import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score, classification\_report

from sklearn.preprocessing import LabelEncoder

# Load the dataset

file\_path = 'path\_to\_tested.csv' # Replace with the actual path to tested.csv

data = pd.read\_csv(file\_path)

# Quick look at the data

print("Dataset Overview:")

print(data.head())

# Handle missing values

data.fillna({'Age': data['Age'].median(), 'Fare': data['Fare'].median()}, inplace=True)

data.drop(columns=['Cabin'], inplace=True) # Dropping Cabin due to high missing rate

# Encode categorical features

encoder = LabelEncoder()

data['Sex'] = encoder.fit\_transform(data['Sex']) # Male: 1, Female: 0

data['Embarked'] = data['Embarked'].fillna('S') # Fill missing with the most common value

data['Embarked'] = encoder.fit\_transform(data['Embarked'])

# Feature selection

features = ['Pclass', 'Sex', 'Age', 'Fare', 'Embarked']

X = data[features]

y = data['Survived']

# Train-test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Model training

model = RandomForestClassifier(n\_estimators=100, random\_state=42)

model.fit(X\_train, y\_train)

# Model evaluation

y\_pred = model.predict(X\_test)

accuracy = accuracy\_score(y\_test, y\_pred)

print(f"Accuracy: {accuracy:.2f}")

print("Classification Report:")

print(classification\_report(y\_test, y\_pred))

# Save the model and encoder

import joblib

joblib.dump(model, 'titanic\_model.pkl')

joblib.dump(encoder, 'label\_encoder.pkl')

# Instructions for GitHub

print("""

Steps for GitHub submission:

1. Organize your code and save it in a script (e.g., `titanic\_prediction.py`).

2. Include a README.md outlining:

- Your approach

- Preprocessing and feature engineering steps

- Challenges faced

- Model performance

3. Push the files (`titanic\_model.pkl`, `label\_encoder.pkl`, and the script) to your GitHub repository.

""")