UNIVERSIDAD NACIONAL AGRARIA LA MOLINA

MODELO PREDICTIVO DE ENFERMEDADES CARDÍACAS

Integrantes del grupo:

Díaz Paniagua, Luis David - 20211813
Gómez Soto, Gustavo Alejandro - 20211816
Vivanco Bartolo, Jimena de los Angeles - 20211834

Docente: Meza Rodríguez, Aldo Richard

Lima - 2025

Contents

1 Introducción

Las enfermedades cardiovasculares representan una de las principales causas de muerte a nivel mundial. Gracias a los avances en ciencia de datos y aprendizaje automático, hoy es posible desarrollar modelos predictivos capaces de identificar patrones de riesgo a partir de datos clínicos.

El presente trabajo tiene como objetivo desarrollar un modelo de **clasificación supervisada** para predecir la probabilidad de que un paciente presente una enfermedad cardíaca. Se emplearon distintos algoritmos de Machine Learning, evaluando su desempeño para seleccionar el modelo más eficaz. Finalmente, se construyó una **interfaz interactiva en Streamlit** que permite al usuario ingresar valores clínicos y obtener una predicción inmediata.

2 Descripción del conjunto de datos

El dataset empleado proviene de un repositorio público de salud cardiovascular. Contiene registros de pacientes con distintas características clínicas, tanto numéricas como categóricas.

2.1 Variables principales

- **Edad:** edad del paciente (años)
- **Sexo:** M = Masculino, F = Femenino
- **ChestPainType:** tipo de dolor torácico (TA: típica, ATA: atípica, NAP: no anginoso, ASY: asintomático)
- **Presión arterial en reposo:** mm Hg
- **Colesterol:** colesterol sérico (mg/dl)
- **Glucemia en ayunas:** 1 = glucemia ¿ 120 mg/dl, 0 = normal
- **ECG en reposo:** resultados del electrocardiograma (Normal, ST, HVI)
- **MaxHR:** frecuencia cardíaca máxima alcanzada
- **Angina de ejercicio:** S = Sí, N = No
- **Oldpeak:** depresión del segmento ST

- **ST_Slope:** pendiente del ST (Up, Flat, Down)
- **Enfermedad cardíaca:** variable objetivo (1 = enfermedad, 0 = normal)

3 Análisis Exploratorio de Datos (EDA)

Se realizó un análisis exploratorio para conocer la distribución de las variables y la presencia de posibles correlaciones.

- Se verificó la ausencia de valores nulos.
- Se transformaron variables categóricas mediante codificación *one-hot*.
- Se observó que los pacientes con mayor edad, colesterol y presión arterial presentaban una mayor prevalencia de enfermedad cardíaca.

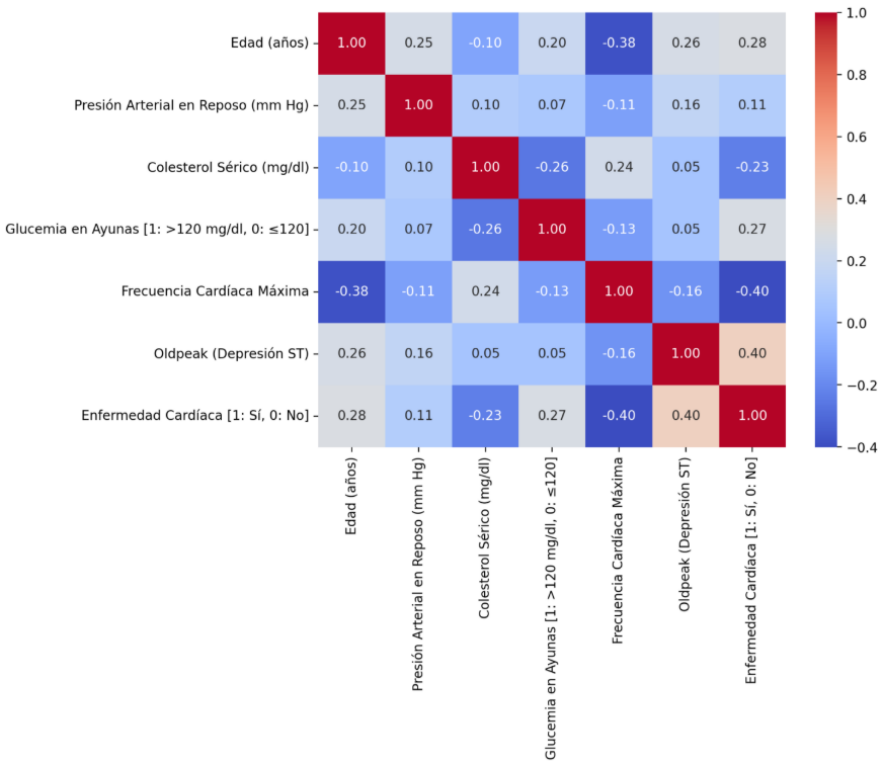

Figure 1: Mapa de correlación entre variables numéricas.

4 Modelado y Entrenamiento

Se probaron distintos algoritmos de clasificación: Naive Bayes, Regresión Logística, Red Neuronal y Random Forest. El dataset se dividió en un 80% para entrenamiento y un 20% para prueba. Las variables fueron escaladas mediante `StandardScaler`.

Listing 1: Entrenamiento y evaluación de modelos

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, f1_score
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.neural_network import MLPClassifier

# Divisi n de datos
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# Escalado
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Modelos
models = {
    "Naive Bayes": GaussianNB(),
    "Regresi n Log stica": LogisticRegression(max_iter=200)
        ,
    "Red Neuronal": MLPClassifier(max_iter=1000),
    "Random Forest": RandomForestClassifier(n_estimators=200)
}

results = []
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    results.append({
        "Modelo": name,
        "Accuracy": accuracy_score(y_test, y_pred),
        "F1-Score": f1_score(y_test, y_pred)
    })
```

4.1 Comparación de desempeño

Modelo	Accuracy	F1-Score
Naive Bayes	0.859	0.874
Regresión Logística	0.853	0.870
Red Neuronal	0.880	0.896
Random Forest	0.880	0.894

Table 1: Métricas de desempeño por modelo.

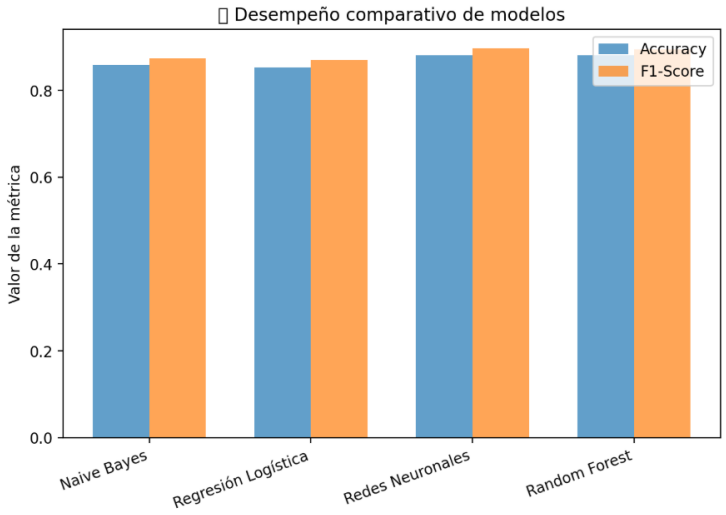

Figure 2: Comparación gráfica del Accuracy y F1-score por modelo.

5 Selección del modelo final

Tras analizar los resultados, se observó que tanto la **Red Neuronal** como el **Random Forest** obtuvieron los mejores desempeños. Sin embargo, el Random Forest presentó un menor tiempo de entrenamiento y mayor interpretabilidad, por lo que se eligió como **modelo final implementado en la interfaz**.

6 Interfaz Interactiva

Se desarrolló una interfaz con **Streamlit**, que permite al usuario ingresar los valores de las variables clínicas y obtener la predicción del modelo en tiempo real.

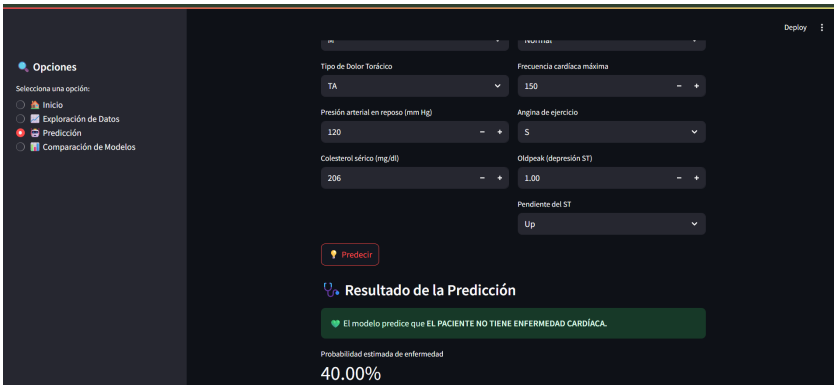

Figure 3: Vista general de la interfaz desarrollada en Streamlit.

Listing 2: Fragmento del código de la interfaz en Streamlit

```
import streamlit as st
import joblib

model = joblib.load("modelo_random_forest.pkl")

st.title("       Predicci n de Enfermedades Card acas")
edad = st.slider("Edad del paciente", 20, 80, 50)
colesterol = st.number_input("Colesterol (mg/dl)", 100, 600,
    200)
presion = st.number_input("Presi n arterial en reposo", 80,
    200, 120)

if st.button("Predecir"):
    pred = model.predict([[edad, presion, colesterol]])
    if pred[0] == 1:
        st.error("       Riesgo de enfermedad card aca
            detectado")
    else:
        st.success("       Sin indicios de enfermedad
            card aca")
```

7 Conclusiones

- El modelo Random Forest logró un **accuracy del 88%**, siendo el mejor equilibrio entre precisión e interpretabilidad.
- La interfaz desarrollada permite un uso intuitivo y rápido, adecuada para aplicaciones de apoyo clínico.

- La metodología empleada puede adaptarse fácilmente a otros problemas de clasificación médica.