**AIML PROJECT**

**FAKE NEWS DETECTION**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION**

**Submitted To: Dr. Ravi Kumar Jatoth**

**Submitted By:**

**1.Name: Mulugu Sai Sree Vadan Sharma**

**Roll No.: 21ECB0B32**

**Branch: ECE**

**Sec: B**

**2. Name: Mankala Srujan**

**Roll No.: 21ECB0B34**

**Branch: ECE**

**Sec: B**

CONTENTS

* ABSTRACT
* INTRODUCTION
* THEORY
* METHODOLOGY
* CODE AND RESULTS
* CONCLUSION

ABSTRACT

Fake news is simply content that has been prepared using fake information in order to deceive readers and propagate misleading claims. Our society is suffering from a rising problem with fake news and a lack of faith in the media. The phrase "fake news" became widely used to characterise the problem, specifically to describe pieces that were produced primarily for the aim of generating revenue from page views but contained factual errors and mis information. In this work, an accurate prediction of the possibility that a particular item is false news is sought after. However, in order to address this issue, it is essential to comprehend what false news is.

The project develops applications of NLP (natural language processing) techniques for identifying "fake news," which refers to news reports that are inaccurate and originate from unreliable sources. The majority of so-called "fake news" is first spread through social media platforms like Facebook and Twitter before making its way onto traditional media channels like television and radio news. In response, the data science community has begun to address the issue. The "false news" Kaggle competition exists. The performance of a false news classifier is examined through the results of a fake news detection research, which are provided in this work.

INTRODUCTION

Access to news information has become considerably more convenient and easier thanks to the internet and social media. The growth of mobile devices has made it much simpler for Internet users to follow the events of interest in online form.  
However, huge opportunities often provide great problems. The media has a significant impact on society, and as is frequently the case, someone seeks to use this fact. Information may occasionally be manipulated by the media in various ways to achieve certain objectives. As a result, news pieces are created that are partially or even entirely fraudulent. There are also several websites that practically solely provide false news. Therefore, false news is both a worldwide issue and a global problem.

Many experts think that artificial intelligence and machine learning may be used to combat the problem of false news. There is a reason for this; lately, artificial intelligence algorithms have begun to perform significantly better on many classification issues (image recognition, speech detection, and so on), thanks to cheaper technology and more readily accessible larger datasets.  
A number of important articles have been written regarding automated deception detection. The writers give a broad review of the approaches to the problem in the article. The authors outline their technique for identifying false news based on comments left on a given piece of news on microblogs. In fact, the authors create two systems for deception detection based on support vector machines and naive Bayes classifier, respectively (this approach is also utilised in the system presented in this study). They get the data by directly asking respondents if certain statements like friendship, the death sentence, and abortion are true or untrue. The system's detection accuracy is somewhere in the range of 90%.

The naive Bayes classifier, one of the artificial intelligence algorithms, is used in this article to explain a straightforward false news detection technique. A manually annotated news dataset is used in the study to investigate how well this strategy performs for this specific challenge and to determine whether or not artificial intelligence should be used to detect false news. The distinction between this article and others on related topics is that the naive Bayes classifier was specifically used in this paper for fake news detection. Additionally, the developed system was tested on a recent data set, which provided an opportunity to assess how well it performed on current data.

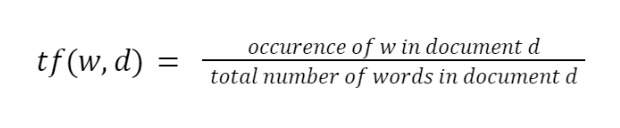
THEORY:

TFIDF VECTORIZER:

* Most machine learning algorithms are fulfilled with mathematical things such as statistics, algebra, calculus and etc. They expect the data to be numerical such as a 2-dimensional array with rows as instances and columns as features.
* The problem with natural language is that the data is in the form of raw text, so that the text needs to be transformed into a vector.
* The process of transforming text into a vector is commonly referred to as text vectorization.
* It’s a fundamental process in natural language processing because none of the machine learning algorithms understand a text, not even computers.
* Text vectorization algorithm namely TF-IDF vectorizer, which is a very popular approach for traditional machine learning algorithms can help in transforming text into vectors.

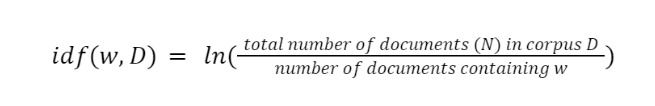
**Term frequency (TF) :**

* It is the ratio of occurrence of word (w) in document (d) per total no. of words in the document.



**Inverse Document frequency (Idf):**

* It is the logarithm of ratio of total no. of documents(N) in corpus(D) to the no. of documents(d) containing word (w)



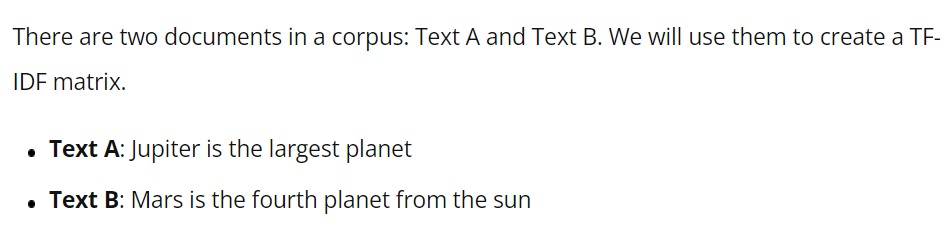
**Term frequency Inverse Document frequency(TFIDF) :**

* It is the product of Term frequency and inverse document frequency .

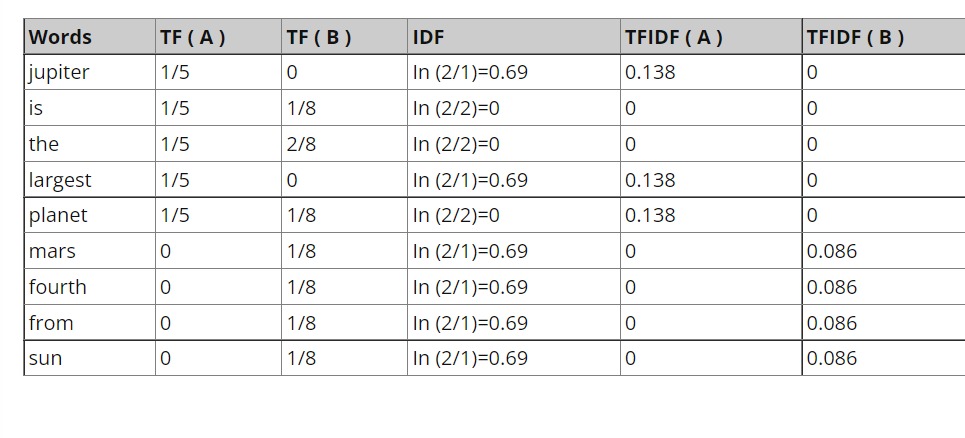


**HOW TO FIND TFIDF MATRIX?**

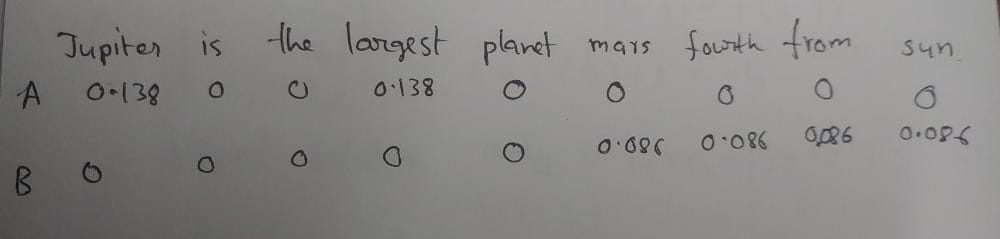
Consider the following dataset



Below we have calculate each words Tf in both texts A and B and IDF for the words.



The conversion of above corpus data by using TFIDF vectorizer gives the following matrix containing TFIDF value for each word.



**Advantages:**

* Easy to compute.
* You have some basic metric to extract the most descriptive terms in a document.
* You can easily compute the similarity between 2 documents using it.

**Disadvantages:**

* TF-IDF is based on the bag-of-words (BoW) model, therefore it does not capture position in text, semantics, co-occurrences in different documents, etc.
* For this reason, TF-IDF is only useful as a lexical level feature.
* Cannot capture semantics (e.g. as compared to topic models, word embeddings)

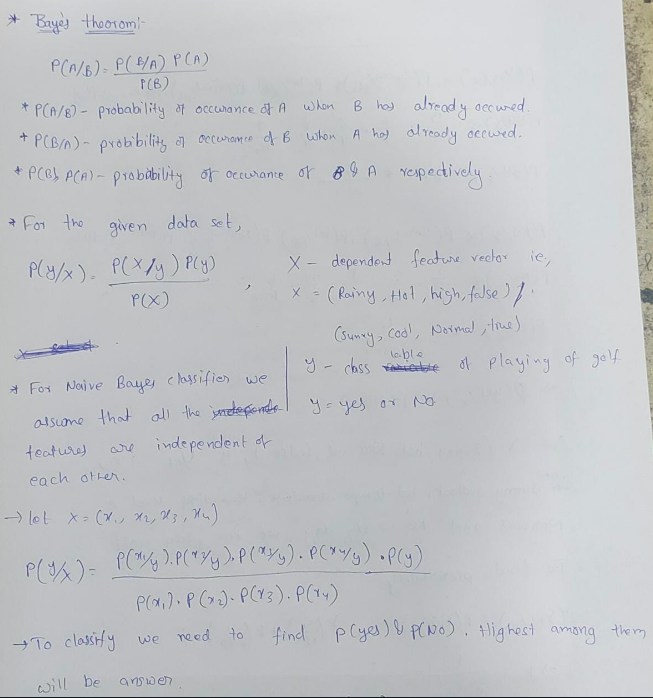
**APPLICATIONS:**

We use TFIDF Vectorizer in machine learning mainly for

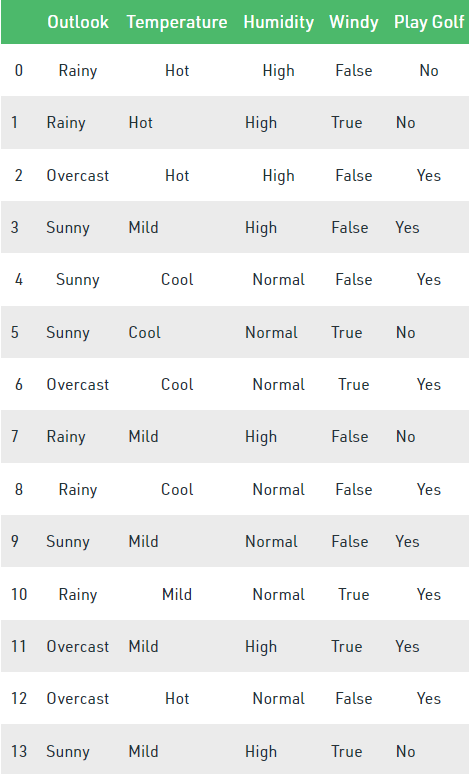
* information retrieval
* text summarization
* keyword extraction

NAÏVE BAYES Classifier:

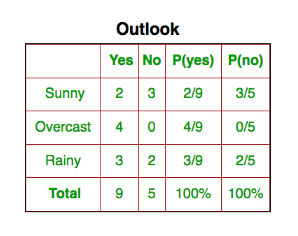
* Naïve Bayes classifier algorithm is the supervised machine learning technique.
* In machine learning Naïve Bayes algorithm belongs to family of simple probabilistic classifiers based on the Bayes theorem ( Conditional Probaility).
* Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set.
* Naïve Bayes algorithm has 2 assumptions:  
  1.All the features we are considering for predicting output are strongly independent i.e., Presence or absence of a feature does not influence the presence or absence of any other feature. Suppose if a,b are 2 independent features then  
   p(a,b)=p(a)\*p(b)  
  2.All the features has same weightage/importance in predicting output.
* Knowing only one of the occurrence of a out a and b can’t predict the output alone because they independent and equally contributing in predicting the output.

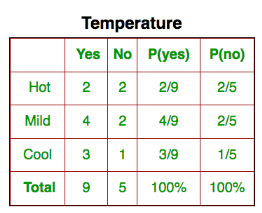


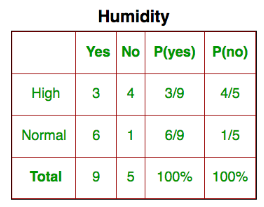
Consider the following data set

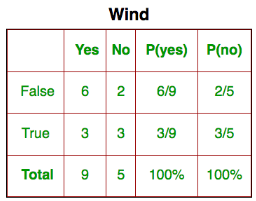


Following are the output prediction for particular variable

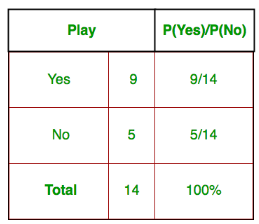






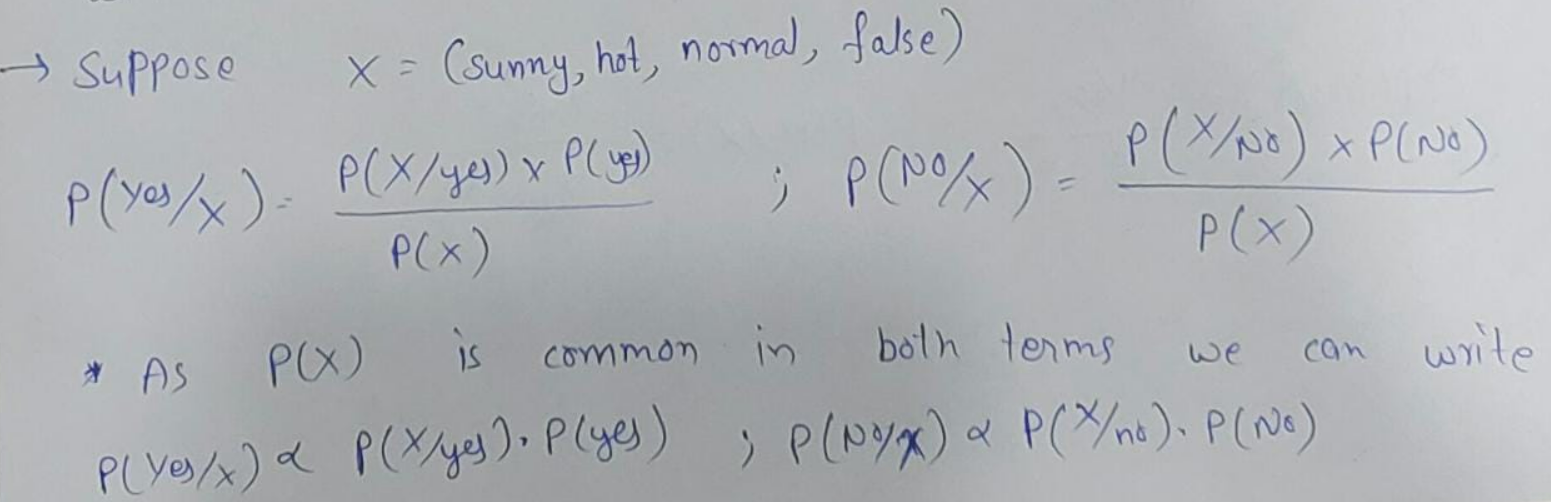


Following denotes the probability of yes and no in the given data set

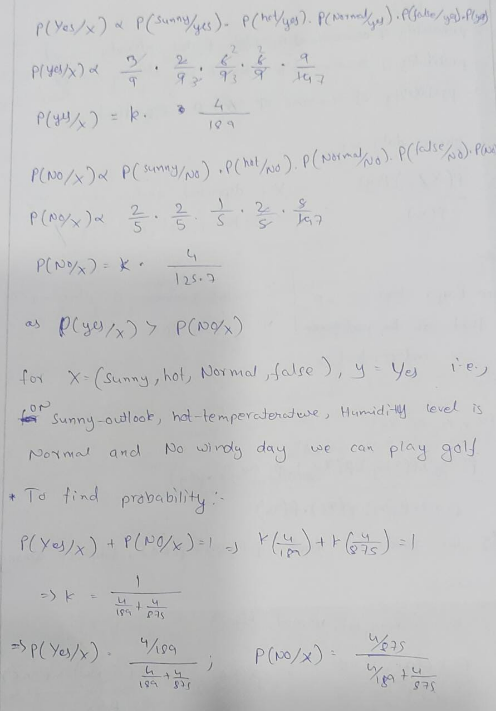


**HOW DO WE APPLY NAÏVE BAYES ALGORITHM?**

Suppose it was asked to predict whether golf can be played or not if the outlook is sunny and temperature is hot and humidity level is normal and no windy conditions.



HAND CALCULATION:



ADVANTAGES OF NAÏVE BAYES CLASSIFIER:

* Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
* It can be used for Binary as well as Multi-class Classifications.
* It performs well in Multi-class predictions as compared to the other Algorithms.
* It is the most popular choice for text classification problems.

DISADVANTAGES OF NAÏVE BAYES CLASSIFIER:

* Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

APPLICATIONS of NAÏVE BAYES CLASSIFIER:

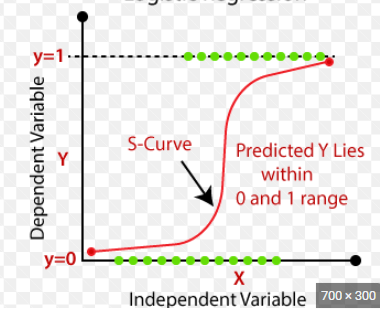
* It is used for Credit Scoring.
* It is used in medical data classification.
* It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
* It is used in Text classification such as Spam filtering and Sentiment analysis.

**LOGISTIC REGRESSION:**

* Logistic Regression is a supervised machine learning technique which predicts the categorical dependent variable using given independent variables.
* As it predicts the categorical dependent   
  variable ,the output must be 0 or 1 ,yes or   
  no,true or false , but instead it gives probabilistic values which lie between 0 and 1.
* We assume the output must be categorical and independent variables ha no multi-collinearity/very less multi-collinearity.
* We use sigmoid function in Logistic Regression.

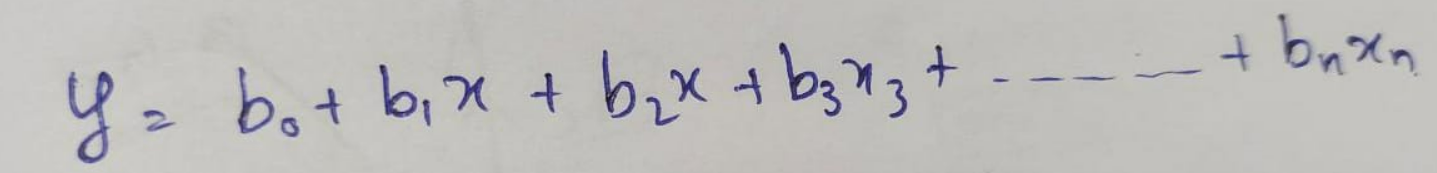
**Sigmoid function:**

* The sigmoid function will map the predicted values to probabilities.
* It is an S-shaped curve which has range from “0” to “1”.
* As we move towards infinity the curve tends to 1 and as we towards negative infinity the curve tends to 0.
* We use threshold value concept to classify . If the probability is higher than threshold value then we can say predicted value is 1 and less than threshold value it is 0.

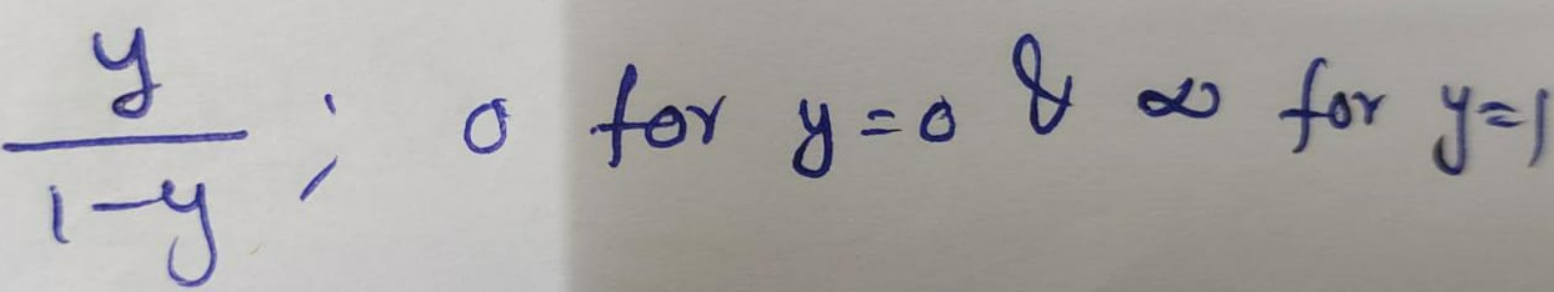


**Proof for equation of logistic regression in sigmoid function:**

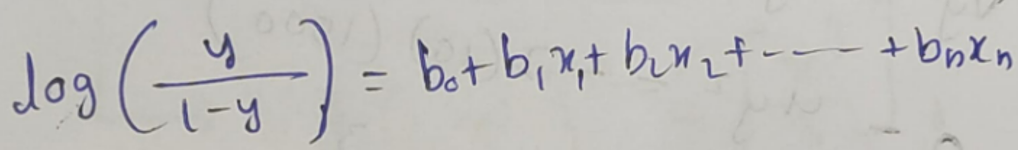
* The Logistic regression equation will obtain from linear regression equation.
* We know linear regression equation with n independent variables is



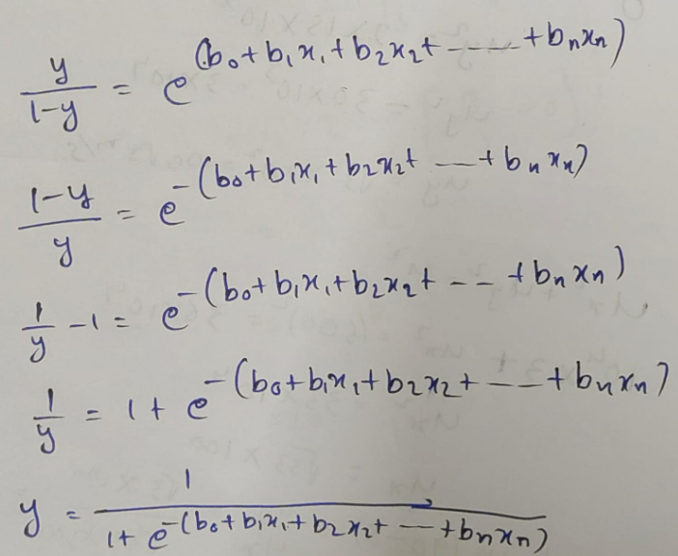
* In Logistic Regression y can be between 0 and 1 only, so for this let's divide the above equation by (1-y).



* But we need range between -[infinity] to +[infinity], then take logarithm of the equation it will become



* By simplification we get the following

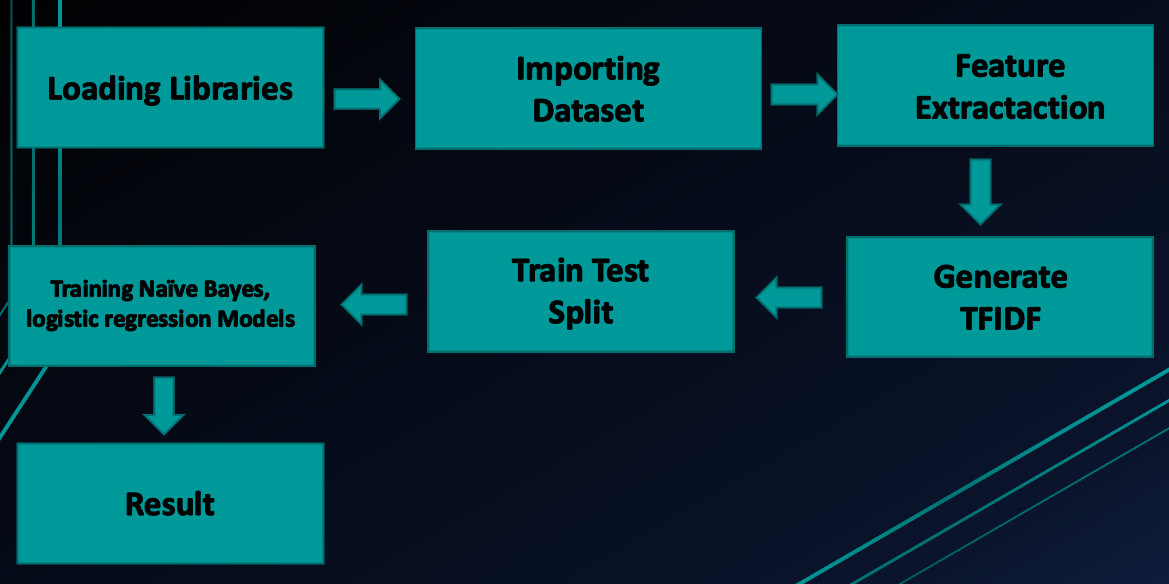


**Advantages of Logistic Regression:**  
**1.** Logistic Regression performs well when the  **dataset is linearly separable.  
2.** Logistic regression is less prone to over-fitting but it can overfit in high dimensional datasets. You should consider Regularization (L1 and L2) techniques to avoid over-fitting in these scenarios.  
**3.** Logistic Regression not only gives a measure of how relevant a predictor (coefficient size) is, but also its direction of association (positive or negative).  
**4.** Logistic regression is easier to implement, interpret and very efficient to train.   
**Disadvantages of Logistic Regression**  
**1.** Main limitation of Logistic Regression is the **assumption of linearity** between the dependent variable and the independent variables. In the real world, the data is rarely linearly separable. Most of the time data would be a jumbled mess.  
**2.** If the number of observations are lesser than the number of features, Logistic Regression should not be used, otherwise it may lead to overfit.  
**3.** Logistic Regression can only be **used to predict discrete functions.** Therefore, the dependent variable of Logistic Regression is restricted to the discrete number set. This restriction itself is problematic, as it is prohibitive to the prediction of continuous data.

**APPLICATIONS:**

Logistic regression is used as classifiers. Some of the applications of logistic regression are:

* Says whether rain falls or not
* Detects whether the email is spam or not
* Detects whether the transaction is fake or real

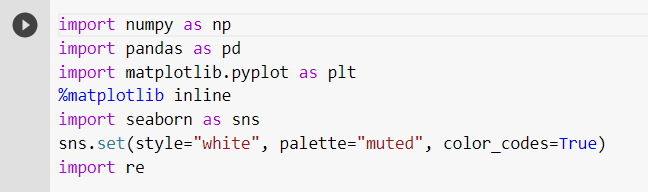
**METHODOLOGY**

**Algorithmic steps:**

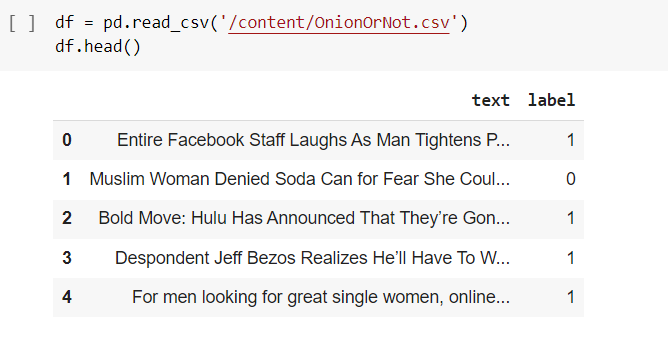
1. We import required libraries. Here we import   
   numpy matplotlib nltk  
   pandas seaborn en\_core\_web\_sm  
   sklearn re word cloud
2. Importing dataset. We have imported dataset from Kaggle which was published by American News publisher “THE ONION”. It contains 2400 rows and 2 columns named text and label.
3. Next whether any data is missed or duplicated in the taken dataset. Next count no.of words, characters, stopwords in each row and add these as columns to dataset.
4. Then convert strings into lowercase and stripe the spaces left and right. Then remove special characters, stopwords from each row. Now lemmatize the each sentence.  
   **Lemmatisation** is nothing but replacing a every word in string by its lemma , where lemma is dictionary or base form of word like lemma of walking is walk and told is tell etc.
5. Now we analyse the dataset like what are most common used words in fake news and in true news through word cloud , where world cloud is picturisation of words in corpus ,larger the frequency of word larger the size of the word. Then we also plot sns plots and bar plots.
6. Then vectorisation of the dataset which will be the final dataset used to train our models
7. Then we split the above data into 75% and 25% for training and testing respectively.
8. Then we train the data to our models Multinomial Naïve Bayes Classifier and Logistic regression.
9. Then we will get outputs in confusion matrices

**CODE and RESULTS:**

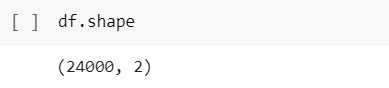
Importing libraries



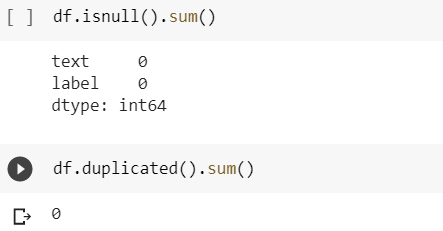
Importing dataset



Checking shape of the data



Checking whether any data is missing or duplicated

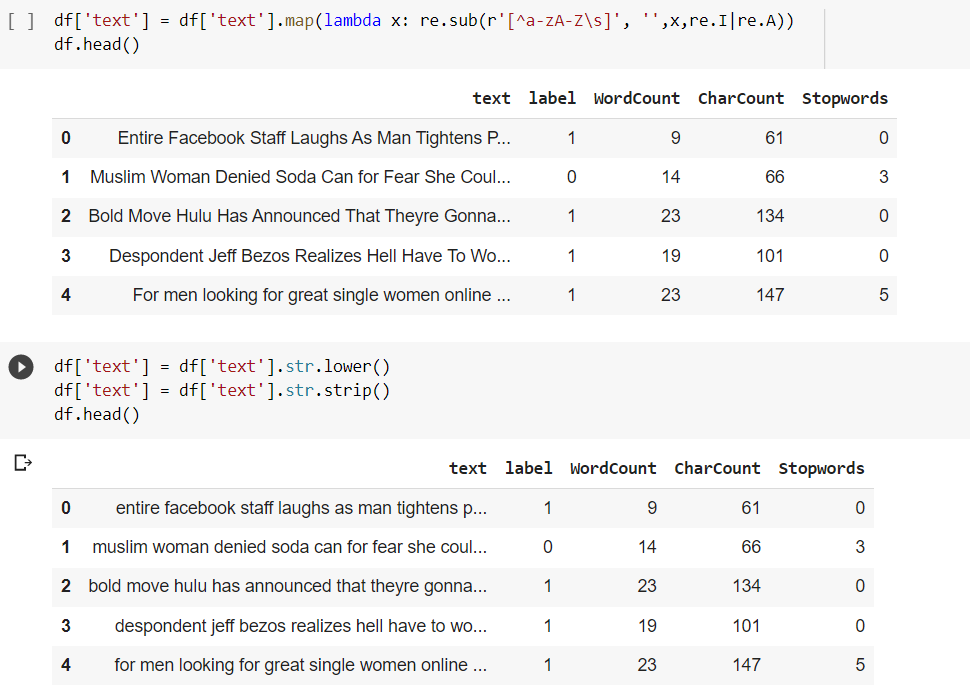


Count no.of words, characters, stopwords:

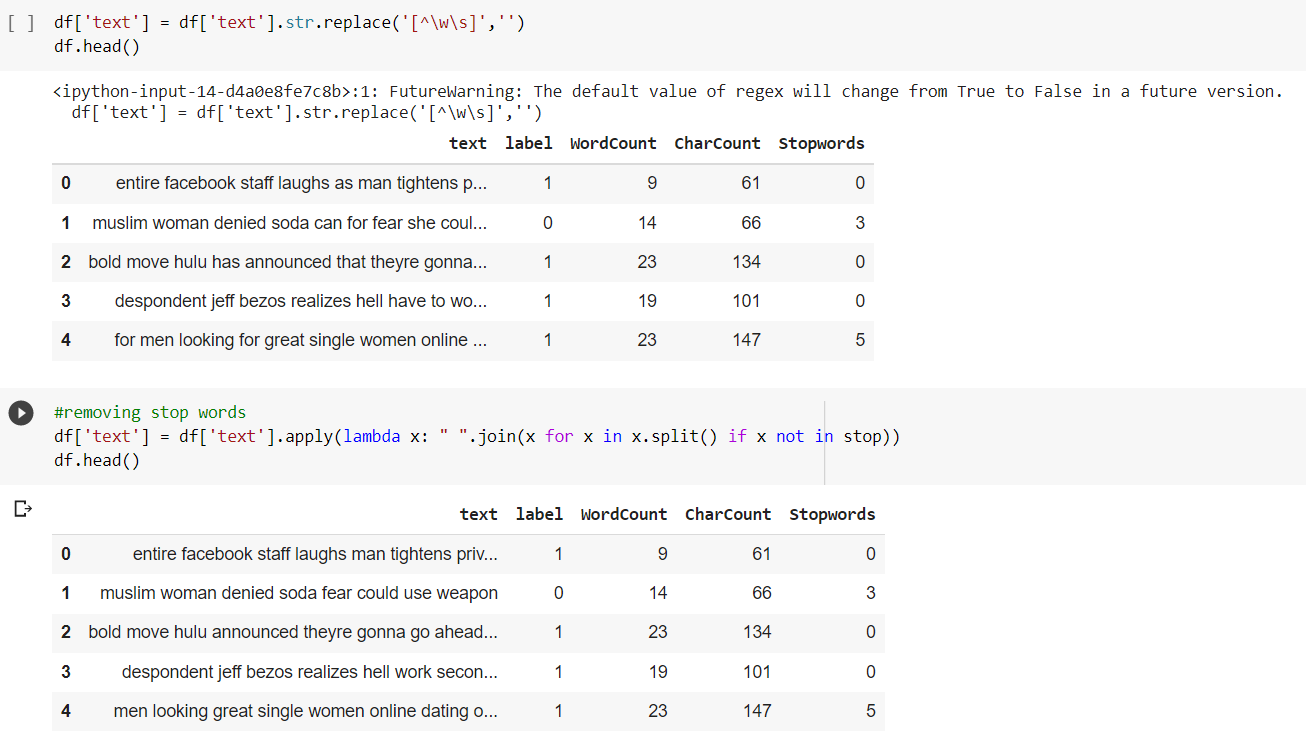




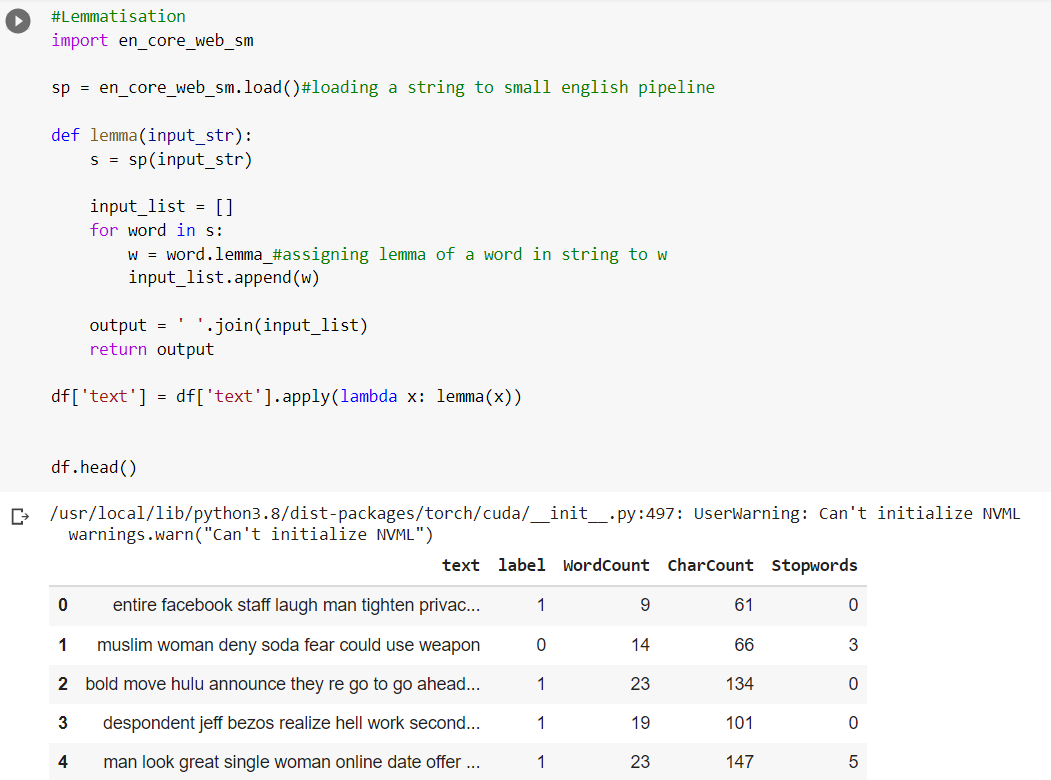
Converting all the uppercase to lowercases and stripe spaces in left and right



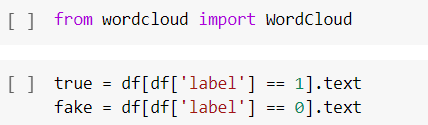
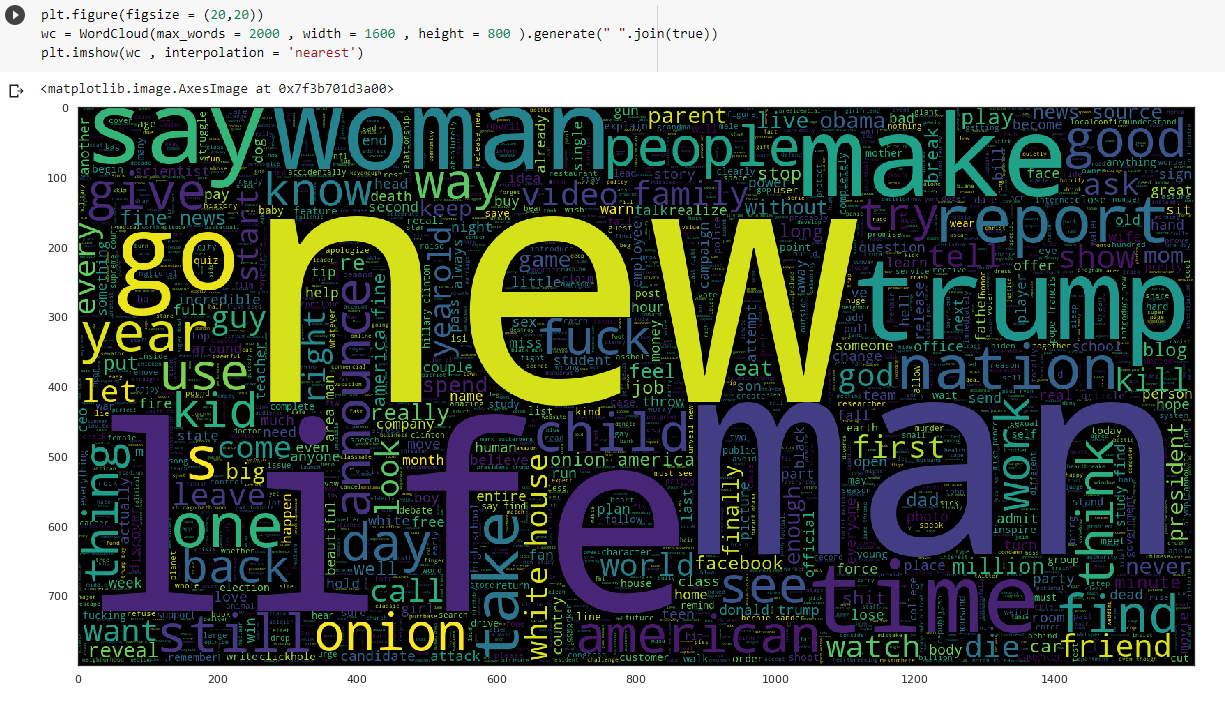
Removing special Characters and stopwords



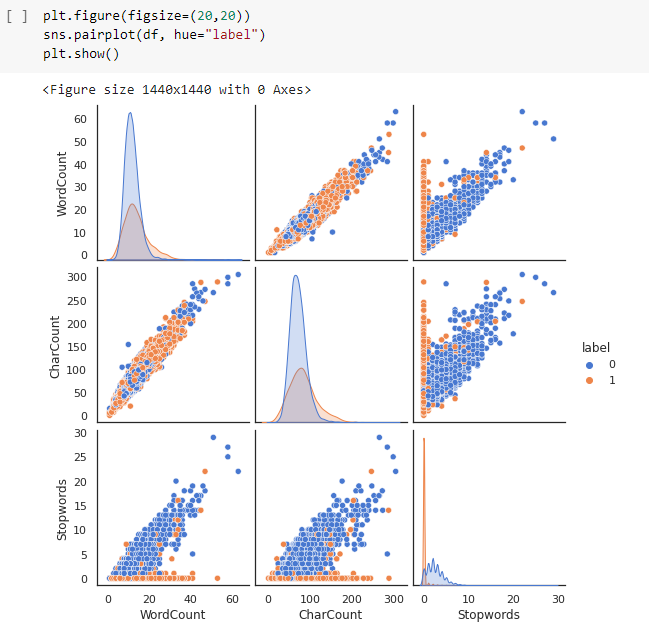
Lemmatisation



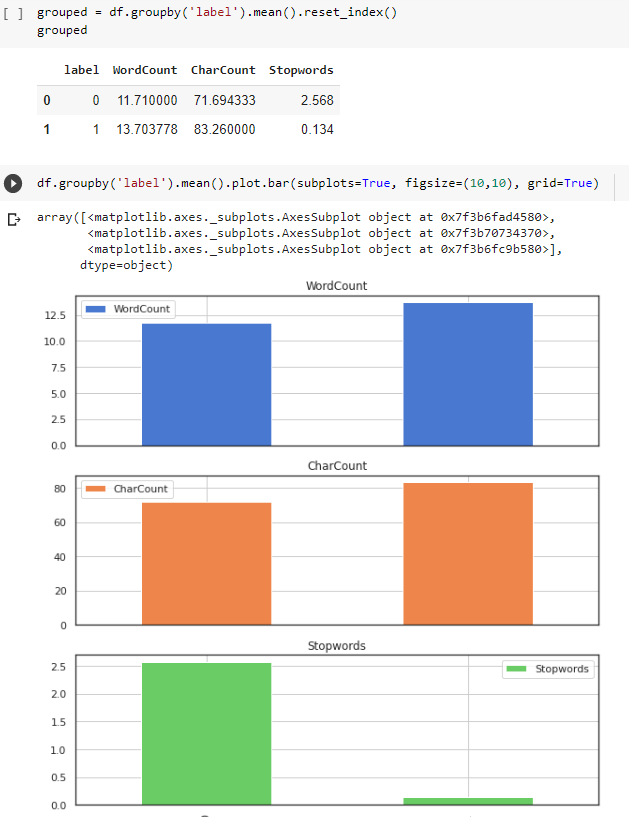
Word Clouds



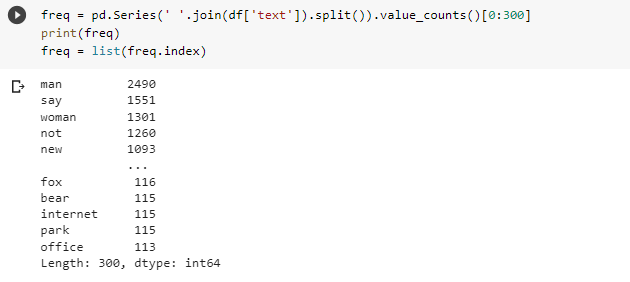
SNS plots

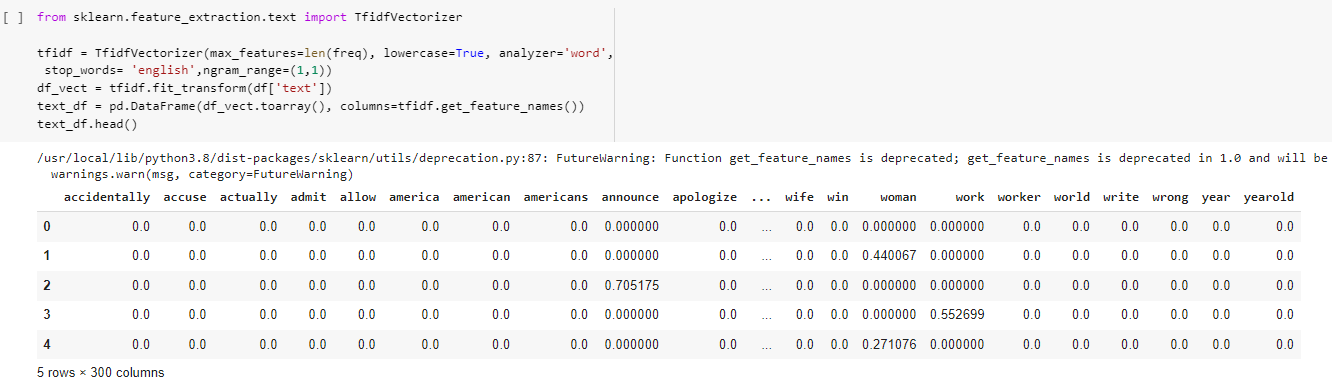


Barplots



Identifying top 300 frequently repeated words for vectorisation



Vectorisation

Splitting data



Training Models and obtaining results

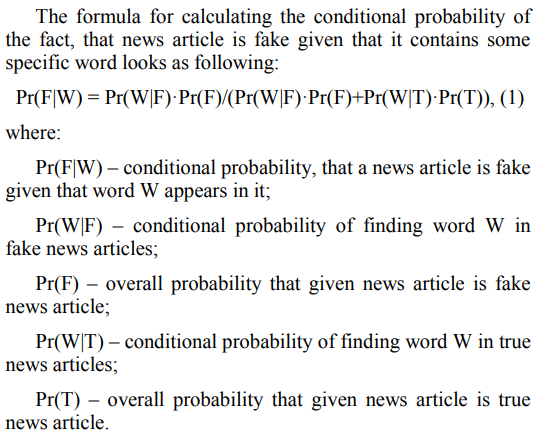
1.Logistic Regression



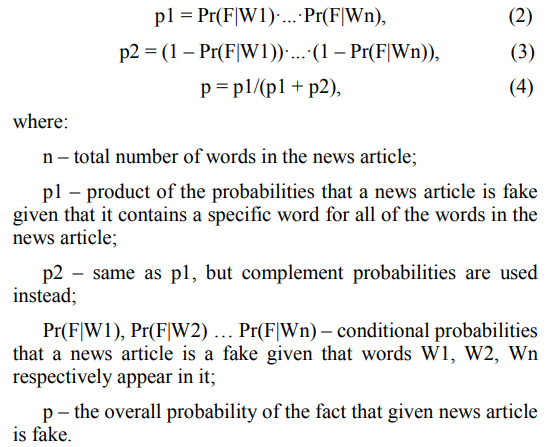
2.Multinomial Naïve Bayes Classifier

Here we are using the TFIDF values to identify whether the news article is fake or true.

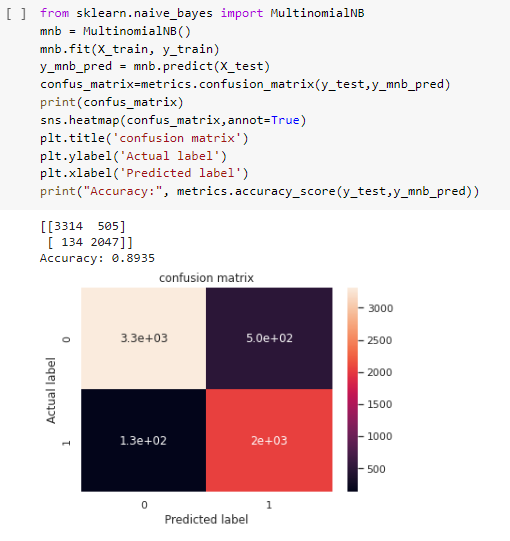
The probability of finding word W in fake news article is ratio of the fake news articles, that contains word W to the total number of fake news articles.



We use threshold value to classify i.e. , if p is greater than threshold value it will be 1 and if p is less than threshold value it will be 0.



The following was the code for the Naive Bayes Classifier model



**CONCLUSION**

The accuracy by the both models was almost same.

But it was slightly greater in Logistic Regression.

The accuracy in Logistic Regression is 90.01 % and in Naïve Bayes Classifier algorithm is 89.35 %.

As the accuracy approaching 90 % we can use these algorithms to detect fake news.