Santander-Customer-Transaction-Prediction

Astha Goyal

Edwisor

15th July, 2019

Table of Contents

Introduction

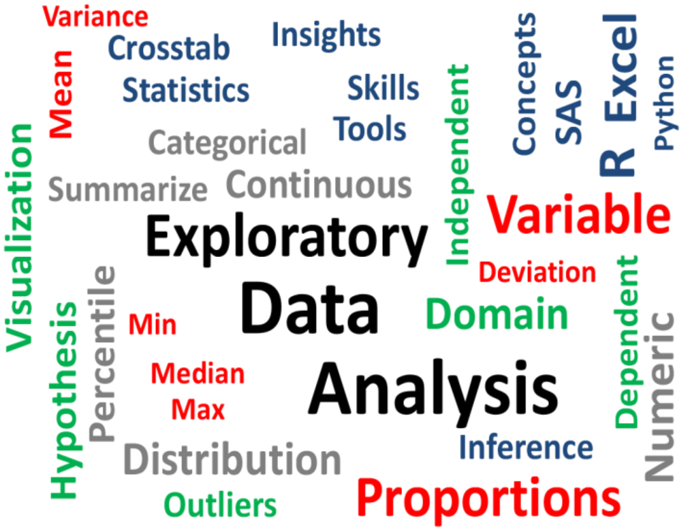
Having adequate customer relations is paramount to success in any service industry. Identifying and analysing your customer’s contentment to improve customer retention can yield many benefits. The longer a client stays with an organisation, the more value he creates. There are higher costs attached to introducing and attracting new customers. The clients also have a better understanding of the organisation and can give positive word-of-mouth promotion. (Colgate et al., 1996) Data mining is essential in this process and this practice is widely applied across industries for instance FMCG retailers (Buckinx and van den Poel, 2005), telecommunications (Mozer et al., 2000) and banking (Clemes et al., 2010) and (Xie et al., 2009).

This paper focuses on Santander Bank, a large corporation focusing principally on the market in the northeast United States. It is the objective to find an appropriate model to predict whether a client will be dissatisfied in the future based on certain characteristics. Having this model in place can ensure that Santander can take proactive steps to improve a customer’s happiness before they would take their business elsewhere.

First the report will discuss related work done on this. Secondly it delves into the data we work with, analyzing groups of variables and individual features to give us insight in what is relevant. Thirdly several cleaning procedures that were employed to lead to better results are outlined. Fourthly we explain the performance measure of this competition and the three models: Logistic Regression, Random Forest and Decision Tree that we utilize to tackle the problem. Lastly the tuning process and results are discussed. We reach an AUC score of 0.8363.

Exploratory Data Analysis

The train set consists of 200000 observations and 201 features plus 1 binary target. There is a large imbalance with 89.95% being 0, meaning the customer was not satisfied and 10.04% being 1, signifying that the customer was satisfied. This is in line with the expectation of the customer satisfaction of a successful bank. There are no missing values in the train or test set. There are only numeric features. No features seem to have substantial outliers.

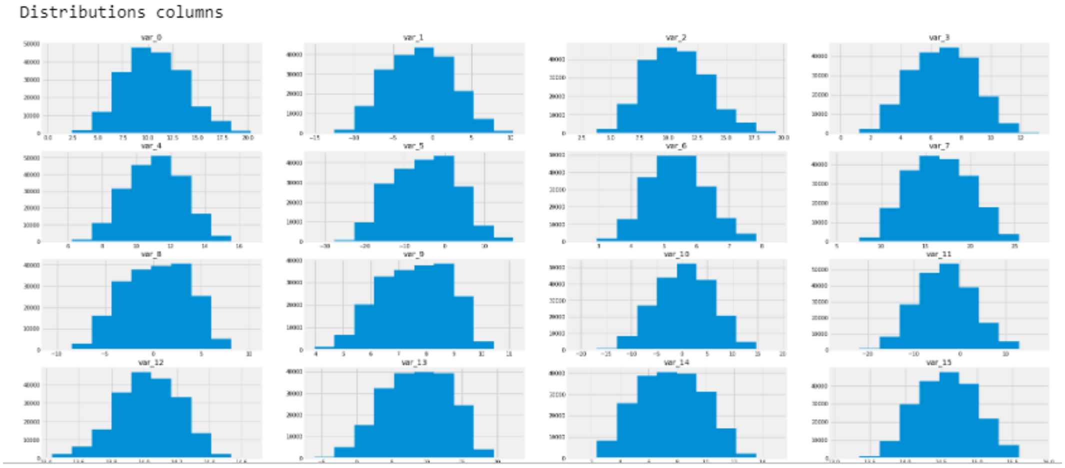


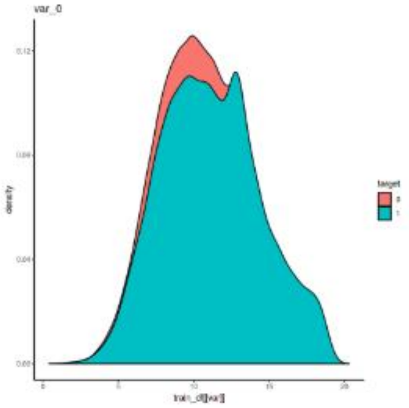
The data set is anonymized dataset containing numeric feature variables, the binary target column, and a string ID_code column.

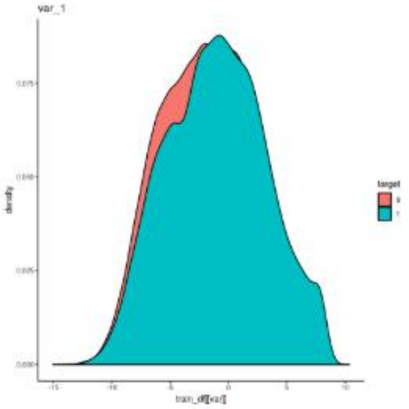
2.1 Individual Features

We need to consider each feature separately.

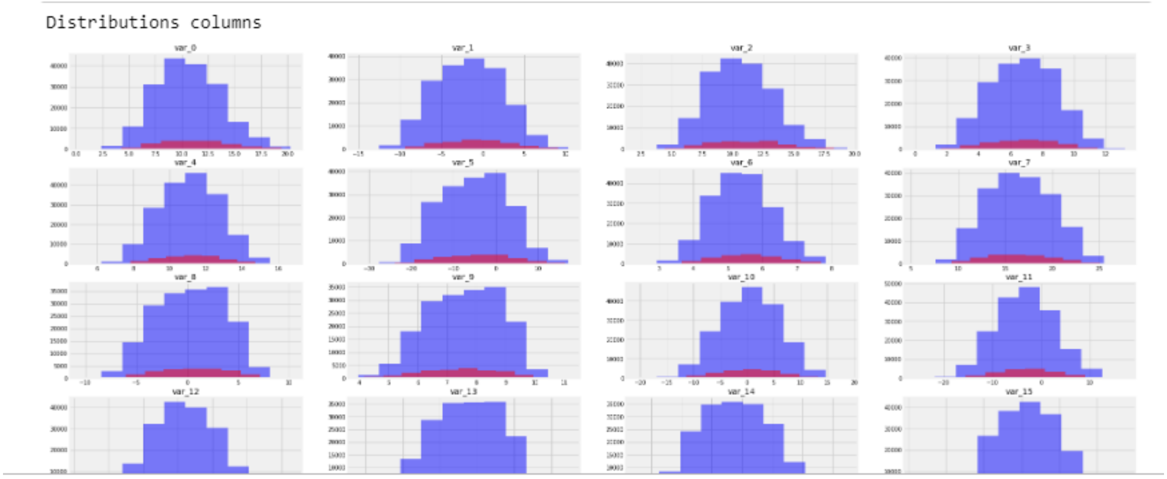
1. Features of each variable with the help of Histogram.







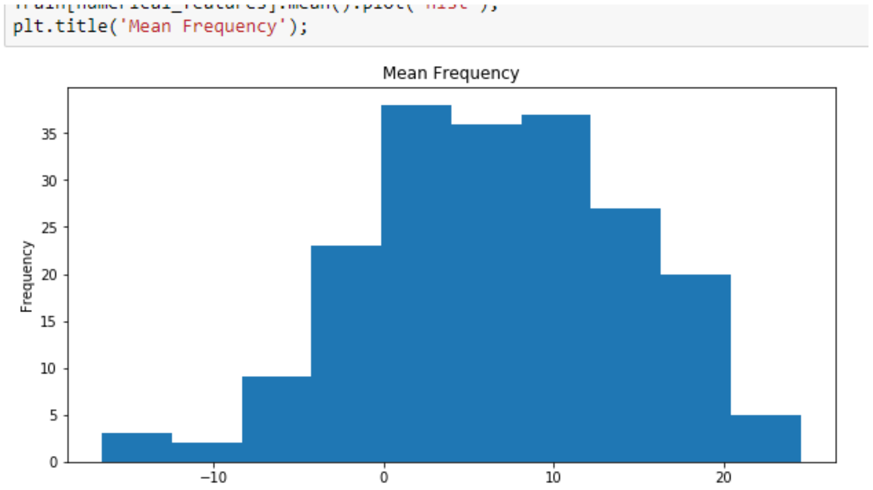
2. Target values with respect to each variable.



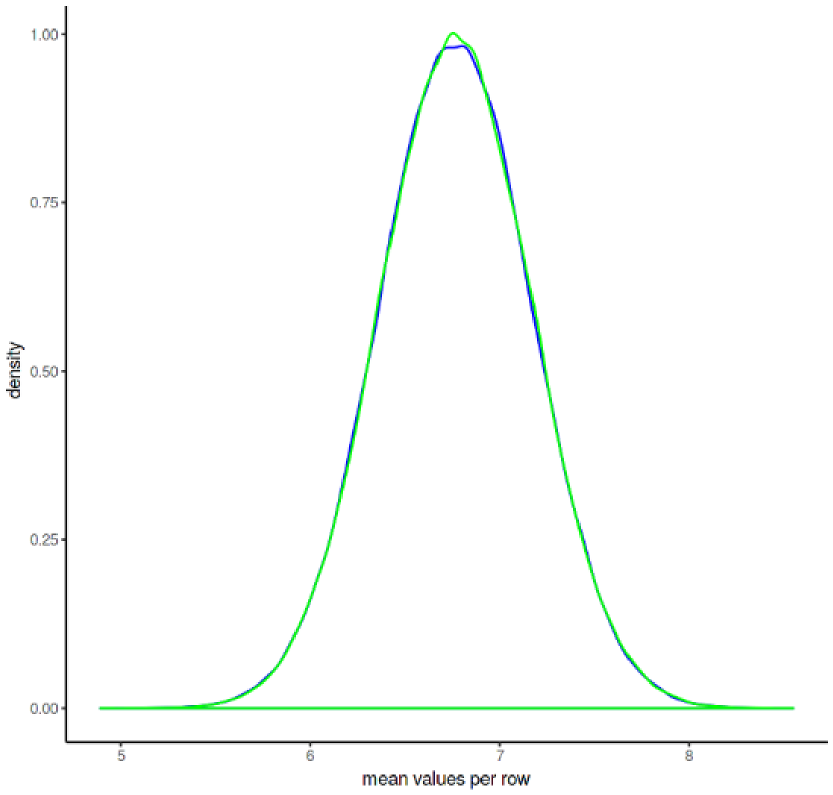
Distributions columns

2.2 Mean Frequency:

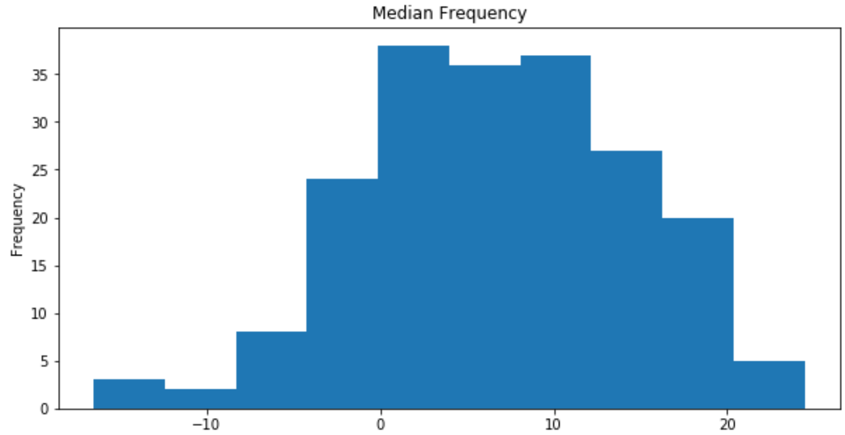
It is a single measure that tries to describe the set of data through a value that represents the central position within that data set. Most popular measures of central tendency used for frequency analysis are Mean, Median and Mode. While the mean is the average value of the data set, the median is the middle observation (observation which has an equal number of values lying above and below it) in the data set.



```
plt.title('Mean Frequency');
```

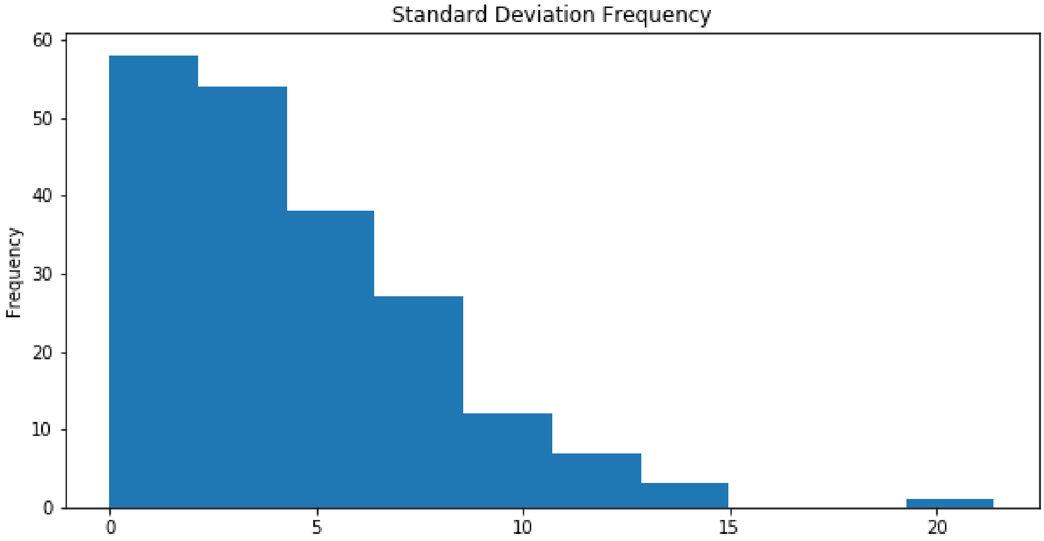


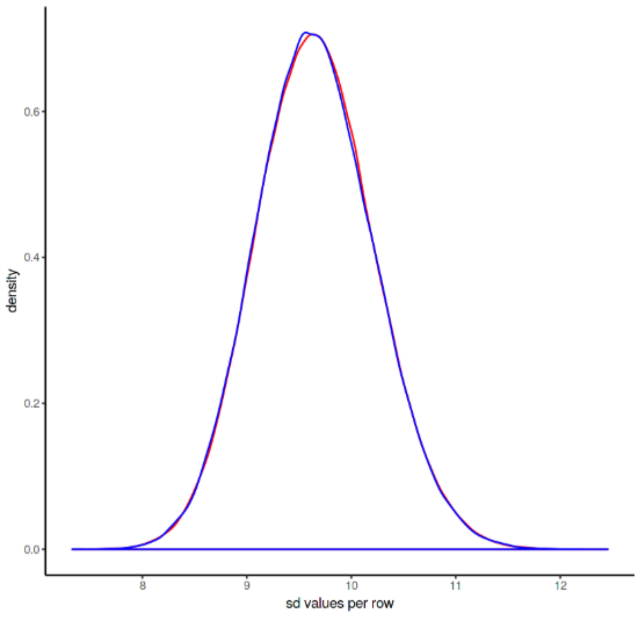
2.3 Median Frequency:



2.4 Standard Deviation:

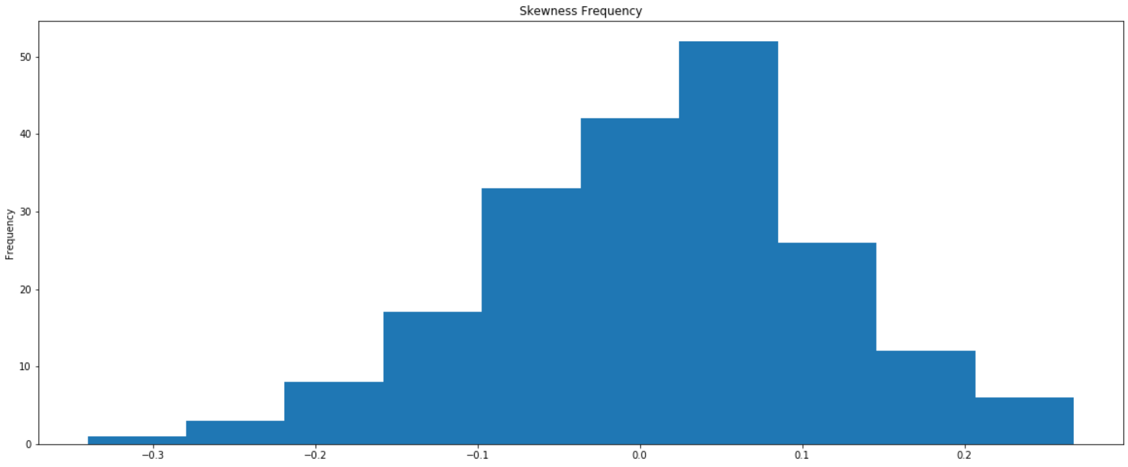
These reflect the spread or variability of data within a data set. Most popular measures of dispersion used for frequency analysis are Standard Deviation, Variance and Range.



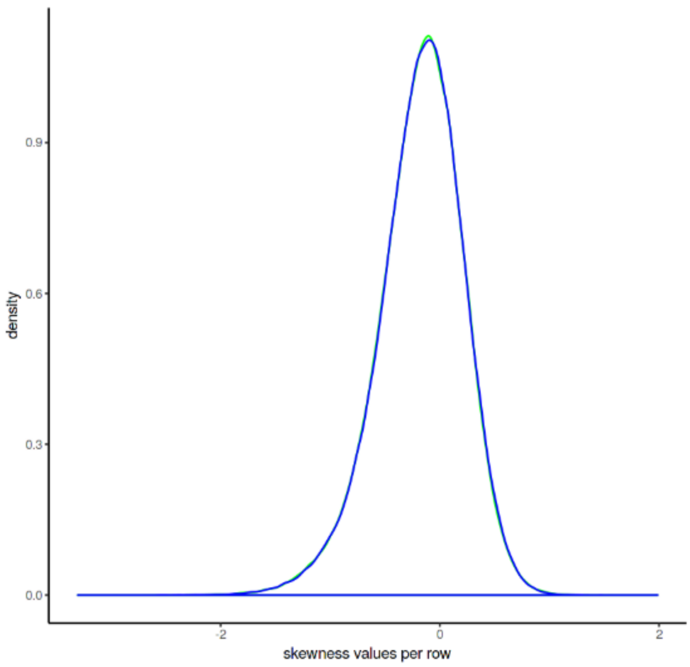


2.5 Skewness Frequency

It is the *degree of distortion* from the symmetrical bell curve or the normal distribution. It measures the lack of symmetry in data distribution.
It differentiates extreme values in one versus the other tail. A symmetrical distribution will have a Skewness of 0

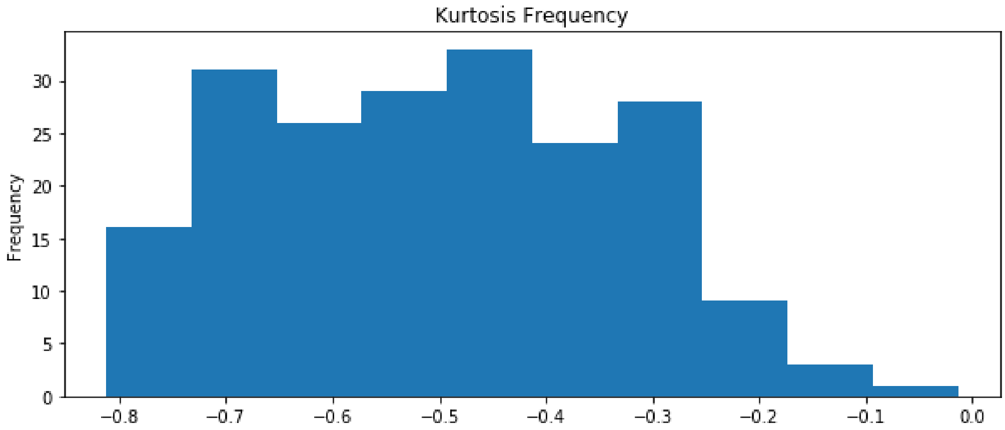


The tail of the left side of the distribution is longer or fatter than the tail on the right side. The mean and median will be less than the mode.

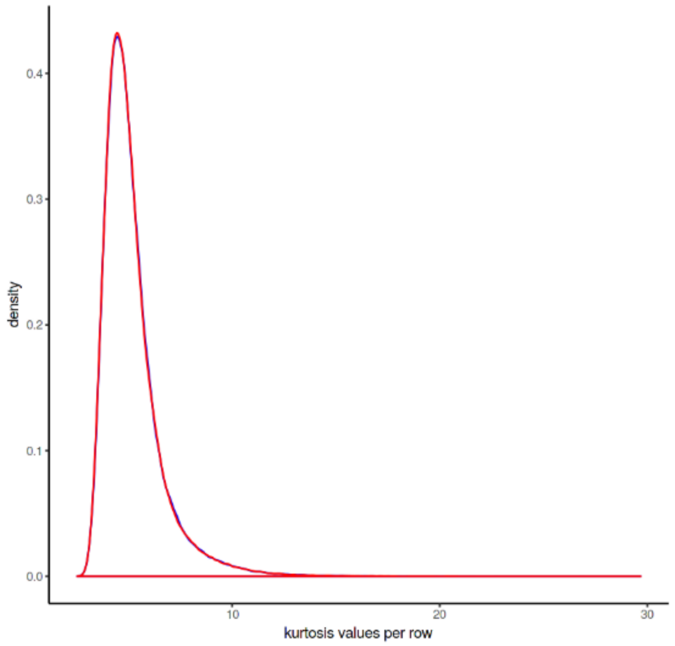


2.6 Kurtosis Frequency:

Kurtosis is all about the tails of the distribution — not the peakedness or flatness. It is used to describe the extreme values in one versus the other tail. It is actually the measure of outliers present in the distribution.



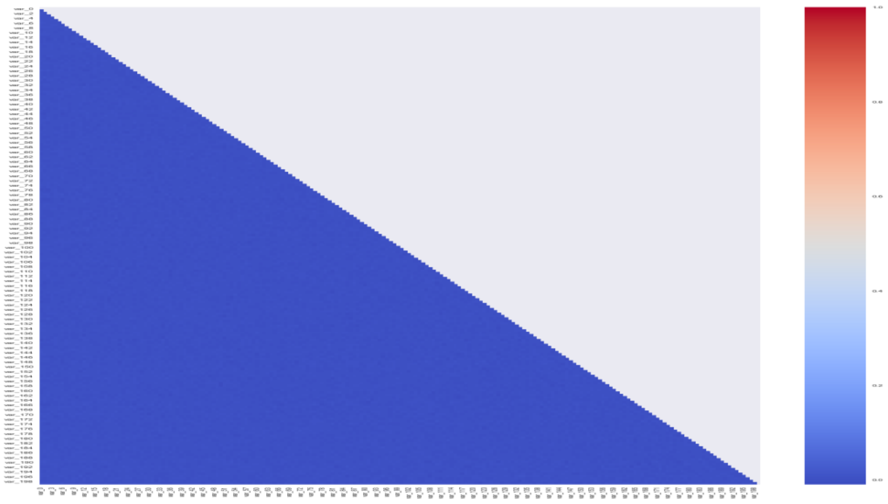
This distribution has kurtosis statistic similar to that of the normal distribution. It means that the extreme values of the distribution are similar to that of a normal distribution characteristic.



2.7 Correlation Plot

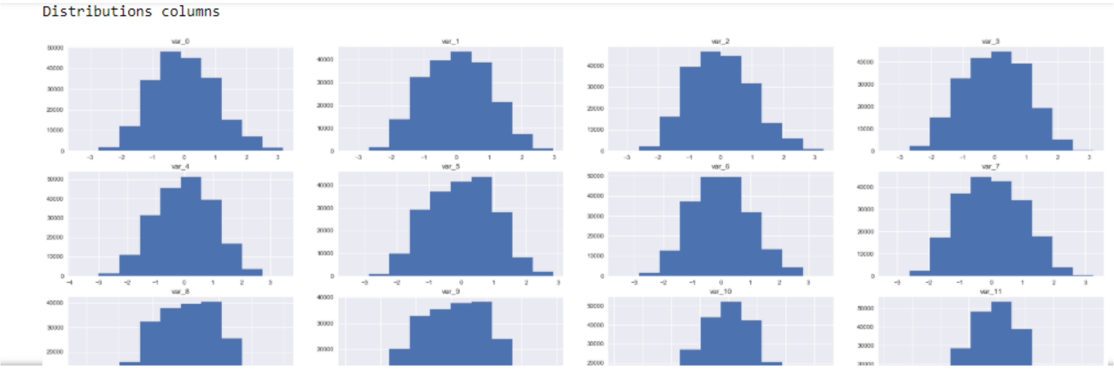
Correlation can be a powerful tool in machine learning. One of a pair of heavily correlated predictors can be removed without harming the predictive power of a model. Also very high absolute correlation with the target can indicate that this is an important variable. We apply the former and we look at the top features in the sense of being correlated with the target. We first apply 10 normalization, where appropriate, to scale all variables between 0 and 1. We note that several of the features are hugely imbalanced and this type of scaling does not damage that. A feature X will become normalized feature Z with the following formula:

zi = xi − min(X) max(X) − min(X) ∀ i = 1, ..., length(X)



2.8 Standardisation

Data standardization is the process of rescaling one or more attributes so that they have a mean value of 0 and a standard deviation of 1. Standardization assumes that your data has a Gaussian (bell curve) distribution



```
Distributions columns
```

Modelling

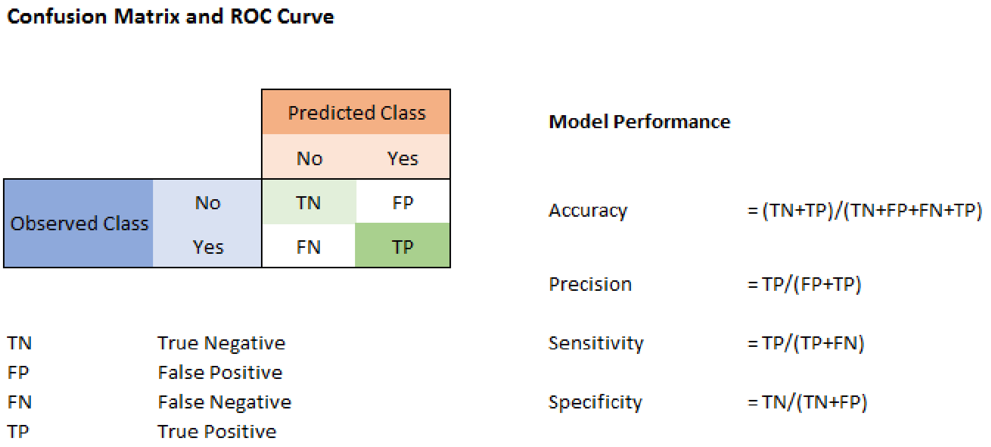
If we consider a binary classifier we have four possible outcomes when we use it make a binary prediction and we call the collection of this a confusion matrix and an example is shown in Figure.

True negative: We predict 0 and the class is actually 0

False negative: We predict 0 and the class is actually 1.

True positive: We predict 1 and the class is actually 1.

False positive: We predict 1 and the class is actually 0.



Confusion Matrix and ROC Curve

		Predicted Class	
		No	Yes
Observed Class	No	TN	FP
	Yes	FN	TP

TN	True Negative
FP	False Positive
FN	False Negative
TP	True Positive

Model Performance

Accuracy	= (TN+TP)/(TN+FP+FN+TP)
Precision	= TP/(FP+TP)
Sensitivity	= TP/(TP+FN)
Specificity	= TN/(TN+FP)

We define the following ratio’s. The True Positive Rate corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points. The higher the better, all else equal. The False Positive Rate corresponds to the proportion of negative data points that are incorrectly classified as positive, with respect to all negative data points. The lower the better all else equal.

True Positive Rate (TPR): TP/ TP + FN

False Positive Rate (FPR): FP/ FP + TN

Binary classifiers usually predict with what probability they expect 1 to occur. The threshold where a high probability leading to predicting a 1 lies is an arbitrary decision.

Solution Methods:

We use 3 different methods to come to a solution: Logistic Regression, Decision Tree and Random Forest from Scikit Learn.

4.1 Logistic Regression:

Logistic regression is a relatively 'simple' machine learning algorithm and we expect fast, but not great results. It tries to attach the best constant values to how features interact with the target, based on the train set, minimizing an error term. It then applies this same formula to the test data set. It is pursued here as a baseline in order to compare to more sophisticated models. For more details see (Bishop, 2006).

4.2 Decision Tree:

This section roughly describes what a decision tree is. The concept is fairly simple. If a prediction needs to be made, go from the top of the tree to the bottom. This tree is constructed by at each depth level greedily finding the best feature to split on, maximizing information gain via the Gini impurity. Let p0 be the proportion of observations that belong to class 0 and let p1 be the proportion of observations that belong to class 1 out of all observations.

$$\mathrm{GINI} = 1 - (P_0^2 + P_1^2)$$

This metric is an example of how to measure how pure a split is. It is more pure the lower it is with minimum 0 and maximum 0.5. A split is purer if in the branches the ratio of positive to negative examples is close to 0 or 1 or in other words the split is highly discriminatory. The maximum depth of this tree is set at 2 to keep it tractable and this effectively also keeps it from over fitting. A decision tree can otherwise perfectly match a training set, which does not generalize well.

4.3 Random Forest

The Random Forest model generates multiple decision trees. (Breiman, 2001) Only a subset of predictive features is now considered during each split that is randomly selected. The decision trees lead to one single prediction together by averaging all the predictions they give individually. The parameters of a Random Forest model are really important and need to be tuned appropriately. N_ESTIMATORS determines the number of trees in the forest. MAX_FEATURES controls the maximum random amount of features to consider when determining a best split during the algorithm. MAX_DEPTH limits the depth of a tree in the random forest. MIN_SAMPLES_SPLIT determines the minimum amount of observations that need to be in a node for it to be considered for splits. MIN_SAMPLES_LEAF constrains that leaf nodes have at least this number of observations. N_JOBS controls the number of processors. Trees in a random forest can be made in parallel, so the more cores working, the less computation time needed. CLASS_WEIGHT can be set to ’balanced’ to deal with imbalanced datasets.

Results

Algorithm	Accuracy	Precision	Recall
Logistic Regression	55.28	14.69	71.40
Decision Tree	83.63	18.68	19.29
Random Forest	90	74.19	0.5

Conclusion

This Project researched how to pre-emptively understand if customers of Santander will be dissatisfied using Machine Learning. An anonymized dataset, to protect the privacy of the customers, provided difficulties in asserting what could be relevant or not, especially in light of a huge feature set. However a thorough data analysis discerned the meaning and interpretation of several features. A Python and R implementation utilized the Logistic Regression, Random Forest and Decision Tree algorithms, carefully tuned, in order to lead to predictions. Further research could for example employ different solution methods, apply more feature engineering or combine several models instead of trying singular models. More specifically they can increase the computation time that goes into tuning and for example make the correlation filtering dependent on the correlation with the target.