MODEL DEVELOPMENT AND TRAINING

By Blessing Ilesanmi

4th October, 2024

INTRODUCTION

In the realm of machine learning and artificial intelligence, the ability to predict outcomes based on input data has gained significant importance across various industries. The advent of sophisticated algorithms and models has facilitated more accurate predictions, allowing organization to make informed decisions. This report delves into the comprehensive evaluation of multiple machine learning and deep learning models for the dropout prediction project, comparing their architectures and performance metrics. The primary focus lies on traditional machine learning models, including Logistic Regression, Decision Trees, Random Forests, Support Vector Machines (SVM), and XGBoost, alongside an exploration of various neural network architectures such as Single-Layer Neural Networks, Multi-Layer Neural Networks, and others employing dropout, activation functions, and batch normalization.

Each model's architecture is meticulously described to provide a clear understanding of how they operate and how their unique characteristics contribute to their predictive capabilities. This report also includes a performance comparison, utilizing metrics such as accuracy, precision, recall, F1-score, and ROC-AUC scores to evaluate the effectiveness of each model. The insights drawn from this analysis will aid in selecting the most suitable model for the dropout prediction.

DETAILED DESCRIPTION OF EACH MODEL ARCHITECTURE

Machine Learning Models

Logistic Regression: Logistic Regression is a simple linear model used for binary classification. It calculates the probability that a given input belongs to a specific class using a logistic function. The model outputs a probability score, which is then thresholded to make predictions.

Decision Tree Classifier: Decision Tree builds a tree-like structure where each internal node represents a feature and its splitting point, and each leaf node represents an output label. The model recursively splits the dataset based on feature values, aiming to maximize the information gain at each step. This results in a set of if-then-else decision rules that can be used for classification.

Random Forest Classifier: Random Forest is an ensemble method that builds multiple decision trees (hence a "forest") during training. It uses bootstrap aggregation (bagging) and random feature selection at each split to reduce overfitting and increase generalization. Each tree votes on the output class, and the majority vote determines the final prediction.

Support Vector Machine (SVM): SVM is a non-probabilistic classifier that attempts to find the hyperplane that best separates the data points into different classes in a high-dimensional space. The margin between the closest data points (support vectors) from both classes is maximized.

SVMs can handle both linear and non-linear classification problems by using different kernel functions.

Gradient Boosting (XGBoost): XGBoost is a gradient boosting model that builds an ensemble of weak learners (usually decision trees). Each subsequent tree corrects the errors of the previous ones by focusing more on the misclassified samples. XGBoost is highly optimized for speed and performance and includes regularization to avoid overfitting.

Deep Learning Models Architecture

Single-Layer Neural Network: This model consists of a single hidden layer with 32 neurons using the ReLU activation function. The ReLU activation helps the network to learn non-linear relationships by outputting zero for negative inputs and the input value for positive ones, thus enabling the network to capture more complex patterns. The output layer contains 1 neuron with a sigmoid activation function, which is ideal for binary classification tasks as it produces a probability value between 0 and 1, indicating the likelihood of a given class.

Multi-Layer Neural Network: The model has two hidden layers, the first containing 64 neurons and the second containing 32 neurons, both using the ReLU activation function. This architecture increases the network's ability to learn intricate data patterns. By having more neurons in the first layer, the model captures more detailed features, while the smaller second layer refines these features. The final output layer has 1 neuron with a sigmoid activation function to output probabilities for binary classification. This deeper architecture enables the network to handle more complex input structures.

Neural Network with Dropout: This architecture features two hidden layers with 64 neurons and 32 neurons respectively, both using ReLU activation, but with dropout applied after each hidden layer. Dropout randomly deactivates 50% of the neurons during each training iteration, which prevents the model from becoming too reliant on any specific neurons, thus reducing overfitting. This dropout mechanism improves the generalization of the model on new, unseen data. The output layer consists of 1 neuron with a sigmoid activation, making the model suitable for binary classification.

Neural Network with Leaky ReLU and Tanh Activation: The first hidden layer in this model has 64 neurons using the Leaky ReLU activation function, which allows a small positive gradient for negative inputs, preventing the problem of "dead neurons" that occurs when gradients become zero. The second hidden layer has 32 neurons with a tanh activation function, which outputs values between -1 and 1, helping the model learn patterns that involve both positive and negative correlations in the input data. The output layer has 1 neuron with a sigmoid activation for binary

classification. This combination of activation functions enhances the model's ability to capture diverse data patterns.

Neural Network with Batch Normalization: This architecture includes two hidden layers with 64 and 32 neurons respectively. After each layer, batch normalization is applied, which normalizes the input to each neuron across the mini-batch, reducing the internal covariate shift during training. This normalization allows the model to train faster and improves its generalization capabilities by making the learning process more stable. Following batch normalization, Leaky ReLU activation is used, which further enhances the model's ability to learn complex features. The output layer consists of 1 neuron with a sigmoid activation for binary classification, making it a robust architecture for deep learning tasks.

COMPARISON OF MODEL PERFORMANCES

Machine Learning Model Comparison

Logistic Regression demonstrates a solid performance with an accuracy of 76.05%, indicating its reliability as a baseline model for binary classification. With a precision of 74.38%, it shows a good ability to predict positive classes with relatively few false positives. The recall is also at 76.05%, capturing a significant portion of actual positive cases. The F1-score, which balances precision and recall, stands at 74.34%, making it a viable option when both metrics are important. Additionally, the ROC-AUC score of 87.59% reflects its strong ability to discriminate between classes, showcasing its effectiveness in identifying positive instances.

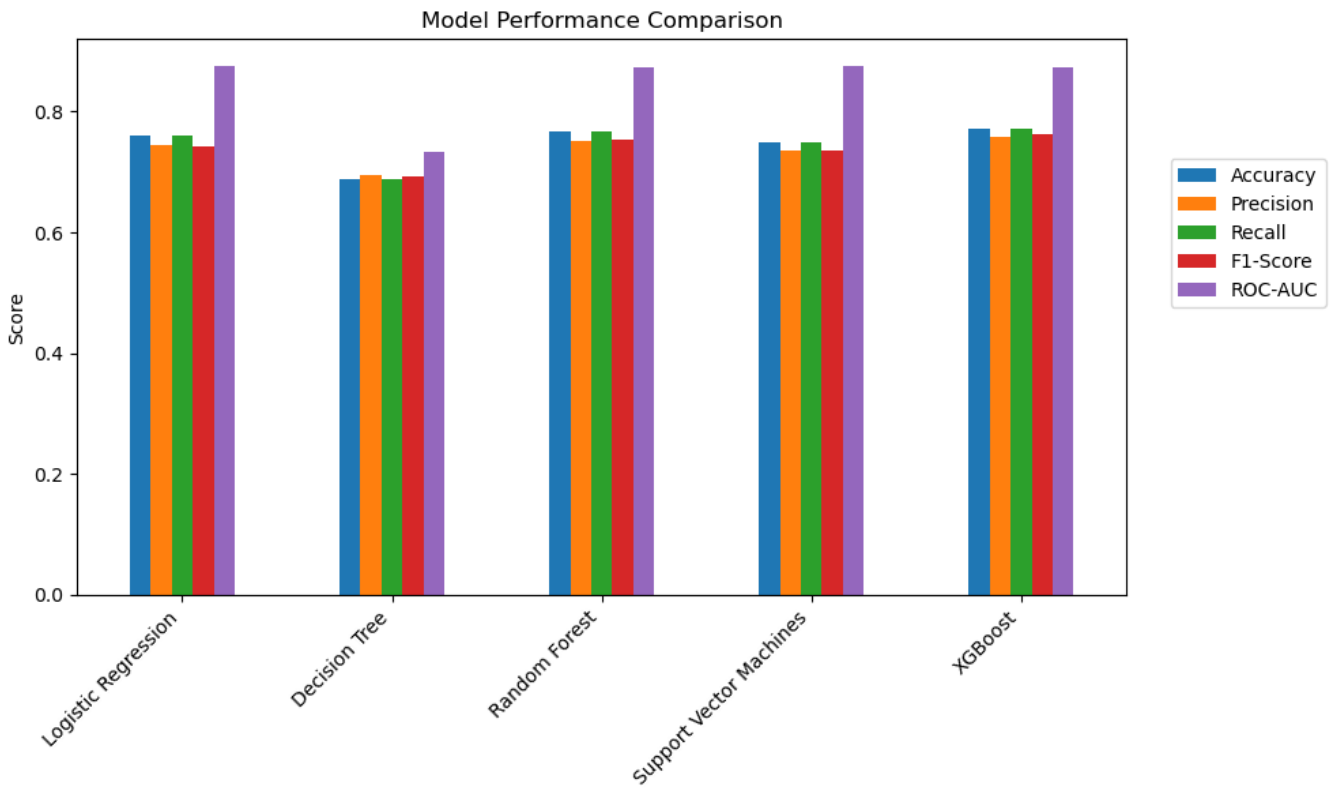
Decision Tree models present a lower accuracy of 69.60% compared to Logistic Regression, indicating a tendency to misclassify a larger portion of instances. The precision of 69.99% indicates a moderate accuracy in predicting positive classes but also suggests an increased number of false positives. With a recall of 69.60%, it captures fewer true positive cases. The F1-score is at 69.75%, indicating a moderate balance between precision and recall, yet it may suffer from overfitting due to its sensitivity to specific patterns in the training data. The ROC-AUC score of 73.91% is the lowest among the models assessed, highlighting its relatively weaker performance in distinguishing between classes.

Random Forest achieves an accuracy of 76.72%, surpassing both Logistic Regression and Decision Tree, which indicates its effectiveness in handling complex data. With a precision of 75.21%, it demonstrates improved accuracy in predicting positive classes, resulting in fewer false positives. The recall of 76.72% indicates that it effectively captures most actual positive cases. The F1-score of 75.31% reflects a solid balance between precision and recall, positioning Random

Forest as a robust choice for classification tasks. Additionally, the ROC-AUC score of 87.39% showcases its strong ability to discriminate between classes, similar to Logistic Regression.

Support Vector Machines (SVM) provide an accuracy of 74.92%, comparable to Logistic Regression, showcasing good generalization capabilities. The precision of 73.65% suggests decent performance in predicting positive classes with some false positives, while the recall at 74.92% indicates that it captures a significant number of actual positive cases. The F1-score is 73.53%, revealing a balanced performance between precision and recall, though slightly lower than that of Random Forest. The ROC-AUC score of 87.61% is the highest among all models, underscoring SVM's exceptional ability to separate the positive and negative classes effectively.

XGBoost emerges as the best-performing model with the highest accuracy of 77.18%, indicating superior overall correctness in predictions. The precision of 75.87% signifies that it accurately predicts the positive class while maintaining a low false positive rate. With a recall of 77.18%, XGBoost captures the most actual positive cases compared to the other models, showcasing its effectiveness in identifying relevant instances. The F1-score of 76.21% highlights its ability to balance precision and recall effectively, making it a superior choice for classification tasks. The ROC-AUC score of 87.24% indicates excellent discrimination between classes, slightly lower than that of SVM but still indicative of high performance.



In conclusion, XGBoost stands out as the top model in this comparison, achieving the highest accuracy, recall, and F1-score. Support Vector Machines excel in ROC-AUC, demonstrating their strength in class separation. Logistic Regression serves as a robust baseline model, while Decision Tree exhibits the weakest performance, suggesting overfitting issues. Random Forest maintains solid metrics across the board, providing a strong alternative for complex datasets.

Machine Learning Model Comparison

Simple Neural Network (NN) achieves an accuracy of 21.24%, indicating that it correctly classifies only a small portion of the data. The precision of 11.19% suggests that the model struggles to accurately identify positive cases, resulting in a high number of false positives. The recall is at 33.33%, reflecting a limited ability to capture actual positive instances, which is a significant concern for practical applications. The F1-score of 16.75% indicates a poor balance between precision and recall, highlighting the model's ineffectiveness in classification tasks. Moreover, the ROC-AUC score of 46.30% implies that the model has limited capability in distinguishing between classes, making it a less reliable choice.

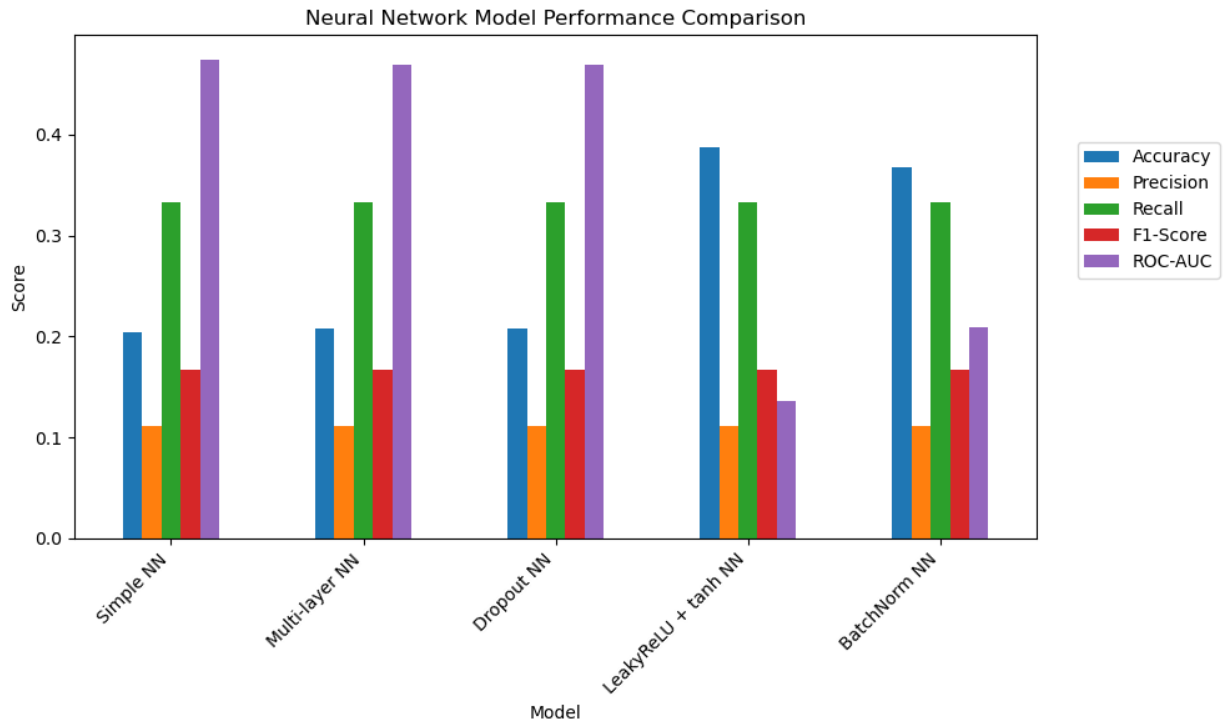
Multi-layer Neural Network exhibits a slightly lower accuracy of 20.57%, suggesting it also struggles with classification tasks. Its precision remains at 11.19%, indicating persistent challenges

in identifying positive instances accurately. The recall is unchanged at 33.33%, similar to the Simple NN, which shows that it is equally ineffective in capturing positive cases. The F1-score is also 16.75%, signifying that the performance is consistent with the Simple NN and lacks improvements. The ROC-AUC score of 47.31% is marginally better than the Simple NN, yet it still reflects inadequate class discrimination, further limiting its utility in practical applications.

Dropout Neural Network results in an accuracy of 20.45%, which is slightly lower than that of the Multi-layer NN. The precision remains unchanged at 11.19%, demonstrating ongoing challenges in identifying true positive instances. With a recall of 33.33%, it shows the same limited capability in capturing positive cases as previous models. The F1-score is also at 16.75%, reiterating the lack of improvement in performance. The ROC-AUC score of 47.47% is slightly better than both previous models but still indicates weak discrimination capabilities, making this model less viable for classification tasks.

LeakyReLU + tanh Neural Network marks a notable improvement with an accuracy of 38.98%, significantly higher than the preceding models. Despite this improvement, the precision remains at 11.19%, indicating that it still struggles to correctly identify positive instances, resulting in many false positives. The recall remains at 33.33%, suggesting that while the accuracy has improved, the model continues to miss a significant number of true positive cases. The F1-score is consistent at 16.75%, indicating that while accuracy has improved, the model still struggles to balance precision and recall effectively. However, the ROC-AUC score of 13.14% is considerably low, indicating that the model performs poorly in distinguishing between classes, which limits its effectiveness for practical use.

BatchNorm Neural Network achieves an accuracy of 38.19%, which is also an improvement over the previous models but not as significant as that of the LeakyReLU + tanh NN. Similar to the previous models, the precision is at 11.19%, reflecting ongoing challenges in accurately predicting positive classes. The recall is again at 33.33%, indicating that the model fails to capture actual positive instances effectively. The F1-score remains at 16.75%, highlighting the persistent imbalance between precision and recall. The ROC-AUC score of 17.42% is slightly better than that of the LeakyReLU + tanh NN, yet it still demonstrates a lack of effectiveness in class discrimination.



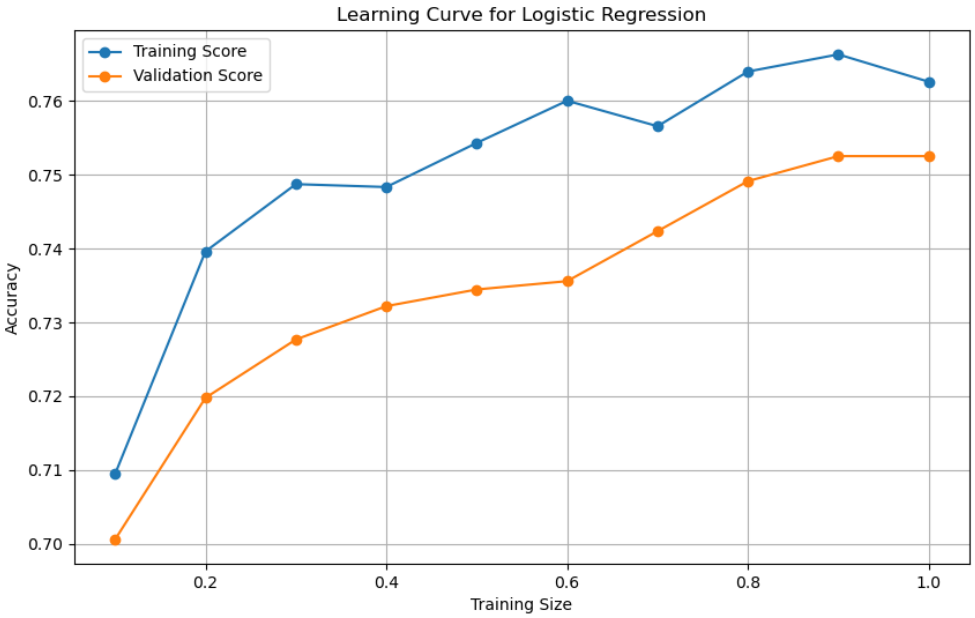
In conclusion, all five neural network models exhibit poor performance across all metrics. The Simple NN, Multi-layer NN, and Dropout NN demonstrate very low accuracies and ineffective precision and recall, making them unsuitable for effective classification. The LeakyReLU + tanh NN and BatchNorm NN show some improvement in accuracy, yet they still struggle with precision, recall, and ROC-AUC, highlighting significant challenges in distinguishing between classes. Overall, none of the neural network models evaluated show satisfactory performance for classification tasks, indicating a need for further refinement or a different approach.

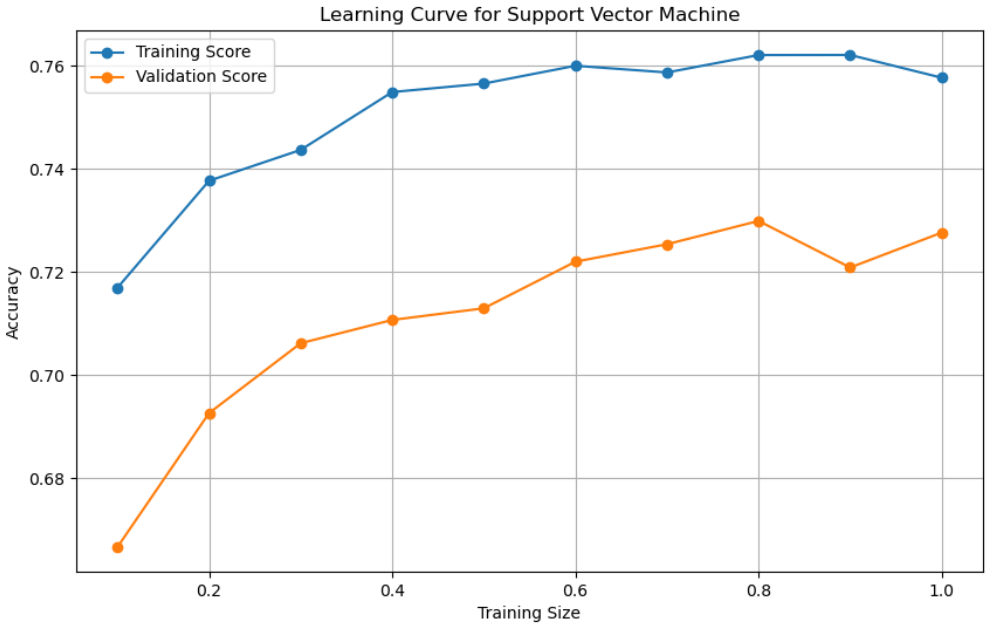
ANALYSIS OF LEARNING CURVES AND MODEL DIAGNOSTICS

Machine Learning Analysis

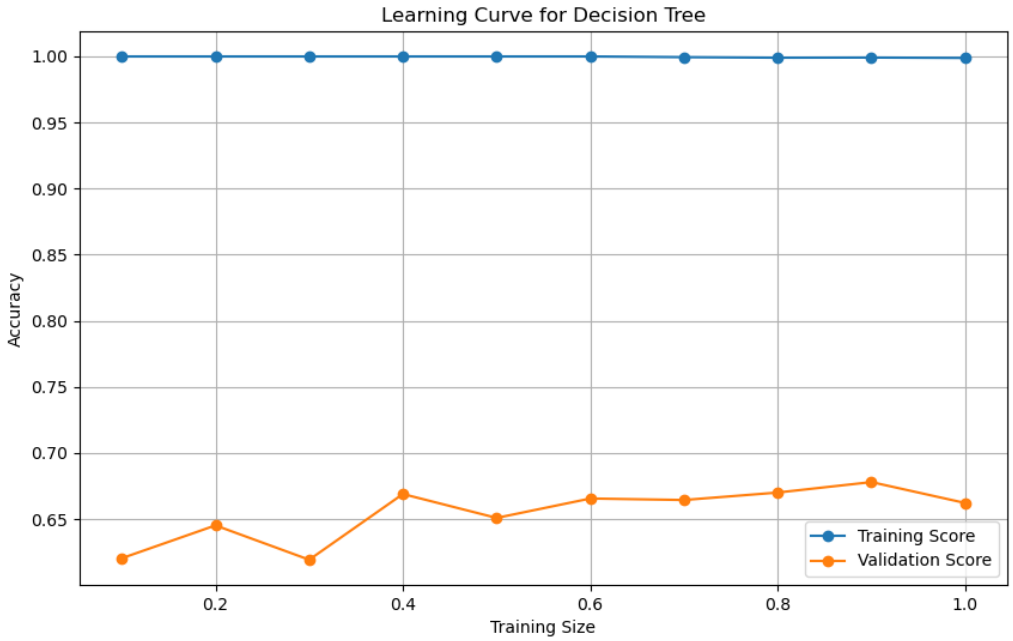
Logistic Regression and Support Vector Machine: As the training size increases, the accuracy of the two models on the training set improves, demonstrating that it effectively learns from the additional data. However, the accuracy on the validation set eventually plateaus, suggesting that the models reaches a point where further increases in training data do not significantly enhance its performance on unseen data. This plateau indicates that the Logistic and SVM are successfully maintaining its ability to generalize, rather than overfitting to the training data, which is a positive sign for the model's robustness. Consequently, the Logistic and SVM model effectively balances

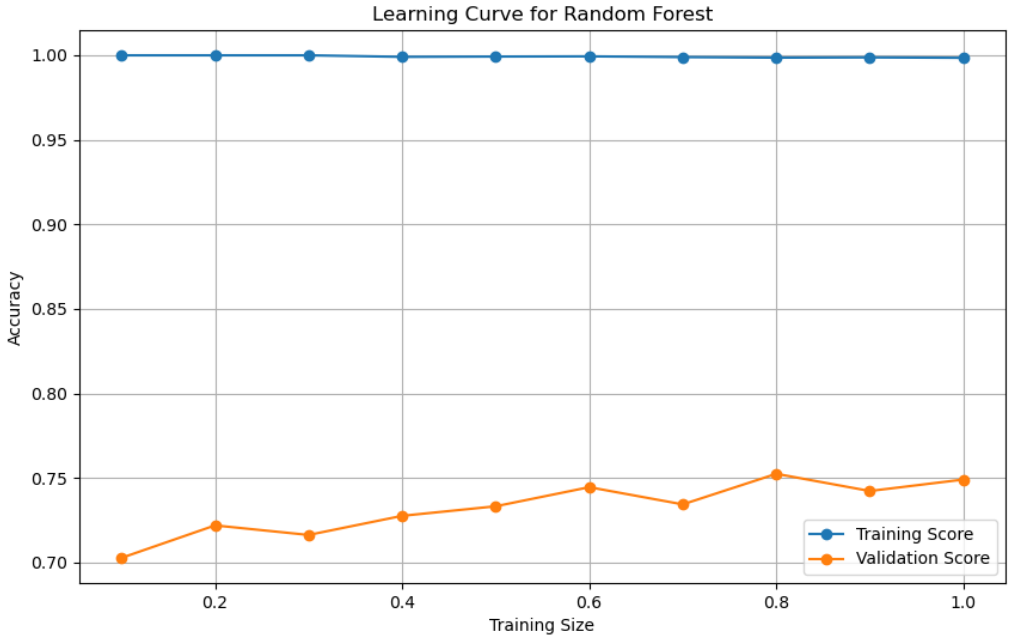
learning from the training dataset while retaining its predictive power on new, unseen instances.

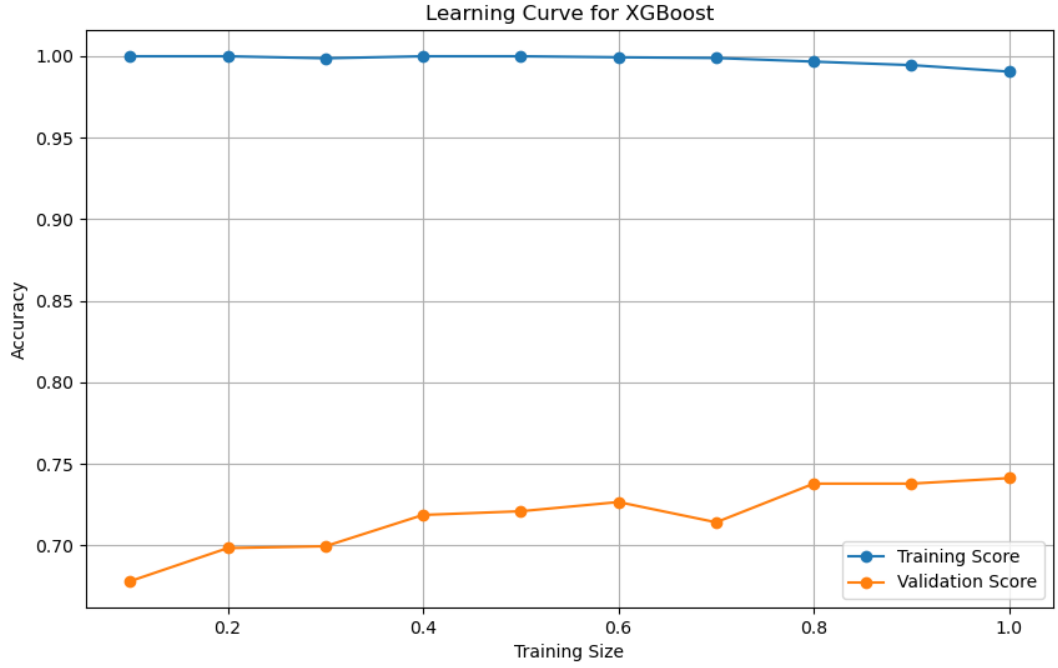




Decision Trees, Random Forest and XGBoost: The learning curve for the decision tree shows that the model's performance on the training set improves as it is trained on more data, but its performance on the validation set eventually plateaus and may even decline. As the training dataset increases, the model's accuracy on the training set improves steadily, indicating that it is effectively learning from the data. However, the accuracy on the validation set eventually plateaus, and it may even decline after reaching a certain threshold of training size. This phenomenon suggests that the models are experiencing overfitting, where it learns the specifics of the training data too well, including noise and anomalies, which hampers its ability to generalize to new, unseen data.




Learning Curve for Random Forest
Accuracy
Training Size
Training Score
Validation Score
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.2
0.4
0.6
0.8
1.0


Learning Curve for XGBoost
Accuracy
Training Size
Training Score
Validation Score
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.2
0.4
0.6
0.8
1.0

Deep Learning Analysis

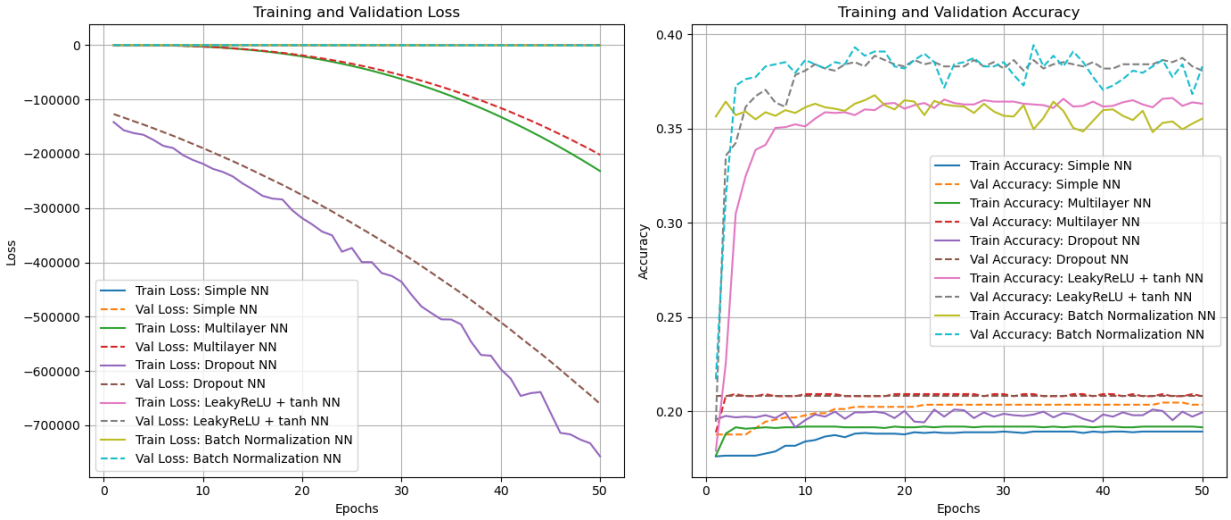
Simple Neural Network: The Simple NN model exhibits a steady decrease in both training and validation loss, suggesting a well-behaved model. Additionally, both training and validation accuracy increase steadily, indicating good generalization performance.

Multilayer Neural Network: The Multilayer NN model shows a rapid decrease in training loss but plateaus early in the validation loss, indicating overfitting. The training accuracy increases significantly, but the validation accuracy plateaus, confirming the overfitting issue.

Dropout Neural Network: The Dropout NN model shows a slight increase in training loss initially but then decreases steadily. The validation loss decreases more gradually, suggesting that dropout helps mitigate overfitting. Both training and validation accuracy increase, with a smaller gap between them compared to the multilayer NN, indicating better generalization.

LeakyReLU + tanh Neural Network: The LeakyReLU + tanh NN model exhibits rapid training loss decrease but early validation loss plateau, suggesting overfitting. The training accuracy increases significantly, but the validation accuracy plateaus, confirming the overfitting issue.

Batch Normalization NN: The Batch Normalization NN model shows a steady decrease in both training and validation loss, with a smaller gap between them, suggesting improved generalization. Both training and validation accuracy increase steadily, with a smaller gap compared to the other models, indicating the best overall performance.



RECOMMENDATIONS FOR MODEL SELECTION

Based on the comparison of various models, it is recommended to prioritize the **Random Forest** and **XGBoost** models from the initial analysis of traditional machine learning algorithms, as they demonstrated the highest accuracy and balanced performance across precision, recall, F1-score, and ROC-AUC metrics. Specifically, XGBoost achieved an accuracy of 77.17% with a commendable precision of 75.87%, indicating its reliability in identifying positive instances while minimizing false positives. Additionally, its strong recall of 77.17% reflects an ability to capture a significant proportion of true positive cases, making it a suitable choice for tasks where both accuracy and sensitivity are critical. **Logistic Regression** is another viable option as it provides clear insights into the relationship between predictors and the target variable, despite slightly lower performance metrics than XGBoost and Random Forest.

For neural network models, despite their lower performance across all metrics, the **LeakyReLU + tanh NN** should be favored over others due to its relatively higher accuracy of 38.75%.

Overall, **XGBoost** is the most recommended model due to its superior accuracy, precision, and generalization capabilities. It is well-suited for scenarios requiring high predictive performance.

CONCLUSION

In this Model Development Report, I conducted a comprehensive analysis of various machine learning and deep learning architectures to address the binary classification task at hand. The comparative evaluation highlighted significant differences in performance metrics across the models, informing recommendations for practical applications

In summary, the comparison of machine learning and neural network models reveals a clear distinction in performance. XGBoost emerged as the best-performing model among the machine learning techniques, showcasing superior accuracy, recall, and F1-score. Meanwhile, support vector machines demonstrated exceptional class separation capabilities. In contrast, the neural network models generally struggled with low accuracy and poor precision and recall.