

predicting the breast cancer

Machine Learning



Smartbridge, Hyderabad

Summer Internship Project Report

*Submitted By,*

Pagadala Vamsi Krishna

SaiKiran

Sarath Reddy

Madhuri

Sahil Shaik

**Certificate**

This is to certify that **Pagadala Vamsi Krishna, Sai Kiran, Sarath Reddy, Madhuri , Sahil Shaik** has completed a four-week **Internship** cum hands-on training program conducted by **Smartbridge** in collaboration with **IBM** at Jawaharlal Nehru Technological University in **“Artificial Intelligent”** during the time period of 3rd May, 2019 to 22th June, 2019.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor

ABSTRACT

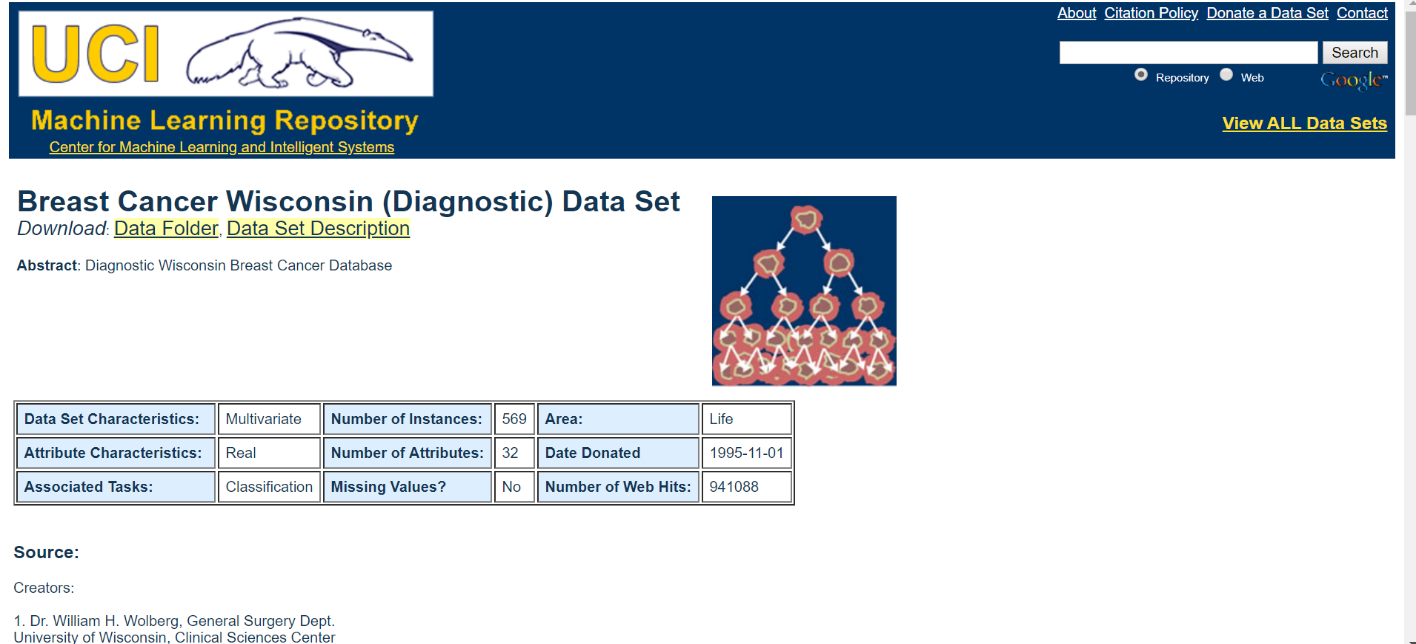
In today’s digital world, food app like Zomato is widely used because it provides a platform for people to share their opinion about the restaurants and cafes they have visited. This research paper includes analysis of client ratings and reviews in Zomato utilizing content mining. Utilizing content mining, break down the content audits/reviews from the client with a specific end goal to create productive result and legit surveys. Gathering surveys dataset and handling it to check the reliability of the rating given and audit composed by client. Ascertaining reliability of the eatery subsequent to dissecting the surveys as indicated by the administration gave and cost estimation. All through this procedure look at the client audit premise on their content setting and it demonstrates that how they feel about their visit to that place.

CONTENTS

1. Title of the project 4
   1. Introduction 4
   2. Objectives of Research 5
   3. Problem Statement 9
   4. Industry Profile 9
2. Review of Literature 10
3. Data Collection 12
4. Methodology 12
   1. Import Libraries 12
   2. Import Data 14
   3. Exploratory Data Analysis 14
   4. Cleaning Of data 17
   5. Splitting of Data 17
   6. Algorithm 18
   7. Evaluation 23
5. Findings and Suggestions 23
6. Conclusion 25
7. References 25
8. **Review of Literature**

A literature review shows that there have been several studies on the survival prediction problem using statistical approaches and artificial neural networks. However, we could only find a few studies related to medical diagnosis and recurrence using Machine Learning approaches such as decision trees. Delen et al. used artificial neural networks, decision trees and logistic regression to develop prediction models for breast cancer survival by analysing a large dataset, the SEER cancer incidence database. Lundin et al. used ANN and logistic regression models to predict 5, 10, and 15 -year breast cancer survival. They studied 951 breast cancer patients and used tumour size, axillary nodal status, histological type, mitotic count, nuclear pleomorphism, tubule formation, tumour necrosis, and age as input variables. Pend Harker patterns in breast cancer. In this study, they showed that data mining could be a valuable tool in identifying similarities (patterns) in breast cancer cases, which can be used for diagnosis, prognosis, and treatment purposes. These studies are some examples of researches that apply data mining to medical fields for prediction of diseases.

AlirezaOsarech, BitaShadgar used SVM classification technique on two different benchmark datasets for breast cancer which got 98.80% and 96.63% accuracies. MandeepRana, PoojaChandorkar, AlishibaDsouza worked on the diagnosis and the prediction of recurrence of breast cancer by applying KNN, SVM, Naïve Bayes and Logistic Regression techniques, programmed in MATLAB. The classification techniques are applied on two datasets taken from UCI depository. A dataset of them is used for identification of disease(WDBC) and the next one is used for recurrence prediction (WPBC)[3].VikasChaurasia, BB Tiwari and Saurabh Pal used three famous algorithms such as J48, Naive bayes, RBF, to build predictive models on breast cancer prediction and compared their accuracy. The results had shown that Naive Bayes predicted well among them with an accuracyof97.36%. Haifeng Wang and Sang Won Yoon compared Naive Bayes Classifier, Support Vector Machine (SVM), AdaBoost tree, Artificial Neural Networks (ANN), to find a powerful model for breast cancer prediction. They implemented PCA for dimensionality reduction. S.Kharya worked on breast cancer prediction and stated that artificial neural networks are widely used. The paper featured about the advantages and short comings of using machine learning methods like SVM, Naive Bayes, Neural network and Decision trees. NareshKhuriwal, Nidhi Mishra took data from Wisconsin Breast Cancer database and worked on breast cancer diagnosis. The results of their experiments proved that ANN and Logistic Algorithm worked better and provided a good solution. It achieved an accuracy of 98.50%.

1. **Data Collection**

**Data Source:**

<https://archive.ics.uci.edu/ml/datsets/Breast+cancer+Wisconsin+(Diagnostic)>

The data was duly collected from the above link and reverified for changed attributes by visiting hospitals and checking for latest changes. The data set was modified accordingly and then used.

1. **Methodology**

**4.1** Importing Libraries

To perform majority of tasks in python it is very much important to import libraries that are used in python same as in C, C++ and java. There are different kinds of libraries that their respective functions in python and some of them are,

* + 1. **NumPy**

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

* A powerful N-dimensional array object.
* Sophisticated (broadcasting) functions.
* Tools for integrating C/C++ and Fortran code.
* Useful linear algebra, Fourier transform, and random number capabilities.
  + 1. **Pandas**

Pandas is the most popular python library that is used for data analysis. It provides highly optimized performance with back-end source code is purely written in C or Python.

We can analyse data in pandas with:

* Series
* Data Frames
  + 1. **Matplotlib**

Matplotlib is a python library used to create 2D graphs and plots by using python scripts. It has a module named pyplot which makes things easy for plotting by providing feature to control line styles, font properties, formatting axes etc. It supports a very wide variety of graphs and plots namely - histogram, bar charts, power spectra, error charts etc. It is used along with NumPy to provide an environment that is an effective open source alternative for MATLAB. It can also be used with graphics toolkits like PyQt and wxPython.

* + 1. **Seaborn**

Seaborn is a library for making statistical graphics in Python. It is built on top of [matplotlib](https://matplotlib.org/) and closely integrated with [pandas](https://pandas.pydata.org/) data structures.

Here is some of the functionality that seaborn offers:

* A dataset-oriented API for examining [relationships](https://seaborn.pydata.org/examples/scatter_bubbles.html" \l "scatter-bubbles) between [multiple variables](https://seaborn.pydata.org/examples/faceted_lineplot.html" \l "faceted-lineplot).
* Specialized support for using categorical variables to show [observations](https://seaborn.pydata.org/examples/jitter_stripplot.html" \l "jitter-stripplot) or [aggregate statistics](https://seaborn.pydata.org/examples/pointplot_anova.html" \l "pointplot-anova).
* Options for visualizing [univariate](https://seaborn.pydata.org/examples/distplot_options.html" \l "distplot-options) or [bivariate](https://seaborn.pydata.org/examples/joint_kde.html" \l "joint-kde) distributions and for [comparing](https://seaborn.pydata.org/examples/horizontal_boxplot.html" \l "horizontal-boxplot) them between subsets of data.
* Automatic estimation and plotting of [linear regression](https://seaborn.pydata.org/examples/anscombes_quartet.html" \l "anscombes-quartet) models for different kinds [dependent](https://seaborn.pydata.org/examples/logistic_regression.html" \l "logistic-regression) variables.
  + 1. **Sklearn**

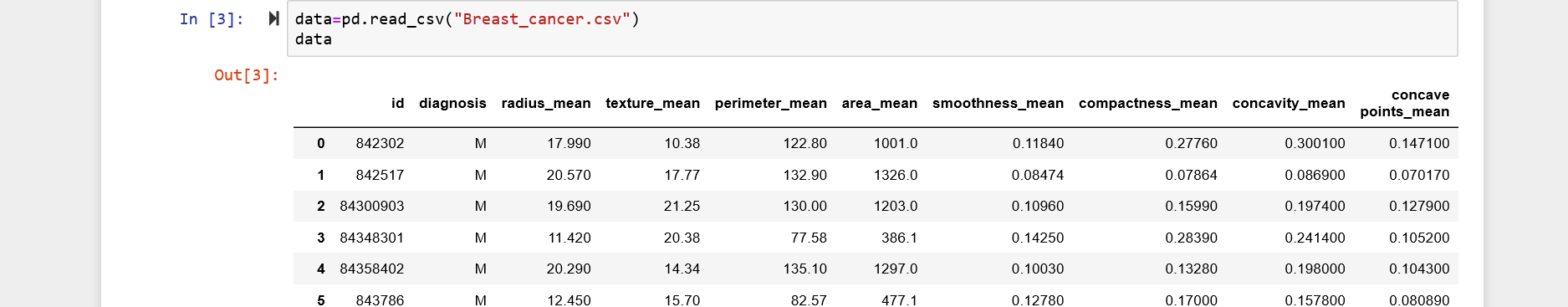
Scikit-learn (formerly scikits.learn) is a [free software](https://en.wikipedia.org/wiki/Free_software" \o "Free software) [machine learning](https://en.wikipedia.org/wiki/Machine_learning" \o "Machine learning) [library](https://en.wikipedia.org/wiki/Library_(computing)" \o "Library (computing)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)" \o "Python (programming language)) programming language. It features various [classification](https://en.wikipedia.org/wiki/Statistical_classification" \o "Statistical classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis" \o "Regression analysis) and [clustering](https://en.wikipedia.org/wiki/Cluster_analysis" \o "Cluster analysis) algorithms including [support vector machines](https://en.wikipedia.org/wiki/Support_vector_machine" \o "Support vector machine), [random forests](https://en.wikipedia.org/wiki/Random_forests" \o "Random forests), [gradient boosting](https://en.wikipedia.org/wiki/Gradient_boosting" \o "Gradient boosting), [k-means](https://en.wikipedia.org/wiki/K-means_clustering" \o "K-means clustering) and [DBSCAN](https://en.wikipedia.org/wiki/DBSCAN" \o "DBSCAN), and is designed to interoperate with the Python numerical and scientific libraries [NumPy](https://en.wikipedia.org/wiki/NumPy" \o "NumPy) and [SciPy](https://en.wikipedia.org/wiki/SciPy" \o "SciPy).

* 1. Import Data

To pull data from a CSV file, you must use the reader function to generate a reader object.

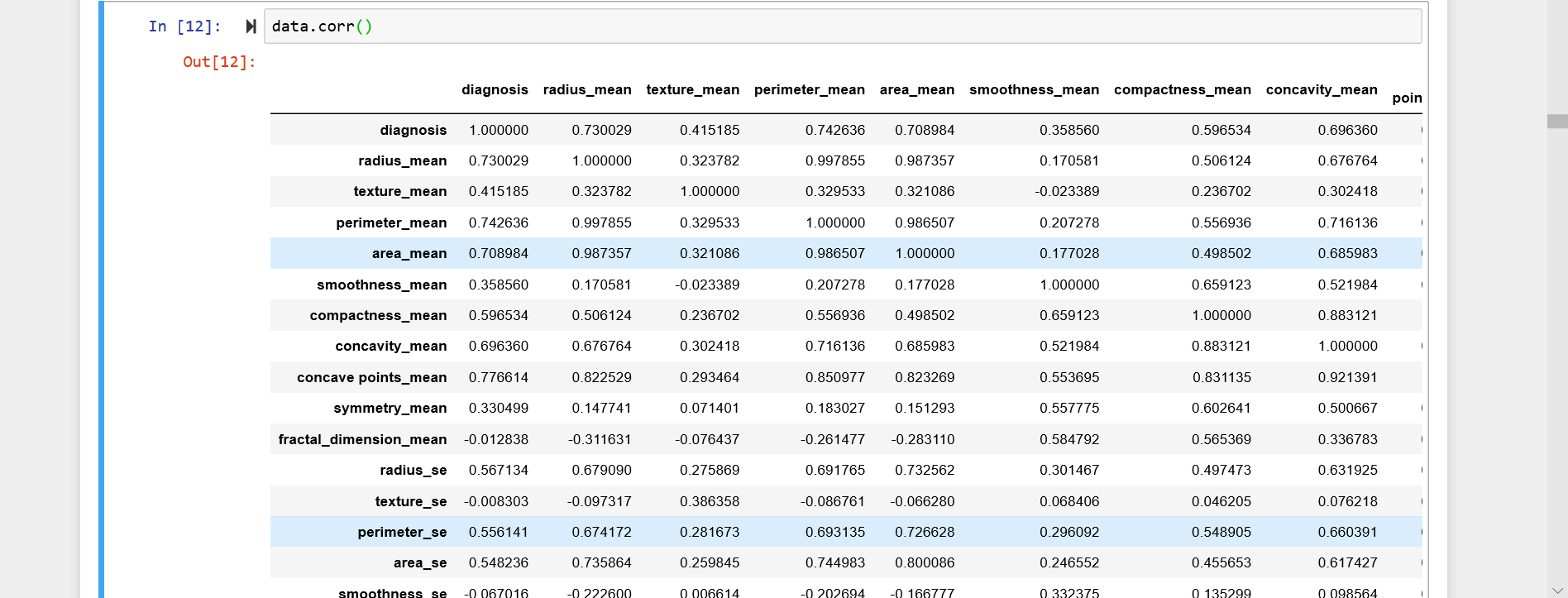
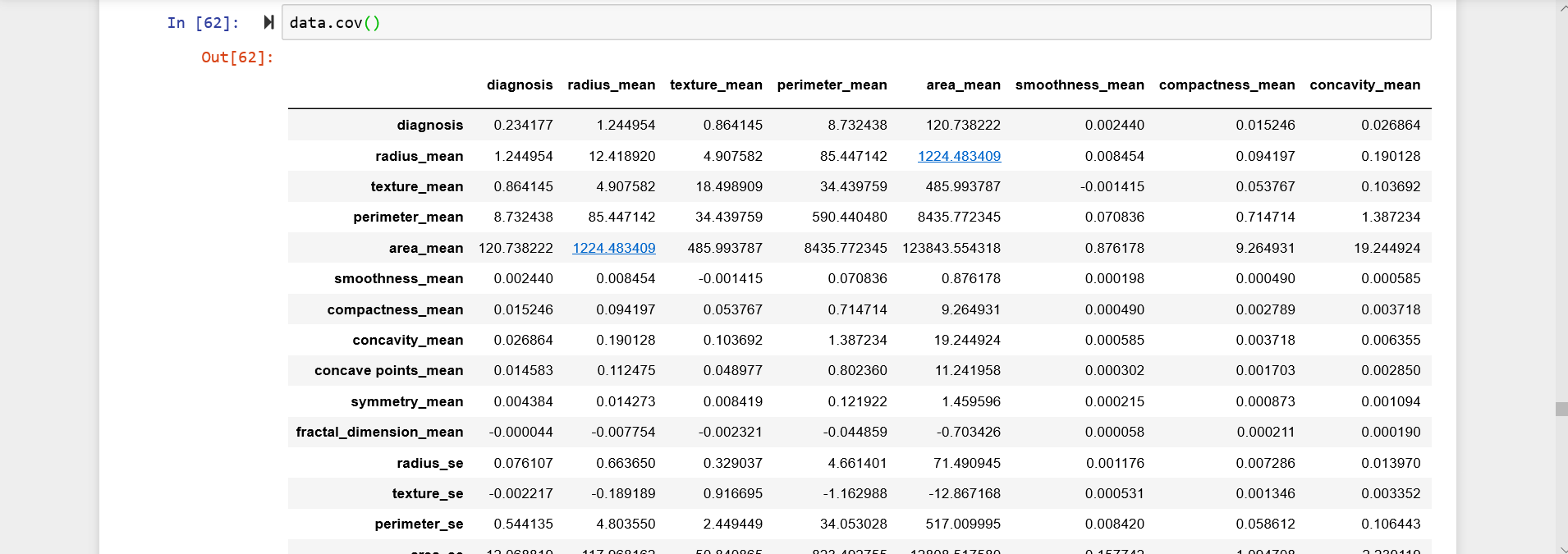
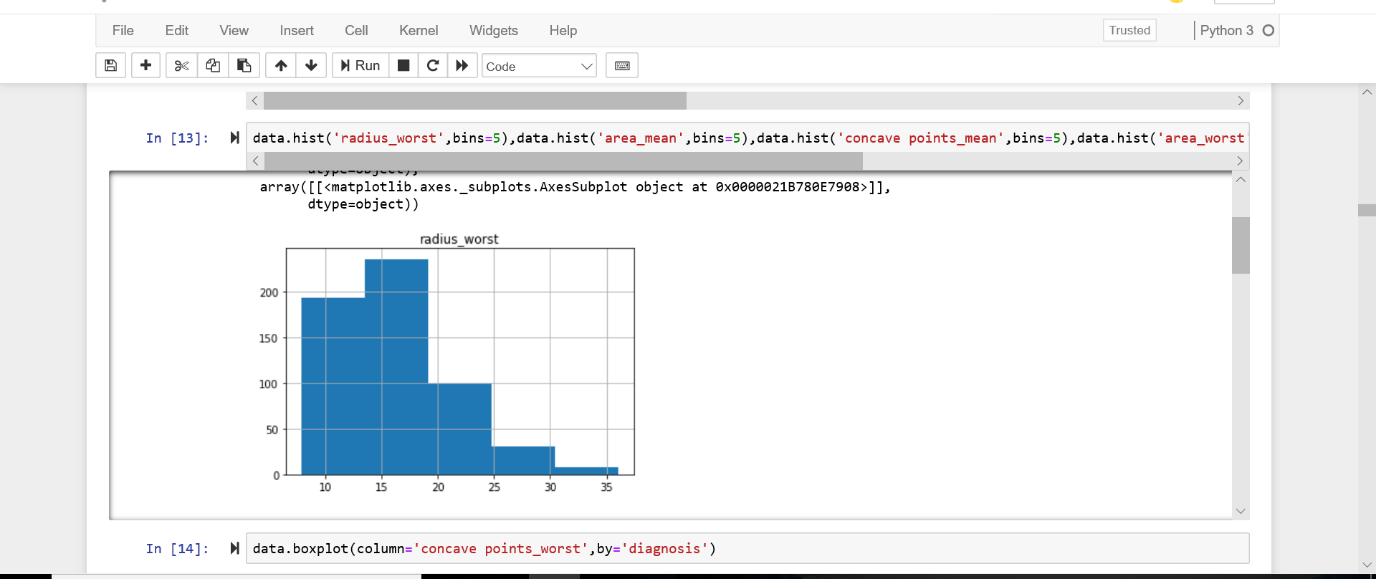
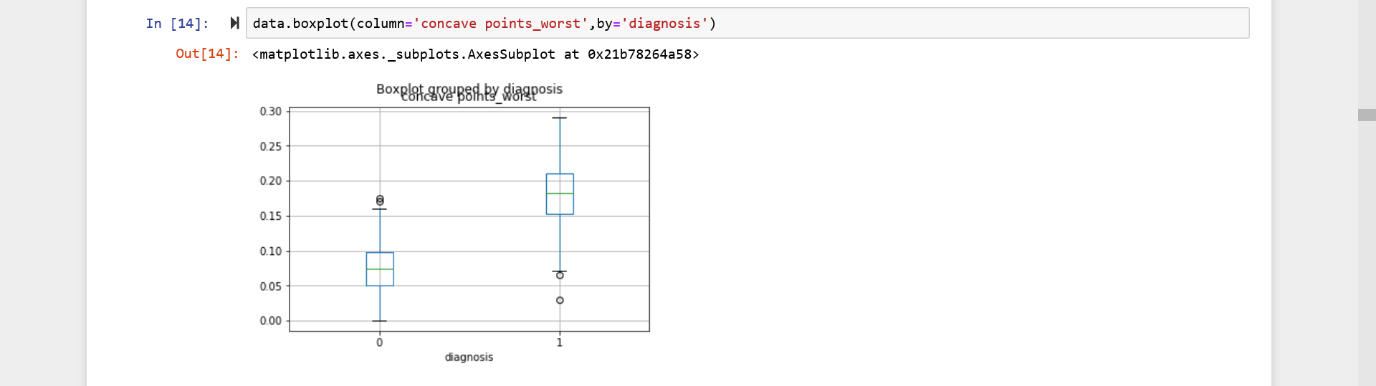
The reader function is designed to take each line of the file and make a list of all columns. Then, you just choose the column you want the variable data for.

It sounds a lot more complicated than it is. To prove it, let’s take a look at an example.

Next, we create the reader object, iterate the rows of the file, and then print them. Finally, we get Data.

* 1. Exploratory Data Analysis

In [statistics](https://en.wikipedia.org/wiki/Statistics" \o "Statistics) and in Machine Learning, exploratory data analysis (EDA) is an approach to analysing [data sets](https://en.wikipedia.org/wiki/Data_set" \o "Data set) to summarize their main characteristics, often with visual methods. A [statistical model](https://en.wikipedia.org/wiki/Statistical_model" \o "Statistical model) can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modelling or hypothesis testing task. Exploratory data analysis was promoted by [John Tukey](https://en.wikipedia.org/wiki/John_Tukey" \o "John Tukey) to encourage statisticians to explore the data, and possibly formulate hypotheses that could lead to new data collection and experiments. EDA is different from [initial data analysis (IDA)](https://en.wikipedia.org/wiki/Data_analysis" \l "Initial_data_analysis" \o "Data analysis), which focuses more narrowly on checking assumptions required for model fitting and hypothesis testing, and handling missing values and making transformations of variables as needed. EDA encompasses IDA. There are many subparts in Exploratory Data Analysis and some of them are,

* + 1. Correlation
    2. Covariance
    3.  Histogram
    4.  Boxplot
    5.  Pie chart
  1. Cleaning of Data

The cleaning of data is very much important to increase all the parameters of the data. Cleaning up of data means the removal of null values from the data that makes it more robust to perform different types of operations.

* 1. Splitting of Data

When you’re working on a model and want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset. For this, you’ll a dataset which is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase.

In such cases, the obviously solution is to split the dataset you have into two sets, one for training and the other for testing; and you do this before you start training your model.

But the question is, how do you split the data? You can’t possibly manually split the dataset into two. And you also have to make sure you split the data in a random manner. To help us with this task, the SciKit library provides a tool, called the Model Selection library. There’s a class in the library which is, aptly, named ‘[train\_test\_split](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html" \t "_blank).’ Using this we can easily split the dataset into the training and the testing datasets in various proportions.

There are a few parameters that we need to understand before we use the class:

**test\_size** — This parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, the dataset will be split 50% as the test dataset. If you’re specifying this parameter, you can ignore the next parameter.

**train\_size** — You have to specify this parameter only if you’re not specifying the test\_size. This is the same as test\_size, but instead you tell the class what percent of the dataset you want to split as the training set.

**random\_state** — Here you pass an integer, which will act as the seed for the random number generator during the split. Or, you can also pass an instance of the RandomState class, which will become the number generator. If you don’t pass anything, the RandomState instance used by np.random will be used instead.

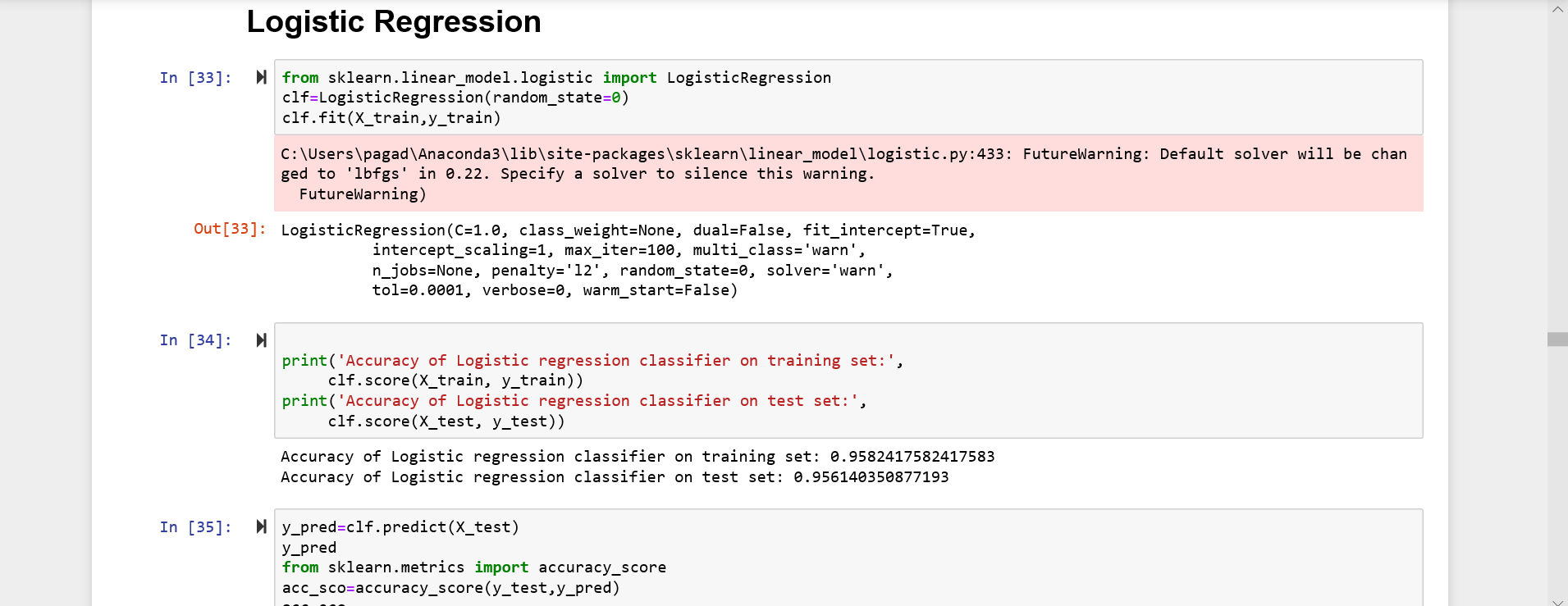
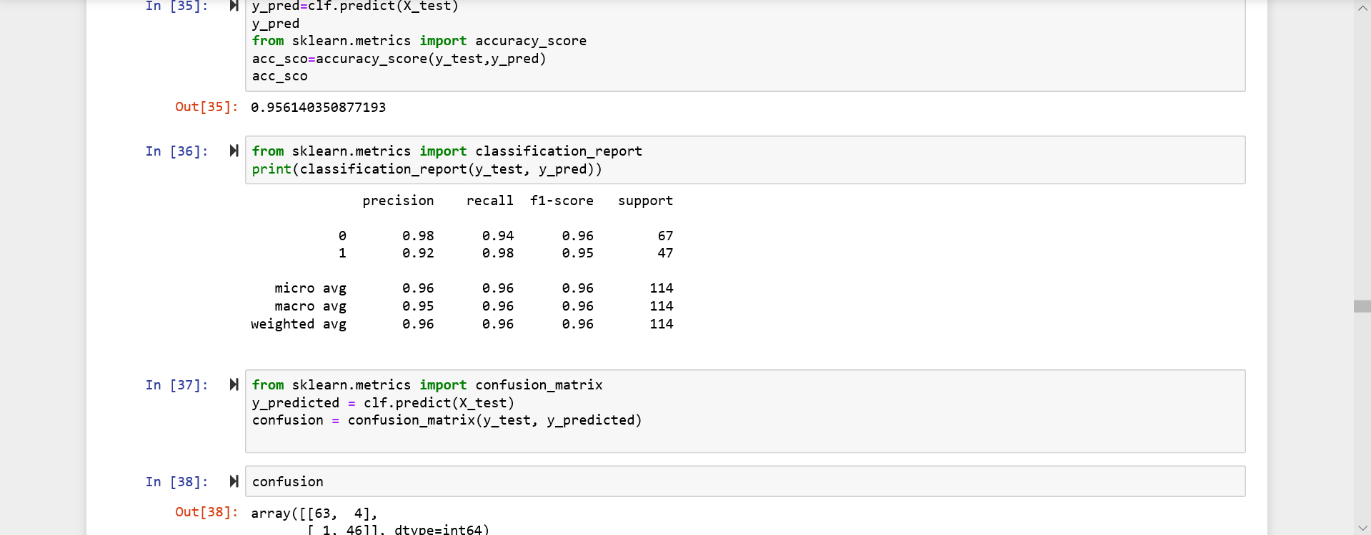
* 1. Algorithm

As the data is read into the python notebook using pandas and os, the data is cleaned i.e. the missing values are replaced and now the data is ready to be used in an algorithm for training and testing the data and predicting new values for the dependent variable. Some of the Algorithms are,

* + 1. Logistic Regression

**Definition:**Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function.

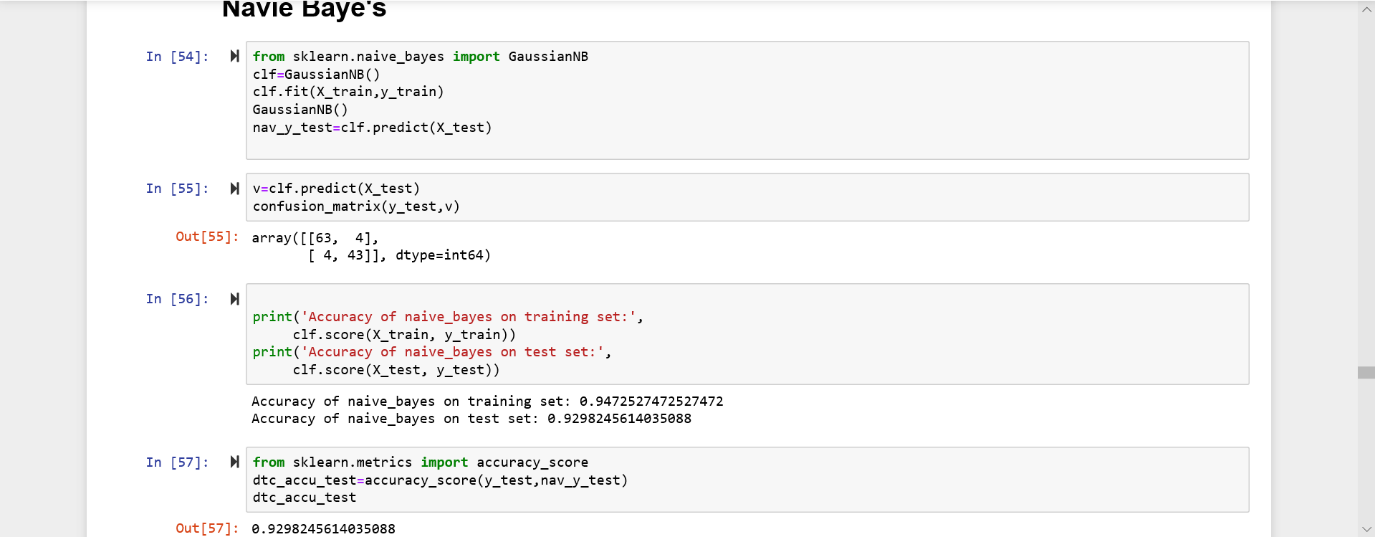
**Advantages:** Logistic regression is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.

**Disadvantages:** Works only when the predicted variable is binary, assumes all predictors are independent of each other, and assumes data is free of missing values.

* + 1. Naive Bayes

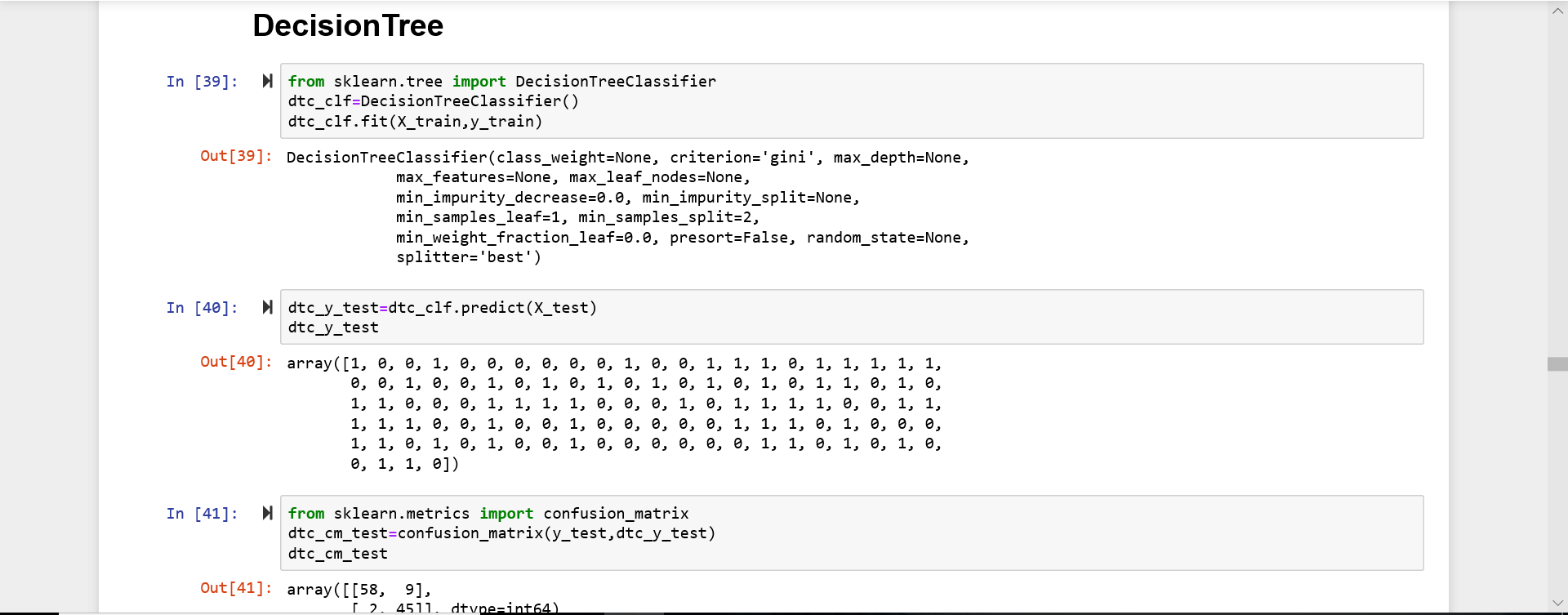
**Definition:**Naive Bayes algorithm based on Bayes’ theorem with the assumption of independence between every pair of features. Naive Bayes classifiers work well in many real-world situations such as document classification and spam filtering.

**Advantages:**This algorithm requires a small amount of training data to estimate the necessary parameters. Naive Bayes classifiers are extremely fast compared to more sophisticated methods.

**Disadvantages:**Naive Bayes is is known to be a bad estimator.

* + 1. Decision Tree

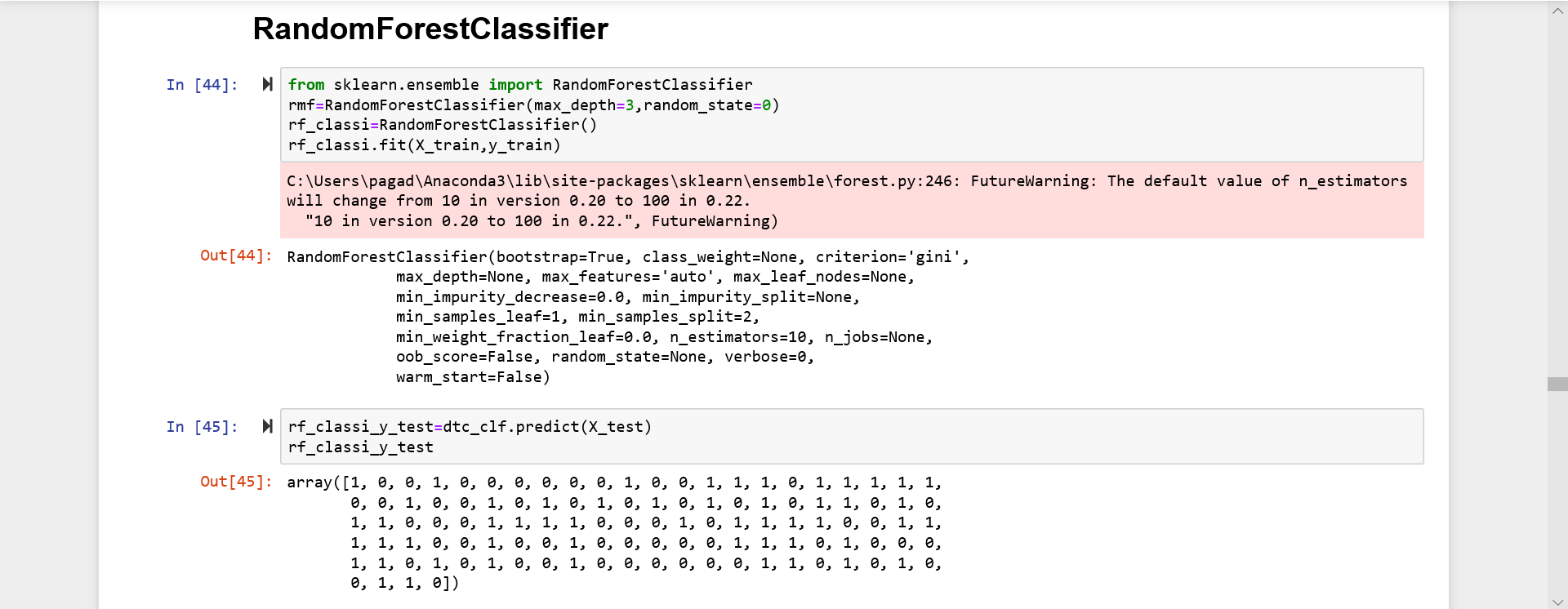
**Definition:** Given a data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data.

**Advantages:** Decision Tree is simple to understand and visualise, requires little data preparation, and can handle both numerical and categorical data. **Disadvantages:**Decision tree can create complex trees that do not generalise well, and decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

* + 1. Random Forest

**Definition:**Random forest classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

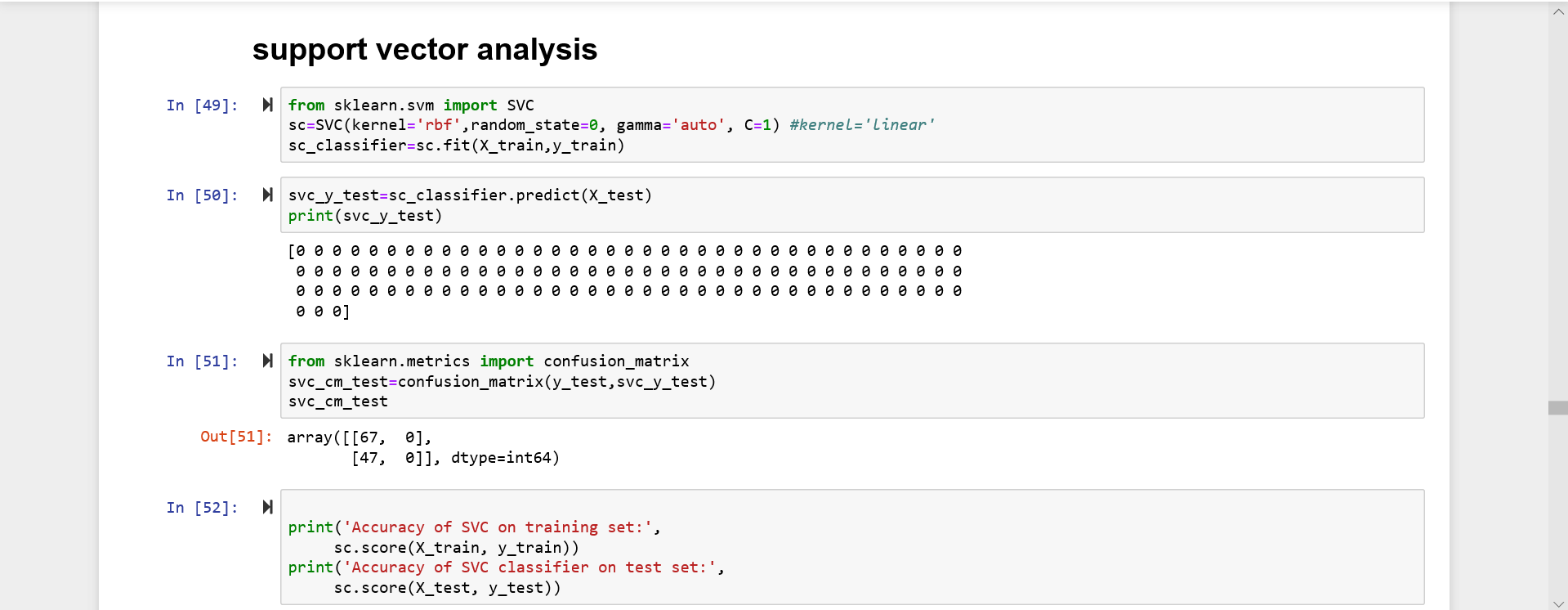
**Advantages:**Reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases.

**Disadvantages:**Slow real time prediction, difficult to implement, and complex algorithm.

* + 1. Support Vector Classification

**Definition:**Support vector machine is a representation of the training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

**Advantages:**Effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient.

**Disadvantages:**The algorithm does not directly provide probability estimates; these are calculated using an expensive five-fold cross-validation.

* + 1. KNN

**Definition:** Neighbours based classification is a type of lazy learning as it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbours of each point.

**Advantages:** This algorithm is simple to implement, robust to noisy training data, and effective if training data is large.

**Disadvantages:** Need to determine the value of K and the computation cost is high as it needs to computer the distance of each instance to all the training samples.

* 1. Evaluation

****As the algorithm is successfully applied and we found the prediction values successfully. Now the question arises that to which extent the predicted values are correct. To Measure that we find the accuracy of that particular algorithm using accuracy score from sklearn and confirm with the accuracy and decide that it may happen or not. Basically, an accuracy of 80% and above are considered to be the best model. In the same way the sensitivity and specificity are also measured to determine some other specific factors of the data and the predicted values. Accuracy is done as follows,

1. **Findings and Suggestions**

The data that we processed is about Breast Cancer prediction collected from a trusted source. The data was imported and then cleaned up. The data was then studied using Exploratory data analysis and the relation between every factor was observed. All the columns of the data except Diagnosis which was the target were taken as the predictors as this is the health care industry, definitely each and every factor can affect the health of the human so there is no was of neglecting/dropping a column whereas we can unconditionally drop the id which has no relation. Then the data is split into training and testing data with a ratio of 80% to the training data and 20% to the test data, which is used to train and predict values.

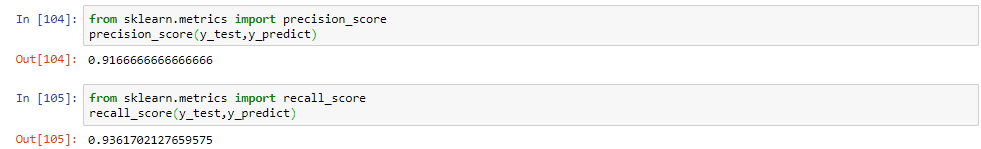
Different classification Algorithms were applied on the data and the Diagnosis values were predicted. To know exactly which model produced the most accurate values we found the testing and training accuracies of the algorithms. The following table on the next page consists the accuracies in percentage. After analysing the algorithms specifically, it is found that KNN algorithm produces the most accurate values as the percentage values of the training and testing data are very closer and hence the KNN algorithm is then used to predict values for the split of data of 90:10, 70:30 , 60:40, 50:50.

|  |  |  |
| --- | --- | --- |
| S. No | Algorithms | Accuracy1 |
| **1** | Logistic Regression | Train - 94% |
|  |  | Test – 92% |
| **2** | Decision Tree | Train – 100% |
|  |  | Test – 88% |
| **3** | Random Forest | Train – 99% |
|  |  | Test – 93% |
| **4** | Support Vector Classification | Train – 100% |
|  |  | Test – 58% |
| **5** | Naive Bayes | Train – 94% |
|  |  | Test – 92% |
| **6** | K- Nearest Neighbours | Train – 94% |
|  |  | Test – 93% |

After the algorithm was applied to different ratios of training and testing data the following are the accuracies respectively,

|  |  |  |
| --- | --- | --- |
| S. No | Ratio of Split | Accuracy |
| **1** | 90% - 10% | Train – 95% |
|  |  | Test – 89% |
| **2** | 70% - 30% | Train – 93% |
|  |  | Test – 94% |
| **3** | 60% - 40% | Train – 93% |
|  |  | Test – 95% |
| **4** | 50% - 50% | Train – 94% |
|  |  | Test – 93.8% |

The next Code shows the specificity and the sensitivity of the data using kNN algorithm,



1. **Conclusion**

To analyse medical data, various machine learning methods are available. An important challenge in machine learning areas is to build accurate and computationally efficient classifiers for Medical applications. In this study, we employed four main algorithms such as SVM, NB, k-NN, logistic, Decision tree and Random Forest on the Breast\_Cancer dataset. We tried to compare efficiency and effectiveness of those algorithms in terms of accuracy, sensitivity and specificity to find the best classification accuracy. Of kNN reaches and accuracy of 93.8% and outperforms, therefore, all other algorithms. In conclusion, kNN has proven its efficiency in Breast Cancer prediction and diagnosis and achieves the best performance in terms of accuracy and low error rate.

1. **References**

7.1 <https://archive.ics.uci.edu/ml/datsets/Breast+cancer+Wisconsin+(Diagnostic)>

7.2 [www.wikipedia.com](http://www.wikipedia.com)

7.3 [www.geeksforgeeks.com](http://www.geeksforgeeks.com)

7.4 [www.tutorialspoint.com](http://www.tutorialspoint.com)

7.5 [www.stackoverflow.com](http://www.stackoverflow.com)

7.6 [www.python.org](http://www.python.org)