**PREDICTION ON PIMA INDIANS DIABETES DATASET USING MACHINE LEARNING APPROACH**

## Problem Statement

Diabetes is a long-term illness that can lead to certain other serious problems. We utilise machine learning to find out about key risk factors so we can avoid these problems. The proposed study will look at how diabetes is diagnosed to figure out how to help. Accurate diagnoses are made with the help of machine learning algorithms like Random Forest, KNN, Decision Tree, SVR and Naïve Bayes. In our research, we looked at the Sensitivity, Accuracy and Specificity, Precision, and F1 score parameter indices.

## Introduction

It is critical for the expansion of the healthcare industry and biotechnology that biotechnology and sophisticated computational research work together to improve computer efficiency. The primary goal of this research is to investigate and get a deeper knowledge of the ever-increasing amount of biological data in order to provide the foundation for future healthcare and the framework of biology. Processes created by systems with access to a large data bank and solid models capable of making predictions. Traumatic events, stress, obesity, and malnutrition are all examples of physical and mental stresses that may hasten advances in the identification and treatment of human disease [1]; this is yet another significant discovery. This is a significant result. According to the World Health Organization (WHO), the elderly have the greatest prevalence of mental illness in the country. Since the World Health Organization launched its campaign in 1986, 451 million individuals have been diagnosed with diabetes. It is anticipated that there will be 693 million persons by 2045 [2]. This will happen throughout the course of the next two decades. Scientists think that both genetics and environment have a role in the progression of diabetes. Despite the fact that there is no known cure for the condition, it may be controlled with a range of medications. Diabetes patients should avoid using this product since it may risk their health and cause heart attacks or organ damage. If you address the problem as soon as it arises, there is a lower chance that it may progress to a more serious health condition that would necessitate treatment.

Using a computer to enhance diagnostic abilities and information may be important and effective when making judgments. This might be quite important. DM field measurements may be analysed in a laboratory to yield a wealth of information about the patient, his or her medicines, and other aspects of therapy. The majority of the time, data does not fit together neatly, and it must be assembled piece by piece. The ineffective data management has an impact on the data organisation. Before it can be used, information must be obtained and processed in a way that increases the amount of data. To satisfy the growing demands of present and future hospital information systems, hospitals commonly deploy a device that facilitates data collection, analysis, and sharing. Finding DM is significantly easier and more exact with automated technologies than with a human physician. It is also more accurate than manual diagnosis. There is no difference in the accuracy of reports that are confusing and untrustworthy and those that are clear and trustworthy. These automated DM systems can be created employing techniques from the field of artificial intelligence, such as approaches based on humans or learning systems. Diabetes occurs when the pancreas fails to create enough insulin or when the body's cells and tissues are unable to utilise the insulin produced adequately. Diabetes manifests itself in three ways: well-controlled diabetes, partially-controlled diabetes, and uncontrolled diabetes [3].

# **Coding Tool**

Jupyter notebook is an integrated development environment (IDE) (IDE). It is open source and supports different programming languages. You may use the Jupyter notebook to produce and distribute reports that include live code, circumstances, and impressions. This might be used to clean, convert, model, and display data.

# **Dataset Description:**

For the sake of this investigation, the repository at the University of California, Irvine was collected and the data was posted in the Kaggle repository [4] . There, Pima Indians used PIMA to diagnose diabetes. There are eight feature numbers for each of the 768 patient occurrences. In order to detect diabetes, this study investigates the characteristics of the dataset. Diabetes is present in 268 of the patient events, whereas it is absent in the remaining 500. In 2011, there were no blanks in the diabetes dataset that was stored in the UCI repository; however, NA values were subsequently discovered and were filled in with zeros. The characteristics of this database are either numbers or integers. The data gathering characteristics are connected to the class label. The outcome is a dependent variable on various medical predictor variables. Insulin level, BMI, age, and number of births of the patient are all independent variables.

Table

Description automatically generated with medium confidence

The procedure data.describe() generated these statistics.

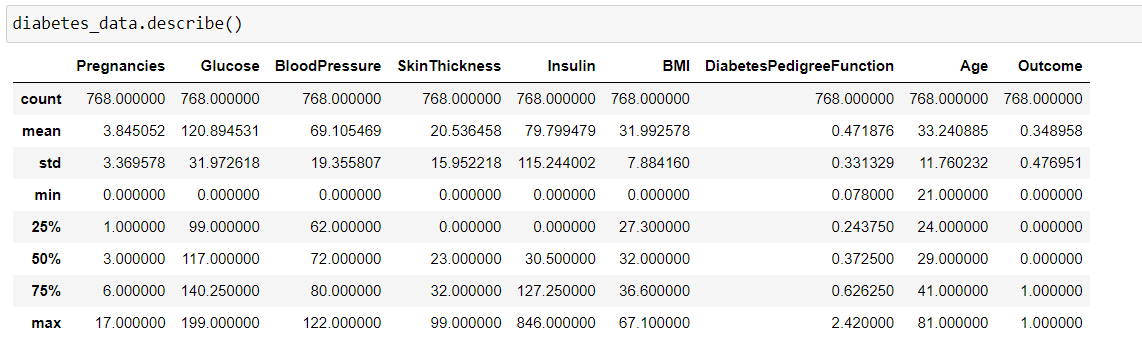
• count delivers the number of columns that are not empty inside a feature.

• Average denotes the mean value of the property.

• std returns the feature's Standard Deviation.

• basic displays the feature's bare minimum.

• The percentiles/quartiles for each attribute are 25%, 50%, and 75%, respectively. Outliers can be identified using this quartile data.

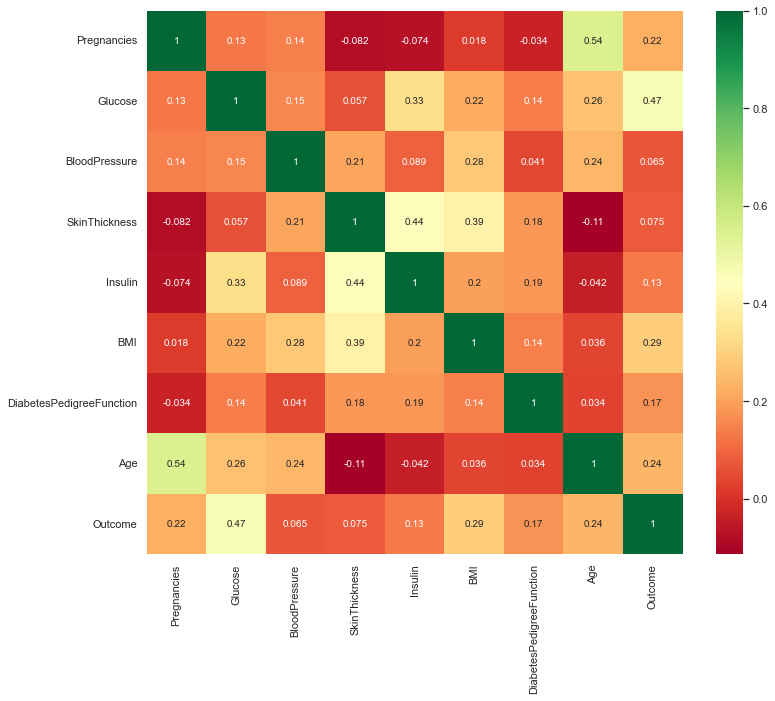
* maximum indicates the maximum value of the characteristic.

The graph depicts unbalanced data with 268 non-diabetic patients and 500 diabetic ones.

Chart, bar chart

Description automatically generated

## Correlation Matrix:

The table in question includes the coefficient of correlation for each set of variables for a particular pair of variables. All of the stochastic process (Xi) within the table are interconnected (Xj). This enables users to determine which combinations have the strongest association

# **Data preprocessing**

# Missing data and/or irregular and outliers are prevalent in real-world data. If the data is of poor quality, it is possible that there will be no quality outcomes. For optimal results, you must first cleanse your data. Data is cleaned, integrated, transformed, reduced, and discretized during preprocessing. Prior to data mining and analysis, the data must be enhanced in regards to cost, time, and quality.

# **6.1 Data cleaning**

Two of the most significant parts of data cleansing are filling in blanks and reducing noise. To settle conflicts, noisy data are cleaned of outliers . In our dataset, various measurements of blood pressure, glucose, insulin, skin thickness, and BMI had 0 (zero) values. In other words, the median of that property was substituted for all the zero values.

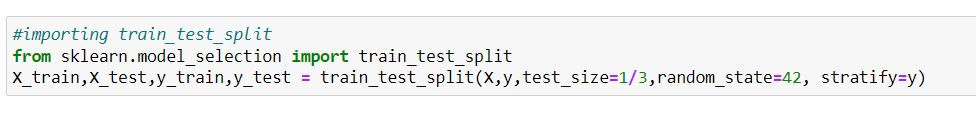
# **6.2 Data reduction**

# The amount of the data can be reduced by data reduction while the findings remain same. Dimensional reduction is the process of reducing the number of characteristics in a dataset. The principal components analysis was utilised so that essential data could be found. A substantial relationship was found between glucose levels, diastolic blood pressure, body mass index, and age.

# **6.3 Data transformation**

All of these processes—correction, normalisation, and accumulation—are covered. The binning procedure was utilised so that the data could be smoothed out. There are five distinct age brackets to choose from. Blood glucose levels in people who do not have diabetes are not the same as those in individuals who do have diabetes. It is divided up into five distinct parts. There is a considerable correlation between normal blood pressure and the risk of developing diabetes. There is a substantial correlation between body mass index and diabetes. Both diabetes and obesity are seeing rising rates of prevalence all across the world. In addition, obesity is the primary factor responsible for the development of type 2 diabetes. After the data has been cleansed, the remaining step in the preprocessing step is to choose the key features and turn them into bins.

**Splitting:** The original data set was divided into two parts: training and testing. The remaining 25% of the data is solely needed for testing, leaving 75% available for training.



Since the data was 768, using SMOTE analysis ; the imbalance observed towards the outcome (1) was improved. SMOTE (synthetic minority oversampling technique) are some of the most often employed oversampling strategies for addressing the imbalance issue. It seeks to balance classification performance through replicating minority class examples at random. SMOTE creates new minority instances by combining existing minority instances. With SMOTE, 768 data was improved to 922.

Below SMOTE

Chart

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

After SMOTE

Chart

Description automatically generated with medium confidenceTable

Description automatically generated

# **Machine Learning**

Constructing a system that is capable of learning on its own, without the need for guidance from a human teacher, is the goal of machine learning. The success of machine learning is strongly reliant on the algorithms that make it possible to generate new models. It is conceivable, based on the data that a machine-learning model receives, to speculate about the output that it will produce. Machine learning models are built with the help of sample datasets, and the techniques that are used to build the model are modified so that they are appropriate for the characteristics of the dataset. The training data set is given as an input at the beginning of step one of the procedures.

• Data is fed into an algorithm for machine learning, and the algorithm is taught. At this stage, the algorithm begins to recognise patterns within the data.

• The construction of the model begins once the algorithm has identified the patterns present in the data that is presented. The model makes projections regarding its output when it obtains new data and bases those projections on the data.

# **Random Forest**

The purpose of ensemble learning is to generate a model that is more accurate by combining a number of different machine learning (ML) techniques; it is the application of collective intelligence to information science. They are based on the principle that it is possible for a group of people who have limited understanding of a given problem to develop a solution that is superior to the solution developed by an individual who has more information.

A decision tree is a well-known metaphor for the process of finding a solution to a problem. A random forest is indeed a collection of choice trees. A decision tree works by asking a series of yes/no questions regarding different pieces of data in order to arrive at a verdict. In the following illustration, in order to calculate an estimate of a family's salary, a decision takes into consideration a number of variables (features), including the individual's job status (yes or no) & homeownership status. In the context of an algorithm, the machine searches incessantly for the property that enables the observations in a set to be partitioned into groups that are as markedly different to one another as is practically possible, while at the same time ensuring that the individuals that make up each distinct subgroup are as similar to one another as is reasonably attainable. The random forest algorithm constructs whole decision trees in parallel by drawing random bootstrap samples from the data set and utilising the "bagging" technique to analyse the features. The success of a random forest is primarily dependent on chance, in contrast to the success of decision trees, which are based on a predetermined set of features and frequently overfit.

A method known as bootstrap aggregating is utilised by each tree that makes up a random forest in order to select chunks of the training data at random (bagging). The model is applied to these more limited data sets, and then the predictions are weighted and averaged. Replacement sampling allows for several instances of the same data to be used again, which results in decision-making trees that not only trained on different data sets, but also make use of different characteristics.

**Experimental Results:**

Table

Description automatically generated

Here the accuracy of random forest is scored around with the 75% and the ROC score was 0.82

Chart, line chart

Description automatically generated

# **KNN**

To group issues, the rule-based expert for machine learning known as "k nearest neighbours" is usually used. It comes from the saying "The apple doesn't fall far from the tree," which indicates that related goods are always close together. This method calculates the location of each data point based on how its neighbours are classified. Each new sample is allocated to a category based on its resemblance to all prior samples. Technically, the criterion for an unclassified item is to analyse the k objects closest to it that have been classified. This is performed by finding which of the nearby items has the greatest characteristics in common with the objects used to map the unknown object. The majority vote is then used to classify the unknown object.

The algorithm is dubbed "lazy" since it does not go through a training phase. Instead, it recalls the learning from the training dataset. All computations were placed on hold until grouping was finished.

A set of case-based learning requirements Because it makes predictions based on the problem domain's raw training examples, this type of learning algorithm is also known as an instance-based or case-based training algorithm. The KNN approach does not directly train a model since it uses case studies to learn. Instead, it recalls training examples and circumstances, which are then used as "knowledge" in the next step, known as "prediction." When we give the algorithm an input and ask it to estimate a label, it analyses the prior training samples to deliver a response.

A non-parametric technique has either a set number of variables that remain constant regardless of the number of data points collected, or no parameters at all. Regardless matter how much data is reviewed, K is the sole unknown parameter in KNN. Even if no assumptions are made about the nature of the issue, a trade-off must be made due to the escalating cost of computing. Even though KNN is simple to grasp, it needs a substantial amount of memory and computer resources. Memory cost, since a large data set must be stored, and compute cost during the test, because categorising a single observation requires traversing the whole data set. These costs are incurred throughout the testing process.

The letter K represents the number of points that the KNN algorithm considers to be close to the current place. The KNN method's most important parameter is K, and obtaining the optimal value for K is tough. Parameter tuning refers to the process of obtaining the ideal value for K, which is required for improved accuracy. If K is set too low, the data may be overfit to the model; yet, if K is set too high, the approach will be difficult to execute on a computer. When analysing only two classes, most data scientists will use an odd K number (learn more about data scientists here). The formula k = k\*sqrt(n), where n is the total number of data points, can also be used to determine K.

The right K value for a given event must be decided on an individual basis. However, testing with different K values and comparing the results is sometimes the most effective way for finding K. The KNN technique may be tested with different K values using cross-validation. The ideal K value is consequently the one that produces the most exact results.

**Experimental Results:**

For 7 neighbors, the accuracy score is 77.73% and the ROC score was 0.81

A picture containing text, receipt

Description automatically generated

Chart, line chart

Description automatically generated

# **Decision Tree**

The majority of the time, problems involving classification may be solved by employing decision trees, which are an example of supervised learning. Because of this, decision trees require prior knowledge about the target variable in order to function properly. This tree can be put to use with discrete or continuous inputs and outputs, depending on the particular circumstances it is used to. The process of training is organised in the form of a flow chart, where each internal node portrays a test of an attribute, each branch illustrates the outcomes of that test, and each leaf (terminal) node has a class label assigned to it. The "root node" of the tree is the node that is at the very top of the tree.

The sample, which represents the population, is partitioned into two or more sub-population sets of the greatest size that can be achieved throughout the process of decision-making. This division is determined by the input variables that have the greatest significant splitter or differentiator built into them.

Building a prediction model that is able to accept observations about a sample (the branches) and utilise those data to produce accurate predictions about the sample's target value is the ultimate objective of this project (the leaves).

**Experimental Results:**

The accuracy of decision tree was 67.96% and the ROC was 0.66

A picture containing text, receipt

Description automatically generated

**Chart, line chart

Description automatically generated**

# **Bayes Theorem**

Naive Bayes is indeed a good illustration of how the simplest answers are often the most potent ones, which is generally the case. Machine learning has come a long way in recent years, but despite these developments, it has shown that it can still be kept straightforward while also maintaining its speed, accuracy, and dependability.

It has proven effective for a wide range of applications, but the natural language processing challenges stand out as particularly amenable to its application. Naive Bayes is just a probabilistic MLmethod that is based just on Bayes Theorem and is employed in a broad range of classification tasks. It was named after the inventor of the Bayes Theorem.

The Bayes theorem gives us a formula to use when determining the likelihood of a hypothesis provided the information that we already possess. The Naive Bayes algorithm is a classification method that may be applied to issues involving binary (two-class) or multi-class categories. When presented utilizing binary and categorical input values, the method is easiest to comprehend in its entirety when read about. The computation of the probability for each hypothesis is simplified in naive Bayes, also known as idiot Bayes, in order to make its calculation tractable. This simplification is the reason for the name.

* P(h|d) = (P(d|h) \* P(h)) / P(d)
* Where
* P(h|d) is the Given the data d, what is the likelihood of the hypothesis h. This is what statisticians refer to as the posterior probability.
* P(d|h) is the Considering that hypothesis h was correct, what is the probability of the data d?
* P(h) is the percentage of chance that hypothesis h is correct . This is what statisticians refer to as the posterior distribution of h.
* P(d) the likelihood given the information .
* **Experimental Results:**
* The accuracy of Naïve Bayes was 71.48% and the ROC Score is 0.79

A picture containing text, receipt

Description automatically generated

**Chart, line chart

Description automatically generated**

# **SVM**

Support Vector Machines, often known as SVMs, are frequently and extensively utilised in the machine learning community for solving classification problems. The objective of regression would be to locate a function that, just on basis of the a training examples, can provide an approximation of a map from such an input space to real numbers. When we move forward with SVR, our goal is to essentially consider the points are within the choice boundary line. This will allow us to make a more informed judgement. The hyperplane that contains the most points is the one that provides the best fit for our data.

**Experimental results**

The accuracy of SVR 76.17%. Since due to the hyper plane , the predict probability was not available in ROC therefore, the roc score or curve was not available

Table

Description automatically generated

# **Performance Metrics**

Accuracy (ACC): Accuracy is the percentage of samples with valid predictions. It's derived by dividing the amount of true positives and negatives by the number of forecasts.

(1)

Precision: Precision is a measure of how many of our model's predictions were accurate out of all forecasts produced.

(2)

Sensitivity: Sensitivity refers to the number of accurately predicted positive records.

Sensitivity= (3)

Specificity: The specificity measures the proportion of all sample that were able to be correctly identified as false.

Specificity (4)

F-Score: The F1 score is a popular performance metric for classification and is frequently favoured above, for instance, accuracy when data is imbalanced, such as when the number of samples belonging to one class much outnumbers those belonging to the other class.

(5)

where TP is related to true positive, TN is true negative, FP and FN are the false positive, and false negative.

AUC-ROC: Both the receiver operating curve (ROC) and the area under curve (AUC) of the ROC together show the degree to which classes can be distinguished from one another.

# The area under the receiver operating characteristic curve, or AUC, is the chance that the recognised model ranks a good example higher than a negative example. The area under the curve (AUC) can range from zero to one and a half, with a maximum value of one. Measuring and monitoring performance is essential to the success of any firm. It is critical for businesses to identify their primary performance measures and concentrate their efforts on improving in these areas since measurements like these are used to direct and evaluate an organization's progress. The usefulness of key success variables is contingent on whether or not they are recognised and recorded.

|  |  |  |
| --- | --- | --- |
| **Model** | **Accuracy** | **ROC-AUC Score** |
| Random Forest | 75% | 0.82 |
| KNN | 77.73% | 0.81 |
| Decision Tree | 67.96% | 0.66 |
| Naïve Bayes | 71.48% | 0.79 |
| SVR | 76.17% | - |

# **Conclusion**

Through the use of predictive analysis, which is made possible by the development of improved classification strategies, we are working toward the goal of enhancing the quality of life of people all over the world by reducing the risk of diabetic complications and identifying them at an earlier stage. Aspects of the work that we have presented include conducting an analysis of the chosen characteristics and choosing the factors that provide the highest correlation values as the most ideal candidates. The objective of the research is to create a model that is capable of more precise classification of the instances contained in the dataset. The overall potential of the dataset has been improved thanks to the application of methods such as feature selection and data cleaning, which have contributed to the development. With KNN, the Classifiers were able to achieve an accuracy of 77.73%. The Support vector regression (SVR) scored 76.17% accuracy which is second best. TheRandom Forest classifier which achieved an accuracy of 75% on the list at third, while the naïve bayes and Decision Tree classifier were at last , which places it in last place on the list among the classifiers. Following a comparison of the available models, we have arrived at the conclusion that the KNN is the one that is most suited for the dataset that contains both diabetic and non-diabetic individuals.

# **References**

[1] V. Patil and D. R. Ingle, "Comparative Analysis of Different ML Classification Algorithms with Diabetes Prediction through Pima Indian Diabetics Dataset," 2021 International Conference on Intelligent Technologies (CONIT), 2021, pp. 1-9, doi: 10.1109/CONIT51480.2021.9498361.

[2] P. Cıhan and H. Coşkun, "Performance Comparison of Machine Learning Models for Diabetes Prediction," 2021 29th Signal Processing and Communications Applications Conference (SIU), 2021, pp. 1-4, doi: 10.1109/SIU53274.2021.9477824.

[3] P. Cıhan and H. Coşkun, "Performance Comparison of Machine Learning Models for Diabetes Prediction," 2021 29th Signal Processing and Communications Applications Conference (SIU), 2021, pp. 1-4, doi: 10.1109/SIU53274.2021.9477824.

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

[5] Sisodia, D., Sisodia, D. S. (2018) “Prediction of diabetes using classification algorithms.” Procedia computer science 132: 1578-1585.

[6] Wu, H., Yang S., Huang, Z., He, J., Wang, X. (2018) “Type 2 diabetes mellitus prediction model based on data mining.” Informatics in Medicine Unlocked 10: 100-107.

[7] F. -J. Yang, "An Implementation of Naive Bayes Classifier," *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2018, pp. 301-306, doi: 10.1109/CSCI46756.2018.00065.