# Telco Churn Prediction - Python Script

import pandas as pd  
import numpy as np  
from sklearn.model\_selection import train\_test\_split  
from sklearn.preprocessing import LabelEncoder, StandardScaler  
from sklearn.ensemble import RandomForestClassifier  
from sklearn.metrics import classification\_report, confusion\_matrix, accuracy\_score  
  
df = pd.read\_csv("telco.csv")  
df.dropna(inplace=True)  
df.drop("customerID", axis=1, inplace=True)  
df["TotalCharges"] = pd.to\_numeric(df["TotalCharges"], errors='coerce')  
df.dropna(subset=["TotalCharges"], inplace=True)  
  
le = LabelEncoder()  
for column in df.select\_dtypes(include=["object"]).columns:  
 if column != 'Churn':  
 df[column] = le.fit\_transform(df[column])  
  
df['Churn'] = df['Churn'].map({'Yes': 1, 'No': 0})  
  
X = df.drop("Churn", axis=1)  
y = df["Churn"]  
  
scaler = StandardScaler()  
X\_scaled = scaler.fit\_transform(X)  
  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_scaled, y, test\_size=0.2, random\_state=42)  
  
model = RandomForestClassifier(random\_state=42)  
model.fit(X\_train, y\_train)  
  
y\_pred = model.predict(X\_test)  
  
print("Accuracy Score:", accuracy\_score(y\_test, y\_pred))  
print("\nClassification Report:\n", classification\_report(y\_test, y\_pred))  
print("\nConfusion Matrix:\n", confusion\_matrix(y\_test, y\_pred))