<Banking Analysis with Predicted Modeling>

Summary

Our team of data scientists has been tasked with developing an effective telemarketing strategy to sell term deposit accounts for a Portuguese bank. This bank has been conducting marketing campaigns, but it has not been effective. Our goal then was to develop an effective marketing strategy by using machine learning and to collect correlated attributes to increase the possibility to get more subscriptions.

The dataset consisted of 4521 rows and 17 attributes – 7 of which were quantitative and 10 which were qualitative including the target attribute. The dataset was prepared by looking at the attribute types and analyzing the mean, max, min and std as well as extreme values. Furthermore, a correlation matrix was used to determine which attributes were most strongly related to the class attribute. This was then taken into account when choosing which attributes to use for the machine learning techniques. The dataset had no missing data, but it contained a lot of outliers, and it was imbalanced.

The two machine learning methods were used in this analysis – Decision Tree and Naïve Bayes. These methods were used for all attributes and the selected attributes (Age, Duration, and Poutcome). The Decision Tree was created using Python weka and Sklearn, while the Naïve Bayes method was applied using Python. The performance metrics – accuracy, recall, and precision – were used to evaluate the accuracy of the models.

Data Preparation

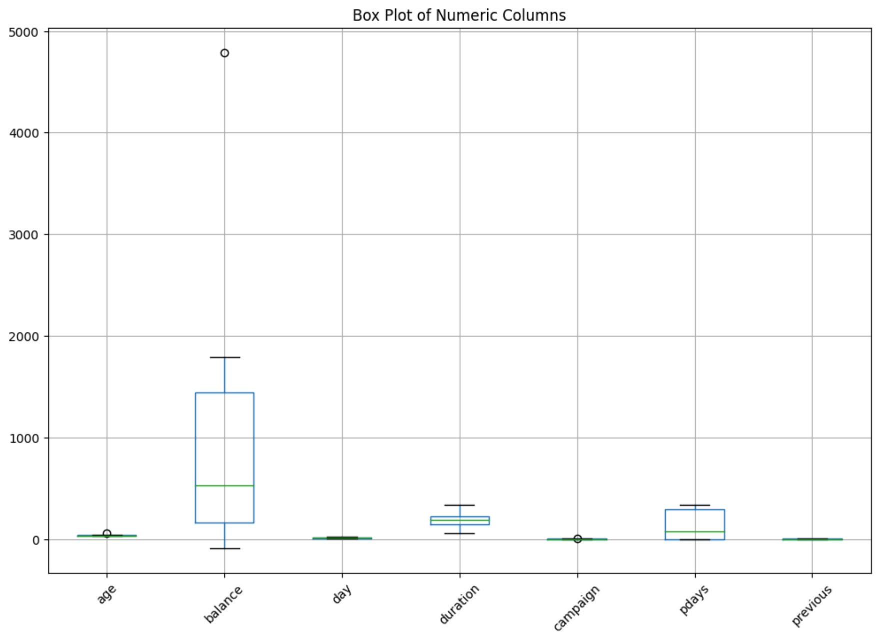
Column Name	Attribute type	Description
Age	quantitative	Age of the customer (numeric).
Job	nominal	Type of job (qualitative).
Marital	nominal	Marital status (qualitative).
Education	ordinal	Education of the customer (qualitative).
Default	nominal	Shows whether the customer has credit in default or not (qualitative).
Balance	quantitative	Average yearly balance in Euros (numeric).
Housing	nominal	Shows whether the customer has a housing loan or not (qualitative).

Loan	qualitative	Shows whether the customer has a personal loan or not (qualitative/categorical).
Contact	nominal	Shows how the last contact for marketing campaign has been made (qualitative)
Day	quantitative	Day: Shows on which day of the month the last time a customer was contacted (numeric).
Month	ordinal	Month: Shows on which month of the year the last time a customer was contacted (qualitative).
Duration	quantitative	Shows the last contact duration in seconds (numeric).
Campaign	quantitative	Number of contacts performed during the marketing campaign and for this customer (numeric).
Pdays	quantitative	Pdays: Number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means client was not previously contacted).
Previous	quantitative	Previous: Number of contacts performed before this campaign and for this client (numeric).
Poutcome	ordinal	Outcome of the previous marketing campaign (qualitative).
Y	nominal	Class attribute showing whether the client has subscribed a term deposit or not (binary: "yes","no").

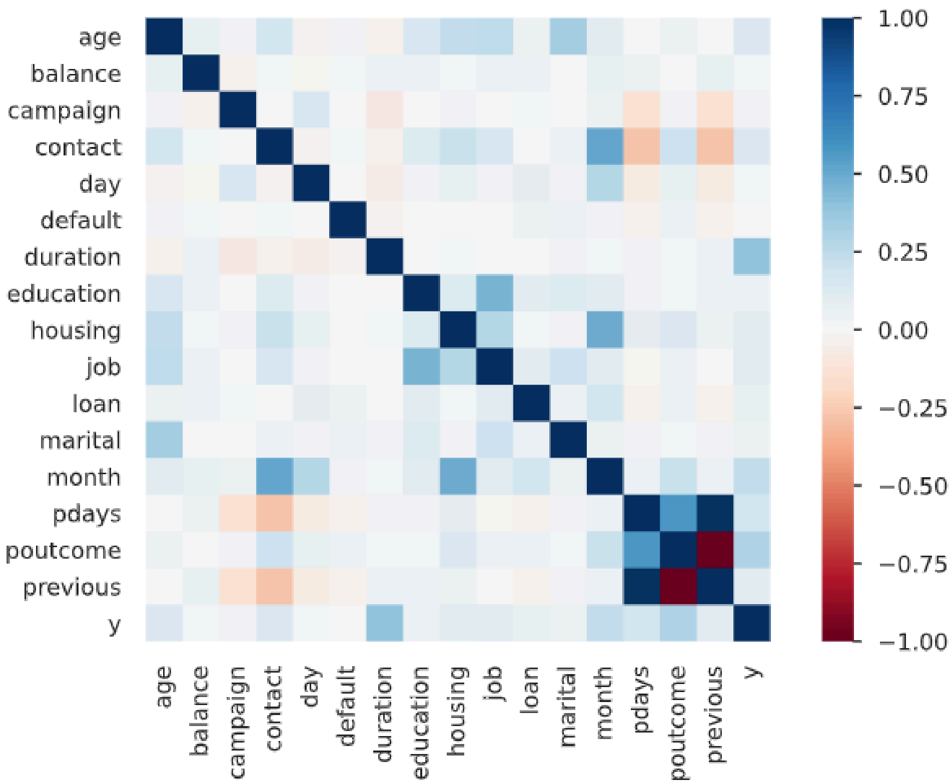
Table of max, min, mean and standard deviation of attributes:

	Age	Balance	Day	Duration	Campaign	Pdays	Previous
Max	87.000000	71188.000000	31.000000	3025.000000	50.000000	871.000000	25.000000
Min	19.000000	-3313.000000	1.000000	4.000000	1.000000	-1.000000	0.000000

Mean	41.170095	1422.657819	15.915284	263.961292	2.793630	39.766645	0.542579
Std	10.576211	3009.638142	8.247667	259.856633	3.109807	100.121124	1.693562



Correlation Matrix:



Correlations:

- Duration is the most correlated with y. poutcome, month. contact and age are also slightly correlated.
- Previous is highly positively correlated with pdays.
- poutcome and previous are highly negatively correlated with each-other

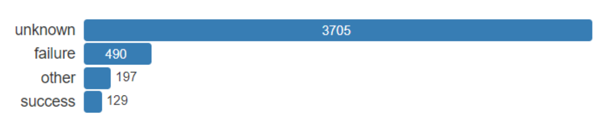
Based on the BestFIRst algorithm in Weka, we decided to filter for the selected attributes of age, duration, poutcome and the class attribute.

Bar Charts/Histograms of Selected Attributes:

poutcome
Categorical

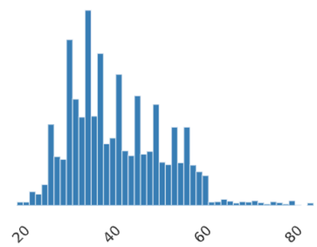
HIGH CORRELATION IMBALANCE

Distinct	4
Distinct (%)	0.1%
Missing	0
Missing (%)	0.0%
Memory size	35.4 KiB



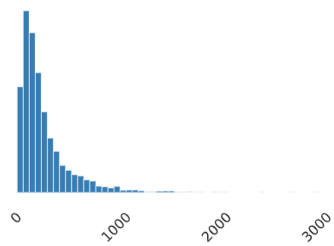
age
Real number (ℝ)

Distinct	67	Minimum	19
Distinct (%)	1.5%	Maximum	87
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	41.170095	Memory size	35.4 KiB



duration
Real number (ℝ)

Distinct	875	Minimum	4
Distinct (%)	19.4%	Maximum	3025
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	263.96129	Memory size	35.4 KiB



y
Boolean

Distinct	2
Distinct (%)	< 0.1%
Missing	0
Missing (%)	0.0%
Memory size	4.5 KiB

Predictive Modeling/Classification

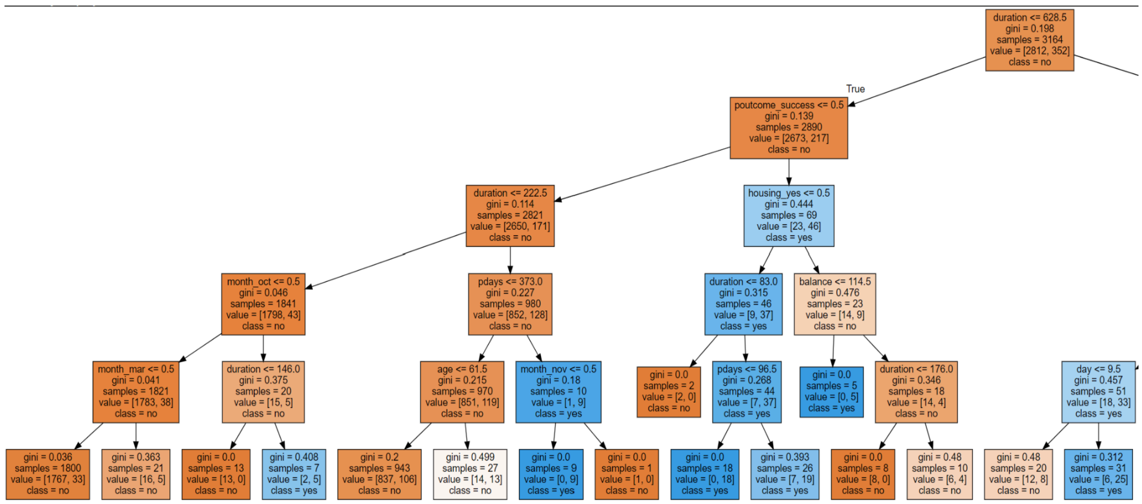
Predictive Modeling/Classification

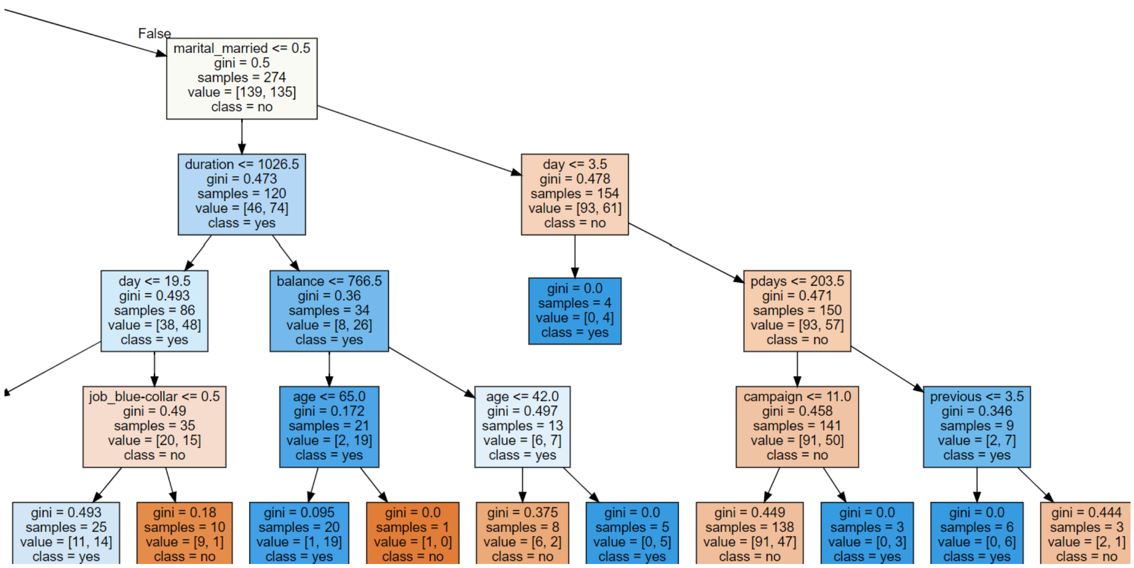
Classification is a supervised machine learning technique used to train models which predict classes based on the training from the training dataset. In this project, the Decision Tree and Naïve Bayes classification techniques were used to train, test, and evaluate the performance of the dataset. The train-test split method was used for the classification.

Classification using Decision Tree

A Decision Tree is a classification technique that uses attributes in a dataset to predict a class based on decisions. Decision trees are referred to as classification for qualitative attributes or regression trees for continuous variables (Gupta, 2017). For this project Python Weka Package and Sklearn were used to create the decision tree.

The current sklearn decision tree graph has depth of 5 and is a decision tree based on all attributes.





Classification using Naive Bayes

Naive Bayes: Naive Bayes is a machine learning technique. For this project we used the Weka Python Package to complete a Naive Bayes from the Banking data.

Summary Table:

<Baseline>

	Position 0 Precision	Position 0 Recall	Position 1 Precision	Position 1 Recall
Decision Tree	0.9180588703261734	0.9632721202003339	0.5555555555555556	0.34810126582278483
Naive Bayes	0.9249793899422918	0.9365609348914858	0.46853146853146854	0.4240506329113924

<Selected Features>

	Position 0 Precision	Position 0 Recall	Position 1 Precision	Position 1 Recall
Decision Tree	0.9246963562753037	0.9532554257095158	0.5371900826446281	0.41139240506329117
Naive Bayes	0.9113029827315542	0.9691151919866444	0.5487804878048781	0.2848101265822785

Machine Learning Comparison: Decision Tree vs Naive Bayes

Evaluation from the perspective of class at position 1 for recall was decided to be the most valuable. This is because we are interested in customers who will say YES to the subscription and we want to recall all potential customers, whether or not it is precise everytime.

Naive Bayes Recall at Position 1 is greater than Decision Tree Recall at Position 1

0.4240506329113924 > 0.34810126582278483

Machine Learning Baseline "all attributes" and "selected features" Comparison

Selected Features: Age, Duration, Poutcome, Yes or No

Machine Learning Recall	Baseline	Selected Features
Naive Bayes	0.4240506329113924	0.2848101265822785
Decision Tree	0.34810126582278483	0.41139240506329117

Highest Recall

Highest Recall with Selected Features

Conclusions and Recommendations

Major findings from different sections

It was found that duration is the most correlated with y. poutcome, month. contact and age are also correlated. Taking this into account and using the BestFIRst algorithm in Weka, we decided to filter for the selected attributes of “age”, “duration”, “poutcome” and the class attribute.

Recall for the decision tree improved by selecting for the attributes “Age”, “Duration”, “Poutcome”, “Yes or No”, but decreased for Naive Bayes. Naive Bayes is more accurate for recall with all attributes. However, the Decision Tree Recall for the Selected Features was increased to a similar recall percentage.

Recommendations:

Because the recall for the decision tree was increased to a similar percentage when selecting for the specific attributes as the Naive Bayes recall for all attributes, it is recommended that when speaking to the company we would recommend using the Decision Tree Machine Learning model using the features: “Age”, “Duration”, “Poutcome”, “Yes or No” to recruit potential subscribers.

As well, since banks would typically have millions of clients, instead of having to go through 16 different attributes and process the dataset using Naive Bayes, it would be faster and have a similar percentage outcome to use Decision Tree on the selected features mentioned above.