Diabetes Prediction using Perceptron

Vadde Venkata Kamala Sindhoori
University of Adelaide
Adelaide, South Australia, 5000
a1905610@adelaide.edu.au

Abstract

This paper introduces deep learning methodology for predicting diabetes using perceptron. The study begins with building baseline model and various methods are explored to improve the baseline model. The techniques include hyper paramteric optimisation, using ensemble methods, data pre-processing improvements and Feature engineering and Feature selection. The findings suggest that these techniques/methods can be employed to improve the performance of basic perceptron model and achieve good results in predicting diabetes.

1. Introduction

Diabetes is a major health concern which can cause dysfunction and worsening of complications related to other body parts like kidney failures, heart failures and eye related diseases. The expected number of people that are going to be affected by diabetes is going to reach 550 million by 2030. Furthermore, the total number of people with diabetes is expected to rise to 640 million by 2040 indicating that every single person out of 10 would be affected by diabetes [7] indicating that a significant number of efforts and attention should be given to this problem.

Recent advancements in machine learning (ML) led to the development of various models which can help in early detection of diabetes and individualized treatments [2]. In this paper, the focus is on implementing perceptron model to predict diabetes using the patient data from PIMA Indian Diabetes Dataset. The paper involves building the initial baseline model and applying various additional improvements to the model which include hyper parametric optimization, bagging and boosting, improvements to preprocessing of data and feature engineering and feature selection

In Section 2, previous works related to predicting diabetes using various ML algorithms and also predicting diabetes by using perceptron were discussed. Furthermore, in the section 3, the data used, its fetaures and the models used on the data were discussed. Finally, the section 4 involves the discussion the performance of various models on the data and which models using perceptron perform best in predicting diabetes.

2. Related Works

2.1. Diabetes Prediction using ML Algorithms

Various ML models have been developed and used to predict diabetes. Alam et al. [1] have explored various supervised methods like Support Vector Machines (SVM), Random Forest, Decision Trees which have predicted diabetes fairly well. They have observed that Decision trees are the most effective when it comes to predicting diabetes with an accuracy of 79.52 percent over SVM and Random Forest.

Similarly, Majumdar and Vaidehi [2] have explored various methods like SVMs , Decision Trees and Logistic regression and have deducted that combining various models by also considering additional predictors like lifestyle would lead to higher prediction accuracy. They have also explored Artificial Neural Networks and Fuzzy logics to improve prediction accuracy. In their paper, Patil et al. [4] have used ML algorithms such as K-Means clustering and Decision trees and were able to deduct that these algorithms can give fairly good predictions. Zou et al. [8] have used neural networks, decision trees and feature selection through Principal Component Analysis (PCA). They have found that PCA does not perform well with Diabetes prediction. However, they found that Random Forest to be most effective with an accuracy of 78 percent.

2.2. Diabetes Prediction using Perceptron

A perceptron is a simple ML algorithm which is frequently used to predict Diabetes. Mirshahvalad and Zanjani [3] have proposed Ensemble Perceptron Alrgorithm (EPA), where the basic perceptron model was combined with boosting algorithm and was tested on the datasets from National Health and Nutrition Examination Survey. It was observed that compared to the basic perceptron model, EPA

has improved ROC by 3%. Furthermore, Sivasankari et al. [5] have studied Application of Multilayered perceptron which can be used to handle complex data and non-linear relationships on PIMA dataset. The have concluded in their paper that MLP was able to out perform all the other algorithms used in their study with a good accuracy of 86% making it a good choice for predicting diabetes at an early stage.

3. Methodolody

The complete code script for this assignment is available on Github ,for review purposes, please refer to the link [6]

3.1. Data

The PIMA Indian Diabetes Dataset, which can be accessed through Kaggle, is a widely used dataset for ML research particularly for diabetes prediction. The dataset consists of 768 samples with 8 features which are health metrics related to diabetes prediction namely Pregnancies, Glucose, Blood_ Pressure, Skin_Thickness, Insulin, BMI, Diabetes_Pedigree_Function and age. Each sample has an outcome of 0 or 1 indicating whether the diabetes is present or not present. The dataset is smaller in size, well balanced, and hence it serves as a good source for evaluating performances of various ML algorithms.

3.2. Exploratory Data Analysis

The Exploratory Data Analysis of the dataset provided key insights of the data. Summary statistics have been used to understand various numerical statistics of the dataset. For instance, the mean value for Blood_Pressure is around 69 and the mean value for age is around 33.

The dataset has no null values. Different plots have been used to have a broader picture of distribution of data. The histograms 1 indicated that features like glucose, Blood_Pressure and BMI have nearly normal distribution and features like Insulin and Age are highly skewed. The box plots 3 show that most of the features have significant outliers.

Finally, the correlation heatmap 2 indicates that certain features like BMI, pregnancies and age have high correlation with the outcome variable indicating that they could be possibly important features for predicting the presence of diabetes.

3.3. Baseline Perceptron model

The perceptron is a type of supervised learning which is used to classify data into two classes on the basis of linear decision boundary. In the baseline model, a simple perceptron model is initialised with random_state of 16, along with other default parameters

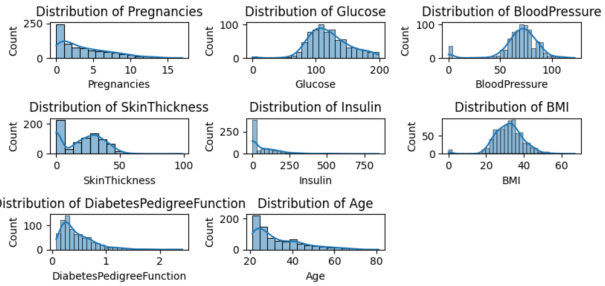


Figure 1. Histogram of Diabetes data

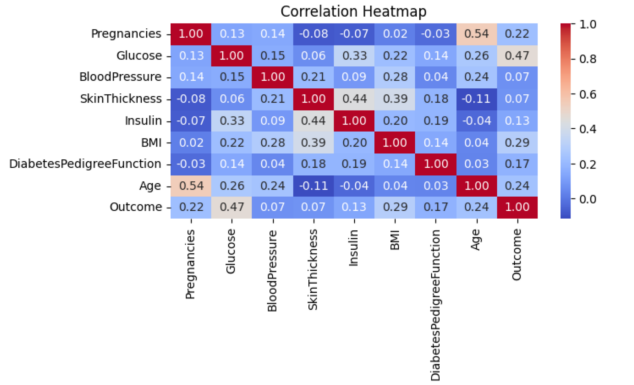


Figure 2. Correlation Heatmap of Diabetes data

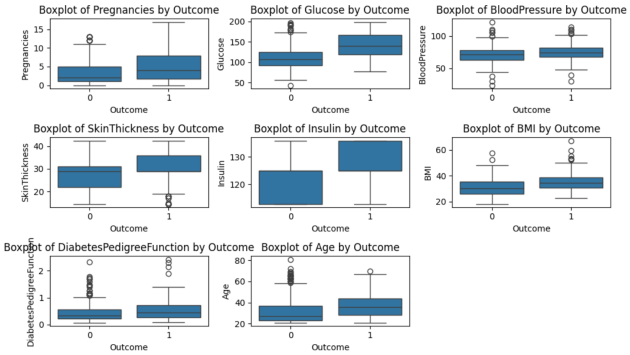


Figure 3. Boxplot of Diabetes data

3.4. Hyper parametric optimisation

Hyper parametric optimisation is an essential step in machine learning and is used in selecting the best combination of hyper parameters. In this method, a parameter grid is initialised to analyse various combinations of hyper parameters. The grid includes 'l2', 'elasticnet' options for penalty used for regularisation, various values of 'alpha', 'max_iter' and 'eta0' are explored. The 'tol' is set to 1e-4, 'early_stopping' and 'fit_intercept are set to 'True' and finally random_state is set to 32.

A balanced class distribution is ensured with the use of Stratifies K-Fold Cross Validation. We divide the data into 5 folds for cross validation by setting 'n_splits' to 5. The function 'GRidSearchCV' is used to perform an exhaustive search on the defined hyperparameter grid.

3.5. Ensemble Methods

3.5.1 Bagging

Bagging is an ensemble method which combines different models trained on different subsets of data to improve the performance of the model. In the code, a 'BaggingClassifier' is used to create an ensemble of 50 perceptron models. The 'max_samples' and 'max_features' are set to 0.8 and 1.0 respectively. To ensure that sampling is done with replacement, the 'Bootstrap' option is set to 'True'. Finally, 'random_state' is set to 16.

3.5.2 Boosting

Boosting is a ML technique used to improve performance of a model by combining different weak models. Similar to the Bagging technique used, an AdaboostClassifier is used to create an ensemble of 50 perceptron models. The 'learning_rate' is set to 1.0 and the 'random_state' is set to 16.Also, The 'SAMME' algorithm which is used to handle discrete classification is employed in this technique

3.6. Data Pre-Processing improvements and Feature Engineering

Pre-processing and feature engineering are very crucial steps in ML where the data is cleaned, transformed and made ready for the data analysis. Better pre-processing and feature engineering will lead to better performance of the model.

In this method, various predictors have zero values in them indicating erroneous or outlier values. These are intially replaced with NaN values and further the NaN values are imputed by the median. After that, the outliers in the data are identified and then capped using Inter-Quartile Range (IQR) method. Furthermore, the data is scaled by RobustScaler. After scaling, two degree polynomials are added to ensure that the non-linear relationships are well captured.

3.7. Feature Selection

3.7.1 Recursive Feature Elimination

Recursive Feature Elimination (RFE) is a type of feature selection where the less crucial features for the model are removed recursively and the model is built on the remaining features which helps in deciding the most important predictors for the model. In this model, the 5 most important features through RFE are selected which can be inferred from the line 'n_features_to_select=5' (refer github link)

3.7.2 K-Best Feature Selection

Similar to RFE, K-Best feature selection is a type of feature selection where K number of best features are selected based on scoring function which is the F-function in the paper. The parameter k decides the number of best features to be selected which set to 5 in the model.

3.8. Metrics

3.8.1 Accuracy

Accuracy is the ratio of correct predictions made out of all predictions. The formula for Accuracy is given as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

where TP=True Positive,
TN= True Negative,
FP=False Positive,
FN=False Negative

3.8.2 Precision

Precision is the ratio of number of correct positive predictions made to the number of all positive predictions made. The formula for preicision is given as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

3.8.3 Recall

Recall is the ratio of the number of correct positive predictions made to the number of actual positive instances. The formula for recall is given as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

3.8.4 F1-Score

F1- Score is the harmonic mean of precision and recall and is a good metric with the datasets where the classes are imbalanced. The formula for F1 Score is given as :

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

$$\text{F1-Score} = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{5}$$

4. Results and Evaluation

4.1. Learning Curves

The learning curve for baseline model 4 indicates a possibilty of over-fitting as there is a gap between training and validation accuracies. Also, the validation accuracy shows high fluctuation, suggesting that the baseline model is not consistent.

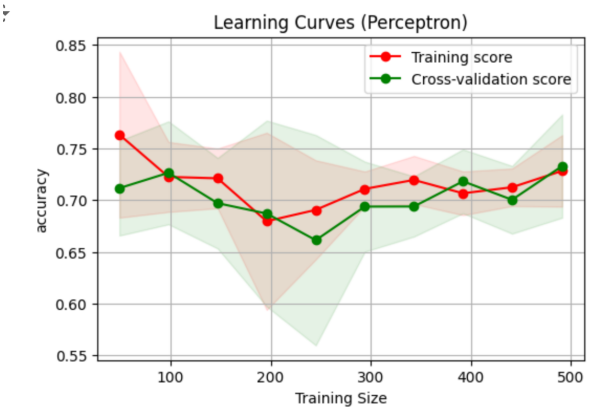


Figure 4. Learning curve for Baseline model

The learning curve for Hyper parametric optimisation 5 indicates that there is an improvement in the model compared to baseline model which is observed by stability in both training and validation accuracies.

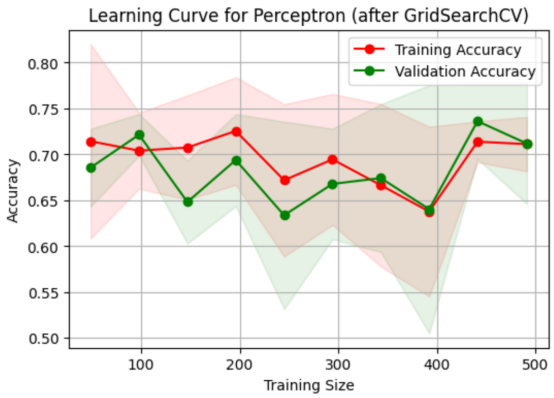


Figure 5. Learning curve for Hyper parametric optimisation model

From the learning curve of Bagging technique 6, it can be deduced that there is better stability. Even though there are gaps between training and validation accuracies, good training performance implies that Bagging technique is reliable as it reduces variance. Similar to Bagging, the learning curve of boosting technique 7 also indicates that both bias and variance can be reduced with the help of Boosting technique.

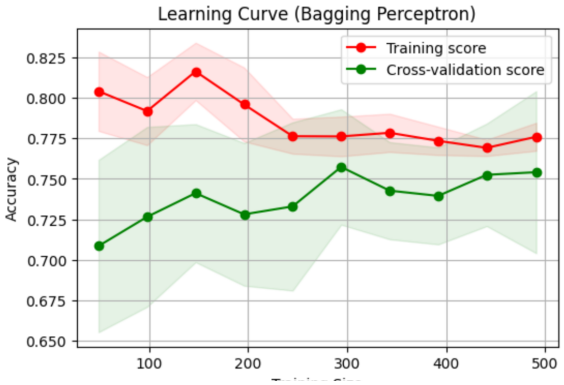


Figure 6. Learning curve for Bagging model

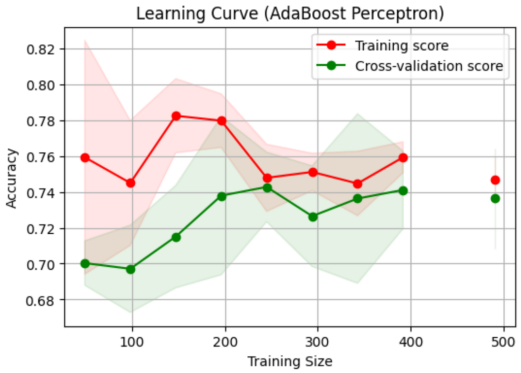


Figure 7. Learning curve for Boosting model

Since polynomial features are added during feature engineering to capture non-linear relationships, the learning curve for pre-processing improvements and feature engineering 8 indicates that the model can capture underlying patterns through its tighter convergence between scores of training and validation.

Finally, the curves for Feature selection models, both RFE 9 and Select K Best 10, imply reduced model complexity because of selection of important features. However, the learning curve for RFE is more stable while the learning curve for Select K best is fluctuating.

4.2. ROC Curves

The ROC Curve for baseline model 11 has an AUC of 0.69, implying that it's performance is moderate. The AUC for hyper parametric optimisation 12 is 0.85, indicating that it has very strong performance compared to the baseline model. The ROC Curve for Bagging 13 has an AUC of 0.81, implying it has better performance than baseline model but it is not as great as hyper paramteric optimisation. Boosting 14 has an AUC of 0.76, which is slightly less compared to Bagging in the ensemble methods.

The AUC obtained for feature engineering and pre-

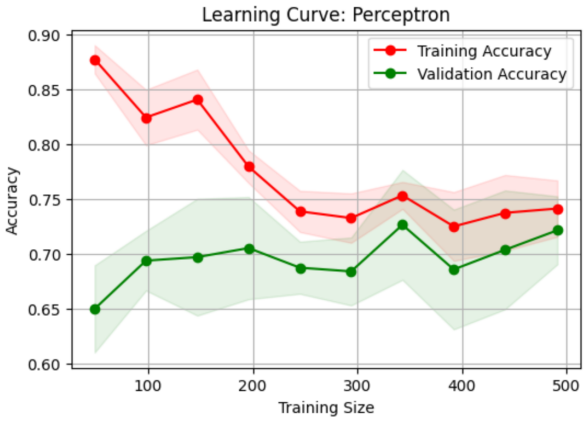


Figure 8. Learning curve for Pre-Processing improvements and Feature Engineering model

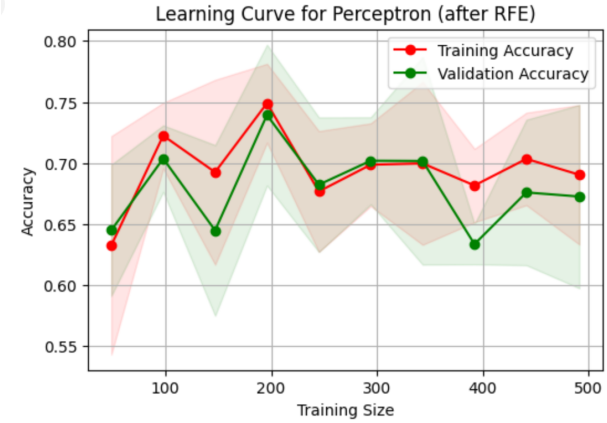


Figure 9. Learning curve Feature Selection (RFE) model

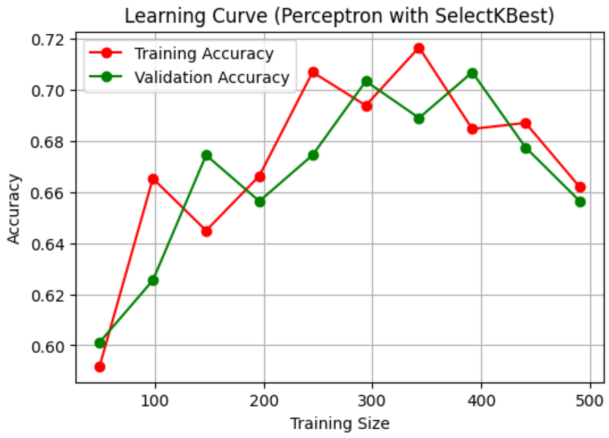


Figure 10. Learning curve Feature Selection (Select K Best) model

processing improvements 15 is same as boosting which is 0.76. The AUC obtained by performing Feature Selection (RFE) 16 is almost equal to the highest AUC obtained by hyper parametric optimisation. Finally, similar to Boosting and Feature engineering, the AUC obtained for Feature Selection (Select K Best) 17 is 0.76. Overall, by looking at ROC Curve and the AUC values obtained, hyper parametric optimisation seems to be the most effective model out of all other models closely followed by Feature Selection(RFE).

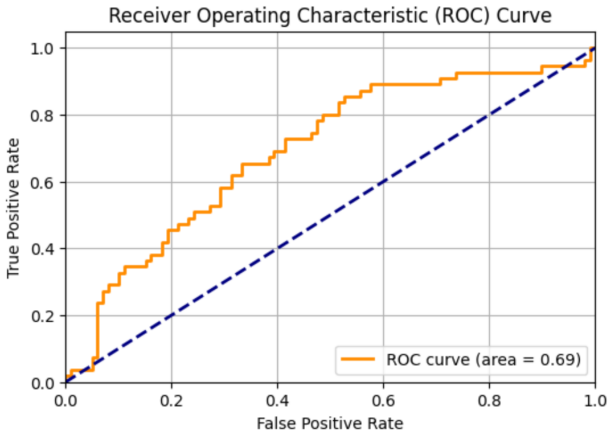


Figure 11. ROC curve for Baseline model

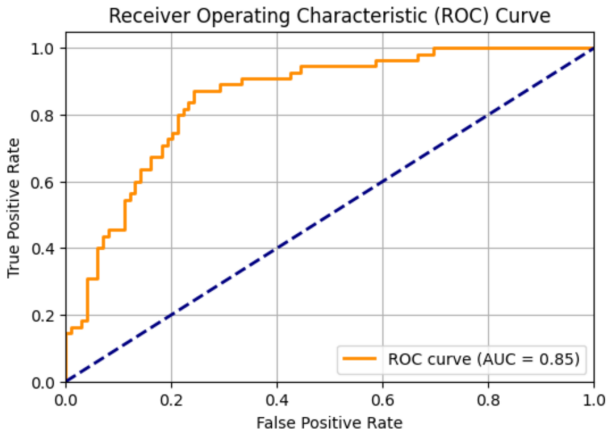


Figure 12. ROC curve for Hyper parametric optimisation model

4.3. Confusion matrix

The confusion matrix for the baseline model 18 indicates that there is a very high class imbalance with more correct predictions of negative class compared to the positive class. For the hyper parametric optimisation, the confusion matrix 19 illustrates better performance of the model in predicting positive classes. Furthermore, the confusion matrices for Bagging 20 and Boosting 21 techniques present similar numbers, but Bagging has more number of correct predictions. After feature engineering and pre-processing,

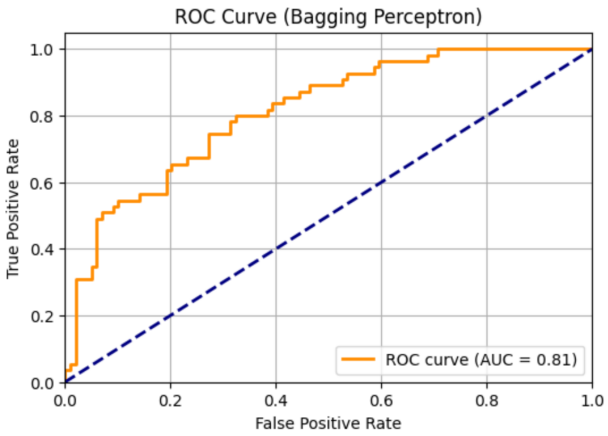


Figure 13. ROC curve for Bagging model

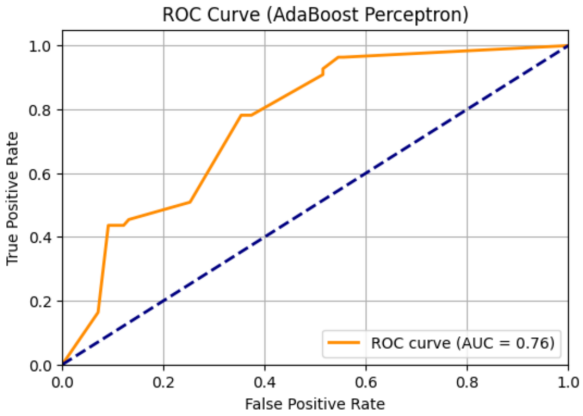


Figure 14. ROC curve for Boosting model

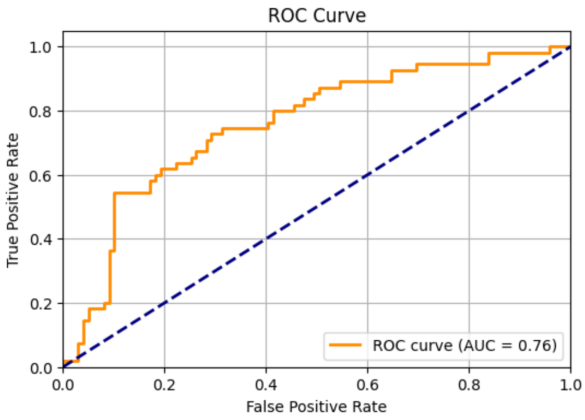


Figure 15. ROC curve for Pre-Processing improvements and Feature Engineering model

the confusion matrix 22 provided similar results to that of Boosting but an increase in the number of positive predictions. The confusion matrix for Feature Selection (RFE) 23

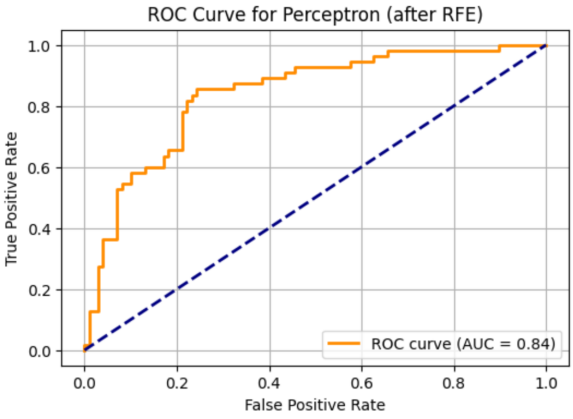


Figure 16. ROC curve Feature Selection (RFE) model

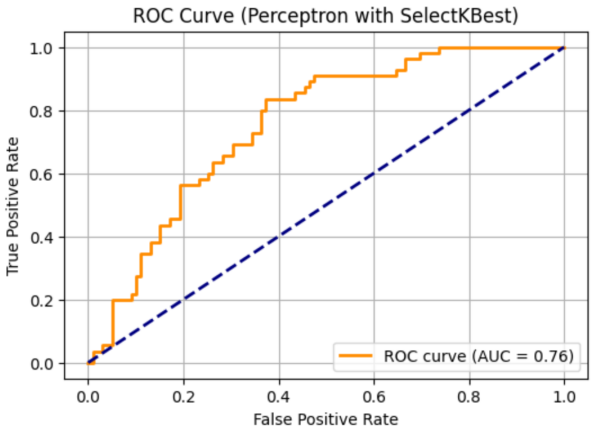


Figure 17. ROC curve Feature Selection (Select K Best) model

shows the highest number of correct negative predictions and also shows an marginal increase in the number of correct positive predictions. Finally, the Feature Selection (Select K Best) confusion matrix 24 displayed similar results to that of Boosting and feature engineering.

4.4. Classification report

The classification report for Baseline model 25 indicates that the accuracy for baseline model is 66% and the model performs better with classifying negative instances (Class 0). With Hyper paramteric optimisation 26, there is a good improvement in accuracy with 77% and also the precision, recall and F1- scores increase for positive class (Class 1). Very close to Hyper parametric optimisation is Bagging 27 with an accuracy of 76% and also shows similar improvements for recall, precision and F1-scores for Class 1. By using the Boosting technique 28, the model achieves 66% accuracy, but there is drop in precision, recall and F1-scores for both classes. Feature engineering and pre-processing improvements 29 yield an accuracy of 77% . The accuracy

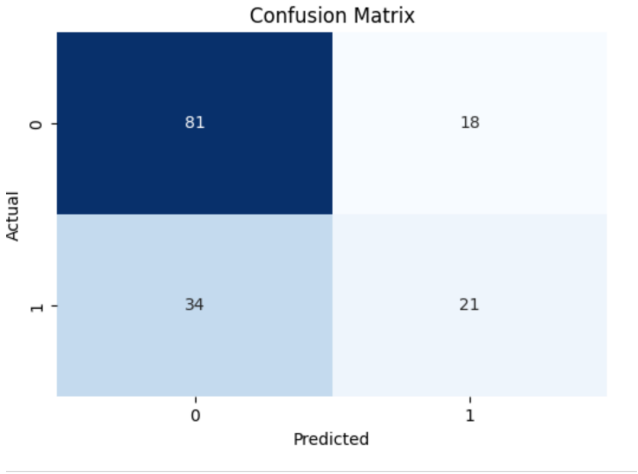


Figure 18. Confusion Matrix for Baseline model

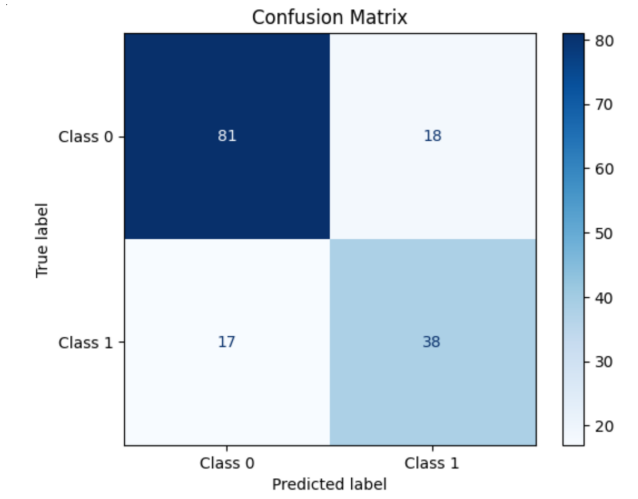


Figure 19. Confusion Matrix for Hyper parametric optimisation model

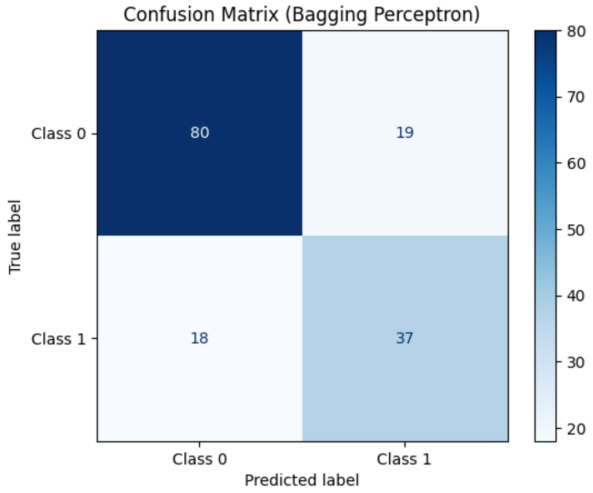


Figure 20. Confusion Matrix for Bagging model

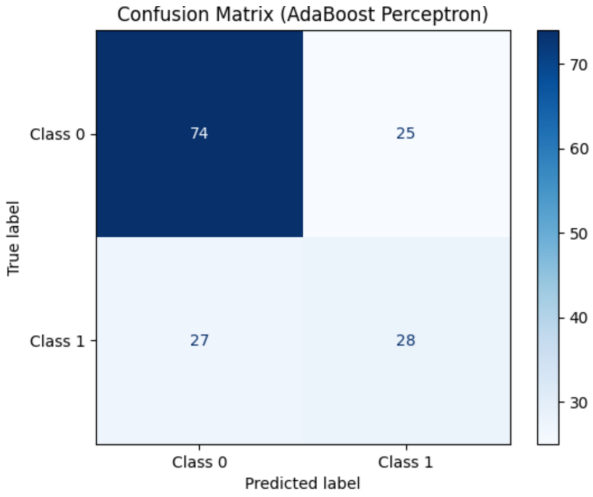


Figure 21. Confusion Matrix for Boosting model

for Feature Selection (RFE) 30 is 78%, which is highest among all other techniques and there is significant improvement in precision, recall, F1-values of both classes. However, by using Feature Selection (SelectKBest) 31 the accuracy is 69% and the precision, recall and F1 scores indicate a slight class imbalance.

5. Conclusion

Overall, among all the models used to improve the baseline model, Hyper parametric optimisation and Feature Selection (RFE) show better results.

If accuracy is considered, which is a general metric used to evaluate models, Hyper paramteric optimisation achieved 77% indicating significant improvement from Baseline model whose accuracy is 66%. Feature Selection (RFE) obtains slightly higher accuracy than hyper parametric optimisation at 78%. So if accuracy is considered as the metric, Feature Selection (RFE) performs best among all other models in predicting the diabetes.

Since medical data is being used, F1-score can be considered a good metric. If F1-score is considered for predictions related to Class 1, then hyper parametric optimisation performs well compared to Feature Selection (RFE) with an F1 score of 0.68 over 0.65.

Therefore, Hyper parametric optimisation and Feature Selection (RFE) with perceptron stand out as the best models with high accuracy and good F1-scores for predicting diabetes.

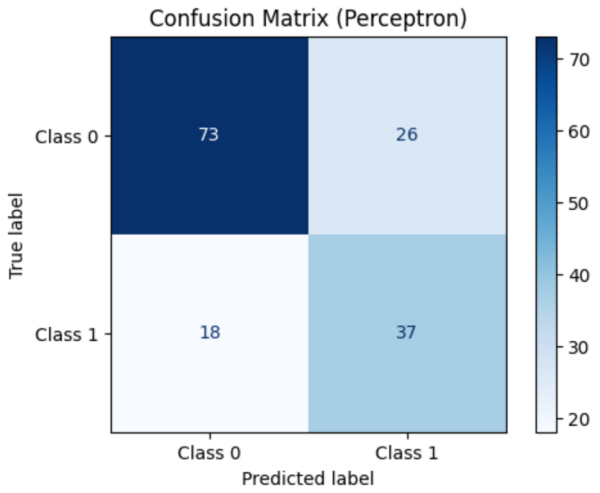


Figure 22. Confusion Matrix for Pre-Processing improvements and Feature Engineering model

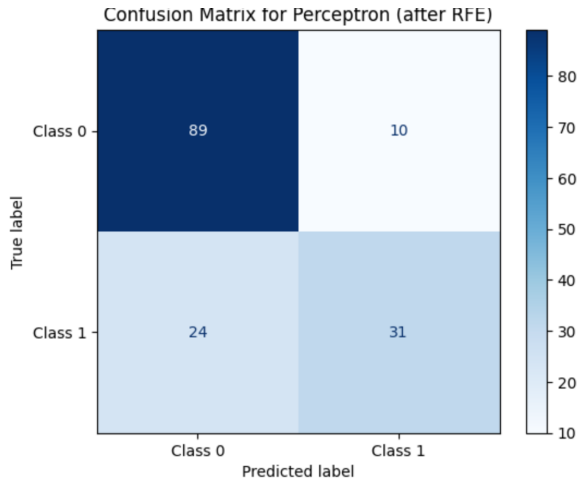


Figure 23. Confusion Matrix Feature Selection (RFE) model

References

[1] Mehtab Alam, Ihtiram Raza Khan, Mohammad Afshar Alam, Farheen Siddiqui, and Safdar Tanweer. The diabacare cloud: predicting diabetes using machine learning. *Acta Scientiarum. Technology*, 46(1), 2024. 1

[2] P Meenakshidevi, T R Logesh, G Navayugan, and M Sugesh Kannan. Efficient machine learning models for the accurate prediction of diabetes. In *2024 International Conference on Science Technology Engineering and Management (ICSTEM)*, pages 1–5. IEEE, 2024. 1

[3] Roxana Mirshahvalad and Nastaran Asadi Zanjani. Diabetes prediction using ensemble perceptron algorithm. In *2017 9th international conference on computational intelligence and communication networks (CICN)*, pages 190–194. IEEE, 2017. 1

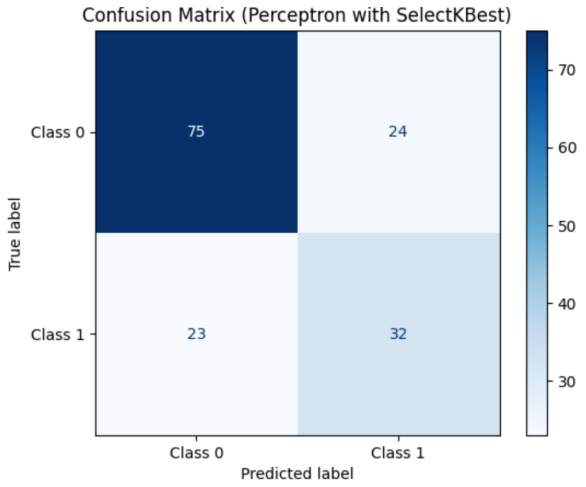


Figure 24. Confusion Matrix for Feature Selection (Select K Best) model

	precision	recall	f1-score	support
0	0.70	0.82	0.76	99
1	0.54	0.38	0.45	55
accuracy			0.66	154
macro avg	0.62	0.60	0.60	154
weighted avg	0.65	0.66	0.65	154

Classification Report:

Figure 25. Classification Report for Baseline model

	precision	recall	f1-score	support
0	0.83	0.82	0.82	99
1	0.68	0.69	0.68	55
accuracy			0.77	154
macro avg	0.75	0.75	0.75	154
weighted avg	0.77	0.77	0.77	154

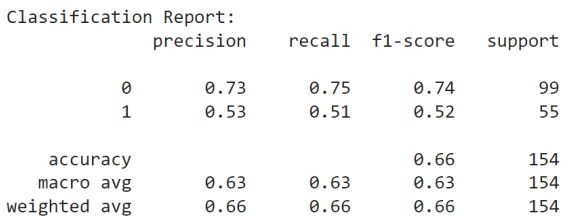
Classification Report:

Figure 26. Classification Report for Hyper parametric optimisation model

	precision	recall	f1-score	support
0	0.82	0.81	0.81	99
1	0.66	0.67	0.67	55
accuracy			0.76	154
macro avg	0.74	0.74	0.74	154
weighted avg	0.76	0.76	0.76	154

Classification Report:

Figure 27. Classification Report for Bagging model

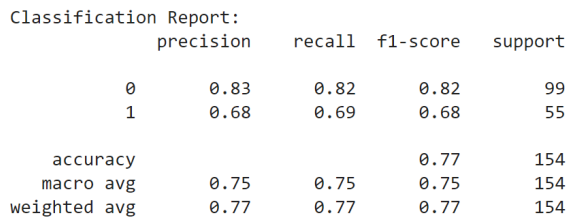


```
Classification Report:
              precision    recall  f1-score   support

           0       0.73      0.75      0.74        99
           1       0.53      0.51      0.52        55

    accuracy                           0.66       154
   macro avg       0.63      0.63      0.63       154
weighted avg       0.66      0.66      0.66       154
```

Figure 28. Classification Report for Boosting model

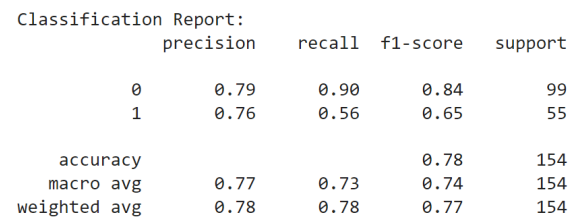


```
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.82      0.82        99
           1       0.68      0.69      0.68        55

    accuracy                           0.77       154
   macro avg       0.75      0.75      0.75       154
weighted avg       0.77      0.77      0.77       154
```

Figure 29. Classification Report for Pre-Processing improvements and Feature Engineering model

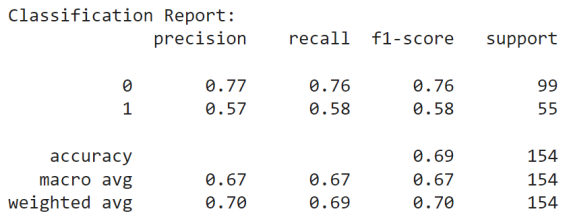


```
Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.90      0.84        99
           1       0.76      0.56      0.65        55

    accuracy                           0.78       154
   macro avg       0.77      0.73      0.74       154
weighted avg       0.78      0.78      0.77       154
```

Figure 30. Classification Report for Feature Selection (RFE) model



```
Classification Report:
              precision    recall  f1-score   support

           0       0.77      0.76      0.76        99
           1       0.57      0.58      0.58        55

    accuracy                           0.69       154
   macro avg       0.67      0.67      0.67       154
weighted avg       0.70      0.69      0.70       154
```

Figure 31. Classification Report for Feature Selection (Select K Best) model

[4] BM Patil, RC Joshi, and Durga Toshniwal. Association rule for classification of type-2 diabetic patients. In *2010 second international conference on machine learning and computing*, pages 330–334. IEEE, 2010. 1

[5] SS Sivasankari, J Surendiran, N Yuvaraj, M Ramkumar, CN Ravi, and RG Vidhya. Classification of diabetes using multilayer perceptron. In *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, pages 1–5. IEEE, 2022. 2

[6] Venkata Kamala Sindhoori Vadde. Deep learning assignment 1. https://github.com/Sindhu-Vadde/DeepLearning, 2024. Accessed: 2024-10-05. 2

[7] Nuzhat Ahmad Yatoo, Ishok Sathik Ali, and Imran Mirza. Comparing hyperparameter optimized support vector machine, multi-layer perceptron and bagging classifiers for diabetes mellitus prediction. *International Journal of Electrical & Computer Engineering (2088-8708)*, 14(5), 2024. 1

[8] Quan Zou, Kaiyang Qu, Yamei Luo, Dehui Yin, Ying Ju, and Hua Tang. Predicting diabetes mellitus with machine learning techniques. *Frontiers in genetics*, 9:515, 2018. 1