SHORT PAPER FOR ECG SIGNALS PROJECT

Claudio Hernández
Universidad de Talca, Chile
Electivo: Machine Learning

Abstract

El propósito de este trabajo es aplicar técnicas de machine learning para el análisis y clasificación de señales ECG en la detección de arritmias. El modelo propuesto se entrena utilizando un **Random Forest** optimizado con **GridSearch** para encontrar los mejores hiperparámetros. Se realiza un análisis exhaustivo de las métricas de rendimiento, como precisión, recall y F1-score, para evaluar el desempeño del modelo en los datos de prueba. Se destacan también las etapas de preprocesamiento, tales como la estandarización de datos y la reducción de dimensionalidad mediante PCA.

Creación del Dataset Curado

El proceso comienza con la descarga de los datos desde Google Drive mediante el uso de **gdown**. Tras la descarga, se descomprime el archivo ZIP y se organizan los archivos de señales ECG en una estructura de carpetas adecuada. Los datos consisten en archivos de formato .hea y .mat, que contienen las señales de ECG y sus metadatos.

Transformación a Problema Binario

Posteriormente, el dataset se transforma en un problema binario, clasificando las señales como "Normal" o "Anormal" basándose en las etiquetas proporcionadas en los metadatos de los archivos. Inicialmente existen 4 tipos: **Normal, AFib, Other o Noisy**, se optó por transformar Other y AFib a una sola clase y eliminar los datos ruidosos. Este paso es crucial para aplicar técnicas de clasificación, donde las señales anormales se etiquetan según corresponda.

Desbalance de Clases

Luego de modificar el dataset, se observa un problema de desbalance de clases en el mismo, donde la clase "Normal" (N) es significativamente más grande que la clase "Anormal". Para abordar este problema y generar un dataset balanceado, se utiliza **submuestreo**. Esta técnica implica eliminar datos de la clase mayoritaria (en este caso, la clase "Normal") para equilibrar las clases y evitar que el modelo favorezca la clase más representada. El resultado es un conjunto de datos balanceado, adecuado para entrenar el modelo de machine learning sin que la distribución de clases

afecte negativamente la precisión del modelo. En este caso particular se eliminaron cerca de 1800 registros de “Normal” para dejar un conjunto de datos de 3000 por cada clase aproximadamente.

Eliminación de Ruido y Extracción de Características

Para eliminar el ruido en las señales ECG y mejorar la precisión en la extracción de características, se incorpora un filtro **bandpass** que elimina las frecuencias de baja y alta frecuencia. Esto mejora la detección de los picos R, lo cual es crucial para obtener intervalos RR más precisos. La eliminación del ruido reduce los errores causados por artefactos y permite obtener resultados más fieles sobre la variabilidad del ritmo cardíaco. Las características extraídas incluyen **registro**, **fs** (frecuencia de muestreo), **duracion_seq** (duración de la secuencia), **media_mv** (media de voltaje), **mstd_mV** (desviación estándar del voltaje), **skewness** (sesgo), **kurtosis**, **rr_mean_s** (media de los intervalos RR), y **rr_std_s** (desviación estándar de los intervalos RR). Estas características son fundamentales para el posterior análisis y entrenamiento del modelo de machine learning.

Inspección del Dataset

Al analizar el dataset, se observa que las características extraídas del conjunto de datos no presentan valores nulos, lo que garantiza la fiabilidad de los datos. La duración de la secuencia muestra variaciones que reflejan la longitud de las señales de los registros, mientras que la **media_mv** y **mstd_mV** presentan una dispersión significativa, indicando diferencias en la amplitud de las señales de los latidos. La asimetría de las señales, medida a través de **skewness** y **kurtosis**, muestra una mayor variabilidad en los registros **Anormales** (afectados por arritmias como la fibrilación auricular) en comparación con los registros **Normales**. Estas métricas indican que las señales de fibrilación auricular tienen una distribución más sesgada y colas más pronunciadas, lo que es consistente con la fisiología de la arritmia, donde los intervalos de latidos son irregulares.

Por otro lado, la matriz de correlación revela que varias características están relacionadas entre sí, particularmente aquellas que miden la variabilidad del ritmo, como **rr_mean_s** y **rr_std_s**. Además, la relación entre **skewness** y **kurtosis** indica que las señales con una distribución sesgada tienden a mostrar colas más pronunciadas, lo cual es un patrón común en las señales anormales, como las de **AFib**. Las correlaciones bajas entre otras variables, como **duracion_seq**, sugieren que la duración de la señal no afecta significativamente las características de la variabilidad del ritmo o la forma de la distribución.

Conclusión Etapa 1

El análisis de las distribuciones de los estadísticos derivados de los intervalos RR, en particular **rr_mean_s** y **rr_std_s**, muestra una clara diferenciación entre

las clases "Normal" y "Anormal". Los registros **Normales** tienden a presentar intervalos **RR** más largos, lo que está asociado con una frecuencia cardíaca más baja y estable. En contraste, los registros **Anormales** presentan intervalos **RR** más irregulares y más cortos, lo que refleja mayor variabilidad en los intervalos de los latidos, característica típica de arritmias. Estas diferencias se evidencian también en las métricas de **skewness** y **kurtosis**, que muestran que las señales **Anormales** tienen distribuciones más sesgadas y con colas más pronunciadas, lo que es consistente con ritmos irregulares o anómalos.

En cuanto a las variables que más diferencian las clases, **rr_std_s** es la que más destaca, ya que muestra una dispersión más amplia en los intervalos **RR** para las señales **Anormales**, lo cual es característico de la irregularidad del ritmo cardíaco. También se observa que **skewness** y **kurtosis** son útiles para identificar señales **Anormales**, ya que las distribuciones sesgadas y con colas más pronunciadas reflejan la mayor irregularidad en los latidos.

Preprocesamiento: Estandarización, Shuffle y División del Dataset

En esta etapa, se aplica la **estandarización** de las características del dataset para que cada una de ellas tenga una media de 0 y una desviación estándar de 1. Esto es crucial para asegurar que todas las características sean comparables y que el modelo no sea influenciado desproporcionadamente por características con escalas mayores. Para ello, se utiliza la clase **StandardScaler** de la librería sklearn, que ajusta las características de acuerdo con estas normas y luego transforma los datos. Además se eliminan algunas dimensiones como el id, registro y fs, ya que no aportan información para resolver el problema.

Tras la estandarización, se realiza un **shuffle** (mezcla aleatoria) de los datos. Esto ayuda a evitar cualquier sesgo que pueda existir debido al orden en que se presentan las muestras, garantizando que el modelo no se vea influenciado por secuencias o patrones no deseados en los datos.

Finalmente, el dataset se divide en tres conjuntos: **entrenamiento (60%)**, **validación (20%)**, y **prueba (20%)**. Primero, se divide el dataset en un conjunto de entrenamiento y un conjunto temporal (40%). Luego, el conjunto temporal se divide a su vez en los conjuntos de validación y prueba, manteniendo la misma distribución de las clases mediante la opción **stratify**. Esto asegura que las proporciones de las clases se mantengan consistentes en todos los conjuntos.

Este proceso de estandarización, shuffle y partición en conjuntos es esencial para garantizar que el modelo de machine learning se entrene de manera efectiva y generalice bien en datos nuevos.

Creación del Modelo y Ajuste de Hiperparámetros con GridSearch

Para esta etapa, se crea un modelo de **Random Forest** utilizando el **RandomForestClassifier** de sklearn. Random Forest es un modelo de ensamble basado en la combinación de múltiples árboles de decisión, lo que mejora la precisión y reduce el riesgo de sobreajuste. Este modelo es especialmente útil en tareas de clasificación debido a su capacidad para manejar datos con relaciones no lineales, su robustez ante valores atípicos y su eficiencia al manejar grandes volúmenes de datos. Además, Random Forest puede manejar tanto variables categóricas como continuas, lo que lo convierte en una opción versátil.

A continuación, se utiliza **GridSearchCV** para ajustar los hiperparámetros del modelo. **GridSearchCV** realiza una búsqueda exhaustiva sobre un conjunto de parámetros definidos previamente y selecciona la combinación que mejor optimice el rendimiento del modelo. En este caso, los hiperparámetros que se ajustan son:

1. **n_estimators**: El número de árboles en el bosque. Se prueba con valores de 100, 200 y 300.
2. **max_depth**: La profundidad máxima de cada árbol. Se prueba con **None** (sin límite) y valores de 10 y 20.
3. **min_samples_split**: El número mínimo de muestras requeridas para dividir un nodo. Se prueba con 2, 5 y 10.
4. **min_samples_leaf**: El número mínimo de muestras requeridas para ser una hoja. Se prueba con 1, 2 y 4.

Este proceso de ajuste busca la mejor configuración de hiperparámetros para mejorar la precisión del modelo de clasificación, dicha configuración resulta ser: {'max_depth':

1. **max_depth:** 20
2. **min_samples_leaf:** 2
3. **min_samples_split:** 10
4. **n_estimators:** 100

Testeo y Evaluación del Modelo

Una vez entrenado el modelo con los mejores hiperparámetros encontrados mediante **GridSearch**, se evalúa su rendimiento en el conjunto de **prueba**. La predicción del modelo sobre los datos de test genera un reporte de clasificación que incluye métricas como **precision**, **recall**, **f1-score** y **support** para cada clase (Normal y Anormal). Estos resultados proporcionan una visión más detallada de cómo el modelo clasifica cada clase.

En el **reporte de clasificación** se observa lo siguiente:

1. La **precisión** para la clase **Anormal** es de 0.72, lo que significa que el modelo tiene una buena capacidad para identificar correctamente los casos anormales, aunque aún hay margen de mejora.

2. El **recall** para **Anormal** es de 0.69, lo que indica que el modelo está perdiendo algunas instancias de esta clase.
3. Por otro lado, la clase **Normal** muestra un **recall** ligeramente superior (0.72), lo que sugiere que el modelo tiene una mejor capacidad para detectar los casos normales.

El **F1-score** para ambas clases es de 0.71, lo que refleja un balance entre la precisión y el recall. Sin embargo, se observa un número significativo de **falsos positivos** y **falsos negativos**, lo cual indica que el modelo podría mejorar.

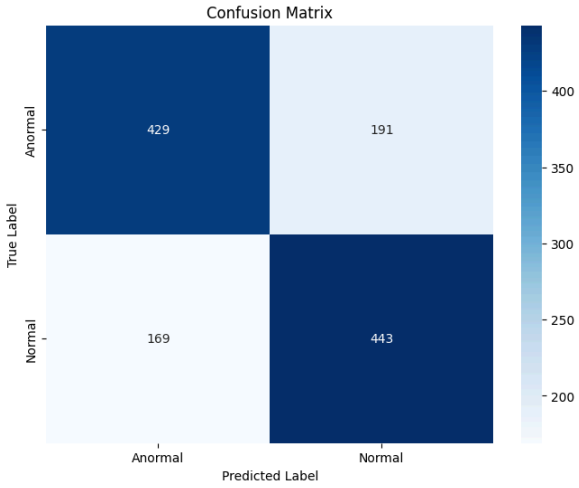

Fig. 1 Confusion Matrix

Conclusión

El **Random Forest** tiene un rendimiento razonable, con un **accuracy** del 71% en el conjunto de prueba. Aunque este valor es aceptable, los valores de precisión y recall sugieren que el modelo podría mejorar, especialmente en la detección de la clase **Anormal**. El ajuste de los hiperparámetros o una mejor gestión del desbalance de clases podrían ayudar a mejorar los resultados.

El **análisis de la matriz de confusión** también confirma que el modelo presenta algunos errores en la clasificación, ya que hay un número considerable de instancias de la clase **Anormal** que son mal clasificadas como **Normales**, y viceversa. Esto podría ser un área clave de mejora al seguir ajustando el modelo o aplicar técnicas de re-muestreo para balancear las clases.

En resumen, el modelo proporciona una clasificación decente, pero existen áreas de oportunidad para mejorar su desempeño ajustando parámetros o mejorando el preprocesamiento de datos.

Propuestas de Mejora

Para mejorar el rendimiento del modelo, se puede aumentar el número de estimadores (**n_estimators**) en el Random Forest, lo que aumentará su precisión y estabilidad. Además, aplicar **PCA** (Análisis de Componentes Principales) podría reducir la dimensionalidad del conjunto de datos, eliminando redundancias y ruido, lo que haría el modelo más eficiente y podría mejorar su rendimiento general, especialmente si algunas características no aportan valor significativo.