**Logotipo

Descripción generada automáticamente**

**Reconocimiento de posturas de mano del dataset Ninapro usando aprendizaje automático**

Jairo Alberto Agudelo Medina

Henry Alberto Arcila Ramírez

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesor  
Daniel Escobar Saltaren, Magíster en Ingeniería

Universidad de Antioquia  
Facultad de Ingeniería

Especialización en Analítica y Ciencia de Datos

Medellín, Antioquia, Colombia

2025

|  |  |
| --- | --- |
| **Cita** | (Agudelo Medina & Arcila Ramírez, 2025) |
| **Referencia**  **Estilo IEEE (2022)** | Agudelo Medina, J. A., & Arcila Ramírez, H. A. (2025). *Reconocimiento de posturas de mano del dataset Ninapro usando aprendizaje automático* [Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia. |

**** 

Grupo de Investigación Intelligent Information Systems Lab In2Lab

Especialización en Analítica y Ciencia de Datos, CohorteVIII.

Centro de Investigación Ambientales y de Ingeniería (CIA).

|  |  |
| --- | --- |
|  | Diagrama  Descripción generada automáticamente con confianza media |

Centro de Documentación Ingeniería (CENDOI)

**Repositorio Institucional:** http://bibliotecadigital.udea.edu.co

Universidad de Antioquia - www.udea.edu.co

**Rector:** John Jairo Arboleda Céspedes.

**Decano:** Julio Cesar Saldarriaga Molina

**Jefe departamento:** Danny Alexandro Múnera Ramírez

**Coordinadora del Programa:** Maria Bernarda Salazar Sánchez

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

**Dedicatoria**

Texto de dedicatoria centrado.

**Agradecimientos**

Texto de agradecimientos centrado.

**Tabla de contenido**

[Resumen 9](#_Toc199527718)

[1. Introducción 10](#_Toc199527719)

[2. Materiales y Métodos 12](#_Toc199527720)

[2.1. Fuente de datos 12](#_Toc199527721)

[2.2. Procesamiento de las señales musculares 13](#_Toc199527722)

[2.3. Modelos de inteligencia artificial 13](#_Toc199527723)

[2.4. Métricas 14](#_Toc199527724)

[3. Extracción de características 15](#_Toc199527725)

[4. Resultados 17](#_Toc199527726)

[5. Conclusiones 17](#_Toc199527727)

[Referencias 17](#_Toc199527728)

**Lista de Figuras**

[Figura 1. Señales de interés para un sujeto cualquiera. 11](#_Toc199580315)

[Figura 2. Muestra señales sEMG sujeto 1 13](#_Toc199580316)

[Figura 3. Posturas seleccionadas 14](#_Toc199580317)

[Figura 4. Método de preprocesamiento y extracción de características. 14](#_Toc199580318)

[Figura 5. Datos crudos del sujeto 1 para las posturas de interés. 15](#_Toc199580319)

[Figura 6. Señales sEMG crudas y filtradas 15](#_Toc199580320)

**Lista de Tablas**

[Tabla 1. Características dataset ejercicio 1 de los 27 sujetos. 10](#_heading=h.twrj2wolxbbz)

[Tabla 1. Reporte de claificación para la red U-Net. 11](#_heading=h.8tfkq4jmywne)

**Siglas, acrónimos y abreviaturas**

**IHM** Interfaz humano maquina

**ML** Machine Learning

**DL** Deep Learning

**ANN** Artificial Neural Network

**CNN** Convolutional Neural Network

**SVM** Support Vector Machines

**LDA** Linear Discriminant Analysis

**RF** Random forest

**KNN** K-Nearest Neighbors

**PCA** Principal Component Analysis

**ICA** Independent Component Analysis

**RBF** Radial Basis Function

**sEMG** Surface Electromyography

**RNN** Recurrent neural networks

**RMS** Root Mean Square

**WL** Waveform Length

**IAV** Integrated Absolute Value

**LSTM** Long Short-Term Memory Network

# Resumen

La pérdida de una extremidad corporal superior afecta significativamente la calidad de vida de quien lo padece. Para abordar esto se ha impulsado el desarrollo de prótesis, tanto comerciales como de código abierto, habilitando el acceso a esta tecnología a los pacientes. Un aspecto clave es el desarrollo del sistema de control, que involucra diseño electrónico, adquisición de datos y pruebas clínicas.

Los investigadores pueden utilizar bases de datos públicas como Ninapro y CapgMyo, que contienen señales electromiográficas, para entrenar modelos de aprendizaje automático y aprendizaje profundo , esenciales para el desarrollo de los controles. En este campo, debido a las limitaciones de los microcontroladores utilizados, técnicas clásicas de aprendizaje de maquina como máquinas de vectores de soporte, bosque aleatorio o k-vecinos más cercanos siguen siendo relevantes por su simplicidad y bajo requerimiento de recursos.

En este artículo se compara la capacidad de modelos de aprendizaje de maquina y aprendizaje profundo para clasificar dos tipos de agarre (movimientos 1 y 23) y el estado de reposo (movimiento 0) de la base de datos Ninapro DB1, utilizando como datos de entrada a los modelos, características extraídas las señales de electromiografía como RMS, WL e IAV.

***Palabras claves:*** *Señales electromiográficas, machine learning, NinaPro, modelos de clasificación, Root Mean Square.*

# Introducción

La pérdida de una extremidad superior es un problema que afecta la calidad de vida de cualquier persona, ya que tras una amputación su vida normal, independencia, funcionalidad, imagen de sí misma y su salud no solo física sino mental se ven comprometidas [1]. Con el fin de mejorar la calidad de vida perdida tras una amputación, el campo HMI (siglas en ingles de Interfaz Hombre Máquina) ha impulsado la investigación y desarrollo de prótesis tanto comerciales [2], [3], [4] como de código abierto [5], [6], [7] lo que ha contribuido a la democratización del acceso a esta tecnología, especialmente en contextos académicos y de investigación aplicada. Uno de los elementos claves de una prótesis es la concepción y diseño del sistema de control pues involucra no solo tareas de diseño electrónico, sino también el desarrollo de protocolos de adquisición de datos y pruebas clínicas [8].

El desarrollo de pruebas clínicas y de ingeniería involucran procesos de adquisición de datos que son costosos debido a las limitaciones en recursos humanos y tecnológicos. Afortunadamente existen datasets públicos como Ninapro [9] o CapgMyo [10] que comparten registros de diferentes señales electromiográficas. Estas bases de datos son ampliamente usadas por   la comunidad científica como insumo para el desarrollo de modelos de aprendizaje automático, (ML, del inglés Machine Learning) involucrados en el diseño de los sistemas de control de prótesis [11] que apuntan a mejorar la experiencia de usuario y la adherencia de estos.

En aprendizaje automático la extracción de características es una etapa fundamental en la que se busca capturar la mayor cantidad posible de información, con características de temporalidad y frecuencia, que luego será llevada a los modelos de clasificación y/o de clustering para entrenamiento y validación. Utilizar un conjunto de características cuidadosamente seleccionado puede proporcionar una tasa de clasificación más alta que la señal cruda [12]. Múltiples técnicas temporales relacionadas en [2b-c] y frecuenciales relacionadas en [2c] son utilizadas para extraer las características.

De igual forma la reducción de la dimensionalidad es a menudo necesaria especialmente en dominios tiempo-frecuencia de alta dimensionalidad. Se usa para reducir la carga del clasificador y/o el clustering y el tiempo computacional al reducir los datos a un espacio vectorial de baja dimensión mientras conservan la máxima información de la señal. La técnica más ampliamente usada es Análisis de Componentes Principales (PCA, del inglés - Principal Component Analysis).

Actualmente los desarrolladores de controles automáticos tienen algunas dificultades con el uso de sEMG, una de ella es la base de datos utilizada en el entrenamiento y validación, que a su vez depende de la cantidad de sensores usados, la posición de estos y el estado de salud y ánimo de los sujetos muestreados, etc.; otra es la escogencia de la técnica de ML o DL adecuada de entre las múltiples opciones que existen; y otra es la capacidad de los microcontroladores involucrados, generalmente con reducida memoria. Esto supone un gran reto para ellos.

El objetivo planteado en este trabajo es, usando las señales de EMG disponibles en la base de datos DB1 de Ninapro, comparar la capacidad de clasificación de diferentes de modelos de clasificación de DL y ML de agarres de mano a través del uso de las características individuales extraídas. Se trabajará con 2 tipos de agarre de la base de datos: 1, 23, y la posición de reposo, 0. Se quiere validar como varía la efectividad de los modelos de clasificación cuando se usa una cantidad específica de agarres, y no todos los disponible en la base de datos, y una característica específica extraída de las señales.

Este artículo se organiza de la siguiente manera. La información sobre los datos experimentales se analiza en la Sección II. La Sección III presenta la arquitectura DLPR propuesta para biorrobótica y prótesis basadas en EMG-PR. El resultado se destaca en la Sección IV. La Sección V abarca la discusión de los resultados. La conclusión de este trabajo se aborda en la Sección VI.

# Materiales y Métodos

En este artículo se analizan datos de EMG de superficie de la base de datos NinaPro DB1. Las señales de EMG se tomaron usando diez electrodos activos, de doble diferencial, marca OttoBock MyoBock 13E200, a un intervalo constante de 100Hz utilizando una tarjeta National Instruments DAQ (NI-DAQ PCMCIA 6024E) con una resolución de 12 bits. El foco se pone en el grupo de ejercicios 3, agarres, también identificados como grupo C, tomando los datos para los 27 sujetos y las 10 repeticiones por ejercicio realizadas por cada uno, y en particular en los ejercicios 1 y 23 y la posición de reposo, 0; el ejercicio 1 corresponde a un agarre de gran diámetro y el 23 a un agarre de cuchillo con extensión del dedo índice. Cada repetición de movimiento dura 5 segundos, seguida de 3 segundos de descanso (ver figura 1).

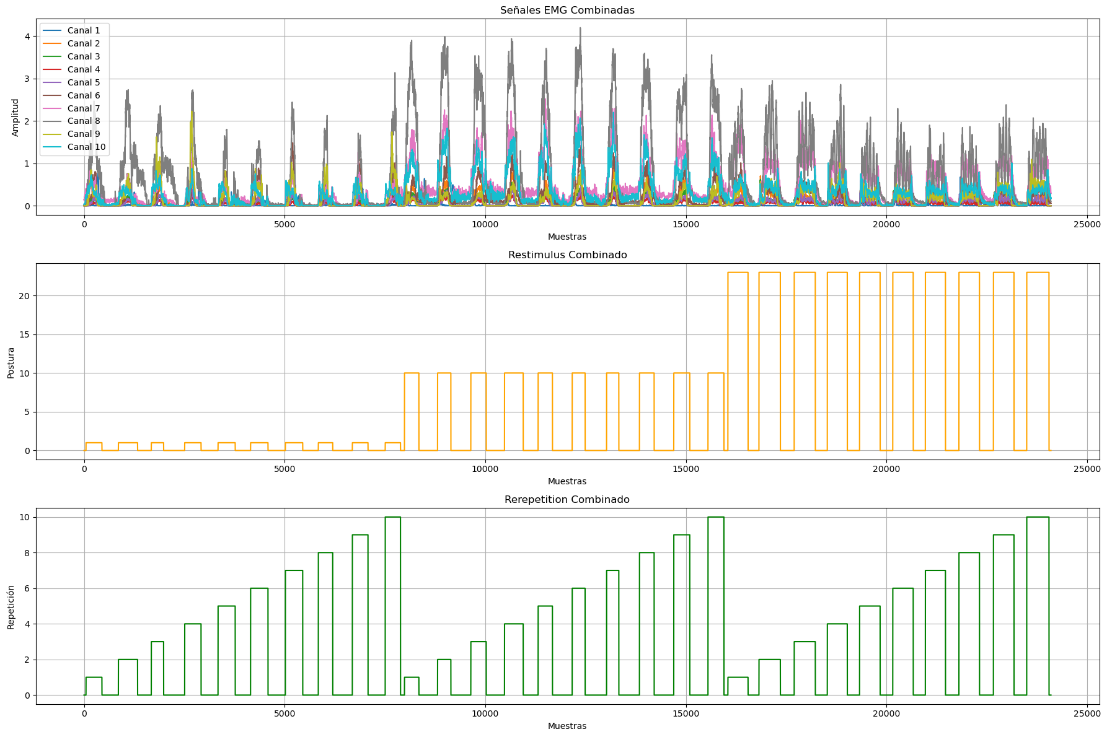


Figura 1. Señales de interés para un sujeto cualquiera.

Los datos se preprocesan y luego se extraen sus características usando RMS (del inglés, Root Mean Square), WL (del inglés, Waveform Length) e IAV (del inglés, Integrated Absolute Value). Es importante anotar, que todos los procedimientos previamente mencionados se realizaron empleando las funciones de la biblioteca LibEMG [13].

Finalmente, empleando los datos asociados a las características extraídas se procede a realizar el entrenamiento y test de algunos modelos de clasificación de ML y DL con estas características.

## Fuente de datos

La base de datos DB1 es de acceso público [13], se descarga como un conjunto de archivos ZIP, cada archivo ZIP corresponde a cada uno de los sujetos que participaron en el experimento, los cuales no presentaban amputación de ninguna extremidad: S1.zip, hasta S27.zip. Al descomprimir el archivo ZIP de un sujeto específico (por ejemplo, S1.zip), se obtienen varios archivos en formato .mat cada uno de estos archivos .mat corresponde a un ejercicio realizado por el sujeto.

Cada archivo .mat[[1]](#footnote-1) (por ejemplo, Sx\_A1\_Ey.mat, Sx\_A1\_Ey.mat, etc.) contiene variables específicas. Las variables más importantes y comunes dentro de cada archivo. mat son (ver Tabla 1):

* **emg**: es una matriz donde cada columna representa un canal de EMG (generalmente 10, correspondientes a los sensores sEMG). Las filas representan las muestras de EMG a lo largo del tiempo.
* **stimulus (o restimulus o gesture)**: es una columna vectorial que contiene la etiqueta del movimiento que se está realizando en cada instante de tiempo. Cero '0' reposo, y los números subsiguientes (1, 2, 3, ...) corresponden a los diferentes movimientos especificados.
* **subject**: escalar que indica el número del sujeto.
* **exercise**: escalar que indica el número del ejercicio (1, 2 o 3 para los tres tipos de ejercicios en DB1).
* **repetition**: vector que indica el número de repetición del gesto.
* **frequency**: frecuencia de muestreo de los datos EMG (100 Hz para DB1 después del preprocesamiento de Ninapro).

Tabla 1. Características dataset ejercicio 3 de los 27 sujetos.

|  |  |  |
| --- | --- | --- |
| **Variable** | **Descripción** | **Tipo de dato** |
| subject | Número del sujeto | Categórico |
| exercise | Número del ejercicio3 | Categórico |
| emg (12 col) | Señales sEMG. | Numérico |
| glove (22 col) | Señales sin procesar de del guante Cyberglove II. | Numérico |
| stimulus (1 col) | Movimiento repetido por el sujeto al seguir las instrucciones. | Categórico |
| restimulus (1 col) | Etiqueta colocada a posteriori para el movimiento repetido por el sujeto. | Categórico |
| repetition (1 col) | Repetición de stimulus | Categórico |
| rerepetition (1 col) | Repetición de rstimulus | Categórico |

Después de descomprimir cada uno de los archivos .zip de cada sujeto, el siguiente paso consiste en agrupar en un mismo directorio todos los archivos correspondientes al ejercicio 3 (Sx\_Ay\_E3.mat). Una vez agrupados los archivos archivos .mat, se procede a generar a partir de estos los archivos .csv correspondientes a cada una de las características de la tabla 1. Finalmente, como solo nos interesan las características asociadas a las señales sEMG (subject, emg, rerepetition y restimulus) solo se seleccionan aquellos archivos .csv que tengan que ver con estas y se ponen en un directorio aparte.

Luego, se procedió combinar cada uno de estos archivos csv en una única matriz numpy que se almacena como un binario (extensión .npy) para las señales de cada sujeto para operaciones posteriores de preprocesamiento sobre las señales. Cada uno de estos archivos contienen todos los datos sEMG crudos de cada usuario para cada una de las posturas originalmente calculadas. En la figura 3 se muestra una de las señales asociadas al sujeto 1.

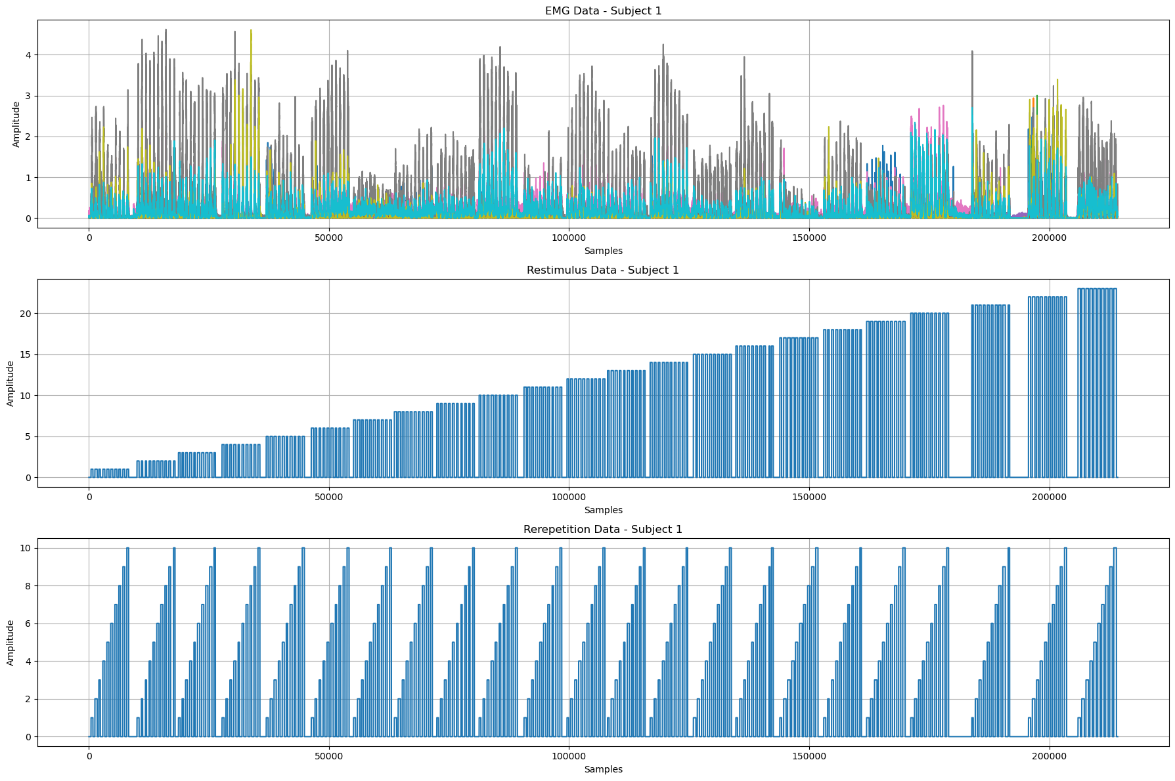


Figura 2. Muestra señales sEMG sujeto 1

Como solo interesan las posturas para los ejercicios 1 y 23 incluyendo los descansos se procede a seleccionar las muestras correspondientes solo a estas (figura 3) y se proceden a guardar en otro archivo .npy para realizar acciones de procesamiento digital de señales como filtrado, enventanado y extracción de características.

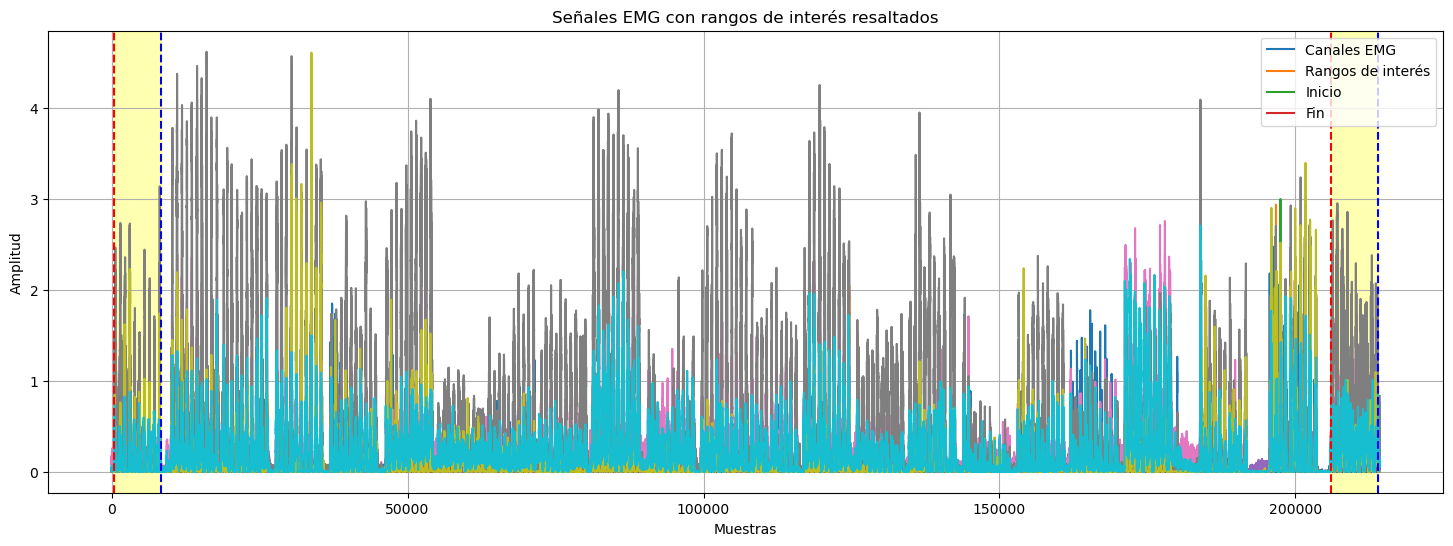


Figura 3. Posturas seleccionadas

## Procesamiento de las señales musculares

El procesamiento de los datos se realiza en 2 fases, como se ilustra en la gráfica 4: preprocesamiento o preparación de los datos y extracción de características.

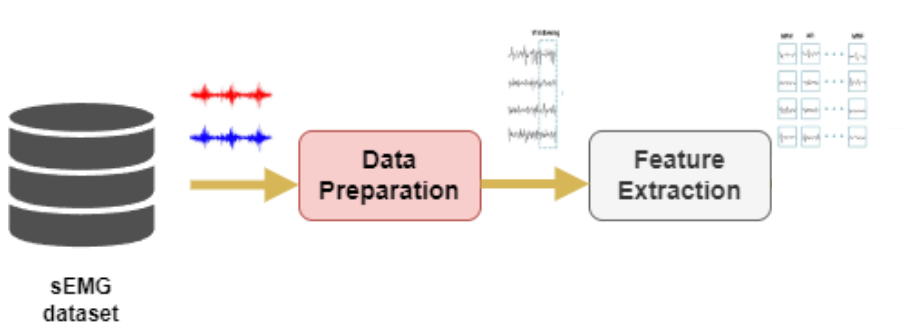


Figura 4. Método de preprocesamiento y extracción de características.

Para este caso, el preprocesamiento se refiere a la obtención de los datos crudos con las posturas elegidas (1, 23 y descanso) desde los archivos .npy y la extracción de estos hasta convertirlos en dataset que será empleado como entrada para el entrenamiento y test de los modelos. En la figura 5 se ilustra los datos crudos correspondientes a las posturas 1 y 23 (incluyendo descansos para el sujeto 1).

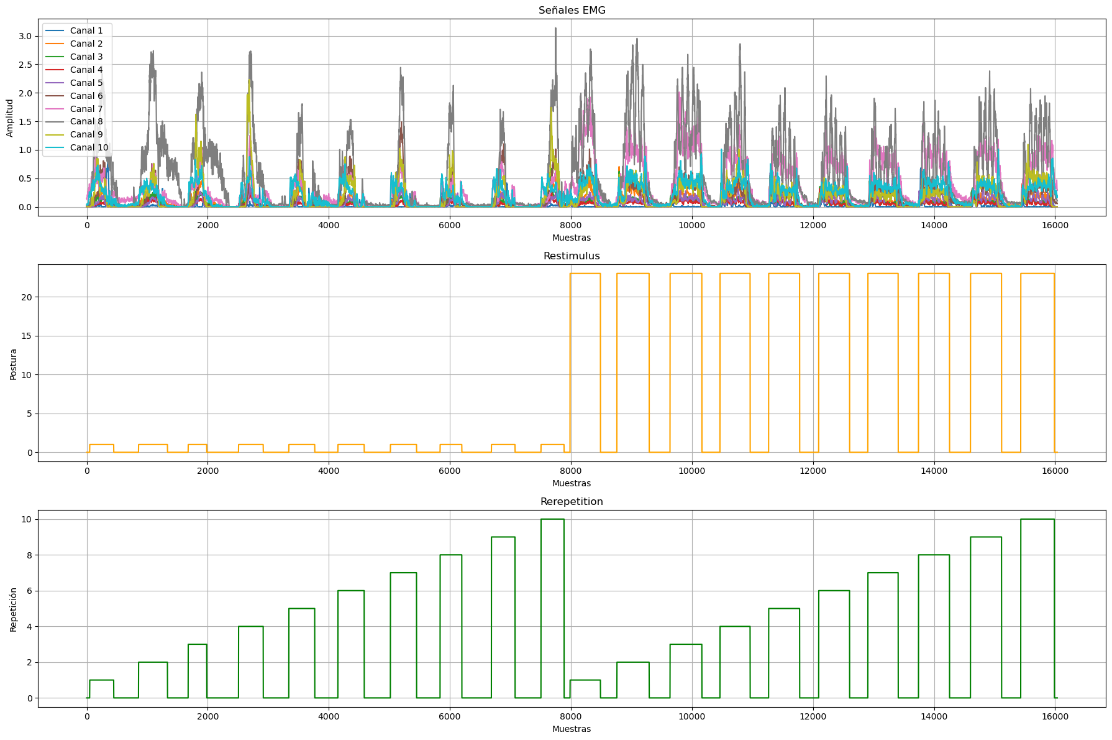


Figura 5. Datos crudos del sujeto 1 para las posturas de interés.

Como la señal es muy ruidosa, lo primero que se debe realizar es filtrarla para lo cual se emplea un filtro pasa bajas digital cuya frecuencia de corte y muestreo son respectivamente 1 Hz y 100 Hz. El resultado de este procedimiento es una señal más limpia y lista para la extracción de características (figura 6):

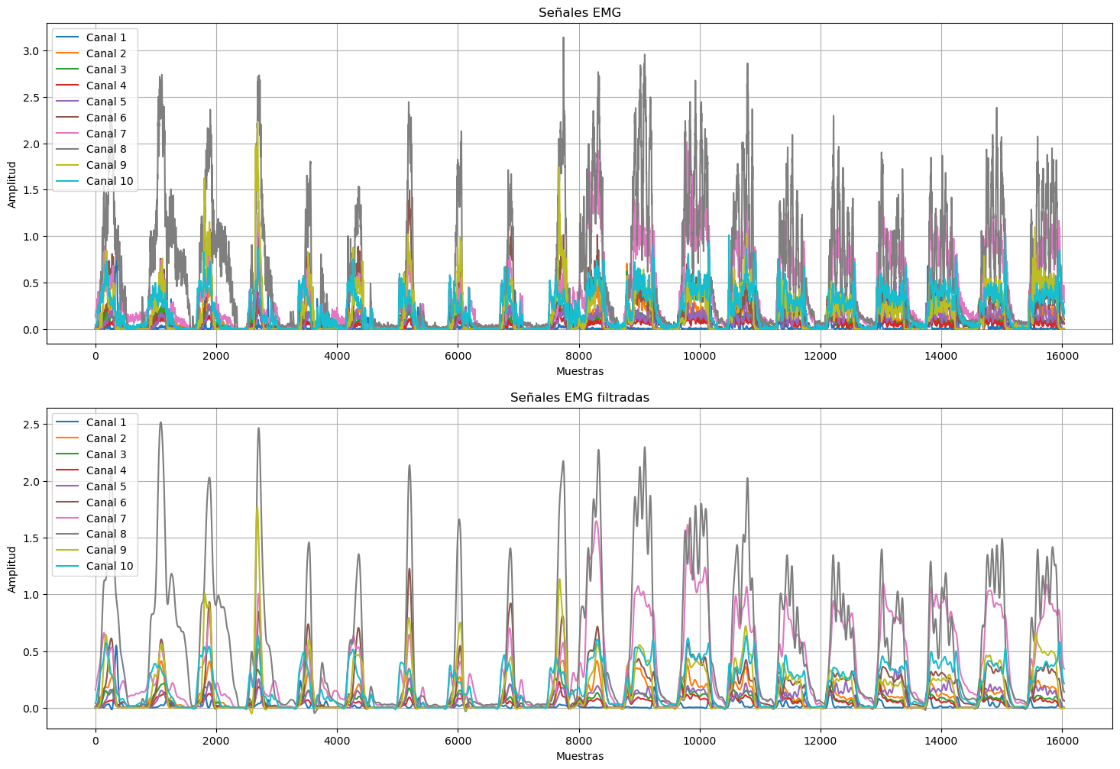


Figura 6. Señales sEMG crudas y filtradas

En esta fase, se llevó a cabo un proceso de segmentación sobre la señal previamente filtrada. A partir de cada segmento o ventana de señal seleccionada, se calcularon las características relevantes, que en este caso correspondieron al valor RMS (Root Mean Square), la Longitud de la Señal (WL, por sus siglas en inglés) y el Valor Absoluto Integrado (IAV).

Se emplearon ventanas de 30 muestras (equivalentes a 300 ms), con un solapamiento de 10 muestras (100 ms) entre ventanas consecutivas. Una vez calculadas las características para todas las ventanas, se generó un archivo .csv por cada sujeto, en el cual se almacenaron las características extraídas junto con la etiqueta (label) correspondiente a la postura asociada a cada ventana.

Cabe destacar que, además de las etiquetas asociadas a las posturas de interés (0, 1 y 23), se introdujo una etiqueta adicional, identificada con el valor -1, para aquellas ventanas que contenían muestras correspondientes a transiciones entre el estado de reposo y una postura, o viceversa. Este nuevo label tuvo como finalidad señalar que la ventana incluía una fase de transición y no debía considerarse como una postura estable.

## Modelos de inteligencia artificial

En esta etapa, las características extraídas en la fase anterior se utilizaron como entrada para los modelos de clasificación. Se evaluaron diferentes modelos tanto de aprendizaje automático (Machine Learning, ML) como de aprendizaje profundo (Deep Learning, DL), con el objetivo de clasificar tres clases: las posturas 1 y 23, y el estado de reposo (0), pertenecientes al conjunto de ejercicios 3 de la base de datos Ninapro DB1. En esta configuración, se optó por una estrategia de modelado independiente del sujeto, considerando que los datos entre individuos presentan una alta variabilidad y una baja correlación.

A partir de cada archivo .csv generado por sujeto, se construyeron tres matrices distintas, una por cada característica extraída (RMS, WL e IAV), acompañadas de sus respectivas etiquetas (labels), que incluyeron las clases 0, 1, 23 y -1. Esta última clase, como se explicó anteriormente, representa ventanas asociadas a transiciones.

En particular, para cada característica se generaron 10 columnas correspondientes a los 10 canales de adquisición de la señal EMG. Por ejemplo, en el caso de la característica RMS, las columnas se nombraron como RMS\_ch1, RMS\_ch2, ..., RMS\_ch10. De forma análoga se nombraron las columnas para las características WL (WL\_ch1 a WL\_ch10) e IAV (IAV\_ch1 a IAV\_ch10). A continuación, se muestra la estructura general de estas matrices:

Tabla 2. Dataset de características de las señales sEMG para las posturas elegidas.

|  |  |  |  |
| --- | --- | --- | --- |
| **Matriz** | **Variable** | **Descripción** | **Tipo de dato** |
| RMS | RMS\_chx[[2]](#footnote-2) (10 col) | Característica RMS | Numérico |
| label | Postura (0, 1, 23, -1) | Categórico |
| WL | WL\_chx (10 col) | Característica WL | Numérico |
| label | Postura (0, 1, 23, -1) | Categórico |
| IAV | IAV\_chx (10 col) | Característica IAV | Numérico |
| label | Postura (0, 1, 23, -1) | Categórico |

Dado que los datos de entrada incluían cuatro etiquetas distintas (0, 1, 23 y -1), el primer paso consistió en eliminar las entradas correspondientes a ventanas con la etiqueta -1, las cuales representaban transiciones entre posturas. De este modo, el conjunto de datos resultante para el entrenamiento de los modelos incluyó únicamente las clases de interés: reposo (0) y las posturas 1 y 23.

Es importante resaltar que las clases se encontraban desbalanceadas, siendo la mayoría de las muestras correspondientes al estado de reposo (clase 0). Esta situación requirió aplicar estrategias diferenciadas según el tipo de modelo utilizado.

Para cada característica (RMS, WL, IAV), el conjunto de datos fue dividido en un 80% para entrenamiento y un 20% para prueba. En el caso de los modelos de aprendizaje automático, se aplicó un proceso de submuestreo aleatorio (undersampling) para balancear las clases durante el entrenamiento. Por otro lado, para el modelo de aprendizaje profundo se utilizó un muestreo estratificado, con el fin de conservar la proporción original de clases tanto en el conjunto de entrenamiento como en el de prueba.

Adicionalmente, antes de alimentar los modelos, se aplicó un procedimiento de escalado de características para normalizar los datos y facilitar la convergencia durante el entrenamiento.

La siguiente tabla resume los modelos que fueron considerados en el proceso de evaluación:

Tabla 3. Resumen de las características de los modelos probados.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Categoría** | **Tipo** | **Modelo** | **Entradas** | **Balanceo** | **Salidas** |
| Aprendizaje de maquina | SVM | 1 | RMS | Si | 0, 1, 23 |
| 2 | WL | Si | 0, 1, 23 |
| 3 | IAV | Si | 0, 1, 23 |
| KNN | 4 | RMS | Si | 0, 1, 23 |
| 5 | WL | Si | 0, 1, 23 |
| 6 | IAV | Si | 0, 1, 23 |
| RF | 7 | RMS | Si | 0, 1, 23 |
| 8 | WL | Si | 0, 1, 23 |
| 9 | IAV | Si | 0, 1, 23 |
| Aprendizaje profundo | CNN-LTSM | 10 | RMS | No | 0 [0], 1 [1], 2 [23] |
| 11 | WL | No | 0 [0], 1 [1], 2 [23] |
| 12 | IAV | No | 0 [0], 1 [1], 2 [23] |

Como se puede observar en el caso del modelo de aprendizaje profundo, las salidas presentan una codificación diferente, ya que se realizó un procedimiento de **relabeling**, en el cual las etiquetas originales 0, 1 y 23 (valores entre corchetes) fueron mapeadas a los valores 0, 1 y 2, respectivamente. Esta transformación facilitó el manejo de las clases en la función de salida (output layer) del modelo, que se diseñó para realizar clasificación multiclase con codificación categórica.

Las características espeficas de los modelos empleados se describen a continuación:

* **Maquinas de vectores se soporte (SVM)**: Clasificador eficaz en espacios de alta dimensión, sensible a datos desbalanceados.

|  |  |  |
| --- | --- | --- |
| **Modelo** | **Entradas** | **Características específicas** |
| 1 | RMS | * **kernel**: RBF * **C**: 100 * **gamma**:1 |
| 2 | WL | * **kernel**: RBF * **C**: 100 * **gamma**:0.1 |
| 3 | IAV | * **kernel**: RBF * **C**: 100 * **gamma**:1 |

* **K-Nearest Neighbors (KNN)**: Clasificación basada en distancia.

|  |  |  |
| --- | --- | --- |
| **Modelo** | **Entradas** | **Características específicas** |
| 3 | RMS | * **Vecinos**: 3 * **Métrica**: Manhattan |
| 4 | WL | * **Vecinos**: 9 * **Métrica**: Manhattan |
| 5 | IAV | * **Vecinos**: 3 * **Métrica**: Manhattan |

* **Random Forest (RF)**: Ensamble de árboles de decisión.

|  |  |  |
| --- | --- | --- |
| **Modelo** | **Entradas** | **Características específicas** |
| 6 | RMS | * **Número de árboles (n\_estimators)**: 200 * **Profundidad máxima de los árboles (max\_depth)**: 20 * **Mínimo número de muestras por hoja (min\_samples\_leaf)**: 1 * **Mínimo número de muestras para dividir un nodo (min\_samples\_split)**: 2 * **Criterio de división**: Gini (valor por defecto) |
| 7 | WL | * **Número de árboles (n\_estimators)**: 200 * **Profundidad máxima de los árboles (max\_depth)**: None (sin límite) * **Mínimo número de muestras por hoja (min\_samples\_leaf)**: 1 * **Mínimo número de muestras para dividir un nodo (min\_samples\_split)**: 2 * **Criterio de división**: Gini (valor por defecto) |
| 8 | IAV | * **Número de árboles (n\_estimators)**: 200 * **Profundidad máxima de los árboles (max\_depth)**: 20 * **Mínimo número de muestras por hoja (min\_samples\_leaf)**: 1 * **Mínimo número de muestras para dividir un nodo (min\_samples\_split)**: 2 * **Criterio de división**: Gini (valor por defecto) |

* **Arquitectura hibrida CNN-LTSM**: Arquitectura híbrida que combina una red convolucional 1D para extraer patrones locales de las señales EMG, seguida de una capa LSTM que modela la dependencia temporal entre ventanas. Esta combinación permite capturar tanto la estructura espacial como la secuencialidad de los datos. La arquitectura empleada en nuestro caso fue propuesta por Lin Chen et al [14] que se muestra en la siguiente figura:

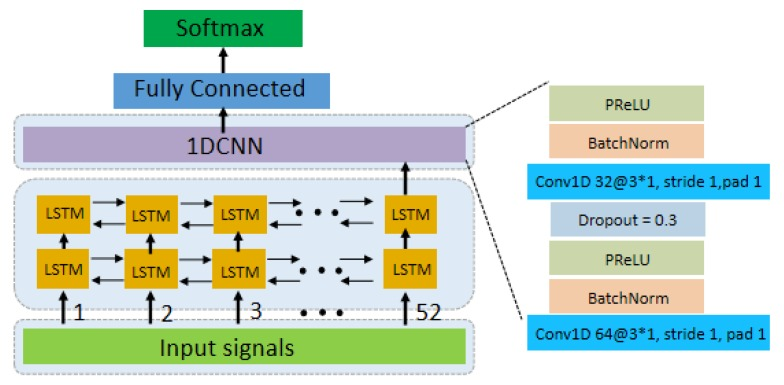


Figura 7. Arquitectura red CNN-LSTM propuesta por Lin Chen et Al [14]

La siguiente tabla resume la arquitectura de red previamente descrita la cual es la misma para las diferentes características empleadas como entrada RMS (Modelo 10) , WF (Modelo 11) e IAV(Modelo 12):

|  |  |  |
| --- | --- | --- |
| **Capa (Tipo)** | **Forma de salida** | **Parámetros** |
| input\_signals (Capa de entrada) | (Ninguno, 1, 10) | 0 |
| lstm\_1 (LSTM) | (Ninguno, 1, 64) | 19200 |
| lstm\_2 (LSTM) | (Ninguno, 1, 64) | 33024 |
| conv1d\_1 (Conv1D) | (Ninguno, 1, 64) | 12352 |
| batchnorm\_1 (Normalización por lotes) | (Ninguno, 1, 64) | 256 |
| prelu\_1 (PReLU) | (Ninguno, 1, 64) | 64 |
| dropout\_1 (Dropout) | (Ninguno, 1, 64) | 0 |
| conv1d\_2 (Conv1D) | (Ninguno, 1, 32) | 6176 |
| batchnorm\_2 (Normalización por lotes) | (Ninguno, 1, 32) | 128 |
| prelu\_2 (PReLU) | (Ninguno, 1, 32) | 32 |
| global\_avg\_pooling (Agrupación promedio global 1D) | (Ninguno, 1, 32) | 0 |
| fully\_connected\_dense (Densa) | (Ninguno, 1, 32) | 1056 |
| prelu\_fc (PReLU) | (Ninguno, 1, 32) | 32 |
| output\_softmax (Densa) | (Ninguno, 3) | 99 |

## Métricas de evaluación

El rendimiento de los modelos de clasificación se midió utilizando las siguientes métricas:

* **Precisión:** proporción de predicciones positivas correctas:

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

Donde,

* + **TP**: verdaderos positivos (correctamente clasificados como positivos).
  + **FP**: falsos positivos (incorrectamente clasificados como positivos).
* **Recall**: proporción de casos positivos reales que fueron correctamente identificados:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

Donde,

* + **FN**: falsos negativos (casos positivos incorrectamente clasificados como negativos).
  + **TP**: verdaderos positivos (correctamente clasificados como positivos).

Para ambos casos se considera porcentajes superiores al 80% para la selección de los modelos.

* **F1**-**score**: Media armónica de las métricas de precisión y recall:

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

* **ROC** (Del inglés, Receiver Operating Characteristic): gráfico de Tasa de Verdaderos Positivos (Recall) vs. Tasa de Falsos Positivos (FP/(FP+TN)) para distintos umbrales de clasificación.
* **AUC** (Área bajo la curva ROC): medida de la capacidad del clasificador para distinguir entre clases. El valor de AUC se encuentra entre 0 y 1 (1 = clasificador perfecto, 0.5 = aleatorio).

# Extracción de características

RMS, WL e IAV son métodos comunes para la extracción de características en señales de electromiografía superficial (sEMG). Su objetivo es reducir la dimensionalidad de la señal y capturar información relevante para aplicaciones como control de prótesis, diagnóstico médico o interfaz cerebro-máquina.

* **RMS** (Root Mean Square): representa la energía promedio de la señal sEMG en una ventana de tiempo. Es útil para estimar la activación muscular (mayor RMS = mayor contracción). Presenta robustez frente a ruido (suaviza variaciones rápidas). Útil para diferenciar entre reposo y contracción.

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

Donde,

* + **xi**: Muestra de la señal sEMG en el instante i.
  + **N**: Número de muestras en la ventana.
* **WL** (Waveform Length): mide la complejidad temporal de la señal (suma de diferencias entre muestras consecutivas). Es sensible a cambios en la amplitud y frecuencia de la sEMG. Indicador de patrones de activación muscular. Ayuda a distinguir gestos rápidos (ej: pinza) vs. lentos (ej: agarre sostenido).

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

Donde,

* + **xi**: Muestra de la señal sEMG en el instante i.
  + **N**: Número de muestras en la ventana.

Tiene aplicación en discriminación de gestos (ej: mano abierta vs. puño), detección de movimientos dinámicos.

* **IAV** (Integral Absolute Value): calcula el área bajo la curva de la señal sEMG en valor absoluto. Relacionado con la actividad muscular total en un intervalo. Similar al RMS pero menos sensible a picos abruptos.

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

Donde,

* + **xi**: Muestra de la señal sEMG en el instante i.
  + **N**: Número de muestras en la ventana.

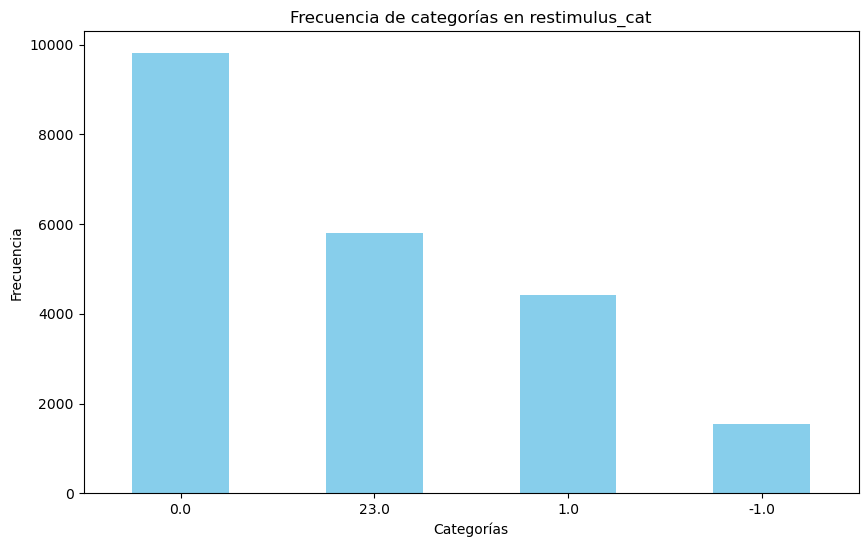
Su aplicación en sEMG es útil para estimación de la intensidad de la contracción, cuantificará la fuerza aplicada. Usado ampliamente en sistemas de control en tiempo real por su bajo costo computacional.

# Resultados y Discusión

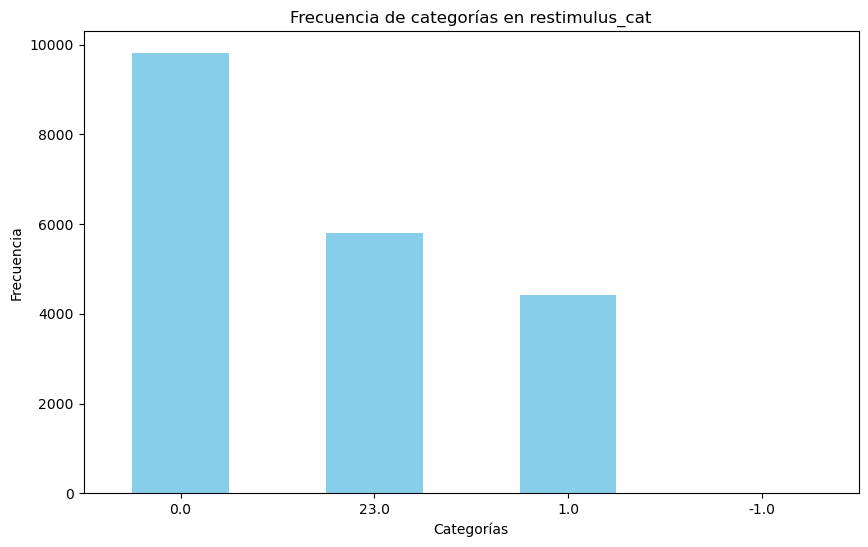
Ejemplo de la sEMG

Características

Sin quitar -1



Quitando el -1



**RMS (sin tener en cuenta el -1)**

Matriz de Correlación

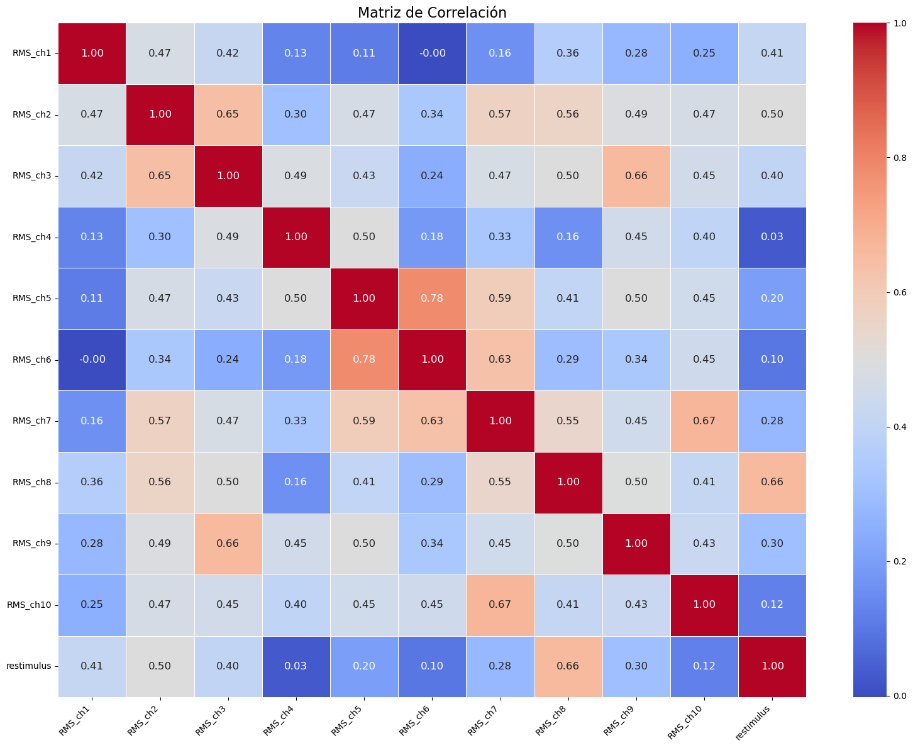


Diagrama de cajas y bigote por categoría por canal

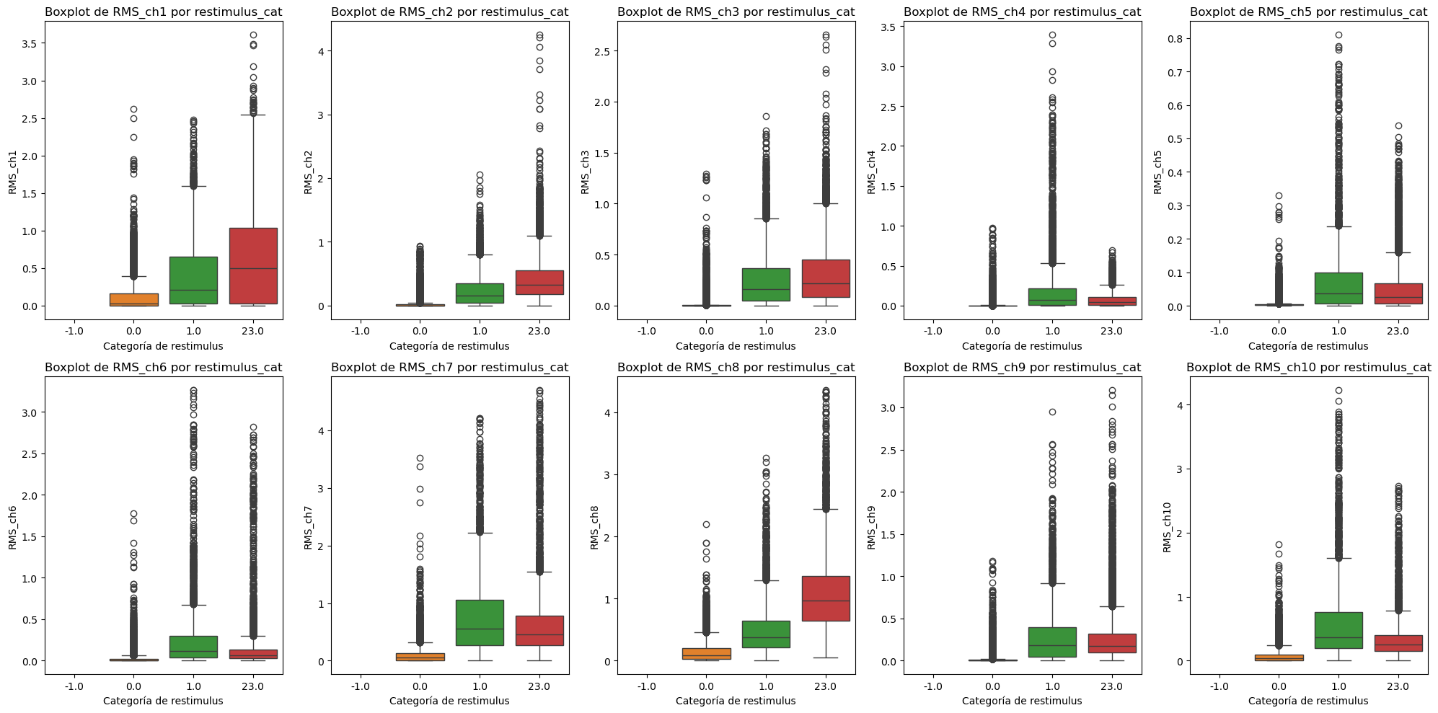
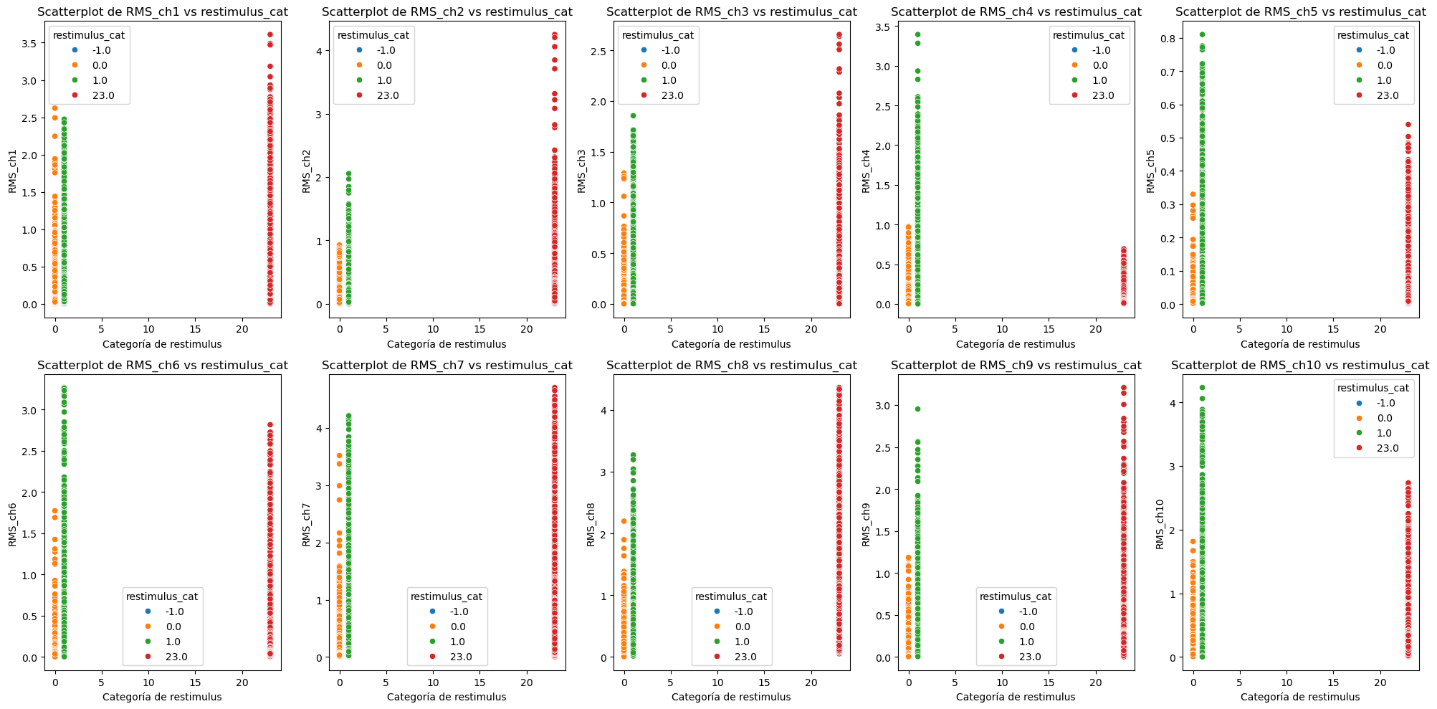


Diagrama de dispersión por canal



**WL (sin tener en cuenta el -1)**

Matriz de correlación

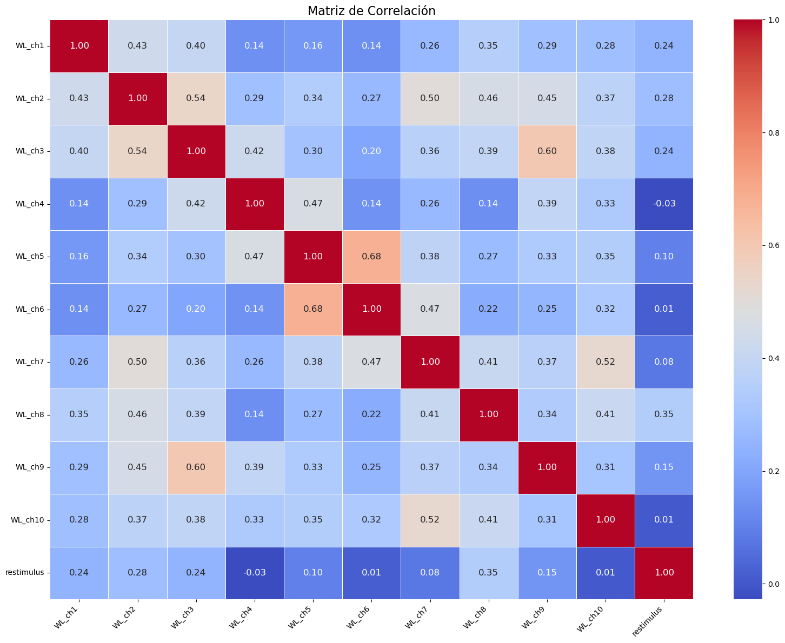


Diagrama de cajas y bigote por categoría por canal

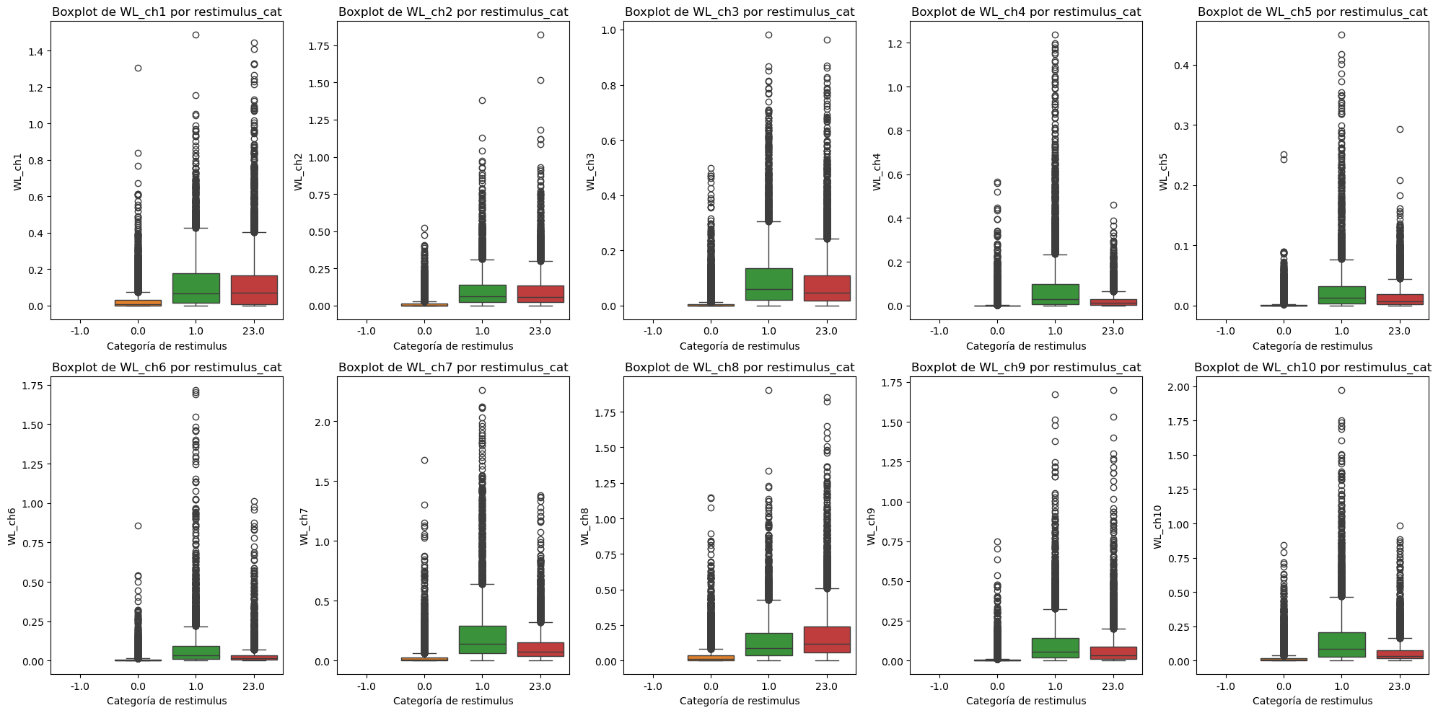
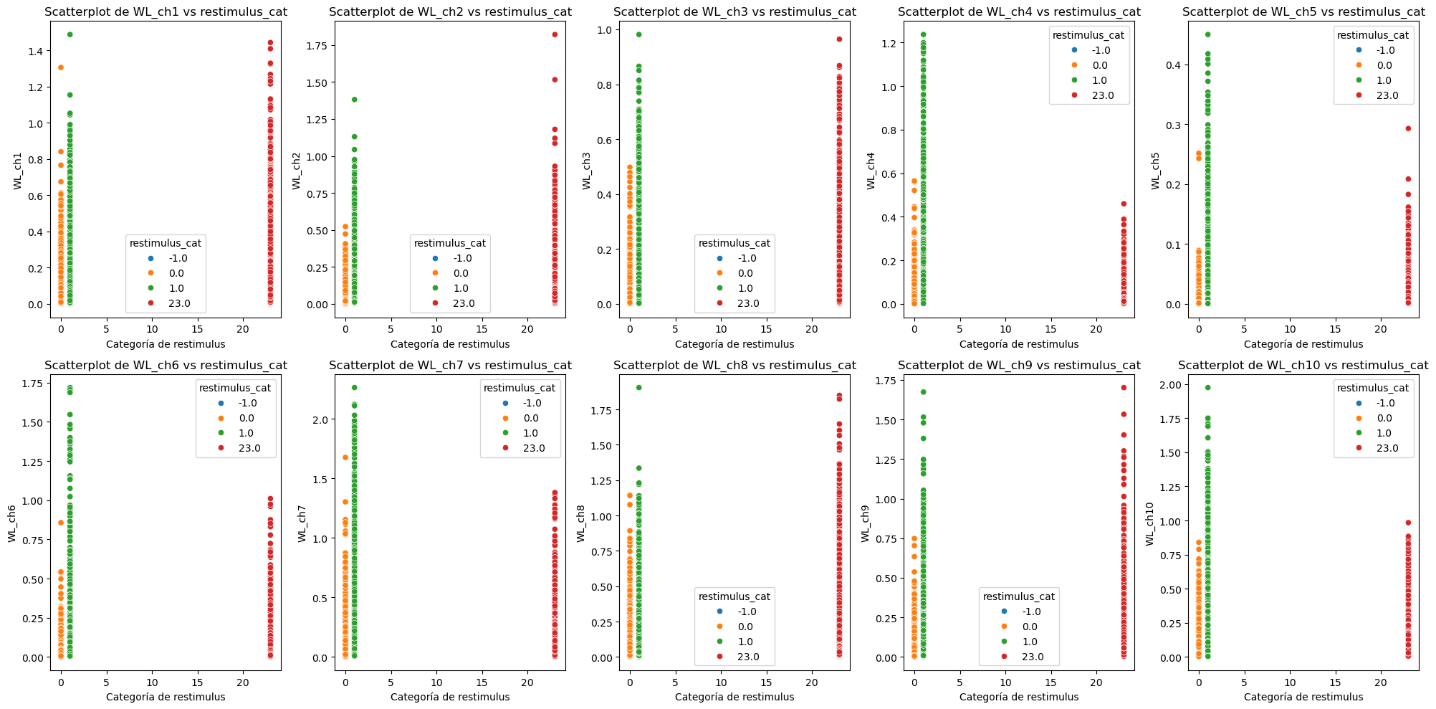


Diagrama de dispersión por canal



**IAV (sin tener en cuenta el -1)**

Matriz de correlación

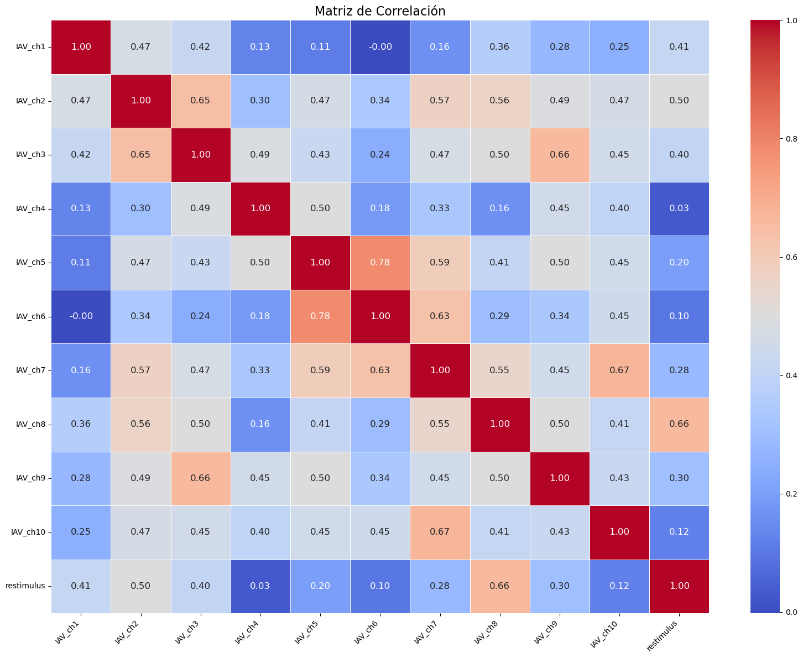


Diagrama de cajas y bigote por categoría por canal

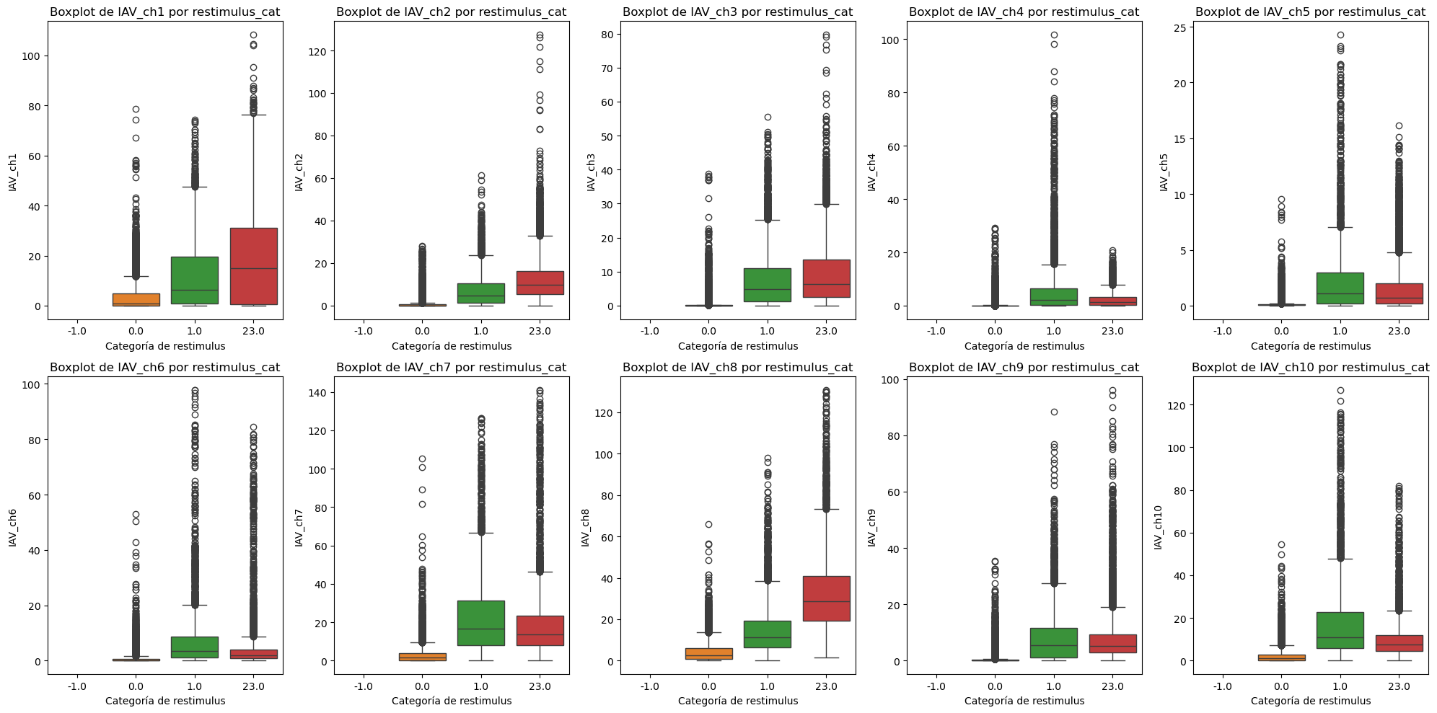
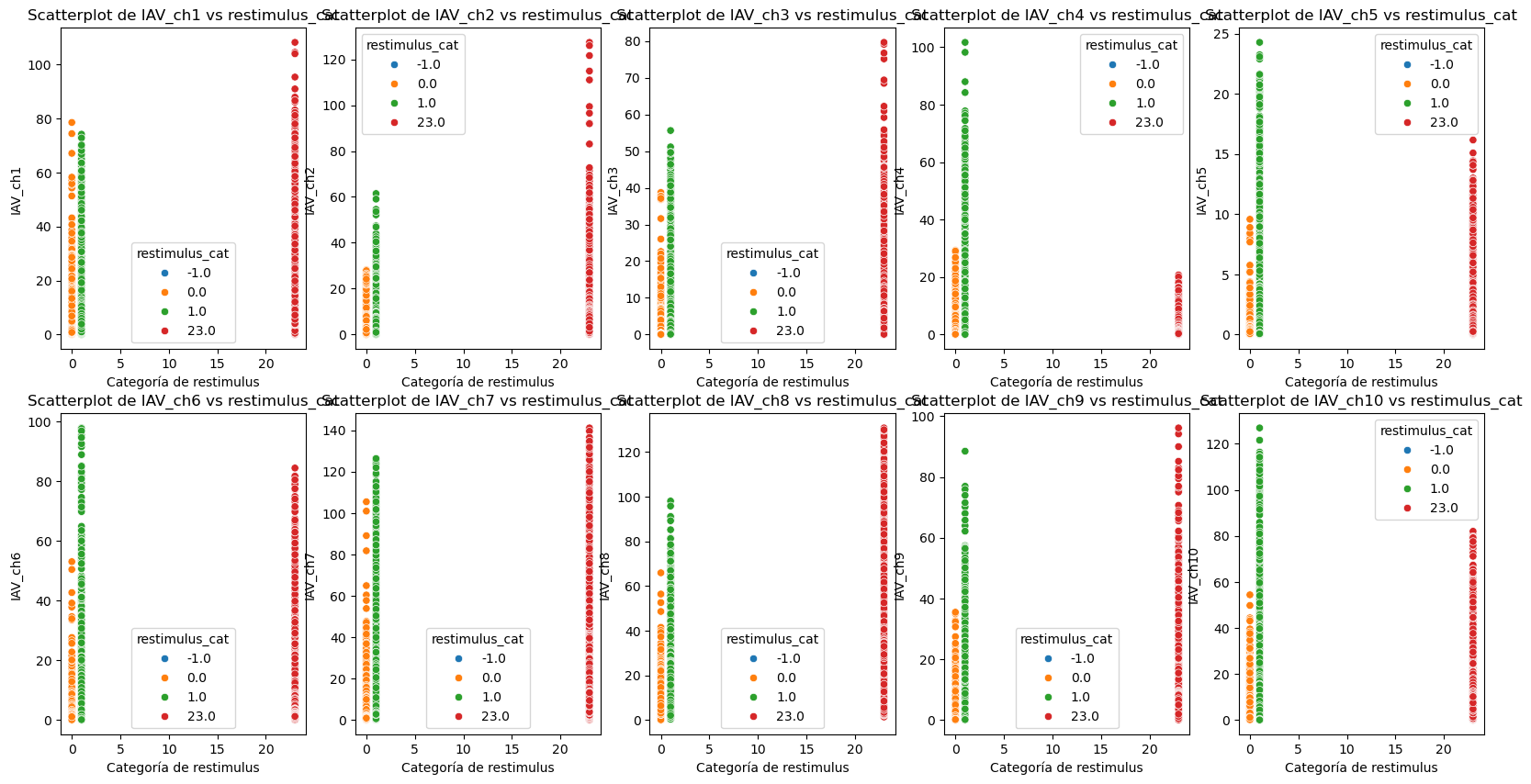


Diagrama de dispersión por canal



Resultados de modelos

**Modelos SVM**

Matrices de confusión…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Matrices de confusión modelos SVM(RMS), SVM(WF) y SVM(IAV)

Curvas ROC…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Curvas ROC modelos SVM(RMS), SVM(WF) y SVM(IAV)

**Modelos KNN**

Matrices de confusión…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Matrices de confusión modelos KNN(RMS), KNN(WF) y KNN(IAV)

Curvas ROC…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Curvas ROC modelos KNN(RMS), KNN(WF) y KNN(IAV)

**Modelos RF**

Matrices de confusión…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Matrices de confusión modelos RF(RMS), RF(WF) y RF(IAV)

Curvas ROC…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Curvas ROC modelos RF(RMS), RF(WF) y RF(IAV)

**Modelos CNN-LTSM**

Es importante anotar lo de que no se balancean los datos y lo del relabeling que se hace sobre los parametros

Matrices de confusión…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Matrices de confusión modelos CNN-LSTM(RMS), CNN-LSTM (WF) y CNN-LSTM (IAV)

Curvas ROC…

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Fig**. Curvas ROC modelos CNN-LSTM (RMS), CNN-LSTM(WF) y CNN-LSTM(IAV)

Tabla resumen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Modelo** | **Características** | **Precisión (M/W)** | **Recall (M/W)** | **F1-Score (M/W)** | **AUC** |
| 1 - SVM(RMS) | * **kernel**: RBF * **C**: 100 * **gamma**:1 | 0.95 | 0.95 | 0.95 | * **0:** 0.99 * **1:** 0.98 * **23:** 0.99 |
| 2 - SVM(WL) | * **kernel**: RBF * **C**: 100 * **gamma**:0.1 | 0.83 | 0.83 | 0.83 | * **0:** 0.96 * **1:** 0.93 * **32:** 0.94 |
| 3 - SVM(IAV) | * **kernel**: RBF * **C**: 100 * **gamma**:1 | 0.94 | 0.94 | 0.94 | * **0:** 0.98 * **1:** 0.98 * **23:** 0.99 |
| 4 - KNN(RMS) | * **Vecinos**: 3 * **Métrica**: Manhattan | 0.95 | 0.95 | 0.95 | * **0:** 0.98 * **1:** 0.98 * **23:** 0.99 |
| 5 - KNN(WL) | * **Vecinos**: 9 * **Métrica**: Manhattan | 0.84 | 0.84 | 0.84 | * **0:** 0.96 * **1:** 0.94 * **23:** 0.95 |
| 6 - KNN(IAV) | * **Vecinos**: 3 * **Métrica**: Manhattan | 0.95 | 0.95 | 0.95 | * **0:** 0.98 * **1:** 0.98 * **23:** 0.99 |
| 7 - RF(RMS) | * **Número de árboles**: 200 * **Profundidad máxima de los árboles**: 20 * **Mínimo número de muestras por hoja**: 1 * **Mínimo número de muestras para dividir un nodo**: 2 | 0.94 | 0.94 | 0.94 | * **0:** 0.99 * **1:** 0.99 * **2:** 0.99 |
| 8 - RF(WL) | * **Número de árboles**: 200 * **Profundidad máxima de los árboles**: None * **Mínimo número de muestras por hoja**: 1 * **Mínimo número de muestras para dividir un nodo**: 2 | 0.85 | 0.85 | 0.85 | * **0:** 0.97 * **1:** 0.95 * **23:** 0.95 |
| 9 - RF(IAV) | * **Número de árboles**: 200 * **Profundidad máxima de los árboles**: 20 * **Mínimo número de muestras por hoja**: 1 * **Mínimo número de muestras para dividir un nodo**: 2 | 0.94 | 0.94 | 0.94 | * **0:** 0.94 * **1:** 0.94 * **23:** 0.94 |
| 10 - CNN-LSTM(RMS) | * **Conv1D** (×2) * **LSTM** (×2, 52 celdas, 64 unidades) * **Capa de salida** | 0.94 / 0.93 | 0.92 / 0.93 | 0.92 / 0.93 | * **0 [0]:** 0.99 * **1 [1]:**  0.98 * **2 [23]:** 0.99 |
| 11 - CNN-LSTM(WL) | 0.82 / 0.84 | 0.81 / 0.84 | 0.82 / 0.84 | * **0 [0]:** 0.96 * **1 [1]:** 0.94 * **2 [23]:** 0.94 |
| 12 - CNN-LSTM(IAV) | 0.93 / 0.93 | 0.91 / 0.93 | 0.92 / 0.93 | * **0 [0]:** 0.99 * **1 [1]:**  0.98 * **2 [23]:** 0.99 |

# Conclusiones

To Do…

# Referencias

[1] A. C. Roșca, C. C. Baciu, V. Burtăverde, and A. Mateizer, “Psychological Consequences in Patients With Amputation of a Limb. An Interpretative-Phenomenological Analysis,” *Front Psychol*, vol. 12, 2021, doi: 10.3389/fpsyg.2021.537493.

[2] Ottobock, “Myoelectric Hand System 8E70,” 2025. Accessed: Apr. 13, 2025. [Online]. Available: https://www.ottobock.com/en-us/product/8E70

[3] Open Bionics, “Hero Arm Overview,” Open Bionics. Accessed: Apr. 13, 2025. [Online]. Available: https://openbionics.com/en/hero-arm-overview/

[4] Prótesis Avanzadas Colombia, “Prótesis A3D,” Prótesis Avanzadas Colombia. Accessed: Apr. 13, 2025. [Online]. Available: https://www.protesisavanzadas.co/pr%C3%B3tesis-a3d

[5] exiii Inc., “Hackberry - Open-source 3D printable bionic hand,” exiii Inc. Accessed: Apr. 13, 2025. [Online]. Available: https://www.exiii-hackberry.com/

[6] OpenBionics, “OpenBionics – Open-source robotic and bionic devices,” OpenBionics. Accessed: Apr. 13, 2025. [Online]. Available: https://openbionics.org/

[7] Enabling The Future, “Enabling The Future – A Global Network Of Passionate Volunteers Using 3D Printing To Give The World A Helping Hand,” Enabling The Future. Accessed: Apr. 13, 2025. [Online]. Available: https://enablingthefuture.org/

[8] M. Suárez García, “Adaptación de la Prótesis de Mano Basada en Soft-Robotics PrExHand para la Evaluación con Usuarios no Patológicos en Pruebas Funcionales,” Escuela Colombiana de Ingeniería Julio Garavito, 2021. doi: 10.48713/10336\_31566.

[9] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori, “Comparison of six electromyography acquisition setups on hand movement classification tasks,” *PLoS One*, vol. 12, no. 10, p. e0186132, Oct. 2017, doi: 10.1371/journal.pone.0186132.

[10] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng, “Surface EMG-Based Inter-Session Gesture Recognition Enhanced by Deep Domain Adaptation,” *Sensors*, vol. 17, no. 3, 2017, doi: 10.3390/s17030458.

[11] J. Yang, M. Soh, V. Lieu, D. J. Weber, and Z. Erickson, “EMGBench: Benchmarking Out-of-Distribution Generalization and Adaptation for Electromyography,” 2024. [Online]. Available: https://arxiv.org/abs/2410.23625

[12] R. H. Chowdhury, M. B. I. Reaz, M. A. B. M. Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, “Surface Electromyography Signal Processing and Classification Techniques,” *Sensors*, vol. 13, no. 9, pp. 12431–12466, 2013, doi: 10.3390/s130912431.

[13] Ninapro Project Team, “DB1 Guidelines,” Ninapro Project – HEVS. Accessed: Apr. 13, 2025. [Online]. Available: https://ninapro.hevs.ch/instructions/DB1.html

1. Cara archivo .mat tiene el siguiente formato: Sx\_A1\_Ey.mat; donde x es el número del sujeto (1 hasta 27) e y corresponde al número del ejercicio (1, 2 o 3). [↑](#footnote-ref-1)
2. En RMS\_chx, la letra x está asociada al canal y toma valores entre 1 y 10 incluidos. [↑](#footnote-ref-2)