Diabetes Prediction Analysis and Web Application

A report on

Diabetes Prediction Analysis and Web Application

[CSE- 3183]

Submitted By

Anurag Kasat 210962180
T Praneeth 210962162

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MANIPAL INSTITUTE OF TECHNOLOGY,
MANIPAL ACADEMY OF HIGHER EDUCATION

NOV 2023

Diabetes Prediction Analysis and Web Application

Anurag Kasat[1], T Praneeth[2]

[1]CSE AIML MIT Manipal, India

[2]CSE AIML MIT Manipal, India

[1]anuragkasat1606@gmail.com;
[2]praneeththiragabathina123@gmail.com ;

***Abstract*—This study explores the use of machine learning algorithms to predict diabetes risk based on a diverse dataset. Various models are evaluated for accuracy, sensitivity, and specificity, revealing promising results for early detection and creating a web application to take input from the users and give a prediction based on the input parameters. The research contributes to the advancement of personalized healthcare by harnessing machine learning for proactive diabetes risk assessment.**

Keywords—Diabetes, machine learning prediction, healthcare algorithm, risk assessment dataset, feature selection, early detection, model evaluation, chronic disease preventive healthcare, personalized medicine, neural networks, support vector machines, naïve bayes

I. Introduction

The fusion of technology and healthcare in recent years has brought forth revolutionary methods for managing and preventing illness. Diabetes stands out among chronic illnesses as a global health problem, impacting millions of people globally. The key to reducing the effects of diabetes and enhancing patient outcomes is early identification and proactive care. A viable way to accomplish these goals is to combine the strength of predictive analytics with user-friendly online apps.

Diabetes prediction analysis uses sophisticated data analytics methods to estimate a person's risk of getting the disease. Predictive models study a wide range of variables, from genetic predispositions to lifestyle decisions and medical history, and they can offer important insights into an individual's risk profile.

II. Literature Review

According to [1], The focus is on predicting diabetes risk by establishing a correlation with daily lifestyle activities, including eating and sleeping habits, physical activity, and indicators like BMI and waist circumference. The research employs statistical methods, such as the Chi-Squared Test of Independence, and machine learning algorithms like CART to analyze the relationship between lifestyle factors and diabetes risk. Additionally, the study evaluates the performance of various classifiers, including Naive Bayes, RBF network, and SVM, indicating that SVM demonstrates superior accuracy in classifying heart, cancer, and diabetes datasets. The research, conducted in the WEKA environment, concludes that SVM is a robust and effective classifier for medical datasets.

In [2] The study employed the k-fold cross-validation method, and subsequently applied the CART method for validation. The dataset was divided into 5 folds (k=5), with each fold serving as a test set iteratively and as a training set for the remaining iterations. This approach aimed to enhance the accuracy of the model and produce unbiased results. Following a thorough assessment of accuracy and bias, the study generated a CART plot and confusion matrix [11], revealing an overall accuracy of 75%. This indicates that the built prediction model is correct 75% of the time, with a 25% chance of error. In essence, the research successfully constructed a predictive model capable of determining whether an individual is likely to develop diabetes based on their daily lifestyle activities, achieving a 75% accuracy rate.

The methodology employed in [3] is the Cross Industry Standard Process for Data Mining, emphasizing significant time allocation to business and data understanding, with the remaining time devoted to model building, assessment, and deployment. The constructed models, such as Decision Trees and Regression models, aim to predict the binary target variable. Following data manipulation and the transformation of specific variables, decision tree and regression models are developed. Multiple models with diverse properties, including backward regression, forward regression, stepwise regression, and decision trees with entropy, are built. Decision trees are also constructed with varying decision node properties based on the misclassification rate as the assessment measure. Subsequently, a variety of models are run, and a model comparison node is utilized to identify the best model for predicting the binary target variable. The selection of the optimal model is based on the valid misclassification rate.

The methodology proposed in paper [4] comprises three main steps. Initially, in step 1, the diabetes dataset is loaded into RStudio for pre-processing purposes. The loaded dataset undergoes further pre-processing using a 10-fold cross-validation method, and this process is iterated three times, following a standard procedure for comparing various models. Subsequently, the preprocessed data is randomly split into training and test sets, with an 80:20 ratio, a commonly employed practice in literature. Multiple machine learning algorithms, including RF, LDA, CART, and k-NN, are then employed to discern data patterns and train the models for predictions. The subsequent step involves testing these models with the test dataset. Finally, an analysis is conducted, focusing on accuracy and kappa metrics.

[5] employs classification techniques on various datasets to determine the presence or absence of diabetes in individuals. The dataset for diabetic patients is compiled by collecting information from a hospital warehouse, comprising two hundred instances with nine attributes. These instances are

categorized into two groups, specifically blood tests and urine tests. The study utilizes WEKA for implementing the classification of the data, and the data's performance is evaluated through a 10-fold cross-validation approach, known for its effectiveness on small datasets. The results obtained from different classifiers, namely Naïve Bayes, J48, REP Tree, and Random Tree, are compared. The findings suggest that J48 exhibits the highest accuracy at 60.2% compared to the other classifiers.

[6] uses six machine learning methods to forecast diabetes. These six algorithms are Random Forest (RF), Support Vect or Machine (SVM), Decision Tree (DT), Logistic Regressio n (LR), KNearest Neighbours (KNN), Naive Bayes (NB), an d KNN.Key integrated tools for application development, ite rative data analysis, and data visualisation are provided by E nthought Canopy, a validated scientific and analytical Pytho n package distribution following the acquisition of the datas et using the UCI machine learning repository.The data under went data preprocessing in the initial stage.Data must be org anised in order for analysis and performance to be effective. Data was examined for missing data, and diabetes cases were converted to numeric values, such as 1 or 0. The data analysis revealed that there were a significant number of cases with zero values. To overcome missing or zero values in the dataset, data imputing was done. Following that, eight features were chosen from a total of nine features. The data was split into training and testing sets in the next stage. Afterwards, a machine learning model was developed for predictions using those training data. Following the model's self-training phase utilising training data, testing data was utilised to predict answers and verify accuracy before the model was assessed.

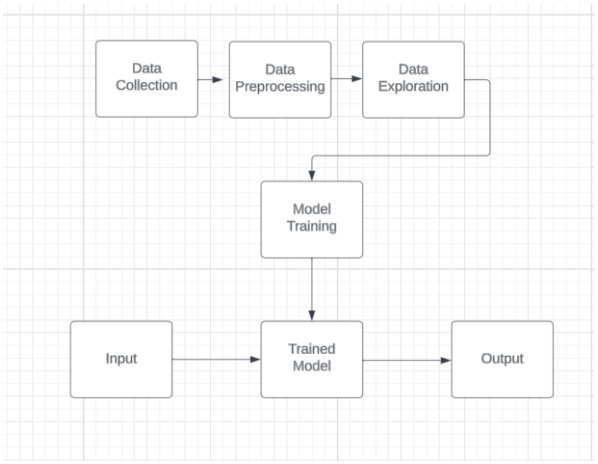
In [7] In the first step, the necessary libraries are imported, and the diabetes dataset is loaded. Subsequently, in the second step, a pre-processing stage is implemented to handle missing data effectively. The third step involves splitting the dataset into training and test sets with an 80-20 percentage split. Moving forward, in step four, a variety of machine learning algorithms, including K-Nearest Neighbor, Support Vector Machine, Decision Tree, Logistic Regression, Random Forest, and Gradient Boosting, are selected for experimentation. Following this, in step five, classifier models are constructed for each chosen algorithm based on the training set. In step six, the performance of these classifier models is assessed using the test set. The seventh step involves a comprehensive comparison and evaluation of the experimental results, considering various performance metrics. Finally, in step eight, after careful analysis of the results obtained for each classifier, the best-performing algorithm is determined, providing valuable insights into its efficacy for predicting diabetes in the given dataset.

In [8] The categorization techniques of Decision Tree, ANN, Naive Bayes, and SVM algorithms serve as the foundation for model construction. The models provide accuracy of 85% for Decision Tree, 77% for Naive Bayes, and 77.3% for Support Vector Machine. Results indicate that the procedures are significantly accurate.

In [9] Data was examined for missing data, and diabetes cases were converted to numeric values, such as 1 or 0. The data analysis revealed that there were a significant number of cases with zero values. To overcome missing or zero values in the dataset, data imputing was done. Following that, eight features were chosen from a total of nine features. The data was split into training and testing sets in the next stage. Afterwards, a machine learning model was developed for predictions using those training data. Following the model's self-training phase utilising training data, testing data was utilised to predict answers and verify accuracy before the model was assessed.

In [10] They suggest using a classification algorithm with increased accuracy to identify patients who have diabetes. Several classifiers, including Decision Trees, KNN, and Naïve Bayes, are used in this model. The main goal is to improve the accuracy by using the resample approach to a benchmark, well recognised diabetes dataset. This dataset was obtained from the UCI machine learning repository's PIMA Indian Diabetes Dataset, which consists of eight characteristics. The framework consists of the subsequent significant stages: • Principal component analysis (PCA) for feature extraction; • Resample filter application; • Dataset selection (PIMA Indian Diabetes Dataset); • Data pre-processing; • Learning via classifier (Training) using Naïve Bayes, KNN, and Decision Trees • Reaching the best accuracy for the trained model.

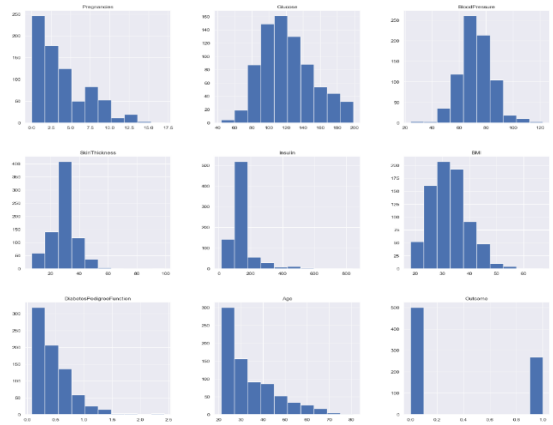
III. METHODOLOGY



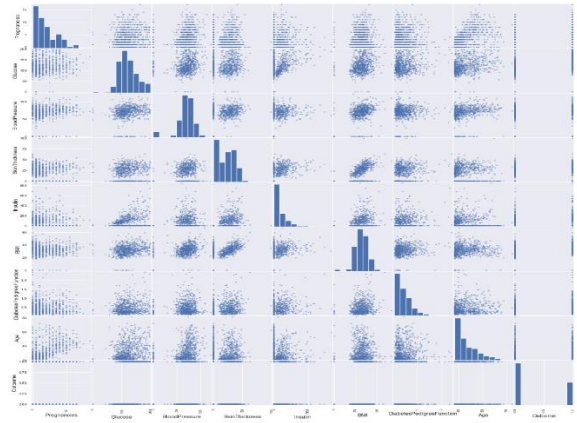
- Data collection:
  The Pima Indians Diabetes [11] dataset is a well-known dataset in the field of machine learning and healthcare. It contains information about Pima Indian women, aiming to predict whether or not a woman will develop diabetes based on certain diagnostic measures.
  Attributes of the dataset:
  - **Pregnancies**: Number of times pregnant.
  - **Glucose**: Plasma glucose concentration 2 hours in an oral glucose tolerance test.
  - **BloodPressure**: Diastolic blood pressure (mm Hg).
  - **SkinThickness**: Triceps skinfold thickness (mm).
  - **Insulin**: 2-Hour serum insulin (mu U/ml).
  - **BMI (Body Mass Index):** Weight in kg/(height in m)^2.
  - **DiabetesPedigreeFunction**: Diabetes pedigree function (a function which scores likelihood of diabetes based on family history).
  - **Age**: Age in years.
  - **Outcome** (Target Variable): 1 if the person has diabetes, 0 if not.

- Data pre-processing:
  The dataset contains '0' values which make no sense in these columns. So, in order to tackle this situation, we replaced the 0 with Nan values and obtained the count values of the 0s in each column.

- Data Exploration:
  We replaced the Nan values with Mean in Pregnancies and Glucose columns and with median in the remaining respective columns.

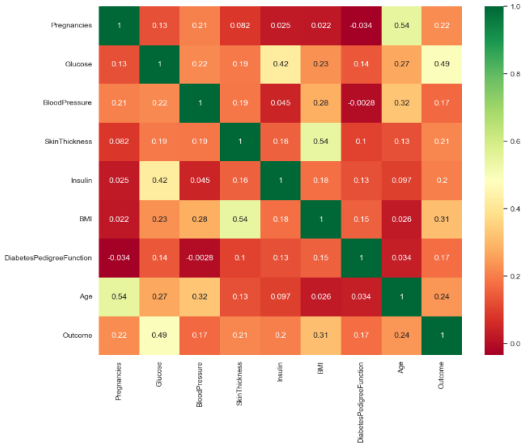


Histograms (hist function) are generated to visualize the distribution of individual features in the dataset.

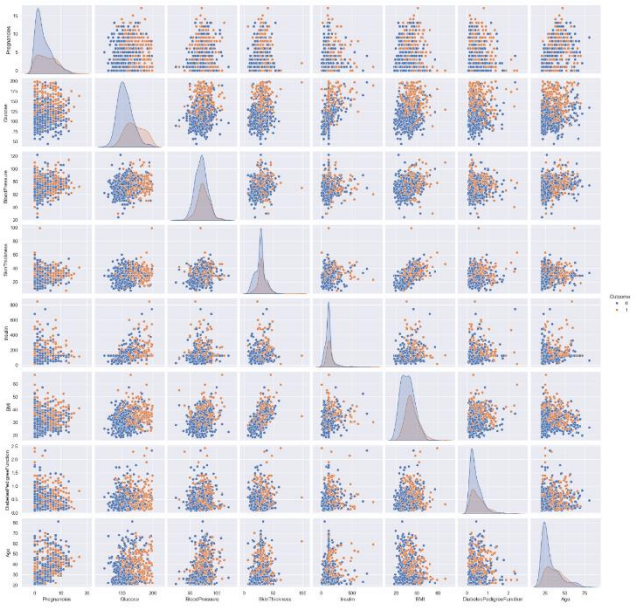


Another function is used to display scatter plots for each pair of features, providing insights into potential relationships.

- Visualization:
  The correlation matrix of the features is visualized.



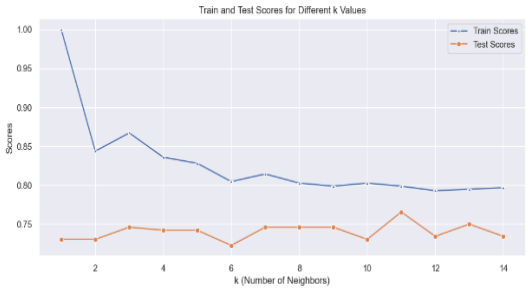
Further, a pairplot is plotted to visualize the relation between the selected features and target variable.



- Choosing classifier:
  KNN, Logistic regression classifier, Support Vector Machine classifier and naïve bayes classifier are compared to select the best accurate algorithm that can be implemented to be used in the final prediction model

  KNN: The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.
  Finding a new data point's k nearest neighbours inside the training set is how the KNN algorithm

operates. The majority class among the new data point's k closest neighbours is then used to determine its class. One hyperparameter that may be adjusted to maximize the efficiency of the algorithm is the value of k.
The test and train scores for different k values from 1 to 15 were as follows



Minkowski distance metric was utilized to calculate the distances.

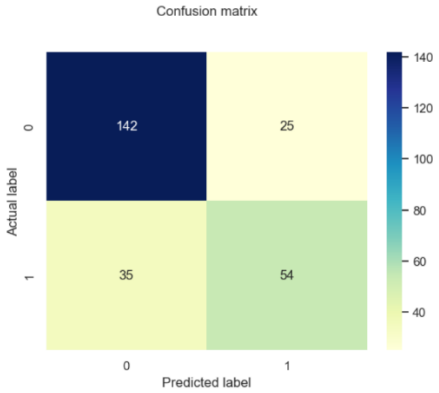
- Logistic regression: This type of statistical model (also known as *logit model*) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1. In logistic regression, a logit transformation is applied on the odds—that is, the probability of success divided by the probability of failure. This is also commonly known as the log odds, or the natural logarithm of odds, and this logistic function is represented by the following formulas:
Logit(pi) = 1/(1+ exp(-pi))
ln(pi/(1-pi)) = Beta_0 + Beta_1*X_1 + … + B_k*K_k
- SVM classifier: Support Vector Machine (SVM) is a robust classification and regression technique that maximizes the predictive accuracy of a model without overfitting the training data. SVM is particularly suited to analyzing data with very large numbers (for example, thousands) of predictor fields. SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane. Following this, characteristics of new data can be used to predict the group to which a new record should belong.
The random forest algorithm works by training many decision trees on different subsets of the training data. Each tree is trained on a random subset of the data, and each feature is considered only a subset of the total features. This helps to reduce the correlation between the trees and make the forest less prone to overfitting.

- Naïve bayes classifier: The Naïve Bayes classifier is a supervised machine learning algorithm, which is used for classification tasks, like text classification. It is also part of a family of generative learning algorithms, meaning that it seeks to model the distribution of inputs of a given class or category. Unlike discriminative classifiers, like logistic regression, it does not learn which features are most important to differentiate between classes.

Results of all the models are present in the next section

IV. Analysis of Results:

The confusion matrix of KNN was as follows:



The scores from the above models are as shown below:

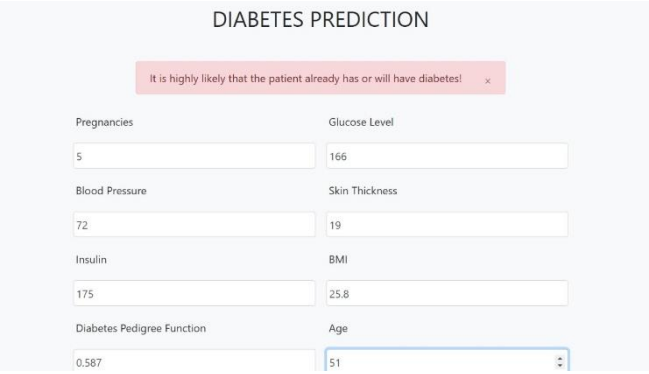
Model	Score
KNN	0.7656
Logistic regression	0.7305
SVM	0.7871
Naïve Bayes	0.7637

We can observe that the SVM classifier performed with best accuracy among the implemented models and Hence, SVM classifier is used for the development of the web application to predict the occurrence of diabetes by taking in the input parameters from user.

Various performance parameters like recall, F1 score, precision and support can be calculated from the confusion matrix. The results are as seen below:

```
Accuracy of our SVM model is : 0.7871
Accuracy: 0.7344
Confusion Matrix for Support Vector Amchine
[[ 45  44]
 [ 24 143]]

Classification Report

              precision    recall  f1-score   support

           1       0.65      0.51      0.57        89
           0       0.76      0.86      0.81       167

    accuracy                           0.73       256
   macro avg       0.71      0.68      0.69       256
weighted avg       0.73      0.73      0.73       256

Accuracy score of the training data:  0.7866449511400652
Accuracy score of the test data:  0.7727272727272727
```

Overall accuracy is 78% using SVM classifier.



Final web app that has been developed by using the flask library of python, after implementing the SVM classifier as the prediction method works as shown above.

V. Conclusion and Future work

The final web application that has been developed by using SVM classifier works and displays the predicted outcome of whether the person gets diabetes or not based on the parameters that have been given as input. The model can be further refined by using Deep Learning algorithms and more diverse dataset for accurate and better results

References

[1] B Nithya and V Ilango, "Predictive analytics in health care using machine learning tools and techniques", *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 492-499, 2017 Jun 15.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Anand Ayush and Shakti Divya, "Prediction of diabetes based on personal lifestyle indicators", *1st International Conference on Next Generation Computing Technologies (NGCT)*, pp. 673-676, 4-5 Sept. 2015.

[3] P. Repalli, "Prediction on diabetes using data mining approach", *Oklahoma State University*.

[4] PS Kumar and S Pranavi, "Performance analysis of machine learning algorithms on diabetes dataset using big data analytics", *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS)*, pp. 508-513, 2017 Dec 18.

[5] Choubey, D.K., Paul, S., Kumar, S., Kumar, S., 2017. Classification of Pima indian diabetes dataset using naive bayes with genetic algorithm as an attribute selection, in: Communication and Computing Systems: Proceedings of the International Conference on Communication and Computing System (ICCCS 2016), pp. 451– 455.

[6] M. A. Sarwar, N. Kamal, W. Hamid and M. A. Shah, "Prediction of Diabetes Using Machine Learning Algorithms in Healthcare," *2018 24th International Conference on Automation and Computing (ICAC)*, Newcastle Upon Tyne, UK, 2018, pp. 1-6, doi: 10.23919/IConAC.2018.8748992.

[7] Mitushi Soni , Dr. Sunita Varma, 2020, Diabetes Prediction using Machine Learning Techniques, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 09 (September 2020)

[8] P. Sonar and K. JayaMalini, "Diabetes Prediction Using Different Machine Learning Approaches," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2019, pp. 367-371, doi: 10.1109/ICCMC.2019.8819841.

[9] Zou Q, Qu K, Luo Y, Yin D, Ju Y and Tang H (2018) Predicting Diabetes Mellitus With Machine Learning Techniques. Front. Genet. 9:515. doi: 10.3389/fgene.2018.00515

[10] Saru, S. and Subashree, S., Analysis and Prediction of Diabetes Using Machine Learning (April 2, 2019). International Journal of Emerging Technology and Innovative Engineering, Volume 5, Issue 4, April 2019, Available at SSRN: https://ssrn.com/abstract=3368308

[11]https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database