Introduction to Data Science Assignment

2023/24

Professors: Pedro G. Ferreira, Alípio Jorge

Gonçalo Rocha
up201707455
Manuela Pinheiro
up200400020
Sofia Coelho
up202103646

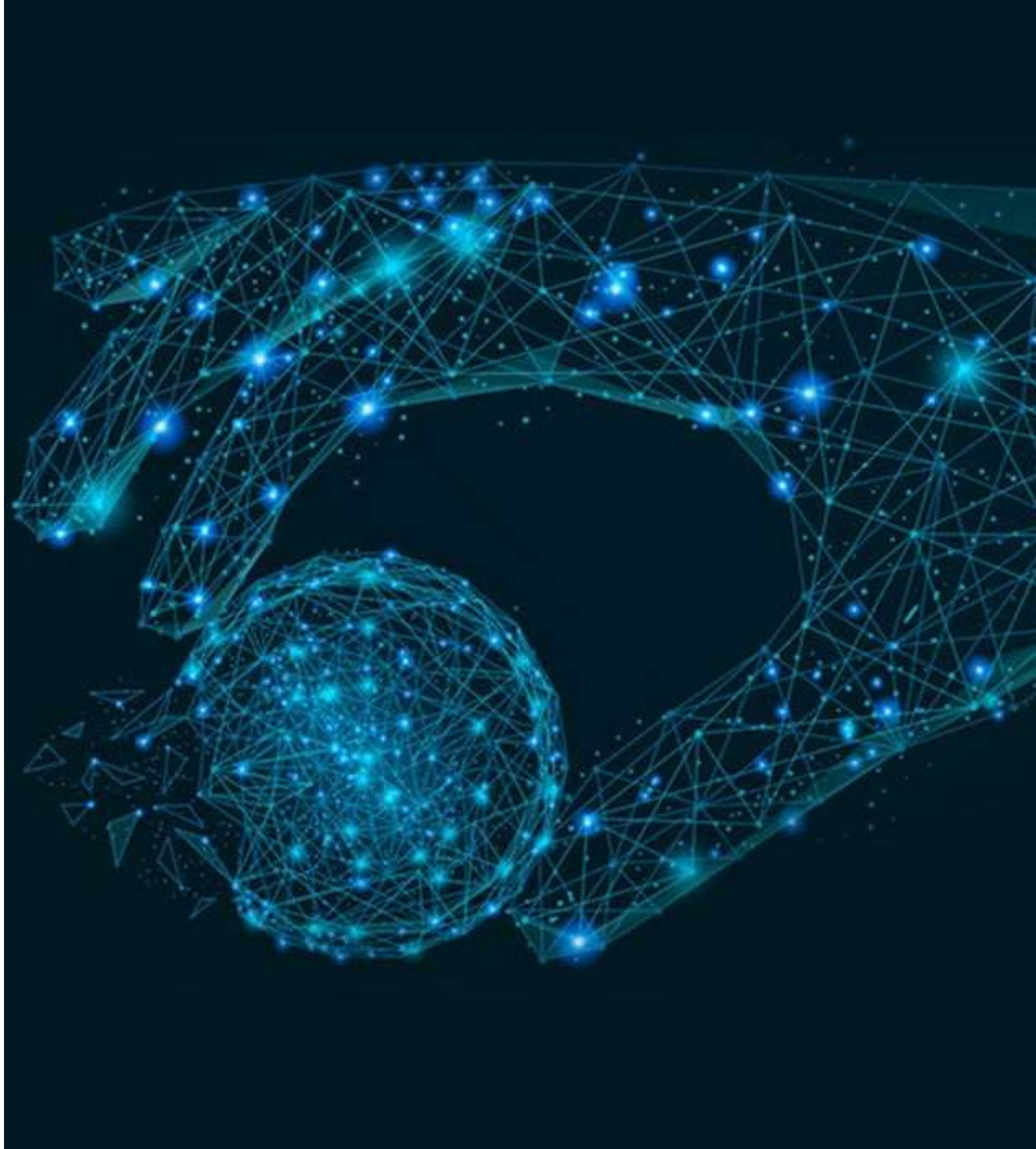


Table of Contents

Business Understanding

The Business problem

The telecommunications company wants to make better business decisions to decrease the rate at which customers cease their subscriptions.
Implementing targeted retention efforts for potential churners is expected to be more cost-effective than acquiring new customers.

The Business criteria:

Reducing costs by increasing customer retention.

The Machine Learning problem

A classification problem: given a dataset of different variables of a client finding a classification model that can predict if the client is going to cease their subscription or not.

The Machine Learning criteria:

1. Achieve **a high accuracy** rate in predicting customer churn.
2. Balance **precision and recall** to effectively identify potential churners without excessive false positives.
3. Develop a **robust model** that generalizes well to new data and changing patterns.

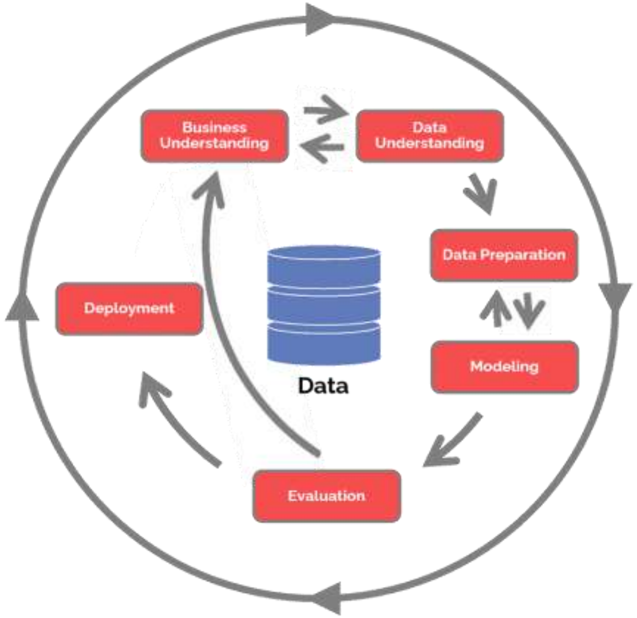
Problem Definition

Features

- Account length
- International plan
- Voicemail plan
- Number mail messages
- Total day minutes
- total day calls
- total day charge
- Total eve minutes
- Total eve calls
- Total eve charge
- Total night minutes
- Total night calls
- Total night charge
- Total intl minutes
- Total intl calls
- Total intl charge
- Customer service calls

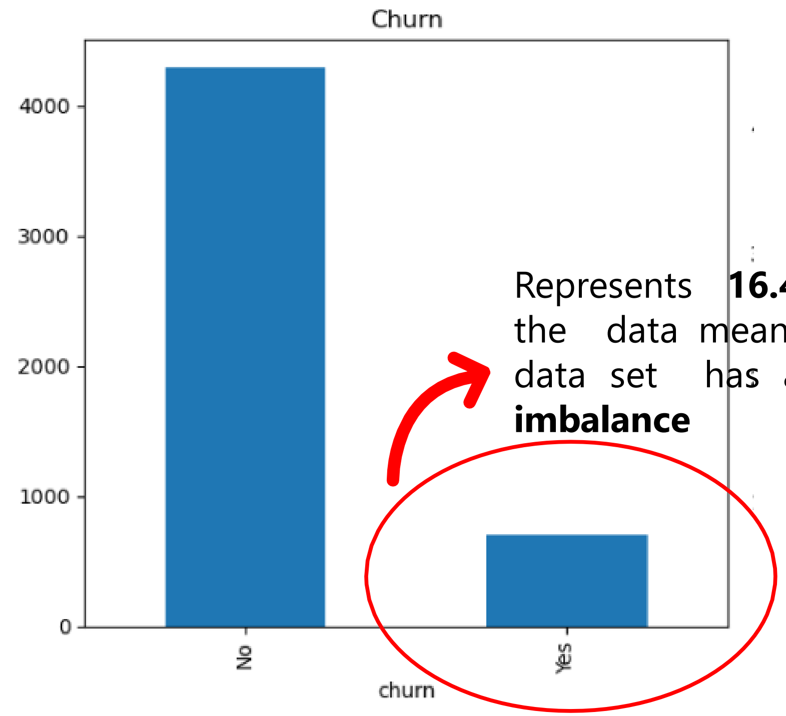
Targets

"Churn"

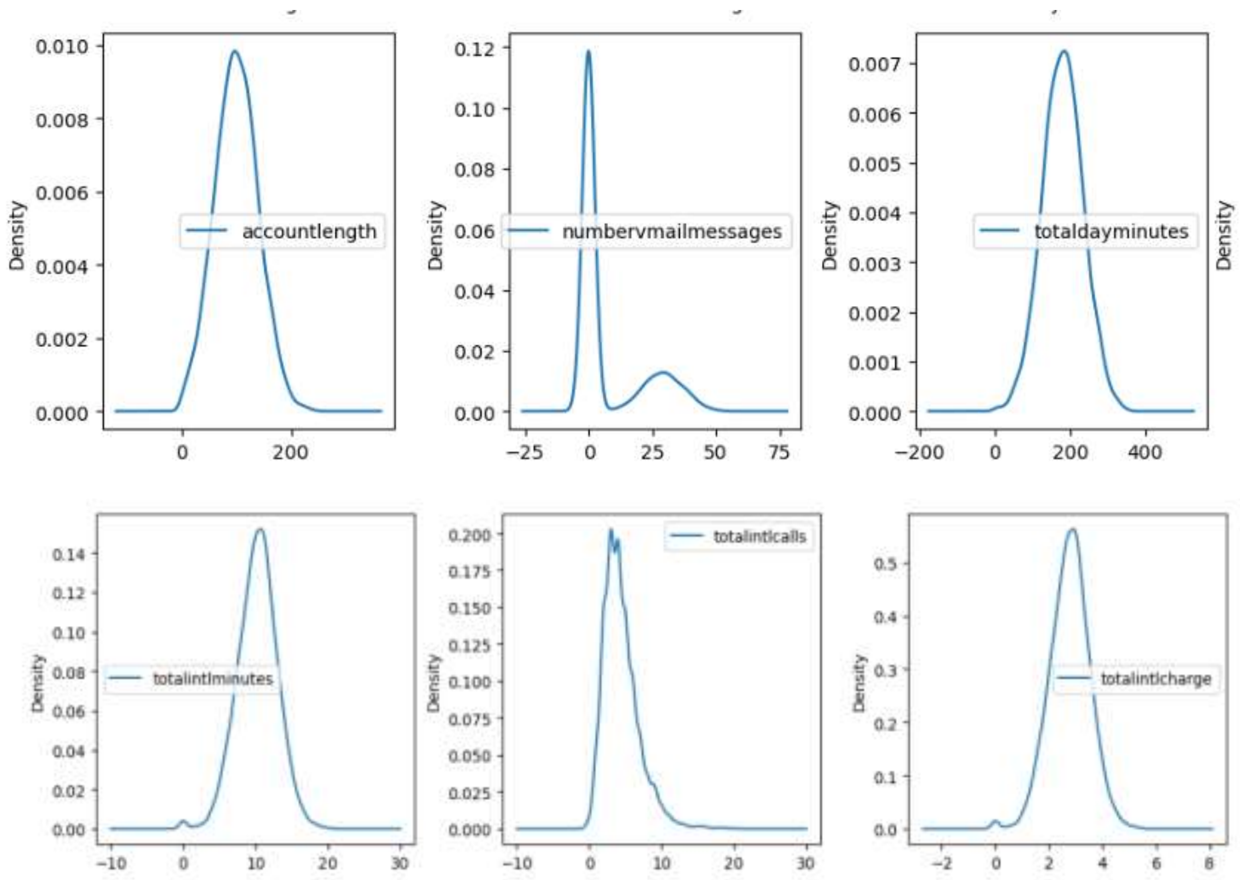


Data Understanding

Exploring the Variables:



Represents **16.46%** of the data meaning this data set has **a small imbalance**



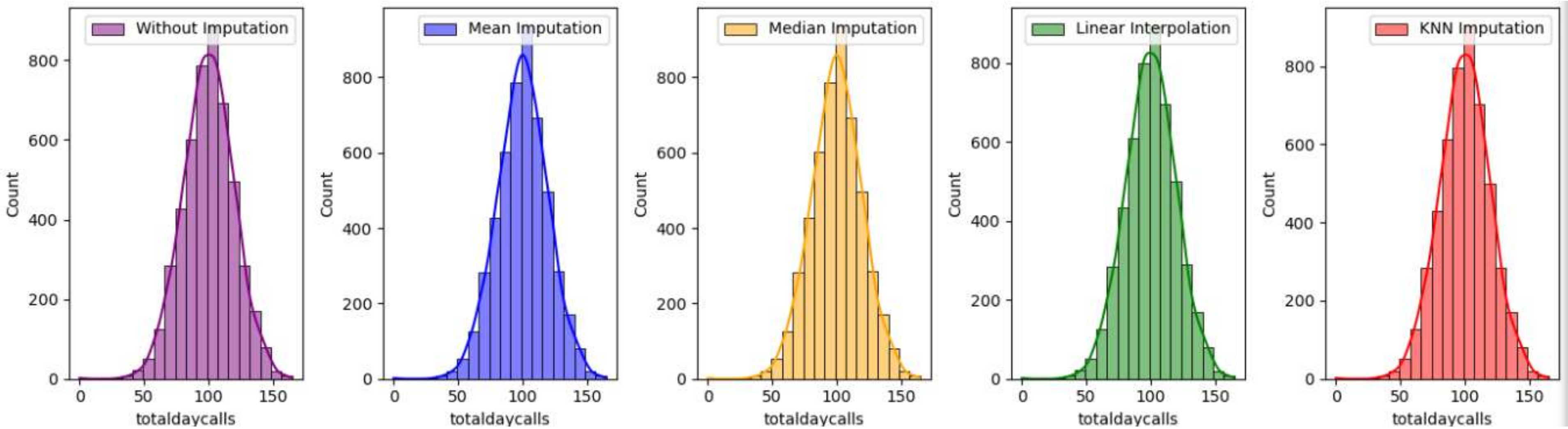
*According to: Dados desequilibrados |

Data Preparation

Data Enconding: Churn", "Internationplan" and "Voicemailplan **were encoded into binary variables.**

Missing values: We did imputation of the missing values using a **knn approach**, considering that it's a method that can handle both continuous and categorical variables and does not require assumptions about the distribution of data.

Discretising : For continuous variables with a somewhat normal distribution, we decided to do equal-width binning, while for continuous variables with non-normal distributions, we chose equal-frequency binning.



Data Preparation

Detecting Outliers:

Interquartile Range method

Z-score

Median Absolute Deviation

For all the variables that presented a normal distribution we identified **the common outliers for the three methods**, while for the variables with a skewed distribution we only used the IQR and MAD methods.

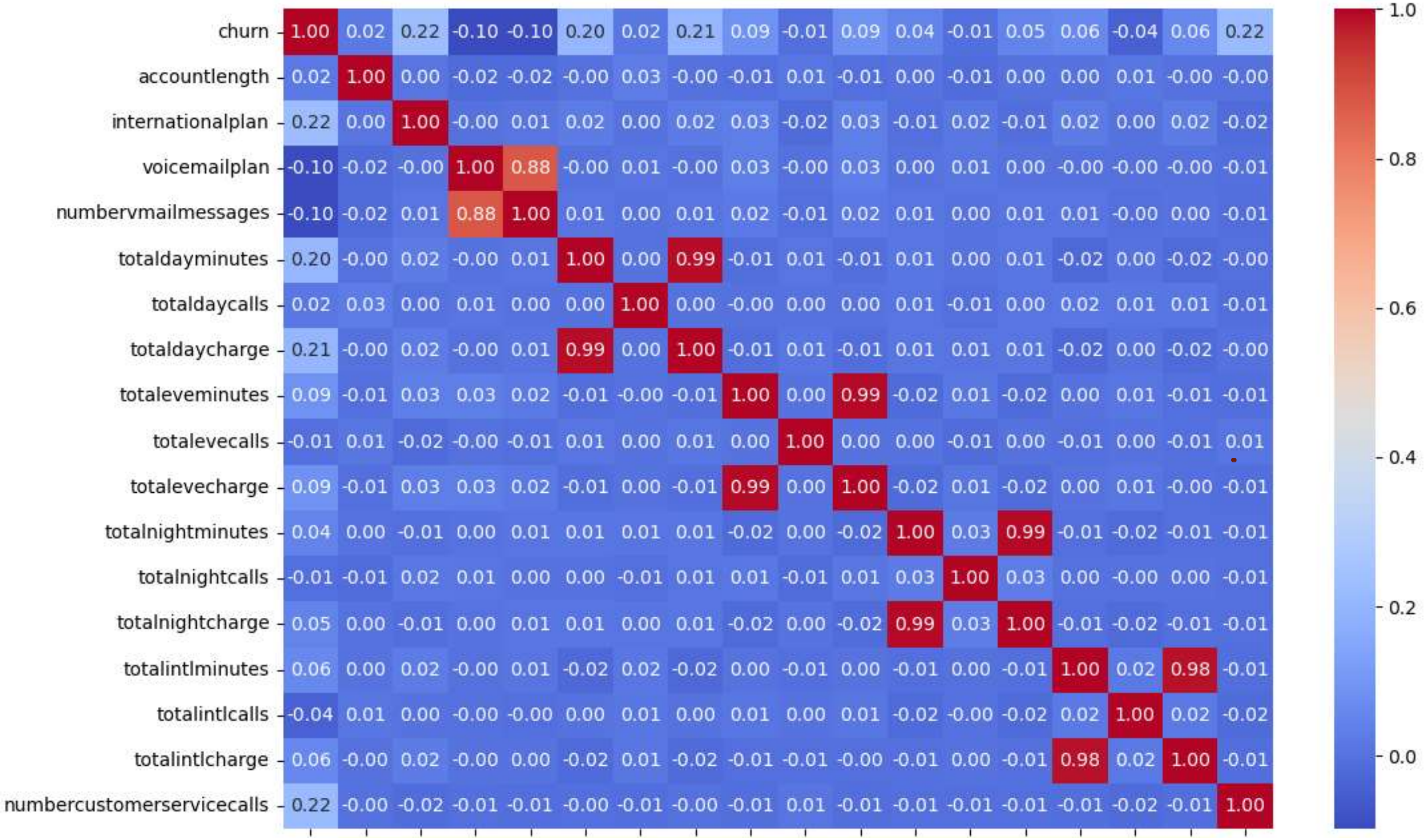
```
Common outliers for 'accountlength':
{416, 1408, 4260, 4389, 4395, 1551, 3216, 817, 1886, 1751, 4379, 4798}
Number of common outliers for 'accountlength': 12

Number of common outliers for 'numbercustomerservicecalls': 145
```

Data Preparation

Feature selection:

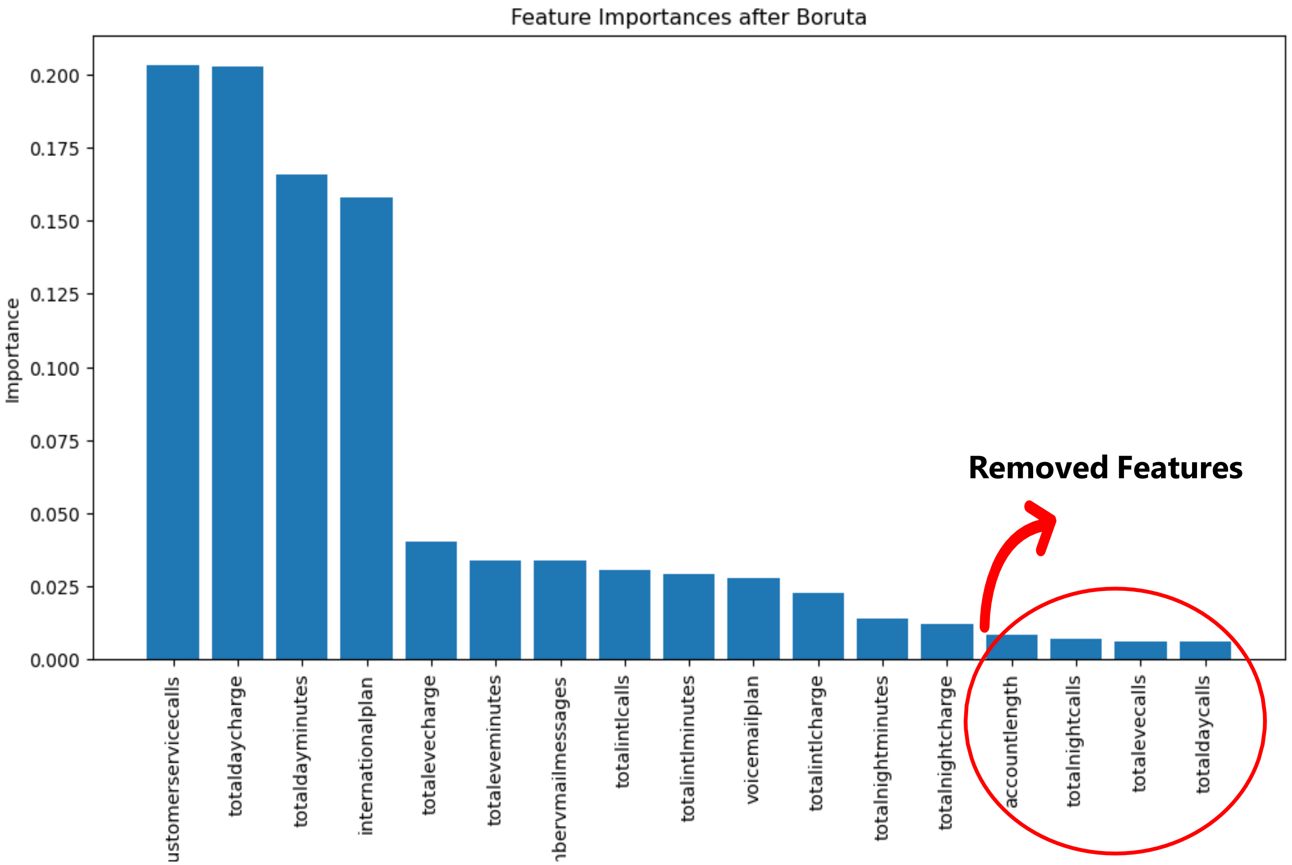
1. **Identifying Redundant Features** through a **Correlation Matrix.**
2. **A**pplying Ensemble Methods**: used the Boruta Method** to feature ranking from a random forest approach.
3. **Recursive Feature Elimination (RFE).**

Correlation Matrix

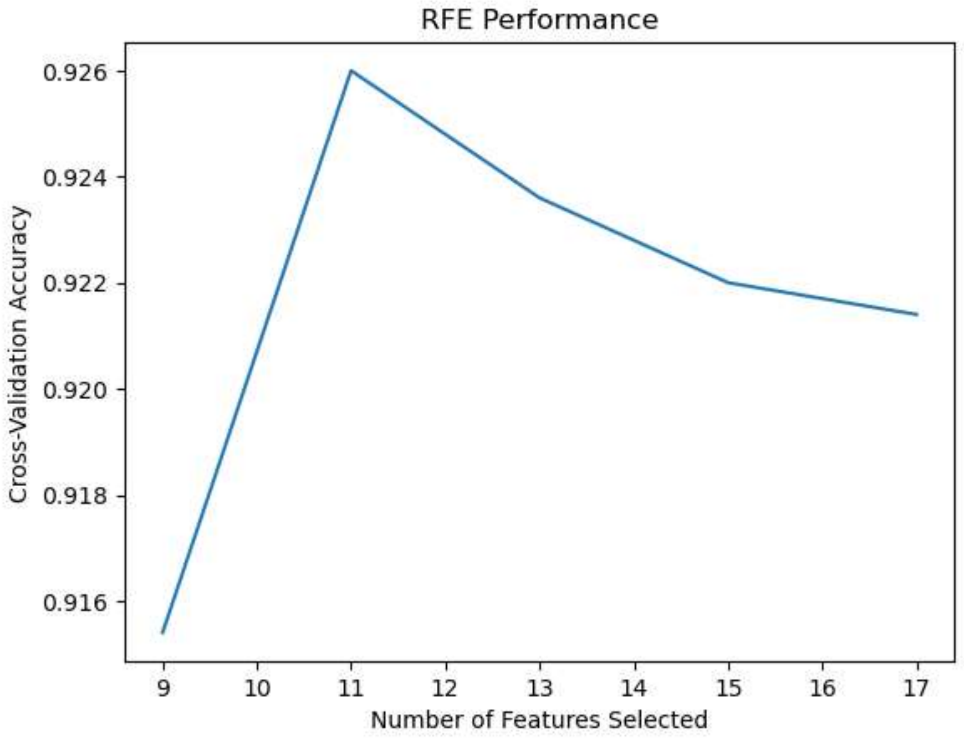


Data Preparation

Boruta Method:



Recursive Feature Elimination:




```
Removed Features: Index(['accountlength', 'totaldaycalls', 'totalevecalls', 'totalnightminutes',
       'totalnightcalls', 'totalnightcharge'],
```

Modeling

Total data set

Synthetic Minority Over-sampling Technique (SMOTE)
(oversampling churn = Yes)

Random Under-Sampling
(undersampling churn = No)

Balanced data set

Data shuffle and split

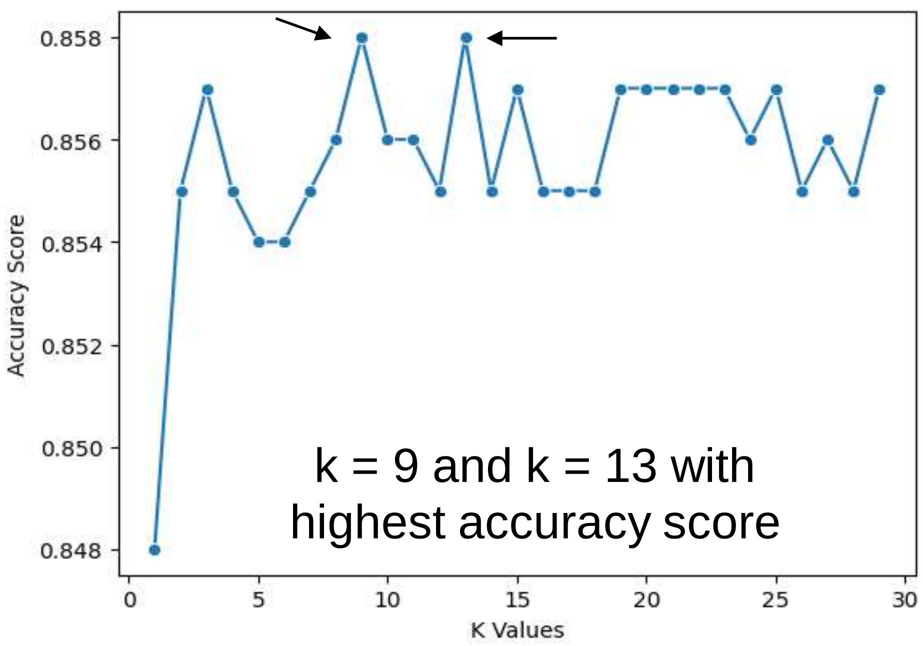
Train data set (80%)

Test data set (20%)

Modeling

Nearest neighbor:

- training the model with k=3: 0.81 accuracy
- K value optimizing: k = 9



Classification Report for X_train_balanced data:

	precision	recall	f1-score	support
0.0	0.95	0.81	0.87	855
1.0	0.40	0.77	0.53	145
accuracy			0.80	1000
macro avg	0.68	0.79	0.70	1000
weighted avg	0.87	0.80	0.82	1000

Modeling

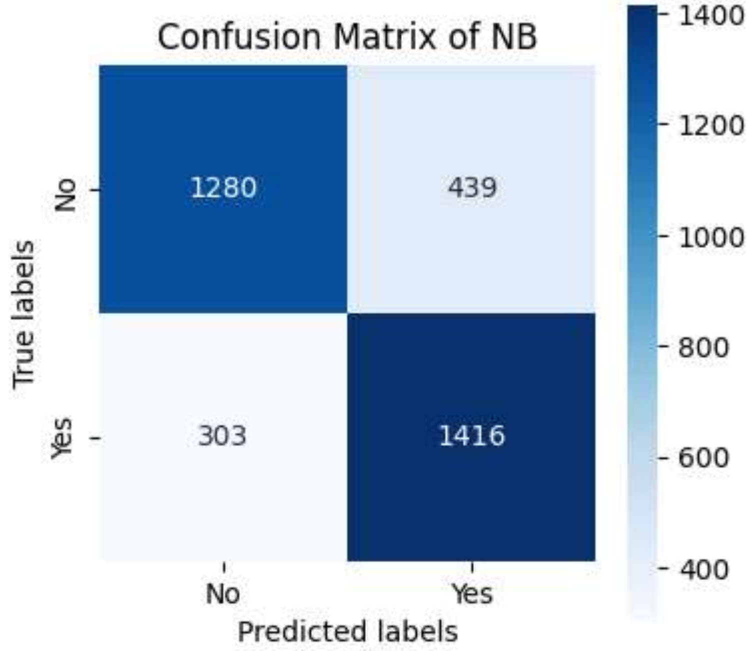
Bayesian Classifier:

MixedNB method, that handles both continuous and categorical variables.

Accuracy on X_train_balanced data: 0.78

Classification Report for X_train_balanced data:

	precision	recall	f1-score	support
0	0.81	0.74	0.78	1719
1	0.76	0.82	0.79	1719
accuracy			0.78	3438
macro avg	0.79	0.78	0.78	3438
weighted avg	0.79	0.78	0.78	3438



Modeling

Decision Tree:

- Without Pruning: 100% accuracy on the training data but lower on the test data, possible indicating **overfitting**.

- Pruning the decision tree by varying hyperparameters **max depth** and **accuracy**: accuracy on the test set peaks at a **max depth of 7**

- Cross validation approach and tunning using the **GridSearchCV.**

```
DecisionTreeClassifier
DecisionTreeClassifier(ccp_alpha=0.001, max_depth=16, random_state=0)
```

Classification Report for test data:

	precision	recall	f1-score	support
0.0	0.97	0.93	0.95	855
1.0	0.65	0.82	0.73	145
accuracy			0.91	1000
macro avg	0.81	0.87	0.84	1000
weighted avg	0.92	0.91	0.91	1000

Modeling

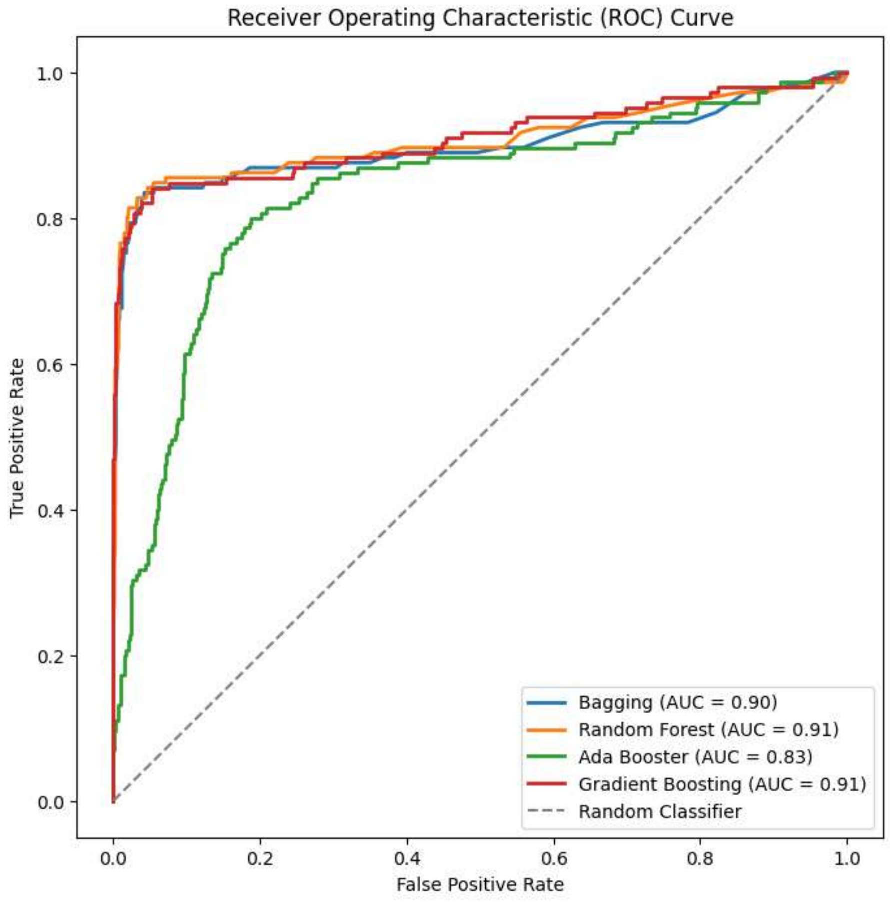
Tree Ensembles Methods:

- Bagging, Random Forest and Boosting models

- Fine-tuned the number of estimators with a smaller set of data

Classification Report for test data:

	precision	recall	f1-score	support
0.0	0.97	0.96	0.96	855
1.0	0.77	0.82	0.80	145
accuracy			0.94	1000
macro avg	0.87	0.89	0.88	1000
weighted avg	0.94	0.94	0.94	1000



Modeling

Support vector machines:

- Tested with kernel linear and RBF functions and different C hyperparameter

C= 100 the best

Accuracy on X_train_balanced data: 0.80

Classification Report for X_train_balanced data:

	precision	recall	f1-score	support
0	0.82	0.76	0.79	1719
1	0.78	0.84	0.80	1719
accuracy			0.80	3438
macro avg	0.80	0.80	0.80	3438
weighted avg	0.80	0.80	0.80	3438

Accuracy on X_train_balanced data: 1.00

Classification Report for X_train_balanced data:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1719
1	1.00	1.00	1.00	1719
accuracy			1.00	3438
macro avg	1.00	1.00	1.00	3438
weighted avg	1.00	1.00	1.00	3438

Overffiting

Modeling

Neural Network Classifier:

- hyperparameter grid to search for the best hidden layer sizes.

```
Classification Report for X_train_balanced data:
              precision    recall  f1-score   support

         0.0       0.94      0.81      0.87       855
         1.0       0.39      0.70      0.50       145

    accuracy                           0.80      1000
   macro avg       0.66      0.75      0.68      1000
weighted avg       0.86      0.80      0.82      1000
```

```
▸ GridSearchCV
▸ estimator: MLPClassifier
    ▸ MLPClassifier
```

```
print('Best parameters found:\n', clf.best_params_)
```

```
Best parameters found:
 {'activation': 'tanh', 'alpha': 0.05, 'hidden_layer_sizes': (200,), 'learning_rate': 'adaptive', 'solver': 'adam'}
```

Results:

	Model	Data	Accuracy	Recall Score	Precision Score	F1 Score
0	Decision Tree	Balanced	0.919	0.786207	0.695122	0.737864
1	Decision Tree	Unbalanced	0.941	0.668966	0.898148	0.766798
2	Bagging	Balanced	0.932	0.834483	0.733333	0.780645
3	Bagging	Unbalanced	0.949	0.731034	0.898305	0.806084
4	Random Forest	Balanced	0.932	0.834483	0.733333	0.780645
5	Random Forest	Unbalanced	0.953	0.751724	0.908333	0.822642
6	Ada Booster	Balanced	0.851	0.634483	0.489362	0.552553
7	Ada Booster	Unbalanced	0.874	0.365517	0.609195	0.456897
8	Gradient Boosting	Balanced	0.923	0.841379	0.693182	0.760125

	Model	Data	Accuracy	Recall Score	Precision Score	F1 Score
0	KNN	Balanced	0.80	0.80	0.87	0.82
1	KNN	Unbalanced	0.89	0.42	0.77	0.54
2	NeuralNetwork	Balanced	0.80	0.86	0.80	0.82
3	NeuralNetwork	Unbalanced	0.80	0.80	0.86	0.82
4	Bayes	Balanced	0.78	0.79	0.78	0.78
5	Bayes	Unbalanced	0.87	0.87	0.87	0.87
6	SVMLinear	Balanced	0.80	0.80	0.80	0.80
7	SVMLinear	Unbalanced	0.85	0.85	0.73	0.79

Model	Data	Accuracy	Recall	Precision	F1- Score
9 Gradient Boosting	Unbalanced	0.95	0.99	0.96	0.99

Results:

Best Overall Model:

Based on accuracy and F1 scores, the model with the best performance was the **Gradient Boosting Tree Ensemble model**, both in dealing with balanced and unbalanced data. However, both tree based algorithms (Decision Tree and Tree Ensemble) were high performing.

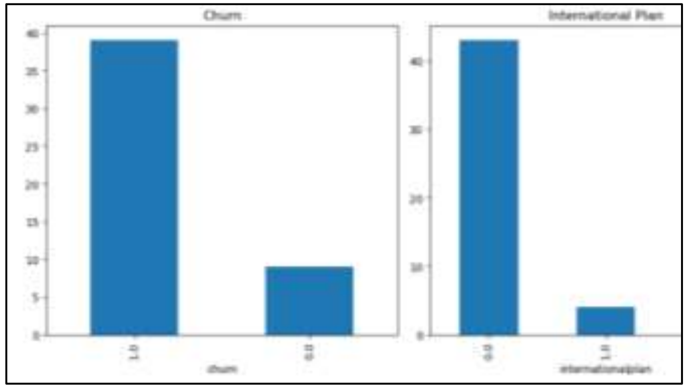
Combining Models?

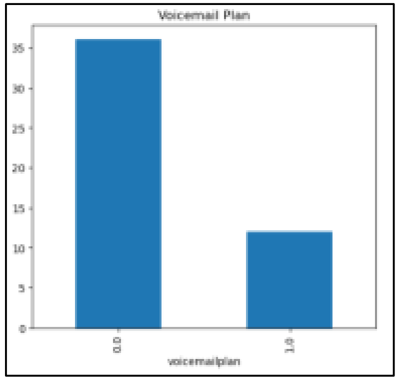
	Data	Accuracy	Recall Score	Precision Score	F1 Score
0	Balanced	0.932	0.834483	0.733333	0.780645
1	Unbalanced	0.956	0.751724	0.931624	0.832061

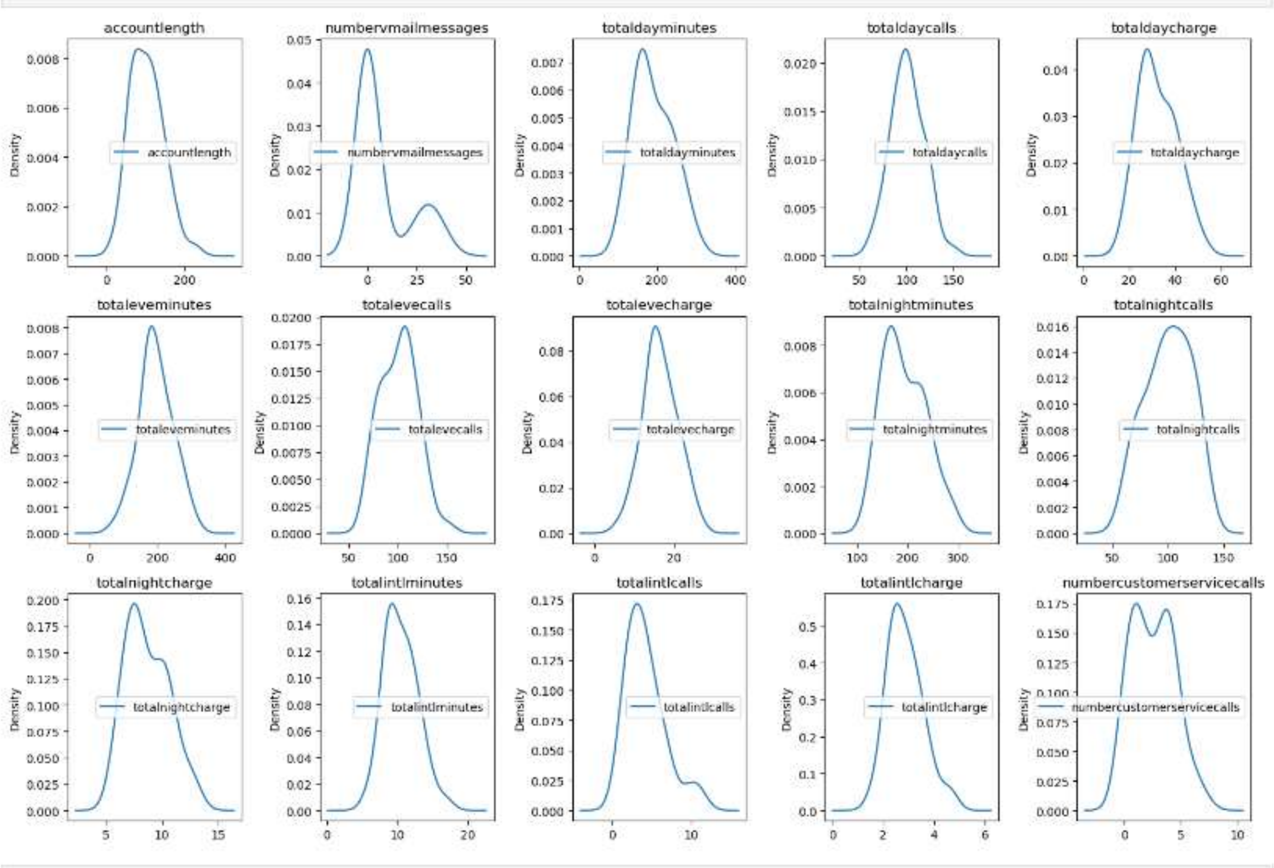
Model	Data	Accuracy	Recall	Precision	F1- Score
9 Gradient Boosting	Unbalanced	0.95	0.99	0.96	0.99

Results:

Mis-classified variables:







Summary and Model Recommendations

The best prediction model, when trained either with a balanced or unbalanced dataset, was the **Tree Ensemble.** With a high **accuracy score of 95%**, and a **F1 score of 0,82** has the best overall performance and the best at predicting churn.

Gradient Boosting Tree Ensemble

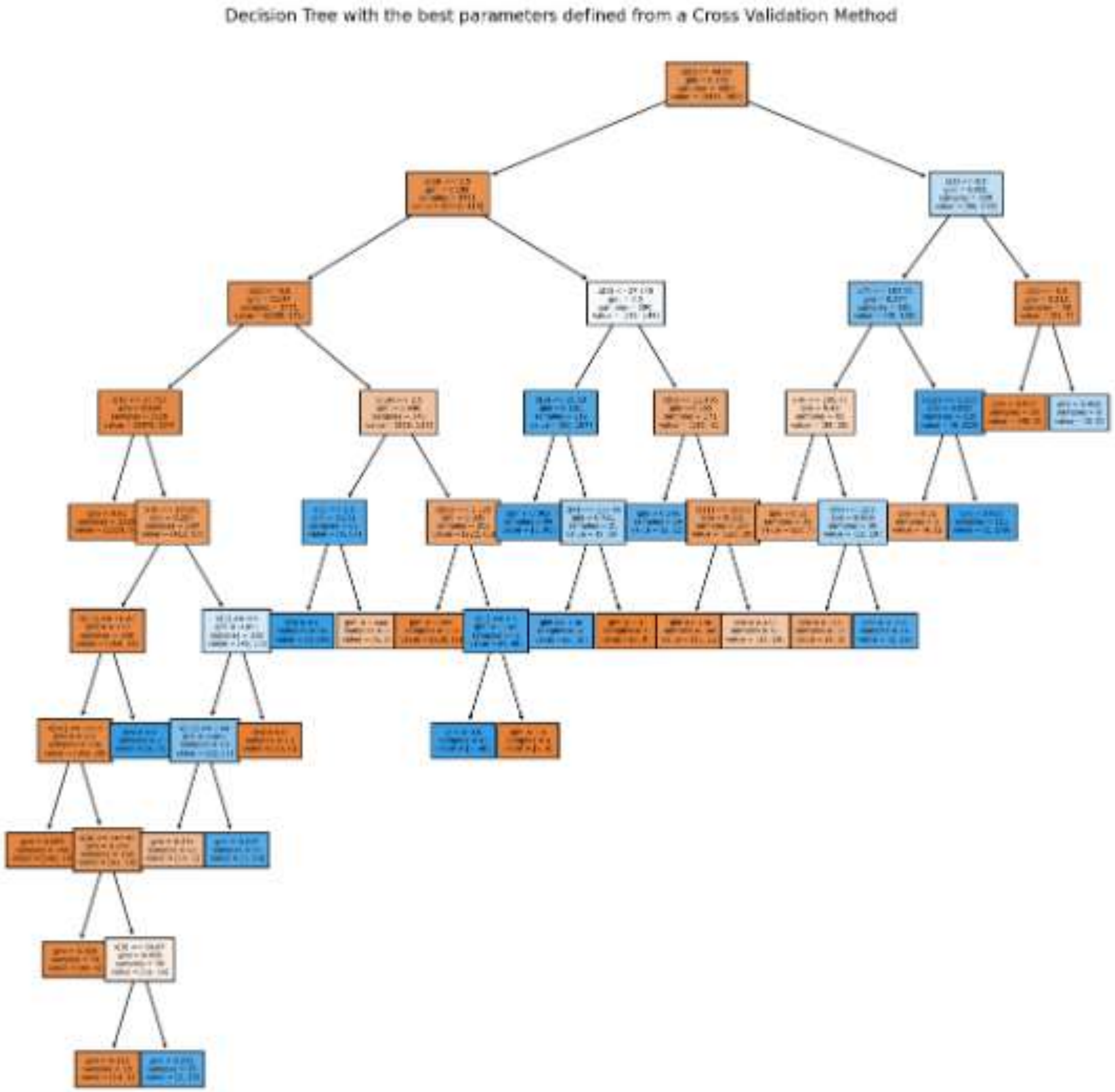
Classification Report for test data:

	precision	recall	f1-score	support
0	0.96	0.99	0.97	855
1	0.91	0.75	0.82	145
accuracy			0.95	1000
macro avg	0.93	0.87	0.90	1000
weighted avg	0.95	0.95	0.95	1000

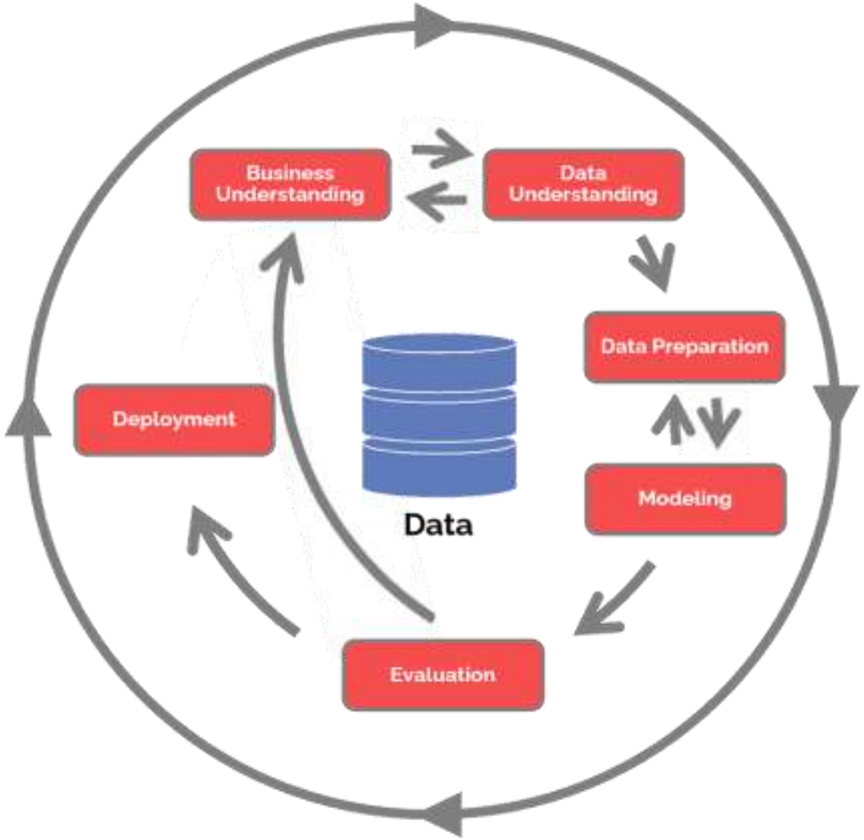
Model Recommendations

However, the **decision tree model** is probably the easiest to explain to the marketing team of a company, being the **easiest to understand** because they can be **represented visually** and are very **intuitive**.

Decision Tree



Model Recommendations



The **Machine Learning criteria** was successfully achieved, since several of the created models have a high accuracy rate in predicting customer churn, with high precision and recall. The **business success criteria** can only be evaluated by using the model to implement changes to reduce costs by increasing customer retention.

Thank you!

Gonçalo Rocha
up201707455

Manuela Pinheiro

up200400020

Sofia Coelho

up202103646