BioGecko Vol 12 Issue 03 2023

ISSN NO: 2230-5807

A Machine Learning Approach for Prediction of Diabeties for Early Prevention

Praveen Kumar Misra[1], Anuradha Misra[2*], Sumit Pal[3]

[1]Assistant Professor& Co-ordinator, Department of Mathematics and Statistics, Dr.Shakuntala Misra National Rehabilitation University, Lucknow, India
ORCID: 000-0002-3539-2994
[2]Assistant Professor, Department of Computer Science & Engineering, Amity School of engineering and Technology, Amity University, Lucknow Campus, India, [*]*
ORCID: 0000-0002-3790-8798
[3]P.G. Student, Department of Mathematics and Statistics, Dr.ShakuntalaMisra National Rehabilitation University, Lucknow, India

Corresponding Email:[2]amisra@lko.amity.edu*
Email: [1]praveenkumarmisra@gmail.com, [3]spal22228@gmail.com

Abstract
Diabetes is a common, chronic disease. Prediction of diabetes at an early stage can lead to improved treatment. Datamining techniques are widely used for prediction of disease at an early stage. In this research paper, diabetes is predicted using significant attributes, and the relationship of the different attributes is also characterized. Various tools are used to determine significant attribute selection, and for clustering, prediction, and association rule mining for diabetes. Significant attributes selection was done via the principal component analysis method. Our findings indicate a strong association of diabetes with body mass index (BMI) and with glucose level, which was extracted via the A priori method. K-nearest neighbor (KNN), support vector machine (SVM) and K-means clustering techniques were implemented for the prediction of diabetes. The KNN technique provided a best accuracy of 77%, and may be useful to assist medical professionals with treatment decisions.

1. INTRODUCTION

The disease or condition which is continual or whose effects are permanent is a chronic condition. This type of diseases a effect quality of life, which is the major adverse effect. Diabetes is one of the mostacute diseases, and is present worldwide. A major reason of deaths in adults across the globe includes this chronic condition. Chronic conditions are also cost associated. A major portion of budget is spent on chronic diseases by governments and individuals.

The world wide statistics for diabetes in the year 2013 revealed around 382 million individuals had this ailment around theworld. It was the fifth leading cause of death in women and eight leading cause of death for both sexes in 2012. Higher income countries have a high probability of diabetes. In 2017, approximately 451 million adults were treated with diabetes worldwide. It is projected that in 2045, almost 693million patients with diabetes will exist around the globe and half of the population will be undiagnosed. In addition, 850million USD were spent on patients with diabetes in 2017. Research on biological data is limited but with the passage of

time enables computational and statistical models to be used for analysis. Data mining is the process of extractingfrom data and can be utilized to create a decision making process withefficiency in the medical domain. Several data mining techniques have been utilized for disease prediction as well as for knowledge discovery from biomedicaldata.

Diagnosis of diabetes is considered a challenging problem for quantitative research. Some parameters likeA1C, fructosamine, white blood cellcount, fibrinogen and hematological indices were shown to be in effective due to some limitations. Different research studies used these parameters for the diagnosis of diabetes. A few treatments have thought to raise A1C including chronic ingestion of liquor, salicylates and narcotics. Ingestion of vitamin C may elevate A1cwhen estimated by electrophores is but levels may appear to diminish when estimated by chromatography. Most studies have suggested that a higher white blood cell count is due to chronic inflammation during hypertension. A family history of diabetes has not been associated with BMI and insulin. However, an increased BMI is not always associated with abdominal obesity. A single parameter is not very effective to accurately diagnose diabetes and may be misleading in the decision making process. There is a need to combine different parameters to effectively predict diabetes at an early stage. Several existing techniques have not provided effective results when different parameters were used for prediction of diabetes. In our study, diabetes is predicted with the assistance of significant attributes, and the association of the differing attributes. We examined the diagnosis of diabetes using Logistics Regression, Naïve Bayes Classifier, K-Nearest Neighbour and Support Vector Machine.

Data Source

Pima Indian Diabetes Database is afamiliar and commonly used data set for the prediction of diabetes. This data set consists of 768 rows and 9 columns. The attributes included in the column are glucose, pregnancies, skin thickness, blood pressure, BMI, insulin, age, and outcomes. The outcome variable predicts whether the patient is diabetic, positive or diabetic-negative. Pandas function is utilized to read. CSV file where the data set file is in excel format.

The data set used in this study, is originally taken from the National Institute of Diabetes and Digestive and Kidney Diseases. The main Objective of using this dataset was to predict through diagnosis whether a patient has diabetes, based on certain diagnostic measurements included in the dataset.

Data Description

Table 1

Variable Name	Description
GLUCOSE LEVEL	A blood sugar level **less than 140 mg/dL (7.8 mmol/L)** is normal.
BLOOD PRESSURE	BLOOD PRESSURE(IN 120/80 mm/Hg)
PREGNANCIES	no. of pregnant women in the dataset.
SKIN THICKNESS	The mean skin thickness of **males ranged from 0.6 mm to**

	3.30 mm and in females it ranged from 1.30 mm to 3.10 mm
BMI	**BMI = weight/height²**
DIABETIC PREDICTION FUNCTION	Body mass index (weight in kg/(height)2 in m) (BMI or x6), 7. Diabetes pedigree function (**DPF or x7**), 8. Age (years) (AGE or x8), and 9. Class variable (non-diabetes = 0 or diabetes =1)(CV or x9).
AGE	Patient Age(in years)
OUTCOME	Target column(1=yes; 0= No)

2. METHADOLOGY

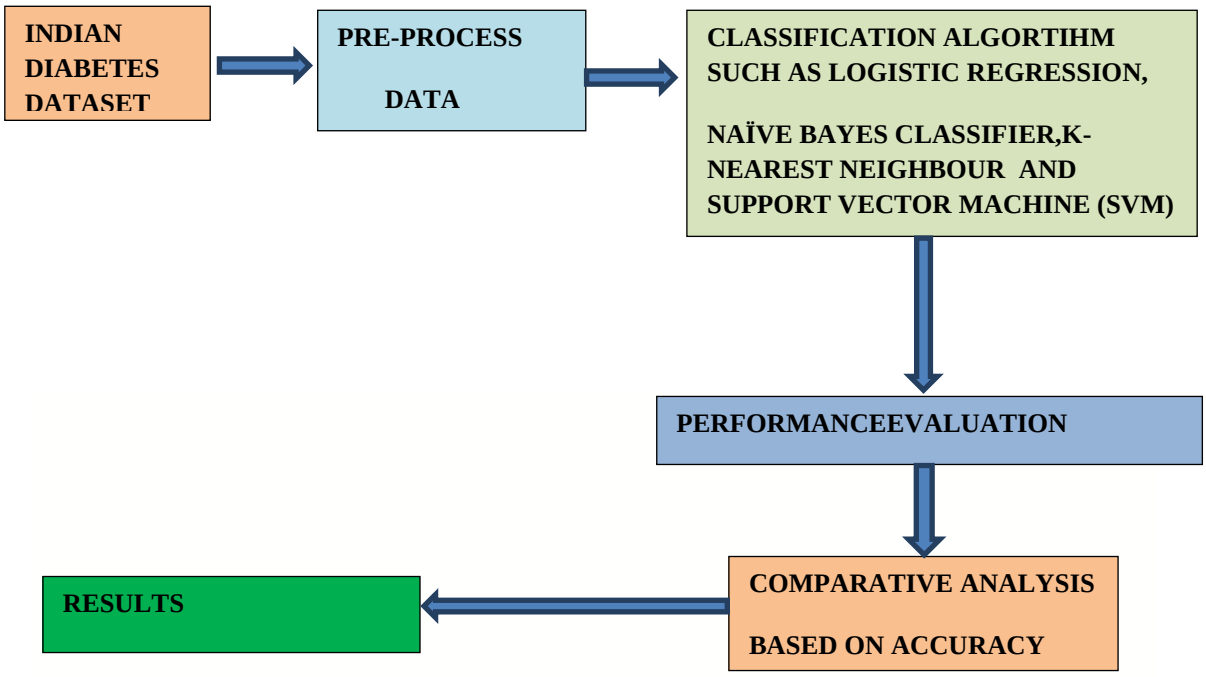


Fig. 1

2.1 Data Visualization

Data visualization helps to understand the data better by putting it in a visual form. In this phase, data are represented in the form of bar chart. The analysis reveals the percentage of people affected by diabetes diseases. It also displays the information of the dataset such as age, blood pressure, pregnancies, and glucose. Apart from that, it predicts how many people are affected by diabetes from 768. For displaying output, the graphical representation functions such as plot axis, pyplot, and several others have been used.

2.2Training Dataset

This section includes the removal of outliers and standardizing the data. The processed data have been used for creating a model. The data should be pre-processed and arranged properly before applying classifiers to the data index. These data should be handled carefully before connecting.

2.3 Scaling the Dataset

In this phase, inconsistent data are handled and removed to obtain more precise and accurate results. This dataset contains missing values. Few selected attributes such as blood pressure, skin thickness, glucose level, and BMI are assigned with missing values because these parameters cannot have null values. Then, we normalized all values by scaling the dataset.

2.4 Checking the Accuracy of the Dataset

Classification Accuracy- It is the ratio of number of correct predictions to the total number of input samples.

$$\textbf{Accuracy} = \frac{\textbf{Number of correct predictions}}{\textbf{Total number of predictions made}}$$

Logistics Regression algorithm

- Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
- Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
- Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas logistic regression is used for solving the classification problems.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Naïve Bayes Classifier

Naive Bayes classifier is a series of a simple probability classifier based on the use of Bayes theorem under the assumption of strong (naïve) independence between features. The classifier model assigns class labels represented by feature value to problem instances, and class labels are taken from a limited set. For the given item to be classified, the probability of each category appearing under the condition of the occurrence of the item is solved, whichever is the largest, and the category to be classified is considered to be. This prediction of the most likely class by probability is

suitable for diabetic prediction. The specific classification formulas are shown in equation 1 to 4. Where **xp** represents people who are at risk of diabetes,**Xn** represents people who are not at risk of diabetes, and **X** is the dataset.

$$P(X|X_p) = \prod_{d=1}^{D} P(x_d|x_p) = P(x_1|x_p)P(x_2|x_p)\dots P(x_D|x_p) \quad \dots\dots 1$$

$$P(X|X_n) = \prod_{d=1}^{D} P(x_d|x_n) = P(x_1|x_n)P(x_2|x_n)\dots P(x_D|x_n) \quad \dots\dots\dots\dots\dots 2$$

$$P(x_d|x_p) = \frac{Total\,(x_d|x_p)}{Total\ \ x_p} \quad \dots\dots\dots\dots\dots 3$$

$$P(x_d|x_n) = \frac{Total(x_d|x_n)}{Total\ \ x_n} \quad \dots\dots\dots\dots\dots 4$$

Here D is the attribute with dimension D.

K- Nearest Neighbour(KNN)

KNN is one of the MLsupervised learning techniques. It is mostly applied in classification problems. KNN is used to classify objects depending on the closest measure/distance, i.e., the distance between the object and all objects of training data. Based on K-neighbors, the item is classified. Positive integer K is defined before executing the algorithm.

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm 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.
- K-NN 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.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Support Vector Machine (Svm)

SVMisageneralizedlinearclassifierthatperformsbinaryclassificationofdataaccordingtos upervisedlearning. Its decision boundary is the maximum-margin hyperplane for solving learning samples.SVM uses the hinge loss function to calculate empirical risk and adds a regularization term to thesolution system to optimize structural risk. It is a classifier with sparsity and robustness. SVM canperform non-linear classification

through the kernel method, which is one of the common kernellearningmethods.

Data Anaylsis

Data Analysis is the process of systematically applying statistical and/or logical techniques to describe and illustrate, condense and recap, and evaluate data. It has been performed on googlecolab. First the data set has been imported into the software, then the analysis has been performed.

Data Visualization

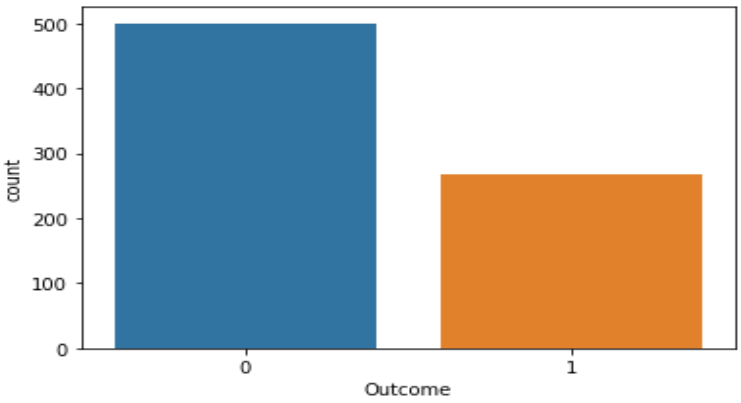


Fig 5

We observe the count for person who has diabetic is 500 and the person who doesn't has diabetic is 268.

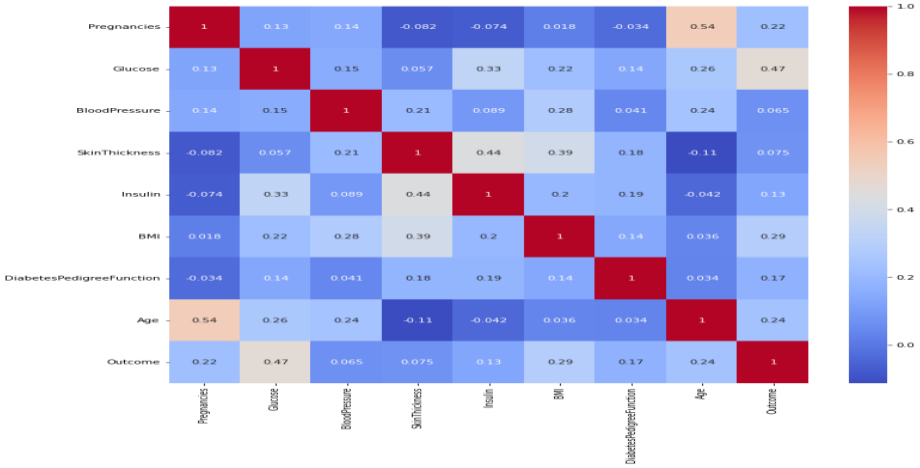
```
[ ]  data['Outcome'].value_counts()

     0    500
     1    268
     Name: Outcome, dtype: int64
```

We observe the count for person who has diabetic is 500 and the person who doesn't has diabetic is 268.

Correlation among the attributes

- The correlation matrix in machine learning is used for feature selection. It represents dependency between various attributes.

Fig 6



Co-relation matrix

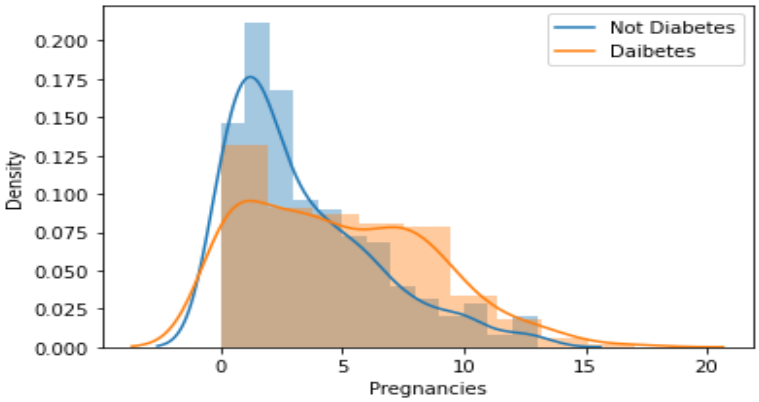
Co-relation matrix interpretation

- The magnitude 1 indicatingcorrelation between same attributes.
- Like wise the darker the colour will be stronger will be the relationa whereas fader the colour will be loosely the relation will be
- Age is strongly related with the glucose level, pregnancies, whereas it is not having any relation with BMI achieved.
- Similarly, resting glucose level, is also related with the blood pressure level and oldpeak.
- Maximum blood pressure is not having any link with old peak.

Distribution Plot

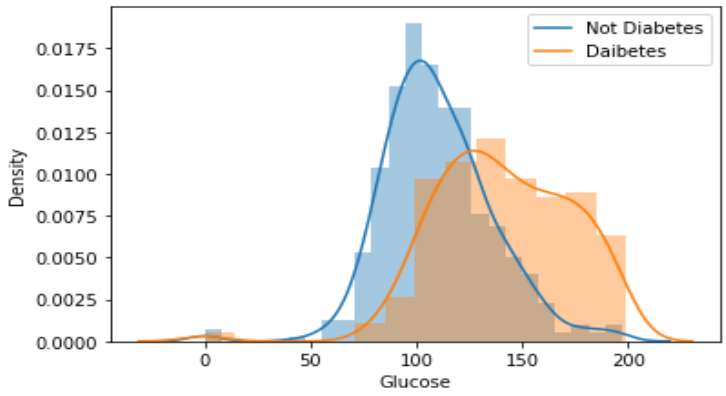
```
[ ]  sns.distplot(data.Pregnancies[data.Outcome==0])
     sns.distplot(data.Pregnancies[data.Outcome==1])
     plt.legend(['Not Diabetes','Daibetes'])
     plt.show()
```

The above function represents the person has diabetic or not.



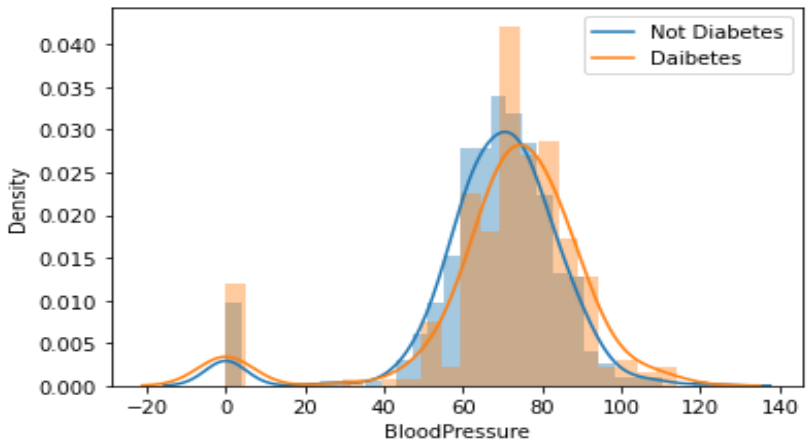
- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic.

```
sns.distplot(data.Glucose[data.Outcome==0])
sns.distplot(data.Glucose[data.Outcome==1])
plt.legend(['Not Diabetes','Daibetes'])
plt.show()
```



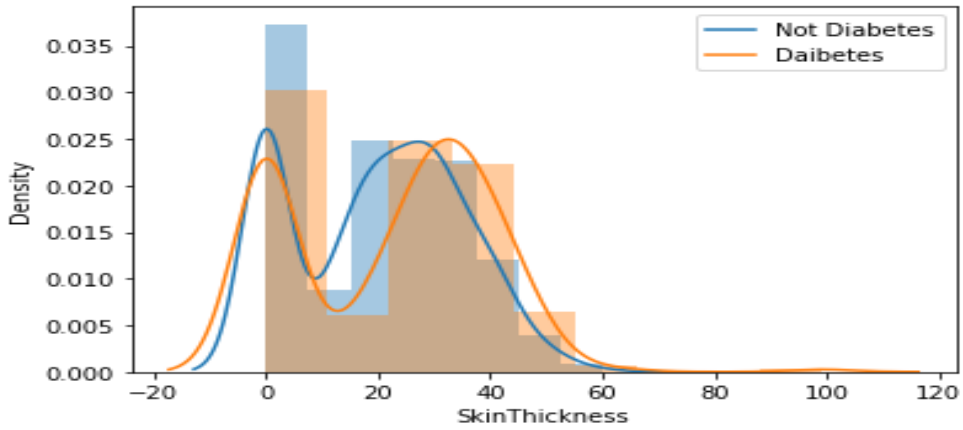
- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to glucose level.

```
sns.distplot(data.BloodPressure[data.Outcome==0])
sns.distplot(data.BloodPressure[data.Outcome==1])
plt.legend(['Not Diabetes','Daibetes'])
plt.show()
```



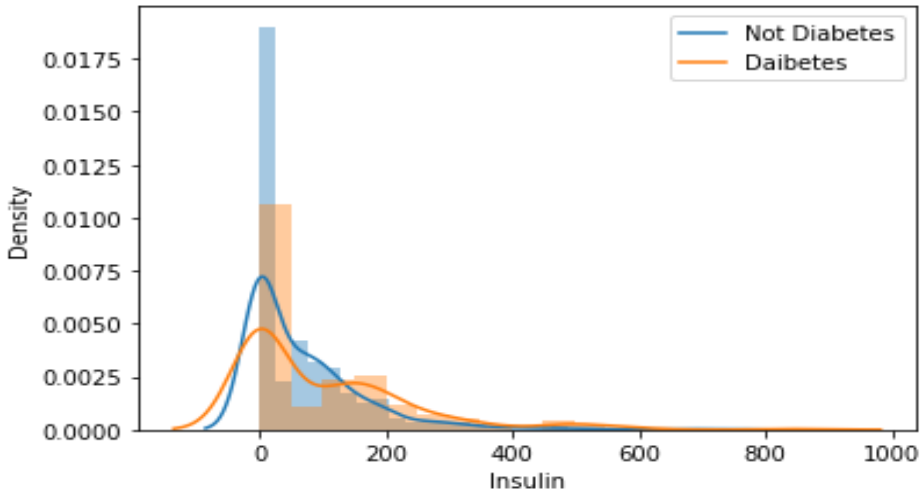
- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to blood pressure level.

```
[ ]  sns.distplot(data.SkinThickness[data.Outcome==0])
     sns.distplot(data.SkinThickness[data.Outcome==1])
     plt.legend(['Not Diabetes','Daibetes'])
     plt.show()
```



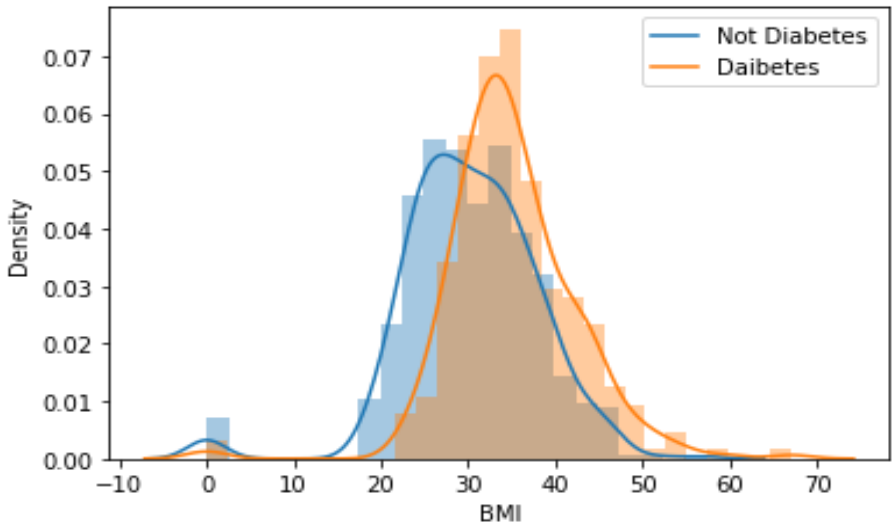
- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to skin thickness level.

```
sns.distplot(data.Insulin[data.Outcome==0])
sns.distplot(data.Insulin[data.Outcome==1])
plt.legend(['Not Diabetes','Daibetes'])
plt.show()
```



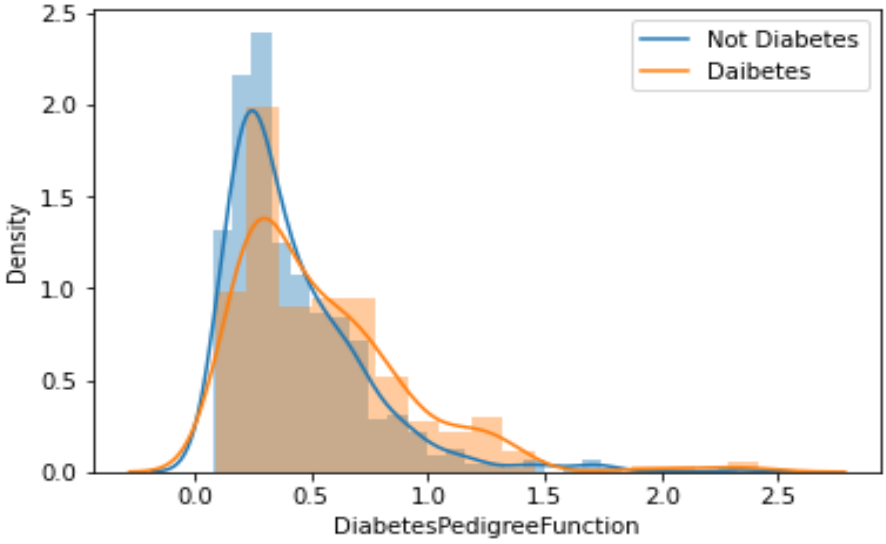
- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to insulin level.

```
sns.distplot(data.BMI[data.Outcome==0])
sns.distplot(data.BMI[data.Outcome==1])
plt.legend(['Not Diabetes','Daibetes'])
plt.show()
```



- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to BMI level.

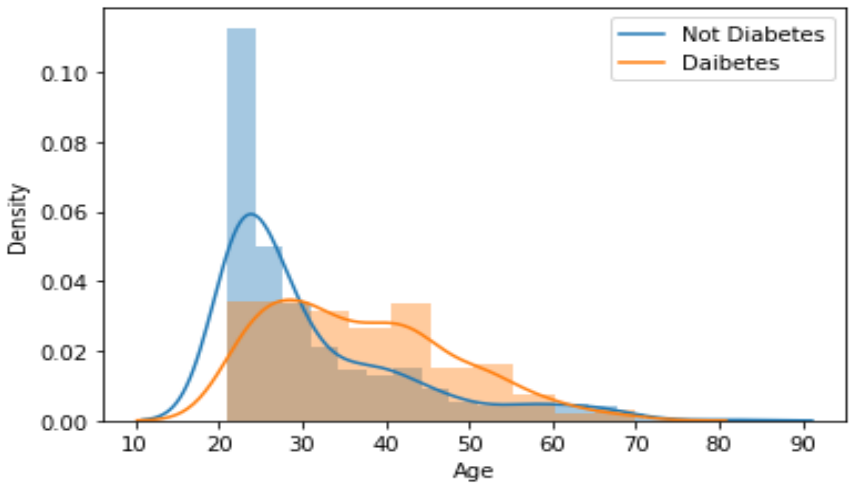
```
sns.distplot(data.DiabetesPedigreeFunction[data.Outcome==0])
sns.distplot(data.DiabetesPedigreeFunction[data.Outcome==1])
plt.legend(['Not Diabetes','Daibetes'])
plt.show()
```



- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to diabetic prediction

function level.

```
[ ]  sns.distplot(data.Age[data.Outcome==0])
     sns.distplot(data.Age[data.Outcome==1])
     plt.legend(['Not Diabetes','Daibetes'])
     plt.show()
```



- The above distribution plot shows the orange line represent the person is diabetic and blue line shows the person is not diabetic with respect to age of a person.

Pairplot For The All 8 Variable

```
sns.pairplot(data=data,hue='Outcome',diag_kind='kde')
plt.show()
```

Fig 8

The above pair plot shows the all 8 variable if person is diabetic or not. Where orange indicate the outcome values.

***Logistic Regression**

Here, the logistic regression is a regression model where the dependent variable is categorical.

The prediction will be 0 or 1, Yes or No

Pima daibetes dataset contain all numerical values

```
ip=data.drop(['Outcome'],axis=1)
op=data['Outcome']
```

***Train and Test the Dataset**

here we train the dataset for the model

.

```
[ ] from sklearn.model_selection import train_test_split
    xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.4)
```

***Scaling the dataset**

here we scaling the dataset which indicate that 1 is person is diabetic.

```
ip=data.drop(['Outcome'],axis=1)
op=data['Outcome']
```

*** Train and Test the Dataset**

- Here we train the dataset for classification model used in the prediction function.

```
from sklearn.model_selection import train_test_split
xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.4)
```

```
[ ] from sklearn.preprocessing import StandardScaler
    sc=StandardScaler()
    sc.fit(xtr)
    xtr=sc.transform(xtr)
    xts=sc.transform(xts)
```

```
[ ] from sklearn.linear_model import LogisticRegression
    alg=LogisticRegression()
```

```
[ ] #train the algorithm with the training data
    alg.fit(xtr,ytr)
    yp=alg.predict(xts)
```

***Checking the accuracy of the model**

```
from sklearn import metrics
cm=metrics.confusion_matrix(yts,yp)
print(cm)
```

```
[[184  31]
 [ 39  54]]
```

```
[ ] accuracy=metrics.accuracy_score(yts,yp)
    print(accuracy)
```

```
0.7727272727272727
```

```
[ ] precission=metrics.precision_score(yts,yp)
    print(precission)
```

```
0.6352941176470588
```

```
[ ] recall=metrics.recall_score(yts,yp)
    print(recall)
```

```
0.5806451612903226
```

- Here we check the accuracy of the model which gives 72% accuracy in this model

***Naive Bayes classifier**

It is easy and fast to predict class of test data set. It also perform well in multi class prediction.

```
from sklearn.model_selection import train_test_split
xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.1)
```

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(xtr)
xtr=sc.transform(xtr)
xts=sc.transform(xts)
```

```
from sklearn.naive_bayes import GaussianNB
GNB=GaussianNB()
GNB.fit(xtr,ytr)
yp=GNB.predict(xts)
```

```
from sklearn import metrics
cm=metrics.confusion_matrix(yts,yp)
print(cm)
```

```
[[42  6]
 [11 18]]
```

```
accuracy=metrics.accuracy_score(yts,yp)
print(accuracy)
```

```
0.7792207792207793
```

```
recall=metrics.recall_score(yts,yp)
print(recall)
```

```
0.6206896551724138
```

- Here we check the accuracy of naïve bayes classifier model which gives 77% accuracy in this model.

***K-Nearest Neighbour**

K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems.
KNN algorithm the nearest distance is calculated.

```
from sklearn.neighbors import KNeighborsClassifier
neighbors=np.arange(1,9)
train_accuracy=np.empty(len(neighbors))
test_accuracy=np.empty(len(neighbors))
for i,k in enumerate(neighbors):
    knn=KNeighborsClassifier(n_neighbors=k)
    knn.fit(xtr,ytr)
```

```
    train_accuracy[i]=knn.score(xtr,ytr)
    test_accuracy[i]=knn.score(xts,yts)
plt.xlabel('neighbors of number')
plt.ylabel('accuracy')
plt.title('k-NN Varying number of neighbors')
plt.plot(neighbors, test_accuracy, label='Testing Accuracy')
plt.plot(neighbors, train_accuracy, label='Training accuracy')
plt.legend()
plt.show()
```

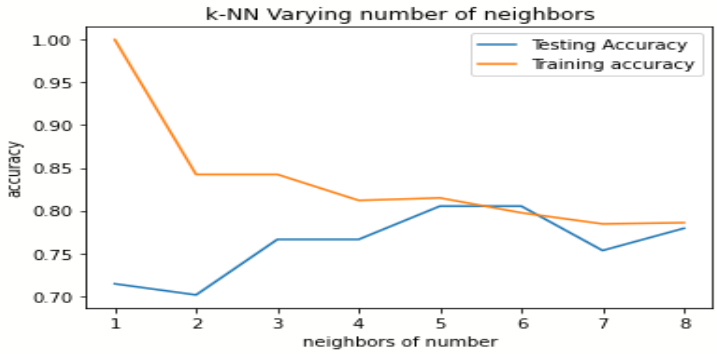


Fig 9

```
from sklearn.model_selection import train_test_split
xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.1)
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(xtr)
xtr=sc.transform(xtr)
xts=sc.transform(xts)
knn=KNeighborsClassifier(n_neighbors=3)
knn.fit(xtr,ytr)
yp=knn.predict(xts)
from sklearn import metrics
cm=metrics.confusion_matrix(yts,yp)
print(cm)
```

```
[[38  9]
 [19 11]]
```

```
accuracy=metrics.accuracy_score(yts,yp)
print(accuracy)
```

```
0.6363636363636364
```

```
recall = metrics.recall_score(yts,yp,average='macro')
print(recall)
```

```
0.5875886524822694
```

- Here we check the accuracy of KNN model which gives 77% accuracy in this model.

***Support Vector Machines-SVM**

Support vector machines (SVMs) are powerful yet flexible supervised machine learning algorithms which are used both for classification and regression.

```
from sklearn.model_selection import train_test_split
xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.3)
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(xtr)
xtr=sc.transform(xtr)
xts=sc.transform(xts)
from sklearn import svm
alg=svm.SVC(C=30,gamma=0.03)
```

***Train the algorithm with training data**

```
alg.fit(xtr,ytr)
yp=alg.predict(xts)
from sklearn import metrics
cm=metrics.confusion_matrix(yts,yp)
print(cm)
```

```
[[124  18]
 [ 42  47]]
```

```
from sklearn import metrics
accuracy=metrics.accuracy_score(yts,yp)
print(accuracy)
```

```
0.7402597402597403
```

```
recall = metrics.recall_score(yts,yp)
print(recall)
```

```
0.5280898876404494
```

- Here we check the accuracy of SVM model which gives 74% accuracy in this

model.

***Decision Tree**
Decision tree is a type of supervised learning algorithm (having a predefined target variable) that is mostly used in classification problems. It works for both categorical and continuous input and output variables.

***Spliting training and testing data**

```
from sklearn.model_selection import train_test_split
xtr,xts,ytr,yts=train_test_split(ip,op,test_size=0.2)
```

***Standard scalar transform**

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(xtr)
xtr=sc.transform(xtr)
xts=sc.transform(xts)
```

***Model training**

```
from sklearn.tree import DecisionTreeClassifier
dtc=DecisionTreeClassifier(criterion='gini')
```

***Train Decision Tree Classifer**

```
dtc=dtc.fit(xtr,ytr)
```

***Predict the response for test dataset**

```
y_pred=dtc.predict(xts)
from sklearn import metrics
cm=metrics.confusion_matrix(yts,y_pred)
print(cm)
```

```
[[80 22]
 [20 32]]
```

```
accuracy=metrics.accuracy_score(yts,y_pred)
print(accuracy)
```

```
0.7272727272727273
```

```
recall=metrics.recall_score(yts,y_pred)
print(recall)
```

```
0.6153846153846154
```

- Here we check the accuracy of all model with decision tree which gives which model is best fitted for prediction for diabetes then the accuracy is 72% in this model.

3. CONCLUSION

Diabetes is a chronic disease that continues to be a significant and global concerns in ceita effects the entire population's health. It is a metabolic disorder that leads to high blood sugar levels and many other problems such as stroke, kidney failure, and heart

and nerve problems. Several researchers have attempted to construct an accurate diabetes prediction model over the years. However, this subject still faces significant open research issues due to a lack of appropriate data sets and prediction approaches, which pushes researchers to use big data analytics and machine learning based methods. Applying four different machine learning methods, the research tries to overcome the problems and investigate healthcare predictive analytics. The study's primary goal was to see how big data analytics and machine learning-based techniques may be used in diabetes. The examination of the results show that the suggested ML-based framework may achieve a score of 86. Health experts and other stakeholders are working to develop categorization models that will aid in the prediction of diabetes and the formulation of preventative initiatives. The authors perform a review of the literature on machine models and suggest an intelligent framework for diabetes prediction basedon their findings. Machine learning models are critically examined, and an intelligent machine learning-based architecture for diabetes prediction is proposed and evaluated by the authors. In this study, the authors utilize our framework to develop and assess decision tree, NAÏVE BAYES CLASSIFIER (NBC) and support vector machine (SVM) learning models for diabetes prediction, which are the most widely used techniques in the literature at the time of writing. It is proposed in this study that a unique intelligent diabetes mellitus prediction framework (IDMPF) is developed using machine learning. According to the framework, it was developed after conducting a rigorous review of existing prediction models in the literature and examining their applicability to diabetes. Using the framework, the authors describe the training procedures, model assessment strategies, and issues associated with diabetes prediction, as well as solutions they provide. The findings of this study may be utilized by health professionals, stakeholders, students, and researchers who are involved in diabetes prediction research and development. The proposed work gives 74% accuracy with the minimum errorrate.

4. REFERENCES

1. Diabetes, World Health Organization (WHO): 30 Oct 2018.
2. Vapnik, V.. Statistical learning theory. 1998 (Vol. 3). . New York, NY：Wiley，1998：Chapter 10-11, pp.401-492
3. Zhou Zhihua. Machine learning. Beijing: Tsinghua University Press，2016：pp.121-139, 298-300
4. Li Hang. Statistical learning methods. Beijing: Tsinghua University Press, 2012: Chapter 7，pp.95-135
5. Qin, J. and He, Z.S., 2005, August. A SVM face recognition method based on Gabor-featured key points. In Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on (Vol. 8, pp. 5144-5149). IEEE.
6. ZayritSoumayaa, BelhoussineDrissiTaoufiqa, NsiriBenayadb, KorkmazYunusc, AmmoumouAbdelkrim, The detection of Parkinson disease using the genetic algorithm and SVM classifier,Elsevier Ltd Applied Acoustics:2021.doi:10.1016/j.apacoust.2020.107528
7. Agrawal, P., Dewangan, A.: A brief survey on the techniques used for the

diagnosis of diabetes-mellitus. Int. Res. J. Eng. Technol. (IRJET).02(03) (2015). e-ISSN: 2395-0056; p-ISSN: 2395-0072

8. Ahmed TM. Using data mining to develop model for classifying diabetic patientcontrol level based on historical medical records. J TheorApplInfTechnol 2016;87.
9. Falvo D, Holland BE. Medical and psychosocial aspects of chronic illness and dis-ability. Jones & Bartlett Learning; 2017.
10. Skyler JS, Bakris GL, Bonifacio E, Darsow T, Eckel RH, Groop L, et al. Differentiationof diabetes by pathophysiology, natural history, and prognosis. Diabetes2017;66:241–55.
11. Organization WH. World health statistics 2016: monitoring health for the SDGssustainable development goals. World Health Organization; 2016.
12. Merad-boudia HN, Dali-Sahi M, Kachekouche Y, Dennouni-Medjati N. Hematologicdisorders during essential hypertension," diabetes & metabolic syndrome. ClinicalResearch & Reviews; 2019.
13. Cho N, Shaw J, Karuranga S, Huang Y, da Rocha Fernandes J, Ohlrogge A, et al. IDFDiabetes Atlas: global estimates of diabetes prevalence for 2017 and projections for2045. Diabetes Res ClinPract 2018;138:271–81.
14. Dorcely B, Katz K, Jagannathan R, Chiang SS, Oluwadare B, Goldberg IJ, et al.Novel biomarkers for prediabetes, diabetes, and associated complications. Diabetes,Metab Syndrome Obes Targets Ther 2017;10:345.
15. Singh PP, Prasad S, Das B, Poddar U, Choudhury DR. Classification of diabeticpatient data using machine learning techniques. Ambient communications andcomputer systems. Springer; 2018. p. 427–36.
16. Alghamdi, M., Al-Mallah, M., Keteyian, S., Brawner, C., Ehrman, J., and Sakr, S. (2017). Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: the henry ford exercise testing (FIT) project. PLoS One 12:e0179805. doi: 10.1371/journal.pone.0179805
17. American Diabetes Association (2012). Diagnosis and classification of diabetes mellitus. Diabetes Care 35(Suppl. 1), S64–S71. doi: 10.2337/dc12-s064
18. Duygu,ç., and Esin, D. (2011). An automatic diabetes diagnosis system based on LDA-wavelet support vector machine classifier. Expert Syst. Appl. 38, 8311–8315.
19. Habibi, S., Ahmadi, M., and Alizadeh, S. (2015). Type 2 diabetes mellitus screening and risk factors using decision tree: results of data mining. Glob. J. Health Sci. 7, 304–310. doi: 10.5539/gjhs.v7n5p304
20. Jegan, C. (2014). Classification of diabetes disease using support vector machine. Microcomput. Dev. 3, 1797–1801.
21. Jia, C., Zuo, Y., and Zou, Q. (2018). O-GlcNAcPRED-II: An integrated classification algorithm for identifying O-GlcNAcylation sites based on fuzzy undersampling and a K-means PCA oversampling technique. Bioinformatics 34, 2029–2036. doi: 10.1093/bioinformatics/bty039