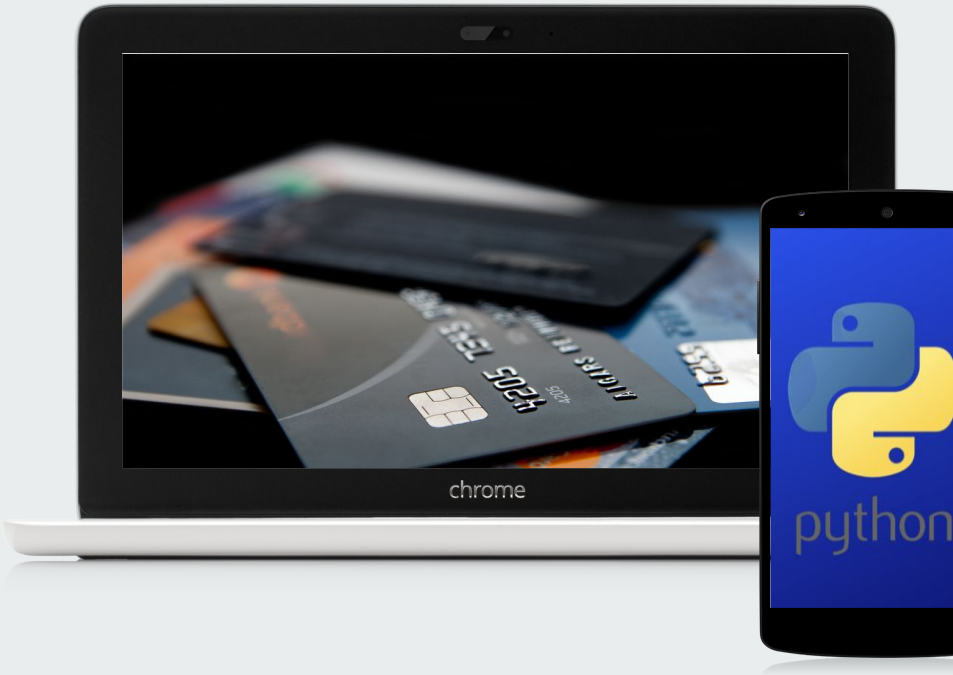
Credit Card Classification Project

Machine Learning



IRON HACK

Table of contents

IRON HACK

Scenario

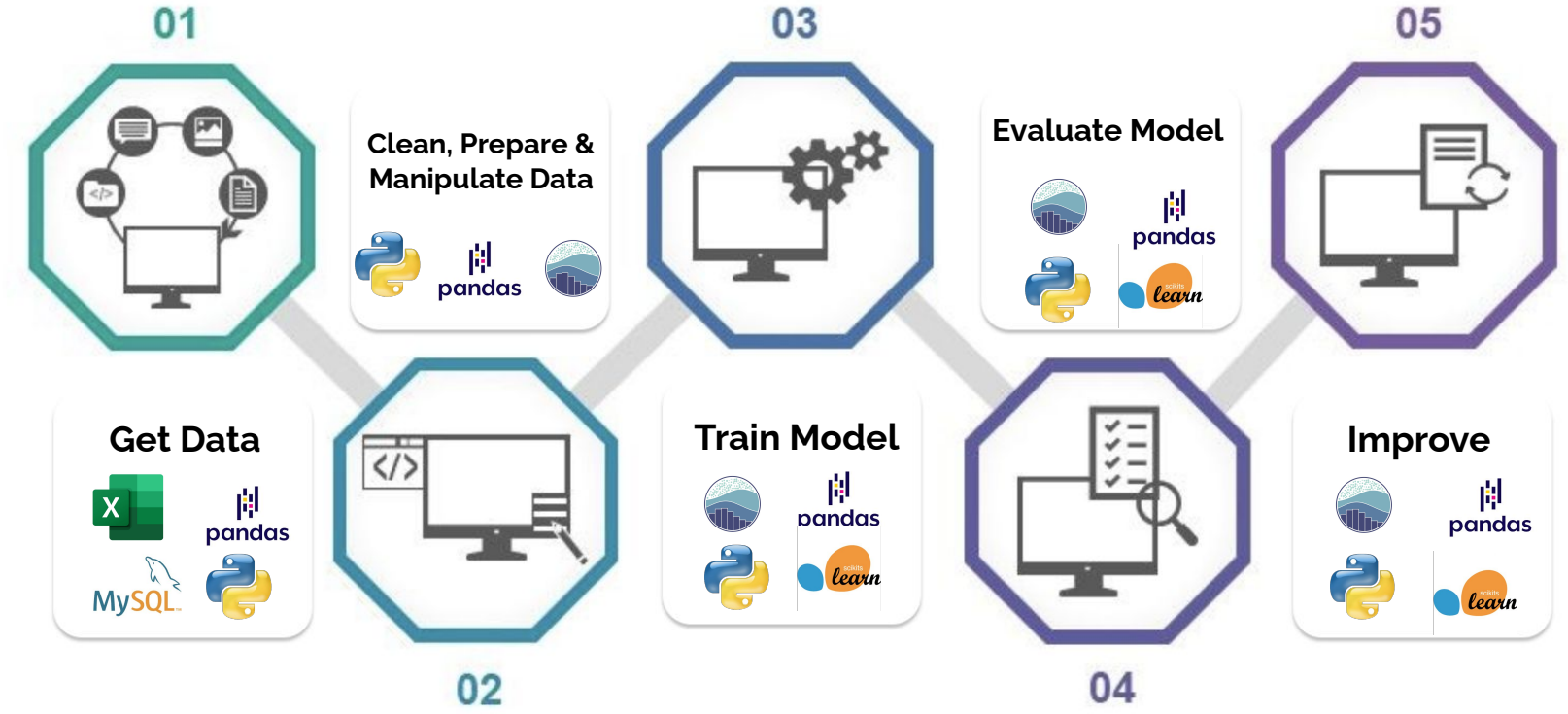
The bank that hired us wants to understand the demographics and other characteristics of its customers that accept or decline a credit card offer.

Based on the result of this study the bank will be able to create tailored marketing campaigns for specific clusters within their customer base.

Luis
Senior Data Analyst

Pol
Data Analyst

Process



IRON HACK

Model Comparison

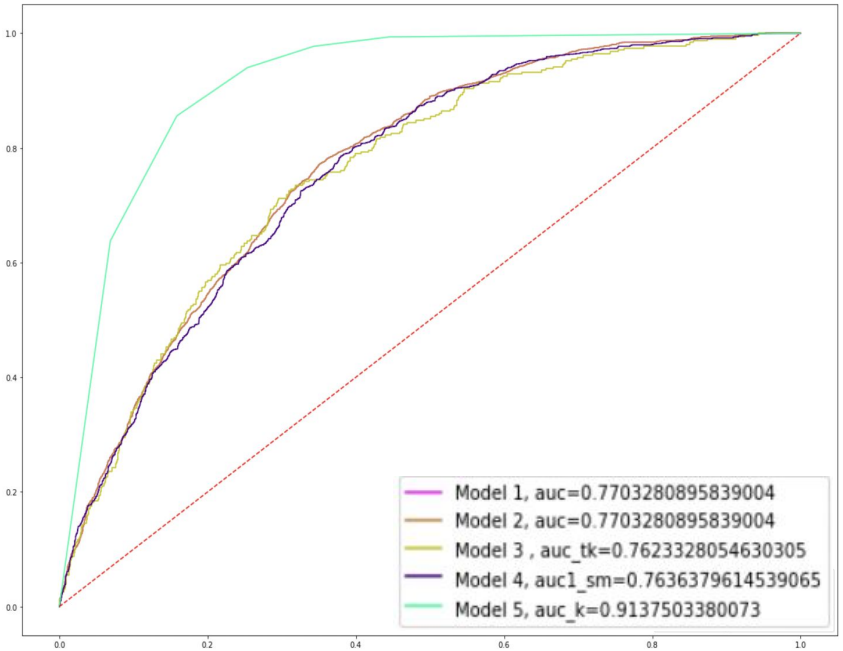
Baseline - SMOTE - TomekLink - Dropping Q Balances - KNN

Accuracy score

- **Model 1:** 0.9415909512330799
- **Model 2:** 0.6994003735377962
- **Model 3:** 0.9409403669724771
- **Model 4:** 0.7027425538189325
- **Model 5:** 0.8471444018480291

TP/TN - True Negatives / True Positives

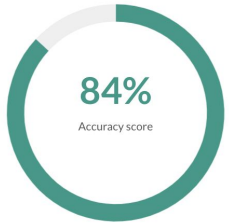
- **Model 1:** *TN:* 94.16% | *TP:* 0.00%
- **Model 2:** *TN:* 33.50% | *TP:* 36.44%
- **Model 3:** *TN:* 94.09% | *TP:* 0.00%
- **Model 4:** *TN:* 33.48% | *TP:* 36.79%
- **Model 5:** *TN:* 36.35% | *TP:* 47.87%

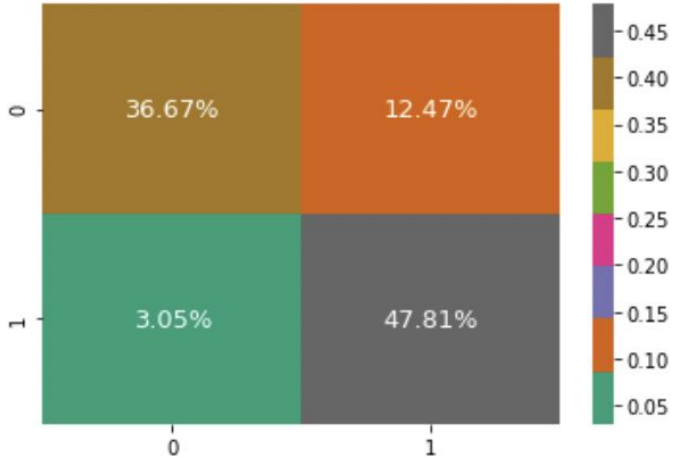


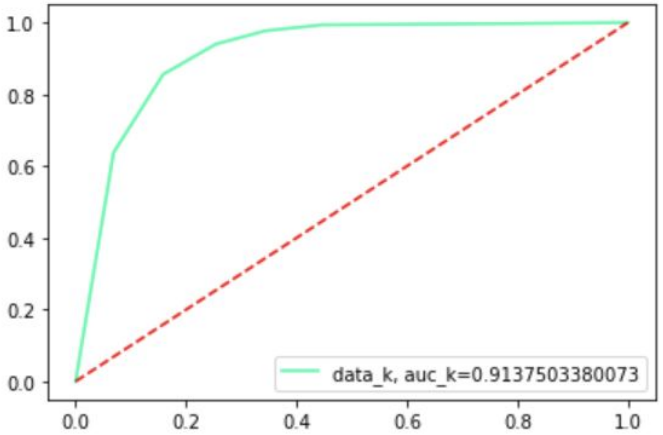
Model Applied

Normalizer - SMOTE - KNN

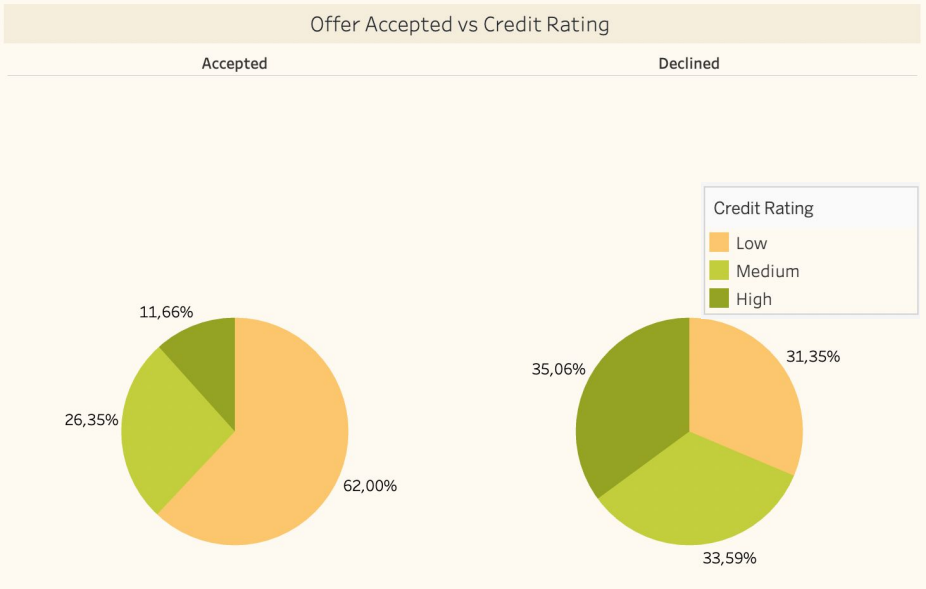
```
## Importing KNeighborsClassifier
from sklearn.neighbors import KNeighborsClassifier

## Defining Knn giving it 5 neighbours
Knn = KNeighborsClassifier (n_neighbors = 5, p = 2)

## Splitting data into train and test set
X1_sm_train, X1_sm_test, y1_sm_train, y1_sm_test =
train_test_split (X1_sm, y1_sm, test_size = 0.3, random_state=40)

## Fitting Knn to the training sets
Knn.fit(X1_sm_train,y1_sm_train)

## Predicting the response for the new smote test dataset using Knn
y_pred_k = Knn.predict(X1_sm_test)

## Obtaining the accuracy of the prediction
accuracy_score(y1_sm_test,y_pred_k)
```

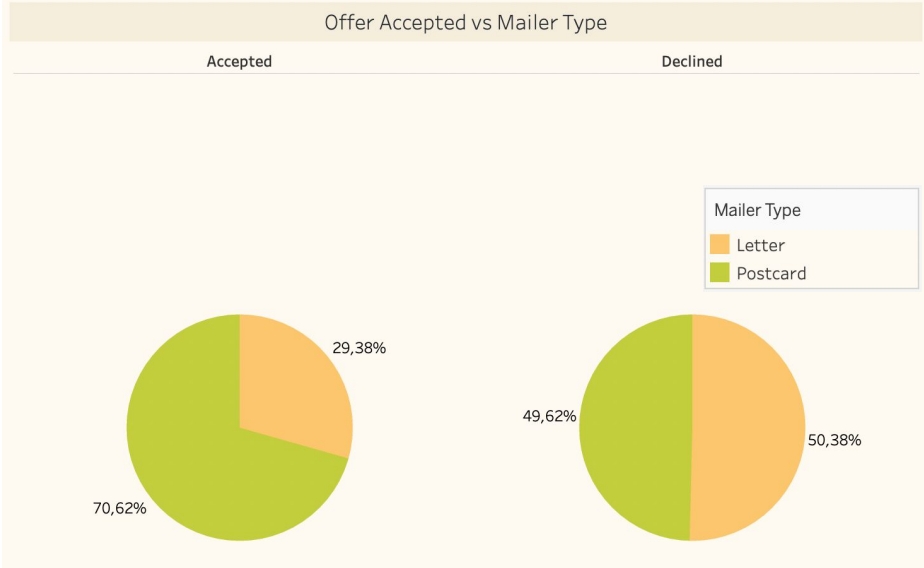




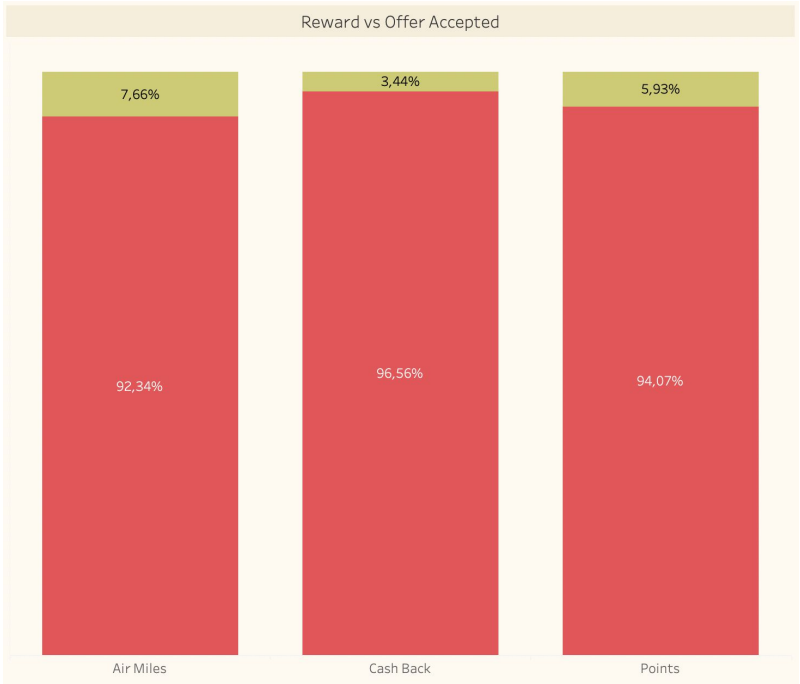


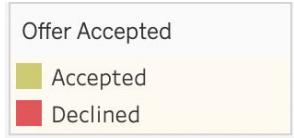
Visualizations





Visualizations





Thank you!

IRON HACK