Diabetes Prediction with Data Mining Algorithm

Akshaj Patil
MTECH CSE
IIITD
Delhi, India
akshaj19111@iiitd.ac.in

***Abstract*—This paper is about predicting diabetes prediction using data mining techniques. Nowadays there are a lot of computer science applications used in the field of medical science. Here we are using Pima-Indians-Diabetes-Database, this data set contains data of people with diabetes and without diabetes. Here we have used Decision Tree, Naive bayes, SVM and KNN classifiers to predict whether a person is diabetic or not.**

I. Introduction

Diabetes is caused due to an abnormal increase in blood sugar concentration. Diabetes can lead to various problems or can even lead to premature death. There are various ways to check whether a person is diabetic or not but consumes too much time and cost. Nowadays diabetes can be predicted using data mining techniques. Pima-Indian-Diabetic-Data-sets is used to train and evaluate data mining algorithms.
Data-sets can contain irrelevant data such as noise. This paper tries to overcome the problem of predicting null values and removing outliers.

II. Data-set

We have used Pima-Indian-Database.
There are 8 attributes that are used for predicting and the 9th attribute is a class attribute that is used for training and testing. Attributes are 1:Pregnancy 2:Plasma Glucose 3:Blood Pressure 4:Skin Thickness 5:Insulin 6:BMI 7:Pedigree 8:age 9:class
There are 35 null values in Blood Pressure, 11 null values in BMI, 395 null values in Insulin, 229 null values in Skin Thickness.
Class labels are True/False.

A. Data Distribution is of type

There are 768 rows in data. None of the attributes are co-related to each other.
Here we can also see that 25 percent of values of Skin Thickness and Insulin are 0.

	Pregnency	Plasma Glucose	Blood Pressure	Skin Thickness	Insuline	BMI	Pedigree	age
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000

Fig. 1.

III. Proposed System

For diabetes prediction, this paper has two main stages.
1:Data Pre-processing
2:Classification
Input for the system is Pima-Indian-Dataset and output form the system is predicted class whether a person is diabetic or not.

A. Data Pre-processing

First we will see co relation matrix for every attribute.

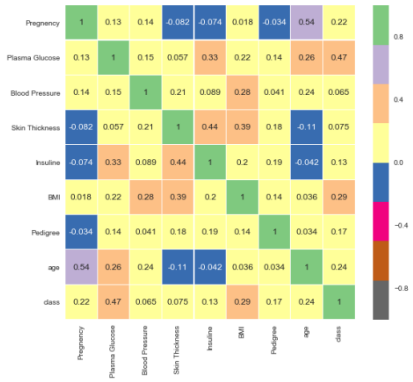

Fig. 2.

From this we can see that no attribute are highly related to each other, So from this, we can infer that now two attributes can merge and become new single attribute and also no attribute can be deleted.

Here in this data, we can see that there are many rows of Insulin and Skin Thickness with null values, but as they

are important attributes so instead of deleting these we will predict these missing values with help of classifier.

For predicting these missing values we will use the KNN classifier. We will train the KNN model with all attributes in which predicted class will be Insulin/Skin Thickness attribute and fill missing values with predicted values. Here KNN is used because the missing value can be the nearest neighbor of it. So using KNN we can get a more accurate value. Since missing values in all other remaining attributes are less so we can replace those missing values with mean value of that attribute.
There is outlier in Skin Thickness and Insulin so delete that particular row.

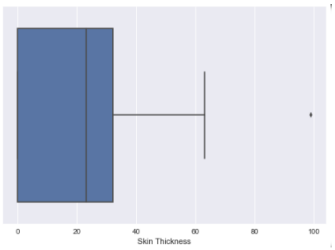

Fig. 3. Outlier in Skin Thickness

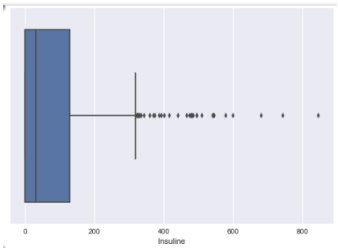

Fig. 4. Outlier in Insulin

Now divide data into two parts 70 percent of data will be for training and 30 percent of data will be for testing.

B. Classification

1:Decision Tree
Decision Tree solves the problem by transforming the data into a tree representation. Each internal node of the tree representation denotes an attribute and each leaf node denotes a class label. Decision Tree is used because missing values in the data also do NOT affect the process of building a decision tree to any considerable extent.
The accuracy obtained by decision tree is 77.23 percent

2:Naive bayes
Naive bayes assigns a class labels to the data by checking its probability. It checks the highest probability to which data have to be assigned. Naive bayes can be used because Naive bayes performs well on small data and it also works well on both discrete and continuous data.
The accuracy obtained by naive bayes classifier is 76.8 percent.

3:KNN Classifier
KNN assigns labels by comparing with 'k' nearest neighbors. KNN is used because KNN classifier requires no training period so data can be added seamlessly later.
The accuracy obtained by KNN classifier is 79.03 percent.

4:SVM
This algorithm plots each data item as a point in n-dimensional space with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well. SVM is used because it helps in avoiding Over-fitting.
The accuracy obtained by SVM is 80.02 percent.

C. Experiment and Discussion

Here we have compared accuracy's between two ways of data pre-processing.
1: Missing values filled by prediction with KNN.
2: Missing values are replaced with their respective attribute mean.
From the graph we can see that accuracy by replacing

Fig. 5. Comparision Between two ways of preprocessing

missing values with KNN prediction is more than accuracy by replacing missing values with their respective attribute mean.

Conclusion

Fast and more accurate diabetes prediction can be achieved. Here we have first pre-processed the data to make it more reliable so accuracy is increased. We have predicted class using four classifiers by seeing the accuracy we can see that SVM and KNN work best with this data pre-processing. We can also conclude that replacing missing values with prediction is more reliable than replacing missing values with mean because we can see a rise in accuracy in the comparison graph.

References

https://ieeexplore.ieee.org/abstract/document/8276012
https://pdfs.semanticscholar.org/7b8d/f4a0d81e2d819e4d0c21400fc0a9ffcf8bb4.pdf