

**MALIGNANT COMMENTS CLASSIFIER**

**Submitted by:**

*SHIKHA KAMAL*

**ACKNOWLEDGMENT**

The proliferation of social media enables people to express their opinions widely online. However, at the same time, this has resulted in the emergence of conflict and hate, making online environments uninviting for users. Although researchers have found that hate is aproblem across multiple platforms, there is a lack of models for online hate detection.

This project is more about exploration, feature engineering and classification that can be done on this data. Since the data set is huge and includes many categories of comments, we can do good amount of data exploration and derive some interesting features using the comments text column available.

You need to build a model that can differentiate between comments and its categories.

**INTRODUCTION**

* Business Problem Framing

Online hate, described as abusive language, aggression, cyber bullying, hatefulness and many others has been identified as a major threat on online social media platforms. Social media platforms are the most prominent grounds for such toxic behaviour.

There has been a remarkable increase in the cases of cyber bullying and trolls on various social media platforms. Many celebrities and influences are facing backlashes from people and have to come across hateful and offensive comments. This can take a toll on anyone and affect them mentally leading to depression, mental illness, self-hatred and suicidal thoughts.

Internet comments are bastions of hatred and vitriol. While online anonymity has provided a new outlet for aggression and hate speech, machine learning can be used to fight it. The problem we sought to solve was the tagging of internet comments that are aggressive towards other users..

* Conceptual Background of the Domain Problem

Our goal is to build a prototype of online hate and abuse comment classifier which can used to classify hate and offensive comments so that it can be controlled and restricted from spreading hatred and cyber bullying.

* Review of Literature

Internet comments are bastions of hatred and vitriol. While online anonymity has provided a new outlet for aggression and hate speech, machine learning can be used to fight it. The problem we sought to solve was the tagging of internet comments that are aggressive towards other users. This means that insults to third parties such as celebrities will be tagged as un-offensive

* Motivation for the Problem Undertaken

This project is more about exploration, feature engineering and classification that can be done on this data. Since the data set is huge and includes many categories of comments, we can do good amount of data exploration and derive some interesting features using the comments text column available.

You need to build a model that can differentiate between comments and its categories.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

We start with the library calling, basic libraries like pandas , numpy , matplotlib , seaborn and warnings filters.

Further additional we run the observation commands of data sets like head, tail, shape, data types, null function, columns, sample, info functions. Through these functions we get to know the data sets values with their different varying values which are highlighted in every different command.

Then taking the additional step, then we come to the analytics part or can say some mathematical calculations done in internally by python and display the results values. Here we add the describe functions which show some calculations among the different columns in data sets and statistics calculated.

The describe functions show the different values like maximum value, minimum value, mean, standard deviation, counts and different percent percentile like 25 %, 50 %, 75 % percentile.

Some observation done in describe function table:

* The count of all the columns are same that is 159571 which is equivalent to the number of rows present. All rows contains values and no null values except comment text column.
* In comment text column , there are 132 rows whose having null value .
* Describe function only calculates the numerical columns and ignore the string type columns.
* It shows the mean of different columns and standard deviation and there is vast difference between mean and std deviation which can relate to spreading of data in normal distribution curve .
* Data Sources and their formats

The data set contains the training set, which has approximately 1,59,000 samples and the test set which contains nearly 1,53,000samples. All the data samples contain 8 fields which includes ‘Id’, ‘Comments’, ‘Malignant’, ‘Highly malignant’, ‘Rude’, ‘Threat’, ‘Abuse’ and ‘Loathe’.

The label can be either 0 or 1, where 0 denotes a NO while 1 denotes a YES. There are various comments which have multiple labels. The first attribute is a unique ID associated with each comment.

The data set includes:

* **Malignant:** It is the Label column, which includes values 0 and 1, denoting if the comment is malignant or not.
* **Highly Malignant:** It denotes comments that are highly malignant and hurtful.
* **Rude:** It denotes comments that are very rude and offensive.
* **Threat:** It contains indication of the comments that are giving any threat to someone.
* **Abuse:** It is for comments that are abusive in nature.
* **Loathe:** It describes the comments which are hateful and loathing in nature.
* **ID:**It includes unique Ids associated with each comment text given.
* **Comment text:** This column contains the comments extracted from various social media platforms.
* Data Preprocessing Done

As we know comment text columns have null values present and we use Impute library function to fill that comment using mode function .

Label encoder used to provide the uni-code to the data values in data sets .

After providing the label encoder all columns converted to numerical data form and contain the integer data types.

In pre-processing the data values is being observed and some calculations is being provided like statistics and correlation . the imputer function also used to input the values and label encoder to gave integer code to every data value .

* Data Inputs- Logic- Output Relationships

The output columns is malignant column which is a label column and it has 2 type of output 0 or 1. So target variable is classifier type , and it depends upon the input variables for rest of the left columns. The target variable is dependent variable and all input variables is independent variables .

It is classifier relations between the input and output which need to trained in classifier data model to have the output.

* Hardware and Software Requirements and Tools Used

Here we check for outliers also in data sets, outliers are those data points which lie outside the normal distribution curve . the outliers are recognized by plotting the graph called box plot . if the outliers lie outside the box plot then we can say that outliers are present.

Most of the columns are having outliers in it, then we can also remove this by having the z score method. Z score method helps in removing the outliers , and we can remove the rows count which contains the outliers in it.

The actual count of data sets , 159571 but after removing the outliers the rows count is 143346.

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

First we start with the random state, we find the maximum random state which contains the maximum percentage while running the model. Then we save the random state value and utilize in the train test split method.

Train test split method splits the data sets into input and output variables . the max random state is value is 65 and its maximum score in percent is 90.77 % . now this random state value is being used for further training of data .

* Testing of Identified Approaches (Algorithms)

We use the 5 various models for training and 2 models for testing. The models which used for training are:

* Gaussian NB model
* Logistic model
* Decision tree classifier
* Random forest classifier
* SVM ( support vector machines )

Algorithms used for testing the models are:

* Cross validation score
* Grid search CV
* Run and Evaluate selected models

Results of different models are ( accuracy score ) :

* Logistic regression model having 90.76 % accuracy score
* Gaussian NB model having 90.76 % accuracy score
* Decision tree classifier having 92.60 % accuracy score
* Random forest classifier having 95.78 % accuracy score
* Svm ( support vector machines ) having 90.76 % accuracy score
* Key Metrics for success in solving problem under consideration

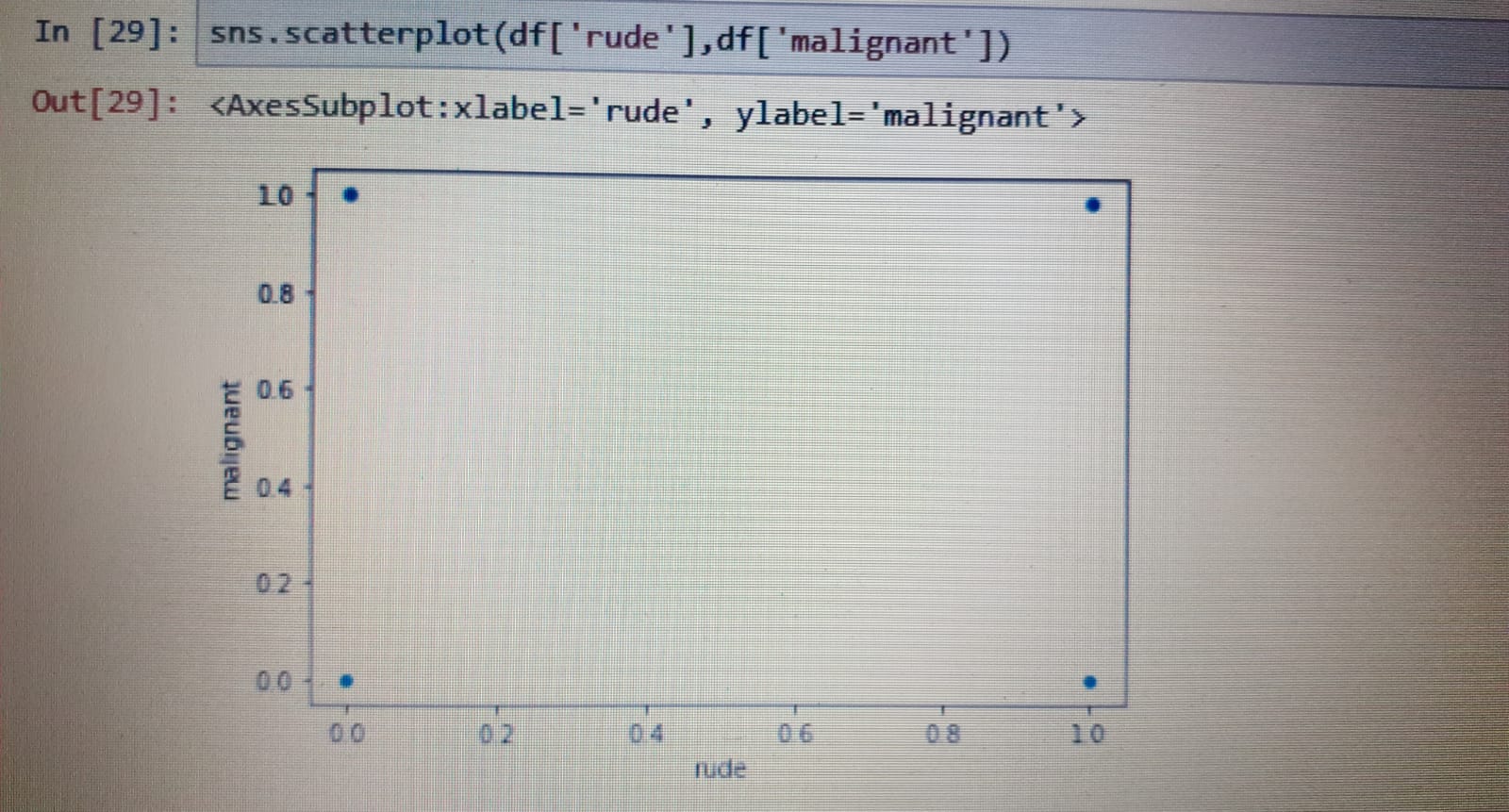
Cross validation score used to cross verify which model is best trained and having best accuracy score .

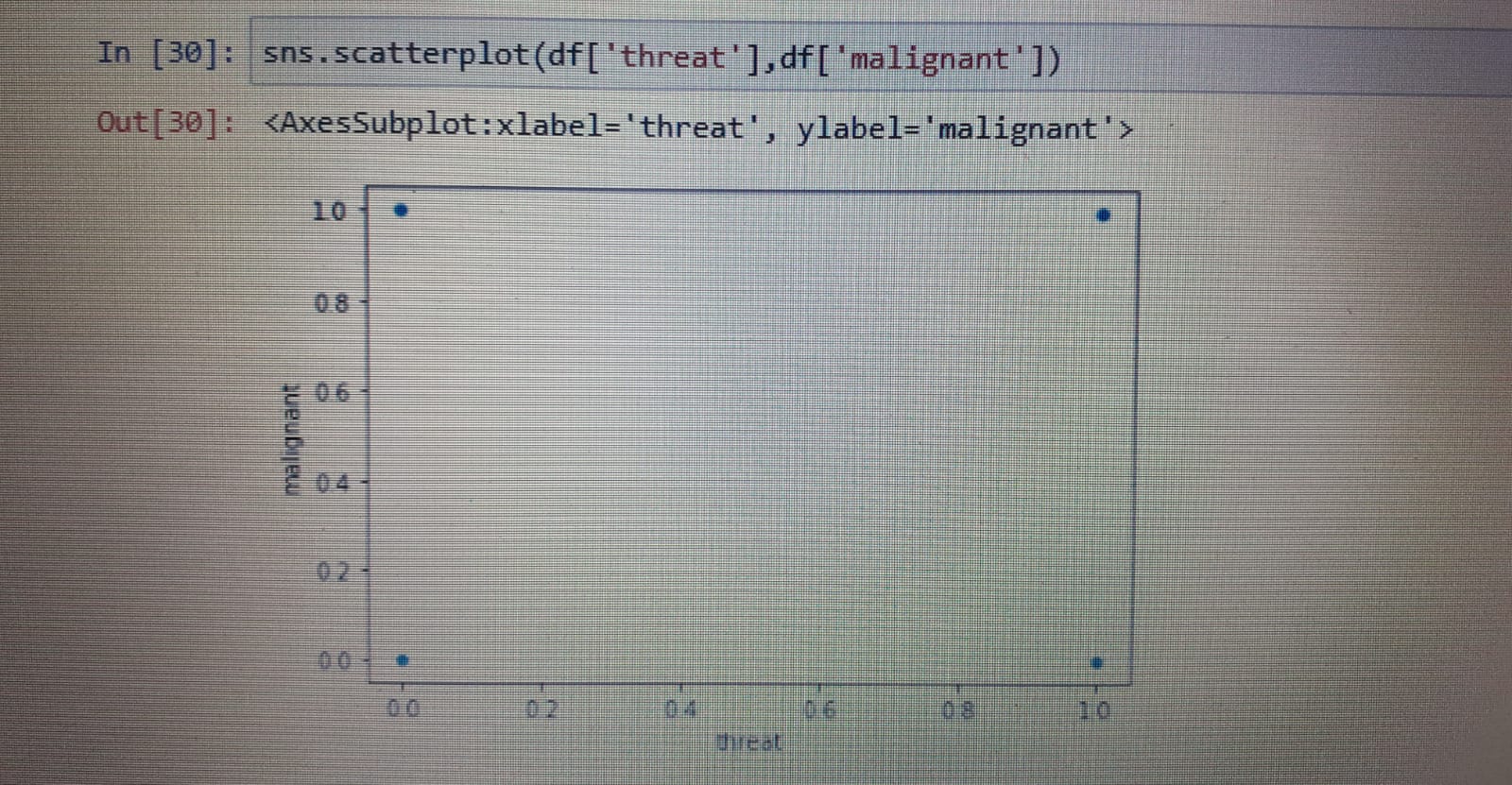
The results of best cross validation score is:

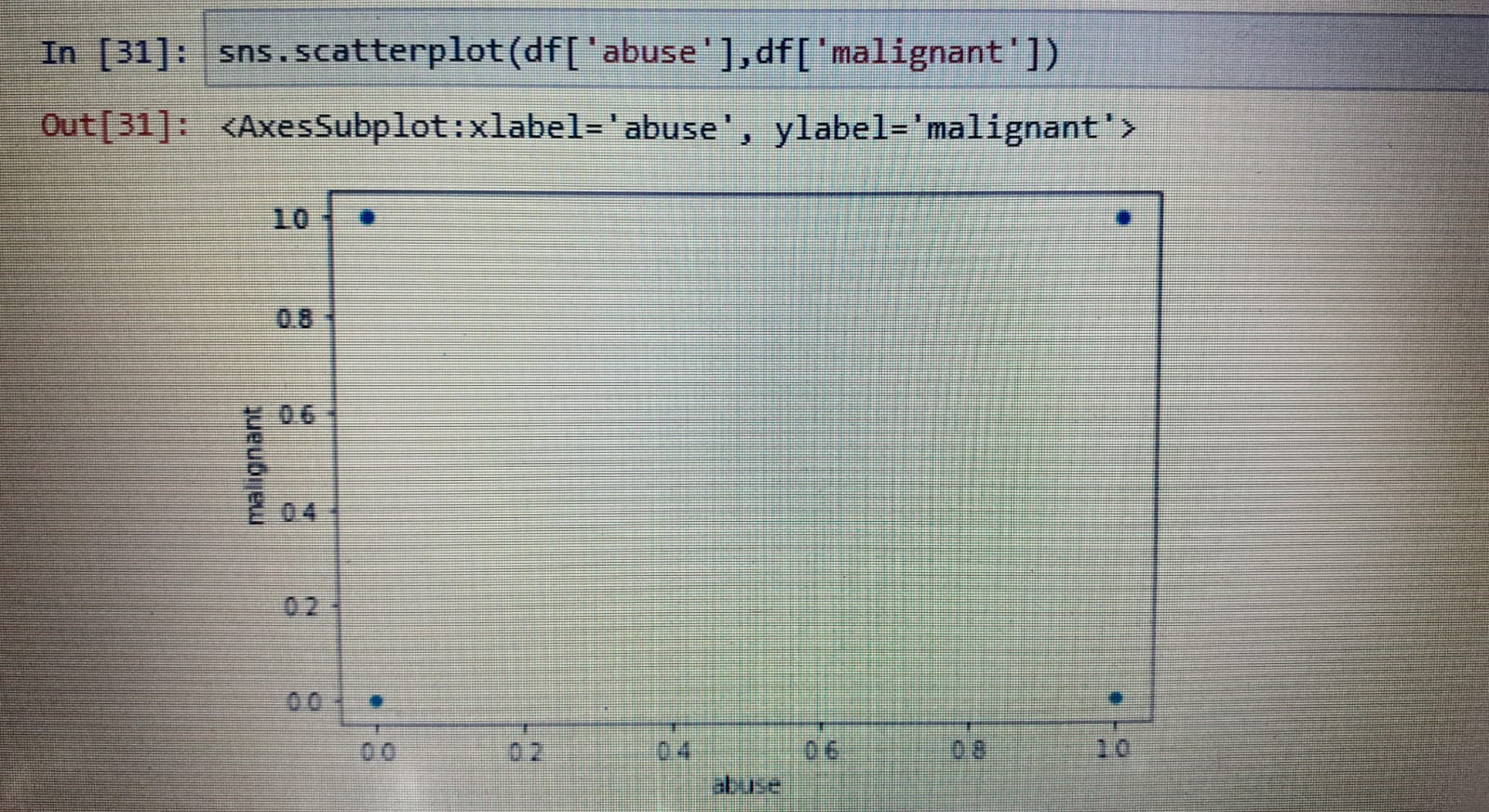
* Logistic regression model having 90.41 % cross validation score
* Gaussian NB model having 90.41 % cross validation score
* Decision tree classifier having 92.53 % cross validation score
* Random forest classifier having 95.65 % cross validation score
* Svm (support vector machine ) having 90.41 % cross validation score
* Visualizations

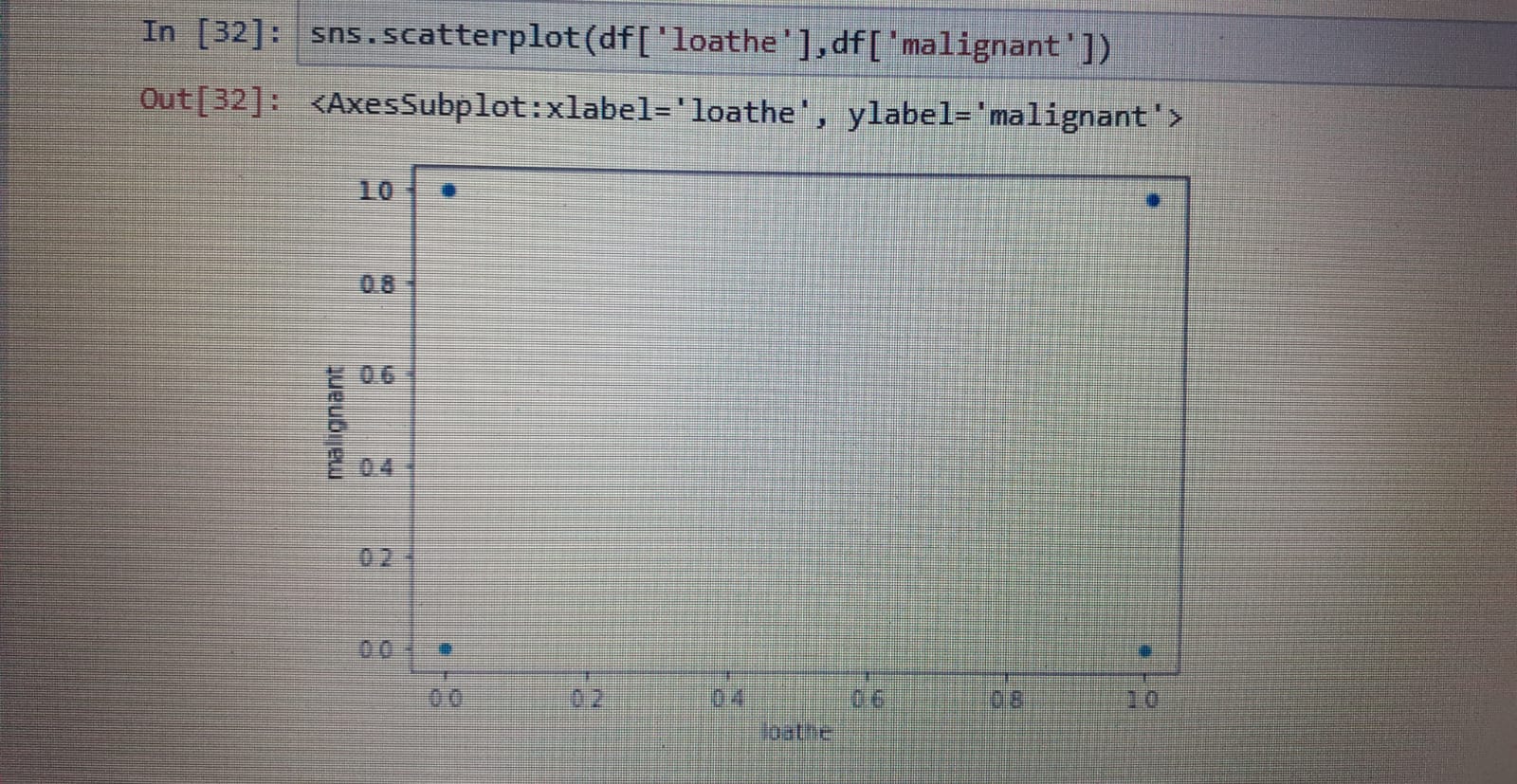
We use the different graphs to have certain observation for data set values, like we use scatter plot, box plot, count plot, distribution plot.

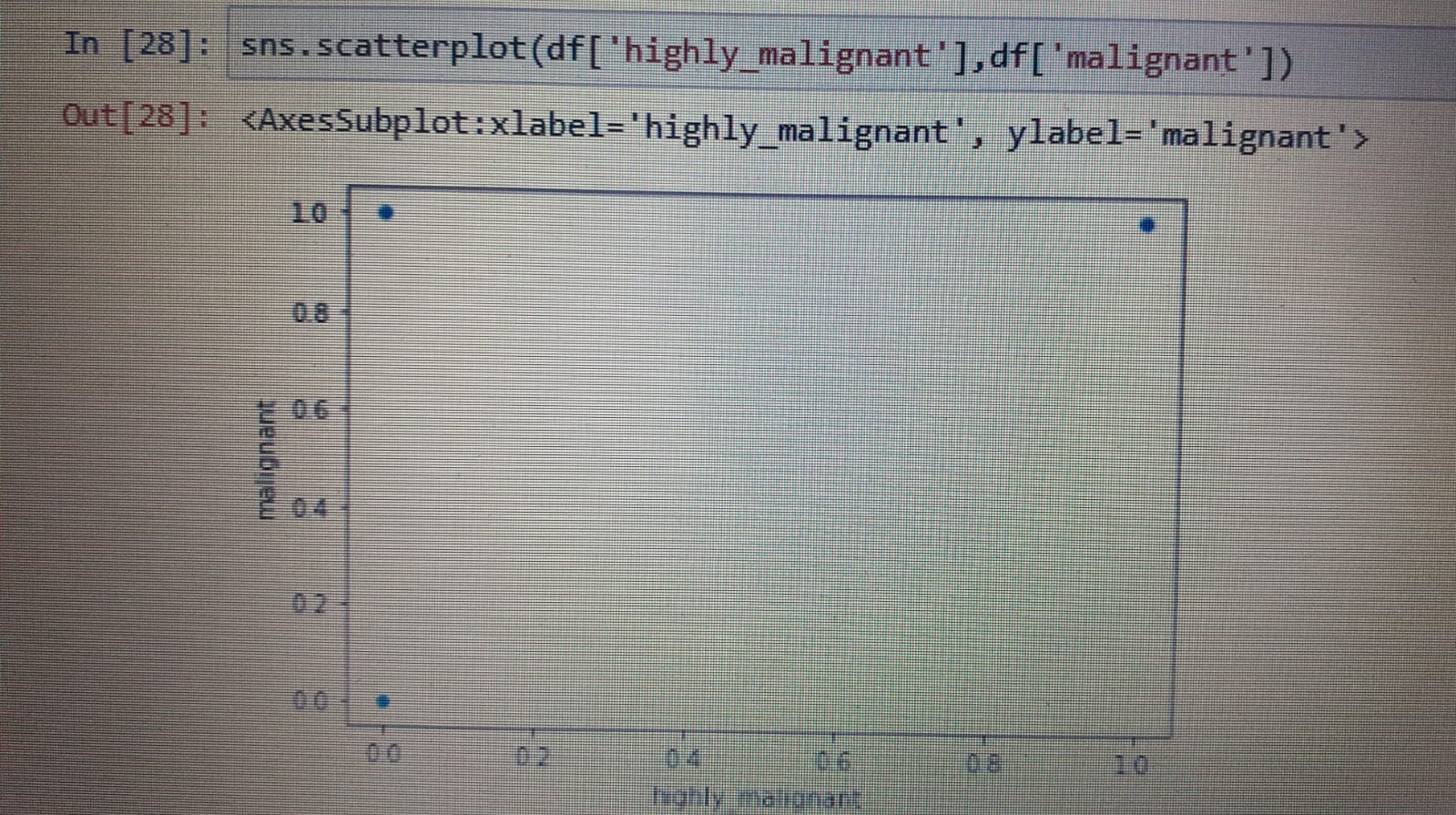
There are many other graphs which we can also use , like bar graph, histogram graph, violin graph, etc.

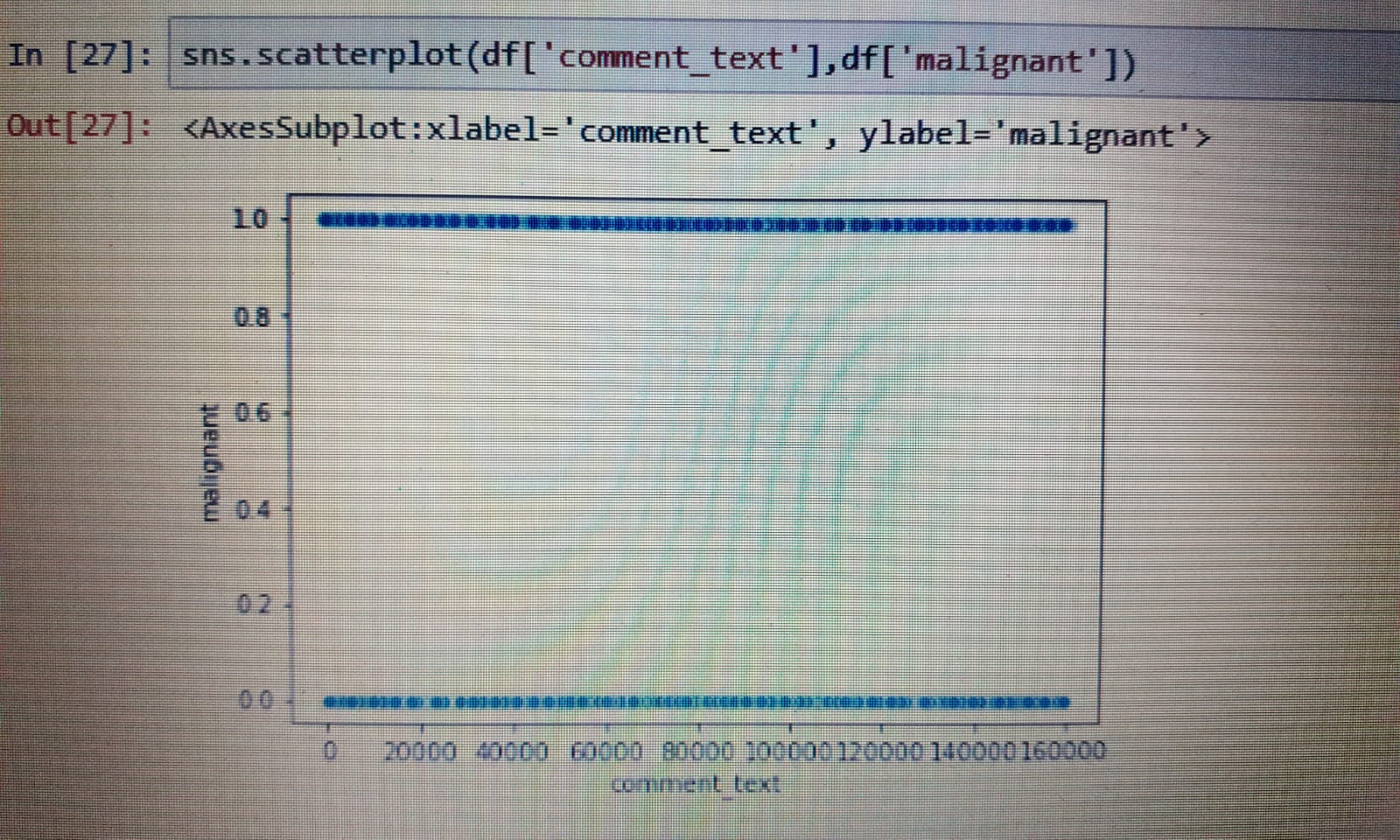


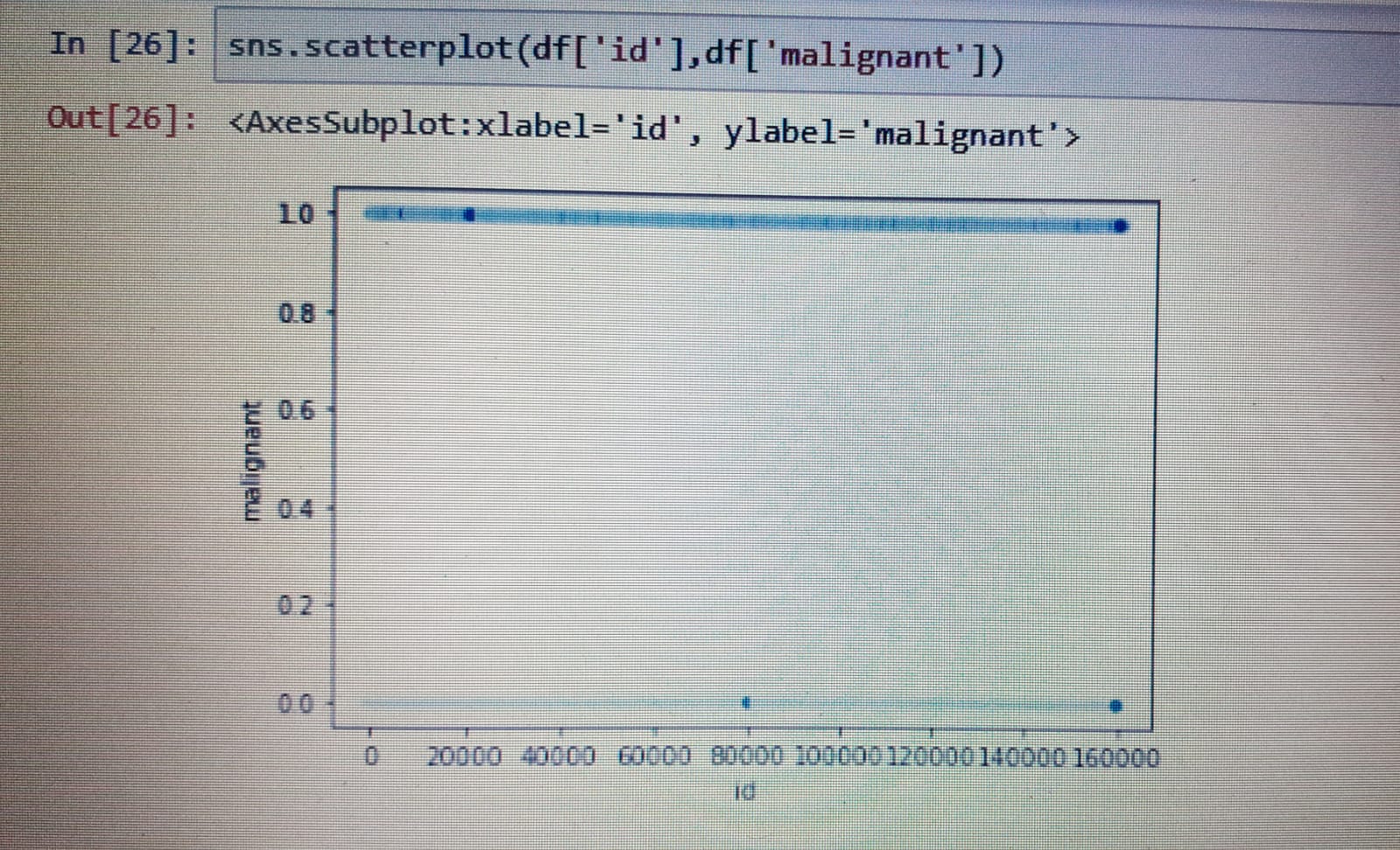












**CONCLUSION**

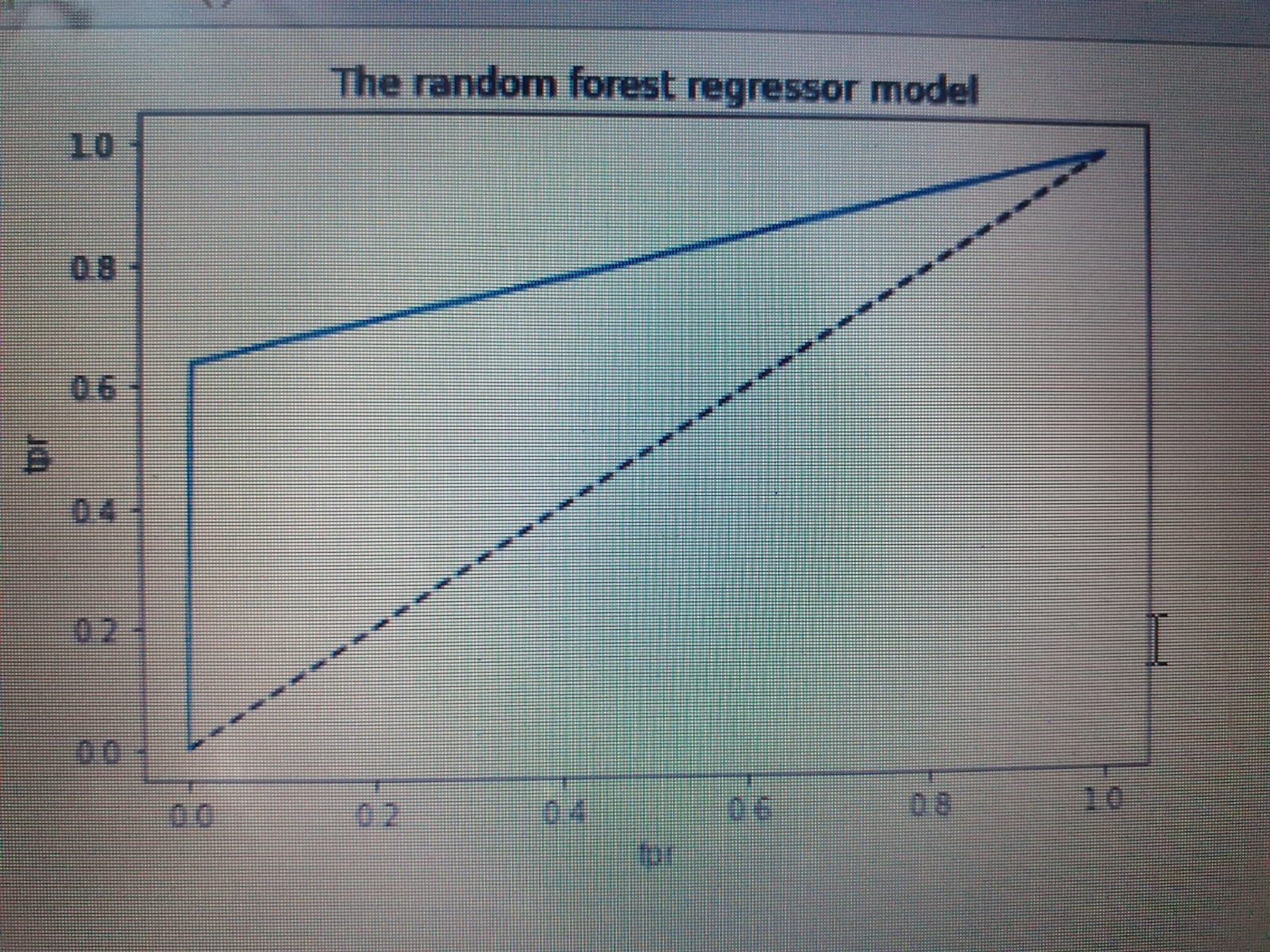
* Key Findings and Conclusions of the Study

After cross validation , hyper parameter tuning is next step. This also used for prediction and find out the best parameter from the best scoring model. And then again trained the model and once again the model is predicted for the best results .

In hyper parameter tuning, certain steps conclude:

* The best fitted model is being carried forward and here random forest is best fitted.
* Parameters name are max depth, min samples splits, max features
* Grid search cv passed with parameters of model name, parameters for random model, n jobs , scoring and cross validation count
* Then grid search cv is trained for data model and get prediction for the next step.
* After training the model , then best parameters are displayed which is best suited.
* Then we continue with the model that is random forest, and prediction made.
* The best accuracy score is 95.99 %
* Learning Outcomes of the Study in respect of Data Science

We run the roc auc curve of random forest classifier model and its percentage is coming over 60 %in true positive rate and false positive rate.



The best fitted model is random forest and we cross validify also . the accuracy score of random forest model is 95.99 % and it reduced the loss of data also as it score good accuracy score.