A comparative study of a Multilayer Perceptron and a Support Vector Machine for predicting Airline Passenger Satisfaction

Yuna Lee

Abstract

This paper aims to critically evaluate, compare and contrast two algorithm models, the Support Vector Machine (SVM) and the Multilayer Perceptron (MLP) for a binary classification task predicting the satisfaction of airline passengers. The best performing models are selected through hyperparameter tuning via grid search with 10-fold cross validation. The evaluation of the two models involves comparing various evaluation metrics. For such classification problem, results suggest that the SVM slightly outperforms MLP in terms of performance metrics, while MLP demonstrates marginally better generalisation capabilities and greater efficiency in testing time.

1. Introduction

Customer satisfaction is crucial for businesses, particularly in competitive sectors such as the airline industry, where service quality directly impacts business performance [1]. Therefore, understanding and meeting customer expectations is essential for airlines to provide excellent service and ensure profitability [2]. Examining the satisfaction of airline passengers can serve as a valuable resource for devising strategies aimed at helping airline management in better understanding passenger perceptions of their services. This, in turn, can offer insights to airlines on effectively meeting customer needs, thereby enhancing management practices and strengthening competitive advantages.

The aim of this paper is to critically evaluate two algorithm models for a binary classification task of predicting the satisfaction of Airline passengers. The models considered are the Support Vector Machine (SVM) and the Multilayer Perceptron (MLP). The task involves predicting whether a person is classified as 'Satisfied' or 'Neutral or dissatisfied' based on a set of attributes. Through this analysis, we assess the effectiveness of MLP and SVM models in predicting passenger satisfaction prediction within the airline industry. By identifying their strengths and weaknesses, we provide insights to guide airline management decisions, ultimately enhancing management practices.

1.1 Support Vector Machines (SVM)

The Support Vector Machine (SVM) is a supervised machine learning algorithm designed to identify the optimal hyperplane, maximising the margin - the distance between the hyperplane, and the support vectors - the nearest data points from each class. SVM is specifically adept at handling classification tasks where data is linearly separable. However, it can also be transformed into higher-dimensional spaces through a kernel function, which allows for non-linear decision boundaries [3]. This versatility allows SVM to effectively capture complex data relationships and navigate non-linear decision boundaries. Nevertheless, training SVM models can be computationally intensive, especially with large datasets, as it involves solving a convex optimisation problem that scales quadratically with the number of samples.

1.2 Multilayer Perceptron (MLP)

The Multilayer Perceptron (MLP) is a type of artificial neural network composed of multiple layers of nodes or neurons. As a supervised learning classifier, MLP computes the weighted sum of its inputs, applies an activation function, and produces an output. Organised into layers – an input layer, one or more hidden layers, and an output layer – MLPs are trained using techniques such as backpropagation, which adjusts the weights of the neuron connections to minimise the difference between predicted and true outputs. Despite their ability to learn complex non-linear relationships in the data, MLPs typically require a large amount of training data to generalise well and prevent overfitting [4]. Moreover, MLPs are often considered as black-box models, implying limited interpretability, and making the understanding of their internal workings and predictions challenging.

2. Dataset

The dataset used for this classification task is obtained from the 'Airline Passenger Satisfaction' dataset from Kaggle [5]. It contains 129,880 entries, with 24 attributes (23 predictors and 1 target column) that classify passengers into 'Satisfaction' or 'Neutral or Dissatisfaction' categories based on a set of attributes. Table 1 presents numerical statistics for the numerical predictors. Notably, the 'Arrival Delay in Minutes' column contains 393 missing entries.

	mean	std	min	max
Age	39.43	15.12	7	85
Flight Distance	1190.32	997.45	31	4983
Departure Delay in Minutes	14.71	38.07	0	1592
Arrival Delay in Minutes	15.16	38.60	0	1584

Table 1: Summary statistics of numerical variables.

2.1 Initial Data Analysis

Understanding the business problem and identifying key features in the dataset are essential for modelling. As illustrated in Figure 1, the target variable displays a fairly balanced distribution, with 73,252 passengers (56.4%) classified as 'satisfied' and 56,428 (43.6%) as 'neutral or dissatisfied.' Figure 2 shows the distribution of categorical variables, 'Customer Type', 'Type of travel,' and 'Class', segmented by satisfaction. In particular, disloyal customers exhibit a lower satisfaction ratio compared to loyal customers. Furthermore, when segmenting satisfaction groups by trip type, customers on personal vacations demonstrate notably lower satisfaction ratios. Conversely, when categorised by class, passengers in the 'Business' class demonstrate higher levels of satisfaction. Figure 3 presents the correlation matrix of the attributes, highlighting a significant correlation coefficient of 0.97 between 'Arrival Delay in Minutes' and 'Departure Delay in Minutes' columns.

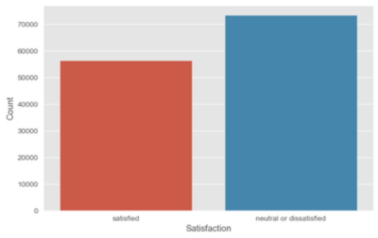

Figure 1: Barplot of the target variable.

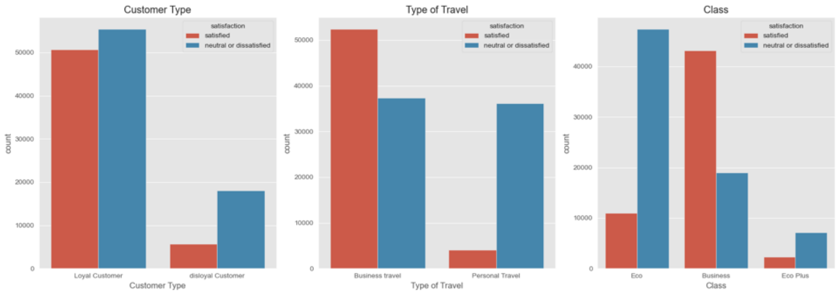

Figure 2: Barplot of categorical variables classified by satisfaction.

To pre-process the data, we replace missing values in the 'Arrival Delay in Minutes' column with the corresponding values from the 'Departure Delay in Minutes' column, given their high correlation. Additionally, following a methodology outlined in a prior study utilising this dataset [6], we identify and eliminate features that do not significantly contribute to predictive modelling. This involves dropping irrelevant columns such as "id" and removing columns with weak correlations (below 0.1) with the target column (Figure 3). These steps streamline the dataset, preserving only predictors essential for predicting the target variable, resulting in a refined set of

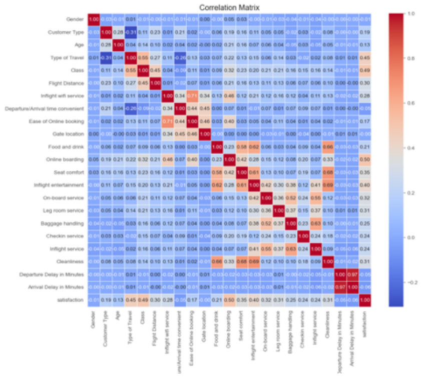

Figure 3: Heatmap correlation matrix of variables.

17 predictors. Furthermore, categorical variables are encoded, and numerical predictors are scaled to a range of 0 to 1.

3. Methods

In this section, we outline the methodology employed to train, validate, and test the SVM and MLP models, along with a description of their architectures and hyperparameters.

3.1 Methodology

Initially, we divided the dataset into training (80%) and testing (20%) sets. The training set is used for model training and validation, while the test set assesses the performance of the best models selected.

To identify the most effective SVM and MLP models, we conduct a grid search to optimise hyperparameters using 10-fold cross-validation. This consists of randomly partitioning the training set into 10 folds, with model performance evaluated as the average validation accuracy across these folds. Subsequently, the best model is chosen based on this metric.

In the comparative analysis between the two algorithms, the performance of the best-performing models is examined using the test data. Performance metrics of each model are compared to determine which one demonstrates better performance for the given task.

3.2 Architecture and Parameters used for SVM

The SVM architecture aims to find the optimal hyperplane to separate data into distinct classes while maximising the margin between them.

To identify the best-performing hyperparameters, we conduct a grid search focusing on parameters such as the kernel type, regularisation strength (C), and the gamma parameter. The choice of the kernel function plays a crucial role as it can transform the input data into a higher-dimensional space, facilitating the identification of a separating hyperplane. Considering previous research that utilised this dataset to build SVM models [6], we specifically explore the Linear and Gaussian Radial Basis (RBF) kernels, with a preference towards favouring the RBF kernel based on prior findings. Furthermore, we investigate various values of the regularisation parameter C. This parameter determines the balance between maximising the margin and minimising classification errors; smaller C values allow for a wider margin but may result in misclassifications, while larger C values lead to a narrower margin but reduce misclassification errors. Lastly, we examine the gamma parameter, especially in relation to the RBF kernel.

3.3 Architecture and Parameters used for MLP

The MLP architecture is more complex compared to SVM, requiring careful consideration of multiple hyperparameters. To comprehensively explore these parameters, we start with a random search approach, varying parameters like the number of hidden layers, neurons per layer, activation functions, learning rates, batch sizes, regularisation techniques, weight initialisation methods, and the number of epochs.

Our MLP architecture consists of an input layer with 17 neurons corresponding to the predictors in the dataset, along with a single output neuron tailored for binary classification. Employing the sigmoid output function, we represent the probability of belonging to a positive class, yielding generating output values within the range of 0 to 1. Additionally, we compute loss using binary cross-entropy, aligning with the binary nature of our classification task.

Effective optimisation training is essential in ensuring algorithmic generalisation and minimising loss functions to facilitate accurate predictions on new data. Drawing from previous research [7], we adopt the Adaptive Moment Estimation (Adam) optimisation algorithm to update neuron weights. Adam is a highly recommended optimiser in deep learning, dynamically adjusting learning rates and momentum [7], thereby enhancing accuracy and improving training and overall performance. Given its adaptive nature in learning rate and momentum, we refrain from using additional momentum techniques in our approach.

Following the evaluation of the random search results, we refined our focus to select a narrowed set of hyperparameters, delving deeper into tuning parameters such as learning rates, sizes of hidden layers, neuron sizes, and activation functions.

4. Results, Findings & Evaluation

4.1 Model Selection

Tables 2 and 3 present the results of the hyperparameter grid search and the selection of the best models for SVM and MLP, respectively. The outcomes are arranged in descending order of mean accuracy scores. For SVM (Table 2), the model appears to be significantly influenced by the kernel parameters. The Gaussian Radial Basis (RBF) kernel consistently outperforms the linear kernel, implying that the dataset may not be linearly separable in its original feature space. Instead, the RBF kernel's ability to map input features to a higher-dimensional space suggests that the dataset likely exhibits complex relationships not captured by a linear model. While variations in hyperparameters C and gamma lead to some changes in model performance, the differences are not significant. Higher values of C, indicating a more complex decision boundary, tend to yield slightly better performance. The distinction between configurations with 'auto' and 'scale' for gamma selection is not substantial; however, configurations with 'auto' tend to perform better when C values are equal. Overall, the highest score is achieved when employing the RBF kernel, with C = 10 and automatic gamma selection.

Support Vector Machine			
Kernel	**C**	**Gamma**	**Mean Score**
rbf	10	auto	**0.9606**
rbf	1	auto	**0.9548**
rbf	10	scale	**0.9544**
rbf	1	scale	**0.9453**
rbf	0.1	auto	**0.9430**
rbf	0.1	scale	**0.9339**
linear	10	scale	**0.8741**
linear	10	auto	**0.8741**
linear	0.1	scale	**0.8741**
linear	0.1	auto	**0.8741**
linear	1	scale	**0.8740**
linear	1	auto	**0.8740**

Table 2: Grid search results for SVM.

The initial random search for MLP hyperparameters (outlined in Appendix 2, Table 5), suggests a preference for lower values of weight decay (0 or 0.001) and learning rate (0.001) for optimal performance. Additionally, the choice of weight initialisation method significantly impacts model performance, with the He initialisation method generally producing higher accuracy scores. Simpler architectures with fewer hidden layers (1 or 2) and a moderate number of neurons (16, 32, 64) tend to outperform more complex models, highlighting the importance of model simplicity in preventing overfitting. While the tanh activation function is generally favoured over ReLU in the provided configurations, previous findings [7] suggest ReLU may perform better, prompting its inclusion in subsequent grid searches. Increasing the number of epochs or batch sizes does not consistently improve performance, emphasising the need to balance training time and model effectiveness. This exploration underscores that complexity does not guarantee superior performance; simpler models with appropriate regularisation often achieve comparable or better results.

The insights from the random search guide a more focused grid search, enabling systematic exploration of the hyperparameter space. Based on the results, we set the weight decay as 0 for no regularisation, utilised He initialisation for weights, and capped the maximum number of epochs at 30, with a batch size of 32. The refined grid search results are summarised in Table 3. Mean scores vary across different hyperparameter configurations, highlighting the significant impact of parameter choice on model performance. While both activation functions perform competitively, ReLU is slightly more prevalent, consistent with our assumption from previous literature [7]. Models with a hidden layer size of 32 neurons generally yield higher mean than those with fewer neurons (16). Additionally, models with two hidden layers generally outperform single-layer models. Notably, a lower learning rate of 0.001 typically leads to higher mean scores, indicating its effectiveness in training the neural network model on the dataset.

Multilayer Perceptron				
Activation function	**Num. of hidden neurons**	**Num. of hidden layers**	**Learning rate**	**Mean score**
relu	**32**	**2**	**0.001**	**0.9556**
relu	32	1	0.001	**0.9535**
tanh	32	2	0.001	**0.9532**
tanh	32	1	0.001	**0.9503**
relu	16	2	0.001	**0.9492**
relu	16	1	0.001	**0.9481**
tanh	16	2	0.001	**0.9477**
tanh	16	1	0.001	**0.9446**
relu	32	2	0.01	**0.9381**
tanh	16	1	0.01	**0.9359**
tanh	32	1	0.01	**0.9358**
relu	16	2	0.01	**0.9336**
relu	16	1	0.01	**0.9317**
tanh	16	2	0.01	**0.9277**
relu	32	1	0.01	**0.9267**
tanh	32	2	0.01	**0.9124**

Table 3: Grid search results for MLP.

4.2 Algorithm Comparison

Metric	SVM	MLP
Train accuracy	0.9678	0.9585
Test accuracy	0.9595	0.9550
Precision	0.9700	0.9643
Recall	0.9351	0.9302
F1 Score	0.9522	0.9469
ROC - AUC	0.9566	0.9520
Testing time	17.7756	0.2428

Table 4: Performance metrics of SVM and MLP

Table 4 and Figure 6 show the performance metrics and confusion matrix results of the top performing SVM and MLP models during testing. Overall, while there are some discernible differences in performance metrics, with SVM generally outperforming MLP, the disparities are not substantial. Upon reviewing Table 4, the SVM model achieves slightly higher training and test accuracy, precision, recall, and F1 scores compared to the MLP model. Moreover, the area under the ROC curve (ROC-AUC) displayed in Figure 7 is very slightly higher for the SVM model. In our business context, high precision is critical for accurately identifying satisfied customers as it allows airlines to understand what factors contribute to customer satisfaction [8]. Both models demonstrate high precision scores of 0.97 and 0.9643 for SVM and MLP, respectively, indicating positive outcomes in identifying satisfied customers. Therefore, while SVM may seem slightly superior, the distinction is negligible in practical terms.

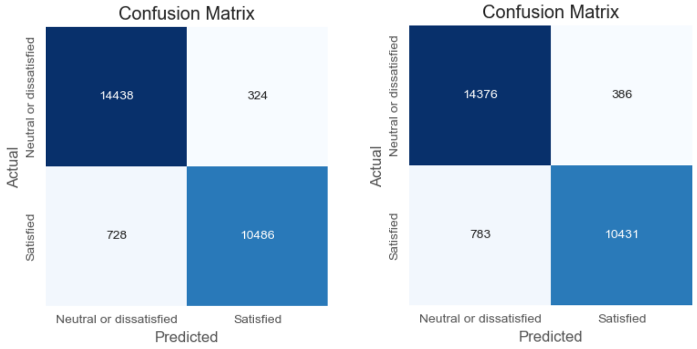

Figure 6 - Confusion matrix results of SVM (left) and MLP (right),

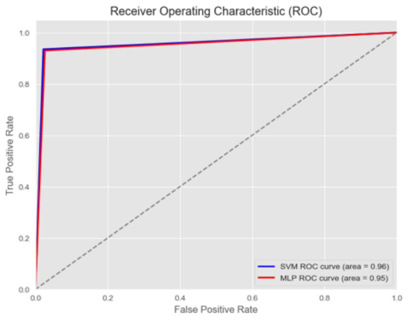

Figure 7 – ROC curve of SVM (blue) and MLP (red).

Despite exploring early stopping for MLP to address overfitting concerns, it did not significantly improve outcomes (Appendix 2, Table 6), leading us to abstain from its implementation. Particularly, there is a difference between the train accuracy and test accuracy for SVM, which appears to be relatively higher compared to MLP. This indicates a potential overfitting issue due to the stronger, more complex non-linear kernel decision boundary and higher regularisation parameter C (10) used in SVM. On the other hand, MLP shows a narrower gap between training and test accuracy, suggesting better generalisation capabilities. While addressing the overfitting in SVM requires investigation and potential adjustments to complexity or regularisation, the relatively minor difference between train and test accuracy suggests it may not be critical. Furthermore, MLP displays notably faster testing times, taking less than 0.3 seconds compared to SVM taking almost 18 seconds, indicating superior prediction efficiency. In summary, while SVM slightly outperforms MLP in various metrics, MLP excels in testing time efficiency, and the choice between the two should consider balancing performance metrics and computational efficiency.

5. Conclusion

In conclusion, this study critically evaluated the performance of SVM and MLP models for predicting airline passenger satisfaction. Through a comprehensive analysis of hyperparameters and model selection, we gained insights into the strengths and weaknesses of each algorithm in addressing the binary classification task. The results revealed that SVM generally slightly outperformed MLP in various metrics. However, SVM exhibited signs of overfitting with higher train accuracy compared to test accuracy, possibly due to its stronger, more complex non-linear kernel decision boundary. In contrast, MLP showed better generalisation capabilities and faster testing time efficiency, despite slightly lower performance metrics than SVM.

The study emphasised the significance of balancing model complexity with performance metrics. Although SVM demonstrated superior performance, its complexity raised potential overfitting concerns. In contrast, although we expected some overfitting with MLP due to its ability to capture complex relationships, our model's simpler architecture exhibited better generalisation capabilities and lower testing time. Nonetheless, the overfitting issue with SVM was not considered significant. Future research could focus on mitigating the overfitting issue observed in SVM models by exploring regularisation techniques or reducing model complexity.

6. Reference

[1] W. Li, S. Yu, H. Pei, C. Zhao and B. Tian, “A hybrid approach based on fuzzy AHP and 2-tuple fuzzy linguistic method for evaluation in-flight service quality”, *Journal of Air Transport Management,* vol. 60, pp. 49-64, Jan. 2017.

[2] S. Leon and J.C. Martín, “A fuzzy segmentation analysis of airline passengers in the U.S. based on service satisfaction”, *Research in Transportation Business and Management*, vol. 37, Dec. 2020.

[3] A. Ukil, “Support Vector Machines,” in *Intelligent Systems and Signal Processing in Power Engineering*, Berlin, Germany: Springer, 2007, ch. 4, pp. 161-226.

[4] K. Gurney, “Multilayer nets and backpropagation,” in *An Introduction to Neural Networks*, London, UK: Taylor & Francis Group, 1997, ch. 6, pp. 41-56.

[5] *Airline Passenger Satisfaction,* Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction

[6] S.H. Hong, B. Kim and Y.G. Jung, “Correlation Analysis of Airline Customer Satisfaction using Random Forest with Deep Neural Network and Support Vector Machine Model”, *International Journal of Internet, Broadcasting and Communication*, vol. 12, no. 4, pp. 26-32, Aug.

[7] S. Ouf, “An Optimized Deep Learning Approach for Improving Airline Services’*”, Computers, Materials & Continua*, vol. 75, no.1, pp. 1213-1233, Dec. 2022.

[8] B.H. Hayadi, J.M. Kim, K. Hulliyah and H.T. Sukmana, “Predicting Airline Passenger Satisfaction with Classification Algorithms”, *International Journal of Informatics and Information System*, vol. 4, no.1, pp. 82-94, Mar. 2021.

Appendix 1 – Glossary

Term	Definition
Kernel	In SVM, kernels transform data which may not be linearly separable in the original feature space to a space where it can be separated by a hyperplane. E.g. linear, polynomial, rbf, sigmoid kernel.
Regularisation	Method to prevent overfitting and improve the generalisation of a model. In SVM, regularisation is controlled by a parameter C.
Gaussian Radial Basis Function (RBF)	Type of kernel function used to transform into a higher-dimensional space, where it computes the similarity between pairs of data points by measuring their Euclidean distance in the input space.
Gamma	The parameter gamma is a SVM hyperparameter regulating the impact of individual training samples on the decision boundary. It specifically determines the extent of the RBF kernel and the adaptability of the decision boundary.
He initialisation	Weight initialisation method named after Kaiming He, addresses the issue of vanishing gradients during training of neural networks. Sets the initial weights according to a specific distribution, typically a Gaussian or normal distribution.
Adaptive Moment Estimation (Adam) optimisation	Optimisation algorithm that merges the benefits of two popular optimisation methods: RMSprop and momentum. It adaptively adjusts the learning rates for each parameter during training by managing two moving averages of gradients, the first moment and the second moment.
Binary entropy loss	Loss function used in binary classification tasks measuring the difference between two probability distributions, the true distribution of the target variable and the predicted distribution generated by the model.
Momentum	Method to accelerate convergence during gradient descent optimisation. It addresses challenges such as slow convergence or getting stuck in local minima.
Sigmoid function	Mathematical function particularly used for binary classification tasks that transform input values to an output range between 0 and 1, where it predicts the probability that a given input belongs to a certain class.
Overfitting	A common problem where a model learns to perform exceptionally well on the training data but fails to generalise well to unseen data.
Precision	Metric used to evaluate the performance of a classification model measuring the proportion of true positive predictions among all positive predictions. Precision = $\frac{True\ Positives}{True\ Positives+False\ Positives}$

Appendix 2- Implementation details

Table 5 shows the outcomes of the random search phase, where we investigated various parameters to streamline our focus to identify a narrowed set of hyperparameters, for subsequent grid search. The findings from this phase are summarised briefly in the Model Selection section of the paper.

Following the grid search and finding the optimal hyperparameters, we proceeded to retrain the MLP model using these parameters. Additionally, we implemented early stopping to potentially enhance results, though generalisation was not a significant concern in our initial model. Table 6 shows the performance metrics for MLP with Early Stopping and MLP without Early Stopping.

Early stopping is a regularisation technique aimed to prevent overfitting by stopping the training process before the model starts to overfit the training data. Therefore, we anticipated that while the train accuracy might remain similar or decrease slightly, the test accuracy could potentially improve compared to the model trained without early stopping. However, the observed outcomes were unexpected.

Although the training accuracy was slightly lower for the MLP model with early stopping, as anticipated, the test accuracy was also lower, which came as a surprise. While the performance regarding recall saw an increase, metrics such as precision, F1 score, and ROC-AUC decreased. This decline, particularly in precision—a critical evaluation metric for our business problem—raises concerns. Hence, early stopping in this scenario did not yield significant enhancements in performance metrics. Therefore, for this specific dataset and task, it was decided that early stopping may not be necessary or beneficial.

Weight decay	Learning rate	Weight initialisation	Num. of hidden layers	Num. of hidden neurons	Activation function	Max epochs	Batch size	Mean score
0	0.001	he	1	32	tanh	30	32	**0.9506**
0.001	0.001	he	2	16	relu	30	32	**0.9464**
0	0.001	he	3	16	tanh	30	128	**0.9451**
0	0.001	he	2	16	tanh	20	64	**0.9427**
0	0.001	he	1	64	tanh	20	128	**0.9411**
0.001	0.01	xavier	3	128	relu	30	64	**0.9261**
0.01	0.001	he	3	16	tanh	10	128	**0.9196**
0.001	0.001	xavier	2	128	tanh	20	32	**0.9135**
0.01	0.001	he	1	32	tanh	30	32	**0.8988**
0.01	0.01	xavier	2	128	relu	10	128	**0.8887**
0.01	0.01	xavier	1	128	tanh	30	128	**0.8568**
0.01	0.1	xavier	2	16	tanh	30	64	**0.5648**
0.1	0.2	he	2	32	relu	20	128	**0.5648**
0.1	0.01	xavier	2	64	tanh	20	128	**0.5648**
0.1	0.01	xavier	1	64	relu	10	128	**0.5648**
0.1	0.1	xavier	2	128	tanh	10	64	**0.5648**
0.1	0.1	xavier	2	64	relu	10	128	**0.5648**
0.1	0.2	he	1	16	relu	30	32	**0.5648**
0.1	0.1	he	3	16	tanh	20	64	**0.5648**
0.001	0.1	xavier	1	32	relu	20	64	**0.5648**
0.001	0.1	xavier	1	16	relu	10	32	**0.5519**
0.1	0.2	he	1	16	relu	10	64	**0.5389**
0.001	0.2	xavier	1	16	tanh	10	64	**0.5291**
0.01	0.2	xavier	1	64	relu	20	128	**0.5130**
0	0.1	he	1	32	tanh	20	128	**0.5000**
0.001	0.1	he	3	64	relu	10	32	**0.4741**
0.01	0.2	he	2	64	tanh	20	64	**0.4741**
0.001	0.2	he	2	64	relu	30	128	**0.4741**
0	0.1	he	3	128	tanh	20	32	**0.4481**
0	0.1	he	3	128	relu	30	64	**0.4352**

Table 5: Random search results for MLP. Outcomes are arranged in descending order of mean accuracy scores.

Metric	MLP - Early Stopping	MLP - No Early Stopping
Train accuracy	0.9524	0.9585
Test accuracy	0.9479	0.9550
Precision	0.9358	0.9643
Recall	0.9434	0.9302
F1 Score	0.9399	0.9469
ROC - AUC	0.9474	0.9520

Table 6: Performance metrics for MLP with Early Stopping and without Early Stopping.