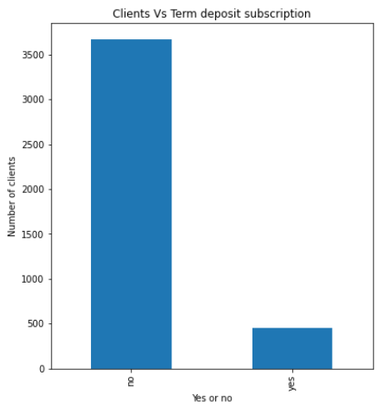
A comparison of Naïve Bayes (NB) and Logistic Regression (LR) on predicting Bank Marketing Campaign (phone calls)

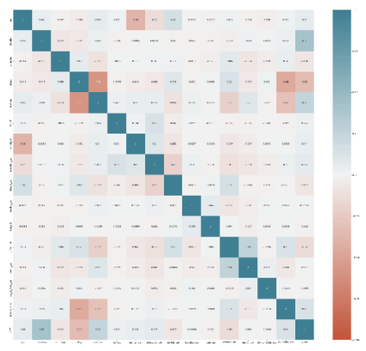
Description and motivation

In this paper, we will be analysing the data of a bank marketing campaign conducted through phone call and predict the subscription of term deposit by their clients. First we perform the analysis with Logistic Regression and analyse the results and performance of the model, then analyse the same dataset with the Naïve Bayes model and compare the results of both the models using performance metrics.

Exploratory Analysis

- Dataset: Bank Marketing from UC Irvine Machine Learning Repository.
- The dataset consists of direct marketing campaigns of a Portuguese banking institution.
- There are two datasets in this repository, one with 41188 instances and one with 4119 instances which is stratified.
- Since running the model with hyperparameter takes longer time, we have chosen the smaller dataset with 4119 instances.
- The cleaned dataset consists of 4119 rows, 21 columns including the target variable.
- As pre-processing step first data has been checked for null values, dropped duplicates and as advised by the bank dropped some unwanted columns which describes the employee details and their monthly indicators.
- Now we have 16 variables out of which 5 are quantitative variables (continuous data), 10 are qualitative variables (categorical data) and 1 target class variable.
- The target variable consist of 'yes' or 'no' which briefs whether the client has subscribed the term deposit or not, respectively.
- The categorical variables are encoded using the label-encoding feature and plotted correlation matrix for the complete dataset, the top three correlated variables are duration (0.42) , pdays (-0.33) and previous (0.26)
- By analysing the target attribute, from fig: 2 it is clear that that there is a class imbalance that 89% of the data is 'no' and only 11% of the data is 'yes'.





Correlation Matrix

Logistic Regression

- Logistic Regression is a Supervised learning method for classification problems where the output or the target variable should be a discrete.
- There are three types of logistic regression, Binary logistic regression, Multinomial logistic regression, Ordinal logistic regression.
- This is a linear model, but the predicted probabilities are transformed using the logistic function to ensure that they fall between 0 and 1.
- It takes a linear combination of features and applies to them a nonlinear sigmoidal function, the logistic function is defined as logistic(η) = 1 / 1+exp(−η) where a threshold value on the probability is set to classify the target variables.
- Maximum Likelihood Estimation, gradient descent can be used for parameters.
- You can regularize the model by using Lasso, Ridge or Elastic Net regularization to reduce complexity of problems and penalize for having too many features.
- The deviance is the measure of the error between the logistic model fit and the outcome data.

Advantages:

- It is a simple and efficient method for predicting a binary outcome.
- It can handle imbalanced classes and missing data.
- It can be extended from binary classification to multi-class classification.
- It can be regularized to prevent overfitting and improve generalization.
- It is a widely used and well-understood method, with a long history in statistical analysis.
- No hyperparameter optimization.

Disadvantages

- It can be sensitive to small changes in the data, which can lead to unstable models.
- The interpretation is more difficult because the interpretation of the weights is multiplicative and not additive.
- It is prone to overfitting when there are many predictor variables and a small sample size.
- It can have difficulty handling highly correlated predictor variables.
- Missing values is a problem.
- Need to give only numerical values as inputs. So, categorical variables has to be encoded and given as input.

Naïve Bayes

- Naive Bayes is a classification algorithm based on the principle of Bayes' Theorem, which states that the probability of an event occurring is equal to the prior probability of the event occurring multiplied by the likelihood of the event occurring given certain evidence
- As the name 'naïve' says - naive assumption, it allows the algorithm to make predictions based on the individual probabilities of each feature, rather than considering the interactions between features
- There are several different variations of the Naive Bayes algorithm, including the Gaussian Naive Bayes, Multinomial Naive Bayes, and Bernoulli Naive Bayes. These variations are used for different types of data, such as continuous data, count data, and binary data, respectively
- Usually, Naïve Bayes models perform surprisingly well despite the assumption of independence.
- Assumes prior probabilities from class distribution in the training set, multinomial distribution for discrete features and gaussian distribution for continuous features but these can be adjusted to find the optimal model.

Advantages:

- It is a simple and efficient algorithm.
- Naive Bayes is suitable for solving multi-class prediction problems.
- Naive Bayes is better suited for categorical input variables than numerical variables.
- It is not sensitive to irrelevant features and it can handle missing data.
- It is not sensitive to the scale of the data and deals well with high dimensional data.
- f its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.

Disadvantages

- It makes the naive assumption that features are independent, which may not be the case in real-world data.
- It can perform poorly on data sets with many features or highly correlated features.
- It can be sensitive to the prior probabilities of the classes.
- This algorithm faces the 'zero-frequency problem' where it assigns zero probability to a categorical variable whose category in the test data set wasn't available in the training dataset
- Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. This limits the applicability of this algorithm in real-world use cases.

Hypothesis Statement:

- A base research on the reference journals it is understood that Logistic regression will have a better accuracy than the Naïve bayes due to its assumption of independence of features.
- Olatunji Apampa from Tilburg University has done analysis on the same dataset in 2016, in which it was evident that the data balance of the class attribute has a significant impact in Naïve bayes but not in Logistic Regression. For unbalanced data precision values are 0.65 and 0.54 for LR and NB, whereas for balanced dataset both the values are around 0.75
- We expect that the losses or errors in predictions are comparatively lesser in Logistic Regression than the Naïve bayes algorithms.

Methodology:

- Split data into a 80: 20 split for train and test data. The test data remains unseen to models until end.

In Logistic Regression:

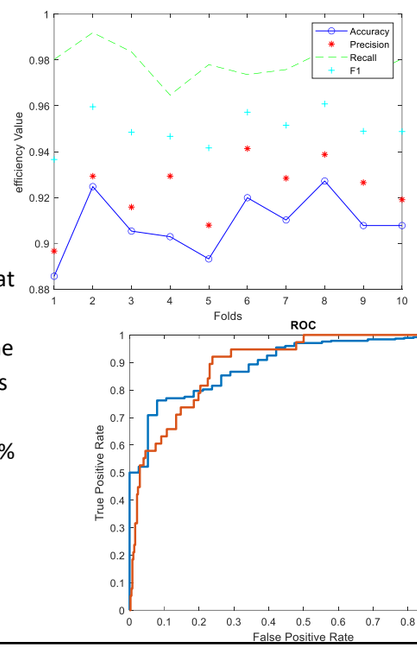
- Used 10 - fold cross validation on the 'train' data to build the model and compare the performance metrics of the 10 fold cross validation models.
- Then test the model with the 'test' data and obtain accuracy, error, recall, and other performance metrics. This will provide information on the goodness of fit of models.

In Naïve Bayes:

- Build the model on the same 'train' dataset used for LR.
- Then test the model with the 'test' data and obtain performance metrics from this model
- Optimize the NB model by using hyperparameter optimization and fetch the performance metrics from this model.
- Evaluate all the results in order to get the optimal model for this dataset.

Experimental results – Logistic Regression:

- Accuracy, Prediction, Recall and F1 Scores for the 10 KFold cross validation is plotted here in the 1st graph, this shows how well the model has performed in each and every validation.
- In this clearly evident that the model has been trained with an average accuracy of 90%



- A receiver operating characteristic (ROC) curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various classification thresholds
- The blue curve represents the performance of the 'no' class with 98% accuracy and the red curve represents the performance of the 'yes' class with 65% accuracy.

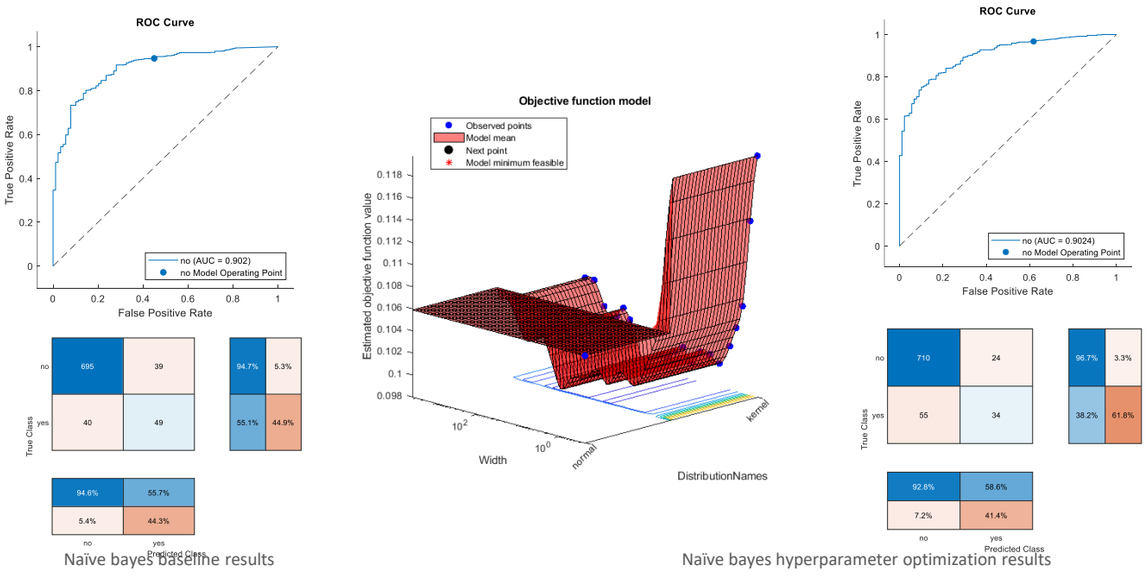
Analysis and Evaluation of results:

- In our 1st model – Logistic Regression the accuracy of the model in predicting the class attribute with the test data is 91%, whereas in the naïve bayes baseline model also it is the same. Whereas in the naïve bayes hyperparameter optimization model after 30 iterations in finding out the best hyperparameter, it has ended up with kernel distribution and even then the resulted accuracy is 92% which is a marginal increase from the other two models.

- From the confusion matrix of all the models it is clearly evident that the accuracy of the True positive is more than 90% (in our dataset it is 'no' class) and the True Negative accuracy always lesser, in the range of 40% to 50% (in our dataset it is 'yes' class). This means that the model is good in predicting one class (i.e. 'no') and the model is not performing good for the other class (i.e. 'yes'). The reason behind this is the data imbalance on the class attribute. As mentioned in the top of the poster 89% of the data is of class 'no' and only 11% of the data is of class 'yes'.
- In this imbalance dataset, the overall accuracy is almost 90% for all three models, but if you see the TP and FN accuracy, all three models have a great TP accuracy of almost 95% but the difference comes in the FN accuracy (accuracy of class with less input data) where Naïve bayes baseline model has 45% and the LR model has 55% and the Naïve bayes with hyperparameter optimization has 65%.
- From this we can infer that Naïve bayes hyperparameter optimization predicts well with imbalanced data when compared to Naïve bayes baseline model and Logistic regression
- On the other hand in order to find the best distribution hyperparameter optimization takes lots of time when compared to the other two models.

Experimental results – Naïve Bayes & Hyperparameter optimization:

- The two charts in the left represents the performance of the Naïve Bayes baseline model with overall 91% accuracy with True positive accuracy of 94% for 'no' class and True Negative accuracy of 45% for 'yes' class.
- The output function model of the hyperparameter optimization is represented in the center.
- The two charts on the right represents the ROC curve and confusion matrix of NB model with hyperparameter optimization. Where, the overall accuracy of the model is 92% and the True positive accuracy of 97% for 'no' class and True Negative accuracy of 62% for 'yes' class.



Naïve bayes baseline results

Naïve bayes hyperparameter optimization results

Lessons Learned:

- Target class imbalance may not have an impact on the overall accuracy of different models but will have a bad FN/TP accuracy and that also varies from model to model (Naïve bayes hyperparameter optimization is good in this case)
- Learnt to model LR algorithm , NB algorithm and optimizing using hyperparameter and evaluating the performance metrics of all the models.

Future Work:

- Other methods to correct target imbalance e.g., SMOTE (Synthetic Minority Oversampling Technique)
- Explore effects of using variants of NB classifier.

Comparison of results from all three models:

- Here is the performance metrics of all three models,

Models / Parameters	Logistic Regression	Naïve Bayes	Naïve Bayes With hyperparameter optimization
Accuracy	0.91	0.91	0.92
Precision	0.92	0.94	0.93
Recall	0.97	0.96	0.96
F1 Score	0.95	0.95	0.96
Loss/error	-	0.1	0.09

References:

[1] Olatunji Apampa Tilburg University T, "Evaluation of Classification and Ensemble Algorithms for Bank Customer Marketing Response Prediction ", 2016.

2 [4] Moereira.J, Carvalho.A and Horvath.T, *"A general introduction to Data Analytics "*, 2019, Chapter 9, page 207.

[5] Tuffery.S, *"Data Mining and Statistics for Decision Making "*, First Edition, 2011, Chapter 11, page 478.

[6] Andrew Y. Ng , Michael I. Jordan, *"On Discriminative vs. Generative classifiers: A comparison of logistic regression and Naive Bayes "*, 2001, https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf (accessed 6th December 2020).