Churn Customer Prediction Using Feature Selection & Machine Learning Models

Done By:
Yazan Hijazi
Raneem Refaie

Supervisor:
Dr.Qusai Alzoubi

Overview

01

Customer Retention plays a pivotal role in sustaining the growth and success of companies across various industries

02

Churn, the phenomenon of customers discontinuing their association with a business

03

This project aims to leverage machine learning models for the prediction of customer churn and the development of proactive retention strategies

Problem Statement

Problem Solve

Problem Statement

In the ever-changing business landscape, the issue of customer churn is a significant concern. It's not just about losing numbers; it represents losing loyal customers and stable revenue. This challenge becomes even more complex in today's markets with many different customer behaviors. This project looks at how advanced analytics can help with this challenge.

Overview

01

Customer Retention plays a pivotal role in sustaining the growth and success of companies across various industries

02

Churn, the phenomenon of customers discontinuing their association with a business

03

This project aims to leverage machine learning models for the prediction of customer churn and the development of proactive retention strategies

Problem Statement

Problem Solve

Problem Solve

1- By using analytics to understand customer relationships, the project aims to find patterns that predict when customers might leave and get insights to help keep them. The project combines data analysis with smart decision-making to help businesses keep customers happy, make more money, and stay competitive.

2-Evaluate the performance of the predictive models and propose effective customer retention based on the insights gained.

Overview

01

Customer Retention plays a pivotal role in sustaining the growth and success of companies across various industries

02

Churn, the phenomenon of customers discontinuing their association with a business

03

This project aims to leverage machine learning models for the prediction of customer churn and the development of proactive retention strategies

Problem Statement

Problem Solve

Churn Customer Prediction Using Feature Selection & Machine Learning Models

Done By:
Yazan Hijazi
Raneem Refaie

Supervisor:
Dr.Qusai Alzoubi

Methodolgy

- Data Collection and Preprocessing
- Feature Engineering and Data Preparation
- Data Balancing
- Exploratory Data Analysis (EDA) using Power BI
- Feature Selection
- Model Building and Evaluation
- Testing

Codes

Codes

Libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report, roc_auc_score, accuracy_score
from xgboost import XGBClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import MinMaxScaler,StandardScaler
from sklearn.feature_selection import chi2,f_classif,SelectKBest
import imblearn
from collections import Counter
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import cross_val_score,RepeatedStratifiedKFold
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import roc_curve, auc
from sklearn.ensemble import StackingClassifier
```

Replacing missing values

```
data['calc_charges'] = data['Monthly Charges'] * data['Tenure Months']
data['diff_in_charges'] = data['Total Charges'] - data['calc_charges']
data.groupby('Contract')[['Total Charges','diff_in_charges']].quantile([.50,.80,.90,.95])
data['Total Charges'] = np.where(data['Total Charges'].isna() == True,data['calc_charges'], data['Total Charges'])
data = data.drop(['calc_charges','diff_in_charges'], axis=1)
```

Shape

```
data.shape()
output = (7043, 33)
```

Preprocessing

```
from sklearn.preprocessing import MinMaxScaler,StandardScaler
mms = MinMaxScaler() # Normalization
ss = StandardScaler() # Standardization

df1['Tenure Months'] = mms.fit_transform(df1[['Tenure Months']])
df1['Monthly Charges'] = mms.fit_transform(df1[['Monthly Charges']])
df1['Total Charges'] = mms.fit_transform(df1[['Total Charges']])
df1.tail(10)
```

SMOTE

```
over = SMOTE(sampling_strategy = 1)

f1 = df1.iloc[:,:13].values
t1 = df1.iloc[:,13].values

f1, t1 = over.fit_resample(f1, t1)
Counter(t1)
```

Libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report, roc_auc_score, accuracy_score
from xgboost import XGBClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import MinMaxScaler,StandardScaler
from sklearn.feature_selection import chi2,f_classif,SelectKBest
import imblearn
from collections import Counter
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import cross_val_score,RepeatedStratifiedKFold
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import roc_curve, auc
from sklearn.ensemble import StackingClassifier
```

```
data['cal
data['dif
data.grou
data['Tot
data = da
```

Preprocessing

```
.groupby('Contract')[['Total Charges','diff_in_charges']].quantile([.50,.80,.90,.95])
['Total Charges'] = np.where(data['Total Charges'].isna() == True,data['calc_charges'], data['Total Cha
 = data.drop(['calc_charges','diff_in_charges'], axis=1)
```

Shape

```
data.shape()
output = (7043, 33)
```

SMOTE

Replacing missing values

```
data['calc_charges'] = data['Monthly Charges'] * data['Tenure Months']
data['diff_in_charges'] = data['Total Charges'] - data['calc_charges']
data.groupby('Contract')[['Total Charges','diff_in_charges']].quantile([.50,.80,.90,.95])
data['Total Charges'] = np.where(data['Total Charges'].isna() == True,data['calc_charges'], data['Total Charges'])
data = data.drop(['calc_charges','diff_in_charges'], axis=1)
```

Shape

SMOTE

```
over = SMOTE(sampling_strategy = 1)

f1 = df1.iloc[:,:13].values
t1 = df1.iloc[:,13].values

f1, t1 = over.fit_resample(f1, t1)
Counter(t1)
```

Preprocessing

```
from sklearn.preprocessing import MinMaxScaler,StandardScaler
mms = MinMaxScaler() # Normalization
ss = StandardScaler() # Standardization

df1['Tenure Months'] = mms.fit_transform(df1[['Tenure Months']])
df1['Monthly Charges'] = mms.fit_transform(df1[['Monthly Charges']])
df1['Total Charges'] = mms.fit_transform(df1[['Total Charges']])
df1.tail(10)
```

Methodolgy

- Data Collection and Preprocessing
- Feature Engineering and Data Preparation
- Data Balancing
- Exploratory Data Analysis (EDA) using Power BI
- Feature Selection
- Model Building and Evaluation
- Testing

Codes

Codes

Feature Selection

```
#CATEGORICAL
features = df1.loc[:,categorical_features]
target = df1.loc[:,'Churn Label']

best_features = SelectKBest(score_func = chi2,k = 'all')
fit = best_features.fit(features,target)

featureScores = pd.DataFrame(data = fit.scores_,index = list(features.columns),columns = ['Chi Squared Score'])

#NUMERICAL
features = df1.loc[:,numerical_features]
target = df1.loc[:,'Churn Label']

best_features = SelectKBest(score_func = f_classif,k = 'all')
fit = best_features.fit(features,target)

featureScores = pd.DataFrame(data = fit.scores_,index = list(features.columns),columns = ['ANOVA Score'])
```

Stacking

Stacking

```
stack = StackingClassifier(estimators = [('classifier_xgb',classifier_xgb),
                                         ('LR',LR),
                                         ('classifier_rf',classifier_rf),
                                         ('classifier_dt',classifier_dt)],
                           final_estimator = classifier_xgb )
```

Name of the column after feature selection

```
Senior Citizen
Partner
Dependents
Tenure Months
Online Security
Online Backup
Device Protection
Tech Support
Contract
Paperless Billing
Payment Method
Monthly Charges
Total Charges
Churn Label
```

Model Build

Model Build

```
x_train, x_test, y_train, y_test = train_test_split(f1, t1, test_size = 0.20, random_state = 10)
classifier_xgb = XGBClassifier(learning_rate= 0.01,max_depth = 5,n_estimators = 1000)
classifier_rf = RandomForestClassifier(max_depth = 5,random_state = 33)
classifier_dt = DecisionTreeClassifier(random_state = 33,max_depth = 5)
LR = LogisticRegression(random_state = 33)
```

Feature Selection

```
#CATEGORICAL
features = df1.loc[:,categorical_features]
target = df1.loc[:,'Churn Label']

best_features = SelectKBest(score_func = chi2,k = 'all')
fit = best_features.fit(features,target)

featureScores = pd.DataFrame(data = fit.scores_,index = list(features.columns),columns = ['Chi
Squared Score'])

#NUMERICAL
features = df1.loc[:,numerical_features]
target = df1.loc[:,'Churn Label']

best_features = SelectKBest(score_func = f_classif,k = 'all')
fit = best_features.fit(features,target)

featureScores = pd.DataFrame(data = fit.scores_,index = list(features.columns),columns =
['ANOVA Score'])
```

Name of the column after feature selection

```
Senior Citizen
Partner
Dependents
Tenure Months
Online Security
Online Backup
Device Protection
Tech Support
Contract
Paperless Billing
Payment Method
Monthly Charges
Total Charges
Churn Label
```

```
ures.columns),columns =
```

Model Build

Model Build

```
x_train, x_test, y_train, y_test = train_test_split(f1, t1, test_size = 0.20, random_state =
10)
classifier_xgb = XGBClassifier(learning_rate= 0.01,max_depth = 5,n_estimators = 1000)
classifier_rf = RandomForestClassifier(max_depth = 5,random_state = 33)
classifier_dt = DecisionTreeClassifier(random_state = 33,max_depth = 5)
LR = LogisticRegression(random_state = 33)
```

Stacking

Stacking

```
stack = StackingClassifier(estimators = [('classifier_xgb',classifier_xgb),
                                         ('LR',LR),
                                         ('classifier_rf',classifier_rf),
                                         ('classifier_dt',classifier_dt)],
                           final_estimator = classifier_xgb )
```

Methodolgy

- Data Collection and Preprocessing
- Feature Engineering and Data Preparation
- Data Balancing
- Exploratory Data Analysis (EDA) using Power BI
- Feature Selection
- Model Building and Evaluation
- Testing

Codes

Codes

Churn Customer Prediction Using Feature Selection & Machine Learning Models

Done By:
Yazan Hijazi
Raneem Refaie

Supervisor:
Dr.Qusai Alzoubi

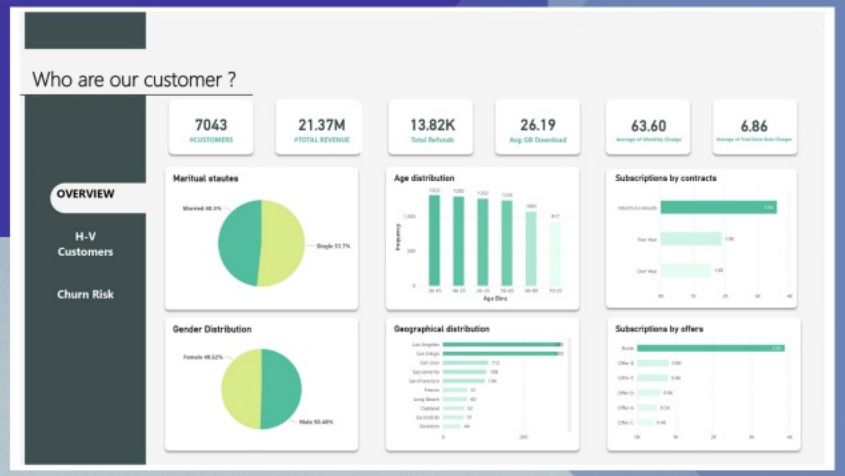
Design and Implementation

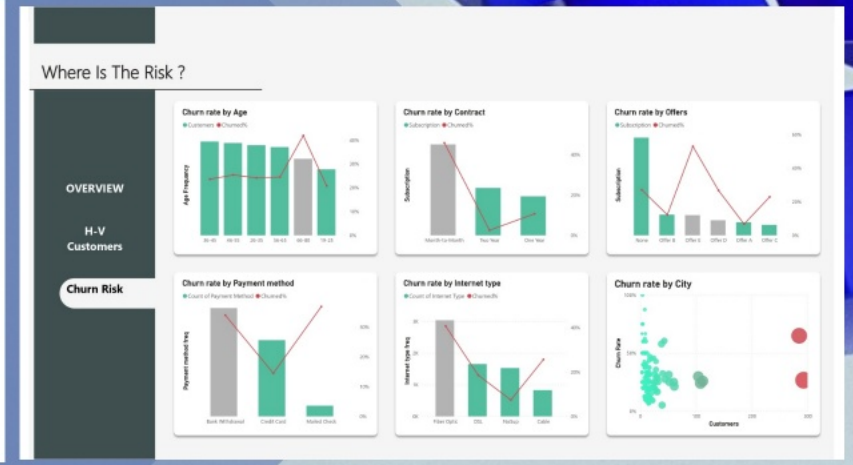
In this section, we provide insights into how we implemented our project, including the technical details of model development and the integration of visualizations.

Power Bi Dashboard

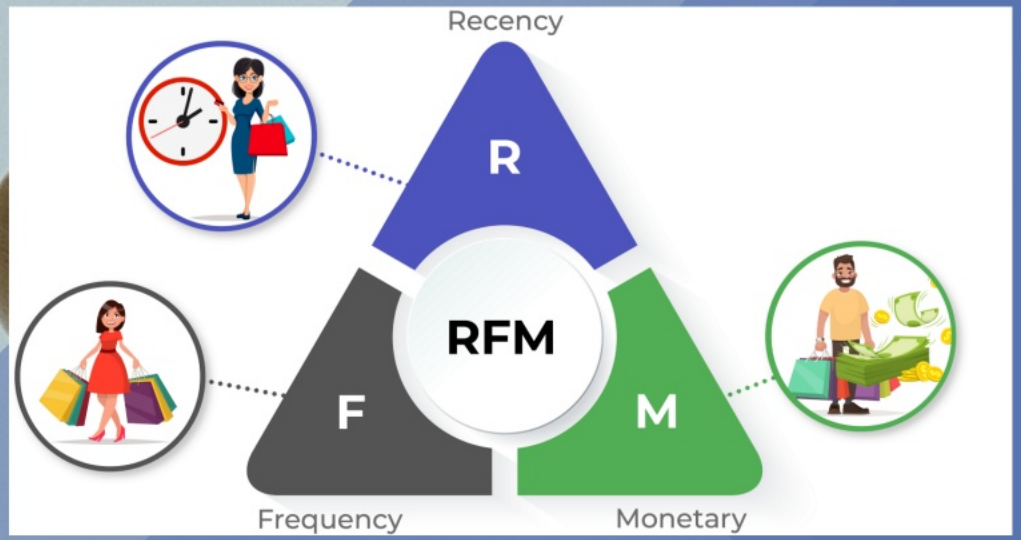
Graphical user interface

RFM





Conclusion



Segment	RFM Score
Champions	555
Champions	554
Champions	545
Churned Lovers	155
Churned Lovers	154
Churned Lovers	145

Dax PowerBi

```
Recency Score = IF(RFM[Customer Status] == "Churned" , "1", "5")
```

```
Freq Score = SWITCH(TRUE(),'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.20),"1",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.40),"2",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.60),"3",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.80),"4","5")
```

```
Monetory Score = SWITCH(TRUE(),'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.20),"1",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.40),"2",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.60),"3",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.80),"4","5")
```

```
RFM Score = RFM[Recency Score]&RFM[Freq Score]&RFM[Monetory Score]
```

Recency

R

RFM

F

M

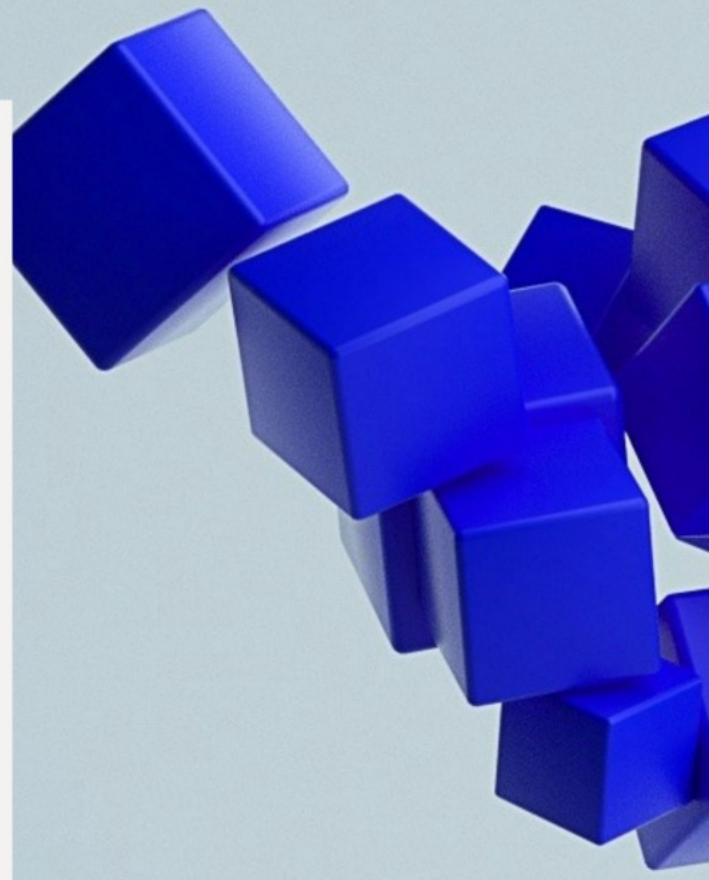
Frequency

Monetary

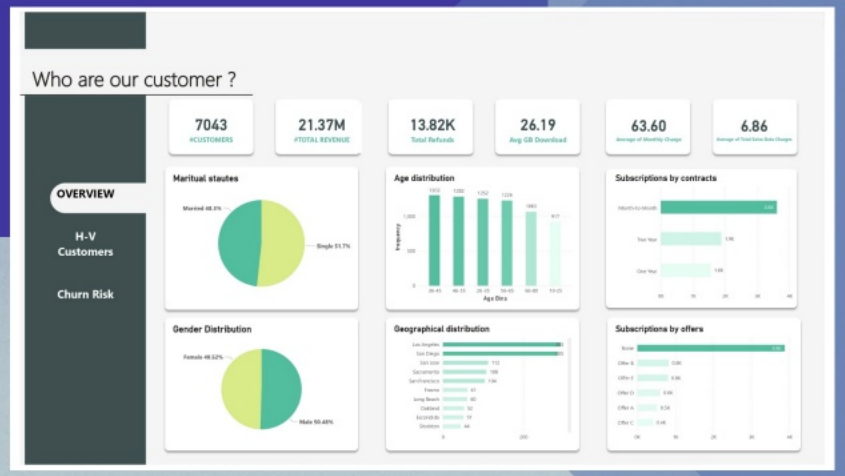
Dax PowerBi

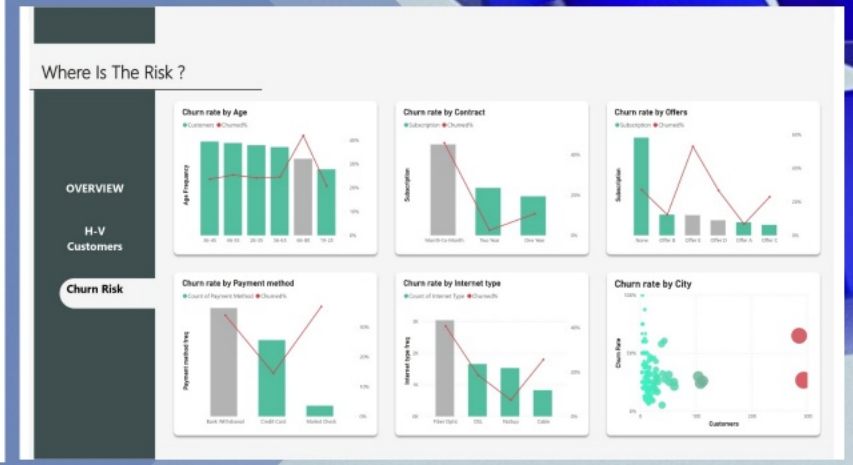
```
Recency Score = IF(RFM[Customer Status] == "Churned" , "1", "5")
-----------------------------------------------------------------------------------------
Freq Score = SWITCH(TRUE(),'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.20),"1",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.40),"2",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.60),"3",
'RFM'[Tenure in Months]<=PERCENTILE.INC(RFM[Tenure in Months],0.80),"4","5")
-----------------------------------------------------------------------------------------
Monetory Score = SWITCH(TRUE(),'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.20),"1",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.40),"2",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.60),"3",
'RFM'[Total Revenue]<=PERCENTILE.INC(RFM[Total Revenue],0.80),"4","5")
-----------------------------------------------------------------------------------------
RFM Score = RFM[Recency Score]&RFM[Freq Score]&RFM[Monetory Score]
```

Segment	RFM Score
Champions	555
Champions	554
Champions	545
Churned Lovers	155
Churned Lovers	154
Churned Lovers	145



RFM





Conclusion

Who are our customer ?

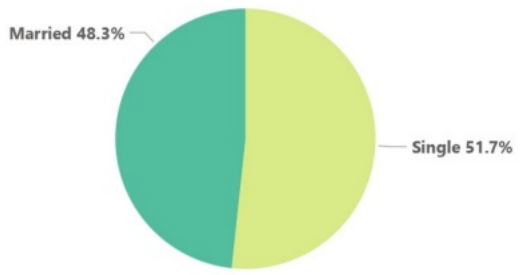
OVERVIEW

H-V Customers

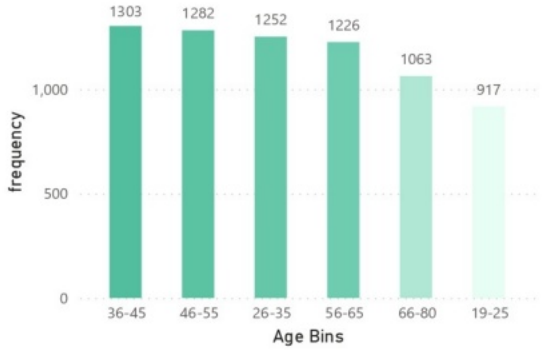
Churn Risk

7043	21.37M	13.82K	26.19	63.60	6.86
#CUSTOMERS	#TOTAL REVENUE	Total Refunds	Avg GB Download	Average of Monthly Charge	Average of Total Extra Data Charges

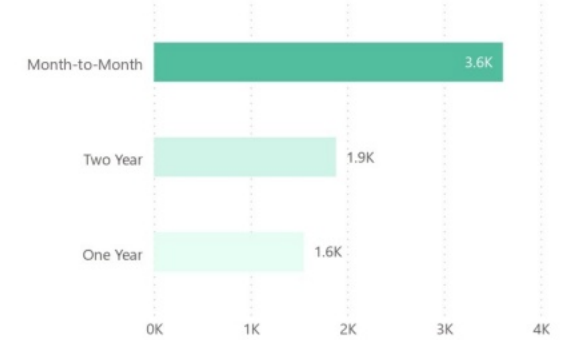
Maritual stautes



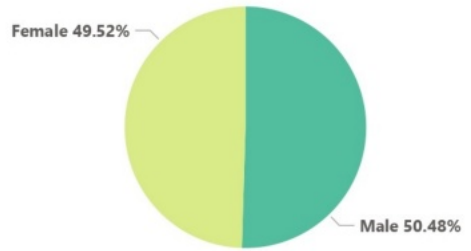
Age distribution



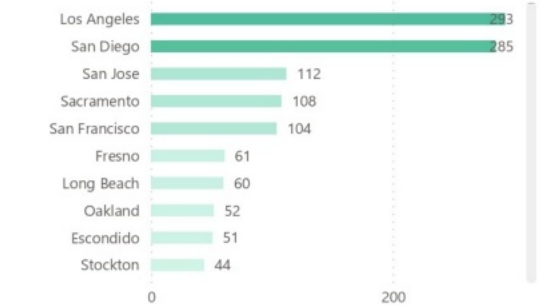
Subscriptions by contracts



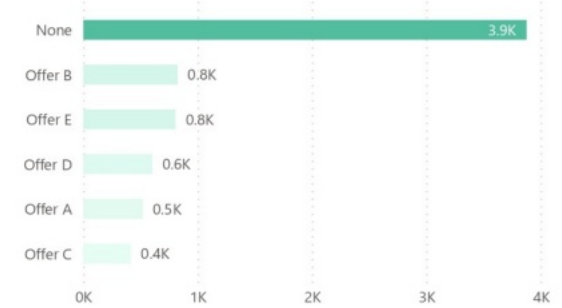
Gender Distribution



Geographical distribution



Subscriptions by offers

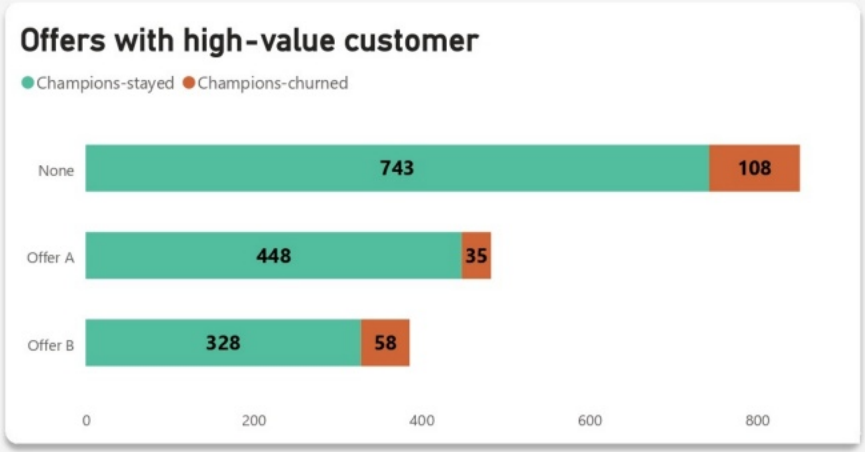


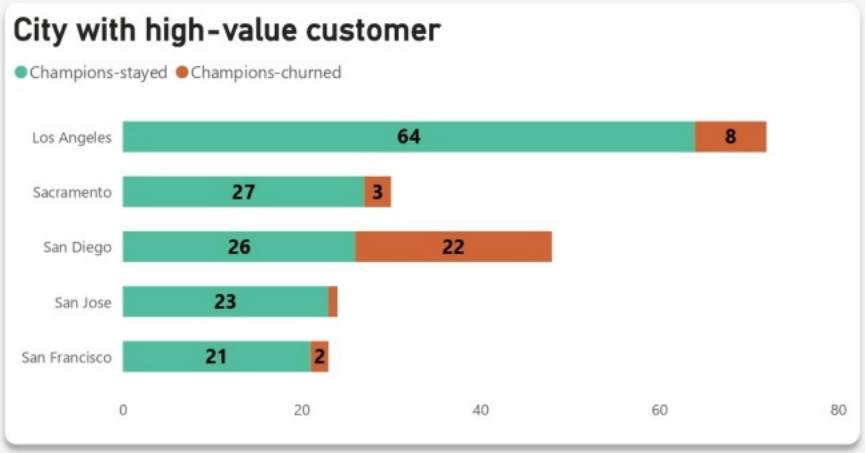
Where Are High-Value Customer?

OVERVIEW

H-V Customers

Churn Risk



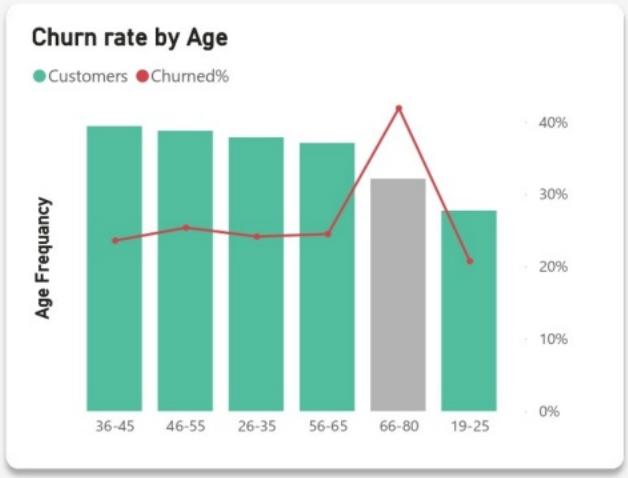


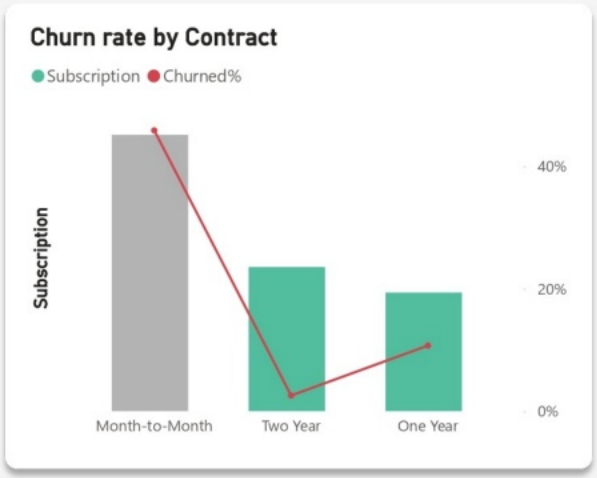
Where Is The Risk ?

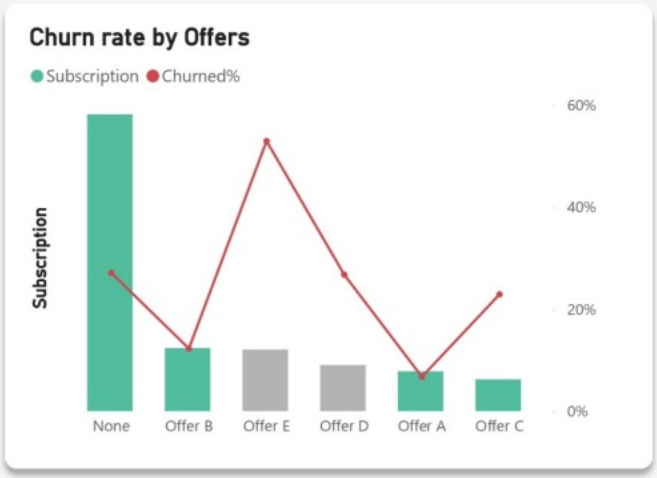
OVERVIEW

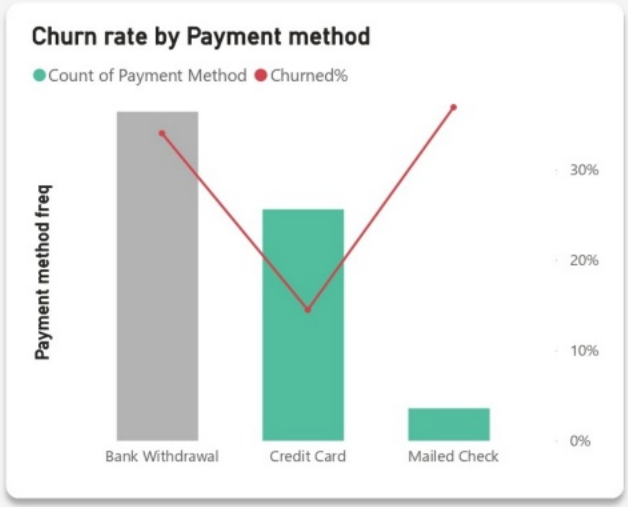
H-V Customers

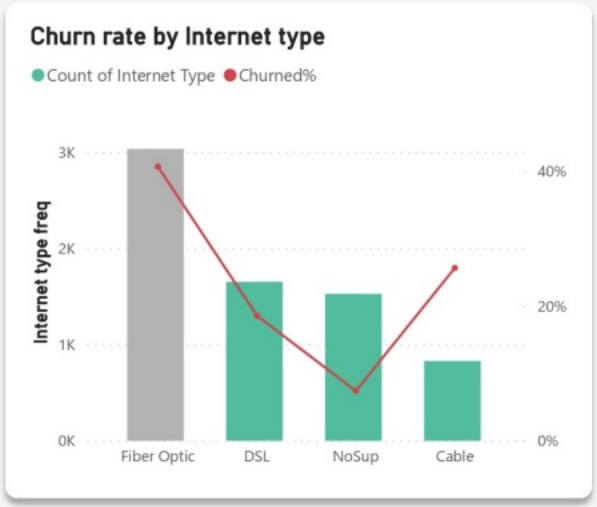
Churn Risk

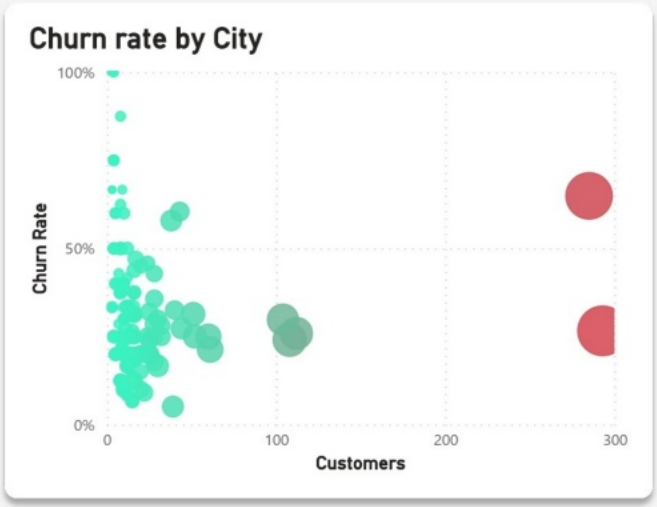








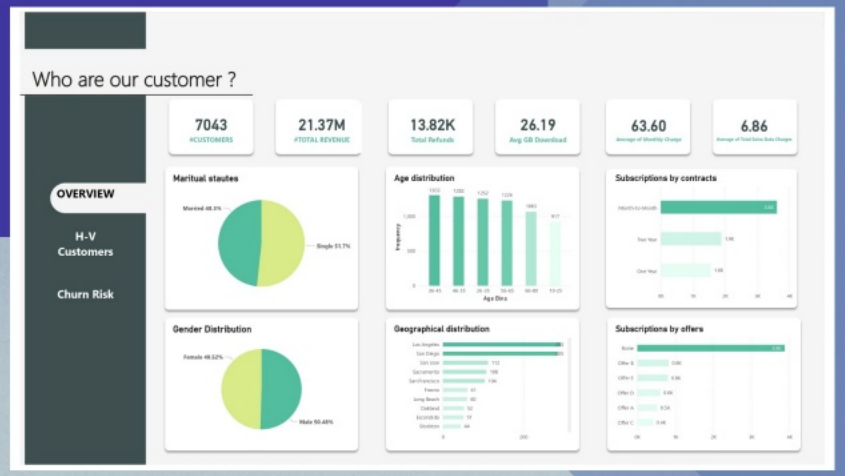


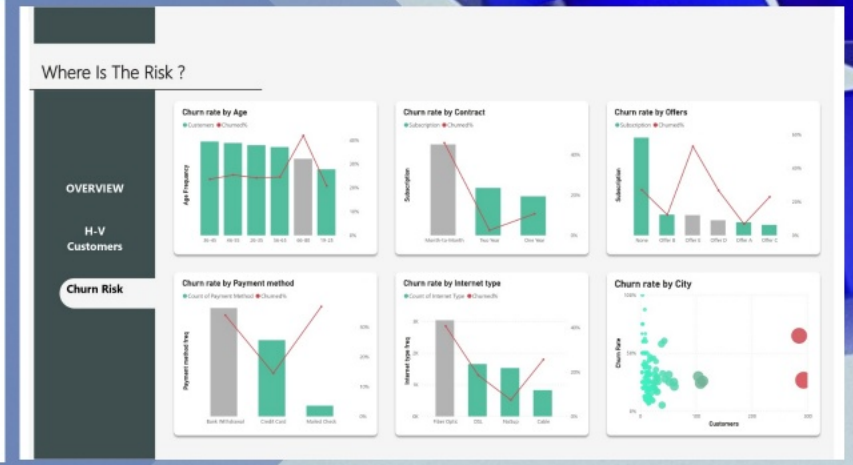


What Should company do?

- Inconsistent Offer Performance
- Geographic Loyalty Contradictions
- Payment Process Concerns

RFM





Conclusion

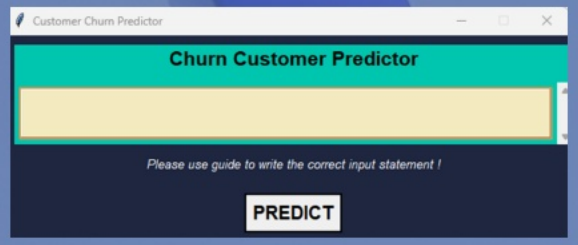
Design and Implementation

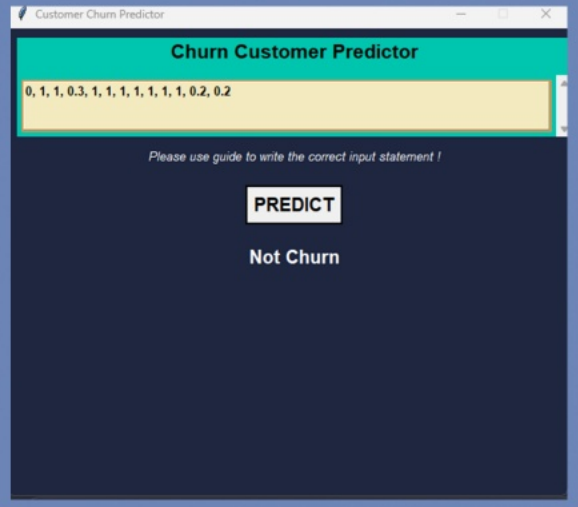
In this section, we provide insights into how we implemented our project, including the technical details of model development and the integration of visualizations.

Power Bi Dashboard

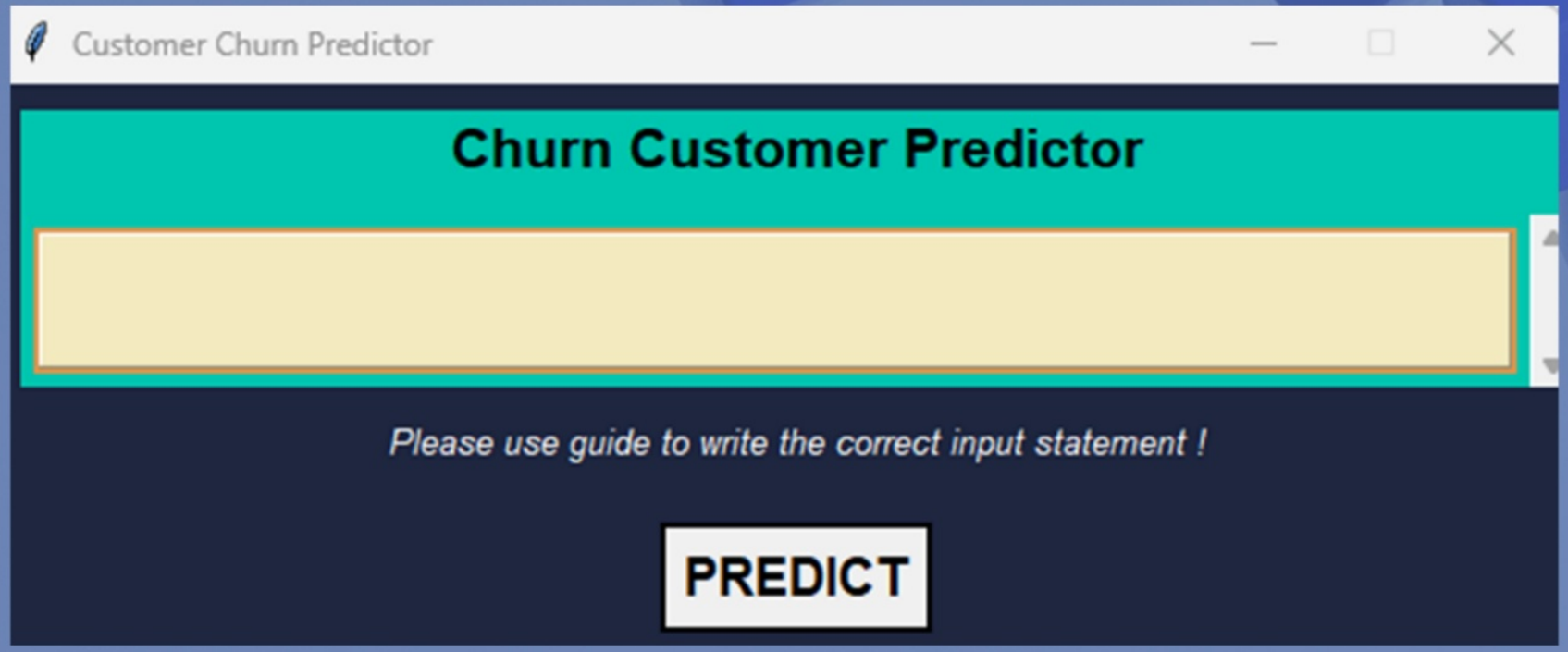
Graphical user interface

Guide





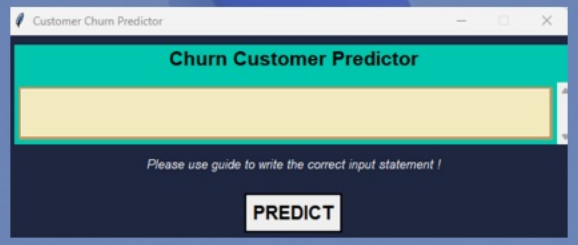
Another GUI

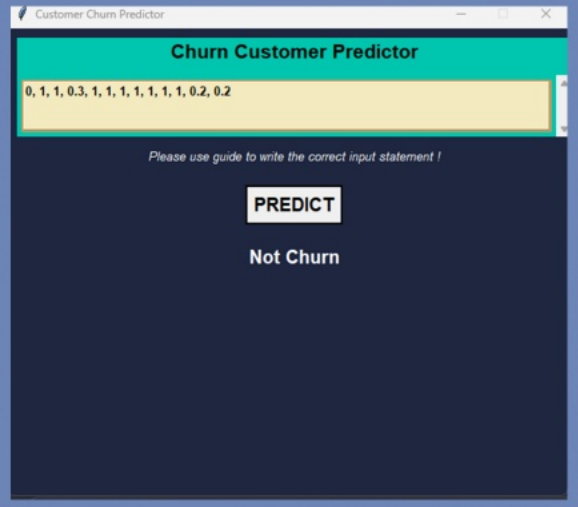

Customer Churn Predictor
Churn Customer Predictor
Please use guide to write the correct input statement !
PREDICT

Guide: How to Input Data

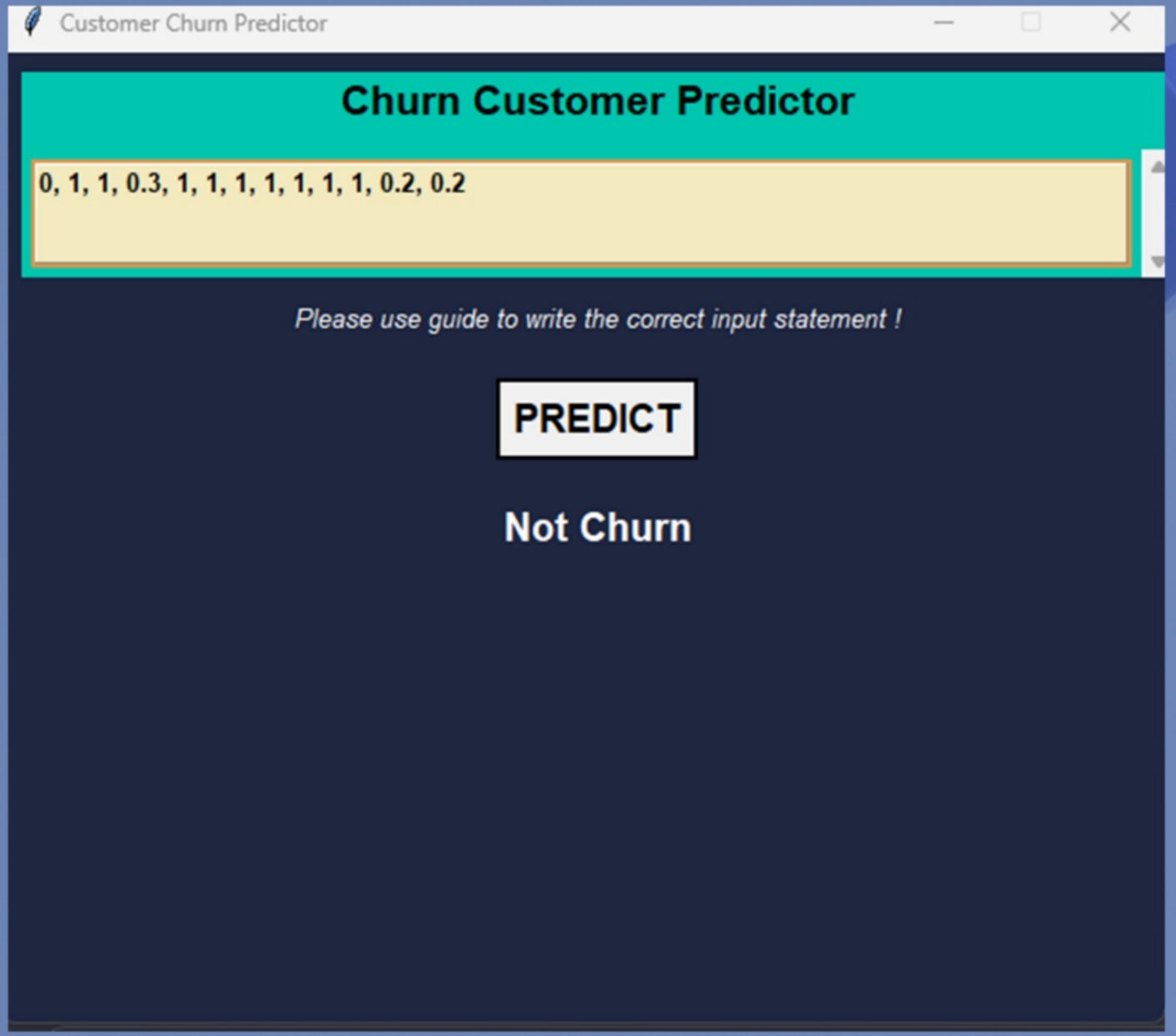
1. Senior Citizen:
• Enter '0' for 'No' and '1' for 'Yes'.
2. Partner:
• Enter '0' for 'No' and '1' for 'Yes'.
3. Dependents:
• Enter '0' for 'No' and '1' for 'Yes'.
4. Tenure Months (normalized):
• Enter a value between 0 and 1, representing the normalized tenure in months.
5. Online Security:
• Enter '2' for 'Yes', '0' for 'No', and '1' for 'No internet service'.
6. Online Backup:
• Enter '2' for 'Yes', '0' for 'No', and '1' for 'No internet service'.
7. Device Protection:
• Enter '0' for 'No', '2' for 'Yes', and '1' for 'No internet service'.
8. Tech Support:
• Enter '0' for 'No', '2' for 'Yes', and '1' for 'No internet service'.
9. Contract:
• Enter '0' for 'Month-to-month', '2' for 'Two year', and '1' for 'One year'.
10. Paperless Billing:
• Enter '1' for 'Yes' and '0' for 'No'.
11. Payment Method:
• Enter '3' for 'Mailed check', '2' for 'Electronic check', '0' for 'Bank transfer (automatic)', and '1' for 'Credit card (automatic)'.
12. Monthly Charges (normalized):
• Enter a value between 0 and 1, representing the normalized monthly charges.
13. Total Charges (normalized):
• Enter a value between 0 and 1, representing the normalized total charges.

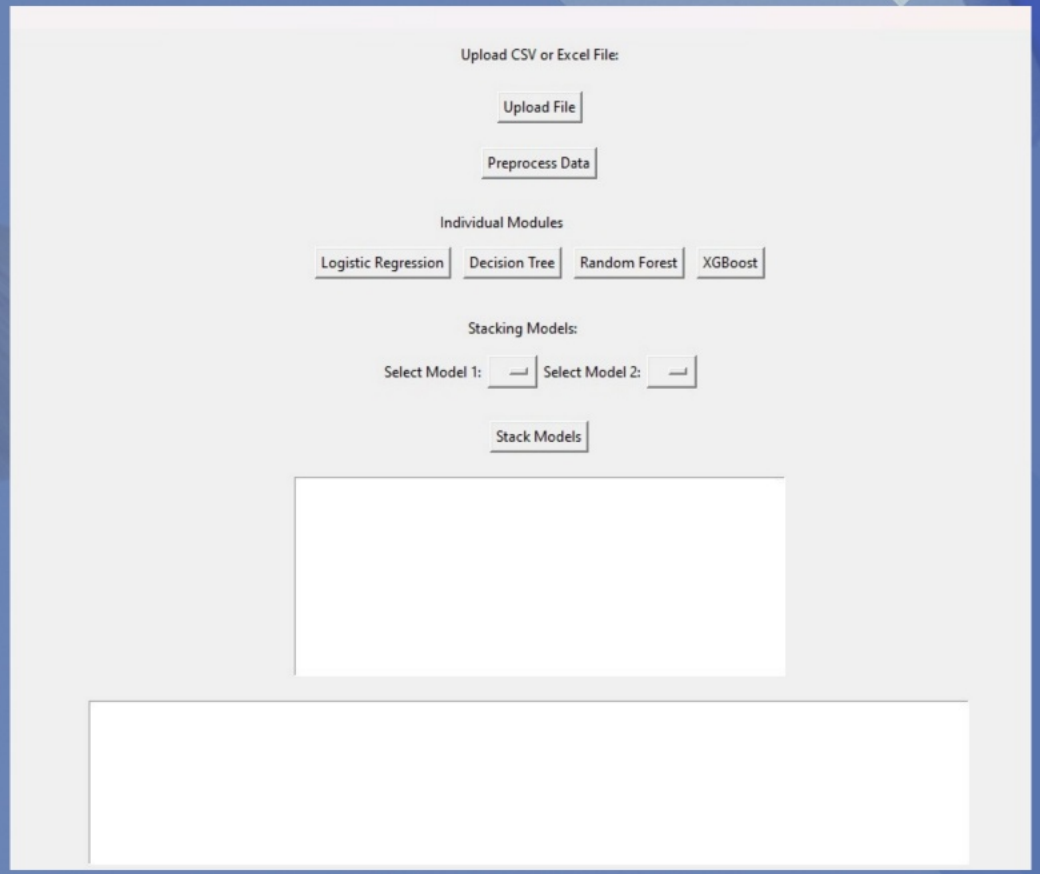
Guide

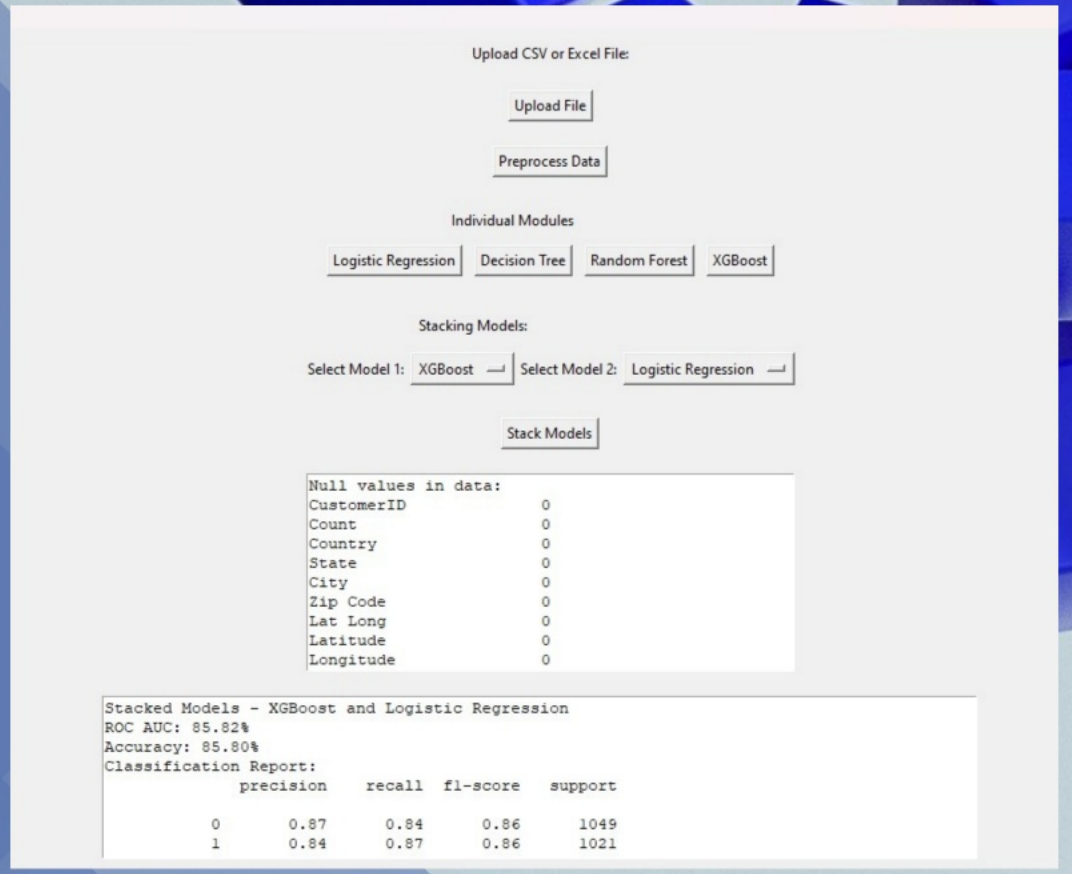




Another GUI


Customer Churn Predictor
Churn Customer Predictor
0, 1, 1, 0.3, 1, 1, 1, 1, 1, 1, 0.2, 0.2
Please use guide to write the correct input statement !
PREDICT
Not Churn


Upload CSV or Excel File:
Upload File
Preprocess Data
Individual Modules
Logistic Regression
Decision Tree
Random Forest
XGBoost
Stacking Models:
Select Model 1:
Select Model 2:
Stack Models


Upload CSV or Excel File:
Upload File
Preprocess Data
Individual Modules
Logistic Regression
Decision Tree
Random Forest
XGBoost
Stacking Models:
Select Model 1: XGBoost
Select Model 2: Logistic Regression
Stack Models
Null values in data:
CustomerID 0
Count 0
Country 0
State 0
City 0
Zip Code 0
Lat Long 0
Latitude 0
Longitude 0
Stacked Models - XGBoost and Logistic Regression
ROC AUC: 85.82%
Accuracy: 85.80%
Classification Report:
precision recall f1-score support
0 0.87 0.84 0.86 1049
1 0.84 0.87 0.86 1021

Upload CSV or Excel File:

Upload File

Preprocess Data

Individual Modules

Logistic Regression | Decision Tree | Random Forest | XGBoost

Stacking Models:

Select Model 1: Select Model 2:

Stack Models

Upload CSV or Excel File:

Upload File

Preprocess Data

Individual Modules

Logistic Regression | Decision Tree | Random Forest | XGBoost

Stacking Models:

Select Model 1: XGBoost Select Model 2: Logistic Regression

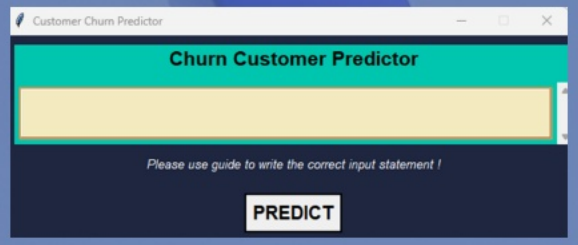
Stack Models

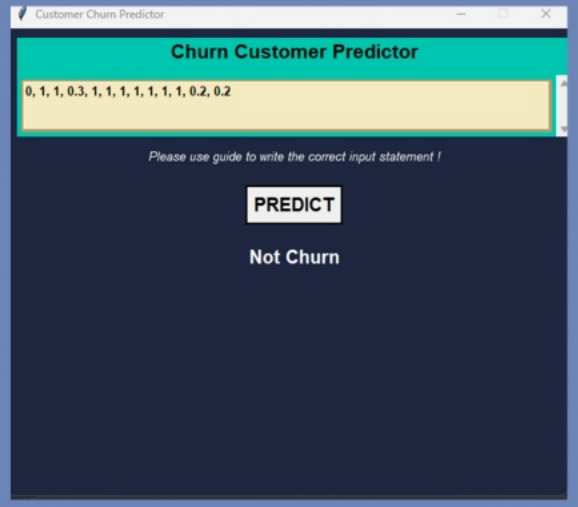
```
Null values in data:
CustomerID           0
Count                0
Country              0
State                0
City                 0
Zip Code             0
Lat Long             0
Latitude             0
Longitude            0
```

```
Stacked Models - XGBoost and Logistic Regression
ROC AUC: 85.82%
Accuracy: 85.80%
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.84      0.86      1049
           1       0.84      0.87      0.86      1021
```

Guide





Another GUI

Design and Implementation

In this section, we provide insights into how we implemented our project, including the technical details of model development and the integration of visualizations.

Power Bi Dashboard

Graphical user interface

Churn Customer Prediction Using Feature Selection & Machine Learning Models

Done By:
Yazan Hijazi
Raneem Refaie

Supervisor:
Dr.Qusai Alzoubi

Result of Models

Model	Cross Validation Score	ROC AUC Score
XGBClassifier	91.98%	82.89%
RandomForestClassifier	87.53%	78.73%
DecisionTreeClassifier	86.62%	78.78%
LogisticRegression	86.12%	77.57%
Stack of Classifiers	92.11%	83.19%

Compare our results with others' results

Model	Our Model	Other Model
XGBClassifier	91.98%	80.80%
RandomForestClassifier	87.53%	79.05%
DecisionTreeClassifier	86.62%	77.61%
LogisticRegression	86.12%	79.19%
Stack of Classifiers	92.11%	86.40%

Churn Customer Prediction Using Feature Selection & Machine Learning Models

Done By:
Yazan Hijazi
Raneem Refaie

Supervisor:
Dr.Qusai Alzoubi