RESEARCH ARTICLE OPEN ACCESS

AI in HealthCare Automated Diagnose and Disease prediction DISEASETAP (Disease Prediction on a single Tap)

Manasseh John Wesley[1*], Bandari Theja[2], Dona Thomas[3], B.Srujana Eleena[4]
[1,2,3] *Department of Electronics and Communication, St Martin's Engineering College, Hyderabad, Telangana, India-500100*
[4]*Department of Computer Science Engineering, Stanley College of Engineering and technology for Women, 500001*
**Corresponding author: Manasseh John Wesley*

Abstract
As we witness the Health Care industry they have been digitally transformed, life scientists and doctors have anocean of data to base their research upon and Additionallyhuge volumes of health-related information are made accessible through many widely spread adoption of wearable tech. This opens up new opportunities for better, more informed healthcare.We are able to collect, structure and process a high volume of data and further make sense of it, to gain a much more deeper understanding of the human body its key objectives.it also has the strongest potential to revolutionize healthcare.This Data can be leveraged and used for a great transformation in Health care– As we see the Health care especially over the PHC- level where the Government PHC's (Primary health care centres)are structured in a way that a lot of people below poverty line and Low income and economic level visit the PHC for their diagnose and health check-up but due to 1. Lack of Availability of Doctor -2. Unavailability of Doctors on times of patient need 3. Gap in initiation of treatment 4. No proper efficient Diagnose for certain Diseases. To bridge this gap, we are bringing DISEASETAP- Disease prediction and Diagnose Automation through using Data Science and Machine learning where, we build Huge Datasetswhich include patient Liquid profiles and Blood test samples. we build Different Models for each disease and run a prediction analysis over these datasets with Careful and Prior knowledge over the parameters prescribed by the doctors in Diagnosing these diseases and then they are inculcated into our Programming interfaces, made into a running Prediction Models and Classifiers which will intern predict the disease the patient is affected with and also his Risk Factor.

Date of Submission: 31-03-2020 Date of Acceptance: 17-04-2020

I. INTRODUCTION

Medicine and healthcare are the most two important part of our human lives. Traditionally, medicine solely relied upon the discretion advised by the doctors. For example, a doctor would have to suggest suitable treatments based on a patient's symptoms. However, this wasn't always correct and was prone to human errors. With the advancements in computers and in particularly Data Science, it is now possible to obtain accurate diagnostic measures [1]. The most well-known application of Data science in healthcare are as follows, medical imaging, drug discovery, genetics, predictive diagnosis and several others that make use of data science [2]. **Medical Imaging** The primary and foremost use of data science in the health industry is through medical imaging [3]. There are various imaging techniques like X-Ray, MRI and CT Scan. All these techniques visualize the inner parts of the human body [4]. Traditionally, doctors would manually inspect these images and find irregularities within them. However, it was often difficult to find microscopic deformities and as a result, doctors could not suggest a proper diagnosis. Now with the Technological enforcement of **deep learning technologies in data science**, it is now possible to find such microscopic deformities in the scanned images [5]. Through image segmentation, it is possible to search for defects present in the scanned images [6]. Other than this, there are also other image processing techniques like image recognition using **Support Vector Machines**, image enhancement and reconstruction, edge detection etc **Drug Discovery with Data Science** Drug Discovery is a highly complicated discipline. Pharmaceutical industries are heavily relying on data science to solve their problems and create better drugs for the people [7]. Drug Discovery is a time-consuming process that also involves heavy financial expenditure and heavy testing. Data Science and **Machine Learning algorithms** are revolutionizing this process and providing extensive

insights into optimizing and increasing the success rate of predictions [8].Pharmaceutical companies use the insights from the patient information such as mutation profiles and patient metadata. This information helps the researchers to develop models and find statistical relationships between the attributes. This way, companies can design drugs that address the key mutations in the genetic sequences. Also, deep learning algorithms can find the probability of the development of disease in the human system. **Monitoring Patient Health** Data Science plays a vital role in **IoT (Internet of Things)** [9].

These IoT devices, that are present as wearable devices that track heartbeat, temperature and other medical parameters of the users. The data that is collected is analysed with the help of data science. With the help of analytical tools, doctors are able to keep track of patient's circadian cycle, their blood pressure as well as their calorie intake.Other than wearable monitoring sensors, doctor can monitor a patient's health through home devices. For patients that are chronically ill, there are several systems that track patient's movements, monitor their physical parameters and analyse the patterns that are present in the data. It makes use of real-time analytics to predict if the patient will face any problem based on the present condition. Furthermore, it helps the doctors to take the necessary decisions to help the patients in distress [10]. **Tracking& Preventing Diseases** Data Science plays a pivotal role in monitoring patient's health and notifying necessary steps to be taken in order to prevent potential diseases from taking place [11]. Data Scientists are using powerful predictive analytical tools to detect chronic diseases at an early level. In many extreme cases, there are instances where due to negligibility diseases are not caught at an early stage.

This proves to be highly detrimental to not only the patient's health but also the economic costs. As the disease grows, the cost of curing it also increases. Therefore, data science plays a huge role in optimizing the economic spending on healthcare. There are several instances where AI has played a huge role in detecting diseases at an early stage. Researchers at the **University of Campinas** in Brazil have developed an AI platform that can diagnose **Zika virus** using metabolic markers [12].Several other companies like IQuity are using machine learning to detect autoimmune diseases [13].

- **What's Disease tap?**

Disease tap is a service comprising of deep tech such as Disease prediction and Automated Diagnose, where it can read through a patient Blood test report or his Liquid profile and predict which disease he is prone to along with the risk factor on which scale he is effected by the disease,this predictive model uses historical data, learns from it, and finds patterns and generates accurate predictions from it. It finds various correlations and association of symptoms, finds habits, diseases and then makes meaningful predictions. This can predict the deterioration in patient's health and provide preventive measures and start an early treatment that will assist in reducing the risk of the further aggravation of patient health.

II. SOFTWARE SYSTEM DESIGN

2.1.1 **Python language: Python** is a general-purpose language;it has the right tools/libraries.

Fig 2.1.1: Logo of the python language

2.1.2 Anaconda navigator:

we are using **Anaconda Navigator** as a desktop graphical user interface (GUI) included in **Anaconda®** we are launching applications and using the conda packages, environments, and channels without using command-line commands [14].

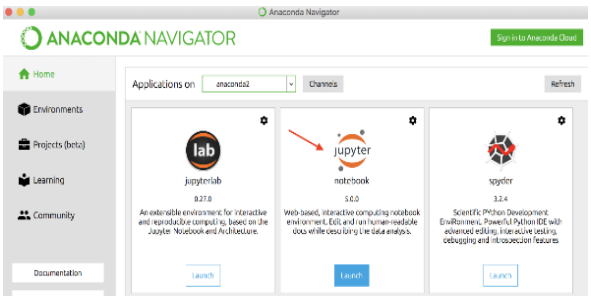


Fig 2.1.2: Anaconda navigator and we have to enable Juypter notebook

2.1.3 Juypter Notebook: we are using the **Jupyter Notebook** in Anaconda Navigator, we are creating and sharing documents that contain live code, equations, visualizations and explanatory text. Uses include: data cleaning and transformation, numerical simulation, statistical modelling, machine learning and much more.

2.1.4 **Libraries needed for the implementation of Code:**

1. **numpy**: To work with arrays
2. **pandas**: To work with csv files and dataframes
3. **matplotlib**: To create charts using pyplot, define parameters using rcParams and color them with cm.rainbow
4. **warnings**: To ignore all warnings which might be showing up in the notebook due to past/future depreciation of a feature
5. **train_test_split**: To split the dataset into training and testing data
6. **Standard Scaler**: To scale all the features, so that the Machine Learning model better adapts to the dataset

2.1.5 **Import dataset**

After downloading the dataset from Kaggle, saved it to the working directory with the name dataset.csv.

2.1.6 Create a New Environment in Anaconda Navigator:

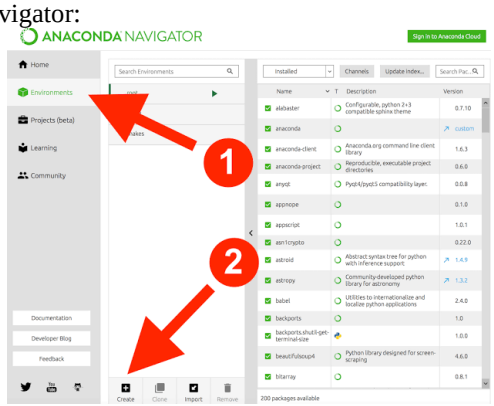


Fig 2.1.6: This figure shows bellow the two steps first to go into environments and then click on create.

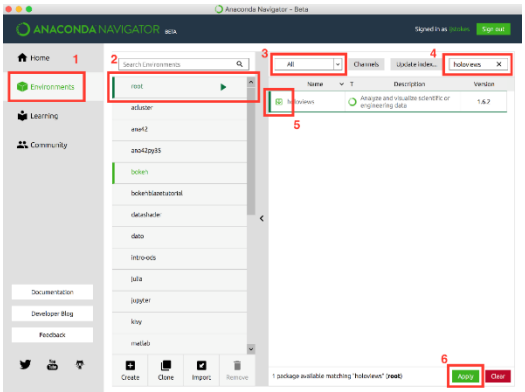


Fig 2.1.4: Enter the name of the environment and also select the language and its version

2.1.7 Now in the new environment installing the above stated libraries / packages (Mentioned in 2.1.4)

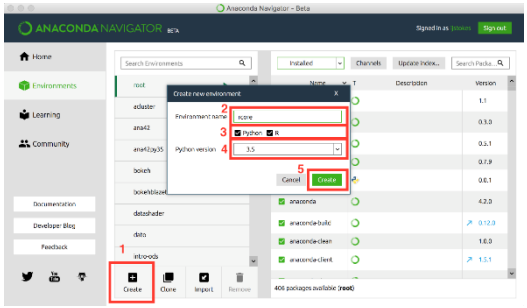


Fig 2.1.5: Type the Package name in the search bar and click apply to install (same way installs all the needed libraries and packages as mentioned in 2.1.4)

III. Proposed model:

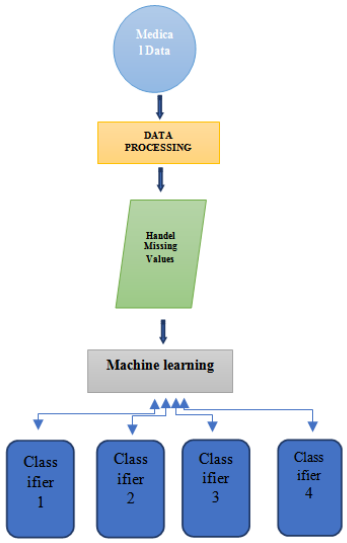


Fig 2.1.6: Compact proposed model .

IV. IMPLEMENTATION

We are importing the data set from Kaggle saved it to the working directory with the name dataset.csv. Next we used describe() method Which reveals that the range of each variable is different. The maximum value of age is 77 but for chol it is 564. Thus, feature scaling must be performed on the dataset.

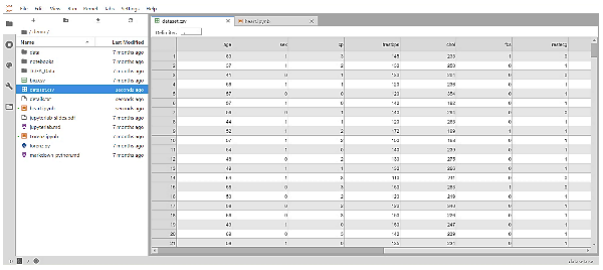


Fig 3: It is a screenshot from the working platform Juypter notebook, there are a total of 13 features and 1 target variable. Also, there are no missing values so we don't need to take care of any null values.

Understanding the data - Correlation Matrix
It enables us to see the correlation matrix of features and also try to analyse it. The figure size is defined to 12 x 8 by using rcParams. Then, used pyplot to show the correlation matrix. Using xticks and yticks, added names to the correlation matrix. colorbar() shows the colorbar for the matrix.

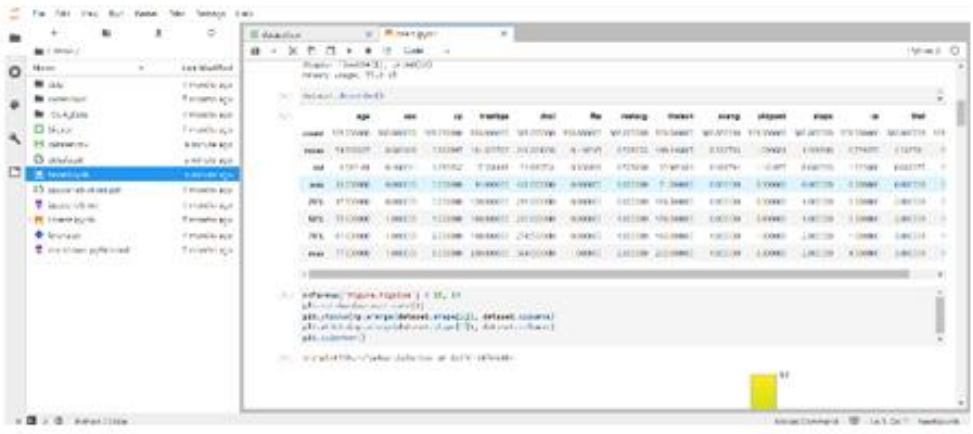


Fig 3.1: Here is a screenshot from the working platform of juypter notebook showing ploting of the correlation matrix

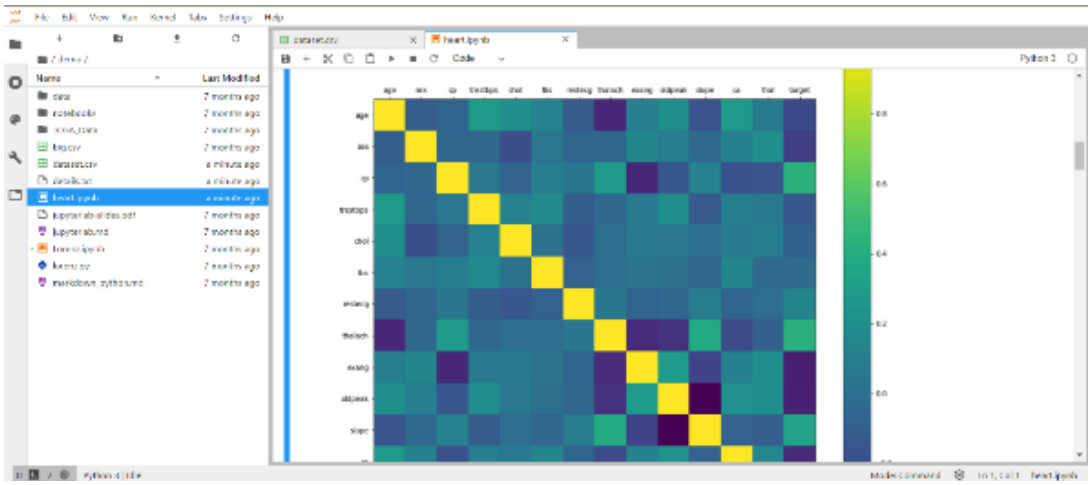


Fig 3.2: Here is a screenshot from the working platform of juypter notebook showing that there is no single feature that has a very high correlation with our target value. Also, some of the features have a negative correlation with the target value and some have positive.

Here we can see that there is no single feature that has a very high correlation with our target value. Also, some of the features have a negative correlation with the target value and some have positive.

Histogram: The best part about this type of plot is that it just takes a single command to draw the plots and it provides so much information in return. Just use dataset.hist().

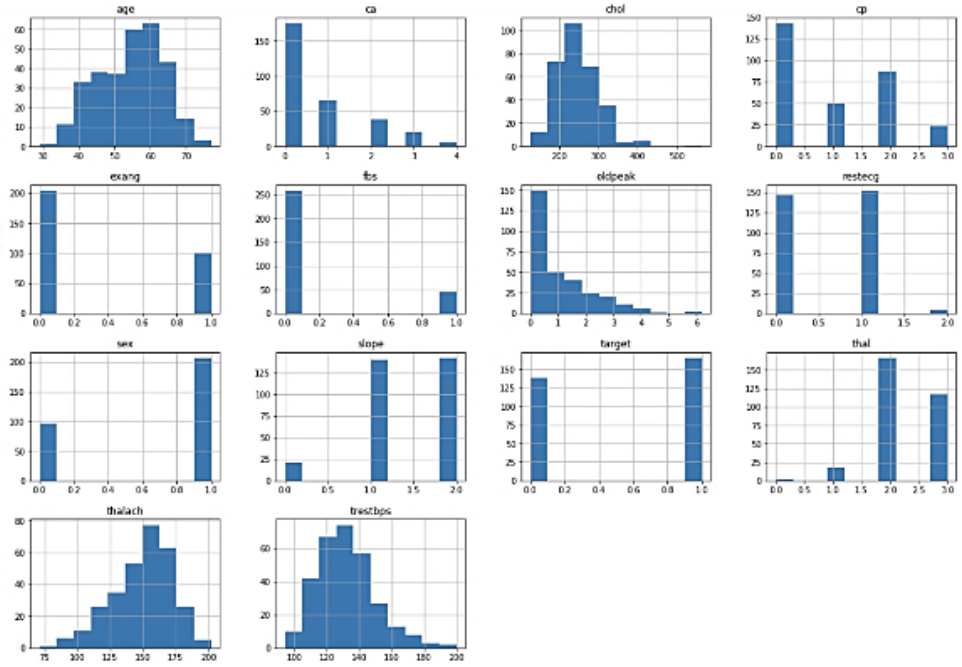


Fig 3.3: This is the visual picture extracted from the Histogram displayed by the code initiated

If we take a closer look at these plots. It shows how each feature and label is distributed along different ranges, which further confirms the need for scaling. wherever we see discrete bars, it basically means that each of these is actually a categorical variable. We will need to handle these categorical variables before applying Machine Learning. Our target labels have two classes, 0 for no disease and 1 for disease.

Bar Plot for Target Class

It's essential that the dataset we are working on should be approximately balanced. An extremely imbalanced dataset can render the whole model training useless and thus, will be of no use, so we plotting a Target class Bar plot to see how balanced the dataset are.

For x-axis we use the unique() values from the target column and then set their name using xticks. For y-axis, we use value_count() to get the values for each class. colored the bars as green and red.

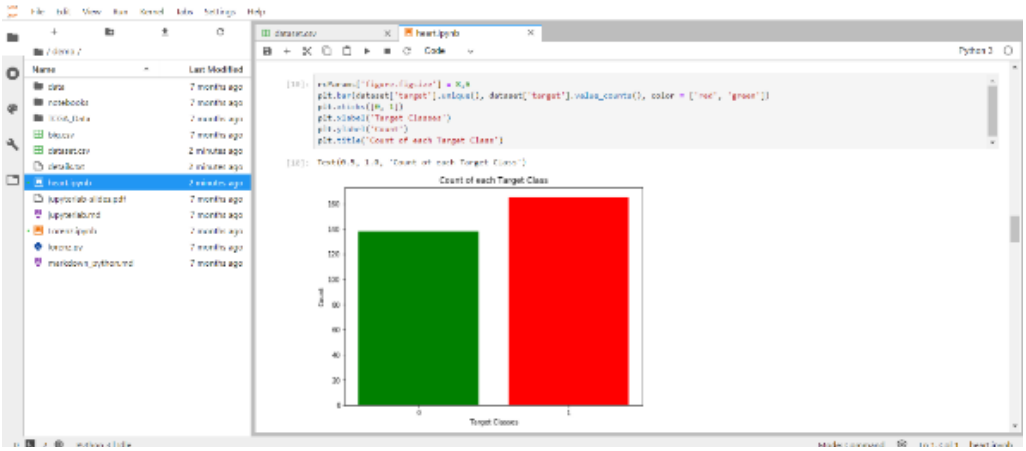


Fig 3.4: Here is a screenshot from the working platform of juypter notebook showing the Barplot of the target class

From the plot, we can see that the classes are almost balanced and we are good to proceed with data processing.

Data Processing

To work with categorical variables, we should break each categorical column into dummy columns with 1s and 0s. Let's say we have a column Gender, with values 1 for Male and 0 for Female. It needs to be converted into two columns with the value 1 where the column would be true and 0 where it will be false.

Original Columm # Dummy Columns
| Gender_0 || Gender_1 |
| 0 || 1 |
| 0 || 1 |
| 1 || 0 |

In order to do this, we use the get_dummies() method from pandas.
we need to scale the dataset for which we will use the StandardScaler. The fit_transform() method of the scaler scales the data and we update the columns.
Now once the DATASET is ready, we can dive into the next part.

Machine Learning

In this project, we took 4 algorithms and varied their various parameters and compared the final models. I split the dataset into 67% training data and 33% testing data.
We use 4 different classifiers

- K Neighbors Classifier
- Support Vector Classifier
- Decision Tree Classifier
- Random Forest Classifier

K Neighbors Classifier

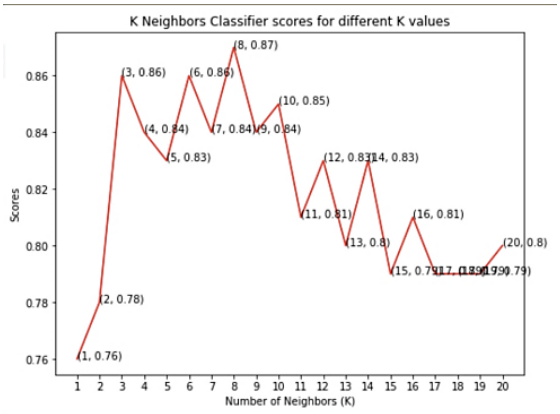
This classifier looks for the classes of K nearest neighbors of a given data point and based on the majority class, it assigns a class to this data point. However, the number of neighbors can be varied. we varied them from 1 to 20 neighbors and calculated the test score in each case.

```
knn_scores
= []
          for k inrange(1,21):
           knn_classifier
          =KNeighborsClassifier(n_neighbors  = k)
           knn_classifier.fit(X_train,  y_train)

          knn_scores.append(knn_classifier.score(X_test
          y_test))
```
Code - Manasseh John Wesley

```
plt.xticks([i  for  i
inrange(1,  21)])
plt.xlabel('Number
of Neighbors (K)')
plt.ylabel('Scores')
plt.title('K
Neighbors Classifier
scores for different
K values')
```
Code – Manasseh John Wesley

Fig 3.5: Here is a screenshot from the working platform of juypter notebook showing us the achieved the maximum score of 87% when the number of neighbours was chosen to be 8.



Similarly, we repeat the process with rest of the other 3 Classifiers and hence we can observe their following results.

Decision Tree Classifier :
We used the rainbow method to select different colours for each bar and plot a bar graph of the scores achieved by each.

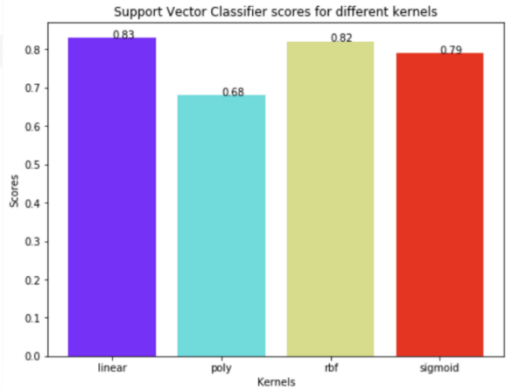
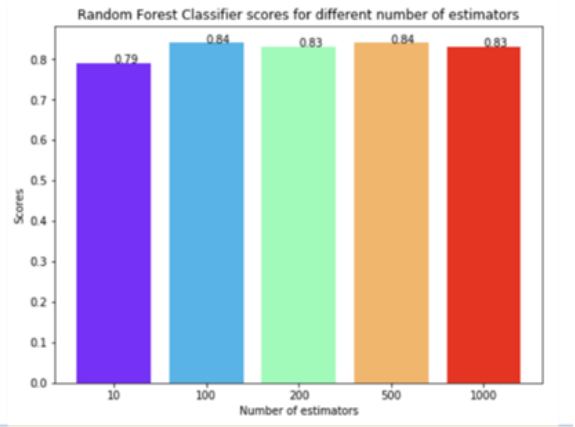


Fig 3.5: Here is a screenshot from the working platform of juypter notebook showing us thelinear kernel performed the best for this dataset and achieved a score of 83%.



Decision Tree Classifier: This classifier creates a decision tree based on which, it assigns the class values to each data point. Here, we can vary the maximum number of features to be considered while creating the model. we range features from 1 to 30 (the total features in the dataset after dummy columns were added).

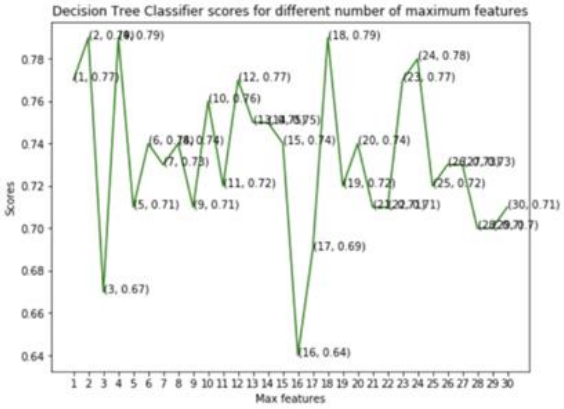


Fig 3.5: Here is a screenshot from the working platform of juypter notebook showing the maximum score is 79% and is achieved for maximum features being selected to be either 2, 4 or 18.

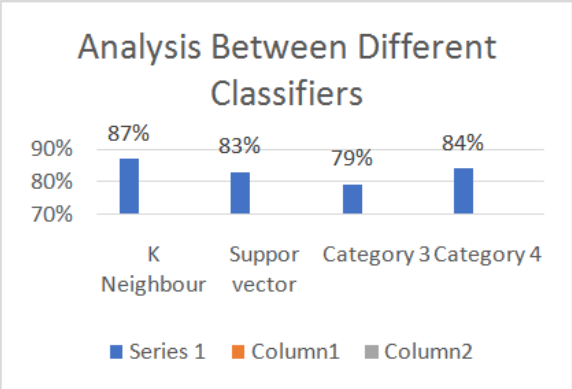
Random Forest Classifier:
This classifier takes the concept of decision trees to the next level. It creates a forest of trees where each tree is formed by a random selection of features from the total features. Here, we can vary the number of trees that will be used to predict the class. we calculate test scores over 10, 100, 200, 500 and 1000 trees.

We plot these scores across a bar graph to see which gave the best results. we do not directly set the X values as the array [10, 100, 200, 500, 1000]. It will show a continuous plot from 10 to 1000, which would be impossible to decipher. So, to solve this issue, we first usethe X values as [1, 2, 3, 4, 5]. Then, I renamed them using xticks.

```
rf_scores  =
[]
            estimators = [10, 100, 200, 500, 1000]
            for i in estimators:
                rf_classifier =RandomForestClassif
                rf_classifier.fit(X_train, y_train
                rf_scores.append(rf_classifier.sco
```
Code – Manasseh John Wesley

Fig 3.5: Here is a screenshot from the working platform of juypter notebook showing the bar graph, we can see that the maximum score of 84% was achieved for both 100 and 500 trees**.**

Result Analysis between different console applications:
The bellow graph consists of the results following of all 4 classifiers which are used in the classification and their percentage of classification range and efficiency is show in the graph.



1. K Neighbors Classifier: 87%
2. Support Vector Classifier: 83%
3. Decision Tree Classifier: 79%
4. Random Forest Classifier: 84%

ConclusionAND FURTHER USE:

The analysis of the disease patient dataset with proper data processing. 4 models were trained and tested with maximum scores as follows.

We use this technology in wide range of Diseases to predict who are affected and their risk factor and we can cover a large number of patients at the same time with fast and efficient analysis.

By using this we can improve patient care, chronic disease management and increasing the efficiency of supply chains and pharmaceutical logistics. Population health management is becoming an increasingly popular topic in predictive analytics. This is a data-driven approach focusing on prevention of diseases that are commonly prevalent in society.

With **Disease tap**, hospitals can predict the deterioration in patient's health and provide preventive measures and start an early treatment that will assist in reducing the risk of the further aggravation of patient health. Furthermore, predictive analytics plays an important role in monitoring the logistic supply of hospitals and pharmaceutical departments.

This has many applications in healthcare. The medicine and healthcare industry have heavily utilized Data Science for the improving lifestyle of patients and predicting diseases at an early stage. Furthermore, with advancements in medical image analysis, it is possible for the doctors to find out microscopic tumours that were otherwise hard to find. Therefore, data science has revolutionized healthcare and the medical industry in large ways.

REFERENCES:

[1]. Van Der Aalst, Wil. "Data science in action." Process mining. Springer, Berlin, Heidelberg, 2016. 3-23.

[2]. Wasan, Siri Krishan, Vasudha Bhatnagar, and Harleen Kaur. "The impact of data mining techniques on medical diagnostics." Data Science Journal 5 (2006): 119-126.

[3]. Kagadis, George C., et al. "Cloud computing in medical imaging." Medical physics 40.7 (2013).

[4]. Joseph, Rohini Paul, C. Senthil Singh, and M. Manikandan. "Brain tumor MRI image segmentation and detection in image processing." International Journal of Research in Engineering and Technology 3.1 (2014): 1-5.

[5]. Shen, Dinggang, Guorong Wu, and Heung-Il Suk. "Deep learning in medical image analysis." Annual review of biomedical engineering 19 (2017): 221-248.

[6]. Hoover, Adam, et al. "An experimental comparison of range image segmentation algorithms." IEEE transactions on pattern analysis and machine intelligence 18.7 (1996): 673-689.

[7]. Dhar, Vasant. "Data science and prediction." Communications of the ACM 56.12 (2013): 64-73.

[8]. Elshawi, R., Sakr, S., Talia, D., & Trunfio, P. (2018). Big data systems meet machine learning challenges: towards big data science as a service. Big data research, 14, 1-11.

[9]. Rahmani, Amir M., et al. "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach." Future Generation Computer Systems 78 (2018): 641-658.

[10]. Gigerenzer, Gerd, et al. "Helping doctors and patients make sense of health statistics." Psychological science in the public interest 8.2 (2007): 53-96.

[11]. Centers for Disease Control, Prevention (US), and National Center for Infectious Diseases (US). Addressing emerging infectious disease threats: a prevention strategy for the United States. Centers for Disease Control and Prevention, 1994.

[12]. Tester, Mark A. "W001: Abiotic Stress Integrating Genomics and Phenomics to Study Genetic Control of Salinity Tolerance Traits."

[13]. Bates, David W., et al. "Big data in health care: using analytics to identify and manage high-risk and high-cost patients." Health Affairs 33.7 (2014): 1123-1131.

[14]. Lakshminarayanan, Vasudevan, et al. Understanding Optics With Python. CRC Press, 2018.