fuente autores

* El * indica que tiene asociado un número, ejemplo: modelo₅

G.3.2. Prueba

En esta opción se encuentran los botones MODULO 3 y MODULO 4, ver [figura 4.1](#), en donde deben ser ingresados los modelos correspondientes a la configuración descrita en las [tablas G.4](#), y [G.5](#) respectivamente. En MODULO 2 se ingresa el archivo **modeloW** en esta misma sección de la interfaz, sin embargo su configuración es igual a la descrita en la [tabla G.3](#).

Tabla G.4: Estructura de configuración del modelo MSV

Nombre [tipo de dato]	Descripción
modelo [struct]	
.Kernel [string]	Tipo de kernel
.Kernel_par [2 x 1]	Parámetro del kernel
.C [1 x 1]	Parámetro de penalización
.clases [1 x 1]	Número de clases K
.exact_entr [1 x 1]	Exactitud de entrenamiento
.exact_prueba [1 x 1]	Exactitud de prueba, este campo aparece por la Validación cruzada
.detalles [struct]	
.metodo [string]	Método usado en la optimización
.descom [string]	Tipo de descomposición
.eps [1 x 1]	Error de parada
.tol [1 x 1]	Tolerancia de las condiciones KKT
.num_SV [1 x 1]	Número de vectores de soporte
.SV_y [1 x K]	Numero de vectores de soporte por clase
.SVM [struct] [1xL]	Donde L es el número de maquinas biclasificadoras que se ejecutan para multclasificar K clases
	L = size(D,1). D matriz de descomposición
	L = K*(K+1)/2 si se usa 'ovo'
	L = K si se usa 'ovr'
	L = K*(K+1) si se usa 'ECOC_2C'
.SV [NixP]	Vectores de soporte. Ni es el número de VS.
.Alpha [Nix1]	Multiplicadores de lagrange
.b [1x1]	Constante de la función de decisión
exact_entr [1x1]	Exactitud de entrenamiento de cada maquina
	Biclasificadora
.clase1 [1xa1]	Clases con etiqueta 1 implicadas en cada maquina
	Biclasificadora
.clase2 [1xa2]	Clases con etiqueta 2 implicadas en cada maquina
	Biclasificadora

Fuente [\[6\]](#)

Tabla G.5: Estructura de configuración de modeloPCA

Nombre [tipo de dato]	Descripción
modeloPCA [struct]	
.cp [N x P]	Componentes principales
.media [1 x N]	Valor de las medias
.desviacion [1 x N]	Valor de las desviaciones

Fuente autores

G.4. Instalación de InterfazEJECUCION

Para instalar *InterfazEJECUCION*, es necesario tener instalado MATLAB 7 o superior. Siga los siguientes pasos para la instalación:

1. Ejecutar MATLAB 7 o superior.
2. Copiar la carpeta *Interfaz_EMG* a un directorio específico
3. Adicionar al *path* de MATLAB la carpeta *Interfaz_EMG* y subcarpetas

Después de los pasos anteriores se procede a correr el archivo llamado *InterfazEJECUCION.m*.

Anexo H

Tablas de resultados en fase de entrenamiento de prueba preliminar

H.1. Introducción

En el presente anexo se adjuntan las tablas que muestran los valores de exactitud en cada una de las maquinas entrenadas para el paciente 1 junto con la cantidad de vectores soporte que necesitan.

Esta exactitud se mide entre 0.0 y 1.0, en donde 0.0 indica que no se obtuvo ningún acierto en la clasificación de los movimientos durante el entrenamiento y 1.0 indica una clasificación del 100% de los movimientos presentados para su entrenamiento.

H.2. Tablas de resultados

En las [tablas H.1](#), a [H.12](#) se muestran los resultados de la fase de entrenamiento de las MSVs usando la Wavelet Daubechies de orden 4 y 5 con cálculos de los estadísticos hasta el 4to o 5to nivel de resolución.

En las [tablas H.13](#), a [H.24](#) se muestran los resultados de la fase de entrenamiento de las MSVs usando la Wavelet Coiflet de orden 3 y 4 con cálculos de los estadísticos hasta el 4to o 5to nivel de resolución.

En las [tablas H.25](#), a [H.36](#) se muestran los resultados de la fase de entrenamiento de las MSVs usando la Wavelet Symlet de orden 4 y 6 con cálculos de los estadísticos hasta el 4to o 5to nivel de resolución.

Tabla H.1. Resultados de Wavelet Daubechies 4 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	27	1	30
Modelo 2	1	37	1	34
Modelo 3	0.73	91	0.73	81
Modelo 4	1	46	1	48
Modelo 5	1	50	1	56
Modelo 6	1	64	1	63
Modelo 7	1	48	1	47
Modelo 8	1	53	1	55

Fuente Autores

Tabla H.2. Resultados de Wavelet Daubechies 4 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		3 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	1	31
Modelo 2	1	35	1	36
Modelo 3	-	-	0.74	88
Modelo 4	1	43	1	46
Modelo 5	1	74	1	61
Modelo 6	1	71	1	90
Modelo 7	1	44	1	44
Modelo 8	1	52	1	53

Fuente Autores

Tabla H.3. Resultados de Wavelet Daubechies 4 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	1	32
Modelo 2	1	37	1	38
Modelo 3	0.71	129	0.72	95
Modelo 4	0.94	75	0.98	66
Modelo 5	1	52	1	55
Modelo 6	1	62	1	65
Modelo 7	0.99	63	0.99	61
Modelo 8	0.99	64	0.99	62

Fuente Autores

Tabla H.4. Resultados de Wavelet Daubechies 4 con resolución hasta el quinto nivel usando *kernel* Polinomico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	29	1	27
Modelo 2	1	42	1	41
Modelo 3	1	49	1	51
Modelo 4	1	54	1	58
Modelo 5	1	51	1	54
Modelo 6	1	64	1	63
Modelo 7	1	55	1	53
Modelo 8	1	59	1	59

Fuente Autores

Tabla H.5. Resultados de Wavelet Daubechies 4 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	61	1	31
Modelo 2	1	43	1	46
Modelo 3	1	46	1	45
Modelo 4	1	49	1	46
Modelo 5	1	76	1	83
Modelo 6	1	89	1	86
Modelo 7	1	53	1	52
Modelo 8	1	60	1	57

Fuente Autores

Tabla H.6. Resultados de Wavelet Daubechies 4 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	35	1	28
Modelo 2	1	42	1	42
Modelo 3	0.98	63	0.98	63
Modelo 4	0.99	66	0.99	64
Modelo 5	1	51	0.99	62
Modelo 6	1	61	1	63
Modelo 7	0.98	65	0.97	64
Modelo 8	1	66	0.99	66

Fuente Autores

Tabla H.7. Resultados de Wavelet Daubechies 5 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	44	1	74
Modelo 2	1	37	1	37
Modelo 3	0.84	61	0.83	77
Modelo 4	1	49	1	48
Modelo 5	1	45	1	48
Modelo 6	0.98	88	1	62
Modelo 7	0.99	46	0.99	42
Modelo 8	0.99	42	0.99	44

Fuente Autores

Tabla H.8. Resultados de Wavelet Daubechies 5 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	81	1	80
Modelo 2	1	39	1	38
Modelo 3	0.83	77	0.84	78
Modelo 4	1	39	1	39
Modelo 5	1	53	1	77
Modelo 6	1	84	1	70
Modelo 7	1	41	1	41
Modelo 8	1	43	1	44

Fuente Autores

Tabla H.9. Resultados de Wavelet Daubechies 5 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	86	0.98	29
Modelo 2	1	36	1	38
Modelo 3	0.84	89	0.84	82
Modelo 4	0.96	62	0.96	63
Modelo 5	0.99	48	1	46
Modelo 6	0.99	61	1	61
Modelo 7	0.99	59	0.99	58
Modelo 8	0.98	60	0.97	62

Fuente Autores

Tabla H.10. Resultados de Wavelet Daubechies 5 con resolución hasta el quinto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	30	1	31
Modelo 2	1	69	1	41
Modelo 3	0.98	45	0.99	43
Modelo 4	1	51	1	49
Modelo 5	1	57	1	56
Modelo 6	1	57	1	59
Modelo 7	1	48	1	46
Modelo 8	0.98	50	0.99	53

Fuente Autores

Tabla H.11. Resultados de Wavelet Daubechies 5 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	33	-	-
Modelo 2	1	71	1	73
Modelo 3	0.99	40	0.99	39
Modelo 4	1	47	1	46
Modelo 5	1	71	1	56
Modelo 6	1	80	1	85
Modelo 7	1	54	1	51
Modelo 8	1	53	1	54

Fuente Autores

Tabla H.12. Resultados de Wavelet Daubechies 5 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	32	0.99	91
Modelo 2	1	41	1	41
Modelo 3	0.94	60	0.94	69
Modelo 4	0.95	63	0.96	63
Modelo 5	0.98	57	1	53
Modelo 6	1	60	1	63
Modelo 7	0.97	60	0.97	55
Modelo 8	0.97	65	0.98	58

Fuente Autores

Tabla H.13. Resultados de Wavelet Coiflet 3 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.98	30	0.94	30
Modelo 2	1	39	1	36
Modelo 3	0.98	52	0.96	51
Modelo 4	1	50	1	52
Modelo 5	1	51	0.97	68
Modelo 6	1	59	1	65
Modelo 7	1	51	1	49
Modelo 8	1	57	0.99	53

Fuente Autores

Tabla H.14. Resultados de Wavelet Coiflet 3 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		3 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	35	0.99	31
Modelo 2	1	35	1	50
Modelo 3	0.98	39	0.99	38
Modelo 4	1	49	1	47
Modelo 5	1	81	1	55
Modelo 6	1	73	1	65
Modelo 7	1	46	1	47
Modelo 8	1	55	1	53

Fuente Autores

Tabla H.15. Resultados de Wavelet Coiflet 3 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	38	0.99	32
Modelo 2	1	37	1	34
Modelo 3	0.93	71	0.93	71
Modelo 4	0.97	66	0.98	68
Modelo 5	0.97	57	1	53
Modelo 6	1	64	1	61
Modelo 7	0.96	63	0.96	64
Modelo 8	0.98	66	0.97	65

Fuente Autores

Tabla H.16. Resultados de Wavelet Coiflet 3 con resolución hasta el quinto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.98	57	0.98	60
Modelo 2	0.99	45	0.99	74
Modelo 3	0.66	98	0.65	92
Modelo 4	1	49	1	48
Modelo 5	0.98	55	0.97	52
Modelo 6	1	66	1	63
Modelo 7	1	53	1	51
Modelo 8	1	48	1	49

Fuente Autores

Tabla H.17. Resultados de Wavelet Coiflet 3 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.98	60	0.98	33
Modelo 2	0.99	65	1	42
Modelo 3	0.66	94	0.68	93
Modelo 4	1	51	1	43
Modelo 5	0.98	86	0.98	64
Modelo 6	0.99	70	0.99	69
Modelo 7	1	57	1	50
Modelo 8	1	52	1	51

Fuente Autores

Tabla H.18. Resultados de Wavelet Coiflet 3 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.98	40	0.98	31
Modelo 2	0.98	92	0.99	81
Modelo 3	0.65	96	0.64	95
Modelo 4	0.97	65	0.97	63
Modelo 5	0.97	55	0.98	52
Modelo 6	1	62	1	65
Modelo 7	0.98	63	0.98	63
Modelo 8	0.98	63	0.97	62

Fuente Autores

Tabla H.19. Resultados de Wavelet Coiflet 4 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	28	0.99	27
Modelo 2	1	28	0.99	49
Modelo 3	0.97	46	0.98	47
Modelo 4	0.99	56	1	53
Modelo 5	0.99	60	0.98	55
Modelo 6	0.99	65	0.99	72
Modelo 7	0.98	48	0.97	55
Modelo 8	0.99	52	0.99	52

Fuente Autores

Tabla H.20. Resultados de Wavelet Coiflet 4 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	0.99	31
Modelo 2	1	32	1	40
Modelo 3	0.98	40	0.98	38
Modelo 4	0.99	46	0.99	40
Modelo 5	1	60	0.98	63
Modelo 6	0.99	66	1	85
Modelo 7	0.98	37	0.96	58
Modelo 8	0.99	46	1	48

Fuente Autores

Tabla H.21. Resultados de Wavelet Coiflet 4 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	27	0.99	28
Modelo 2	1	31	1	34
Modelo 3	0.93	63	0.92	71
Modelo 4	0.95	66	0.95	67
Modelo 5	0.94	70	0.97	60
Modelo 6	0.99	61	0.98	61
Modelo 7	0.95	63	0.93	73
Modelo 8	0.96	60	0.96	64

Fuente Autores

Tabla H.22. Resultados de Wavelet Coiflet 4 con resolución hasta el quinto nivel usando *kernel* Polinomico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	28	1	28
Modelo 2	1	40	1	49
Modelo 3	0.63	116	0.61	94
Modelo 4	0.98	50	0.98	56
Modelo 5	1	57	0.99	61
Modelo 6	0.99	64	0.96	88
Modelo 7	0.99	58	0.99	55
Modelo 8	1	57	0.99	56

Fuente Autores

Tabla H.23. Resultados de Wavelet Coiflet 4 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	34	1	34
Modelo 2	1	42	1	68
Modelo 3	0.63	97	0.60	116
Modelo 4	0.99	44	0.99	50
Modelo 5	1	84	1	60
Modelo 6	0.99	71	1	68
Modelo 7	0.99	56	0.99	62
Modelo 8	1	59	0.99	59

Fuente Autores

Tabla H.24. Resultados de Wavelet Coiflet 4 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	0.99	48	0.99	31
Modelo 2	0.99	43	0.99	43
Modelo 3	0.62	105	0.61	96
Modelo 4	0.93	73	0.93	69
Modelo 5	0.96	5	0.99	57
Modelo 6	0.99	62	0.99	65
Modelo 7	0.96	69	0.96	66
Modelo 8	0.96	67	0.96	69

Fuente Autores

Tabla H.25. Resultados de Wavelet Symlet 4 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	29	1	30
Modelo 2	1	36	1	36
Modelo 3	0.63	113	0.69	91
Modelo 4	0.99	50	1	50
Modelo 5	1	53	1	51
Modelo 6	1	64	1	61
Modelo 7	1	52	1	52
Modelo 8	1	51	1	52

Fuente Autores

Tabla H.26. Resultados de Wavelet Symlet 4 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	29	1	30
Modelo 2	1	33	1	34
Modelo 3	0.69	118	0.70	91
Modelo 4	1	42	1	41
Modelo 5	1	57	1	75
Modelo 6	1	76	1	92
Modelo 7	1	52	1	51
Modelo 8	1	57	1	55

Fuente Autores

Tabla H.27. Resultados de Wavelet Symlet 4 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	1	30
Modelo 2	1	34	1	36
Modelo 3	0.66	97	0.70	129
Modelo 4	0.97	67	0.97	73
Modelo 5	1	52	1	57
Modelo 6	1	65	1	60
Modelo 7	0.96	62	0.96	62
Modelo 8	0.96	65	0.98	65

Fuente Autores

Tabla H.28. Resultados de Wavelet Symlet 4 con resolución hasta el quinto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	31	1	83
Modelo 2	1	43	1	42
Modelo 3	0.99	47	1	45
Modelo 4	1	47	1	46
Modelo 5	1	46	1	52
Modelo 6	1	64	1	63
Modelo 7	1	51	1	50
Modelo 8	1	46	1	47

Fuente Autores

Tabla H.29. Resultados de Wavelet Symlet 4 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	38	1	87
Modelo 2	1	42	1	42
Modelo 3	1	44	1	45
Modelo 4	1	42	1	44
Modelo 5	1	72	1	52
Modelo 6	1	84	1	84
Modelo 7	1	50	1	54
Modelo 8	1	50	1	54

Fuente Autores

Tabla H.30. Resultados de Wavelet Symlet 4 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	99	1	93
Modelo 2	1	44	1	41
Modelo 3	0.97	64	0.97	63
Modelo 4	0.97	63	0.96	67
Modelo 5	1	48	1	49
Modelo 6	1	60	1	62
Modelo 7	0.98	61	0.97	67
Modelo 8	0.96	64	0.97	65

Fuente Autores

Tabla H.31. Resultados de Wavelet Symlet 6 con resolución hasta el cuarto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	28	1	27
Modelo 2	1	30	1	30
Modelo 3	0.68	89	0.71	89
Modelo 4	0.99	51	1	50
Modelo 5	1	45	1	53
Modelo 6	1	61	1	58
Modelo 7	1	50	1	51
Modelo 8	1	50	1	50

Fuente Autores

Tabla H.32. Resultados de Wavelet Symlet 6 con resolución hasta el cuarto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	1	28
Modelo 2	1	31	1	36
Modelo 3	0.72	88	0.69	91
Modelo 4	1	46	1	45
Modelo 5	1	63	1	62
Modelo 6	1	59	1	65
Modelo 7	1	44	1	50
Modelo 8	1	49	1	48

Fuente Autores

Tabla H.33. Resultados de Wavelet Symlet 6 con resolución hasta el cuarto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	26	1	25
Modelo 2	1	30	1	32
Modelo 3	0.69	99	0.65	129
Modelo 4	0.96	72	0.97	68
Modelo 5	1	45	1	48
Modelo 6	1	58	1	60
Modelo 7	0.99	62	0.99	60
Modelo 8	0.99	60	0.98	66

Fuente Autores

Tabla H.34. Resultados de Wavelet Symlet 6 con resolución hasta el quinto nivel usando *kernel* Polinómico.

MODELOS	Kernel POLINOMICO			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	30	0.99	39
Modelo 2	1	36	1	41
Modelo 3	1	48	1	49
Modelo 4	0.99	52	1	49
Modelo 5	1	56	1	57
Modelo 6	1	61	1	63
Modelo 7	1	52	1	59
Modelo 8	1	51	1	59

Fuente Autores

Tabla H.35. Resultados de Wavelet Symlet 6 con resolución hasta el quinto nivel usando *kernel* RBF.

MODELOS	Kernel RBF			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	29	1	30
Modelo 2	1	41	1	52
Modelo 3	1	43	1	41
Modelo 4	1	42	1	42
Modelo 5	1	63	1	64
Modelo 6	1	93	1	94
Modelo 7	1	58	1	60
Modelo 8	1	52	1	60

Fuente Autores

Tabla H.36. Resultados de Wavelet Symlet 6 con resolución hasta el quinto nivel usando *kernel* Sigmoidal.

MODELOS	Kernel SIGMOIDAL			
	3 Grupos		4 Grupos	
	Exactitud	Vectores Soporte	Exactitud	Vectores Soporte
Modelo 1	1	33	1	33
Modelo 2	1	41	1	41
Modelo 3	0.97	64	0.97	64
Modelo 4	0.99	68	0.97	69
Modelo 5	1	61	1	60
Modelo 6	1	69	1	64
Modelo 7	0.97	60	0.97	65
Modelo 8	0.99	66	0.97	61

Fuente Autores

H.3. Discusión

En vista que la mayoría de maquinas han logrado obtener resultados excelentes en el proceso de entrenamiento*, realizar la fase de clasificación con todas ellas demanda una gran cantidad de tiempo lo que resulta en un proceso tedioso. Se propone entonces, analizar las diferencias que existen entre ellas con base en algunos aspectos que pueden influir en el comportamiento de una prótesis real.

Teniendo en cuenta que una prótesis mio-eléctrica debe desempeñar su acción de control en un tiempo restringido, el primer aspecto para seleccionar cuales maquinas pasan a la fase prueba es comparar la exactitud entre maquinas entrenadas con nivel 4 y nivel 5 de resolución de la Wavelet. Observando los valores de exactitud entre ambos niveles es claro que no hay una ventaja relativa ni significativa entre ellas, así es que por este aspecto es mas conveniente usar las maquinas entrenadas con nivel 4 de resolución pues con ellas se requieren menor complejidad y demanda de tiempo en los cálculos.

El segundo aspecto influye de igual manera en el tiempo que puede requerir la prótesis real durante el proceso de control. Se comparan las maquinas entrenadas con 3 y 4 grupos de validación cruzada respecto a la cantidad de vectores soporte que necesitan†. Esta selección se hace únicamente con aquellas maquinas de nivel de resolución 4 de la Wavelet teniendo en cuenta el primer aspecto.

H.4. Tablas selección final de la wavelet

Los resultados de la selección final por Wavelet se muestran en las [tablas H.36](#) a [H.41](#).

* Algo que en particular los investigadores no esperábamos.

† Se tiene en cuenta únicamente la cantidad de vectores soporte dado que la exactitud en la mayoría de ellas es prácticamente igual.

Tabla H.37. Maquinas seleccionadas para fase de prueba Coiflet 3.

Coiflet 3 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	3	4	3
2	4	3	4
3	4	4	3
4	3	4	3
5	3	4	4
6	3	4	4
7	4	3	3
8	4	4	4

Fuente Autores

Tabla H.38. Maquinas seleccionadas para fase de prueba Coiflet 4.

Coiflet 4 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	4	3	3
2	3	4	3
3	3	4	3
4	4	4	3
5	4	3	4
6	3	3	3
7	3	3	3
8	3	3	3

Fuente Autores

Tabla H.39. Maquinas seleccionadas para fase de prueba Symlet 4.

Symlet 4 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	3	3	3
2	3	3	3
3	4	3	4
4	3	4	3
5	4	3	3
6	4	3	4
7	3	4	3
8	3	4	3

Fuente Autores

Tabla H.40. Maquinas seleccionadas para fase de prueba Symlet 6.

Symlet 6 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	4	4	4
2	3	3	3
3	3	4	3
4	4	4	4
5	3	4	3
6	4	3	3
7	3	3	4
8	3	4	3

Fuente Autores

Tabla H.41. Maquinas seleccionadas para fase de prueba Daubechies 4.

Daubechies 4 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	3	3	3
2	4	3	3
3	3	4	3
4	3	3	4
5	3	4	3
6	4	3	3
7	4	3	4
8	3	3	4

Fuente Autores

Tabla H.42. Maquinas seleccionadas para fase de prueba Daubechies 5.

Daubechies 5 Modelo	~ de grupos en Polinomico	~ de grupos en RBF	~ de grupos en Sigmoide
1	3	4	3
2	3	4	3
3	4	4	3
4	4	3	3
5	3	3	3
6	4	4	3
7	4	4	4
8	3	3	3

Fuente Autores