# HW4_Part4_Henry_Romero

March 10, 2025

[81]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler, LabelEncoder
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, confusion_matrix,␣
↪classification_report

# Load the bank dataset
file_path = "~/Downloads/bank.csv" # I wanted a challenge and this one really␣
↪was, but is going to help me with my work!

bank_df = pd.read_csv(file_path, delimiter=';')  #delimiter solution!


# Encode categorical variables using LabelEncoder from sklearn
label_encoders = {}
categorical_cols = ['job', 'marital', 'education', 'default', 'housing',␣
↪'loan', 'contact', 'month', 'poutcome', 'y']

for col in categorical_cols:
    le = LabelEncoder()
    bank_df[col] = le.fit_transform(bank_df[col])
    label_encoders[col] = le  # Store encoder

# Normalize numerical features using Min-Max Scaling
scaler = MinMaxScaler()
numerical_cols = ['age', 'balance', 'day', 'duration', 'campaign', 'pdays',␣
↪'previous']
bank_df[numerical_cols] = scaler.fit_transform(bank_df[numerical_cols])


# correlation matrix
corr_matrix = bank_df.corr()
print("\nCorrelation Coefficient Matrix:")
```

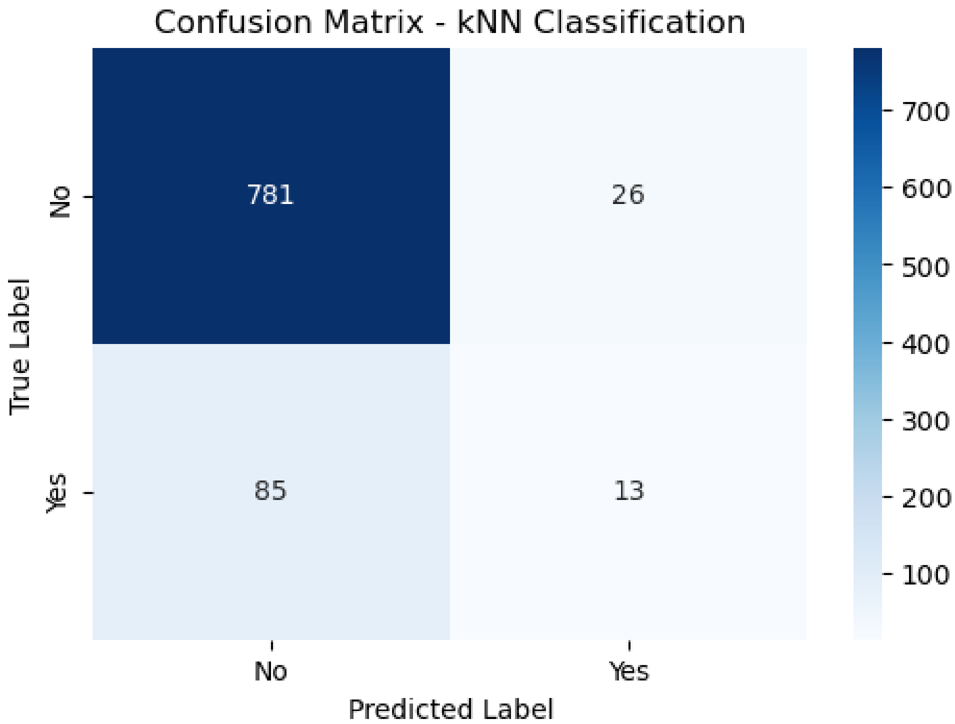
```python
print(corr_matrix)

# Split the dataset into features (X) and target variable (y)
X = bank_df.drop(columns=['y'])  # Features
y = bank_df['y']  # Target variable (0 = no, 1 = yes)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
↪random_state=42) # Split into training (80%) and testing (20%)

# Train the kNN model with k=5
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(X_train, y_train)

# Make predictions
y_pred = knn.predict(X_test)
print("Prediction =", y_pred)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

# Print evaluation metrics
print(f"Model Accuracy: {accuracy:.4f}")
print("\nConfusion Matrix:\n", conf_matrix)
print("\nClassification Report:\n", classification_rep)

# Plot the Confusion Matrix
plt.figure(figsize=(6, 4))
sns.heatmap(conf_matrix, annot=True, fmt="d", cmap="Blues", xticklabels=["No",␣
↪"Yes"], yticklabels=["No", "Yes"])
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.title("Confusion Matrix - kNN Classification")
plt.show()

##Heat Map
plt.figure(figsize=(10, 6))
sns.heatmap(bank_df.corr(), annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.show()

##scatter plot comparison and easy to find outliers in dataset
plt.figure(figsize=(8, 5))
sns.scatterplot(x=bank_df["age"], y=bank_df["balance"], hue=bank_df["y"],␣
↪palette="coolwarm", alpha=0.7)
plt.xlabel("Age")
plt.ylabel("Balance")
```

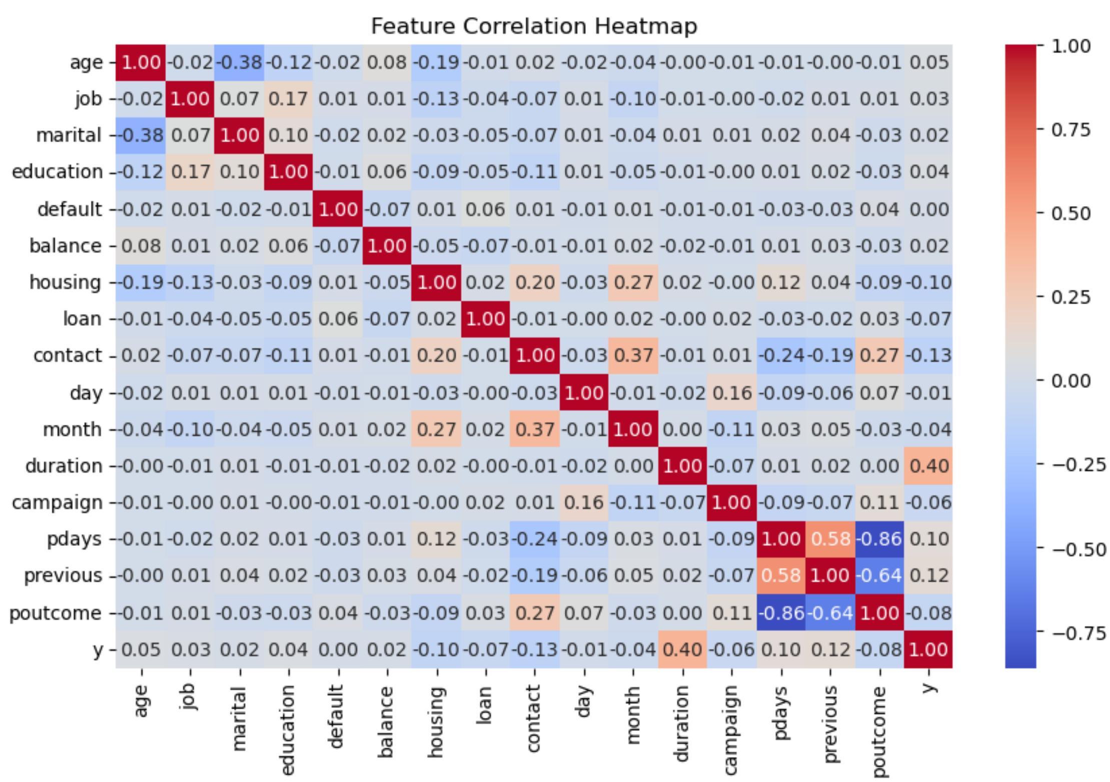
```
plt.title("Age vs. Balance")

plt.show()

##bar graph based on yes and no from the target variable
plt.figure(figsize=(6, 4))
sns.countplot(data=bank_df, x="y", hue="y", palette="pastel", legend=False)
plt.xlabel("Subscription (No = 0, Yes = 1)")
plt.ylabel("Count")
plt.title("Distribution of Subscriptions")
plt.show()


#scatter plot for the most impacted variables with correct predictions on the␣
↪classifier
plt.figure(figsize=(8, 6))
sns.scatterplot(
    x=X_test["pdays"],
    y=X_test["previous"],
    hue=y_pred,  # Predicted churn classification (0 or 1)
    style=y_test,  # Actual labels (different markers)
    palette={0: "red", 1: "orange"},
    edgecolor="black"
)
plt.xlabel("pdays")
plt.ylabel("previous")
plt.title("kNN Classification - pdays vs previous for Churn")

plt.show()
```

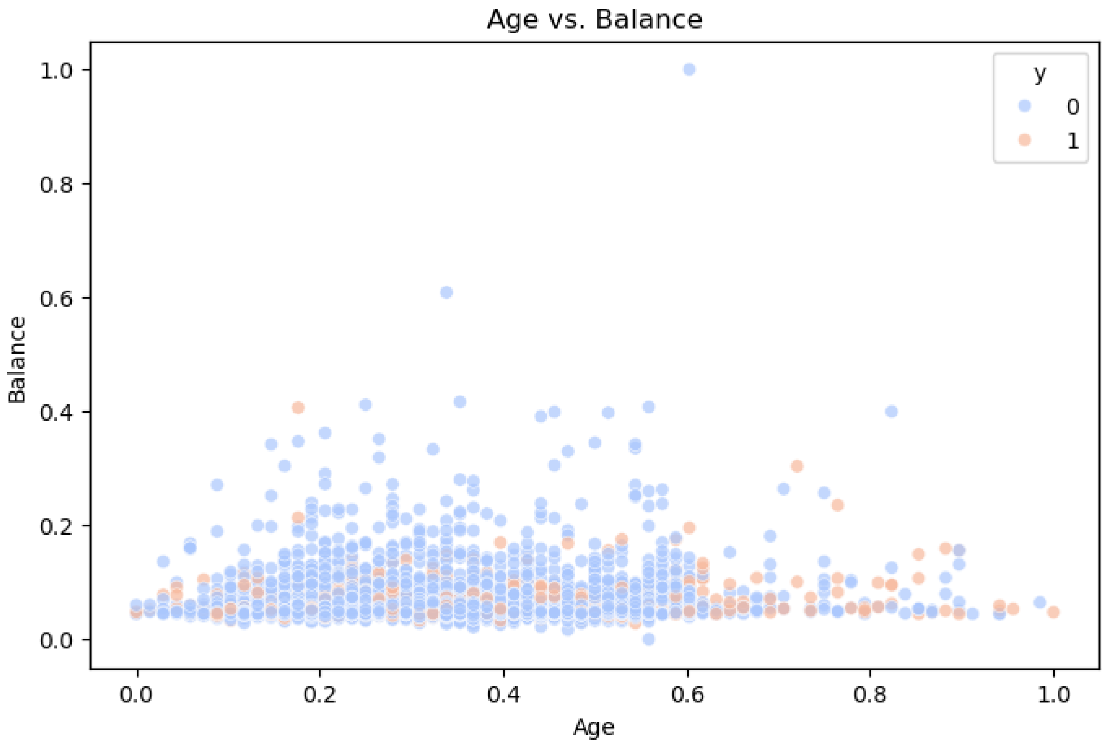
```
Correlation Coefficient Matrix:
                age       job   marital  education   default   balance  \
age        1.000000 -0.021500 -0.381485  -0.121613 -0.017885  0.083820
job       -0.021500  1.000000  0.069390   0.170160  0.008324  0.009797
marital   -0.381485  0.069390  1.000000   0.102714 -0.020745  0.024971
education -0.121613  0.170160  0.102714   1.000000 -0.010534  0.057725
default   -0.017885  0.008324 -0.020745  -0.010534  1.000000 -0.070886
balance    0.083820  0.009797  0.024971   0.057725 -0.070886  1.000000
housing   -0.193888 -0.128353 -0.029851  -0.087070  0.006881 -0.050227
loan      -0.011250 -0.040245 -0.045210  -0.054086  0.063994 -0.071349
contact    0.015161 -0.074068 -0.068236  -0.110554  0.008448 -0.009665
day       -0.017853  0.012865  0.008794   0.014926 -0.013261 -0.008677
month     -0.040714 -0.096613 -0.035855  -0.050086  0.014297  0.023113
duration  -0.002367 -0.006739  0.006619  -0.014878 -0.011615 -0.015950
campaign  -0.005148 -0.002739  0.005915  -0.001723 -0.012348 -0.009976
pdays     -0.008894 -0.022760  0.017050   0.012077 -0.026317  0.009437
previous  -0.003511  0.005029  0.038028   0.023983 -0.026656  0.026196
```

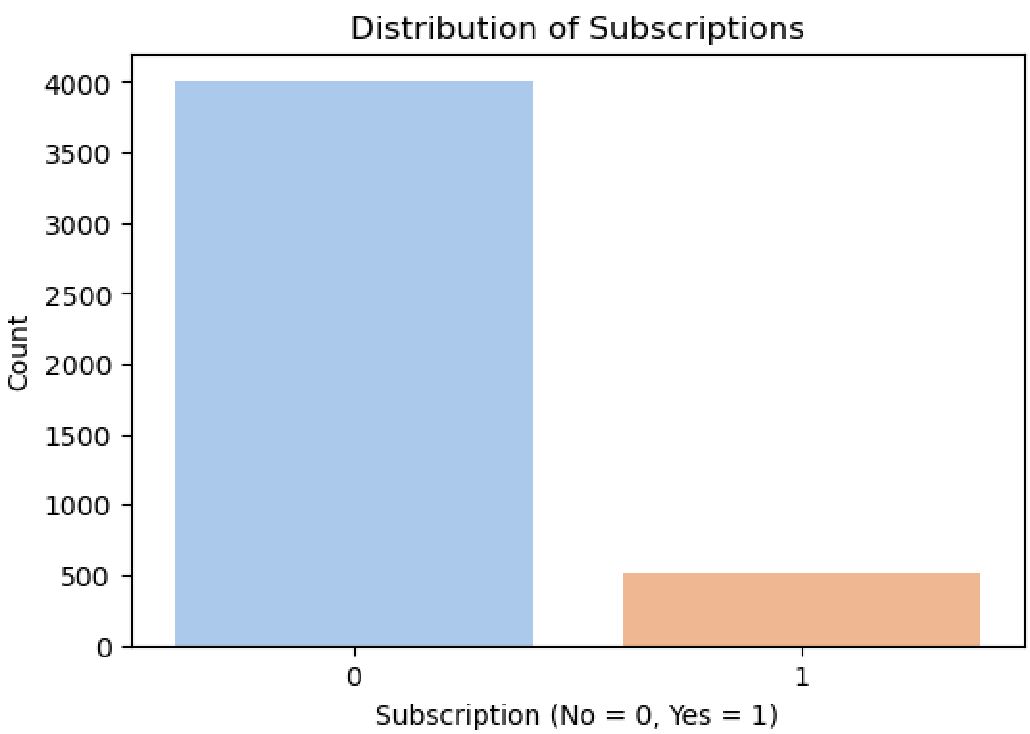
```
poutcome  -0.009320  0.013049 -0.027716  -0.032135  0.039032 -0.029268
y          0.045092  0.027401  0.015042   0.042987  0.001303  0.017905

            housing      loan   contact       day     month  duration  \
age       -0.193888 -0.011250  0.015161 -0.017853 -0.040714 -0.002367
job       -0.128353 -0.040245 -0.074068  0.012865 -0.096613 -0.006739
marital   -0.029851 -0.045210 -0.068236  0.008794 -0.035855  0.006619
education -0.087070 -0.054086 -0.110554  0.014926 -0.050086 -0.014878
default    0.006881  0.063994  0.008448 -0.013261  0.014297 -0.011615
balance   -0.050227 -0.071349 -0.009665 -0.008677  0.023113 -0.015950
housing    1.000000  0.018451  0.196454 -0.031291  0.266630  0.015740
loan       0.018451  1.000000 -0.007319 -0.004879  0.016329 -0.004997
contact    0.196454 -0.007319  1.000000 -0.033807  0.370077 -0.011380
day       -0.031291 -0.004879 -0.033807  1.000000 -0.014795 -0.024629
month      0.266630  0.016329  0.370077 -0.014795  1.000000  0.000851
duration   0.015740 -0.004997 -0.011380 -0.024629  0.000851  1.000000
campaign  -0.003574  0.017120  0.012278  0.160706 -0.108915 -0.068382
pdays      0.116893 -0.031086 -0.243223 -0.094352  0.033292  0.010380
previous   0.038621 -0.022115 -0.187232 -0.059114  0.046899  0.018080
poutcome  -0.093093  0.027028  0.267173  0.073714 -0.030189  0.000478
y         -0.104683 -0.070517 -0.133595 -0.011244 -0.040933  0.401118

           campaign     pdays  previous  poutcome         y
age       -0.005148 -0.008894 -0.003511 -0.009320  0.045092
job       -0.002739 -0.022760  0.005029  0.013049  0.027401
marital    0.005915  0.017050  0.038028 -0.027716  0.015042
education -0.001723  0.012077  0.023983 -0.032135  0.042987
default   -0.012348 -0.026317 -0.026656  0.039032  0.001303
balance   -0.009976  0.009437  0.026196 -0.029268  0.017905
housing   -0.003574  0.116893  0.038621 -0.093093 -0.104683
loan       0.017120 -0.031086 -0.022115  0.027028 -0.070517
contact    0.012278 -0.243223 -0.187232  0.267173 -0.133595
day        0.160706 -0.094352 -0.059114  0.073714 -0.011244
month     -0.108915  0.033292  0.046899 -0.030189 -0.040933
duration  -0.068382  0.010380  0.018080  0.000478  0.401118
campaign   1.000000 -0.093137 -0.067833  0.110703 -0.061147
pdays     -0.093137  1.000000  0.577562 -0.859245  0.104087
previous  -0.067833  0.577562  1.000000 -0.636372  0.116714
poutcome   0.110703 -0.859245 -0.636372  1.000000 -0.082632
y         -0.061147  0.104087  0.116714 -0.082632  1.000000
Prediction = [0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
```

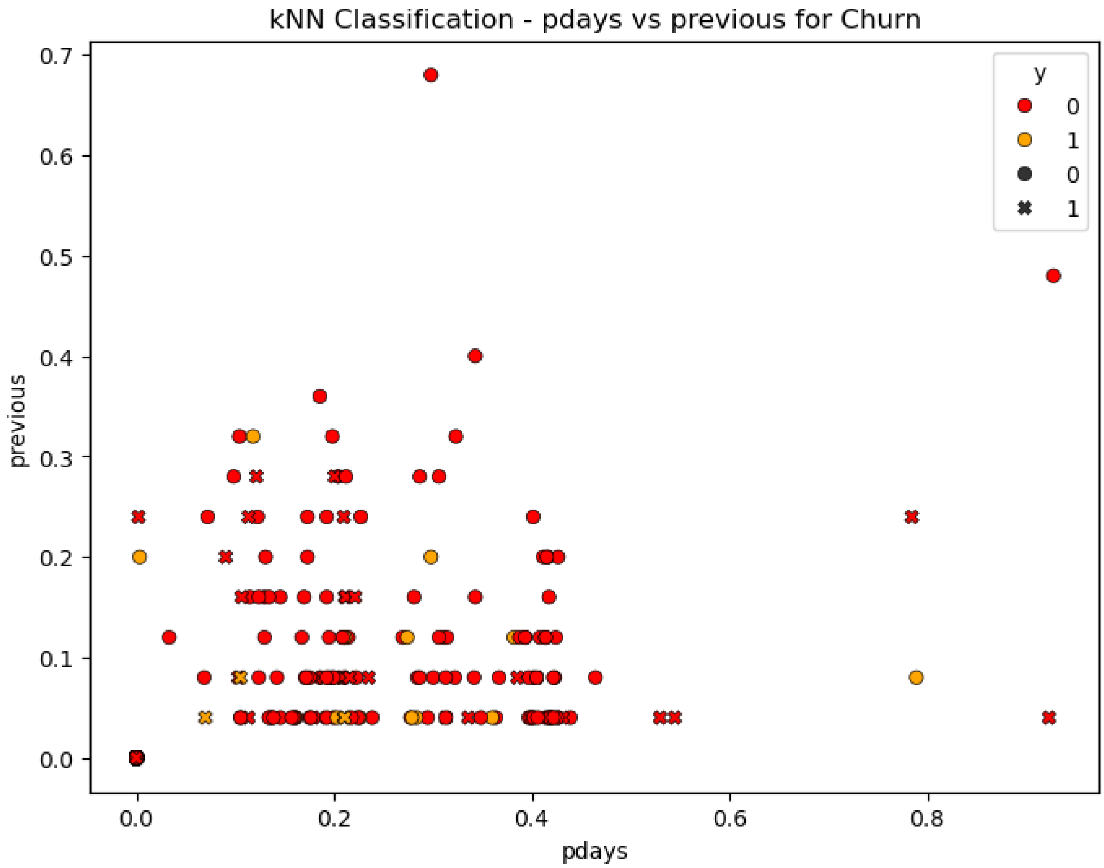
```
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1
 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
Model Accuracy: 0.8773

Confusion Matrix:
 [[781  26]
 [ 85  13]]

Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.97      0.93       807
           1       0.33      0.13      0.19        98

    accuracy                           0.88       905
   macro avg       0.62      0.55      0.56       905
weighted avg       0.84      0.88      0.85       905
```


Confusion Matrix - kNN Classification
True Label
Predicted Label
No
Yes
781
26
85
13
700
600
500
400
300
200
100


Feature Correlation Heatmap
age
job
marital
education
default
balance
housing
loan
contact
day
month
duration
campaign
pdays
previous
poutcome
y


Age vs. Balance
y
0
1
1.0
0.8
0.6
0.4
0.2
0.0
Balance
0.0
0.2
0.4
0.6
0.8
1.0
Age


Distribution of Subscriptions
4000
3500
3000
2500
2000
1500
1000
500
0
Count
0
1
Subscription (No = 0, Yes = 1)


kNN Classification - pdays vs previous for Churn
previous
pdays
y
0
1
0
1
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.0
0.2
0.4
0.6
0.8

```
[ ]:
```