Predicting Diabetes In Patients Using AI Algorithms and Neural Networks

Final Review Document

Group No. 5

ANKIT KUMAR - 19BCE0071

ANSHUL ANAND - 19BCI0065

JAYDEV JANGITI - 19BCT0235

GUNIK LUTHRA - 19BCE2285

Submitted to,

Prof. Gladys Gnana Kiruba B, SCOPE

School Of Computer Science And Engineering

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

TABLE OF CONTENTS

Abstract

Diabetes Mellitus is a chronic, lifelong metabolism disorder that affects the ability of the body system to use the energy found in food. People living with high blood sugar will experience polyuria (frequent urination), which will make them to become increasingly thirsty (polydipsia) and hungry (polyphagia). The improper management of this disease can lead to complication such as cardiovascular disease, kidney disease, eye disease, nerve disease, pregnancy complication.
The database of Pima Indian diabetes has been considered for the diagnosis of the diabetes mellitus. This database comprises of certain attributes which are very adequate for diabetes mellitus diagnosis. The use of this attributes has enhanced the training and test classification of patients, whether diabetes is present or not.
This works aims at performing predictions of the presence of diabetes based on the health data of women present in the dataset. Different machine learning approaches have been employed to perform the predictions and the dataset is cleaned, pre-processed and fed as input to the algorithms that have been used. Supervised machine learning classification algorithms are initially employed such as Nave Bayes, Gradient Boosting Classifier, Logistic Regression, Extra Trees Classifier. After which the dataset was trained and tested with Ensemble models which are a combination of these supervised machine learning classifiers. Voting Ensemble is a combination Of Logistic Regression, SVM & Decision Tree Algorithm. These three algorithms are set up linearly to do the predictions and the results were analysed accordingly. The supervised machine learning approaches were analysed based on the performance measure scores such as F1 scores, accuracy score, Recall scores etc. The Gradient Boosting Classifier performed the best with an average accuracy score of 91%. Alternatively a neural network approach was designed using the Keras library. Deep Learning is becoming a very popular subset of machine learning due to its high level of performance across many types of data. For our proposed solution we have used a Sequential model build a neural network. A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor. Our first network is a single layer network. We have 8 variables, so we set the input shape to 8 and a single hidden layer with 12 nodes to perform the classification. The neural network approach was subsequently trained and tested with 1500 epochs and its performance measured indicated an overall accuracy of about 93%.

Keywords : *Machine Learning, CNN, Random Forest, Gradient Boosting, Diabetes Mellitus, Artificial intelligence, Performance measure, F1 Scores, Ensemble, Dense Neural Network, Sequential Model.*

1. Introduction

Diabetes is a long-lasting disease that happens when the pancreas fails to create enough insulin, or when the body cannot use the insulin produced efficiently. Insulin is a hormone that controls the level of sugar in the blood. Hyperglycaemia is a common result of uncontrolled diabetes and, over time, causes severe damage to many organs, particularly nerves and blood vessels. There are two types of diabetes, type I and type II diabetes. Type I diabetes also named insulin dependent and type II diabetes named relative insulin deficiency. Diabetes is one of the growing extremely fatal diseases all over the world. Diabetes causes a large number of deaths each year and a large number of people living with the disease do not realise their health condition early enough. Diabetes affects between 2% to 4% of the global population and its avoidance and effective treatment are undoubtedly crucial public issues in the 21st century. Although human decision making is often optimal, it is poor when there are huge amounts of data to be classified.

AI is a branch of computer science that aims to create systems or methods that analyse information and allow the handling of complexity in a wide range of applications (in this case, diabetes management). Machine Learning (ML) is a methodology for allowing a machine to learn independently by employing a variety of techniques to detect patterns and relationships in data. AI technologies allow for the automatic diagnosis of any condition, with the two most important components being parameter selection and the instrument used to analyse these parameters carefully examined. These technologies have the potential to transform many aspects of patient care, as well as administrative processes within provider, payer and pharmaceutical organisations. The scalability and versatility of ML algorithms are rapidly used in risk stratification forecasts. AI and ML implementation areas are quickly expanding, and one of the most influential areas of AI has been in medical diagnostics. Although the application of AI algorithms involves highly technical and specialised knowledge, this has not prevented AI from becoming an essential part of the technology industry and making contributions to major advances within the field. These ML techniques can be used to learn from patient blood-work trends and accurately predict the existence of diabetes in women. In this proposed work ML algorithms were employed and such as Nave Bayes, Gradient Boosting Classifier, Logistic Regression, Extra Trees Classifier. After which the dataset was trained and tested with Ensemble models which are a combination of these supervised machine learning classifiers and a neural networks approach.

Authors and Year (Reference)	**Title (Study)**	**Concept / Theoretical model/ Framework**	**Methodology used/ Implementation**	**Dataset details/ Analysis**	**Relevant Finding**	**Limitations/ Future Research/ Gaps identified**
Changsheng Zhu, Christian Uwa Idemudia, Wenfang Feng: Informatics in Medicine Unlocked Volume 17, 2019, 100179	Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques	When the k-means approach is used, this research effort presents PCA for dimensionality reduction, which aids in defining appropriate initial centroids for our dataset. The data is then clustered into similar categories using K-means, with logistic regression serving as the dataset's classifier. A overview of related work by other researchers in the domain of diabetes	The proposed model was created and implemented by combining the advantages of PCA, K-means, and Logistic regression. The problem of correlation, which makes it difficult for the classification algorithm to detect relationships among the data, is then solved by applying PCA to alter the initial collection of features, resulting in a new methodology. The PCA application aids in the filtering of irrelevant information,	This work made use of the Pima Indian Diabetes dataset from the UCI machine learning library. The dataset consists of 768 female patients who were tested for diabetes in Arizona, the United States. There are eight attributes in total (indicating medical diagnosis criteria) in the dataset, with one target class (which represents the status of each tested individual). There are 268 tested positive instances and	In comparison to earlier research, the experimental results revealed that using PCA improves the k-means clustering procedure, as we acquired 614 accurately clustered datasets. Han Wu et al. reached an accuracy of 95.42 percent from a sample size of 589 using k-means clustering and had the closest result to ours. We can clearly demonstrate that the proposed PCA and K-means	The goal of this study was to create an effective model for diabetes prediction. Author suggested a unique approach based on PCA for dimensionality reduction, k-means for clustering, and logistic regression for classification after a thorough review of other published work. They first used the PCA technique on their dataset in order to improve the k-means results of other

		prediction and diagnosis is presented in this study, followed by details of the experimental procedures. The other section summarizes the results of the experiment and finishes the paper by offering prospective directions for further research.	reducing training time and cost while also improving model performance. Because of k-means' capacity to resolve outliers, the output of PCA analysis is subsequently passed for unsupervised clustering using K-means. After cleaning the K-means cluster result, we use Logistic Regression to create our supervised classification for the dataset.	500 tested negative instances in the sample.	techniques increased the classification accuracy of logistic regression for the Pima Indian diabetes dataset using our experimental results.	researchers. Despite the fact that PCA is a well-known technique, it has not received enough attention for its effectiveness in enhancing k-means clustering and, as a result, the logistic regression classification model. They demonstrated that combining PCA with k-means can result in a better logistic regression model for predicting diabetes in our experiment.
Mukesh kumari, Dr. Rajan Vohra, Anshul arora:	Prediction of Diabetes Using Bayesian Network	To predict diabetes and cardiovascular illness, we apply supervised	A Bayesian network is a graphical model that represents probabilistic	This is a sample of the dataset that was used in the prediction. There are 206	The data for this study was gathered from a hospital. Pre-processing is a	This research has certain drawbacks. For starters, there could be other

International Journal of Computer Science and Information Technologies, Vol. 5 (4), 2014		machine learning models in this research. Despite the acknowledged link between the two diseases, we designed the algorithms to predict CVD and diabetes separately so that a larger spectrum of individuals can benefit. As a result, we are able to find feature commonality between diseases that influence their prognosis. Prediction of prediabetes and undiagnosed diabetes is also taken into account. Multiple models for the prediction of	interactions between variables. The graphical model provides various advantages for data analysis when used in conjunction with statistical techniques. One, because the model captures all of the variables' interdependencies, it can easily handle scenarios where certain data entries are missing. Two, a Bayesian network may be used to learn causal linkages, allowing it to obtain a better grasp of a problem area and predict intervention outcomes. Three, the model's causal and probabilistic semantics make it	cases in the dataset. There are nine input attributes (X1 to X8) and one output attribute in each instance (Y1). This dataset's attributes are displayed in the table below. The primary attribute is used to solve this problem. Fast plasma glucose concentration in an oral glucose tolerance test, casual plasma glucose tolerance test, and diastolic blood pressure (mmHg) are the dataset variables utilised for diabetes prediction. If a person's fasting plasma glucose is less	technique for improving data quality. Attribute identification and selection, data normalisation, and numerical discretization are some of the pre-processing techniques used. The Bayesian model is then built by applying a classifier to the changed dataset. Finally, Weka will be used to simulate the model, and the model's accuracy will be calculated and compared to the efficiency of other algorithms. When the results were compared to	risk factors in the diabetes dataset that the data collecting did not address. Other key factors, according to, include gestational diabetes, family history, metabolic syndrome, smoking, sedentary lifestyles, certain food habits, and so on. To improve the accuracy of the prediction model, more data would be needed. This can be accomplished by assembling diabetes datasets from many sources and creating a model from

		various diseases are trained and tested using the National Health and Nutrition Examination Survey (NHANES) dataset. In addition, a weighted ensemble model is investigated in this study, which combines the outcomes of numerous supervised learning models to improve prediction ability. A dataset of people is obtained from the hospital and given into the software, Weka, which returns the total number of diabetic, non-diabetic, and pre-diabetic	an appropriate representation for mixing prior knowledge (which is frequently in causal form) and data. Four, Bayesian statistical approaches combined with Bayesian networks provide a practical and logical method for avoiding data overfitting. To strategies for supervised and unsupervised learning, Bayesian-network methods are used. A real-world case study is used to demonstrate the graphical-modeling approach.	than 100 mg/dl and their casual glucose tolerance test is less than 140 mg/dl, they will receive a score of zero, indicating that they are not diabetic.	clinical diagnosis, the Bayesian network classification had the best accuracy, 99.51 percent, while the classification error was.48 percent. The root mean squared error (MRES =.0596) and mean absolute error (MEA) are both equal to.0053. When comparing classification algorithms, the overall time necessary to develop the model is also an important factor to consider.	each one. Second, we solely used Bayesian networks to predict diabetes in this study. Because several diabetes features have unknown factors, a fuzzy set method will be used in future study to improve Bayes Network prediction. Other machine learning approaches, such as Neural Network, will also be tested to compare the forecasting outcomes in order to determine the best prediction model.

		people. The classification will be based on the value of primary attributes. The dataset would be divided into three groups using this method.				
Saloni Kumari, Deepika Kumar, Mamta Mittal: International Journal of Cognitive Computing in Engineering, Volume 2, 2021, Pages 40-46,	An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier	The proposed ensemble soft voting classifier uses an ensemble of three machine learning algorithms to provide binary classification, including random forest, logistic regression, and Naive Bayes. The proposed methodology was empirically evaluated using state-of-the-art methodologies and base	The goal of this study is to improve the findings and accuracy of diabetes detection. The authors suggested an ensemble of machine learning methods for binary disease classification into positive and negative utilising a soft voting classifier. Before providing input to the model, data pre-processing was performed, followed by data	The Pima Indian Dataset was used for experimentation in this study (Pima Indians Diabetes dataset May 2008). The dataset comprises nine columns and one output column with a binary value indicating whether the person has diabetes or not. It comprises 768 rows, 500 of which are non-diabetics and 268 of which are	Random Forest, Logistic Regression, and Nave Bayes with soft voting classifier are used in the proposed methodology as an ensemble of three machine learning models. The PIMA diabetes dataset was used in the experiments. The dataset contains 769 data points and 10 feature columns, each with a median	The authors present a soft voting classifier model based on a combination of three machine learning algorithms: random forest, logistic regression, and Naive Bayes. The proposed model was first tested on the Pima Indians diabetes dataset, following which it was used to the breast cancer dataset.

		classifiers such as AdaBoost, Logistic Regression, Support Vector Machine, Random Forest, Nave Bayes, Bagging, GradientBoost, XGBoost, CatBoost. The evaluation criteria were accuracy, precision, recall, and F1-score. On the PIMA diabetes dataset, the suggested ensemble strategy has the highest accuracy, precision, recall, and F1 score value, with 79.04 percent, 73.48 percent, 71.45 percent, and 80.6 percent, respectively.	augmentation. Data pre-processing is a crucial step that transforms data into a usable and efficient format that can be fed into a machine learning algorithm. Data normalisation is the first technique used for data pre-processing. This method is used to conduct linear data transformations. We used an ensemble of machine learning algorithms in this proposed methodology, including Logistic Regression, Naive Bayes, and Random Forest classifiers. To improve accuracy, the aforementioned	diabetic patients. The dataset contains nine feature columns, including pregnant month, glucose, plasma, blood pressure, fold thickness of triceps skin, insulin amount, BMI, Pedigree function, and patient age, as well as one target column (0 or 1).	value of zero. With 20% and 70% respectively, the dataset has been separated into testing and training datasets. The most popular assessment measures used to check the resilience and effectiveness of algorithms are accuracy, precision, recall, and F1 score. True positive(tp) indicates that the predicted class value is 1 and the actual class value is also 1. True negative (tn) indicates that the predicted class value is 0 and the actual class value is	On the Pima Indians diabetes dataset, the ensemble soft voting classifier produced 79.08 percent accurate results and 97.02 percent accurate results on the breast cancer dataset. Using alternative deep learning models in the future, this accuracy could be improved.

		The proposed methodology's efficiency has also been evaluated and analysed using a breast cancer dataset. On the breast cancer dataset, the suggested ensemble soft voting classifier had 97.02 percent accuracy.	algorithms were combined with a soft voting classifier.		also 0. When your anticipated class contradicts the actual class, false negatives (fn) and false positives (fp) occur. Accuracy is the most important measure and it is a ratio of total correctly predicted observation to the total number of observations.	
Rahman Shaque, Arif Mehmood, Saleem ullah, Gyu Sang Choi: September 16th, 2019	Cardiovascular Disease Prediction System Using Extra Trees Classier	Blood pressure, cholesterol levels, chest discomfort, and 11 other variables used to predict cardiovascular disease are all included in the dataset. Verdict Tree, also known as Decision Tree, Extra Trees Classifier,	We used a confusion matrix to assess the quality of a classifier's output on the dataset. The predicted label, which is identical to the true label, refers to all diagonal elements that represent correctly labelled points. Other offdiagonal items are labels	The dataset was collected from the Kaggle Cleveland repository and contains both categorical and numerical attributes. cp, FBS, restecg, ca, and thal are the category qualities. Age, sex, trestbps, chol,	Experiments were conducted with a variety of classification models, each of which produced varied results for the target class. Extra Tree Classifier, Logistic Regression, Support Vector Machine, and Naive Base were all used.	The Extra Tree Classifier enabled UCI information has the highest accuracy (90.00%), whereas the LMT, SVM, and NB algorithmic rules have the lowest accuracy (88%), 87%), and 86%), respectively. In

		Random Forest, Support Vector Machine, Naive Bays, and Logistic Regression are some of the most prevalent and effective classification approaches used in mining. Extra classifier trees are the best strategy for diagnosing and managing the ratio of fatalities from cardiovascular disease. We use the assessment parameters Accuracy, Precision, Recall, and F1-score to evaluate these prediction models. Extra trees classifier, Logistic Model tree classifier,	that the classifier hasn't properly labelled. The higher the values on diagonal spots, the more accurate the classifier's predictions. The following are the basic terms for the confusion matrix and how it works. True Positive: Prediction is diagnosed, and it is really branded as such. True Negative: Prediction is not diagnosed, and it is really classified as such. False-Positive: Predictions are diagnosed, but they are actually categorised as undiagnosed. False-Negative Predictions are not diagnosed, but they are branded as such.	thalach, exang, oldpeak, and slope are all numerical qualities. The classes we have in the dataset represent attributes. 0 denotes a female and 1 denotes a male. We identified 135 records against target 0 and 165 records against target 1 in the data set. The dataset is well-balanced. There are 13 attributes and two classes in the dataset.	Because our dataset is categorically based, all of these classifiers work well on categorical data. As a result, all classifiers produce better results. The Extra Tree classifier produces the greatest results of all. The Extra Tree Classifier enabled UCI information has the highest accuracy (90.00%), whereas the LMT, SVM, and NB algorithmic rules have the lowest accuracy (88%), 87%), and 86%), respectively.	conclusion, and based on a review of the literature, we believe that only a competitive landmark has been achieved within the suggestion of a model for patients with cardiovascular disease, and that there is a desire for combination and additional advanced models to increase the accuracies that may help to predict cardiovascular disease more accurately.

		support vector machine, and naïve bays classifiers had 90 percent, 88 percent, 87 percent, and 86 percent accuracy, respectively, according to our experimental data. According to our research, the Extra Tree classifier, which has the highest accuracy, is the best strategy for predicting cardiovascular illness.				
An Dinh, Stacey Miertschin, Amber Young & Somya D. Mohanty: BMC Medical Informatics and Decision Making volume 19, Article	A data-driven approach to predicting diabetes and cardiovascular disease with machine learning	To predict diabetes and cardiovascular illness, we apply supervised machine learning models in this research. Despite the acknowledged link between the	Data-driven techniques that use supervised machine learning algorithms to identify patients with such disorders are being investigated. We construct models	A timeframe of 2007-2014 was employed for the CVD dataset, bringing the total number of accessible variables to 131. Physical activity indicators were included in the	With no laboratory results, the created ensemble model for cardiovascular disease (based on 131 variables) achieved an	We investigated the feature dependence of the models for diabetes and CVD prediction with the goal of building an accurate model depending on a minimal set of

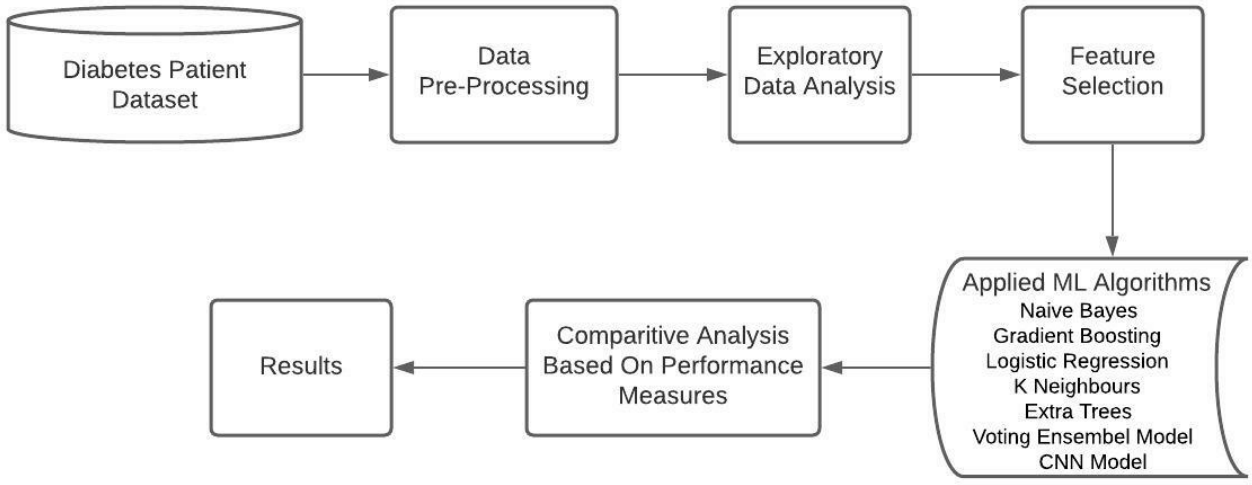
number: 211 (2019)		two diseases, we designed the algorithms to predict CVD and diabetes separately so that a larger spectrum of individuals can benefit. As a result, we are able to find feature commonality between diseases that influence their prognosis. Prediction of prediabetes and undiagnosed diabetes is also taken into account. Multiple models for the prediction of various diseases are trained and tested using the National Health and Nutrition Examination	for cardiovascular, prediabetes, and diabetes detection using the National Health and Nutrition Examination Survey (NHANES) dataset and an extensive search of all accessible feature variables within the data. Multiple machine learning models (logistic regression, support vector machines, random forest, and gradient boosting) were examined on their classification performance using varied timeframes and feature sets for the data (based on laboratory data). The models were then integrated to	dataset, which are key determinants in cardiovascular disease. Each dataset was further divided into two categories: laboratory (which includes test results) and no laboratory (which only includes survey data). Any feature variables in the dataset that were obtained by blood or urine tests were referred to as laboratory results. Recategorization of data into these groups allows for performance monitoring of machine	Area Under Receiver Operating Characteristics (AU-ROC) score of 83.1 percent, and 83.9 percent with laboratory results. The eXtreme Gradient Boost (XGBoost) model achieved an AU-ROC score of 86.2 percent (without laboratory data) and 95.7 percent (with laboratory data) in diabetes classification (based on 123 variables) (with laboratory data). The ensemble model produced the greatest AU-ROC score of 73.7 percent (without laboratory data) for pre-diabetic	available features, i.e. features that did not require significant questioning or testing of patients. The analysis was conducted on the XGBoost ensemble classifier (based on model performance), and features were ranked using an error rate metric. In XGBoost models, feature importance scores are determined by how much the split-point(s) for each feature improves the binary classification error rate, weighted by the number of

		Survey (NHANES) dataset. In addition, a weighted ensemble model is investigated in this study, which combines the outcomes of numerous supervised learning models to improve prediction ability.	create a weighted ensemble model that may take use of the divergent models' performance to improve detection accuracy. The data-learned models utilised information obtained from tree-based models to identify the main variables within the patient data that contributed to the detection of at-risk patients in each of the illnesses groups.	learning models in circumstances where laboratory results are unavailable for patients, making it easier to identify at-risk individuals using only a survey questionnaire.	patients, and XGBoost had the best AU-ROC score of 84.4 percent for laboratory data. 1) waist size, 2) age, 3) self-reported weight, 4) leg length, and 5) sodium intake were the top five predictors among diabetes patients. The models identified age, systolic blood pressure, self-reported weight, occurrence of chest pain, and diastolic blood pressure as major drivers to cardiovascular illnesses.	observations for which the split-point is responsible. The number of misclassified observations divided by the total number of observations is the error rate.

3. Proposed Work

The objective of this work is to choose the best tool for diagnosis and detection of Diabetes in adult women. In this work, a comparative study between various machine learning tools and neural networks were carried out. Furthermore an ensemble model is developed which linearly combines machine learning algorithms to increase the performance of the classification. Finally a Denseneural network model was developed using Keras library and tensor-flow. The performance metric is based on the accuracy rate and the mean square error. The dataset was initially cleaned and pre-processed before training the ML models. The preprocessing techniques involved removing the redundant data values, balancing the data, handling outliers and sampling. Once the data was appropriately preprocessed the exploratory data analysis was done which was analysed to perform the appropriate feature extraction and feature building. ML models were then trained and tested based on the training and testing data. Each model was analysed using various performance measure such as accuracy scores, F1 scores, Recall scores and confusion matrix. A comparative analysis was done of the results and the best model was chosen appropriately.

Architecture



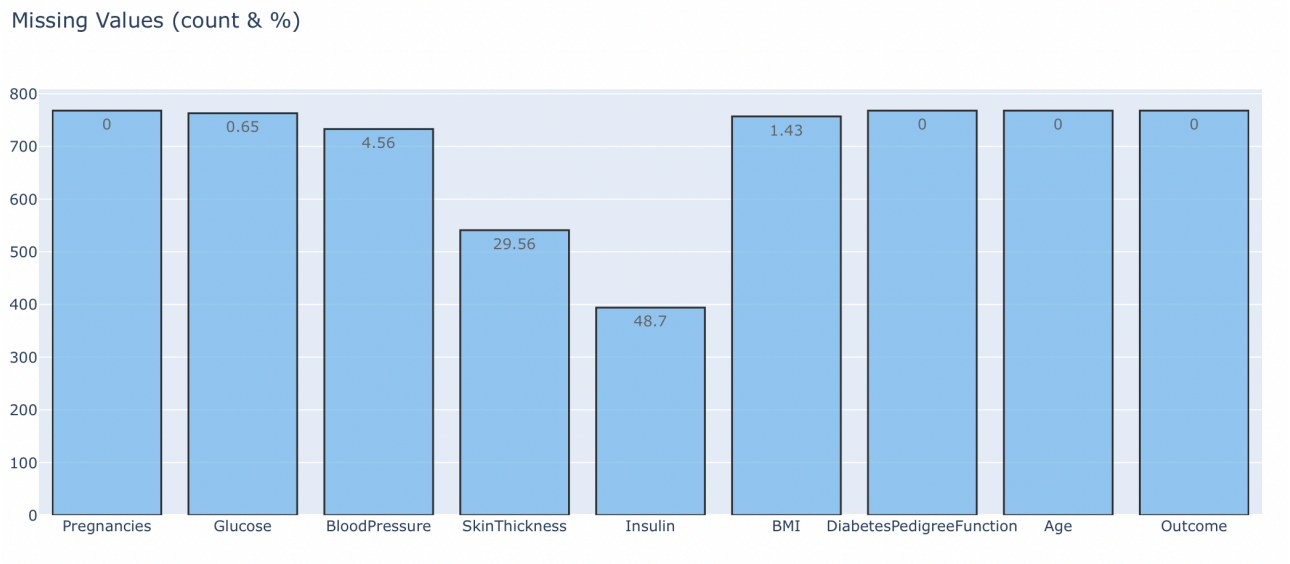
Architecture Diagram

The above diagram represents the architecture of the proposed work. It depicts the sequential order in which the proposed work has been implemented and the performance analysis are done. Initially the dataset was taken from Kaggle website after which data pre-processing was done to clean the data. Then the exploratory data analysis was done and feature extraction was performed

accordingly. Once the data set was ready for predictions different ML models were employed and developed. Each model's performance measures were computed and a comparative analysis was done after which the most optimum model was selected for the prediction of diabetes.

Data Pre Processing

The input dataset contained a lot of redundancies which were needed to death with before performing any analysis. Visualisations were performed to check the number of missing values as shown below. All the redundant data was first converted to NaN values and these values were computed for each of the attributes of the dataset. This revealed the exact number of missing values and to what extent the data was missing. Subsequently appropriate data pre processing technique was employed such as filling the values with the mean or the mode for the respective attribute and were computed and filled for both the outcome that is for diabetic and non diabetic patients.

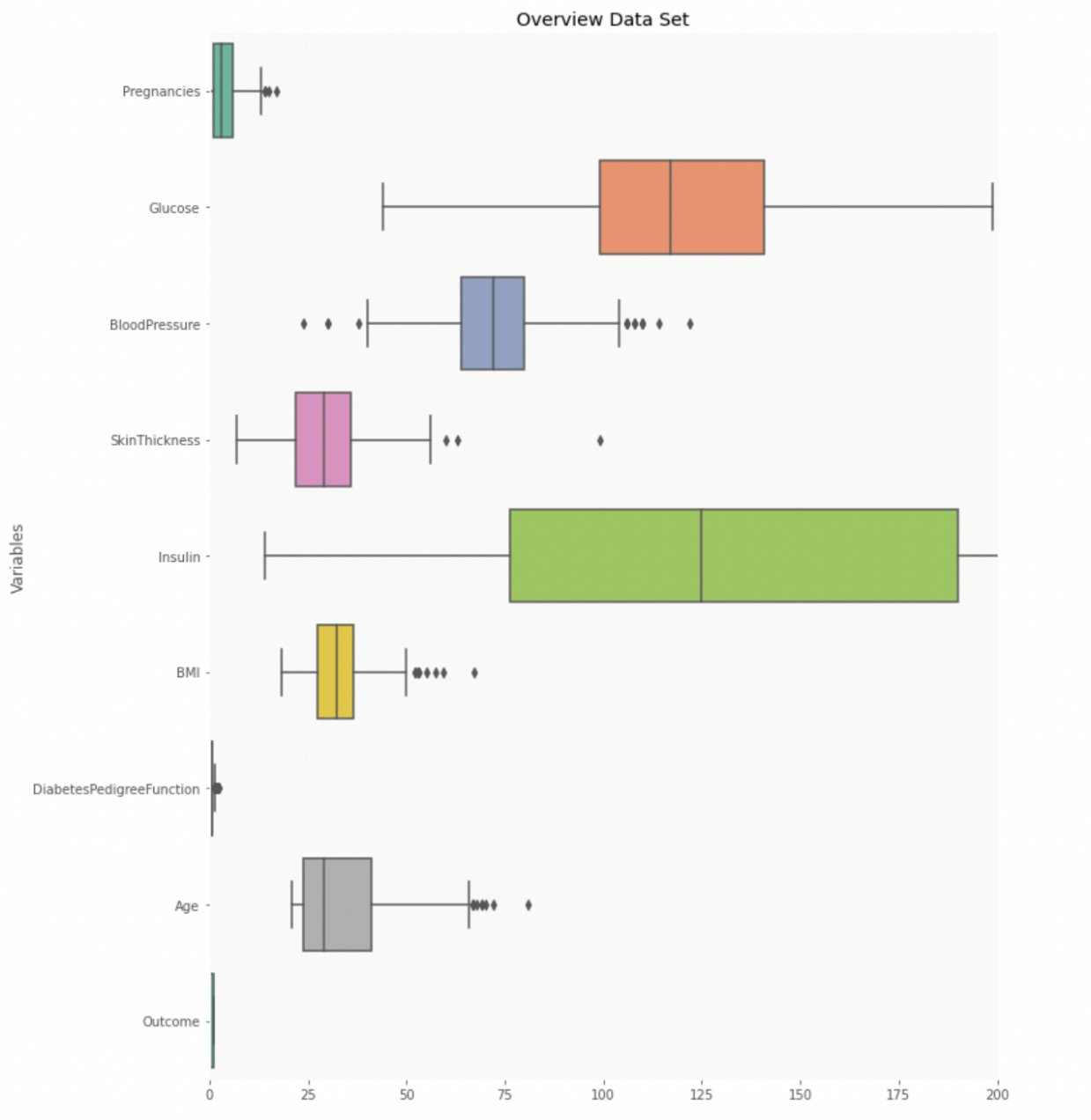


Missing Values

The missing values were visualised as seen above. As we can see Insulin and Skin Thickness attributes had the most number of missing data therefore they would be needed to dealt with appropriately to make sure that the predictions can be as accurate as possible. For dealing with the filling of these missing data the median values were computed for each attribute and for both the outcomes that is diabetic and non diabetic patients. This makes sure that the integrity of the data is maintained and appropriately these median values are filled in and the data set is visualised accordingly. The other attributes such as Blood Pressure, Glucose and BMI were also dealt with by filling in the median values of the respective attributes.

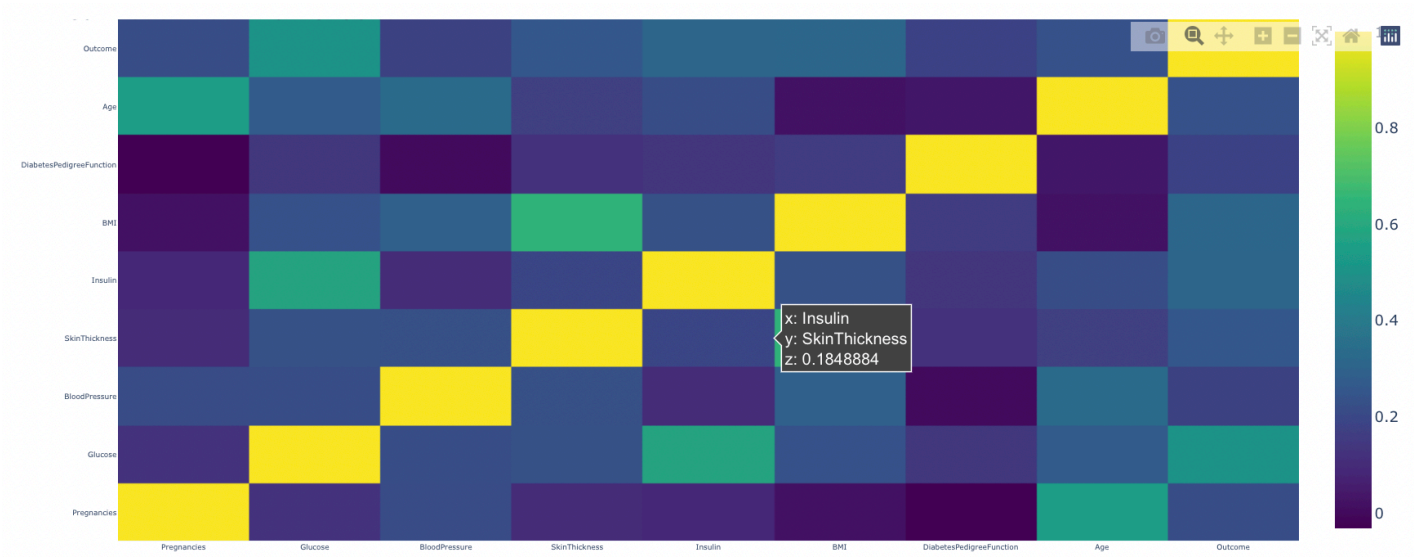
Exploratory Data Analysis

Once the data has been preprocessed accordingly the dataset needed to be understood and visualised to do appropriate feature selections. Exploratory data analysis is done to understand the distribution and correlation of the data. It to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.



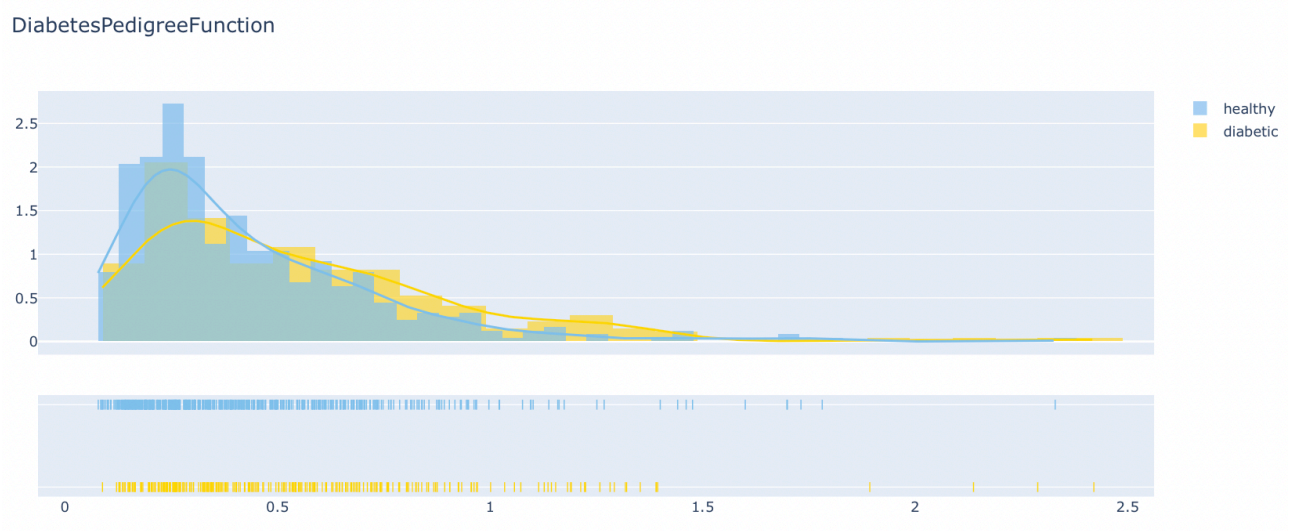
Attribute Data Distribution

The above plot represents the distribution of the attributes, the data distribution shows the values that are being taken up, as we can see the outcome has just two values 0 and 1 where as other attributes such as the Insulin and Glucose have a more varied distribution.



Correlation Matrix

The correlation matrix represents the dependencies of each attribute on one another. This is essential for determining which attributes are more important in influencing the final outcome of the predictions. This shows exactly which attribute is responsible for the higher values of the other. As we can see here Glucose and Insulin are positively correlated which means more is the glucose value more is the Insulin and vice versa. Similarly checking on the attributes which are correlated with the outcome attribute we can see that Glucose is highly influencing in the outcome of the work. At the same time we can see that few of the attributes are not correlated at all insisting that they can be removed to make the predictions faster and more accurate. Therefore the correlation matrix plays an important role in the data engineering techniques that are needed to be performed before training and testing the data.



Diabetes Pedigree Function

The diabetes pedigree function attribute is given in the dataset which basically contains the probability of the person to acquire diabetes based on other factors such as family history, smoking habits etc. This attribute is extremely important in deciding the outcome of the prediction. As we can see in the above graph most healthy patients who are young that is less than 40 years old have a high diabetes pedigree function but are healthy this is because the probability of them acquiring the deceases is higher during those ages due to a variety of factors. However as we move up in ages we see that the pedigree function is much lower however most of them are diabetic. One abnormality was noticed in the plot because for the age group between 20-30 there is a spike in diabetic patients however the number healthy patients exceed. Therefore this plot is important in the feature selection process as this attribute is important it needs to be considered appropriately.

Hardware and Software Specifications Used For Implementation :

The following libraries and softwares were used for the implementation of the proposed work. The work was implemented on Kaggle Notebooks which is an open source platform that allows users to use the dataset available on Kaggle and work simultaneously using the virtual CPU and GPU allocated to each notebook. This allowed us to run the work without the need of any external softwares or hardware equipment. Below is the list of libraries and modules that were used for the proposed work :

- **Kaggle Kernels :** Kaggle Kernels is a free platform to run Jupyter notebooks in the browser. These kernels have high computing power to perform the ML computations efficiently.
- **Kernel Specifications :** 2 CPU Cores. 12GB RAM, Nvidia P100 GPU and 150 GB disk space.
- **Keras Library :** Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.
- **Tensor Flow Library :** pen-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.
- **Matplotlib Library:** Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.
- **Pandas Library :** pandas is a software library written for the Python programming language for data manipulation and analysis.
- **Scikit Learn Library :** It provides a selection of efficient tools for machine learning and statistical modelings including classification, regression, clustering via a consistence interface.

Methodology :

For implementing the proposed work we have used a wide range of machine learning classification algorithms which had unique approaches to the classification problem. The goal of our work was to understand the working of these existing supervised machine learning algorithms and come up with a neural network approach that outperformed these existing algorithms. The novelty of our work lies in the neural network model that we had designed using Denseneural networks which could outperform these ML algorithms. All of the algorithms were trained and tested with the same dataset to ensure authenticity in the comparative analysis that is done. After performing the predictions the performance measures were computed using the confusion matrix. The F1 Scores, Recall Scores, Accuracy Scores, Mean Error Scores were considered to perform the comparative analysis of the algorithms and choose the best approach to this problem.

Below we have first described our neural network approach that we had designed which is the novelty of our work,

Proposed Innovation Carried Out :

Dense Neural Network Model To Predict Diabetes

For our proposed work we have designed a Dense neural network model which could perform the predictions with greater accuracy. To analyse the performance of the model we had utilised the Receiver Operating Curve (ROC) to check for the losses after doing the predictions.

The dataset was normalised which helped the training of neural nets by providing numerical stability. Normalisation changes the values of numeric columns in the dataset to use a common scale, without distorting differences in the ranges of values or losing information therefore provides better overall performance.

A Sequential model was used to build the neural network. A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor.

The first network of our model is a single layer network. We have 8 variables, so we set the input shape to 8 and a single hidden layer with 12 nodes to perform the classification.

We used a Dense layer which is a model of Keras that is the regular deeply connected neural network layer Basically how these dense layers work is that we provide the input data along with the weights that are generated, the layer then performs the numpy dot product of all input and its corresponding weights and adds a bias value which is selected to optimise the model

The rectified linear activation function was used for activating the neural layers which is a

piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero. It is a default activation function for many types of neural networks because a model that uses it is easier to train and often achieves better performance.

The diagram shown below represents the summary of the designed model. As we can see there are two layers designed with a hidden layer consisting of a single neurone to do the final predictions after receiving the output from the subsequent higher layers.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 12)                108
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 13
=================================================================
Total params: 121
Trainable params: 121
Non-trainable params: 0
_________________________________________________________________
```

Nueral Network Model Summary

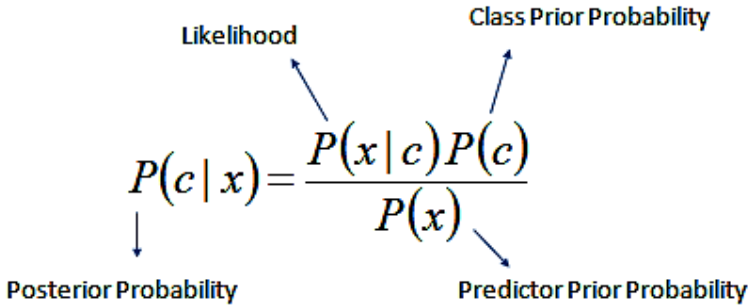
The Param column represents the number of parameters that are trained for each layer. The total number of parameters is shown at the end, which is equal to the number of trainable and non-trainable parameters. In this model, all the layers are trainable and there are 121 total parameters that have been taken. The single hidden layer was sigmoid activated, A weighted sum of inputs is passed through the activation function and this output serves as an input to the next layer. When the activation function for a neuron is a sigmoid function it is a guarantee that the output of this unit will always be between 0 and 1. The weights are created when the model is given some input data The proposed model was trained with about 1500 Epochs. We Compile the model with Optimizer, Loss Function and Metrics Keras provides the SGD class that implements the stochastic gradient descent optimizer with a set learning rate. The learning rate is a hyper-parameter that controls how much to change the model in response to the estimated error each time the model weights are updated. The predictions were then performed accordingly and the performance of our proposed neural network model was analysed based on the ROC accuracy scores and the mean error scores obtained.

Supervised Machine Learning Algorithms Used For Comparative Analysis :

Alternatively we had employed existing supervised machine learning algorithms to compare with the neural network model that we designed. These algorithms were employed with the help of Scikit Learn Library which provides a selection of efficient tools for machine learning and statistical modelings including classification, regression, clustering via a consistence interface. Each of these models was trained and tested with the same dataset and towards the end we had developed an ensemble model which combined the supervised machine learning algorithms to give higher performance.

Naive Bayes Classifier Algorithm

Naive Bayes classifiers are a collection of classification algorithms based on Theorem. It is not a single algorithm but a family of algorithms where all of them Bayes' share a common principle, i.e. every pair of features being classified is independent of each other. In Gaussian Naive Bayes, continuous values associated with each feature are assumed to be distributed according to a Gaussian distribution. A Gaussian distribution is also called Normal distribution.

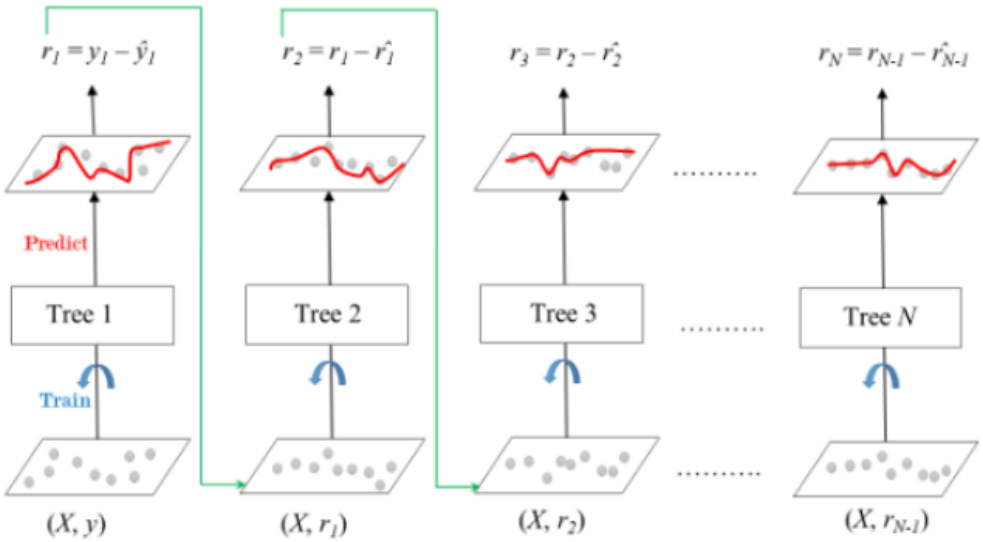


$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Naive Bayes Computation

Gradient Boosting Classifier Algorithm

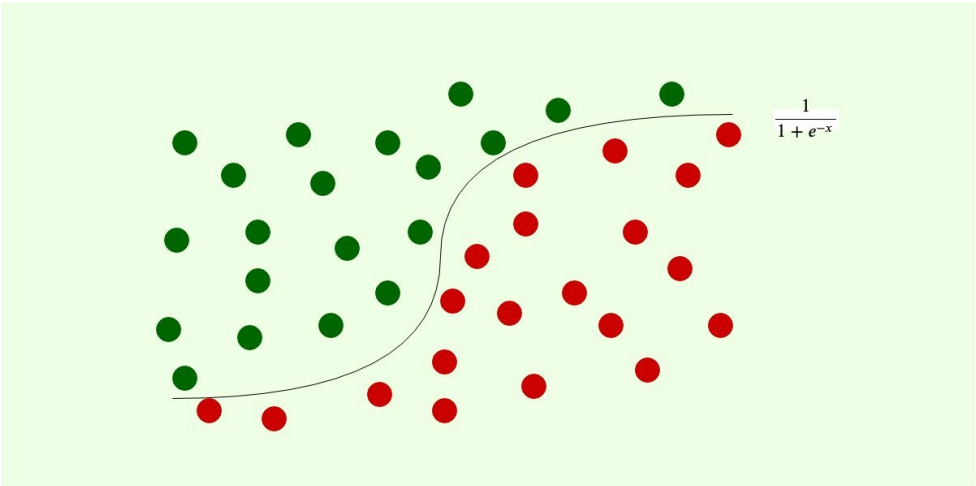
Gradient Boosting is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels. There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees). The below diagram explains how gradient boosted trees are trained for regression problems



Gradient Boosting Classifier Working

Logistic Regression Classifier

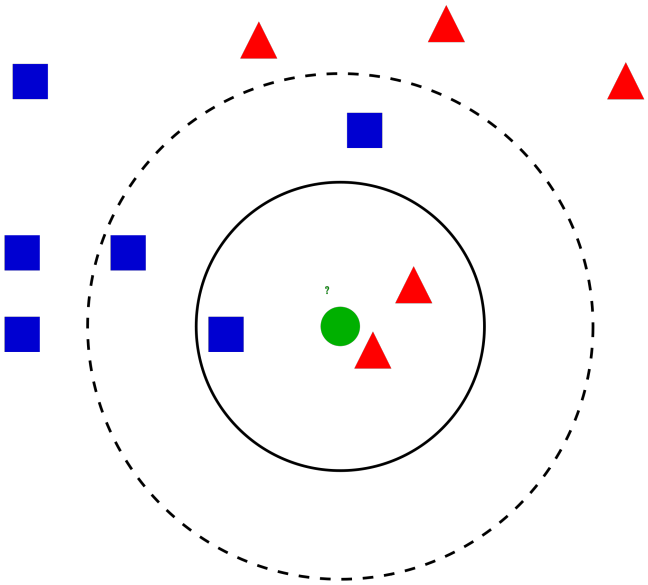
Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable(or output), y, can take only discrete values for a given set of features(or inputs), X. Contrary to popular belief, logistic regression IS a regression model. The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It’s an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.Where e is the base of the natural logarithms (Euler’s number or the EXP()) and value is the actual numerical value that we want to transform.



Logistic Regression Computation

K Nearest Neighbours Classifier

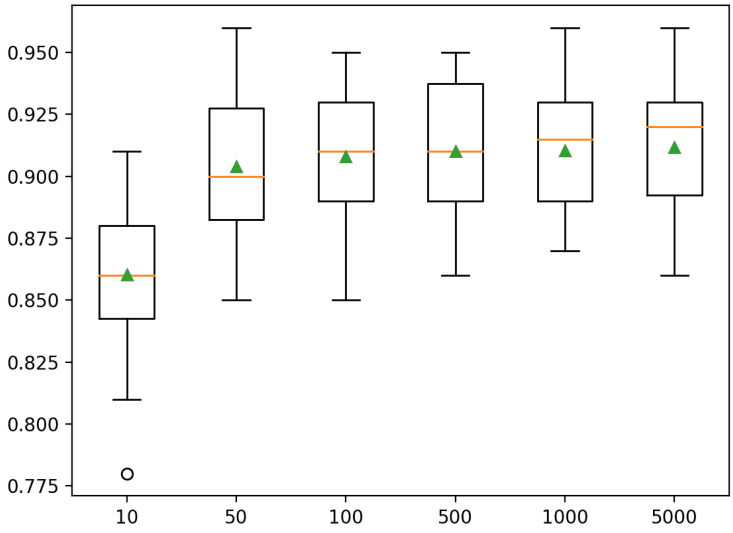
K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. It assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. It stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. It can be used for Regression as well as for Classification but mostly it is used for the Classification problems. It is a non-parametric algorithm, which means it does not make any assumption on underlying data.



KNN Algorithm

Extra Trees Classifier

Extremely Randomised Trees Classifier(Extra Trees Classifier) is a type of ensemble learning technique which aggregates the results of multiple de-correlated decision trees collected in a "forest" to output it's classification result. In concept, it is very similar to a Random Forest Classifier and only differs from it in the manner of construction of the decision trees in the forest. Each Decision Tree in the Extra Trees Forest is constructed from the original training sample. Then, at each test node, Each tree is provided with a random sample of k features from the feature-set from which each decision tree must select the best feature to split the data based on some mathematical criteria (typically the Gini Index). This random sample of features leads to the creation of multiple de- correlated decision trees.

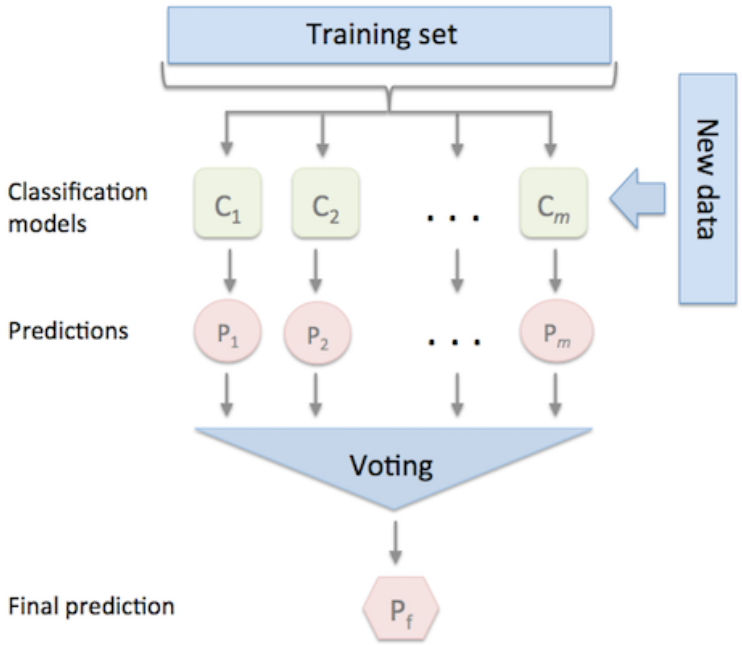


Extra Trees Ensemble Working

Voting Ensemble Model

After analysing the working of the above ML algorithms we developed a unique voting ensemble model. This model is a combination of the ML algorithms that we have discussed above.

A voting ensemble works by combining the predictions from multiple models. It can be used for classification or regression. In the case of regression, this involves calculating the average of the predictions from the models. In the case of classification, the predictions for each label are summed and the label with the majority vote is predicted. For our work we adapted three algorithms namely, Logistic Regression Classifier, Support Vector Machine and Decision Tree Classifier algorithms. This model therefore trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of chosen class as the output.



Voting Ensemble Classifier

4. Dataset Used

This dataset used for our proposed work is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

The dataset has been taken from Kaggle website which allows users to find and publish data sets, explore and build models in a web-based data-science environment. The attributes of the dataset are shown below ,

- Number of times pregnant
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- Diastolic blood pressure (mm Hg)
- Triceps skin fold thickness (mm) 2-Hour serum
- Insulin (mu U/ml)
- Body mass index (weight in kg/(height in m)^2)
- Diabetes pedigree function : Diabetes pedigree function a function which scores likelihood of diabetes based on family history
- Age: Age (years)
- Outcome: Class variable (0 or 1)

The proposed work is taken as a reference from the projects discussed in the literature review that had been documented in the earlier section. However the approach that we had proposed is unique to the work done with this dataset. The dense neural network model that we had developed was unique to solving the diabetes classification problem which on further analysis proved to be a better approach than the conventional work that was done earlier on this model.

Our project is unique with the approach that we had taken to solve the classification problem. The dense neural network and the ensemble model have been newer approaches to do the predictions which out performed the existing works that employed the supervised machine learning algorithms as seen on the literature review. A Sequential model was used to build the neural network. We used a Dense layer which is a model of Keras that is the regular deeply connected neural network layer which had an impressive accuracy of about 93%.

5. Implementation & Results

The work was implanted as described in the proposed architecture above. We have utilised Kaggle kernels to implement our work which allowed us to run the work without the need of any external softwares or hardware equipment. The work was pushed to our GitHub repository as well. Below is the code snippet for the neural network model that we had designed to perform the diabetes classification.

Neural Network Approach

```
def plot_roc(y_test, y_pred, model_name):
    fpr, tpr, thr = roc_curve(y_test, y_pred)
    fig, ax = plt.subplots(figsize=(8, 8))
    ax.plot(fpr, tpr, 'k-')
    ax.plot([0, 1], [0, 1], 'k--', linewidth=.5)  # roc curve for random model
    ax.grid(True)
    ax.set(title='ROC Curve for {} on PIMA diabetes problem'.format(model_name),
           xlim=[-0.01, 1.01], ylim=[-0.01, 1.01])
```

```
normalizer = StandardScaler()
X_train_norm = normalizer.fit_transform(X_train)
X_test_norm = normalizer.transform(X_test)
```

```
model_1 = Sequential([
    Dense(12, input_shape=(8,), activation="relu"),
    Dense(1, activation="sigmoid")
])
model_1.summary()
```

```
from tensorflow.keras.optimizers import SGD
model_1.compile(SGD(lr = .003), "binary_crossentropy", metrics=["accuracy"])
run_hist_1 = model_1.fit(X_train_norm, y_train, validation_data=(X_test_norm, y_test), epochs=1500)
```

```
y_pred_nn_1 = model_1.predict(X_test_norm)
NNScore = roc_auc_score(y_test,y_pred_nn_1)
print('accuracy is {:.3f}%'.format(100*roc_auc_score(y_test,y_pred_nn_1)))
plot_roc(y_test, y_pred_nn_1, 'NN')
```

Neural Network Code Snippet

A Sequential model was used to build the neural network. A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor.

The first network of our model is a single layer network. We have 8 variables, so we set the input shape to 8 and a single hidden layer with 12 nodes to perform the classification.
We used a Dense layer which is a model of Keras that is the regular deeply connected neural network layer Basically how these dense layers work is that we provide the input data along with the weights that are generated, the layer then performs the numpy dot product of all input and its corresponding weights and adds a bias value which is selected to optimise the model
The rectified linear activation function was used for activating the nueral layers which is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero. It is a default activation function for many types of neural networks because a model that uses it is easier to train and often achieves better performance. The single hidden layer was sigmoid activated, A weighted sum of inputs is passed through the activation function and this output serves as an input to the next layer. When the activation function for a neuron is a sigmoid function it is a guarantee that the output of this unit will always be between 0 and 1. The weights are created when the model is given some input data The proposed model was trained with about 1500 Epochs. We Compile the model with Optimizer, Loss Function and Metrics Keras provides the SGD class that implements the stochastic gradient descent optimizer with a set learning rate.The predictions were then performed accordingly and the performance of our proposed neural network model was analysed based on the ROC accuracy scores and the mean error scores obtained.

Github Repository For Source Code :

https://github.com/Jayy109/Diabetes-Prediction---Model-Comparisons/blob/main/ai-project-diabetes.ipynb

Results :

All of the models explained in the proposed work section were trained and tested with the pre processed dataset. The training was done by fitting the train data set with scikit's fit module. After which the predictions were done using the predict module for which the testing data was used. For analysing the performance measures we have used the accuracy scores, F1 scores and Recall scores. Precision factor examines positive predictions, indicating true positives in data set. While recall factor tells us what fraction of all positive samples were correctly predicted as positive by the classifier. It is also known as True Positive Rate.F1 score combines precision and recall into a single measure is then computed using the formula shown below.

$$Precision = \frac{TruePostives}{TruePostives + FalsePostives}$$

$$Recall = \frac{TruePostives}{TruePostives + FalseNegatives}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Performance Measures

Evaluation of these ML algorithms using accuracy of prediction of diabetes, alongside taking other evaluation metrics for mentioning the best and efficient algorithm for prediction. Evaluation metrics used are: Confusion matrix: This matrix compares the predicted values of ML classifier with actual true values F1 score: measures the accuracy model, range is 0 to 1, the value in this range describes the accuracy of the model. F1 score is defined between two aspects which is recall and precision, where recall value is determined by division of positive value (wanted values) by total relevant sample datasets. Precision value is calculated by divining positive values by positives outcome predicted by the ML algorithm classifier.

The results of all the models are shown below and the combative analysis was checked at the end of the results.

Naive Bayes Classifier Algorithm

	precision	recall	f1-score	support
0	0.81	0.83	0.82	343
1	0.69	0.65	0.67	194
accuracy			0.77	537
macro avg	0.75	0.74	0.75	537
weighted avg	0.77	0.77	0.77	537

The Naive Bayes classifier algorithm got an overall accuracy of about 81%, he recall scores were 83% and on an average the F1 scores came out to be around 77% which is satisfactory but not better than the previous works that we had seen during our literature review.

Gradient Descent Classification Algorithm

	precision	recall	f1-score	support
0	0.92	0.94	0.93	157
1	0.86	0.82	0.84	74
accuracy			0.90	231
macro avg	0.89	0.88	0.88	231
weighted avg	0.90	0.90	0.90	231

This supervised algorithm works on the notion that the overall prediction error is minimized when the best potential next model is combined with its past models. From the performance scores seen above we found that the precision for about 92% and the recall scores were also impressive with 94% scores and it had the highest F1 score of about 93%.

Logistic Regression Classifier Algorithm

	precision	recall	f1-score	support
0	0.86	0.80	0.83	343
1	0.69	0.77	0.73	194
accuracy			0.79	537
macro avg	0.77	0.78	0.78	537
weighted avg	0.80	0.79	0.79	537

he model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. From the performance scores seen above we found that the precision for about 86% and the recall scores were also decent with 80% scores and it had a F1 score of about 83%.

K Nearest Neighbours Classifier

	precision	recall	f1-score	support
0	0.86	0.89	0.88	157
1	0.75	0.70	0.73	74
accuracy			0.83	231
macro avg	0.81	0.80	0.80	231
weighted avg	0.83	0.83	0.83	231

This algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. From the performance scores seen above we found that the precision for about 86% and the recall scores were also decent with 89% scores and it had a F1 score of about 88%.

Extra Trees Classifier

	precision	recall	f1-score	support
0	0.92	0.90	0.91	157
1	0.80	0.82	0.81	74
accuracy			0.88	231
macro avg	0.86	0.86	0.86	231
weighted avg	0.88	0.88	0.88	231

In concept this model, it is very similar to a Random Forest Classifier and only differs from it in the manner of construction of the decision trees in the forest. Each Decision Tree in the Extra Trees Forest is constructed from the original training sample. Then, at each test node, Each tree is provided with a random sample of k features from the feature-set from which each decision tree must select the best feature to split the data based on some mathematical criteria. From the performance scores seen above we found that the precision was very impressive about 92% and the recall scores were also goodwith 90% scores and it had a F1 score of about 91%.

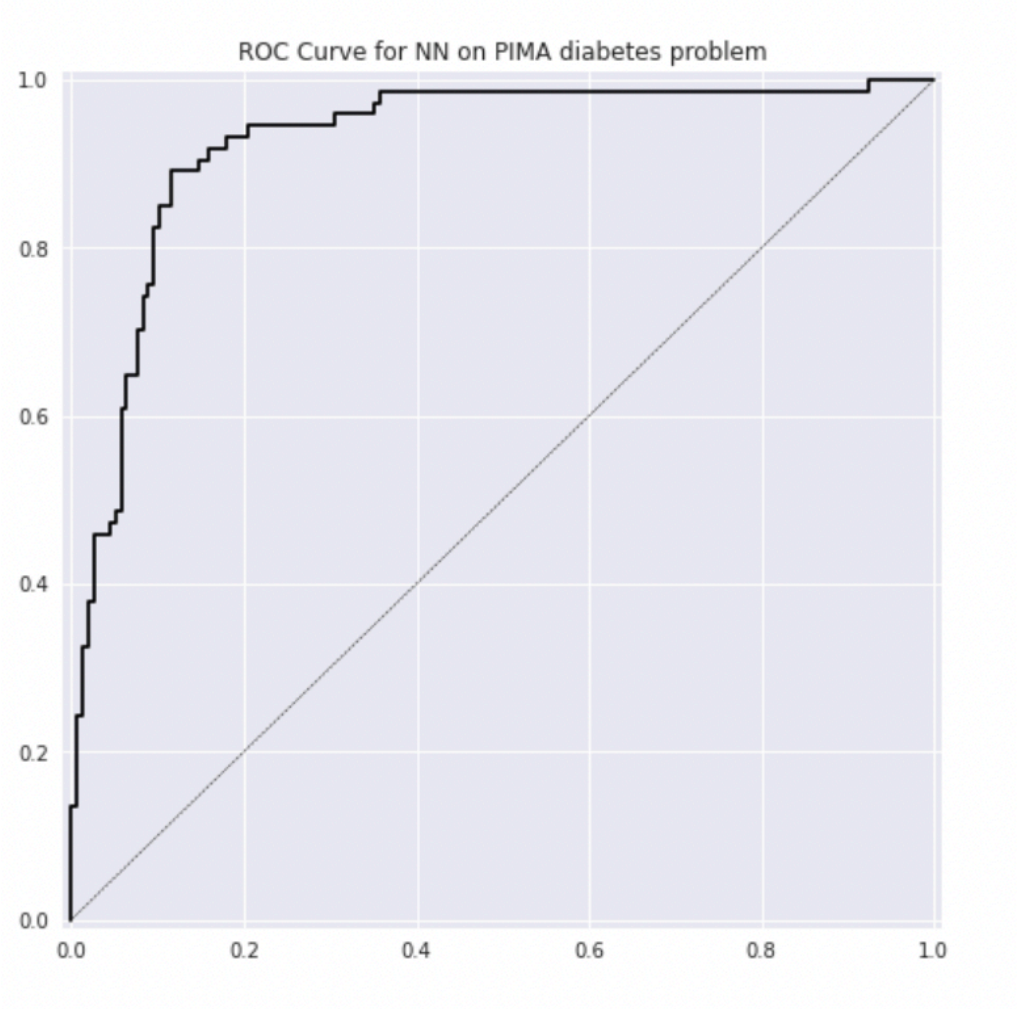
Voting Ensemble Model

	precision	recall	f1-score	support
0	0.90	0.96	0.93	343
1	0.91	0.82	0.86	194
accuracy			0.91	537
macro avg	0.91	0.89	0.90	537
weighted avg	0.91	0.91	0.91	537

voting ensemble works by combining the predictions from multiple models. It can be used for classification or regression. In the case of regression, this involves calculating the average of the predictions from the models. It was seen that this model outperformed some of the generic ML algorithms with 90% accuracy, a very high recall of 96% and an impressive 93% F1 score.

Nueral Network Model

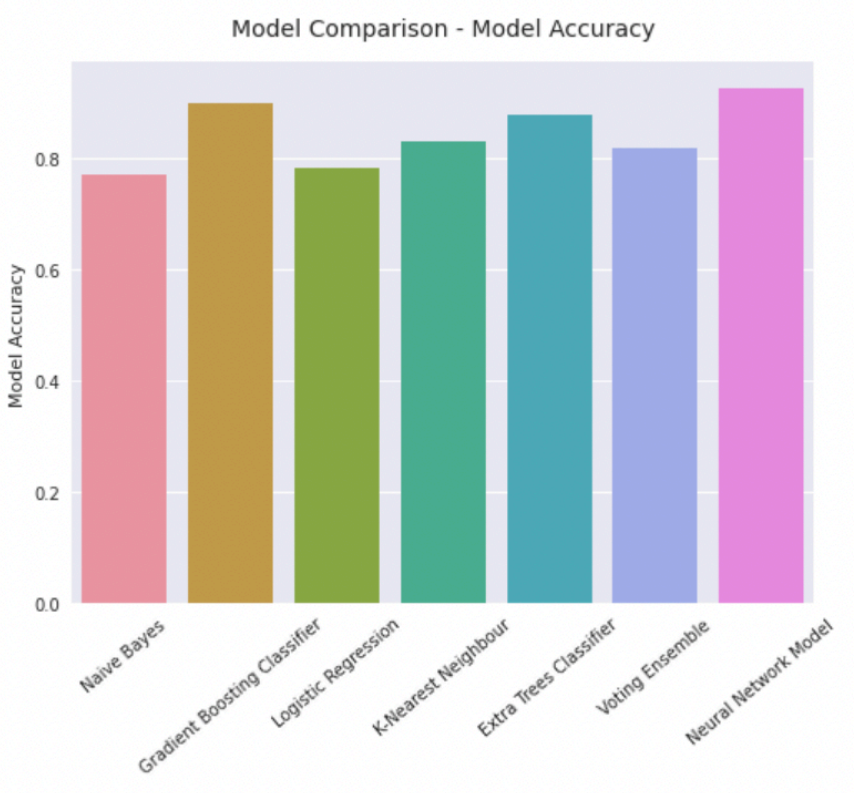
```
accuracy is 92.632%
```



Performance Of Neural Network

The above plot represents the ROC curve of the performance achieved during the training and the testing of the model. A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied.the closer the apex of the curve toward the upper left corner, the greater the discriminatory ability of the test. As we can observe in the ROC curve initially the AUC value was low because of the epochs being untuned but as the number of epochs increased and the threshold value was accurately set we notice the spike in the increase of the AUC score. The precision performance of the neural network model reached up to 93% making this model extremely efficient in comparison to the other models that we had tested. Therefore our proposed work was successful in out performing the existing machine learning algorithms.

Model Comparisons



Model Comparison

Therefore the above plot represents the overall performance of all the models that were trained and tested for our comparative analysis. As we can infer from the comparison plot the neural network model that we had designed using DENSE neural networks performed the best compared to other generic ML algorithms. The gradient boosting classier was close second at about 90% accuracy making it a good alternative approach to perform the predictions of diabetes in the chosen dataset.

6. Conclusion

Diabetes is a long-lasting disease that happens when the pancreas fails to create enough insulin, or when the body cannot use the insulin produced efficiently. In this work, different machine learning techniques and neural networks were used for the diagnosis of diabetes in adult women. A comparison on the accuracy for a particular data set was performed by using various ML algorithms and a neural network model. Our proposed work therefore has been successful at solving the issue of diabetes prediction in adult women based on their heath data with an accuracy of above 90%. The neural network model proposed has an overall accuracy of 93% outperforming the generic supervised ML algorithms.

With this study, it is inferred that out of all models considered and its performance, Nueral Network

is most accurate that gives a good prediction accuracy of 93% percentage and a minimum mean square error.

However the computation time of the proposed model is quite high to be viable enough to deploy it in a application. Moreover we have used a structured dataset from Kaggle but in future unstructured data will also be considered and the used dataset is having only 8 features so if we consider more features for training and testing the model then the accuracy of our model may decrease.

As a future work the same will be implemented using RNN for the prediction of occurrence of other diseases.

7. References

1. Changsheng Zhu, Christian Uwa Idemudia, Wenfang Feng, Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques, Informatics in Medicine Unlocked, Volume 17,2019,100179, ISSN 2352-9148, https://doi.org/10.1016/j.imu.2019.100179
2. Dinh, A., Miertschin, S., Young, A. *et al*. A data-driven approach to predicting diabetes and cardiovascular disease with machine learning. *BMC Med Inform Decis Mak* **19,** 211 (2019). https://doi.org/10.1186/s12911-019-0918-5
3. Mukesh kumari et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5174-5178, https://ijcsit.com/docs/Volume%205/vol5issue04/ijcsit2014050477.pdf
4. Saloni Kumari, Deepika Kumar, Mamta Mittal, An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier, International Journal of Cognitive Computing in Engineering, Volume 2, 2021, Pages 40-46, ISSN 2666-3074, https://doi.org/10.1016/j.ijcce.2021.01.001.
5. Rahman Shaque, Arif Mehmood, Saleem ullah, Gyu Sang Choi, Cardiovascular Disease Prediction System Using Extra Trees Classier, September 16th, 2019 https://doi.org/10.21203/rs.2.14454/v1
6. Kaggle Website : https://www.kaggle.com/learn
7. Health data : https://www.idf.org/type-2-diabetes-risk-assessment/
8. Nueral Network Theory : https://towardsdatascience.com/classification-using-neural-networks-b8e98f3a904f
9. Machine Learning Library : https://scikit-learn.org/stable/