Charlie George Barrow
Charlie.Barrow@city.ac.uk

Multilayer Perception versus Support Vector Machines in data classification.

Abstract:

This paper aims to present a critical evaluation of two neural network models; Feedforward Multilayer Perceptron (MLP) and Support Vector Machines (SVM) and their performances, whereby predicting whether clients of a Portuguese banking firm subscribed to a term deposit account based on their Age, Balance, Day, Duration, Campaign, pdays, and Previous inclination in a binary classification problem. Furthermore, the results from the finest evaluated models were compared by Confusion Matrix, Receiver Operation Curves (ROC) and classification reports. For this classification problem, the SVM proved superior.

1. Introduction:

In the UK, the banking industry and its historical oligopoly has been significantly disrupted, this is due to new market entrants in the form of challenger banks and Fintech orientated businesses [1]. Whom, are competing candidly with longer-established banks in the country, by concentrating in areas often overlooked or otherwise discounted by historic banks. Achieved through utilising advances in technology, marketing transparent business models, and focusing primarily on customer service and experience. Leading new entrants to penetrate this sector and grab a respectable market share [1].

The purpose of this report is to critically evaluate two Artificial Neural Network (ANN) models; MLP and SVM. Constructed, to solve a binary classification problem that predicts the success of a Portuguese Bank's Telemarketing campaign, to determine whether its bank's clients subscribed to a term deposit. Whereby, attempting to distinguish a pattern by applying the stated ANN models to eight specific attributes in the Portuguese Banks marketing dataset [2]. Additionally, this research paper aspires to surpass the results of a previous study [3] that undertook the same binary classification problem using the identical dataset and applying machine learning models K-Nearest Neighbour (K-NN) and Naïve Bayes (NB) to the dataset, which results obtained were inadequate. Lastly, with the anticipation of obtaining satisfactory results, such findings may prove advantageous to established banks, challenger banks, and Fintech organisations who wish to undertake a marketing campaign of their own to attract more customers to subscribe to a product.

1.1. Multilayer Perceptron (MLP):

An MLP is a class of feedforward ANN and supervised backpropagation algorithm. MLP differs from a single-layer perceptron that does not contain a hidden layer, whereas MLP has at least one layer but typically, has multiple where each layer is a non-linear activation function [4]. Furthermore, multilayer ANN construction and non-linear activation, permits MLP's to generate non-linear decision boundaries, making MLP's a desirable model choice in solving binary classification problems from multi-dimensional real-world datasets. Moreover, the MLP architecture can be broken down into three separate parts, the input layer, hidden layers, and output layer. The input layer includes each node or column in a dataset. The hidden layer then applies the non-linear activation function to the input layer. Lastly, the output layer displays the outcome of the model, which is represented by two nodes.

Firstly, the advantages of an MLP ANN are that it is straightforward to setup and train alongside to adapt without the assistance of the user [4]. Secondly, MLP is a non-linear device which is crucial, specifically, if the connection between input and output is inherently non-linear [4]. In addition, it can be linked to statistical models, including Gaussian density and sigmoid. Lastly, MLP's are vigorous in their performance, lowering steadily in the presence of increasing amounts of data noise and maps the input and outputs efficiently creating a stable network [5]. The disadvantages of an MLP are that, it typically cannot handle insufficient data training data [6]. Also, the number of total parameters can develop too high. Consequently, becoming inefficient because there is redundancy in such high dimensions [7]. Finally, MLP attempts to locate a local minimum in the error function output, and, if it ends up locating the wrong one, the findings can be considerably poor.

1.1. Support Vector Machines (SVM):

SVM is a classification supervised machine-learning algorithm, which draws the line between two or more classes in the most efficient way. Which these lines are drawn to predict future data [8]. Achieved through maximum separability, which assesses the widest margins in the line, since the largest width can separate the two classes and the closest point to each group points [8]. Additionally, the centre of the two margin lines is called the hyperplane [8]. Within SVM, you have support vectors, which are the points that lie on the two margins. The advantage of the SVM as a classifier is it operates robustly (similar to MLP), whereby there is a clear margin of separation between classes and when in higher dimensional spaces, the number of dimensions is greater than the number of samples [9]. Finally, SVM is moderately memory efficient and is a unique solution, since the optimality problem is convex. Which is an improvement compared to MLP, which has multiple solutions associated with local minima and for this reason may not be robust over different samples [10].

2. Dataset:

The dataset used is based on Portuguese Banks Marketing Campaign Results, acquired from the UCI Machine Learning Repository [4]. The original dataset contains 45211 instances together with 17 attributes- 7 numeric, 10 categorical with one output variable ‘y’ that states either ‘Yes’ or ‘No’. Nine of the seventeen will be dropped leaving only; Age, Balance, Day, Duration, Campaign, pdays, and Previous columns. Additionally, it was necessary to replace the classifiers factor column (‘y’) with integers ‘yes’ (class 1) = 1 and ‘no’ (class 2) = 2. Alongside, normalising the values of the columns in the dataset, to alter the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values. Lastly, due to the dataset being heavily imbalanced (Appendix A) its classes are not approximately equally represented, therefore the model’s performance metrics such as Receiver Operating Characteristics (ROC), Confusion Matrix are likely to be impacted. To combat an imbalanced dataset the SMOTE (Synthetic Minority Oversampling Technique) technique was applied to increase the number of minority classes in the dataset and balance the data (Appendix A) with the anticipation of significantly improving the ROC curves and Confusion matrix scores.

2.1. Initial Data Analysis:

The initial data analysis process includes comparison histograms (Figure 1) to assess the difference in distributions between the classes, Age, Balance, Day, Duration, Campaign, pdays, Previous, and y. Through which all, apart from pdays and y appeared to be right-skewed indicating that the overall datasets mean is greater than the median [12]. That could be attributed to the data having limited large values that drive the mean upward but do not affect where the exact middle of the data. Moreover, as for the left-skewed data seen in variables pdays is attributed to the data’s mean is less than the median opposite to right-skewed since the data has fewer minor values that drive the mean downward but do not affect where the exact middle of the data is [12]. Additionally, a correlation matrix was created (Figure 2) highlights the correlation coefficients between variables in the dataset. The matrix displayed the variables as uncorrelated, which may prove more difficult for the models, since there are clear disparities between the variables. Nevertheless, the results to be produced will be noteworthy to see how the models coped with the uncorrelated data.

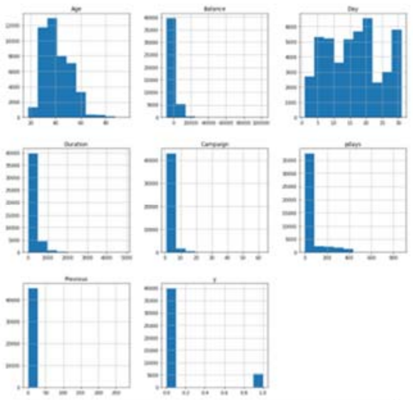


Figure 1: Dataset Comparison Histograms

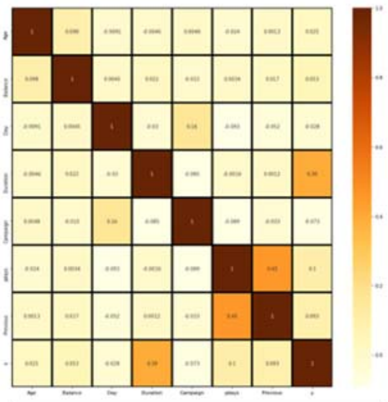


Figure 2: Correlation Matrix

3. Hypothesis:

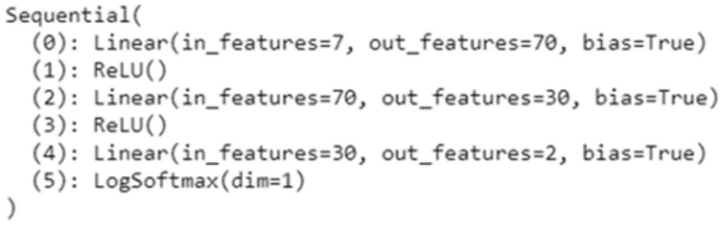
This paper's MLP and SVM models are expected to outperform the previous studies [4] NB and KNN model's predictive performance when predicting whether clients of a Portuguese banking firm subscribed to a term deposit account. This assumption derives from the evolvement in the current studies increase in time complexity, decision boundaries, and increase in attributes being trained in the MLP and SVM models. Alongside, working with a balanced dataset through applying the SMOTE function to its x and y variables. Additionally, regarding the comparative aspect of the MLP and SVM models, it is probable that the SVM could outperform the MLP model classification accuracy. Reasoning can be seen in a similar study [13] which saw SVM produce better results using the kernel function Gaussian Radial Basis Polynomials in a high dimension dataset. Given this study also uses a high-dimensional dataset, results could be comparable.

4. Methodology and evaluation:

Each models' methodology entails holding out 30% of the initial dataset as testing data, for the algorithm assessment process, between the soundest MLP and SVM models. The rest of the data (70%) will be used for training and validating in the process of model selection, alongside training in the process of comparing both algorithms. Furthermore, each model assortment process involves adjusting the hyperparameters of both MLP and SVM models. As such, each model training and testing will be split 70/30 not before applying the SMOTE technique mentioned previously to the x and y label data to balance the dataset. Followed by, normalising the x data before model execution. Doing this approach according to [14] should produce more accurate and reliable results, in comparison to that of [4]. Additionally, For the algorithm comparison, each model will be evaluated through using performance metrics; ROC curves to visualise the TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis. Plus, calculating the area under the curve (AUC) to understand how well each model is capable of identifying classes. Since, the higher the AUC, the better the model is at predicting 0's as 0's and 1's as 1's [15]. Then Finally, a classification report to visualise each model's precision, recall, f1-score, and accuracy of each model. Together with, a confusion matrix for each model to further understand the classification report.

4.1. Architecture and Parameters used for the MLP:

The architecture of the MLP model will entail an ANN with two hidden layers and one output layer. That includes seven predictors; Age, Balance, Day, Duration, Campaign, pdays, and Previous. Alongside the seven predictors, the model utilises Pytorch's sequential function to include the seven predictors with 70 neurons in the first hidden layer. In-between the first layer will be the activation function "ReLu", through which after the second hidden layer goes from 70 neurons to 30 neurons, which again is parsed through the activation, function. Lastly, the data is fed into the output layer of the MLP after passing through the 30 neurons, separating the output data into the two classes yes' (class 1) = 1 and 'no' (class 2) = 2. Moreover, the model will additionally apply a Log NLLLoss and a negative log-likelihood loss to train the stated classification problem since this method is useful with several classes. Together with, a stochastic gradient descent (SGD) optimization method to update the weights that will encompass a learning rate of 0.01% for the MLP model. (Figure 3) Finally, after creating the prediction function to predict the classes using the MLP model, when executing it will have exactly 5000 epochs with a batch size of 30 and batches of 18 to maximise the training accuracy of the model.



```
Sequential(
  (0): Linear(in_features=7, out_features=70, bias=True)
  (1): ReLU()
  (2): Linear(in_features=70, out_features=30, bias=True)
  (3): ReLU()
  (4): Linear(in_features=30, out_features=2, bias=True)
  (5): LogSoftmax(dim=1)
)
```

Figure 3: MLP Architecture

4.2. Architecture and Parameters used for the SVM:

Whilst SVMs do not have training parameters like the MLP do. The SVM model will comparatively have a maximum of 5000 epoch using the same data as the MLP model with the SMOTE adjusted x and y data. Additionally, the SVM model originally applied all types of kernels and selected the top-performing kernel out of the kernels; Polynomial, Gaussian, Hyperbolic and Sigmoid kernel. Which stood to be the Poly kernel with a hyper-parameter set at 8 degrees since higher degree polynomial kernels allow a more flexible decision boundary.

5. Analysis and Critical Evaluation of Results:

Firstly, Figures 4 and 5 highlight the results of each model's testing stage, comparing the greatest MLP and SVM results through a confusion matrix, alongside a classification report. Looking at the confusion matrixes for the MLP and SVM models, SVM misclassified approximately 6% less than the MLP. Overall, both model performances performed quite similarly, misclassifying almost the same number of samples. Additionally, the classification reports statistics for each highlighted SVM to have marginally higher F1-score by 1%, recall by 2%, and precision stats by 3%.

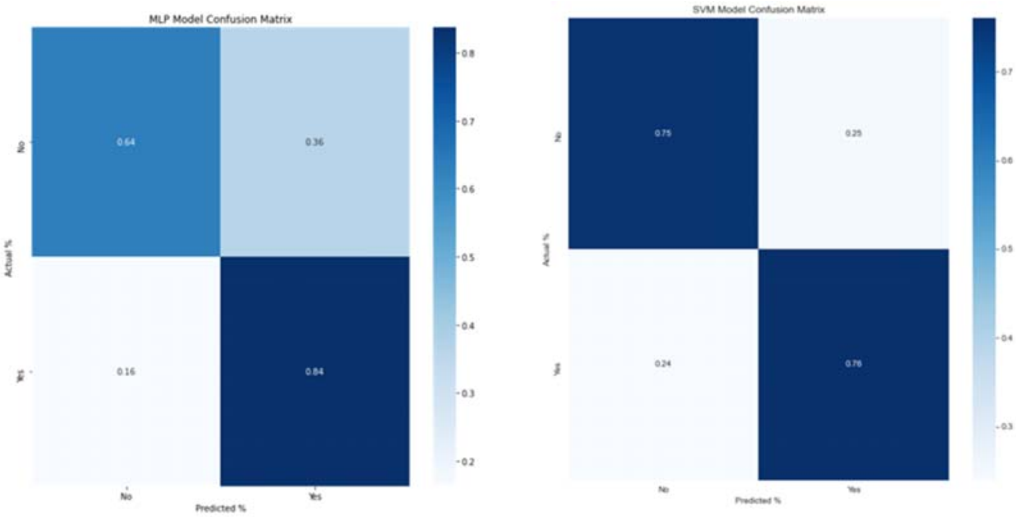
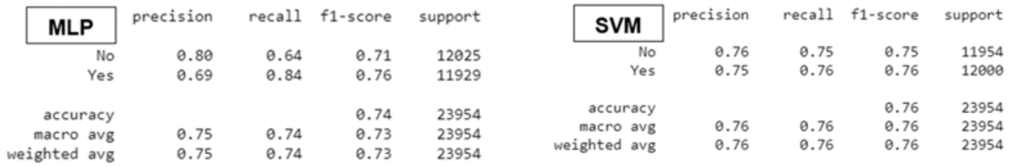


Figure 4: MLP & SVM Confusion Matrix



MLP

```
              precision    recall  f1-score   support

          No       0.80      0.64      0.71     12025
         Yes       0.69      0.84      0.76     11929

    accuracy                           0.74     23954
   macro avg       0.75      0.74      0.73     23954
weighted avg       0.75      0.74      0.73     23954
```

SVM

```
              precision    recall  f1-score   support

          No       0.76      0.75      0.75     11954
         Yes       0.75      0.76      0.76     12000

    accuracy                           0.76     23954
   macro avg       0.76      0.76      0.76     23954
weighted avg       0.76      0.76      0.76     23954
```

Figure 5: MLP & SVM Classification Reports

In the model selection process, the best MLP model could achieve 74% accuracy, whereas the best SVM model 76%, placing the SVM model superior. Furthermore, the algorithm comparison process, where they had 10% and 20% fewer data to train than in the previous process (30%) saw marginal differences in accuracy with the SVM model achieving 72% and 69% and the MLP model achieving 69% and 65%. Overall, the SVM model's dominance could be attributed to the fact its kernel being able to cope better with the higher dimensional-dataset since [13] previously described this who additionally discovered SVM models to surpass MLP models. Conversely, it is worth mentioning that the MLP only used two hidden layers, which are considered a basic MLP model, as the layers could have been increased, and the accuracy subsequently. Additionally, the computational power requirement of the SVM model can't be ignored, whilst the MLP model took no longer than five minutes to run, the SVM model took often upwards of 30 minutes to run which, for the sake of time difference, only increased the accuracy by a couple of percent.

Moreover, ROC curves (Figure 6) and the AUC display the quality of the classifiers and visually deepens the analysis. The results once more show the SVM (76%) model to surpass that of the MLP (74%). Whilst only marginal, practically in all statistics, the SVM model shows dominance when predicting true positives (Figure 5) since from the perspective of the Portuguese bank or other institutions looking to run a marketing campaign, would prefer to know the predicted successful outcome of gaining more customers to subscribe to a product through a marketing campaign. Overall, the ROC and AUC values for both models highlight fair results and, more importantly, demonstrate a significant upgrade in results produced in the comparative study [4]. Whereby Appling the same dataset using NB and K-NN ML models produced AUC results of 0.6019% and 0.5706% in comparison to this paper's ANN's 76% and 74%. Underlining an improvement in accuracy of over 15% in model performance, showing ANN models to be the preferable choice.

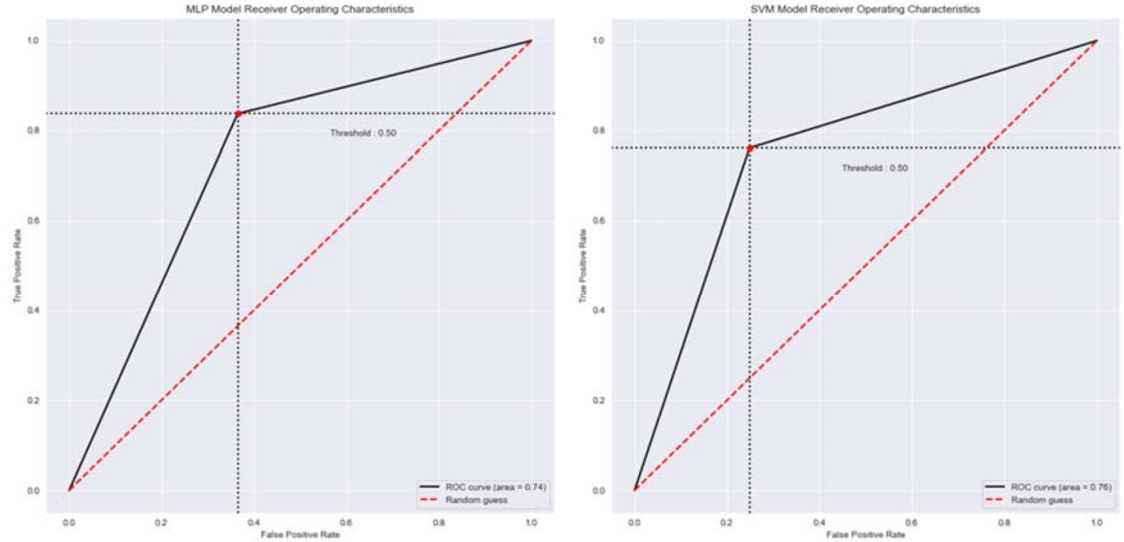


Figure 6: MLP & SVM ROC Curves and AUC Figures

6. Conclusions and Future Work:

This paper analysed how accurately two trained models, an MLP and SVM, can predict whether unseen data [4] can solve a binary classification problem, whereby determining whether or not a bank's client subscribed to a term deposit. In summary, both models performed similarly however, the SVM edged ahead and slightly outperforming MLP with respect to its accuracy, F1-score, recall, and precision statistics visualised in the classification report, confusion matrix, and ROC curves with its AUC. Although, the timing factor for SVM proved troublesome and the MLP could likely outperform its performance in this study if developed further. Additionally, this paper also surpassed the results of the previous machine-learning study that applied the same dataset to the NB and K-NN models.

This paper has demonstrated the power of each model and whilst the SVM proved slightly better, the MLP can’t be discounted for its simplistic but yet powerful results and quick ANN. Furthermore, classification reports, confusion matrix, and ROC curves stood to be great tools when evaluating classifiers, whereby measuring their accuracies separately from the dimensions of classes in the papers dataset and providing a visual perception of the sensitivity and sensibility of the classifier ANN model. For future work, when constructing an MLP model it would be advantageous to add additional hidden layers and increasing batch sizes when formulating, since this would likely increase the model’s prediction capability. Additionally, the way this model was created, the MLP ANN could be modified where instead of using just torch.NN package could use the torch.utils.data to efficiently load the data into the model using the data loaders from this package. Therefore, this should increase the model’s performance. Lastly for SVM, this model should be replaced, and as an alternative, use Auto Encoder or Boltzmann Machines, since they would be able to support the large dataset at such better speeds for binary classification problems, which was the issue with SVM as it is notoriously known for not managing large datasets [16].

7. References:

[1] Deloitte (2021) The DNA of Digital Challenger Banks. Available at: us-dna-of-digital-challenger-banks.pdf (deloitte.com). Accessed at (1st March 2021).

[2] Moro et al., (2014) Bank Marketing Data. Available at: UCI Machine Learning Repository: Bank Marketing Data Set (Accessed 20 November 2020)

[3] Barrow., (2020) A Comparison of K-Nearest Neighbour and Naïve Bayes Applied to a Portuguese Bank Marketing Dataset. Available at: https://moodle.city.ac.uk/mod/assign/view.php?id=1680593. Accessed at (1st March 2021).

[4] Loy, J (2019) Neural Network Projects with Python: Available at: Hardback Accessed at (1st March 2021) – pages 1-23.

[5] Medium (2018) Multilayer Perceptron (MLP) vs Convolutional Neural Network in Deep Learning. Available at: Multilayer Perceptron (MLP) vs Convolutional Neural Network in Deep Learning | by Uniqtech | Data Science Bootcamp | Medium. Accessed at (1st March 2021).

[6] MIT (2006) Why MultiLayer Perceptron/Neural Network? Available at: Why MultiLayer Perceptron (mit.edu) Accessed at (1st March 2021).

[7] Gounder et al., (2021) Hybrid multilayer perceptron-firefly optimizer algorithm for modelling photosynthetic active solar radiation for biofuel energy exploration. Available at: Hybrid multilayer perceptron-firefly optimizer algorithm for modelling photosynthetic active solar radiation for biofuel energy exploration - ScienceDirect Accessed at (1st March 2021)

[8] Stoean et al., (2014) Support Vector Machines and Evolutionary Algorithms for Classification Available at: Support Vector Machines and Evolutionary Algorithms for Classification | SpringerLink (city.ac.uk). Accessed at (1st March 2021) -pages 75-100.

[9] Auria et al., (2008) Support Vector Machines (SVM) as a technique for solvency analysis. Available at: Advantages and Disadvantages of Support Vector Machines (SVMs) Compared to Other Techniques in Bundesbank's Sol-vency Analysis (core.ac.uk) Accessed at (1st March 2021)

[10] Dhiraj K (2019) Top 4 advantages and disadvantages of Support Vector Machine or SVM. Available at: Top 4 advantages and disadvantages of Support Vector Machine or SVM | by Dhiraj K | Medium Accessed at (1st March 2021).

[11] Chawla et al., (2002) SMOTE: Synthetic Minority Over-sampling Technique Available at: chawla2002.dvi (city.ac.uk) Accessed at (1st March 2021).

[12] Rumsey (2016) Statistics For Dummies, 2nd Edition/ Available at: How the Shape of a Histogram Reflects the Statistical Mean and Median - dummies Accessed at (1st March 2021).

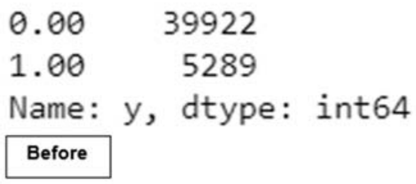
[13] E.A.Zanaty (2012) Support Vector Machines (SVMs) versus Multilayer Perception (MLP) in data classification Available at: Support Vector Machines (SVMs) versus Multilayer Perception (MLP) in data classification - ScienceDirect. Accessed at (1st March 2021).

[14] Svozil (1997) Introduction to multi-layer feed-forward neural networks. Available at: Introduction to multi-layer feed-forward neural networks - ScienceDirect. Accessed at (1st March 2021).

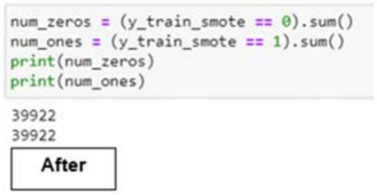
[15] Hosmer & Lemeshow (2013). Applied logistic regression. p.177: Available at: Hardback. Accessed at (1st March 2021).

[16] Cervantes et al., (2007) Support vector machine classification for large data sets via minimum enclosing ball clustering. Available at: Support vector machine classification for large data sets via minimum enclosing ball clustering - ScienceDirect Accessed at (1st March 2021).

Appendix A- SMOTE Technique applied to Output Variable (y)



```
0.00    39922
1.00     5289
Name: y, dtype: int64
```



```
num_zeros = (y_train_smote == 0).sum()
num_ones = (y_train_smote == 1).sum()
print(num_zeros)
print(num_ones)
```

```
39922
39922
```

Appendix B - Glossary:

- **Artificial Neural Network:**
  A piece of a computing system designed to simulate the way the human brain analyses and processes information.

- **Area Under Curve:**
  The area under a curve between two points is found out by doing a definite integral between the two points.

- **Binary Classification:**
  Binary classification is the task of classifying the elements of a set into two groups on the basis of a classification rule.

- **Confusion Matrix:**
  A performance measurement for machine learning classification problem where output can be two or more classes.

- **F1-score:**
  is a weighted harmonic mean of Recall & Precision scores

- **K-Nearest Neighbours:**
  An algorithm that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.

- **Machine learning:**
  The concept that a computer program can learn and adapt to new data without human intervention.

- **Multilayer Perception:**
  A multilayer perceptron (MLP) is a class of feedforward artificial neural network (ANN).

- **Naïve Bayes:**
  A probabilistic machine learning model classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

- **Precision score:**
  Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

- **Receiver Operating Characteristics:**
  A graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied.

- **Sensitivity score:**
  Is the ratio of correctly predicted positive observations to the all observations in the actual classification.

- **Synthetic Minority Oversampling Technique:**
  An application used to deal with imbalanced classification where there are too few examples of the minority class for a model to effectively learn the decision.

- **Support Vector Machines:**
  A type of deep learning algorithm that performs supervised learning for classification or regression of data groups.

Appendix C- Implementation Details:

Multilayer Perception:

When creating the MLP model the main implementation details are described by the template that is highlighted in the code. Firstly, this involved importing the Portuguese Bank Marketing dataset as a CSV file. Then proceeded to pre-process the data by checking for missing variables and filling them. Once this had been completed some basic statistical analysis was conducted through firstly visualising the basic statistics of the data frame in Figure 1, followed by visualising the data frame through a correlation matrix and the distribution skewness of the seven main variables in Figures 2 and 3. Afterward, saw replacing classifiers factor outcomes (y column) with integers and dropping the unneeded columns in the data frame, leaving the “Age”, “Balance”, “Day”, “Duration”, “Campaign”, “pdays”, “Previous” and “y” attributes. Together with, converting NumPy objects to floats to work with Pytorch library and normalising the x label data then applying the smote technique to the x and y label data. After that, came the construction of the MLP model itself, which as seen in the code started with splitting the training and testing data by 70%/30%. Next constructing the MLP model saw two hidden layers and one output layer (multilayer perceptron’s) to train a classification problem. Moreover, applying PyTorch's sequential function enabled the model to include the seven predictors with 70 neurons in the first hidden layer followed by the activation function “ReLu”, through which after the second hidden layer goes from 70 neurons to 30 neurons which again after is parsed through the activation function. Lastly, the data is fed into the output layer of the MLP after passing through the 30 neurons, separating the output data into the two classes yes’ (class 1) = 1 and ‘no’ (class 2) = 2.

Lastly, the model applied a Log NLLLoss and a negative log-likelihood loss to train the stated classification problem since this method is useful with several classes. Alongside, a stochastic gradient descent (SGD) optimization method to update the weights that will encompass a learning rate of 0.01% for the MLP model. After creating the prediction function to predict the classes using the MLP model when executing it will have exactly 5000 epochs with a batch size of 30 and n_batches of 18 to maximise the training accuracy of the model. To summarise the model’s performance, confusion matrix and ROC curves and Classification reports were produced to depict the binary classifications performance depicting the yes/no outcome for the banks marketing campaign.

Support Vector Machines:

Similar to the MLP model using the same model template, this again involved using the imported dataset followed by using the same pre-processed data that was produced during the MLP model construction. der to make sure these data sections were consistent and hosted the same format for the mode. Furthermore, through building a default classifier “svclassifier = SVC(kernel='empyt', degree=8)”, this severed as the foundation to test and compare various SVM kernel methods used to assess accuracy effectiveness on the dataset of each mathematical functions. Kernels tested included; Polynomial, Gaussian, Hyperbolic and Sigmoid kernel. In summary, the Poly kernel proved to most effective with a hyper-parameter set at 8 degrees since higher degree polynomial kernels allow a more flexible decision boundary. Lastly, then similar to the MLP model to summarise the performance, confusion matrix, and ROC curves and Classification reports were produced to depict the binary classifications performance depicting the yes/no outcome for the banks marketing campaign.