THE NORTHCAP UNIVERSITY
POWERED BY
Arizona State University

THE NORTHCAP UNIVERSITY
SECTOR-23A, GURUGRAM
HARYANA – 122017

THE NORTHCAP UNIVERSITY
POWERED BY
Arizona State University

INTRODUCTION TO AI & ML PROJECT REPORT
ON
PIMA INDIANS DIABETES DATASET

TEAM MEMBERS:

BHAVYA SHARMA(19CSU371)
YUKTA BATRA(19CSU364)

INDEX

ABSTRACT:-

Machine learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence.The remarkable advances in biotechnology and health sciences have led to a significant production of data, such as high throughput genetic data and clinical information, generated from large Electronic Health Records (EHRs). To this end, application of machine learning and data mining methods in biosciences is presently, more than ever before, vital and indispensable in efforts to transform intelligently all available information into valuable knowledge. Diabetes mellitus (DM) is defined as a group of metabolic disorders exerting significant pressure on human health worldwide. Extensive research in all aspects of diabetes (diagnosis, etiopathophysiology, therapy, etc.) has led to the generation of huge amounts of data. The aim of the present study is to conduct a systematic review of the applications of machine learning, data mining techniques and tools in the field of diabetes research with respect to a) Prediction and Diagnosis, b) Diabetic Complications, c) Genetic Background and Environment, and e) Health Care and Management with the first category appearing to be the most popular. A wide range of machine learning algorithms were employed. In general, 85% of those used were characterized by supervised learning approaches and 15% by unsupervised ones, and more specifically, association rules. Support vector machines (SVM) arise as the most successful and widely used algorithm. Concerning the type of data, clinical datasets were mainly used. The title applications in the selected articles project the usefulness of extracting valuable knowledge leading to new hypotheses targeting deeper understanding and further investigation in DM.

INTRODUCTION

Significant advances in biotechnology and more specifically high-throughput sequencing result incessantly in an easy and inexpensive data production, there by ushering the science of applied biology into the area of machine learning .The increasingly growing number of applications of machine learning in healthcare allows us to glimpse at a future where data, analysis, and innovation work hand-in-hand to help countless patients without them ever realizing it. Soon, it will be quite common to find ML-based applications embedded with real-time patient data available from different healthcare systems in multiple countries, thereby increasing the efficacy of new treatment options which were unavailable before.

One of the chief ML applications in healthcare is the identification and diagnosis of diseases and ailments which are otherwise considered hard-to-diagnose. This can include anything from cancers which are tough to catch during the initial stages, to other genetic diseases. IBM Watson Genomics is a prime example of how integrating cognitive computing with genome-based tumor sequencing can help in making a fast diagnosis.

Maintaining up-to-date health records is an exhaustive process, and while technology has played its part in easing the data entry process, the truth is that even now, a majority of the processes take a lot of time to complete. The main role of machine learning in healthcare is to ease processes to save time, effort, and money. Document classification methods using vector machines and ML-based OCR recognition techniques are slowly gathering steam, such as Google's Cloud Vision API and MATLAB's machine learning-based handwriting recognition technology. MIT is today at the cutting edge of developing the next generation of intelligent, smart health records, which will incorporate ML-based tolls from the ground up to help with diagnosis, clinical treatment suggestions, etc.

ABOUT THE DATASET

We get our data set from Kaggle.

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether a patient has diabetes, based on certain diagnostic measurements included in the dataset.

Content

The datasets consist of several medical predictor variables and one target variable, Outcome. Predictor variables includes their BMI, insulin level, age, and so on.

MOTIVATION :-

There has been drastic increase in rate of people suffering from diabetes since a decade. Current human lifestyle is
the main reason behind growth in diabetes. In current medical diagnosis method, there can be three different types of
errors-

1. The false-negative type in which a patient in reality is already a diabetic patient but test results tell that the person
is not having diabetes.
2. The false-positive type. In this type, patient in reality is not a diabetic patient but test reports say that he/she is a
diabetic patient.
3. The third type is unclassifiable type in which a system cannot diagnose a given case. This happens due to
insufficient knowledge extraction from past data, a given patient may get predicted in an unclassified type.

However, in reality, the patient must predict either to be in diabetic category or non-diabetic category. Such errors
in diagnosis may lead to unnecessary treatments or no treatments at all when required. In order to avoid or reduce
severity of such impact, there is a need to create a system using machine learning algorithm and data mining
techniques which will provide accurate results and reduce human effort

TRAINING DESCRIPTION:-

About the attributes

Age:-

The risk of type 2 diabetes increases as you get older, especially after age 45. That's probably because people tend to exercise less, lose muscle mass and gain weight as they age. But type 2 diabetes is also increasing dramatically among children, adolescents and younger adults

BMI:-

When you're talking about diabetes and weight, your doctor will likely refer to your body mass index (BMI), which is a measure of how much body fat you have. It's an important measurement for diabetes and weight management, according to the American Diabetes Association (ADA), but it does have its limitations

Insulin:-

Insulin helps control blood glucose levels by signaling the liver and muscle and fat cells to take in glucose from the blood. Insulin therefore helps cells to take in glucose to be used for energy. If the body has sufficient energy, insulin signals the liver to take up glucose and store it as glycogen.

Insulin resistance occurs when your cells stop responding to the hormone insulin. This causes higher insulin and blood sugar levels, potentially leading to type 2 diabetes.

Skin:-

People who have diabetes tend to get skin infections. If you have a skin infection, you'll notice one or more of the following: Hot, swollen skin that is painful. An itchy rash and sometimes tiny blisters, dry scaly skin, or a white discharge that looks like cottage cheese

Diastolic blood pressure:-

The diastolic reading, or the bottom number, is the pressure in the arteries when the heart rests between beats. This is the time when the heart fills with blood and gets oxygen. A normal diastolic blood pressure is lower than 80. A reading of 90 or higher means you have high blood pressure..

Load libraries

We will be sticking to Python . The very first step is to load or import the all the libraries and the packages required to get the results we want. Some very primary and almost necessary packages for Machine Learning are — NumPy, Pandas, Matplotlib and seaborn.

Load dataset

Once the libraries are loaded, we need to get the data loaded. Pandas has a very straightforward function to perform this task — pandas.read_csv. The read.csv function is not just limited to csv files, but also can read other textbased files as well. Other formats can also be read using pandas read functions like html, json, pickled files etc. One thing which needs to be kept in mind is that your data needs to be in the same working directory as your current working directory or you will need to provide the complete path prefixed with a '/' within the function.

Summarize Data

Okay, so the data is loaded and ready to be actioned upon. But we first need to check how the data looks and what all does it contain. To begin with, you would want to see how many rows and columns does the data have and what all are the data types of each column .

A quick way to take a look type and shape of your data is — pandas.DataFrame.info. This tells you how many rows and columns your dataframe has and what data types and values do they contain.

```
df.describe()
```

	num_preg	glucose_conc	diastolic_bp	thickness	insulin	bmi	diab_pred	age	skin
count	768.000000	768.000000	768.000000	768.000000	764.000000	768.000000	731.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.870419	31.992578	0.472880	33.240885	0.809136
std	3.369578	31.972618	19.355807	15.952218	115.433301	7.884160	0.334589	11.760232	0.628517
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243500	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.368000	29.000000	0.906200
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.630500	41.000000	1.260800
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	3.900600

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   num_preg      768 non-null    int64
 1   glucose_conc  768 non-null    int64
 2   diastolic_bp  768 non-null    int64
 3   thickness     768 non-null    int64
 4   insulin       764 non-null    float64
 5   bmi           768 non-null    float64
 6   diab_pred     731 non-null    float64
 7   age           768 non-null    int64
 8   skin          768 non-null    float64
 9   diabetes      735 non-null    object
dtypes: float64(4), int64(5), object(1)
memory usage: 60.1+ KB
```

CHECKING FOR NULL VALUES

A field with a **NULL value** is a field with no **value**. It is very important to understand that a **NULL value** is different than a zero **value** or a field that contains spaces.

```
df.isnull().values.any()
```

```
True
```

```
df.isnull().sum()
```

```
num_preg          0
glucose_conc      0
diastolic_bp      0
thickness         0
insulin           4
bmi               0
diab_pred        37
age               0
skin              0
diabetes         33
dtype: int64
```

REMOVING NULL VALUES:-

Removing null values from the dataset by filling in mean

Data Cleaning

Real life data is not arranged and presented to you nicely and in a dataframe with no abnormalities. Data usually has a lot of so called abnormalities like missing values, a lot of features with incorrect format, features on different scales etc. All this needs to be handled manually which takes a lot of time and coding skills .

Pandas has various functions to check for such abnormalities like pandas.DataFrame.isna to check for values with NaNs etc. You might as well need to transform the data format in order to get rid of useless information like removing 'Mr.' and 'Mrs.' from names when a separate feature for gender is present. You might need to get it in a standard

format throughout the dataframe with the function pandas.DataFrame.replace or drop irrelevant features using pandas.DataFrame.drop.

DATA VISUALIZATIONS

Data Visualizations are very important as they are the quickest way to know the data and the patterns — if they even exist or not. Your data may have thousands of features and even more instances. It is not possible to analyze the numeric data for all of them. And if you do that, then what point is to have such powerful visualization packages like **Matplotlib** and **Seaborn**?

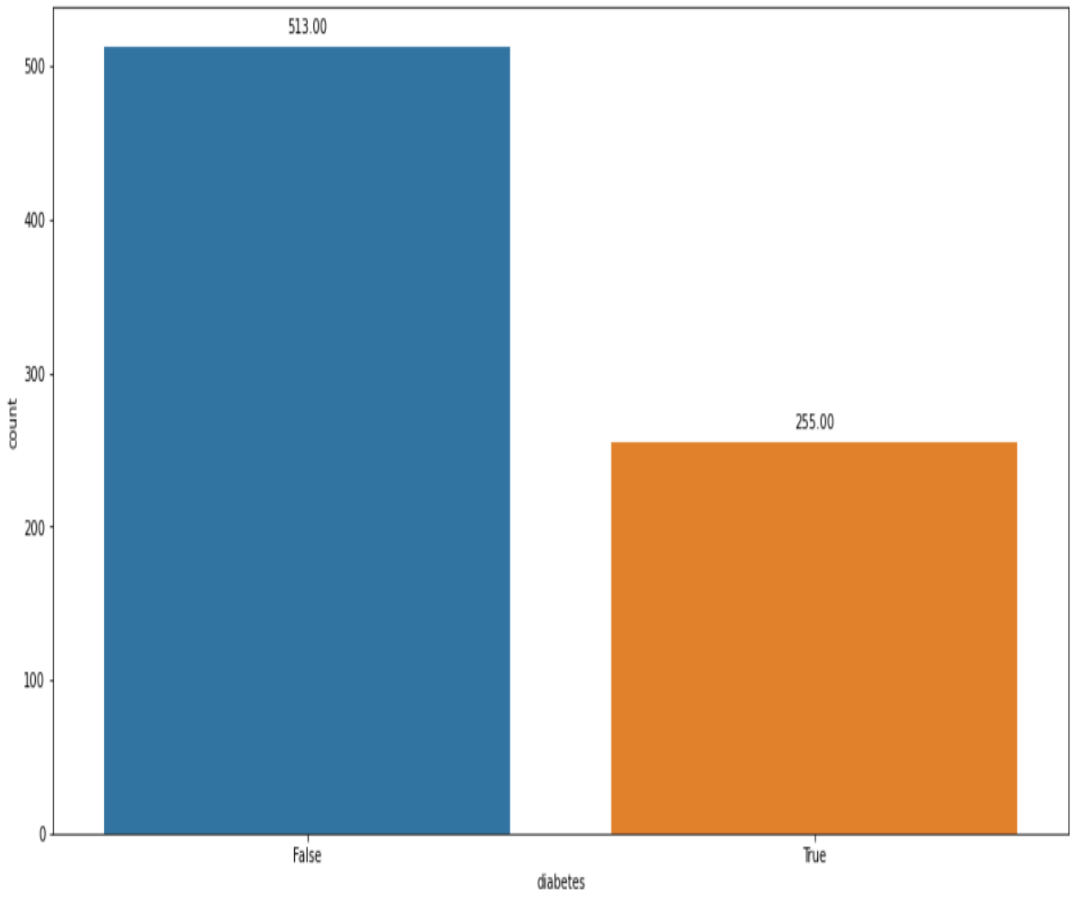
Visualizations using Matplotlib, Seaborn can be used to check the correlations within the features and with the target, scatter plots of data, histograms and boxplots for checking the spread and **skewness** and much more. Even pandas has its own built in visualization library — pandas.DataFrame.plot which has bar plot, scatter plot, histograms etc.

Seaborn is essentially a transformed matplotlib as it is built on matplotlib itself and makes the plots more beautiful and the process of plotting much quicker. Heatmap and pairplot are examples of power of Seaborn to quickly plot the visualization of the whole data to check multicollinearity, missing values etc.

One very efficient way to get most of the above descriptive and inferential statistics of the data is through **Pandas Profiling**. Profiling generates a beautiful report of the data with all the details mentioned above to let you analyze it

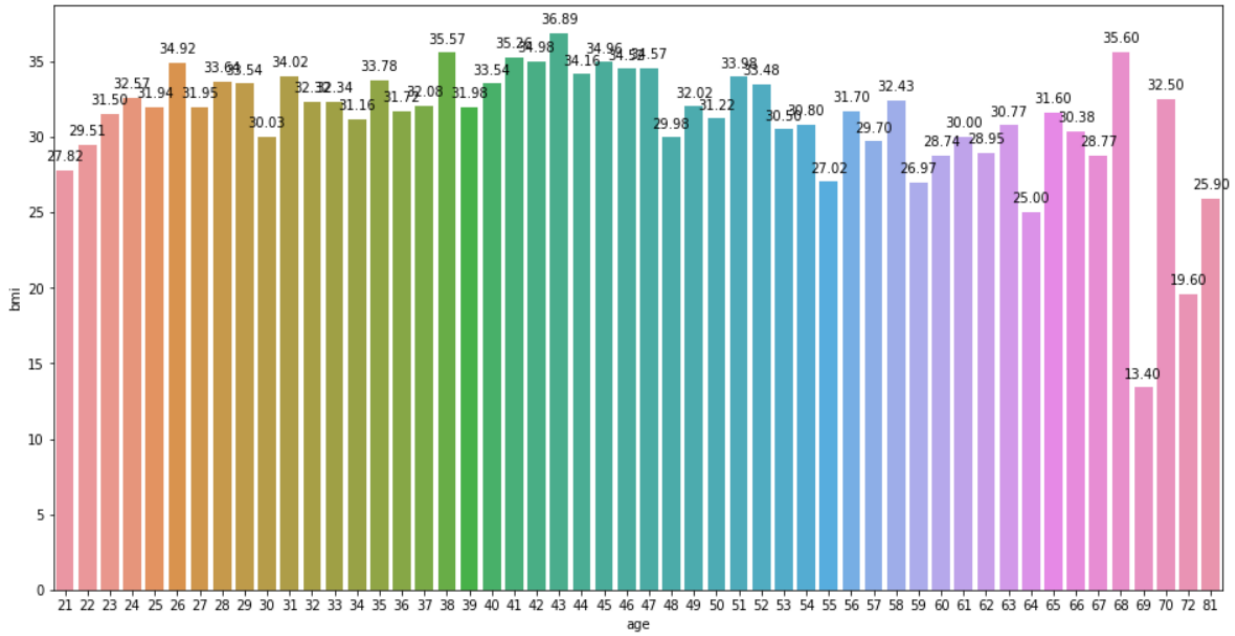
COUNTPLOT:-

A **countplot** is kind of like histogram or a bar graph for some categorical area. It simply shows the number of occurrences of an item based on a certain type of category.



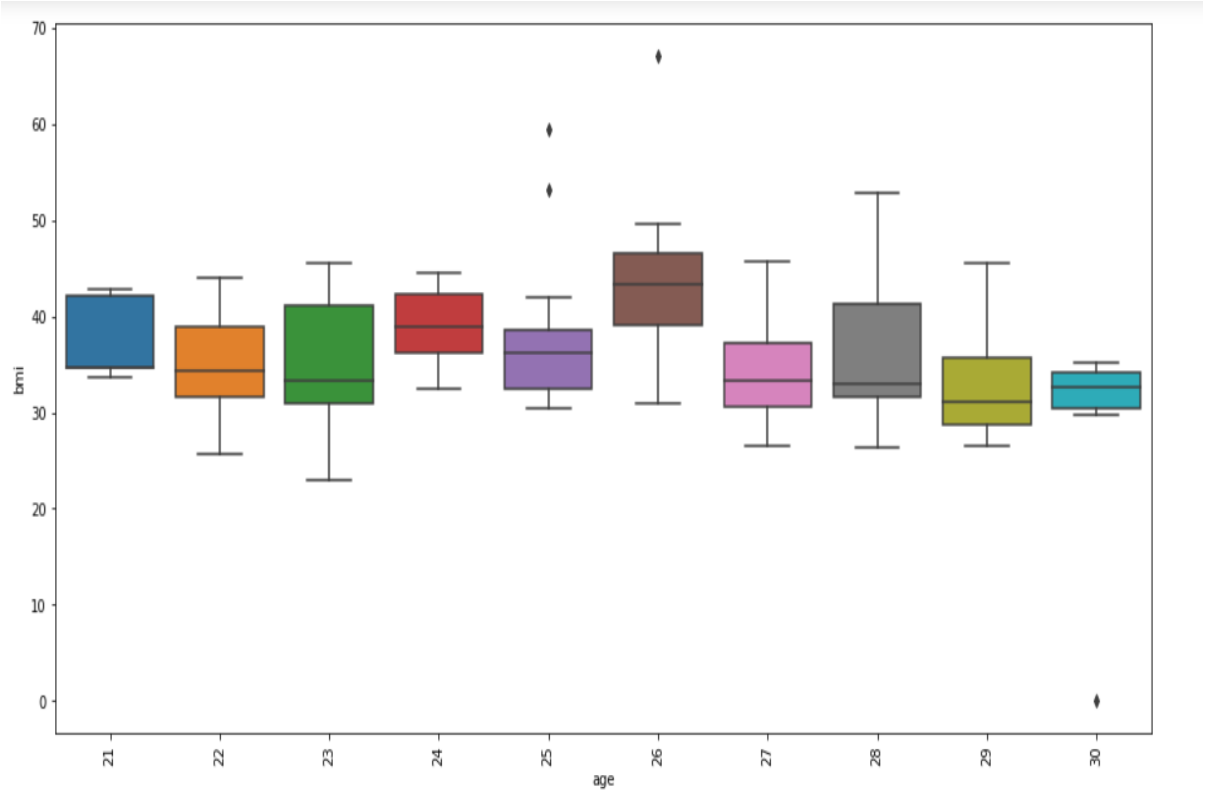
BAR GRAPH (BETWEEN AGE AND BMI):-

A BMI of between 18.5 and 24.9 is ideal



Box Plot

A box plot or boxplot is a method for graphically depicting groups of numerical data through their quartiles. Box plots may also have lines extending from the boxes indicating variability outside the upper and lower quartiles, hence the terms box-and-whisker plot and box-and-whisker diagram.



LABEL-ENCODING:-

Label Encoding is a popular **encoding** technique for handling categorical variables. In this technique, each **label** is assigned a unique integer based on alphabetical ordering. Let's see how to implement **label encoding** in Python using the scikit-learn library and also understand the challenges with **label encoding.**

```
df.head()
```

	num_preg	glucose_conc	diastolic_bp	thickness	insulin	bmi	diab_pred	age	skin	diabetes
0	6	148	72	35	0.0	33.6	0.627	50	1.3790	1
1	1	85	66	29	0.0	26.6	0.351	31	1.1426	0
2	8	183	64	0	0.0	23.3	0.672	32	0.0000	1
3	1	89	66	23	94.0	28.1	0.167	21	0.9062	0
4	0	137	40	35	168.0	43.1	2.288	33	1.3790	1

Feature Selection

Feature selection is the process of selecting a certain number of most useful features which will be used to train the model. This is done in order to reduce the dimensionality when most of the features are not contributing enough to the overall variance. If there are 300 features in your data and 97% of variance is explained by top 120 features, then it makes no sense to pound your algorithm with so many useless features. Reducing features not only saves time but costs as well.

Some of the popular feature selection techniques are SelectKBest, Feature elimation methods like RFE (recursive feature elimination) and embedded methods like LassoCV.

TRAIN - TEST SPLITING DATA

The procedure involves taking a dataset and dividing it into two subsets. The first subset is used to fit the model and is referred to as the training dataset.

Evaluate Algorithms

Once our data is ready, proceed to check the performance of the various regression/classification algorithms (based on the type of problem). we can first make a base model to set a benchmark to compare against.

Split-out validation dataset

Once the model is trained, it needs to be validated as well to see if it really generalized the data or it over/under fitted. The data in hand can be split up beforehand as training set and validation set. This split-out has various techniques — Train Test Split, Shuffle split etc. You can also run Cross Validation on the entire data set for a more robust validation. KFold Cross Validation, Leave-One-Out-CV are the most popular methods.

Test options and evaluation metric

The models need to evaluated based on a certain set of evaluation **metrics** which need to defined. For regression algorithms, some of the common metrics are — MSE and R Square.

Evaluation metrics pertaining to classification are a lot more diverse — Confusion Matrix, F1 Score, AUC/ROC curves etc. These scores are compared for each algorithm to check which ones performed better than the rest.

CLASSIFIER

In statistics, classification is the problem of identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations whose category membership is known

Logistic Regression:-

In statistics, the logistic model is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc

KNN Classifier:-

In statistics, the k-nearest neighbors algorithm is a non-parametric method proposed by Thomas Cover used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space.

Random Forest Classifier :-

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction

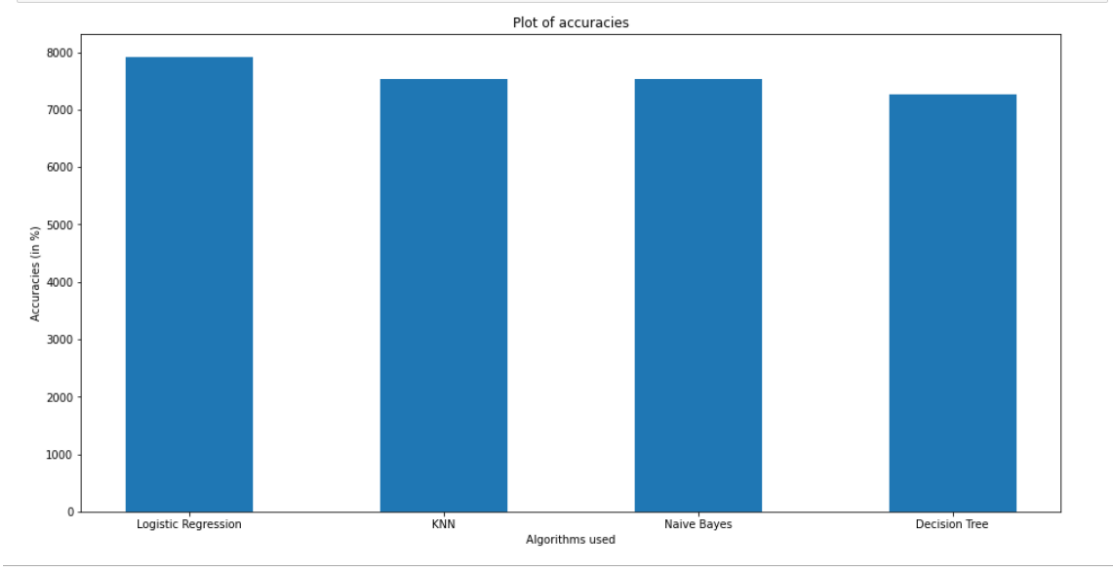
Decision Trees:-

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

Naïve Bayes:

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

CONCLUSION:-


Plot of accuracies
Accuracies (in %)
8000
7000
6000
5000
4000
3000
2000
1000
0
Logistic Regression
KNN
Naive Bayes
Decision Tree
Algorithms used

BIBLIOGRAPHY:-

- .com/uciml/pima-indians-diabetes-database