

Seguimiento 4: Machine Learning

Danna Isabella García Saenz - Solanlly Evenedy Montoya Rivera

Mayo 31, 2024

b. Definir una estrategia de filtrado de las señales del proyecto 1 usando filtros IIR o FIR justificando la selección.

Para filtrar las señales del proyecto 1 se tuvieron en cuenta las características que diferencian los filtros digitales IIR (Respuesta infinita al impulso) y FIR (Respuesta finita al impulso), en donde se tiene que los filtros IIR tienen menor rizado y aunque necesitan menor orden pueden llegar a ser inestables. Los filtros IIR (Infinite Impulse Response) suelen ser preferidos para el procesamiento de señales de EMG por varias razones, primero tienen una respuesta en frecuencia selectiva que permite filtrar con precisión las frecuencias no deseadas en las señales de EMG y además son más eficientes en términos computacionales comparados con los filtros FIR lo que los hace ideales para aplicaciones en tiempo real, como el análisis de señales de EMG durante el movimiento, también, los filtros IIR ofrecen flexibilidad en el diseño, permitiendo ajustar el orden del filtro y la frecuencia de corte para adaptarse mejor a las características específicas de las señales de EMG, por último, los filtros IIR como el butterworth tienen un historial probado de uso exitoso en la eliminación de ruido y artefactos en las señales de EMG lo que respalda su elección para este propósito, la capacidad de seleccionar frecuencias, eficiencia computacional y el historial de éxito hacen que los filtros IIR sean una opción adecuada para procesar señales de EMG en este proyecto.

d. Analizar y discutir los resultados del punto anterior de manera que para los puntos siguientes se usen, debidamente sustentado, los índices obtenidos con la señal filtrada o la señal sin filtrar.

En términos generales se observa que los valores de RMS, MAV y varianza son mayores cuando se utiliza la señal TKEO sin filtrar en comparación con la señal TKEO filtrada, se podría decir que la señal sin filtrar retiene más información y características de la señal original, lo cual podría ser beneficioso para la tarea de reconocimiento de gestos, nos parece importante tener en cuenta que la señal sin filtrar también puede contener ruido y artefactos no deseados que podrían afectar negativamente el rendimiento de los modelos de aprendizaje automático, en cuanto a las gráficas de MAV se pudo evidenciar un comportamiento similar de los valores registrados, por otro lado las gráficas de varianza muestran una mayor fluctuación en los valores para ambas señales en general, lo que nos sugiere puede ser que la característica en específico puede ser más sensible al ruido y a las variaciones individuales de los sujetos como es de esperarse.

Respecto a las gráficas de caja y bigotes (boxplots) la idea consiste en permitir la comparación visualmente de las distribuciones de las características extraídas (RMS, Varianza, MAV, MAVS, SSC, WL y ZC) para cada gesto, tanto para la señal TKEO filtrada como para la señal TKEO sin filtrar, en general se observa que las medianas de cada característica son similares entre la señal filtrada y sin filtrar para un mismo gesto, entonces esta información nos estaría sugiriendo que a pesar del filtrado las características centrales de cada distribución no se ven significativamente tan afectadas, sin embargo se aprecian diferencias notables en el rango y la dispersión de los valores para algunas características con la señal filtrada indicando una mayor variabilidad en los valores de estas características cuando no se aplica el filtro, es importante mencionar que las observaciones pueden variar mucho dependiendo del gesto lo que significa que el impacto del filtrado en las características puede ser diferente para cada gesto y tenemos que evaluar todos los gestos y situaciones muy superficialmente por medio de la simple observación de las gráficas.

f. Consultar, justificar el uso e implementar otros modelos de machine learning que permitan la clasificación de gestos.

El **modelo de Random Forests** es una técnica de machine learning que se destaca por su capacidad para realizar tareas de clasificación con alta precisión y robustez, las ventajas clave del modelo incluyen su capacidad para manejar conjuntos de datos grandes con alta dimensionalidad, su resistencia al sobreajuste, y su capacidad para manejar tanto variables numéricas como categóricas, además, este modelo es relativamente fácil de entrenar y ajustar, lo que lo hace ideal para aplicaciones donde se requiere un rendimiento sólido sin una configuración compleja. En cuanto a los requisitos de datos, se necesita una cantidad suficiente de datos y deben ser de alta calidad, libres de ruido y con etiquetas precisas para garantizar un entrenamiento efectivo del modelo, este modelo en términos de recursos computacionales no requiere una potencia computacional extrema y puede ejecutarse eficientemente en un computador estándar, para implementar el modelo se pueden utilizar bibliotecas como scikit-learn en python, que ofrece una implementación eficiente y fácil de usar de este algoritmo, el proceso de preprocesamiento de datos puede incluir la normalización de características, eliminación de valores atípicos y codificación de variables categóricas, según sea necesario para optimizar el rendimiento del modelo.

Para la arquitectura del modelo no se tienen capas ni neuronas en el sentido tradicional de las redes neuronales, en cambio se compone de múltiples árboles de decisión que trabajan de forma conjunta para realizar la clasificación. Cada árbol se entrena de forma independiente en subconjuntos aleatorios de datos y los resultados individuales se combinan mediante un proceso de votación mayoritaria o promedio para obtener el resultado final del modelo, se podría decir que la técnica de ensamblaje de múltiples árboles de decisión proporciona una mayor precisión y robustez en comparación con un solo árbol de decisión y en el contexto de clasificación de gestos de la mano, este modelo podría ser adecuado debido a su capacidad para manejar múltiples características y patrones complejos presentes en los datos de gestos [1].

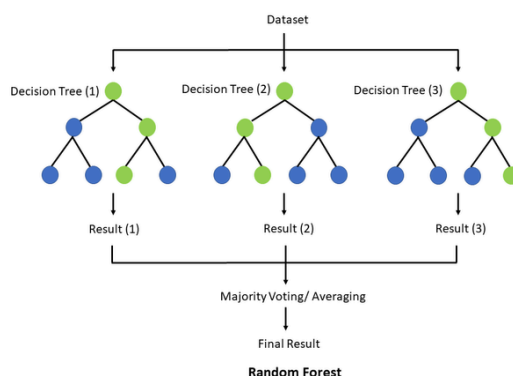


Figura 1. Grafica funcionamiento Random Forest

La **regresión logística** es un modelo ampliamente utilizado en el machine learning para predecir la probabilidad de que una variable dependiente pertenezca a una categoría específica, para este modelo se necesitan una cantidad suficiente de datos etiquetados que representen una variedad de gestos de la mano y la calidad de los datos es crucial para evitar sesgos y garantizar la precisión del modelo, para implementar el modelo, se pueden utilizar bibliotecas como scikit-learn en python y en cuanto al preprocesamiento de datos para el modelo podría incluir la normalización de características, la codificación de variables categóricas y la división del conjunto de datos en conjuntos de entrenamiento y prueba. En cuanto a la arquitectura del modelo, la regresión logística se basa en una función logística para estimar las probabilidades de pertenencia a una clase, la curva de regresión logística se ajusta a los datos de entrenamiento mediante la estimación de los coeficientes de regresión, lo que permite predecir la probabilidad de pertenencia a una clase en función de los valores de las variables independientes, la curva se obtiene a partir de la función logística, que transforma la salida de la regresión lineal en una probabilidad entre 0 y 1. Teniendo en cuenta que se utiliza comúnmente para problemas de clasificación binaria, donde la variable dependiente tiene dos categorías o clases distintas, por lo tanto en el contexto de clasificar gestos de la mano, la

regresión logística no sería el modelo de machine learning mas adecuado debido a la necesidad de para manejar variables categóricas más de dos gestos de mano, se debe de explorar variantes de la regresión logística o modelos de clasificación multiclase [2].

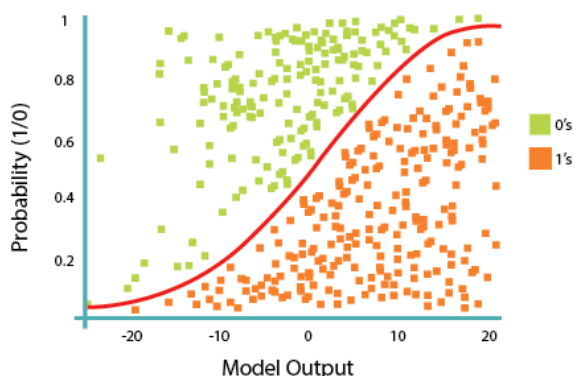


Figura 2. Grafica funcionamiento Regresión logística

El **modelo de k-NN (k-Nearest Neighbors)** es un algoritmo de machine learning que se utiliza para clasificar nuevas muestras basándose en la similitud con las muestras de entrenamiento, cuando se recibe una nueva muestra, el modelo compara esta muestra con todas las muestras existentes y selecciona las 'k' muestras más cercanas, luego asigna la clase mayoritaria de estas 'k' muestras a la nueva muestra, este modelo es de aprendizaje basado en instancias ya que no hay una fase de entrenamiento explícita sino que almacena todas las instancias de entrenamiento y realiza la clasificación en tiempo de consulta, sus ventajas serian la capacidad para trabajar con datos numéricos y nominales sin hacer suposiciones sobre la distribución de los datos, sin embargo hay que tener en cuenta que tiene un alto costo computacional debido a la necesidad de calcular las distancias para cada nueva muestra y su gran requerimiento de memoria para almacenar todas las muestras de entrenamiento, para el modelo la cantidad de datos necesaria depende del problema específico, pero generalmente se requieren muchas muestras, es crucial que los datos de entrenamiento y de prueba provengan de la misma distribución.

Para implementar k-NN, se pueden utilizar bibliotecas como NumPy para cálculos numéricos, SciPy para funciones científicas y estadísticas, y scikit-learn, que proporciona una implementación eficiente y fácil de usar del algoritmo k-NN, es importante manejar los datos faltantes y limpiar cualquier ruido que pueda afectar la precisión del modelo, la arquitectura del modelo se basa en el cálculo de distancias las funciones de distancia más comunes utilizadas son la distancia euclidiana, manhattan y de Minkowski, se selecciona la más adecuada según las características específicas del problema, este modelo podría ser adecuado para estudios de clasificación de gestos de la mano debido a su facilidad de implementación [3].

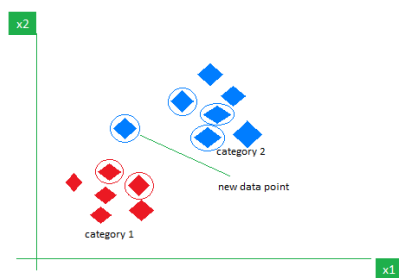


Figura 3. Grafica funcionamiento k-Nearest Neighbors

g. Discutir cómo se podría implementar la solución del punto e y f usando sistemas embebidos seleccionando el tipo de microcontrolador y los posibles tiempos de adquisición y clasificación de las señales

Para el caso del microcontrolador seleccionado debe tener suficiente potencia de procesamiento para ejecutar el algoritmo de clasificación de gestos en tiempo real, algunos podrían ser ARM Cortex-M series (ofrece un buen rendimiento con memoria RAM y Flash, interfaces de comunicación), también microcontroladores de señal digital (DSP) (optimizados para el procesamiento de señales y pueden ser una opción adecuada para aplicaciones que involucran señales EMG)

Para adquirir las señales EMG, se necesitará un circuito de acondicionamiento de señal que amplifica y filtra las señales antes de digitalizarlas. el circuito puede ser externo o interno en el microcontrolador si cuenta con conversores analógicos ADC, el tiempo de adquisición de las señales EMG dependerá de la frecuencia de muestreo requerida, sería de 1000 para capturar adecuadamente las características de las señales EMG, el tiempo de adquisición de la señal serían cada 1 segundos, realizamos un promedio de las muestras adquiridas por los valores de ADC, la fase de procesamiento y clasificación de las señales implican varios pasos, como el filtrado, la extracción de características y la clasificación utilizando el modelo de aprendizaje automático entrenado, es importante tener en cuenta que el procesamiento y la clasificación de señales en tiempo real implican restricciones de tiempo estrictas. Por lo tanto, es crucial optimizar el código y seleccionar cuidadosamente los algoritmos y las técnicas de procesamiento de señales para garantizar un rendimiento adecuado.

Bibliografía

[1] Breiman, L. Random Forests. En Machine Learning (p. 587-604). Springer. (2001) Obtenido de:

<https://www.math.mcgill.ca/yyang/resources/doc/randomforest.pdf>

[2] Marina Domínguez Martín. "Regresión Logística y Técnicas de Aprendizaje. Aplicaciones." Trabajo de Fin de Grado, Universidad de Zaragoza, 2021, pp. 1-58. Obtenido de:

<https://zaguan.unizar.es/record/110300/files/TAZ-TFG-2021-3094.pdf>

[3] Rahul Vishwakarma, Jyothish Kumar J. KNN: K-nearest Neighbours. School of Computer Sciences, National Institute of Science Education and Research, Bhubaneswar, Homi Bhabha National Institute. 2023. Obtenido de:

<https://www.niser.ac.in/~smishra/teach/cs460/23cs460/lectures/lec13.pdf>