# Diabetes Identification Using Machine Learning(ML)

**Vika Li**[a], **Yixin(Crystal) Luo**[a], **and Vaishnavi Singh**[a]

Diabetes | Machine Learning | Big Data Analysis |


## 1. ABSTRACTION

This study addresses the critical challenge of diabetes prediction by leveraging machine learning models to analyze behavioral and health-related data. Our research utilizes a comprehensive dataset from the 2014 Behavioral Risk Factor Surveillance System (BRFSS), which includes 436,668 records post-oversampling to ensure class balance. We implement and compare five machine learning algorithms—Logistic Regression, Decision Tree, K-Nearest Neighbors (KNN), Gaussian Naive Bayes, and Multi-layer Perceptron (MLP)—to identify the most effective model based on the top five features determined through regression analysis. The MLP emerges as the superior model, demonstrating the highest accuracy and balanced performance across various metrics, including precision, recall, F1-score, and an ROC-AUC score of 0.7286. Our findings suggest that integrated computational approaches can significantly enhance the early detection and management of diabetes.


### Significance Statement

In the face of the diabetes epidemic, early detection is paramount for effective management and treatment. This research applies machine learning to identify crucial predictive factors of diabetes, harnessing a dataset representative of a broad U.S. population. By comparing five different algorithms, our study not only determines the most effective model for diabetes prediction but also underscores the potential of artificial intelligence in public health. The Multi-layer Perceptron (MLP) Classifier, recognized for its accuracy and comprehensive performance, stands out, marking a significant step toward integrating advanced analytics in healthcare.

## 2. INTRODUCTION

Diabetes is one of the most common endocrine diseases, with more than 133 million Americans currently living with diabetes or in the pre-diabetic stage (CDC, 2022). While there are many genetic factors that can influence one's diagnosis of diabetes, the surge in diabetes cases is inextricably linked to contemporary human lifestyles (Hu, 2021).

Excessive calorie intake and lack of physical activity, coupled with smoking and alcohol abuse, have largely contributed to the rising incidence of this metabolic disease (Hu, 2021; Magkos et al., 2009). In other words, with proper health management, diabetes can be controlled in an ideal state (Ganie et al., 2022).

In order to better understand what factors predict diabetes, machine learning (ML) techniques have been harnessed for the diagnosis of diabetes, which is famous for its efficient and predictive nature (Dewangan Agrawal, 2015; Ganie et al., 2022). Utilizing advanced computational algorithms, these ML approaches facilitate early and accurate detection, providing valuable insights that enable healthcare professionals to effectively manage and treat diabetes as well as advice on how to reverse prediabetes. As society grapples with the multifaceted impact of lifestyle choices on health, integrated technologies such as ML present a promising avenue to address and reduce the growing burden of diabetes.

Therefore, in this data science report, we seek to understand how to use machine learning for diabetes identification. Our research question is: How do different machine learning models compare in their ability to accurately predict outcomes based on a subset of five key features within the diabete dataset?

## 3. Dataset Description and Workflow

**Data source.** The dataset for our research, retrieved from the UCI Machine Learning Repository, comprises 253,680 records with 20 features, after applying oversampling to balance the data, the dataset for our research now comprises 436,668 records with 21 features, an increase from the original 253,680 records These features cover a range of categories, including demographics (race, sex), personal information (income, education), and health history (drinking, smoking, mental health, physical health).

Author affiliations: [a]Georgetown University

| | Name | Role |
|---|---|---|
| 0 | Diabetes_binary | Target |
| 1 | HighBP | Feature |
| 2 | HighChol | Feature |
| 3 | CholCheck | Feature |
| 4 | BMI | Feature |
| 5 | Smoker | Feature |
| 6 | Stroke | Feature |
| 7 | HeartDiseaseorAttack | Feature |
| 8 | PhysActivity | Feature |
| 9 | Fruits | Feature |
| 10 | Veggies | Feature |
| 11 | HvyAlcoholConsump | Feature |
| 12 | AnyHealthcare | Feature |
| 13 | NoDocbcCost | Feature |
| 14 | GenHlth | Feature |
| 15 | MentHlth | Feature |
| 16 | PhysHlth | Feature |
| 17 | DiffWalk | Feature |
| 18 | Sex | Feature |
| 19 | Age | Feature |
| 20 | Education | Feature |
| 21 | Income | Feature |

**Fig. 1.** Features and Target

Notably, this dataset originates from the 2014 Behavioral Risk Factor Surveillance System (BRFSS), which has undergone significant changes in 2011, particularly in the weighting methodology (raking) and the inclusion of cell phone only respondents. The aggregated BRFSS data, combining both landline and cell phone responses, encompasses data for all 50 states, the District of Columbia, Guam, and Puerto Rico, providing a comprehensive view of the subjects' socioeconomic status and health-related behaviors. This diverse dataset offers a rich foundation for our data analysis.

**Unit of observations.** The primary unit of observation for the BRFSS is individuals residing in the U.S. The data is collected through monthly telephone interviews and includes both landline and cellular telephone subscribers.

**Methods.** This data flow diagram outlines the process of a machine learning workflow for our diabetes dataset. Initially, the data undergoes preprocessing to clean and prepare it for analysis. Then, feature extraction is performed, identifying 21 perceptive features which are then narrowed down to the top 5 most significant features for model training.

These top features are used as inputs into several basic machine learning classifiers: Logistic Regression, Decision Tree, KNN (k-nearest neighbors), Naive Bayes, and Multiple Layer Perceptron. Each classifier will be trained to predict outcomes based on the processed diabetes data.

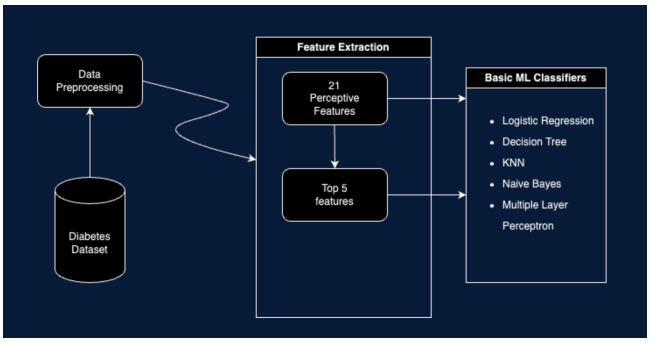



**Fig. 2.** Workflow Diagram

## 4. Feature Engineering and Exploratory Data Analysis (EDA)

**Missing data.** We find the missing data using following function - print(df.isnull().sum()) From the given results, it is clear that we do not have missing data in this data set.

```
HighBP                  0
HighChol                0
CholCheck               0
BMI                     0
Smoker                  0
Stroke                  0
HeartDiseaseorAttack    0
PhysActivity            0
Fruits                  0
Veggies                 0
HvyAlcoholConsump       0
AnyHealthcare           0
NoDocbcCost             0
GenHlth                 0
MentHlth                0
PhysHlth                0
DiffWalk                0
Sex                     0
Age                     0
Education               0
Income                  0
Diabetes_binary         0
dtype: int64
```

**Fig. 3.** Missing Values Check

**Correlation matrix of the data set.** As shown in the figure below, the correlation matrix of our dataset reveals the interrelationships between the 20 feature variables. Notably, there is a strong positive correlation between "self-described general health" and "physical health", implying that those who are optimistic about their general health are also likely to report good physical health. We also observe a negative correlation between general health and income, suggesting that improvements in general health may accompany declines in reported income.

**Class Balance.** Given the substantial imbalance between diabetes and nondiabetic values, we've employed oversampling as a strategy to address this issue. The process starts by segregating the majority and minority classes. Subsequently, we upsample the minority class to align with the size of the majority class. The final step involves concatenating the upsampled minority class with the majority class. The result is a dataset with 218,334 instances of non-diabetic and

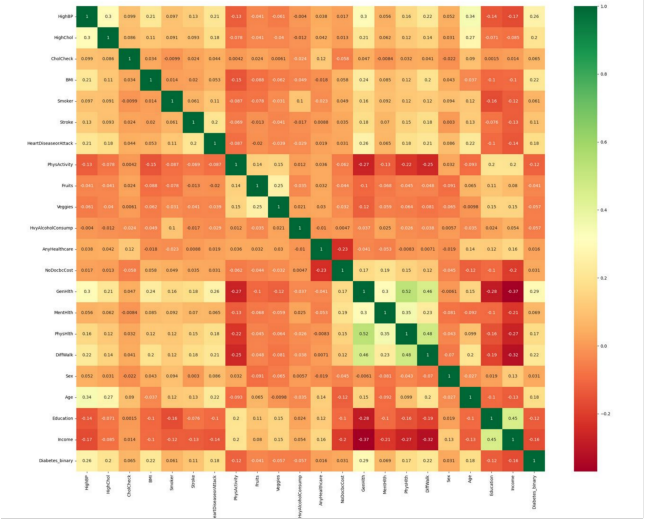


**Fig. 4.** Correlation Heatmap

218,334 instances of diabetic values, offering a more balanced representation of the two classes.

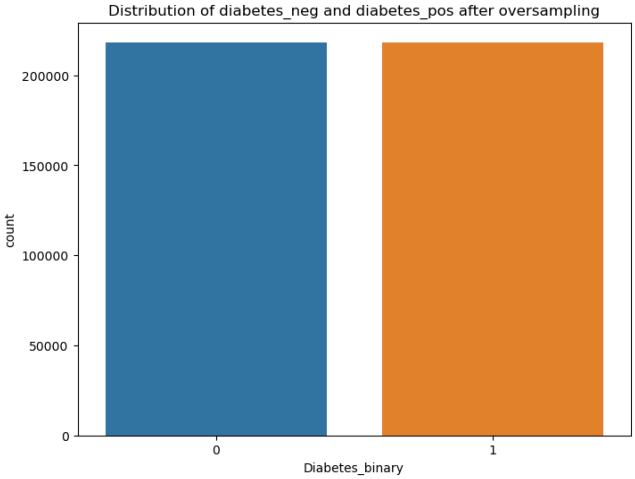



**Fig. 5.** Instances of Class attribute

## 5. Results

**A. Regression Analysis.** The dummy feature "Diabetes_binary" will be used as the target feature for the regression analysis. We performed the regression using the OLS model followed by t-test analysis, F-test analysis, and confidence interval analysis, and based on the t-test score and the confidence intervals values, we picked our top 5 features (BMI, Education, Age, Income, GenHlth).

**B. Classification Analysis.**

***B.1. Logistic Regression.*** A Logistic Regression model was trained to predict the diabetes binary outcome based on a subset of the top 5 features from the dataset. The model's maximum iteration parameter was set to 1000 to ensure convergence.

OLS Regression Results

| | | | |
|---|---|---|---|
| Dep. Variable: | Diabetes_binary | R-squared: | 0.257 |
| Model: | OLS | Adj. R-squared: | 0.257 |
| Method: | Least Squares | F-statistic: | 2.717e+04 |
| Date: | Tue, 12 Dec 2023 | Prob (F-statistic): | 0.00 |
| Time: | 18:39:30 | Log-Likelihood: | −2.2690e+05 |
| No. Observations: | 393001 | AIC: | 4.538e+05 |
| Df Residuals: | 392995 | BIC: | 4.539e+05 |
| Df Model: | 5 | | |
| Covariance Type: | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| const | −0.5518 | 0.006 | −95.324 | 0.000 | −0.563 | −0.540 |
| BMI | 0.0149 | 0.000 | 148.628 | 0.000 | 0.015 | 0.015 |
| Education | −0.0096 | 0.001 | −12.612 | 0.000 | −0.011 | −0.008 |
| Age | 0.0402 | 0.000 | 163.651 | 0.000 | 0.040 | 0.041 |
| Income | −0.0108 | 0.000 | −28.936 | 0.000 | −0.012 | −0.010 |
| GenHlth | 0.1307 | 0.001 | 186.554 | 0.000 | 0.129 | 0.132 |

| | | | |
|---|---|---|---|
| Omnibus: | 121897.496 | Durbin-Watson: | 2.006 |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 19109.722 |
| Skew: | −0.123 | Prob(JB): | 0.00 |
| Kurtosis: | 1.948 | Cond. No. | 277. |

**Fig. 6.** OLS report for the top 5 features

The accuracy of the model, which represents the proportion of true results (both true positives and true negatives) among the total number of cases examined, was found to be approximately 72.49%. This suggests that the model is reasonably effective at classifying the instances correctly, although there is room for improvement.

***Model Evaluation*** The Mean Squared Error (MSE), which provides a measure of the quality of an estimator—it is always non-negative, and values closer to zero are better—was calculated to be 0.2561. The MSE value here indicates that on average, the square difference between the predicted values and the actual values is about 0.2561, suggesting that the model predictions are moderately close to the actual values.

***Confusion Matrix*** The confusion matrix for the model presented the following results. See the figure for Logisitic Regression.

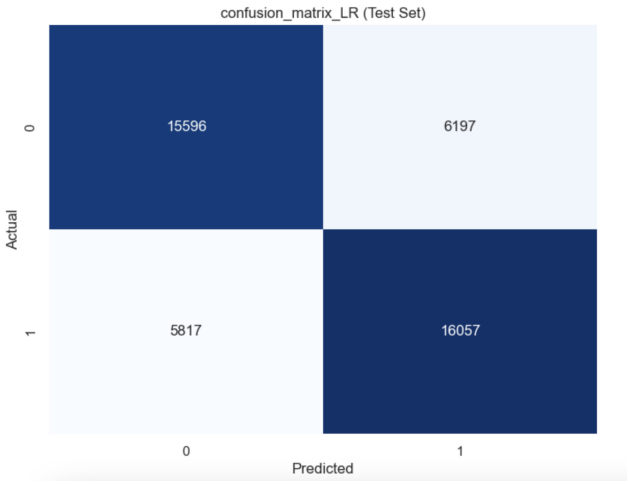



**Fig. 7.** Confusion Matrix for the Logistic Regression Model

From these values, we can infer that the model is more effective at identifying true positives than true negatives. However, there is a significant number of both false positives and false negatives, indicating potential areas of model improvement.

***ROC Curve*** The Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) were used

to evaluate the model's discrimination capacity. The ROC curve (attached as a separate figure) demonstrates the trade-off between the true positive rate and the false positive rate. The AUC for our model was calculated to be 0.80. An AUC of 0.80 indicates a good discriminative ability and suggests that the model has a high chance of distinguishing between the positive class and the negative class.

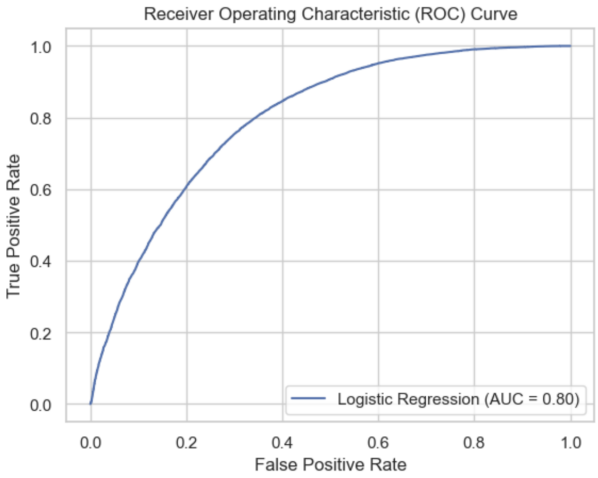



**Fig. 8.** ROC Curve for the Logistic Regression Model

In conclusion, while the Logistic Regression model demonstrates good predictive ability as indicated by the AUC of 0.80, there is room for improvement in terms of accuracy and reducing the number of false predictions. Further refinement of the model could involve feature engineering, hyperparameter tuning, or trying different algorithms to improve the performance.

***B.2. Decision Tree.*** We employed a comprehensive approach that involved pre-pruning and post-pruning techniques to enhance the model's performance and generalization ability.

***Pre-Pruning*** Pre-pruning was performed using GridSearchCV to find the optimal combination of hyperparameters. Our parameter grid included various criteria, depths, minimum samples for leaf nodes and splits, and feature counts. The process aimed to prevent the decision tree from overfitting by limiting its growth during training. The optimal parameters obtained through pre-pruning were as follows: Criterion: entropy Max Depth: 6 Max Features: None Min Samples Leaf: 3 Min Samples Split: 2 With these parameters, the pre-pruned model achieved an accuracy score of 0.7285. The Receiver Operating Characteristic (ROC) curve yielded an Area Under the Curve (AUC) of 0.80, indicating a good predictive performance.

***Post Pruning*** Post-pruning was conducted after the decision tree was fully grown. It involved reducing the complexity of the tree by adjusting the regularization parameter , which simplifies the model and helps in avoiding overfitting. The goal was to find an that retains model accuracy while simplifying its structure. The post-pruning process identified the best value as 0.0, suggesting that no additional regularization was necessary. The post-pruned model showcased an improved accuracy of 0.7798 on the test set. The corresponding ROC curve demonstrated an AUC of

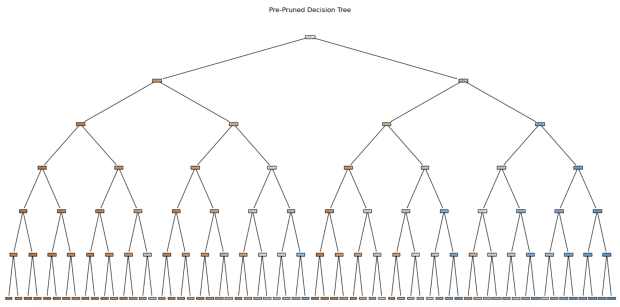



**Fig. 9.** Prepruned Decision Tree

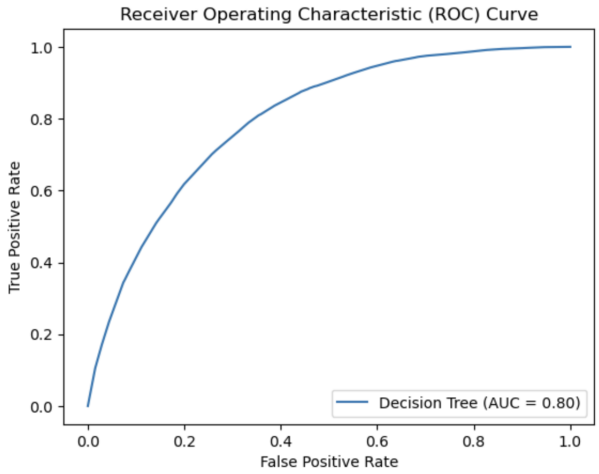



**Fig. 10.** ROC Curve for the Pre Decision Tree Model

0.86, indicating a superior ability to distinguish between the classes compared to the pre-pruned model.

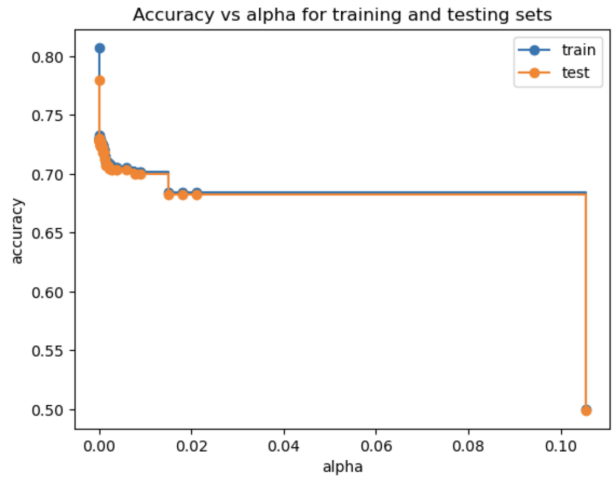
The increase in AUC from 0.80 to 0.86 through post-pruning indicates that the model's discriminatory capacity was enhanced while maintaining a low false-positive rate. The post-pruned model not only outperformed the pre-pruned model in terms of accuracy but also showed a better balance between sensitivity and specificity.

***B.3. K-Nearest Neighbors (KNN).*** The KNN model was implemented with `n_neighbors` set to 5, which defines the number of nearest neighbors to consider for making predictions. The model was trained on a subset of features deemed most significant, referred to as `top_5_features`.

The KNN model achieved an accuracy of 0.7442 on the test dataset, indicating a strong predictive capability. The MSE for the test set was calculated to be 0.2557, which is relatively low. See figure 12 for the matrix.

The ROC curve for the KNN model has an AUC of 0.82. This value is a measure of the model's ability to discriminate between the positive and negative classes. An AUC of 0.82 indicates a good classification performance.

***B.4. Gaussian Naive Bayes.*** The Gaussian NB model achieved a test accuracy of 0.7111, indicating a moderate level of predictive performance. The mean squared error (MSE) for the test set was calculated as 0.2888, which is higher compared




```
Best alpha for post-pruning: 0.0
Accuracy of the best post-pruned tree: 0.7798108411386173
```

**Fig. 11.** Accuracy vs Alpha Decision Tree

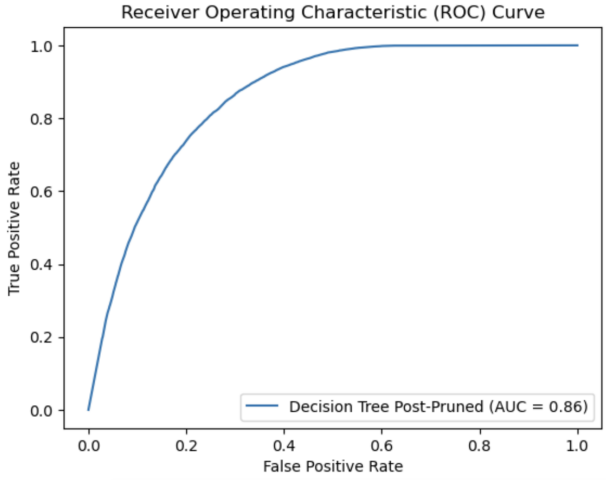



**Fig. 12.** ROC Curve for the Pre Decision Tree Model

to more complex models, signifying potential misclassification errors.

The ROC curve for the Gaussian NB model exhibited an AUC of 0.78. Although this indicates an acceptable ability to discriminate between the positive and negative classes, it is less than the AUCs achieved by the other models. And here is the confusion matrix:

***B.5. Neural Network: Multi-layer Perceptron Classifier(MLPClassifier).*** MLPs consist of multiple layers of neurons, allowing them to learn and represent complex, non-linear relationships in data The training process includes feedforward architecture, and backpropagation (implicit in scikit-learn) is used during the training process to update the weights of the neural network based on the error between predicted and actual values.

- Activation Functions:

Hidden Layers: ReLU (Rectified Linear Unit), tanh (hyperbolic tangent function) Output Layer (Binary Classification): Logistic (Sigmoid)

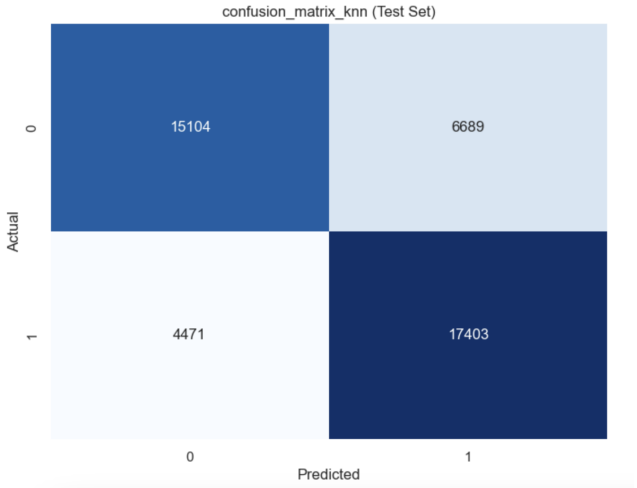



**Fig. 13.** Confusion Matrix for the K-Nearest Neighbors Model

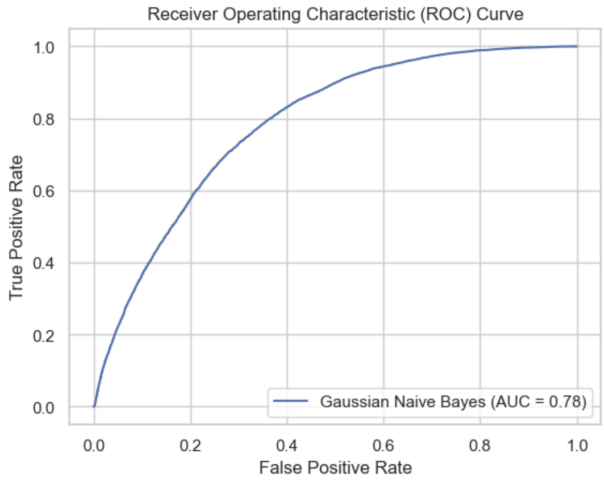



**Fig. 14.** ROC Curve for the K-Nearest Neighbors Model

Based on our MLP model we got the following hypermaters as best suited for the model :

Best Hyperparameters: 'activation': 'relu', 'alpha': 0.001, $'hidden_layer_sizes' : (60, 40), 'learning_rate' : 'constant', 'solver' : 'adam'$

Moreover, the test accuracy estimated by our model is : 0.7286280257402615 and the confusion matrix estimated for our model is provided in the following figure.

The confusion matrix for the MLP classifier can help in evaluating its performance by providing insights into the model's ability to correctly classify instances. It helps in assessing true positives, true negatives, false positives, and false negatives, which are essential for understanding the model's predictive accuracy and error rates.

On Analyzing the true positives, true negatives, false positives, and false negatives in the confusion matrix for the MLP classifier we got several insights: True Positives (TP) of 15,964 indicate the number of instances where the model correctly identified the positive class. True Negatives (TN) of 15,853 show the number of instances where the model correctly identified the negative class. These figures suggest that the model is comparably reliable in identifying both

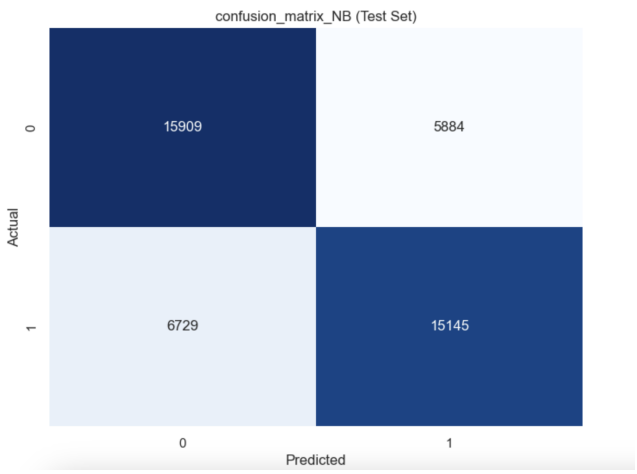



**Fig. 15.** Confusion Matrix for the Gaussian Naive Bayes Model

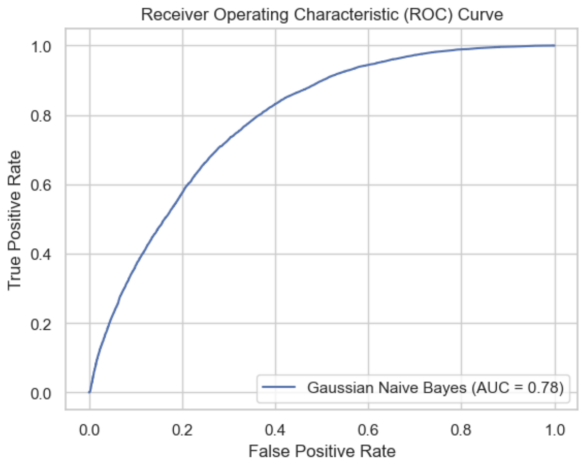



**Fig. 16.** ROC Curve for the Gaussian Naive Bayes Model

positive and negative classes. False Positives (FP) of 6,021 represent instances where the model incorrectly predicted the positive class. False Negatives (FN) of 5,829 are instances that were incorrectly predicted as the negative class. The quantities of false positives and false negatives suggest areas for potential model improvement, as reducing these will improve overall accuracy.

The above ROC AUC score of 0.81 for the MLP classifier indicates strong performance in discriminating between classes, which is essential for diabetes detection. It suggests that the model has a commendable ability to distinguish between positive and negative cases, making it a promising model.

**C. K-fold Cross Validation of the Models:.** In a comparative study of five classification models for diabetes diagnosis, the models are evaluated across several metrics: accuracy, confusion matrix, precision, recall, F1-score, ROC-AUC score, and cross-validation (CV) scores using k-fold validation as presented in the table below:

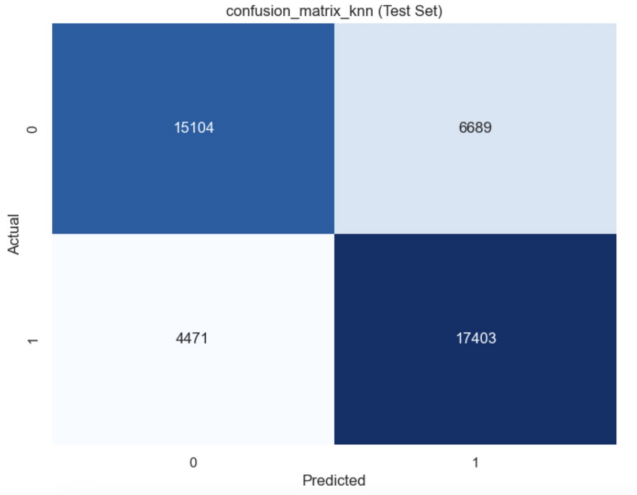



**Fig. 17.** Confusion Matrix for the MLPClassifier

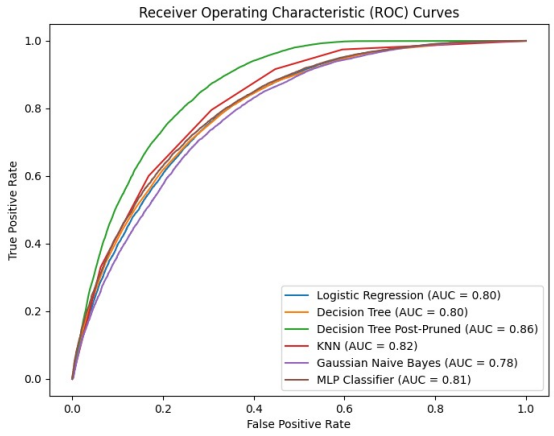



**Fig. 18.** ROC Curve for MLPClassifier

## 6. Limitations

The study presented valuable insights but also faced certain limitations:

**Unbalanced data**: Despite efforts to use oversampling to balance the dataset, synthetic oversampling methods may introduce bias or overfitting. The oversampling technique does not take into account the potential complexity and underlying patterns of minorities. **Feature Selection**: The first 5 features were selected based on the provided dataset. However, other relevant features may not have been included in the study that could have improved the predictive performance of the model. **Interpretability of the model**: Although MLP classifiers show good results,

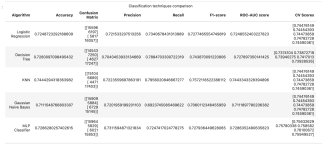


**Fig. 19.** K-fold Cross Validation Table

neural networks are often viewed as "black boxes" with limited interpretability, which can be a significant drawback in clinical settings where understanding the decision-making process is critical. **Generalizability**: models are trained and tested on BRFSS-specific datasets. Without further validation, results may not generalize to other populations or datasets. **Algorithmic Complexity**: Some algorithms, particularly MLP classifiers, involve complex structures that may not be required for all types of prediction tasks and may be computationally expensive for large-scale applications.

## 7. Future Work

To address these limitations and enhance the robustness of our findings, we propose the following future work: Alternative resampling techniques: Experiment with different resampling techniques. Cross-population validation: validate models on different populations and datasets to ensure generalizability and applicability of models to different populations. Hybrid Models: Develop hybrid models that combine the strengths of various algorithms, including MLP, to improve performance while balancing complexity and interpretability.

## 8. Insights and Conclusion based on the cross validation of the models

Upon evaluating the cross-validation scores, which are critical for assessing the stability and generalization ability of the models across different subsets of the dataset, the **MLP (Multilayer Perceptron) classifier** emerges as the superior model for diabetes detection. The MLP has demonstrated the highest mean CV score, with a range between 75.63% and 76.10%, with an approximate mean of 75.86%. This suggests that the MLP is likely to maintain its performance across various unseen data points and is less prone to overfitting compared to the

Prioritizing CV scores over other individual performance metrics is advantageous because CV scores provide insights into how well a model generalizes beyond the specific data it was trained on. Cross-validation involves partitioning the data into subsets, training the model on some subsets, and validating it on the remaining subsets. This process is repeated multiple times to ensure a more reliable estimate of the model's predictive performance. The high CV scores of the MLP model indicate that its performance is robust across different folds of the dataset, which is important for medical diagnostic models that will be applied to real-world scenarios where the data may vary. Such robustness is essential in the medical field, as it implies that the model will consistently perform well in practice, making it the preferred choice for diabetes detection based on the CV score criterion.

## 9. References


Amit Kumar Dewangan, Agrawal, P. (2015). Classification of Diabetes Mellitus Using Machine Learning Techniques. International

Journal of Engineering and Applied Sciences, 2(5), 257905. CDC diabetes health indicators. UCI Machine Learning Repository. (n.d.). https://archive.ics.uci.edu/dataset/891/cdc+diabetes+health+indicators

CDC. (2019). Diabetes basics. Centers for Disease Control and Prevention. https://www.cdc.gov/diabetes/basics/index.html

Hu, F. B. (2011). Globalization of diabetes. Diabetes Care, 34(6), 1249–1257. https://doi.org/10.2337/dc11-0442

Lai, H., Huang, H., Keshavjee, K., Guergachi, A., Gao, X. (2019). Predictive models for diabetes mellitus using machine learning techniques. BMC Endocrine Disorders, 19(1). https://doi.org/10.1186/s12902-019-0436-6

Teboul, A. (2021, November 8). Diabetes health indicators dataset. Kaggle. https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset