BANK CUSTOMER CHURN PREDICTION MODEL

Harnessing Machine Learning to Enhance Customer Retention Strategies

http://3.99.190.226:8501/

by Suha Islaih and Osear Okinga S

INTRODUCTION

- Digital Transformation in Banking
- The Challenge of Customer Loyalty
- Leveraging Data Science

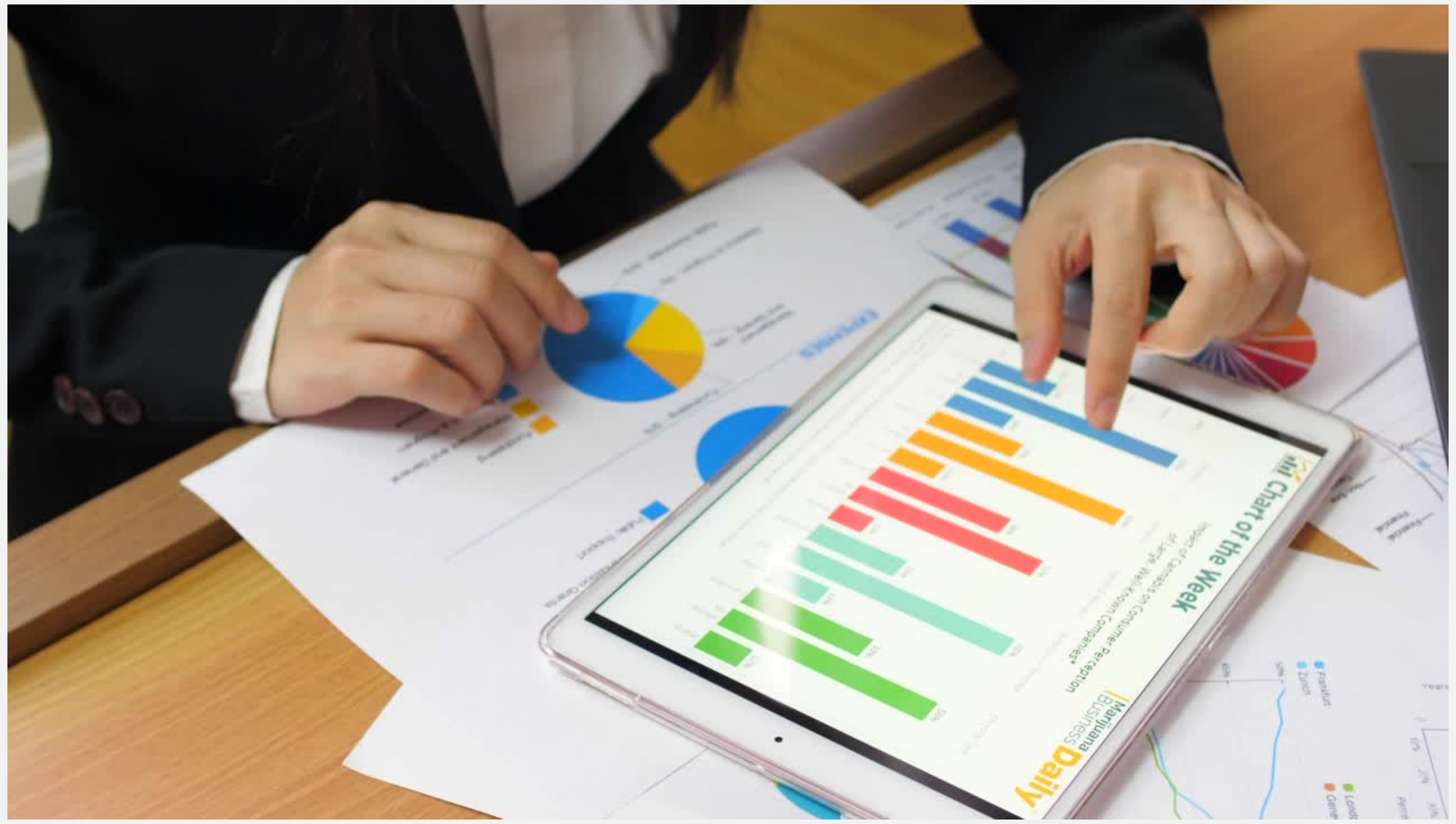
PROBLEM STATEMENT

- Evolving Customer Expectations
- Advanced Analytics for Prediction
- From Insights to Action

OBJECTIVES

- Age Impact: Who's more likely to leave, younger or older customers?
- Gender Difference: Is there a churn gap between men and women?
- Credit Score's Role: How does a customer's credit score affect their likelihood of churning?

OBJECTIVES



- Geographical Trends: From which countries are customers more likely to churn?
- Data Insights: Exploring the best ways to visualize our data for clearer insights.
- Finding the Best Model: Identifying the top machine learning model to predict churn effectively.

BRIEF OVERVIEW OF THE APPROACH

CRISP-DM

METHODOLOGY FRAMEWORK

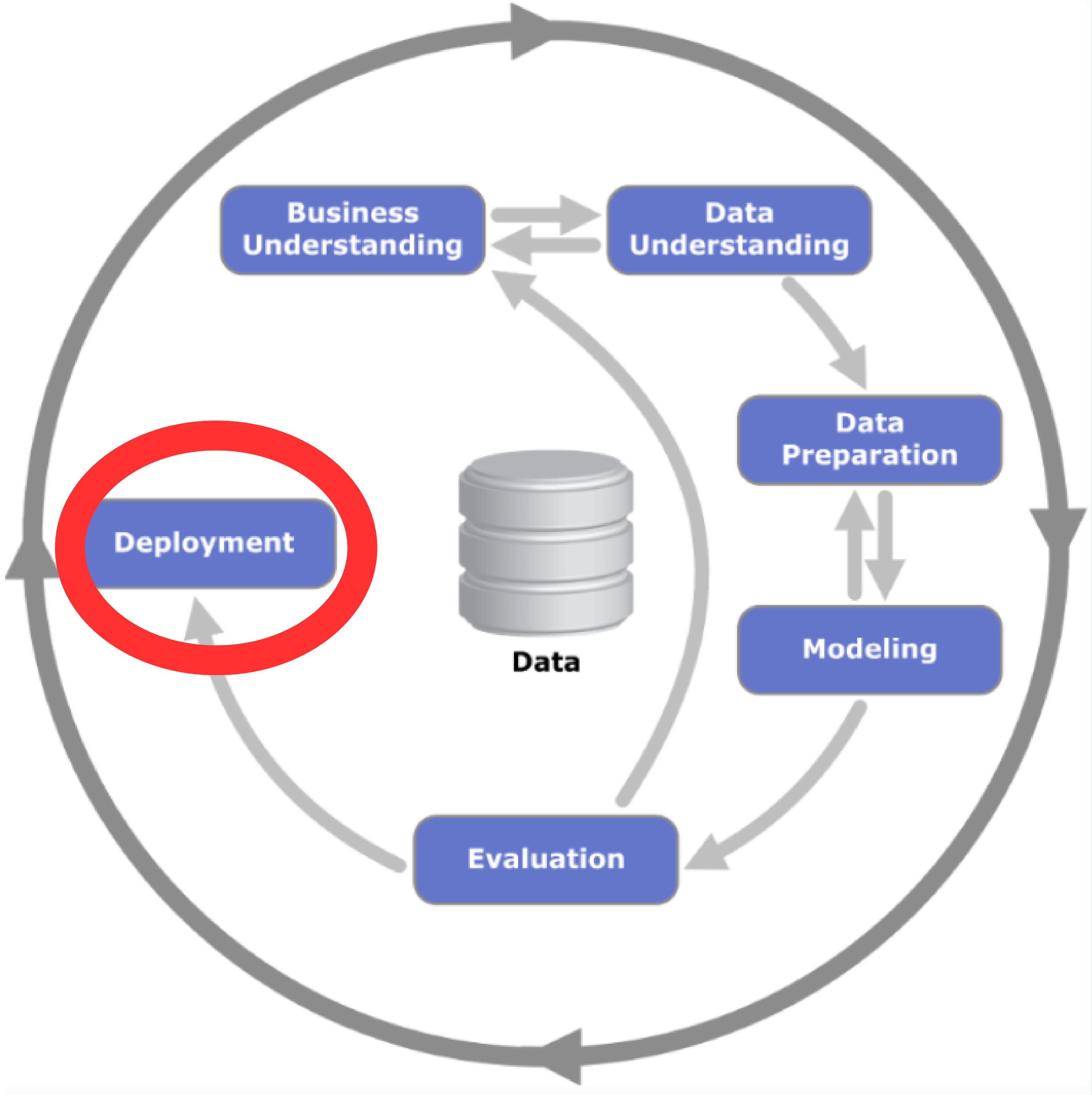
METHODOLOGY FRAMEWORK

METHODOLOGY FRAMEWORK

METHODOLOGY FRAMEWORK

METHODOLOGY FRAMEWORK

METHODOLOGY FRAMEWORK



DATA COLLECTION AND PREPROCESSING

DATA SOURCES

kaggle

SOFTWARE TOOLS & HARDWARE REQUIREMENTS

seaborn

jupyter

pandas

scikit learn

matplotlib

NumPy

DATA DESCRIPTION AND EXPLORATION

- Overview of dataset features (e.g., Age, Geography, Gender).
- Initial analysis to identify patterns.
- Importance of understanding customer demographics and behaviours.

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634602	Hargrave	619	France	Female	42.0	2	0.00	1	1.0	1.0	101348.88	1
1	2	15647311	Hill	608	Spain	Female	41.0	1	83807.86	1	0.0	1.0	112542.58	0
2	3	15619304	Onio	502	France	Female	42.0	8	159660.80	3	1.0	0.0	113931.57	1
3	4	15701354	Boni	699	France	Female	39.0	1	0.00	2	0.0	0.0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	43.0	2	125510.82	1	NaN	1.0	79084.10	0

DATA CLEANING, TRANSFORMATION & FEATURE SELECTION

- Cleaning irrelevant data (e.g., removing unnecessary columns).
- Transforming data for analysis.
- Selecting features critical for predicting customer churn.

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10002 entries, 0 to 10001
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10002 non-null  int64
 1   CustomerId       10002 non-null  int64
 2   Surname          10002 non-null  object
 3   CreditScore      10002 non-null  int64
 4   Geography        10001 non-null  object
 5   Gender           10002 non-null  object
 6   Age              10001 non-null  float64
 7   Tenure           10002 non-null  int64
 8   Balance          10002 non-null  float64
 9   NumOfProducts    10002 non-null  int64
 10  HasCrCard        10001 non-null  float64
 11  IsActiveMember   10001 non-null  float64
 12  EstimatedSalary  10002 non-null  float64
 13  Exited           10002 non-null  int64
dtypes: float64(5), int64(6), object(3)
memory usage: 1.1+ MB
```

df.dtypes

```
RowNumber            int64
CustomerId           int64
Surname             object
CreditScore          int64
Geography           object
Gender              object
Age                float64
Tenure               int64
Balance            float64
NumOfProducts        int64
HasCrCard          float64
IsActiveMember     float64
EstimatedSalary    float64
Exited               int64
dtype: object
```

df.isnull().sum()

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          1
Gender             0
Age                1
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          1
IsActiveMember     1
EstimatedSalary    0
Exited             0
dtype: int64
```

Table 1: Dataset Info; Types & Missing Values

DATA CLEANING, TRANSFORMATION & FEATURE SELECTION

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
count	10002.000000	1.000200e+04	10002.000000	10001.000000	10002.000000	10002.000000	10002.000000	10001.000000	10001.000000	10002.000000	10002.000000
mean	5001.499600	1.569093e+07	650.555089	38.922311	5.012498	76491.112875	1.530194	0.705529	0.514949	100083.331145	0.203759
std	2887.472338	7.193177e+04	96.661615	10.487200	2.891973	62393.474144	0.581639	0.455827	0.499801	57508.117802	0.402812
min	1.000000	1.556570e+07	350.000000	18.000000	0.000000	0.000000	1.000000	0.000000	0.000000	11.580000	0.000000
25%	2501.250000	1.562852e+07	584.000000	32.000000	3.000000	0.000000	1.000000	0.000000	0.000000	50983.750000	0.000000
50%	5001.500000	1.569073e+07	652.000000	37.000000	5.000000	97198.540000	1.000000	1.000000	1.000000	100185.240000	0.000000
75%	7501.750000	1.575323e+07	718.000000	44.000000	7.000000	127647.840000	2.000000	1.000000	1.000000	149383.652500	0.000000
max	10000.000000	1.581569e+07	850.000000	92.000000	10.000000	250898.090000	4.000000	1.000000	1.000000	199992.480000	1.000000

Figure 3: Statistical Description of the Dataset

EXPLORATORY DATA ANALYSIS (EDA)

PAIRPLOT/ SCATTERPLOT MATRIX

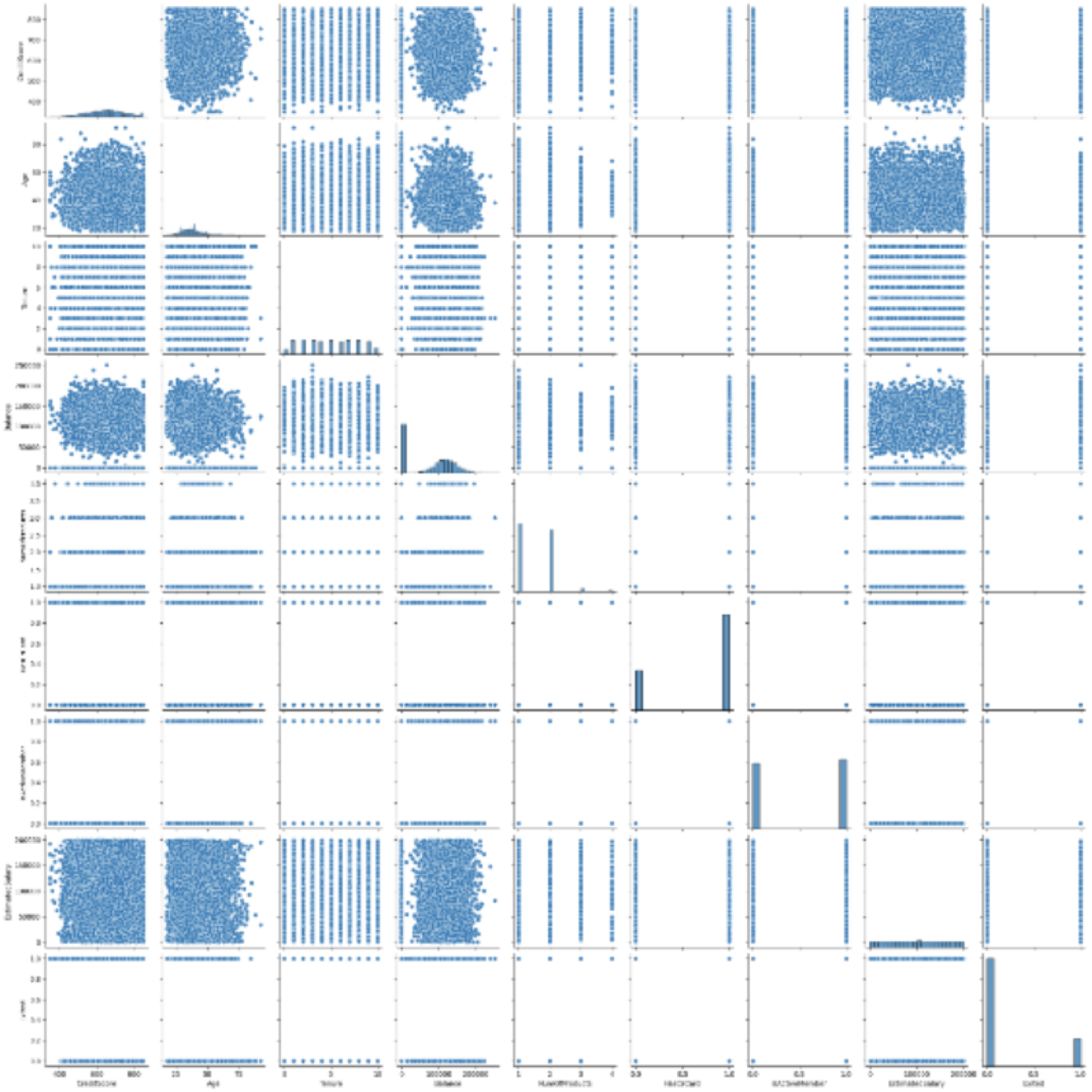
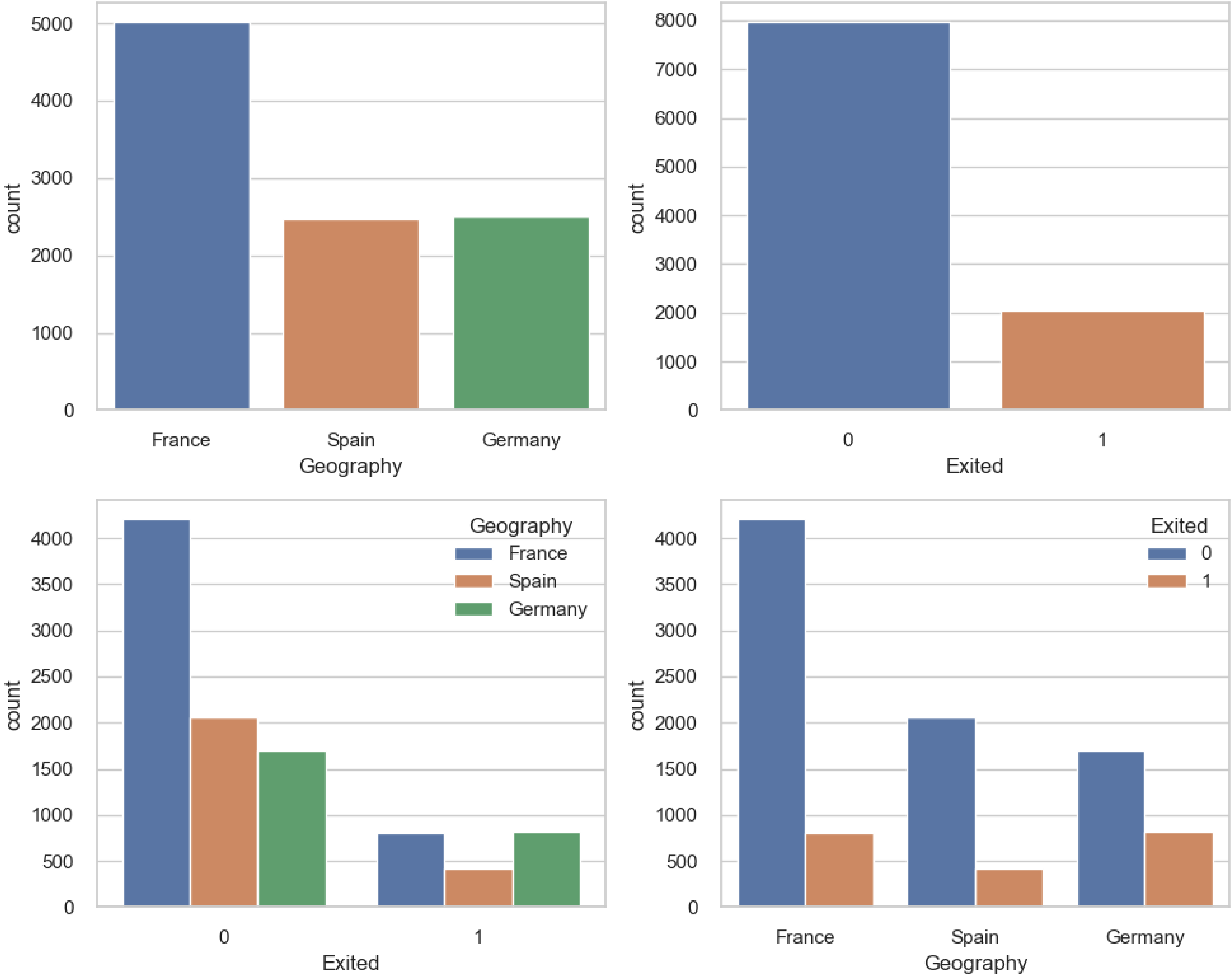


Figure SEQ Figure * ARABIC 4: Dataset - Pairplot

CUSTOMER DEMOGRAPHICS



AGE AND CHURN

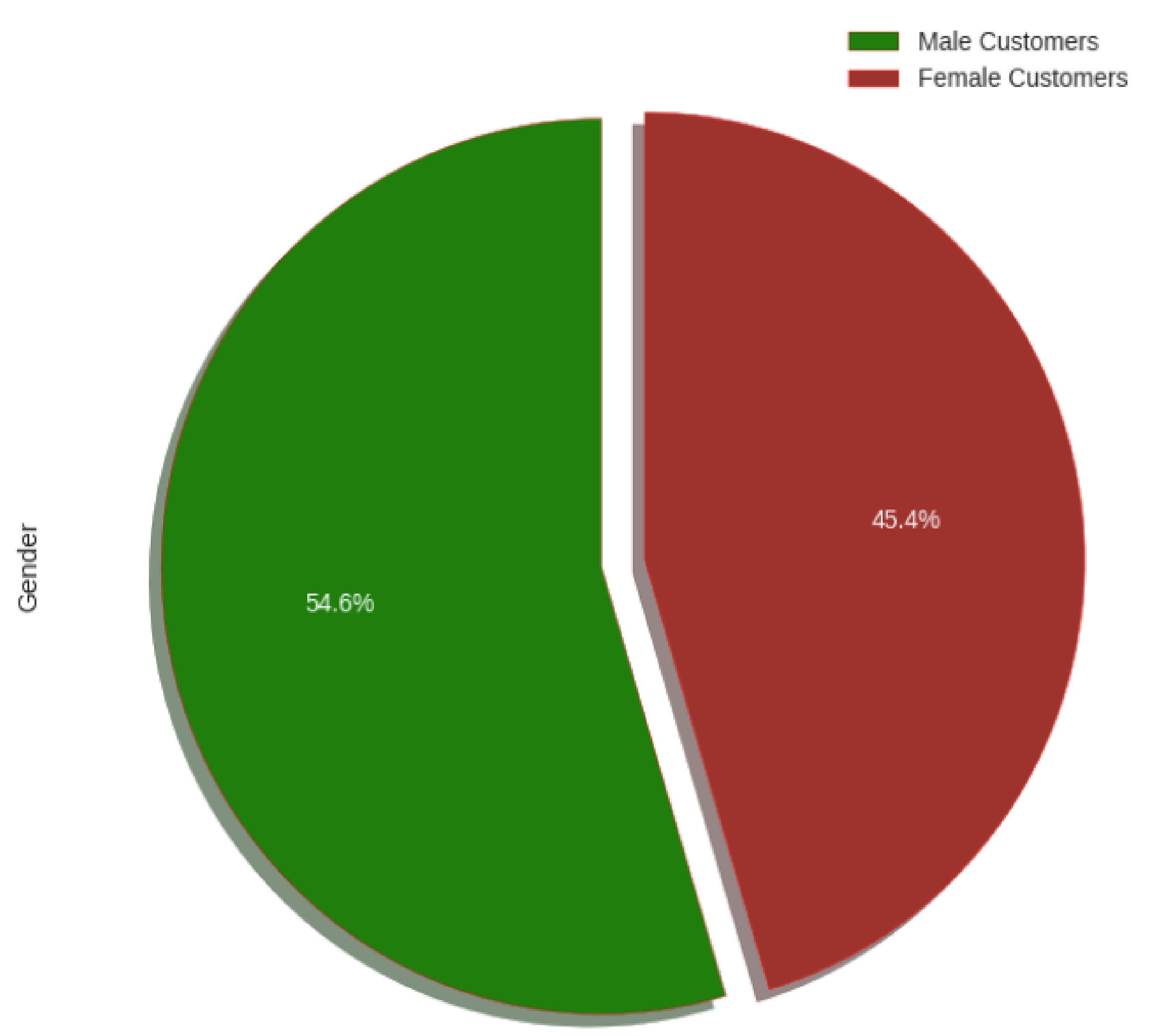
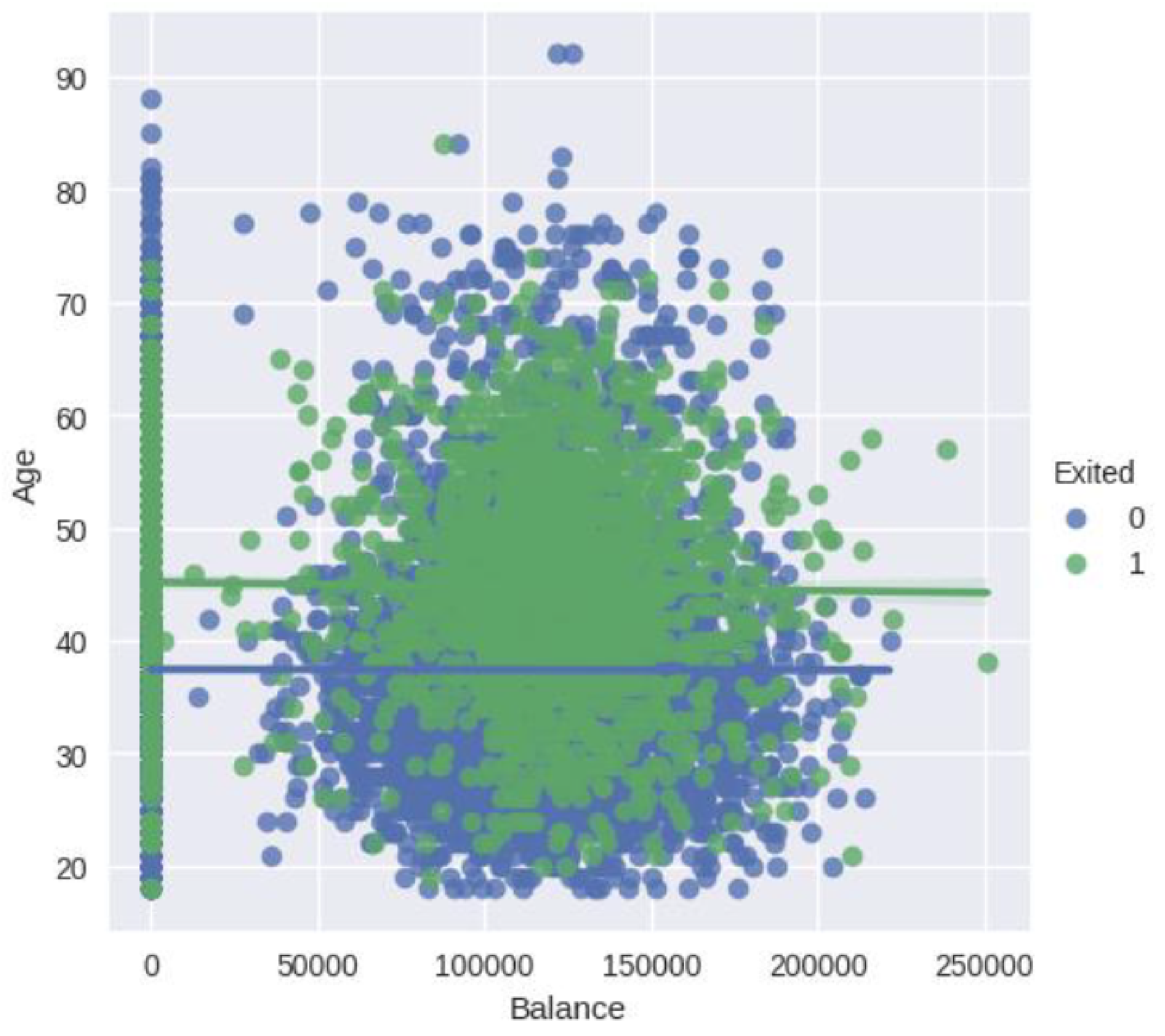


Figure 6: Gender - Pie plot



UNDERSTANDING - CREDIT ANALYSIS

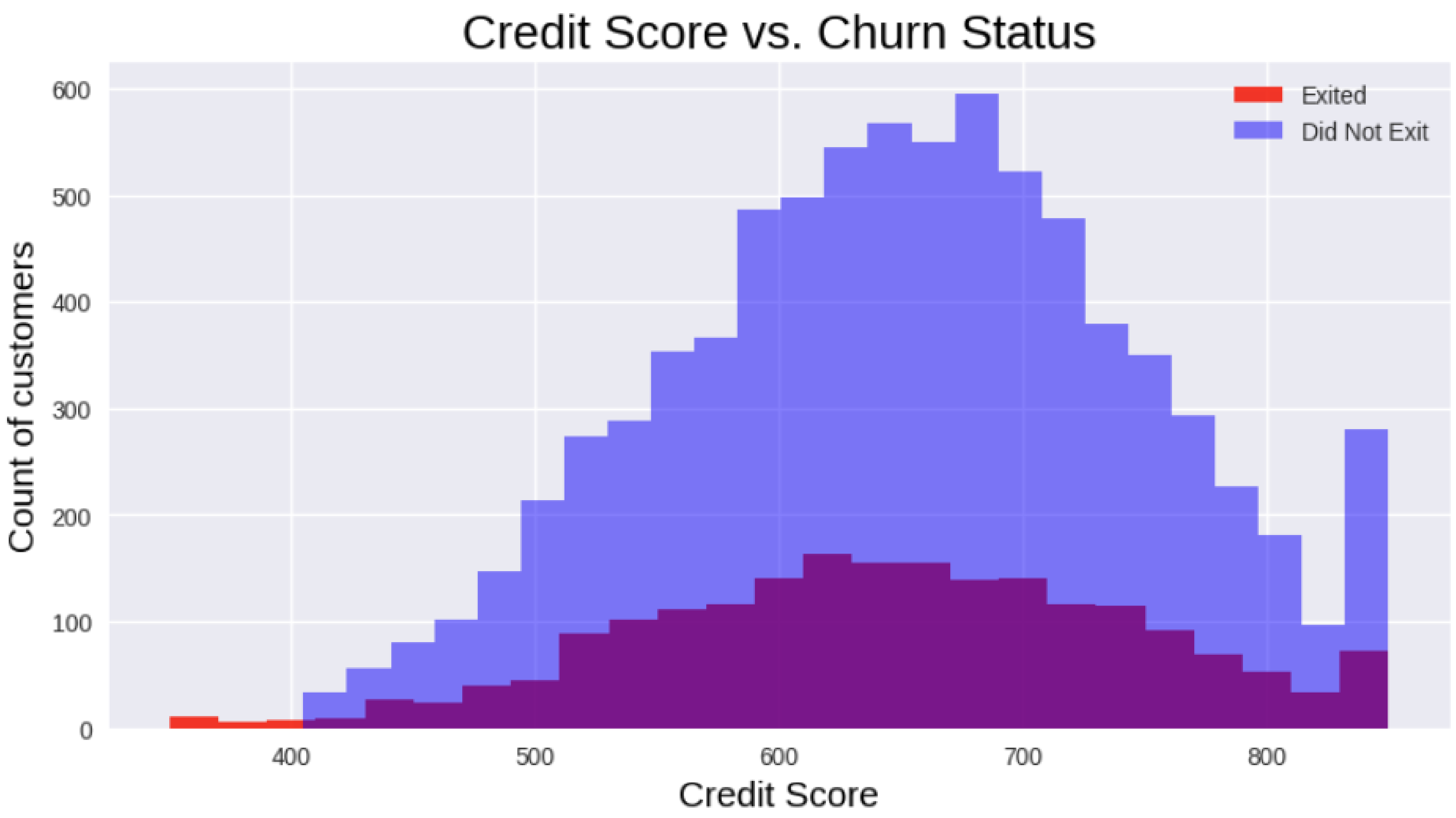


Figure 8: Credit Score and churn status

COUNT PLOTS TO MAP THE DEPENDENCE OF 'EXITED' COLUMN ON CATEGORICAL FEATURES.

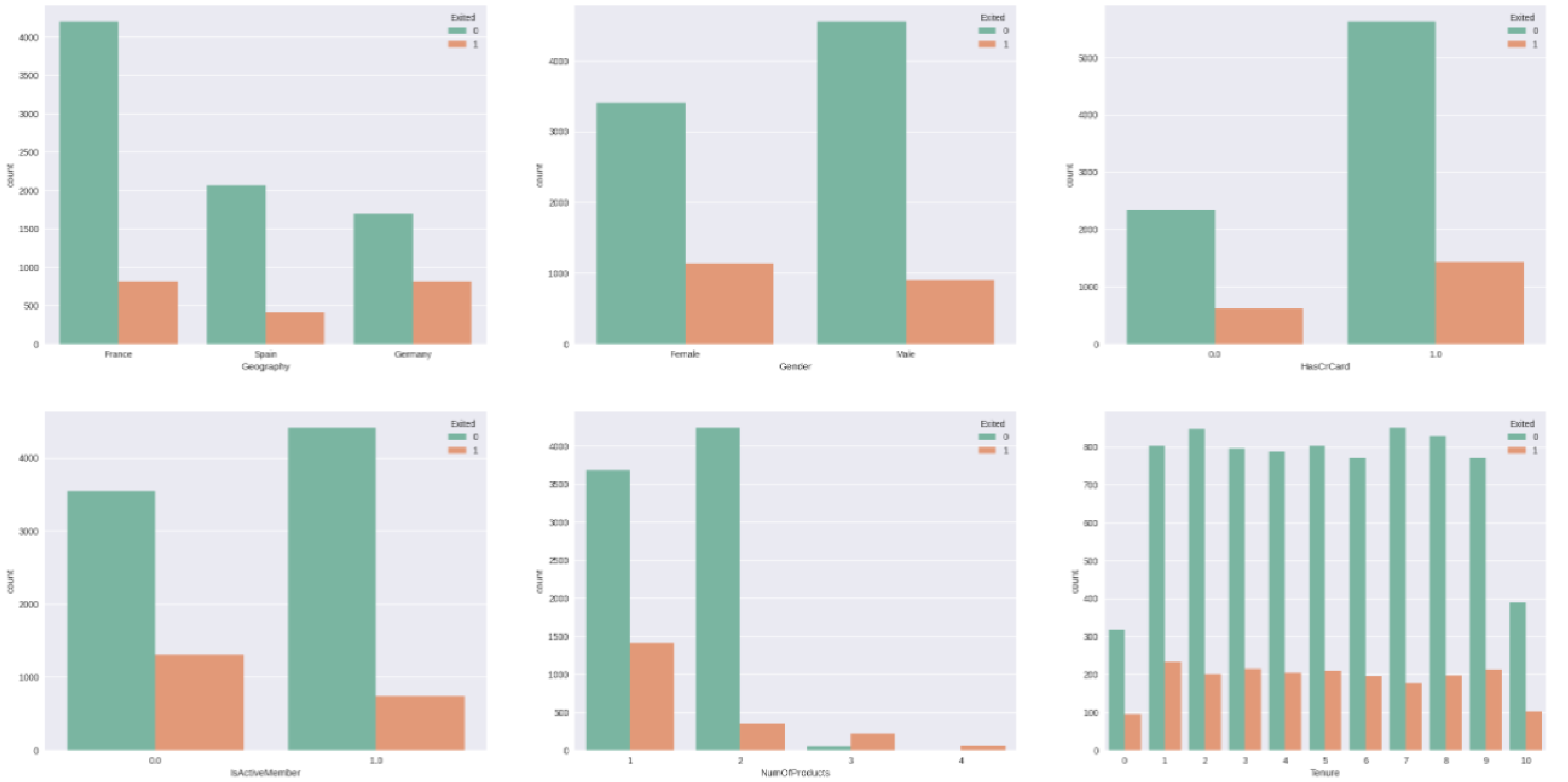


Figure 10: Exited column on categorical features.

CORRELATION ANALYSIS

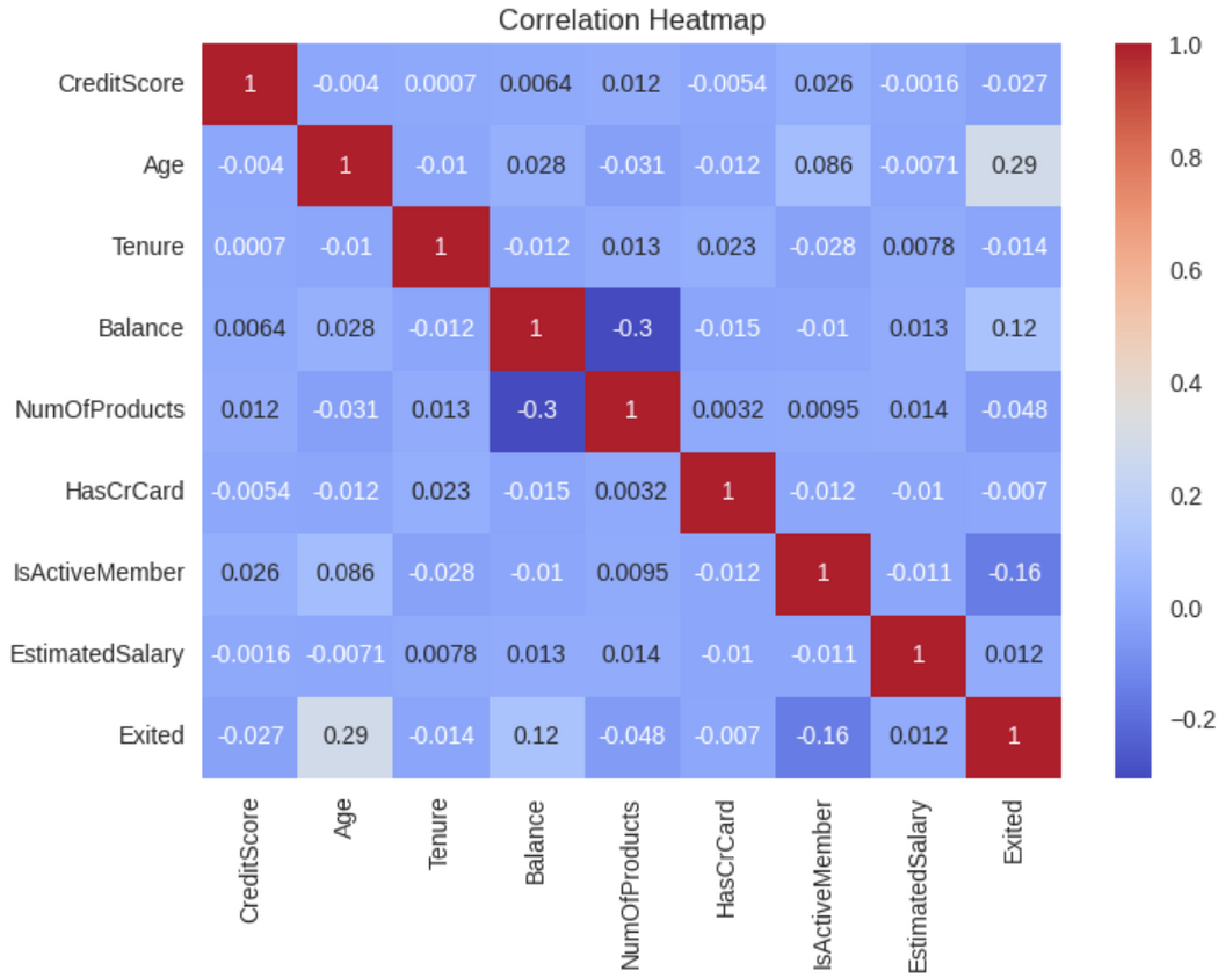


Figure 12: Correlation Heatmap

MODEL DEVELOPMENT

MODEL SELECTION AND TRAINING

```python
#Initialization and Setting Up the Environment
from pycaret.classification import *
clf1= setup(data = data,
            target = 'Exited',
            session_id=123,
            ignore_features=['CustomerId', 'Surname'],
            fix_imbalance_method = 'SMOTE')
```

	Description	Value
0	Session id	123
1	Target	Exited
2	Target type	Binary
3	Original data shape	(8002, 14)
4	Transformed data shape	(8002, 14)
5	Transformed train set shape	(5601, 14)
6	Transformed test set shape	(2401, 14)
7	Ignore features	2
8	Ordinal features	1
9	Numeric features	9
10	Categorical features	2
11	Rows with missing values	0.0%
12	Preprocess	True
13	Imputation type	simple
14	Numeric imputation	mean
15	Categorical imputation	mode
16	Maximum one-hot encoding	25
17	Encoding method	None
18	Fold Generator	StratifiedKFold
19	Fold Number	10
20	CPU Jobs	-1
21	Use GPU	False
22	Log Experiment	False
23	Experiment Name	clf-default-name
24	USI	2878

COMPARE DIFFERENT MODEL

compare_models()

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
rf	Random Forest Classifier	0.8565	0.8481	0.4405	0.7657	0.5577	0.4793	0.5063	0.1380
gbc	Gradient Boosting Classifier	0.8557	0.8609	0.4553	0.7450	0.5638	0.4836	0.5053	0.2140
lightgbm	Light Gradient Boosting Machine	0.8536	0.8563	0.4874	0.7118	0.5773	0.4926	0.5063	0.1010
ada	Ada Boost Classifier	0.8495	0.8376	0.4553	0.7100	0.5540	0.4687	0.4859	0.0750
et	Extra Trees Classifier	0.8493	0.8475	0.4101	0.7418	0.5265	0.4460	0.4745	0.0990
ridge	Ridge Classifier	0.8070	0.0000	0.1294	0.6568	0.2151	0.1585	0.2281	0.0190
lda	Linear Discriminant Analysis	0.8038	0.7653	0.2329	0.5563	0.3270	0.2344	0.2655	0.0190
dummy	Dummy Classifier	0.7945	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0190
lr	Logistic Regression	0.7858	0.6733	0.0504	0.3418	0.0874	0.0383	0.0607	0.0240
dt	Decision Tree Classifier	0.7858	0.6832	0.5091	0.4805	0.4940	0.3583	0.3588	0.0230
nb	Naive Bayes	0.7822	0.7427	0.0634	0.3431	0.1065	0.0451	0.0668	0.0180
knn	K Neighbors Classifier	0.7606	0.5242	0.0826	0.2495	0.1236	0.0245	0.0295	0.0270
svm	SVM - Linear Kernel	0.6428	0.0000	0.2914	0.1390	0.1310	0.0198	0.0356	0.0260
qda	Quadratic Discriminant Analysis	0.5860	0.5048	0.3597	0.1930	0.2007	0.0011	0.0022	0.0190

MODELLING

compare_models()

```
best_model = create_model('rf')
```

Fold	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.8717	0.8574	0.4655	0.8438	0.6000	0.5310	0.5644
1	0.8482	0.8547	0.4696	0.6923	0.5596	0.4719	0.4849
2	0.8732	0.8576	0.4348	0.8929	0.5848	0.5203	0.5673
3	0.8625	0.8276	0.4522	0.7879	0.5746	0.4997	0.5271
4	0.8571	0.8720	0.4522	0.7536	0.5652	0.4861	0.5088
5	0.8482	0.8510	0.4522	0.7027	0.5503	0.4641	0.4804
6	0.8625	0.8468	0.4174	0.8276	0.5549	0.4838	0.5236
7	0.8589	0.8774	0.4870	0.7368	0.5864	0.5056	0.5214
8	0.8500	0.8264	0.3913	0.7627	0.5172	0.4391	0.4735
9	0.8321	0.8103	0.3826	0.6567	0.4835	0.3915	0.4119
Mean	0.8565	0.8481	0.4405	0.7657	0.5577	0.4793	0.5063
Std	0.0117	0.0200	0.0322	0.0699	0.0330	0.0389	0.0438

```
best_model2 = create_model('gbc')
```

Fold	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.8610	0.8624	0.4483	0.7879	0.5714	0.4958	0.5239
1	0.8375	0.8641	0.4348	0.6579	0.5236	0.4305	0.4439
2	0.8750	0.8599	0.4870	0.8358	0.6154	0.5469	0.5754
3	0.8607	0.8488	0.4696	0.7606	0.5806	0.5027	0.5237
4	0.8696	0.8942	0.5217	0.7692	0.6218	0.5465	0.5615
5	0.8500	0.8539	0.4783	0.6962	0.5670	0.4800	0.4924
6	0.8625	0.8558	0.4174	0.8276	0.5549	0.4838	0.5236
7	0.8804	0.8995	0.5478	0.8077	0.6528	0.5838	0.5998
8	0.8393	0.8402	0.3826	0.6984	0.4944	0.4084	0.4346
9	0.8214	0.8298	0.3652	0.6087	0.4565	0.3576	0.3743
Mean	0.8557	0.8609	0.4553	0.7450	0.5638	0.4836	0.5053
Std	0.0176	0.0206	0.0548	0.0724	0.0569	0.0655	0.0664

```
best_model3 = create_model('lightgbm')
```

Fold	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.8556	0.8480	0.5000	0.7160	0.5888	0.5046	0.5166
1	0.8429	0.8640	0.4696	0.6667	0.5510	0.4592	0.4696
2	0.8679	0.8652	0.4609	0.8154	0.5889	0.5173	0.5472
3	0.8625	0.8393	0.5217	0.7317	0.6091	0.5285	0.5397
4	0.8500	0.8657	0.4870	0.6914	0.5714	0.4838	0.4947
5	0.8536	0.8722	0.5043	0.6988	0.5859	0.4997	0.5095
6	0.8643	0.8576	0.4870	0.7671	0.5957	0.5190	0.5384
7	0.8571	0.8909	0.5652	0.6842	0.6190	0.5321	0.5358
8	0.8446	0.8425	0.4435	0.6892	0.5397	0.4515	0.4674
9	0.8375	0.8175	0.4348	0.6579	0.5236	0.4305	0.4439
Mean	0.8536	0.8563	0.4874	0.7118	0.5773	0.4926	0.5063
Std	0.0094	0.0193	0.0366	0.0458	0.0291	0.0333	0.0342

PERFORMANCE METRICS (AUC-ROC, ACCURACY, PRECISION, ETC,...)

```
predictions = predict_model(best_model, data=testData)
print(predictions.columns)

predictions2 = predict_model(best_model2, data=testData)
print(predictions2.columns)

predictions3 = predict_model(best_model3, data=testData)
print(predictions3.columns)
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Random Forest Classifier	0.9350	0.9505	0.7315	0.9196	0.8148	0.7760	0.7835

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited', 'prediction_label',
       'prediction_score'],
      dtype='object')
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Gradient Boosting Classifier	0.8660	0.8759	0.4322	0.7860	0.5578	0.4865	0.5168

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited', 'prediction_label',
       'prediction_score'],
      dtype='object')
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Light Gradient Boosting Machine	0.9105	0.9250	0.6394	0.8681	0.7364	0.6840	0.6956

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited', 'prediction_label',
       'prediction_score'],
      dtype='object')
```

USING TESTDATA TO TEST THE MODEL

```
testData['Predicted Exited'] = predictions['prediction_label']
```

```
testData.head(20)
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	Predicted Exited
0	8003	15753895	Blue	590	Spain	Male	37.0	1	0.00	2	0.0	0.0	133535.99	0	0
1	8004	15595426	Madukwe	603	Spain	Male	57.0	6	105000.85	2	1.0	1.0	87412.24	1	0
2	8005	15645815	Mills	615	France	Male	45.0	5	0.00	2	1.0	1.0	164886.64	0	0
3	8006	15632848	Ferrari	634	France	Female	36.0	1	69518.95	1	1.0	0.0	116238.39	0	0
4	8007	15703068	Nixon	716	Germany	Male	41.0	8	126145.54	2	1.0	1.0	138051.19	0	0
5	8008	15791513	Manfrin	647	France	Male	41.0	4	138937.35	1	1.0	1.0	101617.64	1	0
6	8009	15587210	McCartney	591	Germany	Female	44.0	10	113581.98	1	1.0	0.0	1985.41	0	0
7	8010	15793803	Robinson	574	France	Male	34.0	1	112572.39	1	0.0	0.0	165626.60	0	0
8	8011	15787756	Nkemdirim	467	Germany	Male	51.0	10	114514.71	2	1.0	0.0	177784.68	1	1
9	8012	15723437	Sal	701	France	Female	35.0	2	0.00	2	1.0	1.0	65765.22	0	0
10	8013	15702715	Kao	747	France	Female	34.0	10	0.00	2	1.0	1.0	50759.80	0	0
11	8014	15809872	Ikechukwu	650	France	Male	32.0	2	84906.45	1	1.0	0.0	163216.48	0	0
12	8015	15644295	Hargreaves	731	Spain	Female	39.0	2	126816.18	1	1.0	1.0	74850.93	0	0
13	8016	15778694	Sievier	638	Germany	Female	26.0	1	105249.76	2	1.0	1.0	23491.09	0	0
14	8017	15759555	Murphy	569	Spain	Male	41.0	2	0.00	2	1.0	0.0	134272.57	0	0
15	8018	15631406	Munro	459	Germany	Male	50.0	5	109387.90	1	1.0	0.0	155721.15	0	0
16	8019	15616676	Donnelly	632	Germany	Male	23.0	3	122478.51	1	1.0	0.0	147230.77	1	1
17	8020	15771154	North	683	France	Female	73.0	8	137732.23	2	1.0	1.0	133210.44	0	0
18	8021	15669491	Cruz	850	France	Female	46.0	2	157866.77	1	1.0	1.0	18986.12	0	0
19	8022	15697691	Sinclair	512	France	Female	41.0	6	0.00	1	1.0	1.0	100507.81	0	0

ANALYZE MODEL

CONFUSION MATRIX

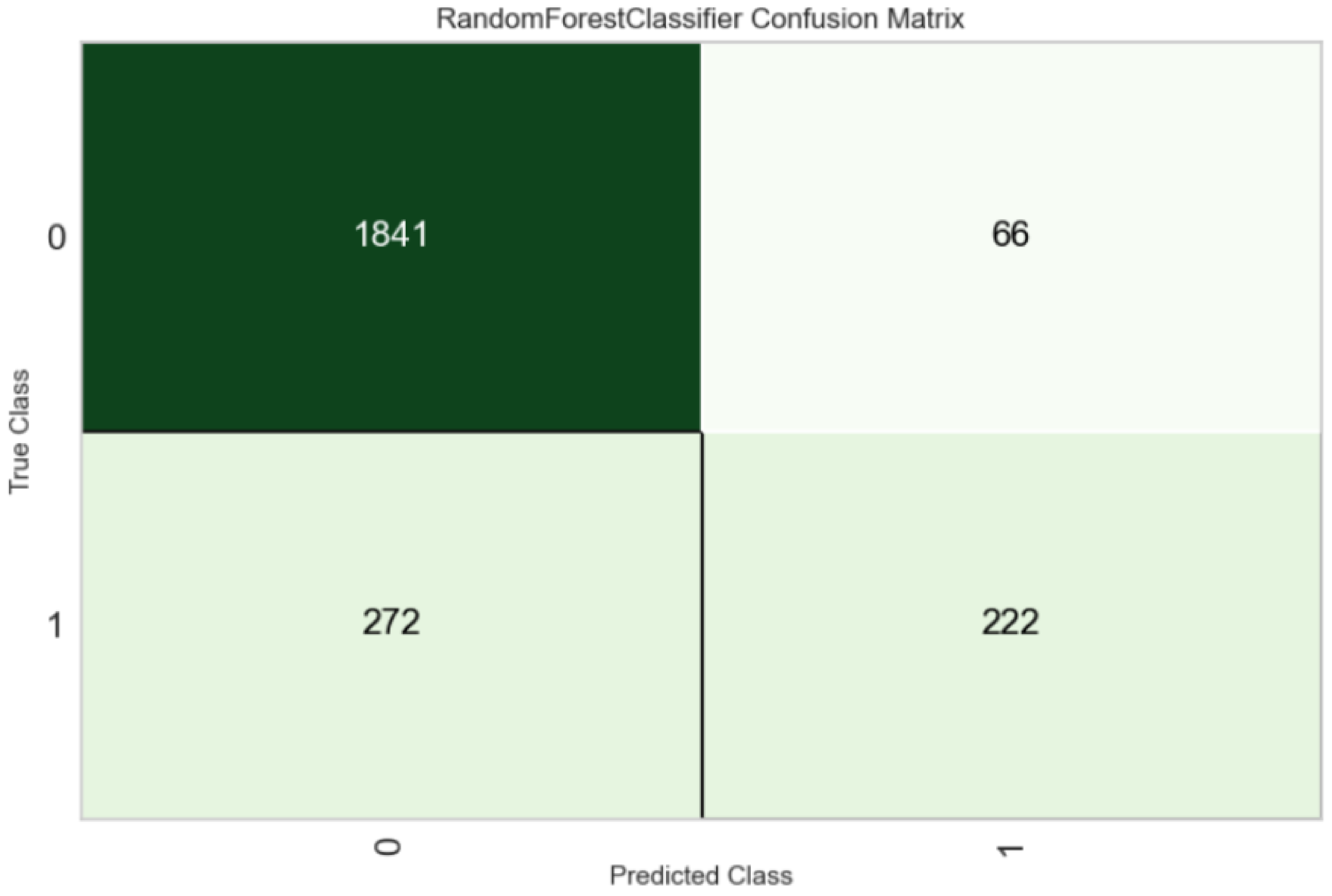


Figure 13: Confusion Matrix of Random Forest Classifier

ROC (RECEIVER OPERATING CHARACTERISTIC) - CURVE

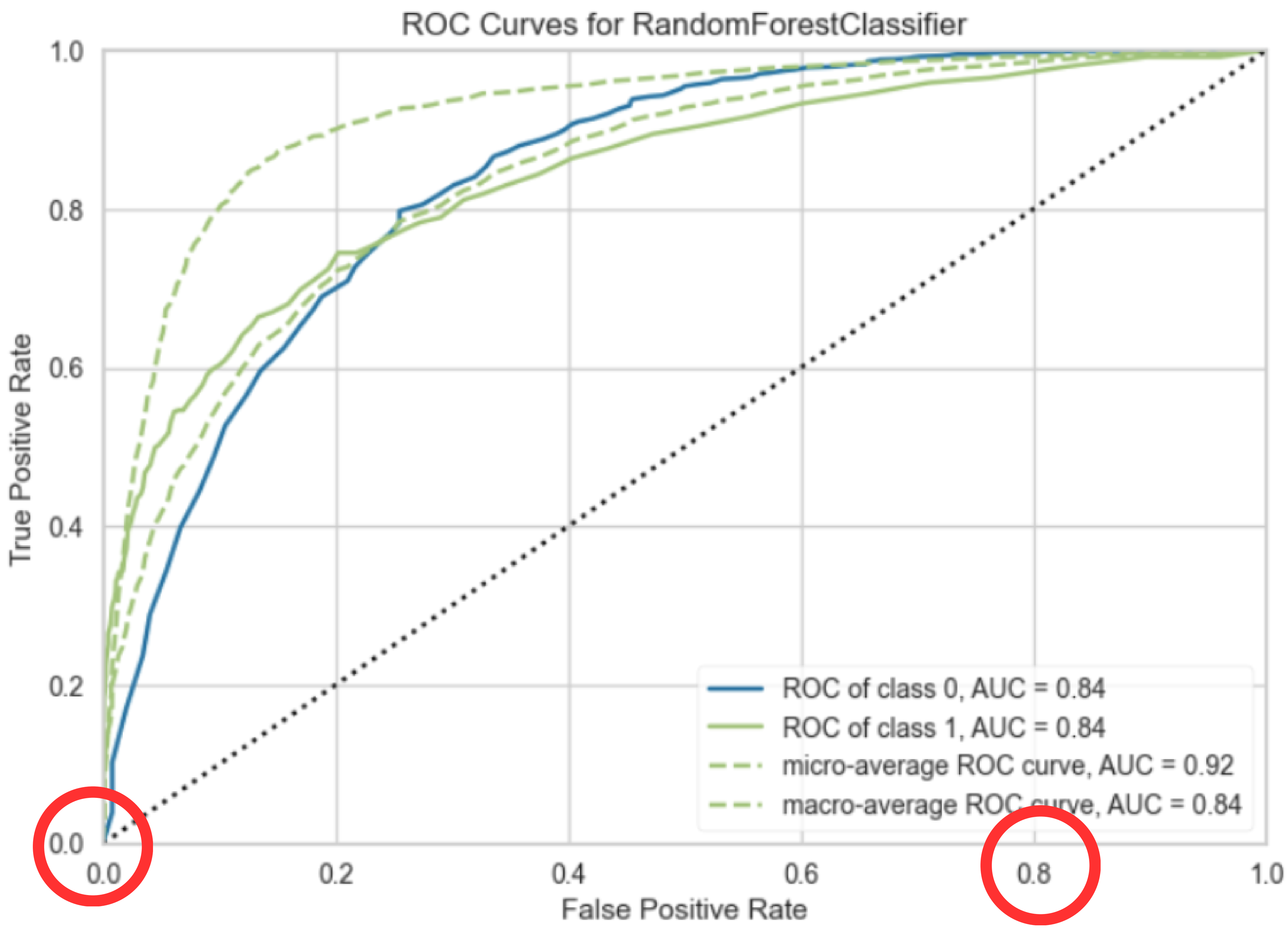


Figure 14: ROC Curve for Random Forest Classifier

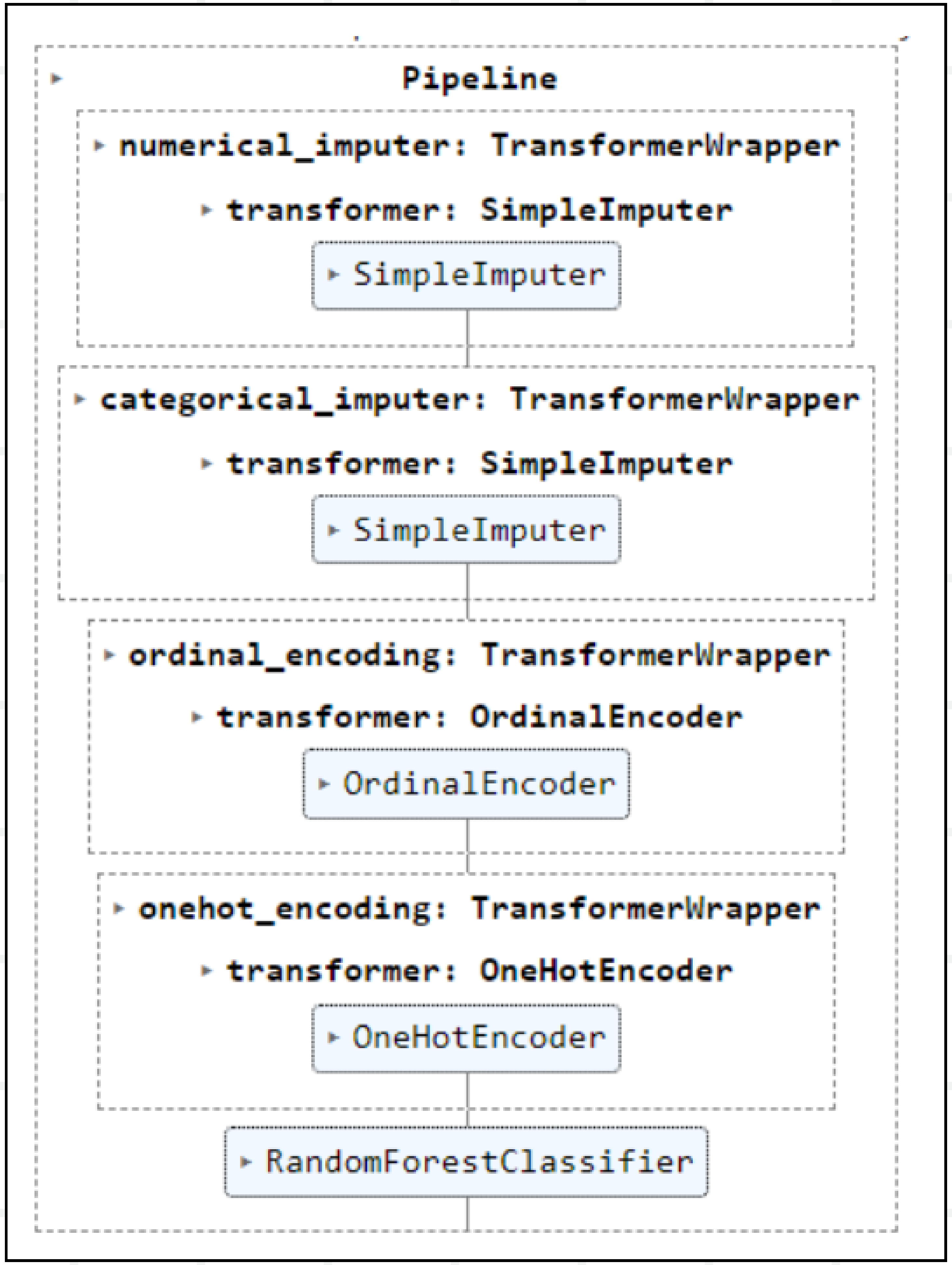
SAVE THE MODEL

```
save_model(best_model, 'churn_modelling_pipeline')
```

```
Transformation Pipeline and Model Successfully Saved
(Pipeline(memory=Memory(location=None),
          steps=[('numerical_imputer',
                  TransformerWrapper(exclude=None,
                                     include=['RowNumber', 'CreditScore', 'Age',
                                              'Tenure', 'Balance',
                                              'NumOfProducts', 'HasCrCard',
                                              'IsActiveMember',
                                              'EstimatedSalary'],
                                     transformer=SimpleImputer(add_indicator=False,
                                                               copy=True,
                                                               fill_value=None,
                                                               keep_empty_features=False,
                                                               missing_values=nan,
                                                               strategy='m...
                  RandomForestClassifier(bootstrap=True, ccp_alpha=0.0,
                                         class_weight=None, criterion='gini',
                                         max_depth=None, max_features='sqrt',
                                         max_leaf_nodes=None, max_samples=None,
                                         min_impurity_decrease=0.0,
                                         min_samples_leaf=1, min_samples_split=2,
                                         min_weight_fraction_leaf=0.0,
                                         n_estimators=100, n_jobs=-1,
                                         oob_score=False, random_state=123,
                                         verbose=0, warm_start=False))],
          verbose=False),
 'churn_modelling_pipeline.pkl')
```

LOAD THE PIPELINE

```
# load pipeline
loaded_best_pipeline = load_model('churn_modelling_pipeline_v3')
loaded_best_pipeline
```



WEB APP

Customer Churn Prediction Tool

http://3.99.190.226:8501/

EDA

Navigation

Choose an Activity

- (•) Exploratory Data Analysis
- ( ) Prediction

Customer Churn Prediction Tool

Exploratory Data Analysis (EDA)

Explore the dataset to understand the distribution of various features and their relation to customer churn.

Preview Dataset

Show Descriptive Statistics

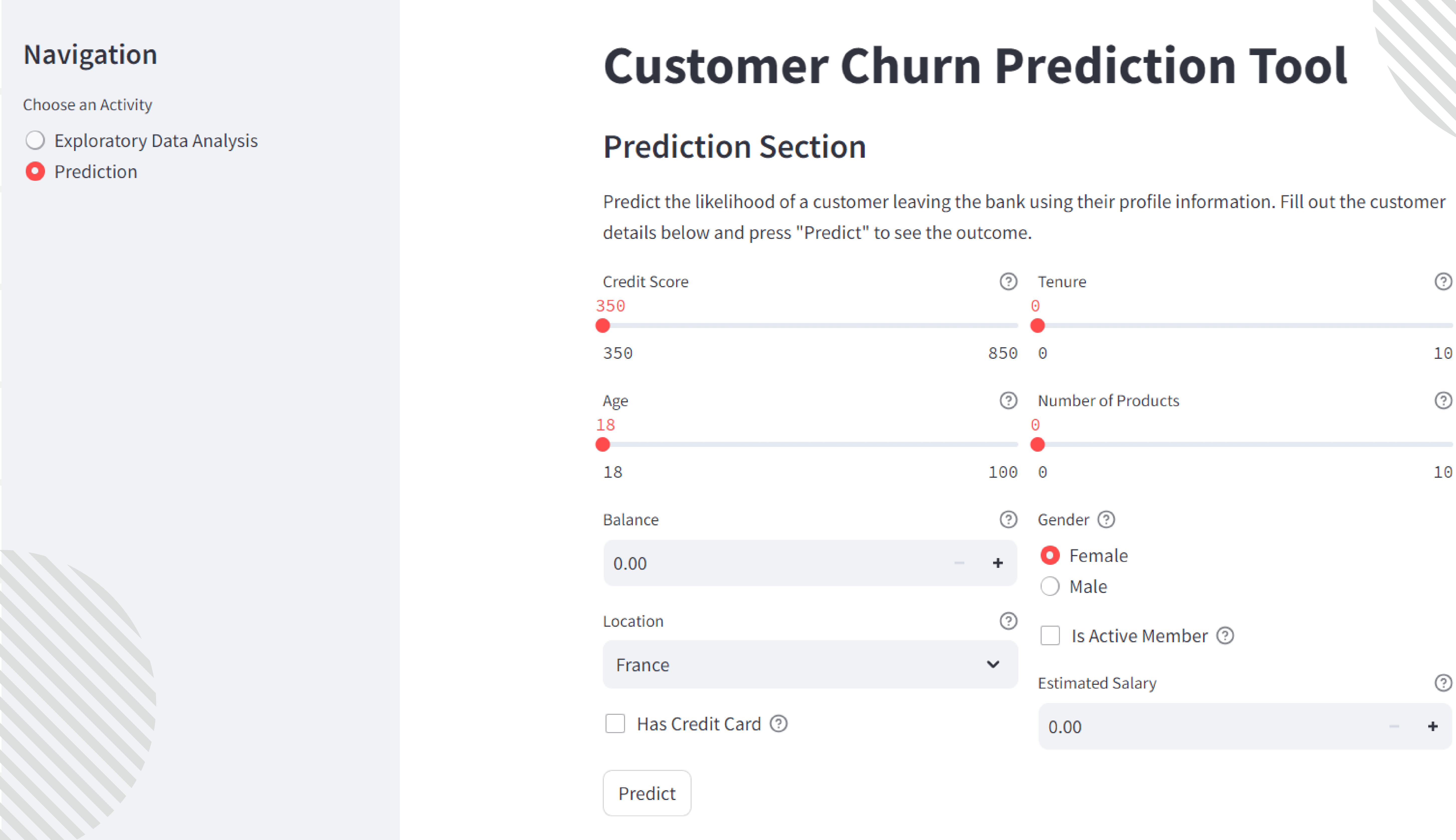
Show Dataset Shape

Show Value Counts for a Column

Show Correlation Matrix Heatmap

Show Age Distribution

PREDICTION


Navigation
Choose an Activity
Exploratory Data Analysis
Prediction
Customer Churn Prediction Tool
Prediction Section
Predict the likelihood of a customer leaving the bank using their profile information. Fill out the customer details below and press "Predict" to see the outcome.
Credit Score
350
350
850
Tenure
0
0
10
Age
18
18
100
Number of Products
0
0
10
Balance
0.00
Gender
Female
Male
Location
France
Is Active Member
Estimated Salary
0.00
Has Credit Card
Predict

FUTURE
WEB APP IMPROVEMENT

CONCLUSION

http://3.99.190.226:8501/

THANK YOU

Presentation by Osear Okinga and Suha Islaih