Data Mining Assignment

1.0 Introduction

There exists a lot of data being put away in the electronic arrangement. With such information it has become a need of such implies that could decipher and examine such information and concentrate such realities that could help in dynamic. Data mining which is utilized for separating concealed data from gigantic databases is a powerful instrument that is utilized for this reason. News data was not effectively and rapidly accessible until the start of a decade ago. Be that as it may, presently news is effectively open through substance suppliers, for example, online news administrations.

A gigantic measure of data exists in type of content in different various zones whose examination can be helpful in a few regions. Arrangement is a significant testing field in content mining as it requires predisposing steps to convert unstructured information to organized data. With the expansion in the quantity of news it has troublesome for clients to get to updates on his advantage which makes it a need to classifications news with the goal that it could be without any problem gotten to. Arrangement alludes to gathering that permits simpler route among articles. Web news needs to be partitioned into classifications. This will assist clients with accessing the updates on their enthusiasm for ongoing without squandering whenever. With regards to news it is a lot of hard to order as news are constantly giving the idea that need to be prepared and those news could be never-seen and could fall in another classification.

In this project we have taken a news data from kaggle. This contains 200k news.

The news data contained 6 columns and 200853 rows (samples) in which category column is our target column.

Attributes: Headline | authors | link | short description | date | category

Terminology:

Task / Scenario / Project all refer to the situation of which this report and data mining project has been presented. The terminology refers to the scenario of the financial establishment, and the end goal of predicting the probable chance of future loans being repaid by customers.

Financial Establishment refers to the bank in which the data saws supplied from.

Record(s) refer to the individual data variables for each row within the provided dataset.

Red text within tables (Data Summary section) indicated flagged issues, which may affect the data, or should be attended to.

Instances / Customers / Individuals / Person / Applicant all refer to an individual set of values for one unique person. One person has 15 attributes, and can be identified by their Customer ID.

Record(s) refer to the results of a particular customer or customers. NB found in

chapter 4 (table data) is shorthand for Naive Bayes.

In the Modelling section: **D** = Defaulted | **P** = Paid Mean refers

to the average of a set o

Data Summary

2.0 Headline

String

In this feature we have headlines of news for examples this is the headline for the first sample in the data set **‘There Were 2 Mass Shootings in Texas Last Week, But Only 1 on TV’**

Headline (Attribute)	Missing	Same	Unique
Num of samples	0	1509	199344
Percentage	0%	0.7%	99.3%

Ok it seem that there are duplicates records but actuals it’s not because some of the news have same headlines but different short description and date. Let’s see an example a row number 366 and 698.

(index)	Headline	Short description	authors	Date	Category
366	What To Watch On Hulu That’s New This Week	There's a well-reviewed documentary about bour...	Todd Van Luling	2018-05-19 00:00:00	ENTERTAINMENT
698	What To Watch On Hulu That’s New This Week	The movie most worth watching has a lot to say...	Todd Van Luling	2018-05-12 00:00:00	ENTERTAINMENT

We can clearly see that headlines can be same but they are duplicate records.

2.1 Short description

String

In this column we have description related to news. In this column same like we have duplicates values around about 22500 but they are not consider as duplicates samples let’s see why. I’m taking just two rows for example.

Index	Headline	Short description	authors	Date	Category
745	'Dogs May Help Prepare You For Babies, But Cat...	The 25 most profound “Shower Thoughts” on Redd...	Lee Moran	2018-05-11	WEIRD NEWS
1136	Dogs Protect Humans From Other Humans. Humans...	The 25 most profound “Shower Thoughts” on Redd...	Lee Moran	2018-05-04	WEIRD NEWS

2.2 Authors

String

This attribute have the name of the author, each news have and author name and we have total **27993** unique authors as we know we have total 200853 rows so they repeats. Here I'm writing names of top 3 authors according to number of news.

Author Name	Number Of News
Patrick Cockburn	**36620**
Lee Moran	**2423**
Ron Dicker	**1913**

2.3 Link

String

Link columns shows that from which link this news have been taken. All links are from www.huffingtonpost.com .We have total **200812** unique links and one link is used maximum 2 number of time. In our model we are note using this column.

2.4 Date

String

Date columns shows the news date. Date format is yy-mm-dd. We can have more than one news on a single date. We have news in date between 2012 to 2018. We are not using this column in our model.

2.5 Category

String: target column

This column is our target column. We have 40 unique values in this columns which means that we have total 40 class labels. I'm showing all of them here with the number of samples related to each class

Classes	Number Of Samples
POLITICS	**32739**
WELLNESS	**17827**
ENTERTAINMENT	**16058**
TRAVEL	**9887**

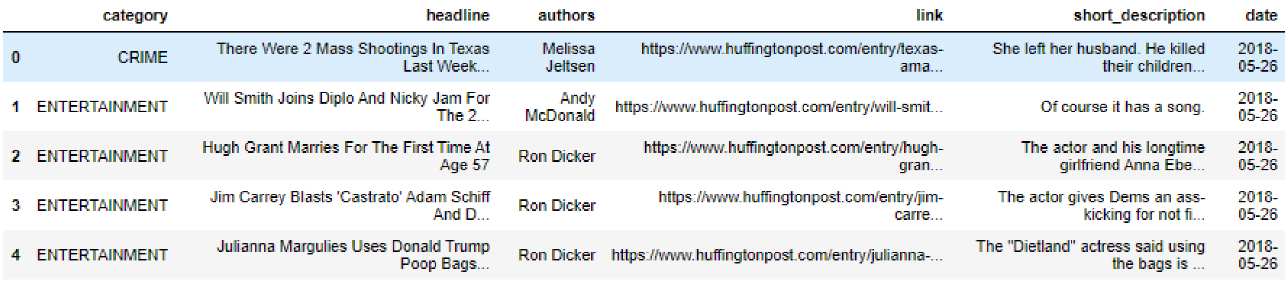
STYLE & BEAUTY	9649
PARENTING	8677
HEALTHY LIVING	6694
QUEER VOICES	6314
FOOD & DRINK	6226
BUSINESS	5937
COMEDY	5175
SPORTS	4884
BLACK VOICES	4528
HOME & LIVING	4195
PARENTS	3955
WORLDPOST	6243
WEDDINGS	3651
WOMEN	3490
IMPACT	3459
DIVORCE	3426
CRIME	3405
MEDIA	2815
WEIRD NEWS	2670
GREEN	2622
RELIGION	2556
STYLE	2254
SCIENCE	2178
WORLD NEWS	2177
TASTE	2096
TECH	2082
MONEY	1707
ARTS	1509
FIFTY	1401

GOOD NEWS	**1398**
ARTS & CULTURE	**2369**
ENVIRONMENT	**1323**
COLLEGE	**1144**
EDUCATION	**1004**
LATINO VOICES	**1129**

In these columns there were 4 classes named as **The WORLDPOST**, **WORLDPOST**, **ARTS & CULTURE** and **CULTURE & ARTS**, so I combine **The WORLDPOST** and **WORLDPOST** as one and make it as **WORLDPOST** and same as **ARTS & CULTURE** and **CULTURE & ARTS** are combined as **ARTS & CULTURE.**

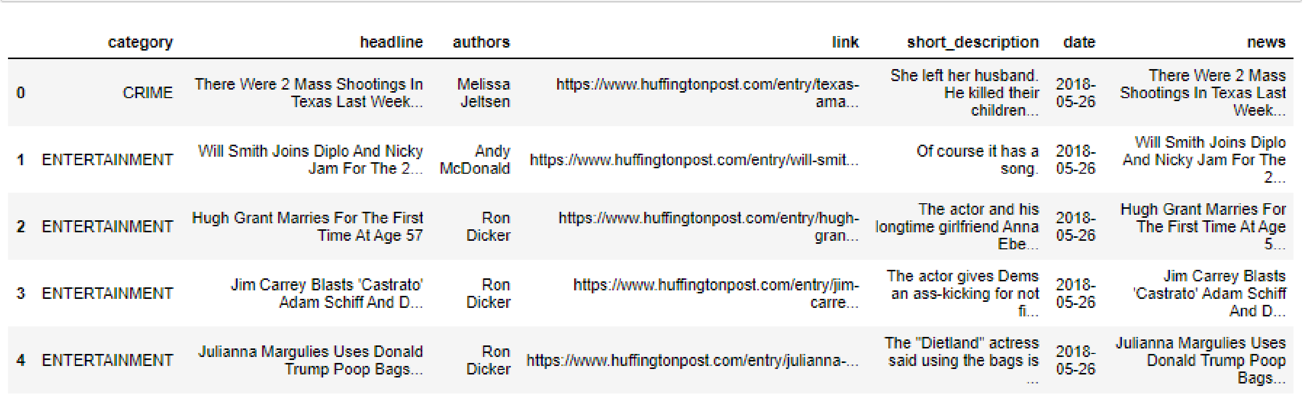
3.0 Data Pre Processing

Data preprocessing is an important step in the data mining process. The phrase "garbage in, garbage out" is particularly applicable to data mining and machine learning projects. In our case we need to do some data preprocessing before we can train our model let's see step by step how we prepare our data.
First we have data in .json format. We read data using pandas and load it as a pandas data frame. Here I'm showing first five rows of our data.



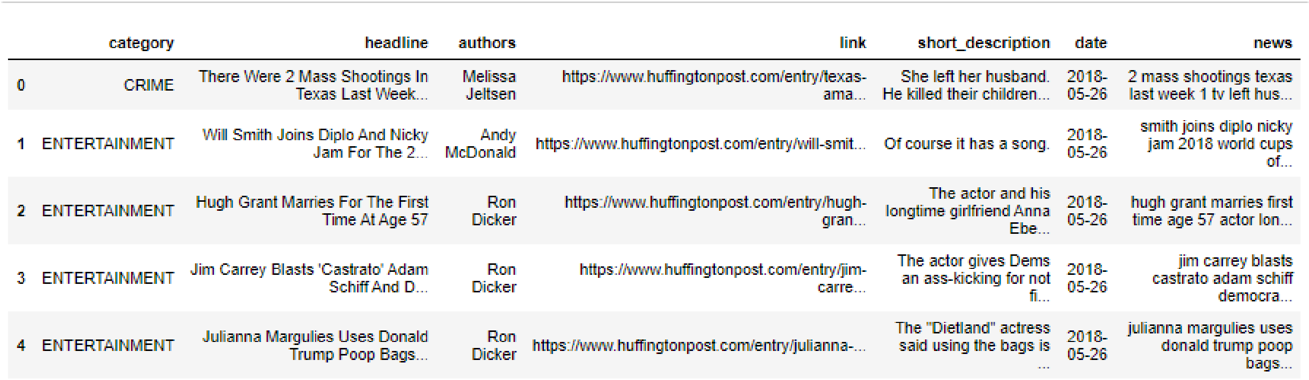
	category	headline	authors	link	short_description	date
0	CRIME	There Were 2 Mass Shootings In Texas Last Week...	Melissa Jeltsen	https://www.huffingtonpost.com/entry/texas-ama...	She left her husband. He killed their children...	2018-05-26
1	ENTERTAINMENT	Will Smith Joins Diplo And Nicky Jam For The 2...	Andy McDonald	https://www.huffingtonpost.com/entry/will-smit...	Of course it has a song.	2018-05-26
2	ENTERTAINMENT	Hugh Grant Marries For The First Time At Age 57	Ron Dicker	https://www.huffingtonpost.com/entry/hugh-gran...	The actor and his longtime girlfriend Anna Ebe...	2018-05-26
3	ENTERTAINMENT	Jim Carrey Blasts 'Castrato' Adam Schiff And D...	Ron Dicker	https://www.huffingtonpost.com/entry/jim-carre...	The actor gives Dems an ass-kicking for not fi...	2018-05-26
4	ENTERTAINMENT	Julianna Margulies Uses Donald Trump Poop Bags...	Ron Dicker	https://www.huffingtonpost.com/entry/julianna-...	The "Dietland" actress said using the bags is ...	2018-05-26

Now we can clearly see our data so first we need to perform the task which we have discus in the section 2.5 about class labels. After doing that.   We need to select feature columns which we will use to train our model for this purpose, we will combine three columns
headline, **short_discription** and **authors** and join the text of them then make another column named as news and put that text in it. After that we will check is there any null value in the news column then drop it so now this news column is consider as our X text on which we will perform some other data preparation techniques and then use it to train or model.



	category	headline	authors	link	short_description	date	news
0	CRIME	There Were 2 Mass Shootings In Texas Last Week...	Melissa Jeltsen	https://www.huffingtonpost.com/entry/texas-ama...	She left her husband. He killed their children...	2018-05-26	There Were 2 Mass Shootings In Texas Last Week...
1	ENTERTAINMENT	Will Smith Joins Diplo And Nicky Jam For The 2...	Andy McDonald	https://www.huffingtonpost.com/entry/will-smit...	Of course it has a song.	2018-05-26	Will Smith Joins Diplo And Nicky Jam For The 2...
2	ENTERTAINMENT	Hugh Grant Marries For The First Time At Age 57	Ron Dicker	https://www.huffingtonpost.com/entry/hugh-gran...	The actor and his longtime girlfriend Anna Ebe...	2018-05-26	Hugh Grant Marries For The First Time At Age 5...
3	ENTERTAINMENT	Jim Carrey Blasts 'Castrato' Adam Schiff And D...	Ron Dicker	https://www.huffingtonpost.com/entry/jim-carre...	The actor gives Dems an ass-kicking for not fi...	2018-05-26	Jim Carrey Blasts 'Castrato' Adam Schiff And D...
4	ENTERTAINMENT	Julianna Margulies Uses Donald Trump Poop Bags...	Ron Dicker	https://www.huffingtonpost.com/entry/julianna-...	The "Dietland" actress said using the bags is ...	2018-05-26	Julianna Margulies Uses Donald Trump Poop Bags...

In this fig we can we have the news column. Ok now we have to perform 2 operation on this news columns first we need to replace all symbols from it and then remove all stop words from it for removing symbols we are using re module of python and for removing stopwords we are using nltk. For this task we have made a function named as clean_text you can see the source code file which will be given, so we apply clean_text   function to the news columns.



	category	headline	authors	link	short_description	date	news
0	CRIME	There Were 2 Mass Shootings In Texas Last Week...	Melissa Jeltsen	https://www.huffingtonpost.com/entry/texas-ama...	She left her husband. He killed their children...	2018-05-26	2 mass shootings texas last week 1 tv left hus...
1	ENTERTAINMENT	Will Smith Joins Diplo And Nicky Jam For The 2...	Andy McDonald	https://www.huffingtonpost.com/entry/will-smit...	Of course it has a song.	2018-05-26	smith joins diplo nicky jam 2018 world cups of...
2	ENTERTAINMENT	Hugh Grant Marries For The First Time At Age 57	Ron Dicker	https://www.huffingtonpost.com/entry/hugh-gran...	The actor and his longtime girlfriend Anna Ebe...	2018-05-26	hugh grant marries first time age 57 actor lon...
3	ENTERTAINMENT	Jim Carrey Blasts 'Castrato' Adam Schiff And D...	Ron Dicker	https://www.huffingtonpost.com/entry/jim-carre...	The actor gives Dems an ass-kicking for not fi...	2018-05-26	jim carrey blasts castrato adam schiff democra...
4	ENTERTAINMENT	Julianna Margulies Uses Donald Trump Poop Bags...	Ron Dicker	https://www.huffingtonpost.com/entry/julianna-...	The "Dietland" actress said using the bags is ...	2018-05-26	julianna margulies uses donald trump poop bags...

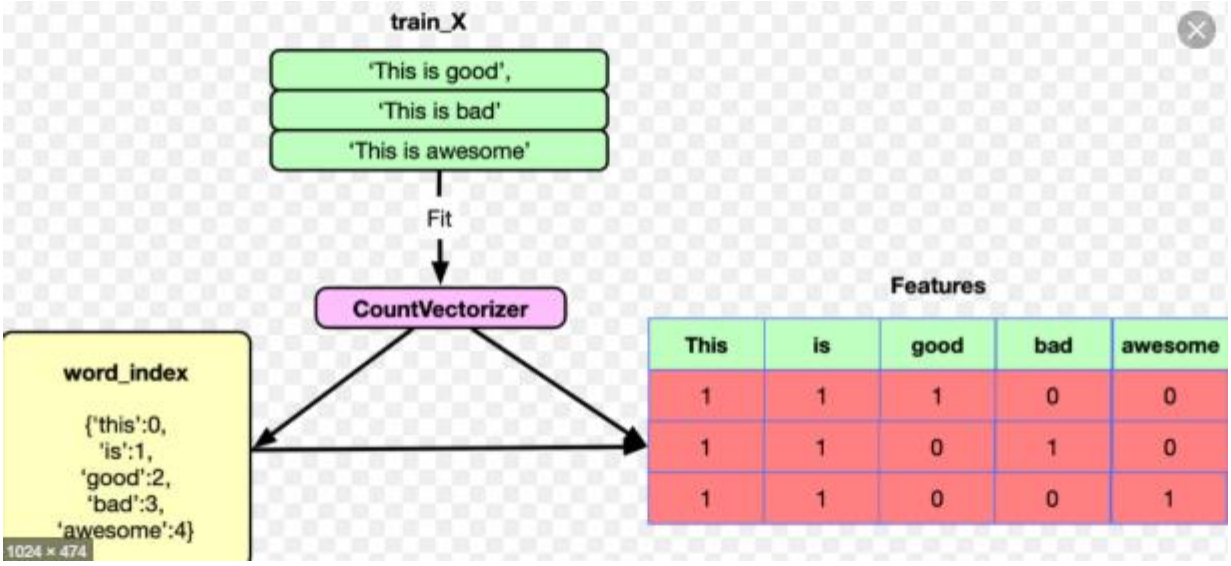
After that we can see this data in our news columns all stopwords are removed and now
We will split or data into train and test for this purpose we are using train_test_split function
From sklearn. We are using 70% data for training and 30% data for testing.

After splitting data we need to train our model but before there are two steps which are very important to do. As we are using multiple machine learning algorithms so before training our models these steps will be done before training every algorithm in our project so let's discuss what are these two steps then we will start explaining our models.

3.1 Count Vectorizer

The CountVectorizer give us simple way to both tokenize a collection of text and build a vocabulary of known words, basically It will give a particular unique number to each of unique word then it will count how many times are particular number is repeating. Process of giving a unique number to each word will be done by fit method which is in CountVectorizer class than counting the number of times that particular number is coming is done by transform method which is also present in CountVectorizer class.

Here for just better understanding I’m attaching a picture so we can have clear understand how CountVectorizer works.



This	is	good	bad	awesome
1	1	1	0	0
1	1	0	1	0
1	1	0	0	1

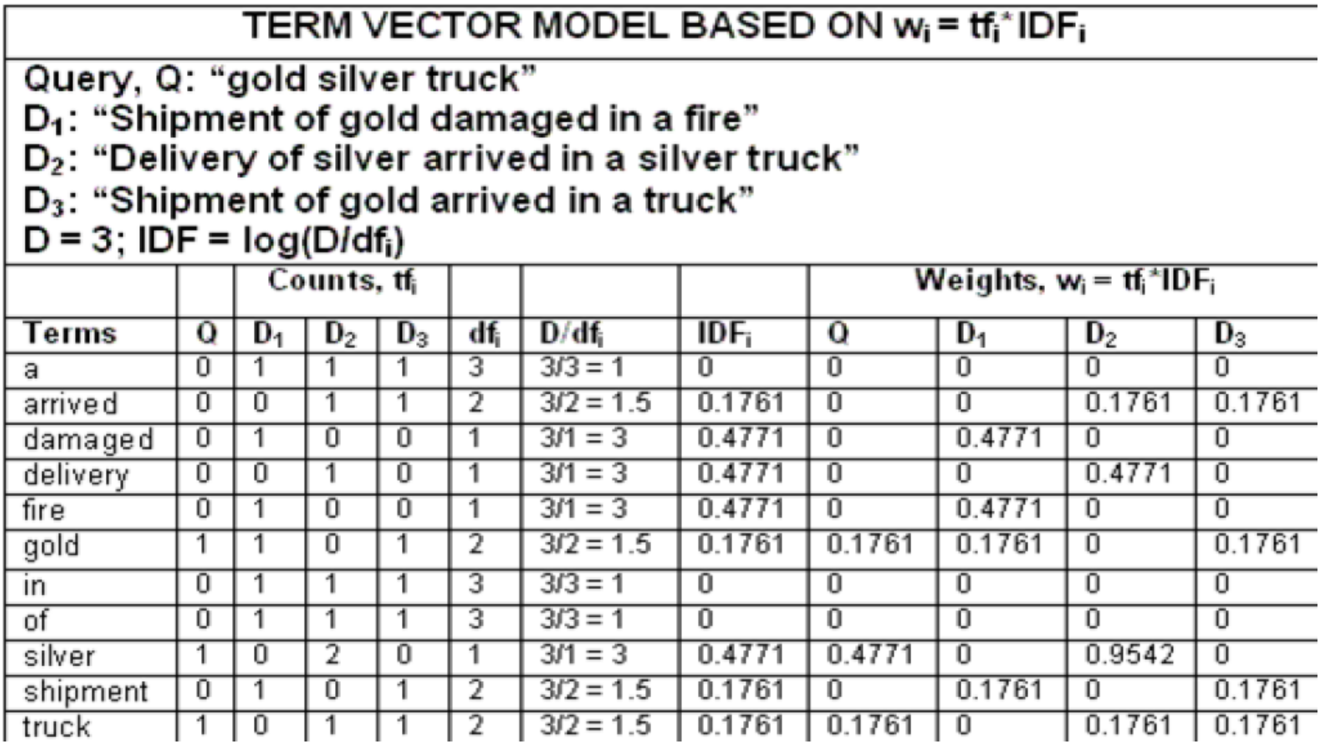
3.2 TF-IDF

TF stands for Term Frequency: TF summarizes how often a given word appears within a document or we can say no of times a term occurred in a documents.

DF stands for document frequency which means that no of documents in which a term occurred

IDF stands for Inverse Document Frequency: IDF downscales words that appear a lot of across documents.

Here I’m attaching a picture just for example how tf-idf works.



TERM VECTOR MODEL BASED ON $w_i = tf_i * IDF_i$

Query, Q: “gold silver truck”
D_1: “Shipment of gold damaged in a fire”
D_2: “Delivery of silver arrived in a silver truck”
D_3: “Shipment of gold arrived in a truck”
D = 3; IDF = log(D/df_i)

		Counts, tf_i						Weights, $w_i = tf_i * IDF_i$			
Terms	Q	D_1	D_2	D_3	df_i	D/df_i	IDF_i	Q	D_1	D_2	D_3
a	0	1	1	1	3	3/3 = 1	0	0	0	0	0
arrived	0	0	1	1	2	3/2 = 1.5	0.1761	0	0	0.1761	0.1761
damaged	0	1	0	0	1	3/1 = 3	0.4771	0	0.4771	0	0
delivery	0	0	1	0	1	3/1 = 3	0.4771	0	0	0.4771	0
fire	0	1	0	0	1	3/1 = 3	0.4771	0	0.4771	0	0
gold	1	1	0	1	2	3/2 = 1.5	0.1761	0.1761	0.1761	0	0.1761
in	0	1	1	1	3	3/3 = 1	0	0	0	0	0
of	0	1	1	1	3	3/3 = 1	0	0	0	0	0
silver	1	0	2	0	1	3/1 = 3	0.4771	0.4771	0	0.9542	0
shipment	0	1	0	1	2	3/2 = 1.5	0.1761	0	0.1761	0	0.1761
truck	1	0	1	1	2	3/2 = 1.5	0.1761	0.1761	0	0.1761	0.1761

Now we know how CountVectorizer and TfidfTransformer works.
The first model we used for news classification is Naive Bayes Classifier for Multinomial.

4.0 Data Mining Algorithms

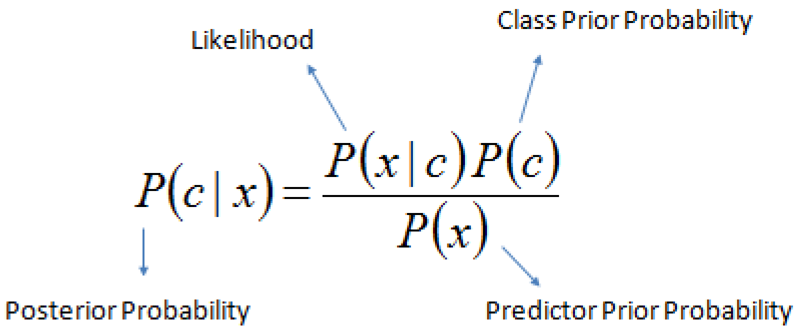
Naïve Bayes

The word Bayes in this algorithm comes from the Thomas Bayes, he was English statistician in 1970s he published a theory named as Bayes' Theorem related to conditional probability. Naïve Bayes is based on the Bayes' Theorem and the word naïve used because it makes the assumption that features of a measurement are independent of each other.

Naïve Bayes classifiers are collection of classification algorithms. It is not a single algorithm it is a pack of algorithms where all of them share a common principle, every pair of features being classified is independent of each other.

Here we are using Multinomial Naïve Bayes because we have multiple class labels.

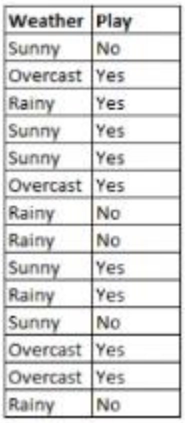
Naïve Bayes first compute the probability of every class label and then with respect to it find out the conditional probability for every element in every feature and base on those probabilities it compute probability for every class w.r.t given x and choose the class label with the highest probability. Formula for naive Bayes is



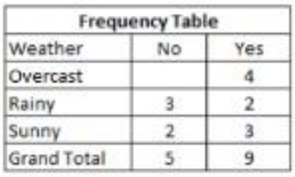
$$P(c \mid x) = \frac{P(x \mid c)\,P(c)}{P(x)}$$

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

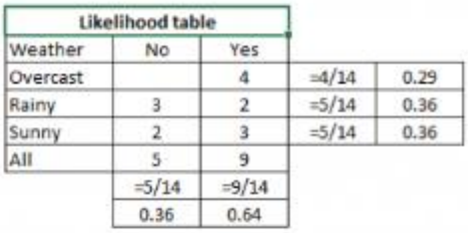
Just for Example how naïve Bayes work



Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No



Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9



Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Pros and Cons of Naïve Bayes

Pros:

It is easy and fast to predict class because all the probabilities are pre-computed in the Naive Bayes algorithm, the prediction time of this algorithm is very efficient.

It needs only one pass on entire data to calculate the posterior probabilities for each value of the feature in the data. So, when we are dealing with large datasets or low-budget hardware, it will be feasible choice as compare to other data mining algorithms.

Cons:

If any categorical feature has any category in test data, which is not in training data, then model will assign a 0 probability and then it will not able to make prediction. This is known as 0 Frequency. To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

Decision Tree

First article I've had the option to find that builds up a " Decision tree" approach dates to 1959 and a British analyst, William Belson, in a paper titled Matching and Prediction on the Principle of Biological Classification, (JRSS, Series C, Applied Statistics, Vol. 8, No. 2, June, 1959, pp. 65-75), whose theoretical portrays his methodology as one of coordinating populace tests and creating rules for doing as such: In this article Dr Belson depicts a procedure for coordinating populace tests. This relies upon the blend of experimentally created indicators to give the best accessible prescient, or coordinating, composite. The basic standard is very unmistakable from that intrinsic in the numerous relationship strategy.

Decision Tree Analysis is a general, prescient demonstrating device that has applications spreading over various zones. By and large, Decision trees are built by means of an algorithmic methodology that recognizes approaches to part an informational index dependent on various conditions.

Basically it is a greedy algorithm. In this algorithm Tree is constructed in a top down recursive and divide-and-conquer manner. At start, all the training examples are the root. Then most important attributes are selected on the based on entropy, information gain. There are many algorithms for decision tree for example.
Hunts Algorithm, CART, ID3, C4.5, C5.0
For ID3 we need to understand two terms

Entropy

Entropy is the measure of uncertainty associated with a random measure, higher entropy means high uncertainty low entropy means low uncertainty
It is also known as measure of dispersion. Formula is:

$$Entropy(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

Information Gain: Attribute Selection Measure

Select the attribute with the highest information gain. Formula for info gain is:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

$$Gain(A) = Info(D) - Info_A(D)$$

There is another better way for attribute selection which is called as gini index.

- If data set contains examples from n classes then formula for gini index is:

- $$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$$

Its works like this in start spit the records based on an attribute, then test that optimizes certain criterion. Now when stop splitting it stop splitting when all records in a table belongs to same class.

Pros and Cons of Decision Tree:

Pros

- Extremely fast at classifying unknown records
- Easy to interpret for small-sized trees
- Accuracy is comparable to other classification techniques for many simple data sets

Cons

- Not scalable (add one attribute, all tree needed to be computed again)
- Not good accuracy for large dataset
- Not robust (less handling of a big set of attributes)

Random Forest

RF is a gathering learning strategy utilized for arrangement and relapse. Created by Breiman (2001), the technique consolidates Breiman's sacking testing approach ((1996a), and the arbitrary choice of highlights, presented autonomously by Ho (1995); Ho (1998) and Amit and Geman (1997), so as to build an assortment of choice trees with controlled variety. Utilizing stowing, every choice tree in the group is developed utilizing an example with substitution from the preparation information. Factually, the example is probably going to have about 64% of occasions showing up at any rate once in the example. Examples in the example are alluded to as in-sack occasions, and the rest of the occurrences (about 36%) are alluded to as out-of-pack cases. Each tree in the gathering goes about as a base classifier to decide the class mark of an unlabeled case. This is done through dominant part casting a ballot where every classifier makes one choice for its anticipated class mark, at that point the class name with the most votes is utilized to characterize the occasion.

The RF consolidates hundreds or thousands of decision trees, prepares everyone on a somewhat unique arrangement of the perceptions, parting hubs in each tree thinking about a predetermined number of the highlights. The last expectations of the irregular woods are made by averaging the forecasts of every individual tree.

Pros and Cons of Random Forest:

Pros

- The predictive performance can compete with the best supervised learning algorithms.
- Forest has the power to handle large data sets with higher dimensionality and identity most significant variables so it is considered as one of the dimensionality reduction method.

Cons

- An ensemble model is inherently less interpretable than an individual decision tree
- Training a large number of deep trees can have high computational costs (but can be parallelized) and use a lot of memory

Support Vector Machine

The first SVM calculation was created by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. In 1992, Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik proposed an approach to make nonlinear classifiers by applying the piece stunt to greatest edge hyperplanes
SVM is a machine learning algorithm which belongs to supervised learning it can be used for classification or regression issues. It utilizes a method called the kernel to change your

information and afterward dependent on these changes it finds an ideal limit between the potential yields.

Properties of SVM:

Duality| Kernels| Margin| Convexity| sparseness

The basic concept of svm is to find a liner decision surface or line which is called as hyper plane that can separate classes and has the largest distance this distance is known as margin between the border line elements of both classes.
Hyper plane is a liner decision surface that split the space into two parts, yes of course it will be a binary classifier. The equation of hyper plane is defined by point p0 and perpendicular vector to plan w at that point.
A Convex function is function if it lies below the straight line segment connecting two points, for any two points in the interval
Kernel method is a similarity function that we provide to a machine learning algorithm, it takes two inputs and output how similar they are, while the similarity criteria is differs from one kernel to another

Pros and Cons of Support Vector Machine

Pros

- SVM is more effective in high dimensional spaces
- SVM is relatively memory efficient
- SVM is effective in cases where number of dimensions is greater than the number of samples.

Cons

- SVM algorithm is not suitable for large data sets.
- Does not perform well in case of overlapped classes
- Selecting the appropriate kernel function can be tricky.

Logistic Regression

Logistic Regression's history can be followed back to the nineteenth century when it was first used to portray the development pace of populaces by Quetelet and Verhulst. Today, strategic relapse is broadly utilized in the field of medication and science. The study of disease transmission is additionally a zone where calculated relapse is broadly utilized for recognizable proof of hazard factors for infections and to anticipate preventive medicine. Studies worried about general wellbeing and related strategy choices utilize calculated relapse as a significant factual device.

Calculated relapse means to quantify the connection between an absolute ward variable and at least one autonomous factors (typically consistent) by plotting the needy factors' likelihood scores. A straight out factor is a variable that can take esteems falling in constrained classes as opposed to being consistent.

Logistic Regression's utilizes relapse to anticipate the result of a clear cut dependent variable based on indicator factors. The plausible results of a solitary preliminary are displayed as an element of the logical variable utilizing a calculated capacity. Strategic displaying is done on absolute information which might be of different sorts including double and ostensible. For instance, a variable may be twofold and have two potential classes of 'yes' and 'no'; or it might be ostensible state hair shading possibly dark, earthy colored, red, gold and dim.

Pros and Cons of Logistic Regression's

Pros

- Logistic regression works well for predicting categorical outcomes
- Efficient, and can be distributed (ADMM)
- Robust to noise

Cons

- Logistic regression attempts to predict based on a set of independent variables, but if we include the wrong independent variables, the model will have little to no predictive value.

- Logistic regression cannot predict continuous outcomes

5 Modeling

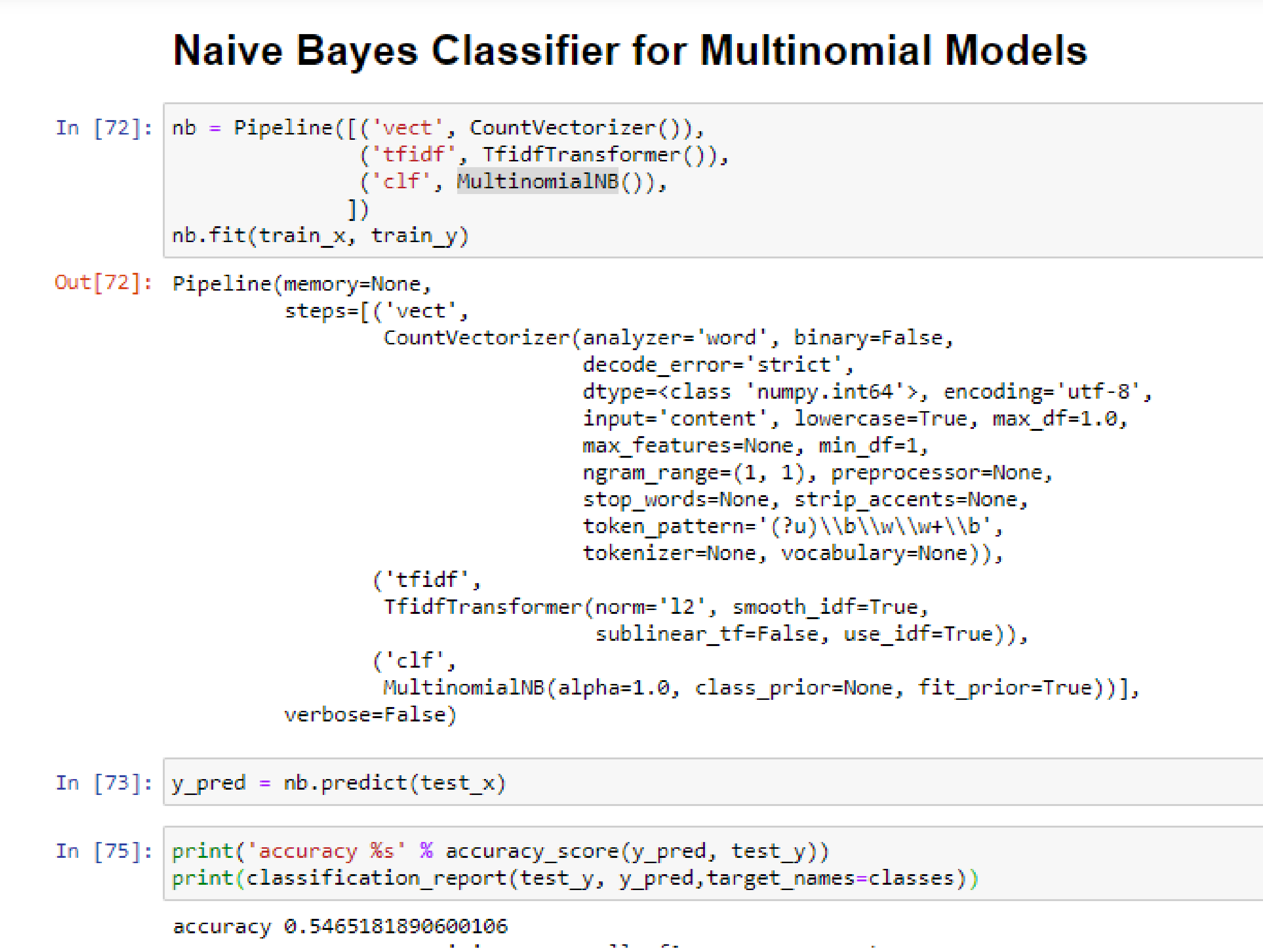
Experiment one (Naïve Bayes)

Build Naïve Bayes model and predict news classes

Aim: to predict a news belongs to which class or category

Methodologies

Before we have already preprocess the data now we need to make naïve Bayes model then train it and then predict using test data.



Naive Bayes Classifier for Multinomial Models

```
In [72]: nb = Pipeline([('vect', CountVectorizer()),
                        ('tfidf', TfidfTransformer()),
                        ('clf', MultinomialNB()),
                       ])
         nb.fit(train_x, train_y)

Out[72]: Pipeline(memory=None,
                  steps=[('vect',
                          CountVectorizer(analyzer='word', binary=False,
                                          decode_error='strict',
                                          dtype=<class 'numpy.int64'>, encoding='utf-8',
                                          input='content', lowercase=True, max_df=1.0,
                                          max_features=None, min_df=1,
                                          ngram_range=(1, 1), preprocessor=None,
                                          stop_words=None, strip_accents=None,
                                          token_pattern='(?u)\\b\\w\\w+\\b',
                                          tokenizer=None, vocabulary=None)),
                         ('tfidf',
                          TfidfTransformer(norm='l2', smooth_idf=True,
                                           sublinear_tf=False, use_idf=True)),
                         ('clf',
                          MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True))],
                  verbose=False)

In [73]: y_pred = nb.predict(test_x)

In [75]: print('accuracy %s' % accuracy_score(y_pred, test_y))
         print(classification_report(test_y, y_pred,target_names=classes))

         accuracy 0.5465181890600106
```

Here we can see that we make a pipe line and in which we have done CountVectorize and then TfidTransform which we have discuss above after that we make a model and then fir it using train_x and train_y and then predict using test_x which gives result y_pred and then we print calculate the accuracy sore using y_pred which are predicted values and test_y are real values. And so our accuracy is 54%

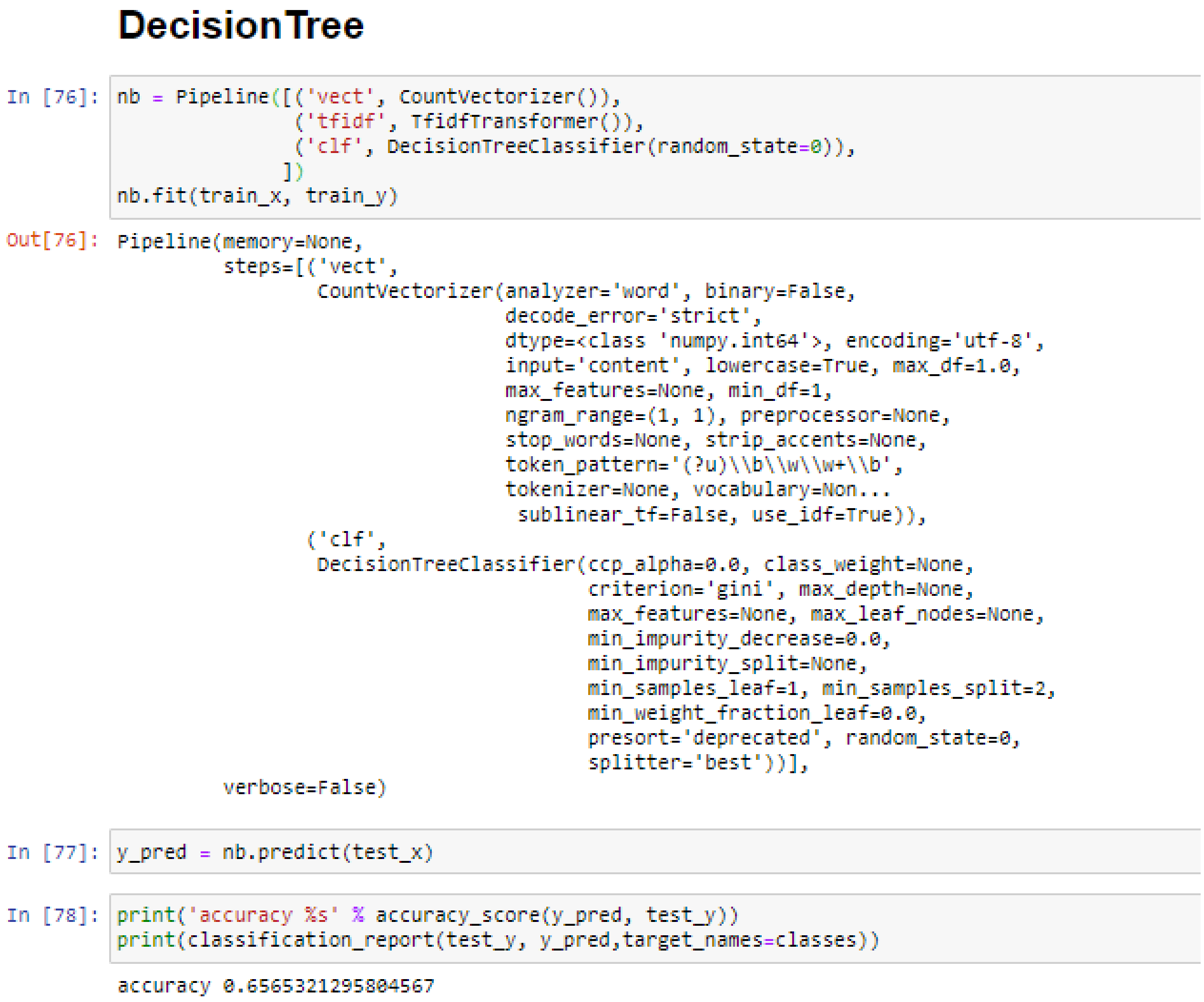
Experiment Two (Decision Tree)

Build Decision Tree model and predict news classes

Aim: to predict a news belongs to which class or category

Methodologies

Before we have already preprocess the data now we need to make Decision Tree then train it and then predict using test data.



DecisionTree

```
In [76]: nb = Pipeline([('vect', CountVectorizer()),
                        ('tfidf', TfidfTransformer()),
                        ('clf', DecisionTreeClassifier(random_state=0)),
                       ])
         nb.fit(train_x, train_y)

Out[76]: Pipeline(memory=None,
                   steps=[('vect',
                           CountVectorizer(analyzer='word', binary=False,
                                           decode_error='strict',
                                           dtype=<class 'numpy.int64'>, encoding='utf-8',
                                           input='content', lowercase=True, max_df=1.0,
                                           max_features=None, min_df=1,
                                           ngram_range=(1, 1), preprocessor=None,
                                           stop_words=None, strip_accents=None,
                                           token_pattern='(?u)\\b\\w\\w+\\b',
                                           tokenizer=None, vocabulary=Non...
                                            sublinear_tf=False, use_idf=True)),
                          ('clf',
                           DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None,
                                                  criterion='gini', max_depth=None,
                                                  max_features=None, max_leaf_nodes=None,
                                                  min_impurity_decrease=0.0,
                                                  min_impurity_split=None,
                                                  min_samples_leaf=1, min_samples_split=2,
                                                  min_weight_fraction_leaf=0.0,
                                                  presort='deprecated', random_state=0,
                                                  splitter='best'))],
                   verbose=False)

In [77]: y_pred = nb.predict(test_x)

In [78]: print('accuracy %s' % accuracy_score(y_pred, test_y))
         print(classification_report(test_y, y_pred,target_names=classes))

         accuracy 0.6565321295804567
```

We can see decision tree accuracy is more than naïve Bayes. 65% Accuracy.

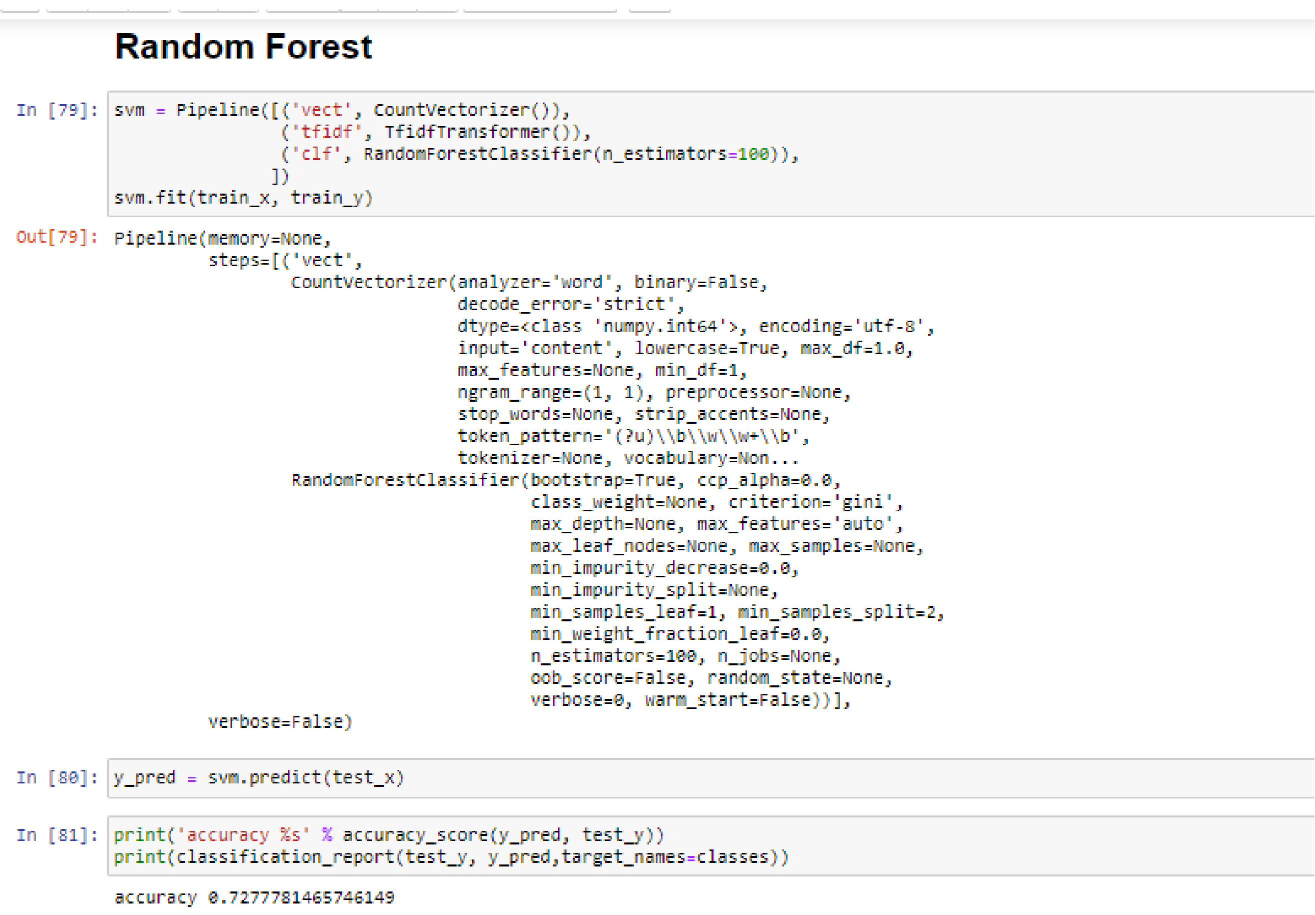
Experiment Three (Random Forest)

Build Random Forest model and predict news classes

Aim: to predict a news belongs to which class or category

Methodologies

Before we have already preprocess the data now we need to make Random Forest then train it and then predict using test data.



Random Forest

In [79]:
```
svm = Pipeline([('vect', CountVectorizer()),
                ('tfidf', TfidfTransformer()),
                ('clf', RandomForestClassifier(n_estimators=100)),
               ])
svm.fit(train_x, train_y)
```

Out[79]:
```
Pipeline(memory=None,
         steps=[('vect',
                 CountVectorizer(analyzer='word', binary=False,
                                 decode_error='strict',
                                 dtype=<class 'numpy.int64'>, encoding='utf-8',
                                 input='content', lowercase=True, max_df=1.0,
                                 max_features=None, min_df=1,
                                 ngram_range=(1, 1), preprocessor=None,
                                 stop_words=None, strip_accents=None,
                                 token_pattern='(?u)\\b\\w\\w+\\b',
                                 tokenizer=None, vocabulary=Non...
                 RandomForestClassifier(bootstrap=True, ccp_alpha=0.0,
                                        class_weight=None, criterion='gini',
                                        max_depth=None, max_features='auto',
                                        max_leaf_nodes=None, max_samples=None,
                                        min_impurity_decrease=0.0,
                                        min_impurity_split=None,
                                        min_samples_leaf=1, min_samples_split=2,
                                        min_weight_fraction_leaf=0.0,
                                        n_estimators=100, n_jobs=None,
                                        oob_score=False, random_state=None,
                                        verbose=0, warm_start=False))],
         verbose=False)
```

In [80]:
```
y_pred = svm.predict(test_x)
```

In [81]:
```
print('accuracy %s' % accuracy_score(y_pred, test_y))
print(classification_report(test_y, y_pred,target_names=classes))
```

```
accuracy 0.7277781465746149
```

Better then decision tree 72% accuracy

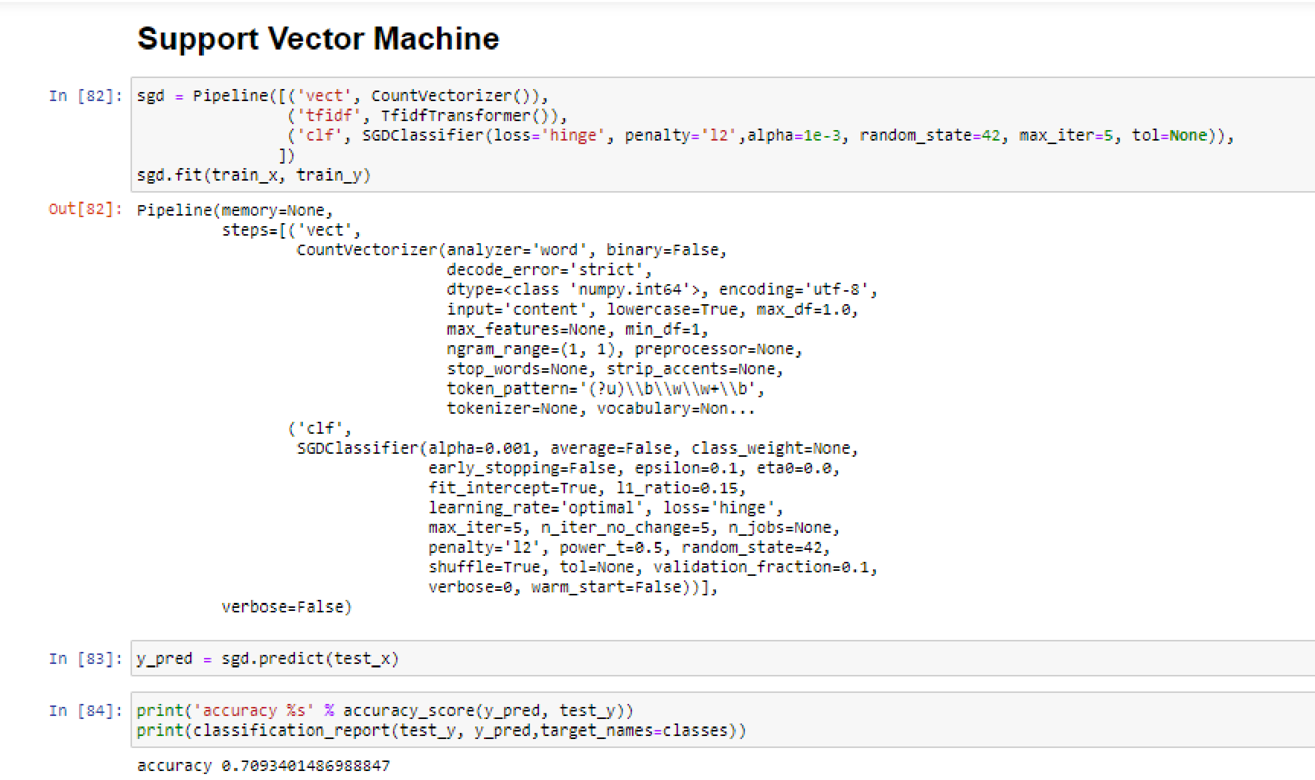
Experiment Four (Support Vector Machine)

Build SVM model and predict news classes

Aim: to predict a news belongs to which class or category

Methodologies

Before we have already preprocess the data now we need to make SVM model then train it and then predict using test data.



Support Vector Machine

```
In [82]: sgd = Pipeline([('vect', CountVectorizer()),
                         ('tfidf', TfidfTransformer()),
                         ('clf', SGDClassifier(loss='hinge', penalty='l2',alpha=1e-3, random_state=42, max_iter=5, tol=None)),
                        ])
         sgd.fit(train_x, train_y)
```

```
Out[82]: Pipeline(memory=None,
                  steps=[('vect',
                          CountVectorizer(analyzer='word', binary=False,
                                          decode_error='strict',
                                          dtype=<class 'numpy.int64'>, encoding='utf-8',
                                          input='content', lowercase=True, max_df=1.0,
                                          max_features=None, min_df=1,
                                          ngram_range=(1, 1), preprocessor=None,
                                          stop_words=None, strip_accents=None,
                                          token_pattern='(?u)\\b\\w\\w+\\b',
                                          tokenizer=None, vocabulary=Non...
                         ('clf',
                          SGDClassifier(alpha=0.001, average=False, class_weight=None,
                                        early_stopping=False, epsilon=0.1, eta0=0.0,
                                        fit_intercept=True, l1_ratio=0.15,
                                        learning_rate='optimal', loss='hinge',
                                        max_iter=5, n_iter_no_change=5, n_jobs=None,
                                        penalty='l2', power_t=0.5, random_state=42,
                                        shuffle=True, tol=None, validation_fraction=0.1,
                                        verbose=0, warm_start=False))],
                  verbose=False)
```

```
In [83]: y_pred = sgd.predict(test_x)
```

```
In [84]: print('accuracy %s' % accuracy_score(y_pred, test_y))
         print(classification_report(test_y, y_pred,target_names=classes))

         accuracy 0.7093401486988847
```

70% Accuracy

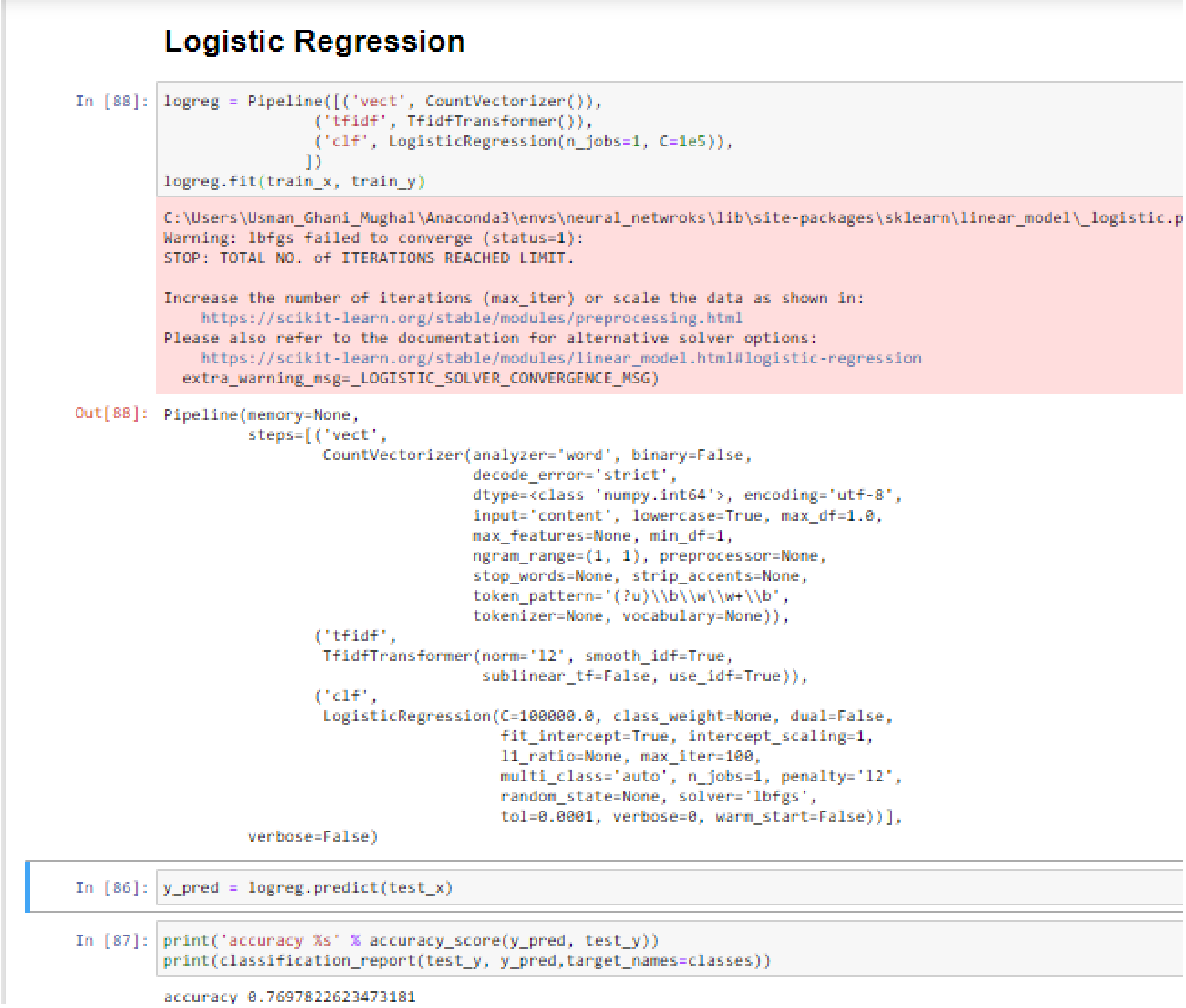
Experiment Five (Logistic Regression)

Build Logistic Regression model and predict news category

Aim: to predict a news belongs to which class or category

Methodologies

Before we have already preprocess the data now we need to make Logistic Regression model then train it and then predict using test data.



Logistic Regression

```
In [88]: logreg = Pipeline([('vect', CountVectorizer()),
                        ('tfidf', TfidfTransformer()),
                        ('clf', LogisticRegression(n_jobs=1, C=1e5)),
                       ])
         logreg.fit(train_x, train_y)

         C:\Users\Usman_Ghani_Mughal\Anaconda3\envs\neural_netwroks\lib\site-packages\sklearn\linear_model\_logistic.p
         Warning: lbfgs failed to converge (status=1):
         STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

         Increase the number of iterations (max_iter) or scale the data as shown in:
             https://scikit-learn.org/stable/modules/preprocessing.html
         Please also refer to the documentation for alternative solver options:
             https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
           extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)

Out[88]: Pipeline(memory=None,
                  steps=[('vect',
                          CountVectorizer(analyzer='word', binary=False,
                                          decode_error='strict',
                                          dtype=<class 'numpy.int64'>, encoding='utf-8',
                                          input='content', lowercase=True, max_df=1.0,
                                          max_features=None, min_df=1,
                                          ngram_range=(1, 1), preprocessor=None,
                                          stop_words=None, strip_accents=None,
                                          token_pattern='(?u)\\b\\w\\w+\\b',
                                          tokenizer=None, vocabulary=None)),
                         ('tfidf',
                          TfidfTransformer(norm='l2', smooth_idf=True,
                                           sublinear_tf=False, use_idf=True)),
                         ('clf',
                          LogisticRegression(C=100000.0, class_weight=None, dual=False,
                                             fit_intercept=True, intercept_scaling=1,
                                             l1_ratio=None, max_iter=100,
                                             multi_class='auto', n_jobs=1, penalty='l2',
                                             random_state=None, solver='lbfgs',
                                             tol=0.0001, verbose=0, warm_start=False))],
                  verbose=False)

In [86]: y_pred = logreg.predict(test_x)

In [87]: print('accuracy %s' % accuracy_score(y_pred, test_y))
         print(classification_report(test_y, y_pred,target_names=classes))

         accuracy 0.7697822623473181
```

Accuracy of 76% better then all.

6.0 Final Results

Algorithm names	Accuracy
Naïve Bayes	54%
Decision Tree	65%
Random Forest	72%
Support Vector Machine	70%
Logistic Regression	76%

As we can see the outputs Logistic regression is on top, then we have Random Forest and then SVM with 70% accuracy and Decision tree and Naive Bayes.
If you want to see other code for Data Preprocessing you can see the '.ipynb' or '.py' file which is in the folder

7.0 References

Statistics (2014)
[Online] Available from: http://www.statistics.com/glossary&term_id=864
[Accessed 09 March 2014]