Bank Customer Churn Prediction Using Machine Learning

Objective :

The goal of this project is to predict whether a bank customer is likely to leave (churn) or stay, using historical customer data. By identifying potential churners, the bank can implement targeted strategies to retain these customers.

Import Library :

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
import matplotlib.pyplot as plt
import seaborn as sns
```

Import Data :

```
# Load the dataset
url = 'https://raw.githubusercontent.com/datasciencedojo/datasets/master/telco-customer-churn.csv'
data = pd.read_csv(url)
data.head()
```

Describe Data :

```
df.head()
data.info()
data.describe()
data.isnull().sum()
```

Data Visualization :

```python
sns.countplot(x='Churn', data=data)
plt.title('Churn Distribution')
plt.show()
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(), annot=False, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

Data Preprocessing :

```python
data = data.drop(['customerID'], axis=1)
data['TotalCharges'] = pd.to_numeric(data['TotalCharges'], errors='coerce')
data['TotalCharges'].fillna(data['TotalCharges'].median(), inplace=True)
label_encoder = LabelEncoder()
for column in data.select_dtypes(include=['object']).columns:
    data[column] = label_encoder.fit_transform(data[column])
```

Define Target Variable (y) and Feature Variables (X) :

```python
X = data.drop('Churn', axis=1)
y = data['Churn']
```

Train Test Split :

```python
# Split the dataset into training and test sets (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Modelling :

```python
classifier = RandomForestClassifier(n_estimators=100, random_state=42)
classifier.fit(X_train, y_train)
```

Model Evaluation :

```python
y_pred = classifier.predict(X_test)
print('Confusion Matrix:\n', confusion_matrix(y_test, y_pred))
```

```
print('\nClassification Report:\n', classification_report(y_test, y_pred))
print('\nAccuracy Score:', accuracy_score(y_test, y_pred))
```

Prediction :

```
sample_data = X_test.iloc[0].values.reshape(1, -1)
predicted_churn = classifier.predict(sample_data)
print('Predicted Churn:', 'Yes' if predicted_churn[0] == 1 else 'No')
```

Explanation :

The model uses features like monthly charges, tenure, payment method, and more to determine the likelihood of a customer churning. Feature importance analysis reveals which factors contribute the most to customer churn, helping the bank make data-driven decisions to improve customer retention.

This structured approach helps banks to predict customer churn accurately and proactively engage with high-risk customers, ultimately enhancing customer satisfaction and reducing turnover rates.