SPAM REVIEW DETECTION

SWE1017- NATURAL LANGUAGE PROCESSING

PROJECT REPORT (SITE)

SCHOOL OF INFORMATIONTECHNOLOGY

By

SOUMYA S

GUIDED BY: PROF.SIVASHANKARI.R

ABSTRACT:

We present Spam Review Detection System in that we are using Natural Language Processing Techniques. We can find comment is spam or non-spam by using some indicators such as: irregular or discontinuous text flow, vulgar language or not related to specific context and check similarity between the comments. At the end we can able to count total review spam and non-spam count and we can generate graphical representation.

INTRODUCTION:

In recent year, online reviews have become the most important resource of customer opinion. Existing research has been focused on extraction, classification and summarization of opinion from reviews in websites, forums and blogs. Nowadays consumer can obtain information for products and service from online review resources, which can help them make decision. The social tools provided by the content sharing applications allow online user to interact, to express their opinions and to read opinions from other users. But the spammers provide comments which are written intentionally to mislead users by redirecting them to web sites to increase their rating and to promote products less known on the market. Reading spam comments is a bad experience and a waste of time for most of the online users but can also be harming and cause damage to the reader. Several researchers in this field focused on only spam or non-spam comments. But, our goal is to detect comments which are likely to represent spam considering some indicators like a discontinuous flow of text, inadequate and vulgar language or not related to the specific context will helps in giving correct feedback of various customers reviews about given product.

SCOPE OF THE PROJECT:

The major scope of the project is to detect spam/ham messages in E-mail and classify them using sentimental analysis. Naïve Bayes classification is used to classify whether the messages are spam or not spam. The training phase of Bayesian spam filter maintains a database to keep a track of the total number of spam and ham messages to be used to train the Bayesian spam filter.

LITERATURE SURVEY:

S.NO	PAPER NAME	AUTHOR	YEAR	TECHNIQUE USED
1.	Conceptual level similarity measure based review spam detection	Siddu P. Algur	2010	conceptual level similarity measure used for detection of spam review
2.	Unsupervised feature learning framework for no-reference image quality assessment	D. Liang, H. Shen.	2012	unsupervised iterative computation framework
3.	Community discovery in twitter based on user interests	A. Gupta, R. Kaushal	2012	based on a number of features at tweet-level and user-level like Followers/Follows, URLs, Spam Words, Replies and Hash Tags.
4.	Exploiting burstiness in reviews for review spammer detection	H. A. Najada, X Zhu.	2013	supervised classification methods. One of the most effective ways to

				distinguish spam and non-spam reviews is by using machine learning techniques,
5.	Detection of review spam: A survey	X. Yang.	2015	iterative computation framework to detect spam reviews based on coherent examination
6.	A study using n-gram features for text categorization	R. Patel, P. Thakkar.	2015	n-gram techniques is extended by means of feature selection and different representation of the opinions. The problem is modelled as the classification problem and Naïve Bayes (NB) classifier and Least Squares Support Vector Machine (LS-SVM) are used on three

				different representations (Boolean, bag-of-words and term frequency–inverse document frequency (TF-IDF)) of the opinions.
7.	Detecting spammers on social networks	M. L. Ramprasad, M. G. Amudha.	2010	propose a mean to enhance the users‟ connectivity by taking benefit of friend recommendation and spammer detection of the online videos.
8.	Spotting fake reviews via collective positive-unlabeled learning	H. Li, Z. Chen, B. Liu, X. Wei, J. Shao	2014	Dianping‟s algorithm has a very high precision. collective classification algorithm called Multi-typed Heterogeneous Collective

				Classification (MHCC) and then extend it to Collective Positive and Unlabeled learning (CPU).
9.	Towards online anti-opinion spam: Spotting fake reviews from the review sequence	Y. Lin, H. Wu, J. Zhang, X. Wang, A. Zhou.	2014	identify the fake reviews orderly with high precision and recall.
10.	An approach to rank reviews by fusing and mining opinions based on review pertinence	J. Z. Wang, Z. Yan, L. T. Yang, B. X. Huang	2015	Review Pertinence" to study the degree of this relevance. Unlike usual methods, they measure the pertinence of review by considering not only the similarity between a review and its corresponding article, but also the correlation among reviews

MODULES:

The architecture of proposed system is composed of three main modules:

- Feature extraction module.
- Post-comment similarity module
- Topic extraction module.

MODULE DESCRIPTION:

1. FEATURE EXTRACTION MODULE:

In feature extraction module the proposed system eliminate comments which are discontinuous and contain vulgar expressions. The identification of these types of comments relies on the identification of countable features: links, white spaces, sentences, punctuation marks, word duplication, stop words, non ASCII characters, new line, and capital letter.

We are implement feature extraction module based on some characteristics of spam comments:

a) Number of links in the given comments.
b) Number of white spaces in the given comments.
c) Number of sentences in the given comment.
d) Number of punctuation marks in the given comment.
e) Comment Word duplication.
f) Comment Stop words ratio.
g) Number of Non ASCII Characters in the given comment.

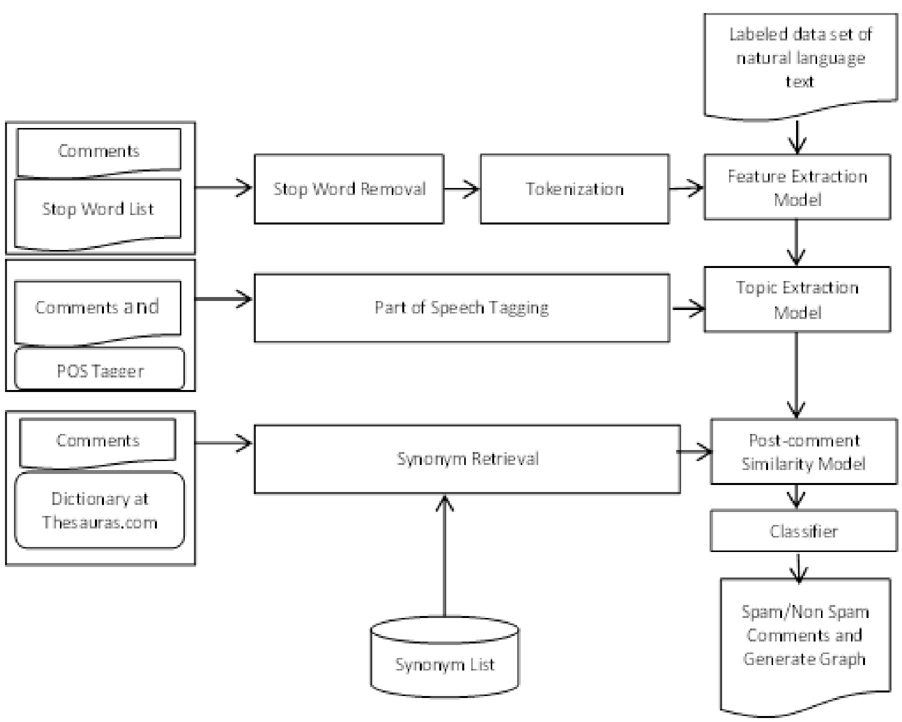
2. POST-COMMENT SIMILARITY:

This module detects whether the comment and post contain similar topic based on the following similarity metric: the normalized value of the sum of the frequencies of occurrences of each word and its synonyms from the comments in the post. The post- comment similarity module which connect to online directory to retrieve all the synonyms for the word in the comment. After the synonyms retrieval process the post-comment similarity formula is applied and the post-comment similarity degree is calculated.

3. TOPIC EXTRACTION MODULE:

The topic extraction module is designed to determine if there are common topic between a comment and a post and to find out if the contents of the comment are related to the content of the related post. The proposed system considers two basic types of topics – bigrams and uni-grams which are extracted using combination of shallow natural language processing technique. To identify uni-gram topic system extracts a collection of candidate nouns. To create set of bigrams topic system extract all bigrams from both the comments and the post which conform to one of two basic part-of-speech co-location patterns.

SYSTEM ARCHITECTURE:



SYSTEM TECHNIQUES:

BAYESIAN CLASSIFICATION:

The purpose of spam filters is to decide whether an incoming message is legitimate (i.e., ham) or unsolicited (i.e., spam). There are many different types of filter systems, including:

Word lists: Simple and complex lists of words that are known to be associated with spam.

Black lists and white lists: These lists contain known IP addresses of spam and non-spam senders respectively.

The training phase of Bayesian spam filter maintains a database to keep a track of the total number of spam and ham messages to be used to train the Bayesian spam filter. The training phase of the filter consists of splitting the decoded message into single tokens, which are the words that make up the message. For each token, a record in the token database is updated that maintains two counts: the number of spam messages and the number of ham messages in which that token has been observed so far.

Once a Bayesian spam filter has created a token database, messages can be analyzed. Just like the training phase, the message is first decoded and split into single tokens. For each token, a spam probability is calculated based on the number of spam and ham messages that have contained this token out of the total number of spam and ham messages that have been used to train the Bayesian spam filter.

ALGORITHM:

ALGORITHM 1 : NAÏVE BAYES CLASSFICATION

MACHINE LEARNING TECHNIQUE:

- SUPPORT VECTOR MACHINE
- LINEAR AND POLYNOMIAL REGRESSION

INPUT: Ham and Spam dataset is given as input to detect the spam messages

OUTPUT: Accuracy – 98% with confusion matrix.

SYSTEM REQUIREMENTS:

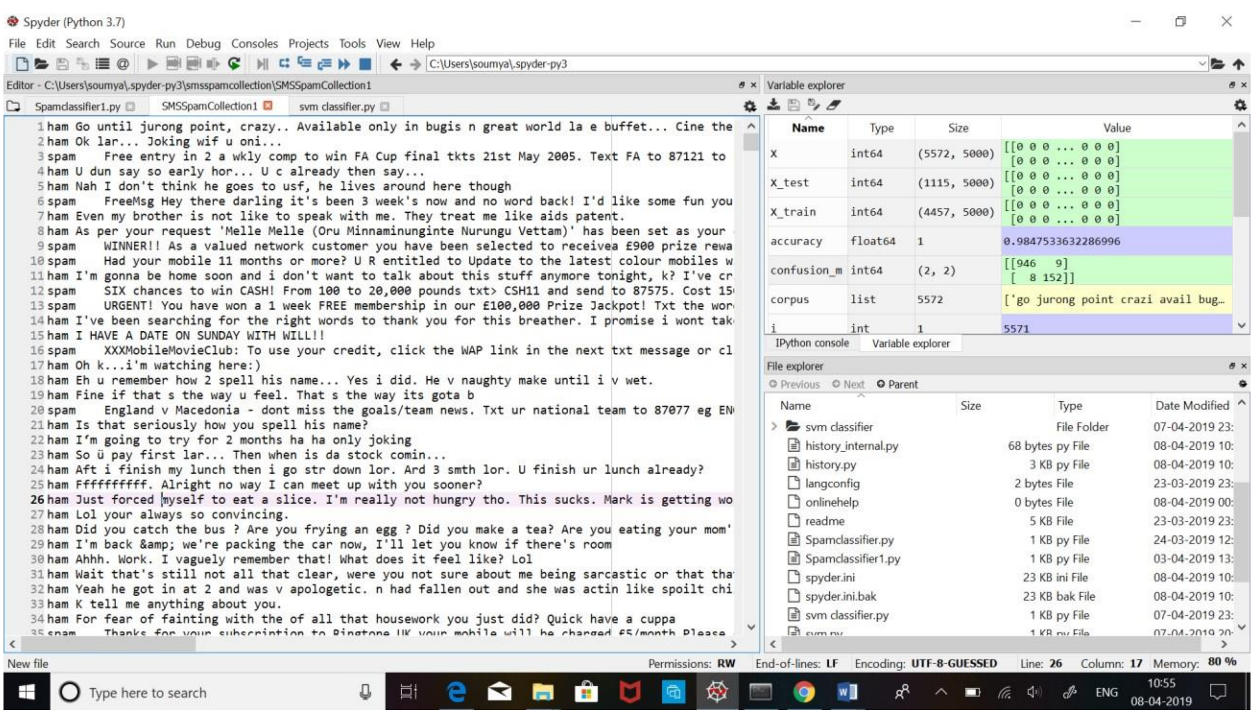
SOFTWARE REQUIREMENTS

Language used : Spyder(Python 3.7.1)

Operating System : Windows 10

IDE : Anaconda, Natural Language ToolKit, Jupyter notebook

DATASET:



DATASET DESCRIPTION:

Collection of Ham and Spam messages is taken as dataset and the SMS Spam Collection is a public set of SMS labeled messages that have been collected for mobile phone spam research.

Data Set Characteristics:	Multivariate, Text, Domain-Theory	Number of Instances:	5574	Area:	Computer
Attribute Characteristics:	Real	Number of Attributes:	N/A	Date Donated	2012-06-22
Associated Tasks:	Classification, Clustering	Missing Values?	N/A	Number of Web Hits:	228190

Data Set Information:

This corpus has been collected from free or free for research sources at the Internet:

-> A collection of 425 SMS spam messages was manually extracted from the Grumbletext Web site. This is a UK forum in which cell phone users make public claims about SMS spam messages, most of them without reporting the very spam message received. The identification of the text of spam messages in the claims is a very hard and time-consuming task, and it involved carefully scanning hundreds of web pages.
-> A subset of 3,375 SMS randomly chosen ham messages of the NUS SMS Corpus (NSC), which is a dataset of about 10,000 legitimate messages collected for research at the Department of Computer Science at the National University of Singapore. The messages largely originate from Singaporeans and mostly from students attending the University. These messages were collected from volunteers who were made aware that their contributions were going to be made publicly available.
-> A list of 450 SMS ham messages collected from Caroline Tag’s PhD Thesis available at [Web Link].
-> Finally, we have incorporated the SMS Spam Corpus v.0.1 Big. It has 1,002

SMS ham messages and 322 spam messages and it is public available at: [Web Link].

Attribute Information:

The collection is composed by just one text file, where each line has the correct class followed by the raw message. We offer some examples bellow:

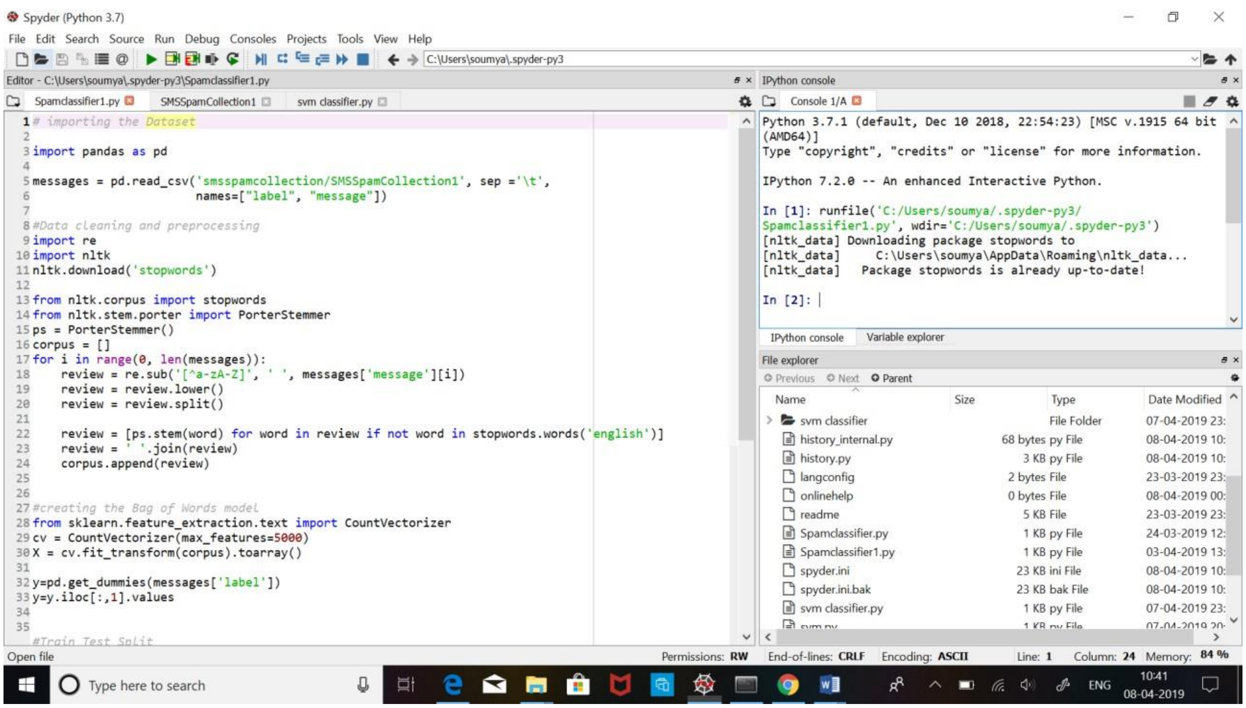
ham What you doing?how are you?
ham Ok lar... Joking wif u oni...
ham dun say so early hor... U c already then say...
ham MY NO. IN LUTON 0125698789 RING ME IF UR AROUND! H*
ham Siva is in hostel aha:-.
ham Cos i was out shopping wif darren jus now n i called him 2 ask wat present he wan lor. Then he started guessing who i was wif n he finally guessed darren lor.
spam FreeMsg: Txt: CALL to No: 86888 & claim your reward of 3 hours talk time to use from your phone now! ubscribe6GBP/ mnth inc 3hrs 16 stop?txtStop
spam Sunshine Quiz! Win a super Sony DVD recorder if you canname the capital of Australia? Text MQUIZ to 82277. B
spam URGENT! Your Mobile No 07808726822 was awarded a L2,000 Bonus Caller Prize on 02/09/03! This is our 2nd attempt to contact YOU! Call 0871-872-9758 BOX95QU

PROPOSED METHOD EXPLANATION:

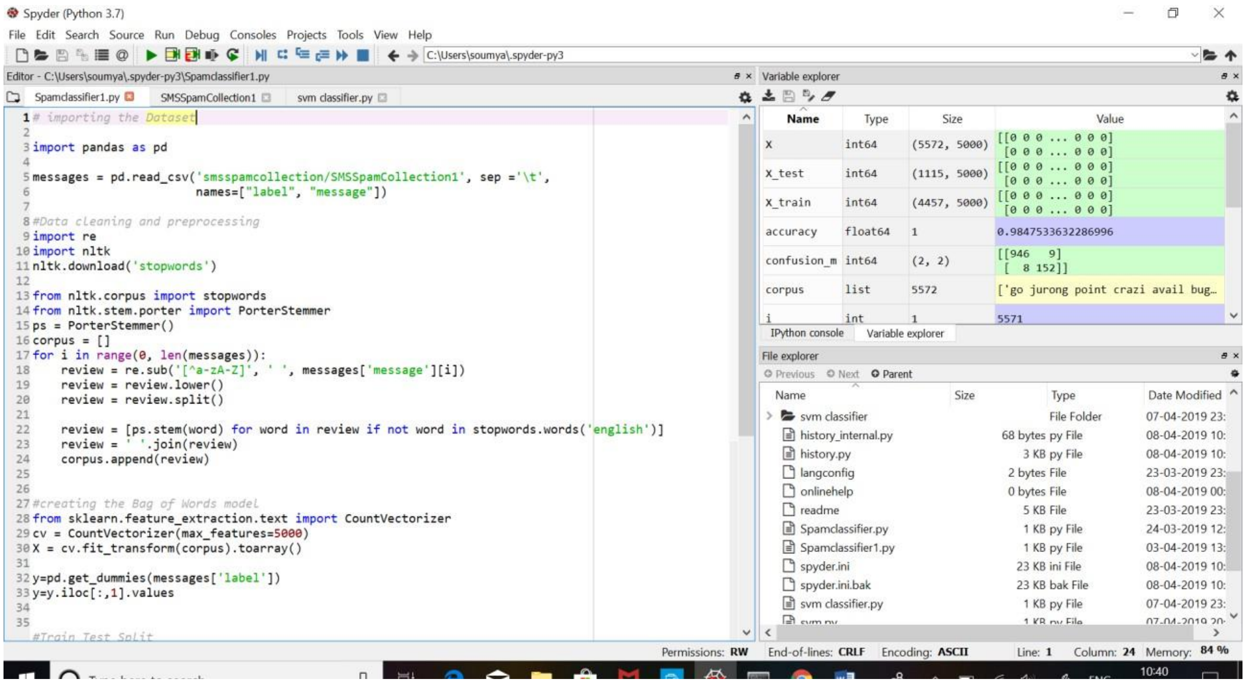
In our research show that it is possible to detect spam comments with the proper selection of features which capture different characteristics of legitimate comments in order to differentiate them from spam comments. In our experiment we consider as spam the following type of documents associated with a review by using some indicators: (i) incoherent comments with increased number of punctuation marks, new lines, stop words, non ASCII characters and white spaces, (ii) inadequate comments which contain offensive words and (iii) coherent comments which do not provide relevant content to a specific topic.Our experiment we makes use of natural language processing techniques in order to identify the relevant features of spam comments. We propose a supervised learning approach and experiment different sets of features to correctly classify comments as spam or not.

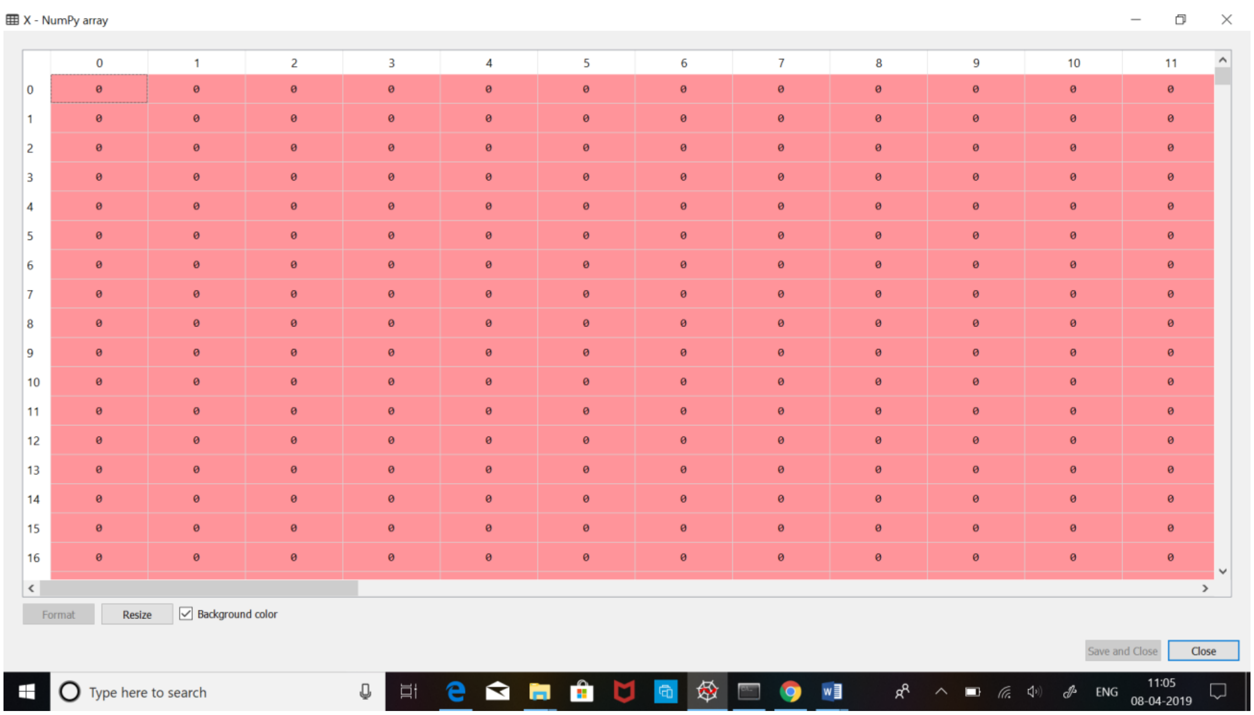
SNAPSHOTS : Using Naïve Bayes Classification

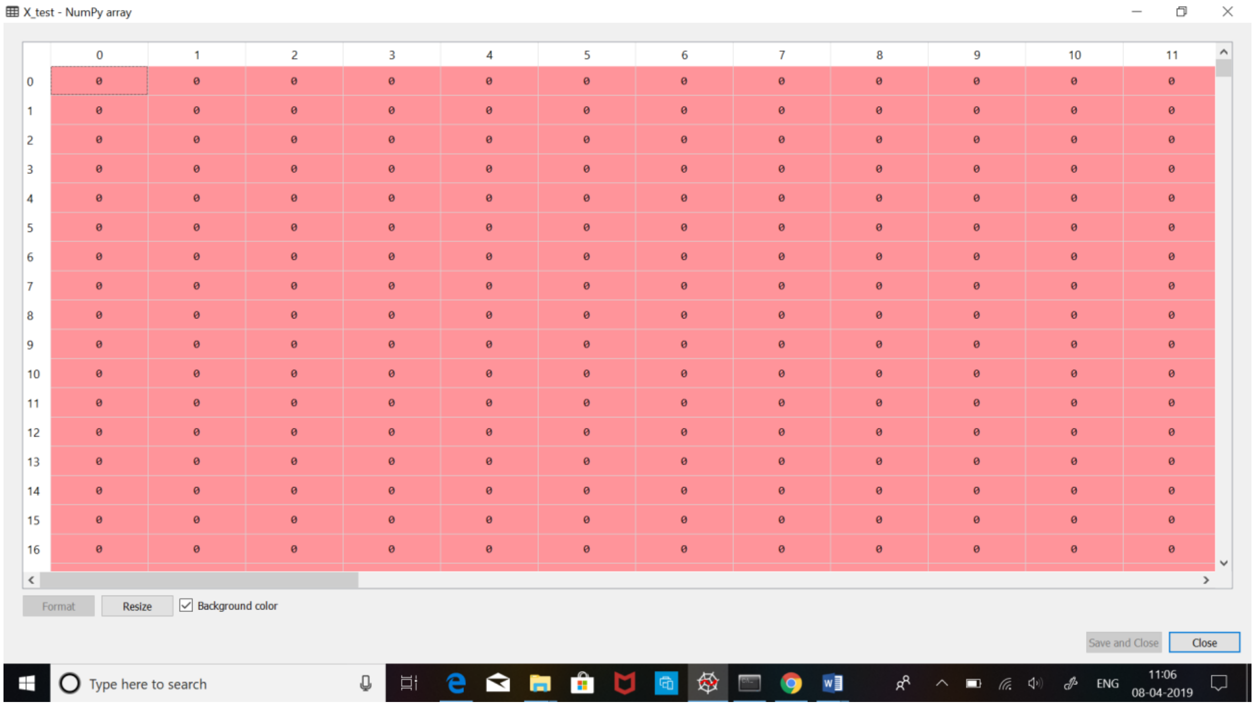
Spam classifier code and console output:



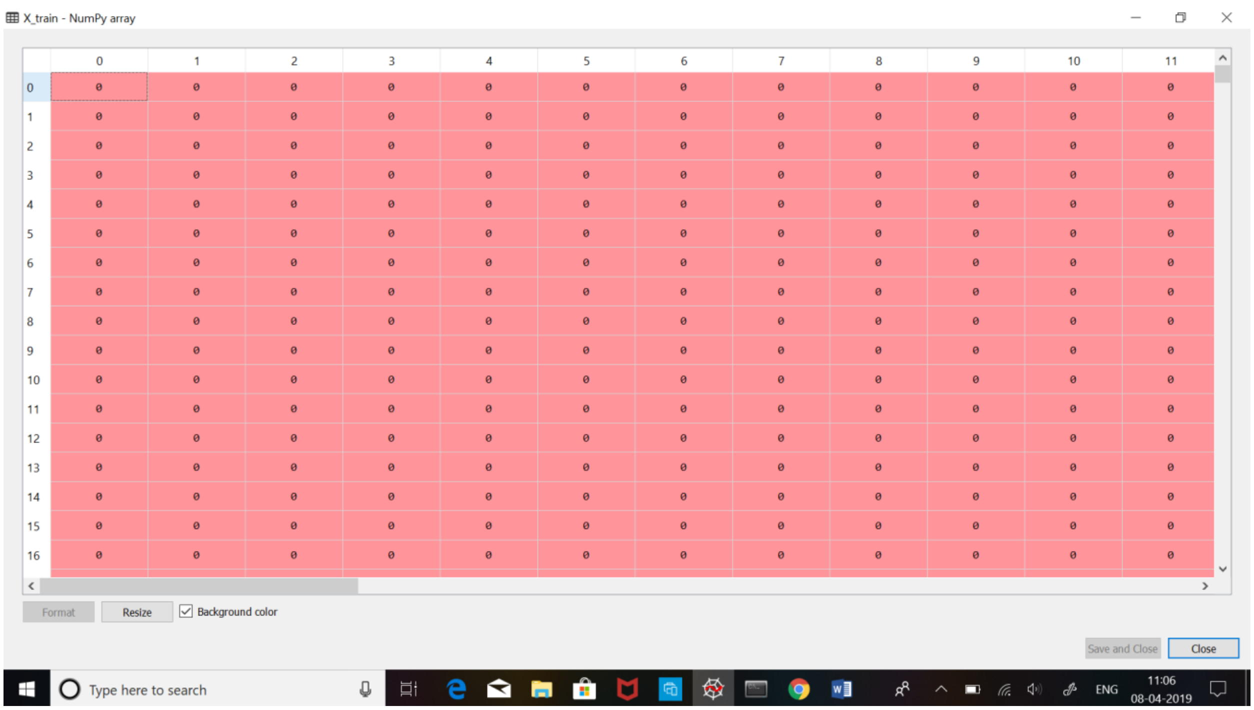
Spam classifier with Variable explorer output:


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\soumya\.spyder-py3
Editor - C:\Users\soumya\.spyder-py3\Spamclassifier1.py
Spamclassifier1.py
SMSSpamCollection1
svm classifier.py
# importing the Dataset
import pandas as pd
messages = pd.read_csv('smsspamcollection/SMSSpamCollection1', sep ='\t',
names=["label", "message"])
#Data cleaning and preprocessing
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
corpus = []
for i in range(0, len(messages)):
review = re.sub('[^a-zA-Z]', ' ', messages['message'][i])
review = review.lower()
review = review.split()
review = [ps.stem(word) for word in review if not word in stopwords.words('english')]
review = ' '.join(review)
corpus.append(review)
#creating the Bag of Words model
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000)
X = cv.fit_transform(corpus).toarray()
y=pd.get_dummies(messages['label'])
y=y.iloc[:,1].values
#Train Test Split
Variable explorer
Name Type Size Value
X int64 (5572, 5000) [[0 0 0 ... 0 0 0] [0 0 0 ... 0 0 0]
X_test int64 (1115, 5000) [[0 0 0 ... 0 0 0] [0 0 0 ... 0 0 0]
X_train int64 (4457, 5000) [[0 0 0 ... 0 0 0] [0 0 0 ... 0 0 0]
accuracy float64 1 0.9847533632286996
confusion_m int64 (2, 2) [[946 9] [8 152]]
corpus list 5572 ['go jurong point crazi avail bug…
i int 1 5571
IPython console
Variable explorer
File explorer
Previous Next Parent
Name Size Type Date Modified
svm classifier File Folder 07-04-2019 23:
history_internal.py 68 bytes py File 08-04-2019 10:
history.py 3 KB py File 08-04-2019 10:
langconfig 2 bytes File 23-03-2019 23:
onlinehelp 0 bytes File 08-04-2019 00:
readme 5 KB File 23-03-2019 23:
Spamclassifier.py 1 KB py File 24-03-2019 12:
Spamclassifier1.py 1 KB py File 03-04-2019 13:
spyder.ini 23 KB ini File 08-04-2019 10:
spyder.ini.bak 23 KB bak File 08-04-2019 10:
svm classifier.py 1 KB py File 07-04-2019 23:
Permissions: RW End-of-lines: CRLF Encoding: ASCII Line: 1 Column: 24 Memory: 84 %
10:40


X - NumPy array
0 1 2 3 4 5 6 7 8 9 10 11
Format
Resize
Background color
Save and Close
Close
Type here to search
ENG
11:05
08-04-2019



	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0



	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0

confusion_m - NumPy array

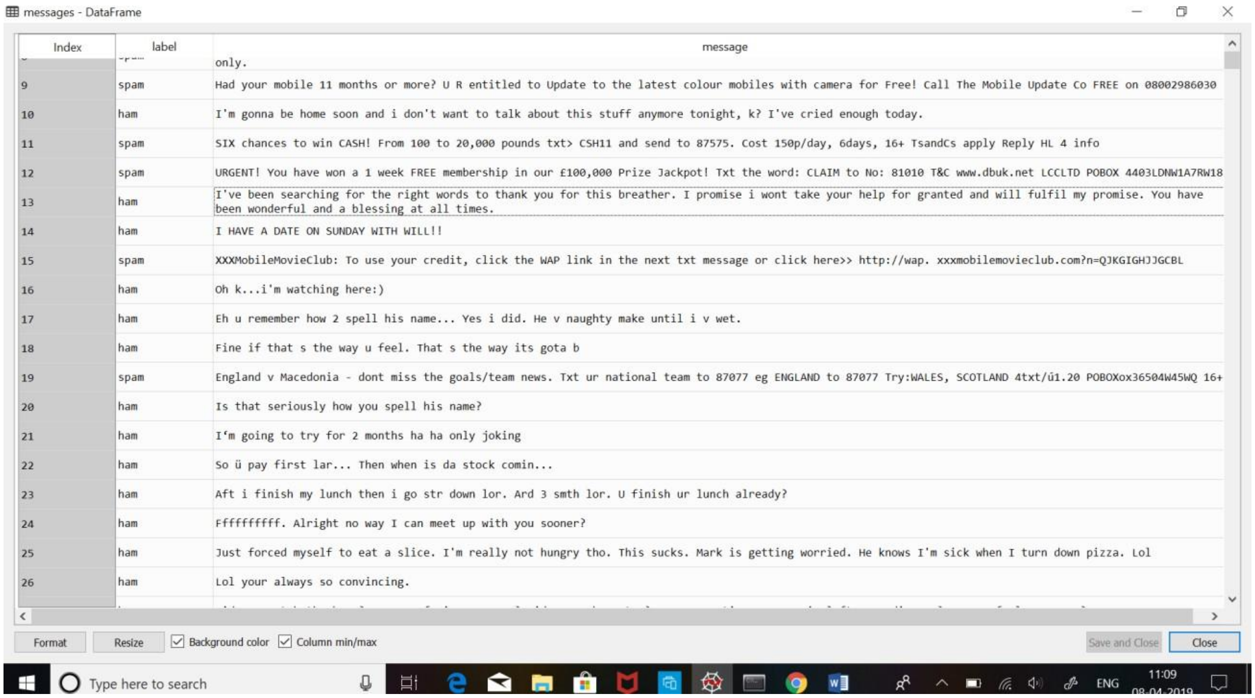
	0	1
0	946	9
1	8	152

corpus - List (5572 elements)

Index	Type	Size	Value
0	str	1	go jurong point crazi avail bugi n great world la e buffet cine got am ...
1	str	1	ok lar joke wif u oni
2	str	1	free entri wkli comp win fa cup final tkt st may text fa receiv entri ...
3	str	1	u dun say earli hor u c alreadi say
4	str	1	nah think goe usf live around though
5	str	1	freemsg hey darl week word back like fun still tb ok xxx std chg send ...
6	str	1	even brother like speak treat like aid patent
7	str	1	per request mell mell oru minnaminungint nurungu vettam set callertun ...
8	str	1	winner valu network custom select receivea prize reward claim call cla ...
9	str	1	mobil month u r entitl updat latest colour mobil camera free call mobi ...
10	str	1	gonna home soon want talk stuff anymor tonight k cri enough today
11	str	1	six chanc win cash pound txt csh send cost p day day tsandc appli repl ...
12	str	1	urgent week free membership prize jackpot txt word claim c www dbuk ne ...
13	str	1	search right word thank breather promis wont take help grant fulfil pr ...
14	str	1	date sunday
15	str	1	xxxmobilemovieclub use credit click wap link next txt messag click htt ...
16	str	1	oh k watch
17	str	1	eh u rememb spell name ye v naughti make v wet
18	str	1	fine way u feel way gota b

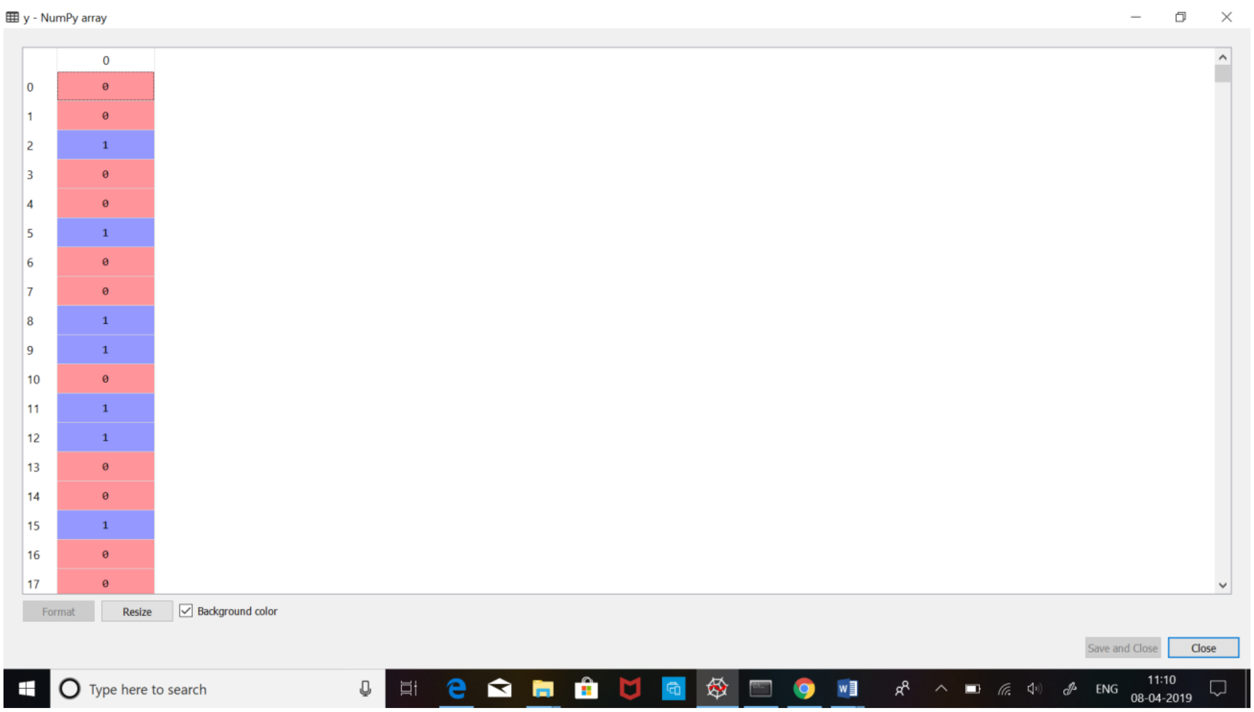
Save and Close Close

Type here to search ENG 11:07


messages - DataFrame

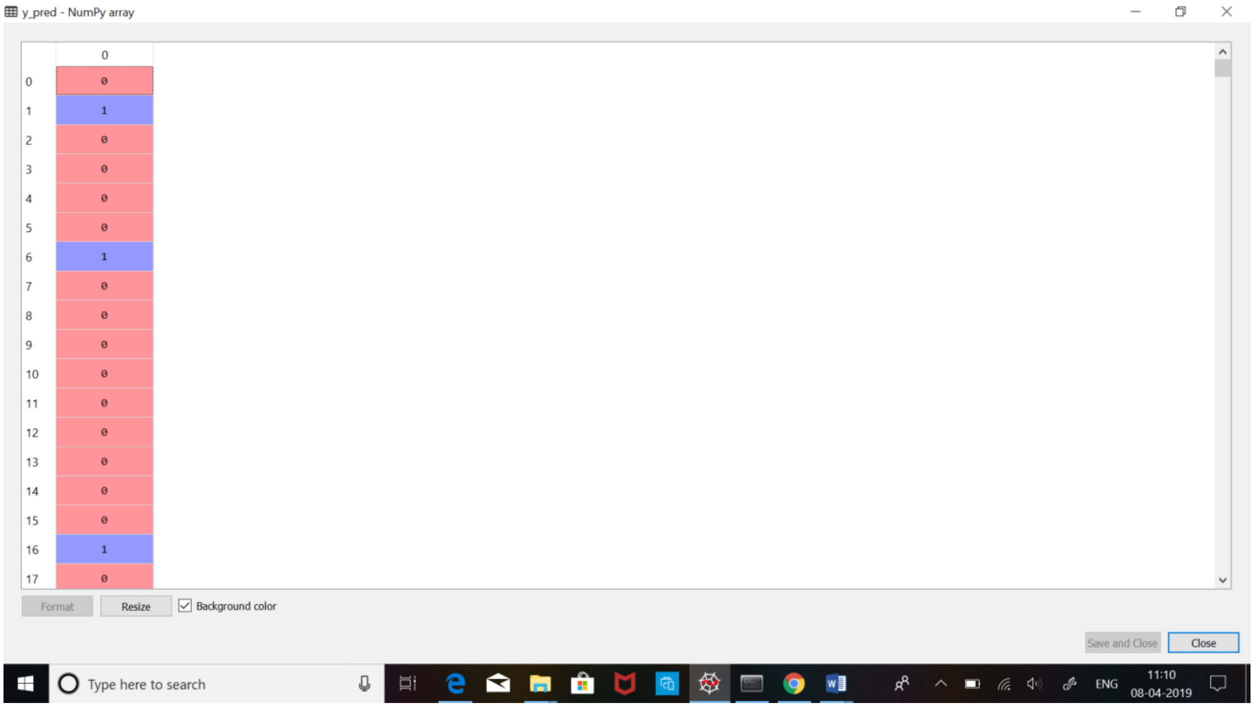
Index	label	message
		only.
9	spam	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030
10	ham	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.
11	spam	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info
12	spam	URGENT! You have won a 1 week FREE membership in our £100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18
13	ham	I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times.
14	ham	I HAVE A DATE ON SUNDAY WITH WILL!!
15	spam	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL
16	ham	Oh k...i'm watching here:)
17	ham	Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
18	ham	Fine if that s the way u feel. That s the way its gota b
19	spam	England v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/ú1.20 POBOXox36504W45WQ 16+
20	ham	Is that seriously how you spell his name?
21	ham	I‘m going to try for 2 months ha ha only joking
22	ham	So ü pay first lar... Then when is da stock comin...
23	ham	Aft i finish my lunch then i go str down lor. Ard 3 smth lor. U finish ur lunch already?
24	ham	Fffffffff. Alright no way I can meet up with you sooner?
25	ham	Just forced myself to eat a slice. I'm really not hungry tho. This sucks. Mark is getting worried. He knows I'm sick when I turn down pizza. Lol
26	ham	Lol your always so convincing.

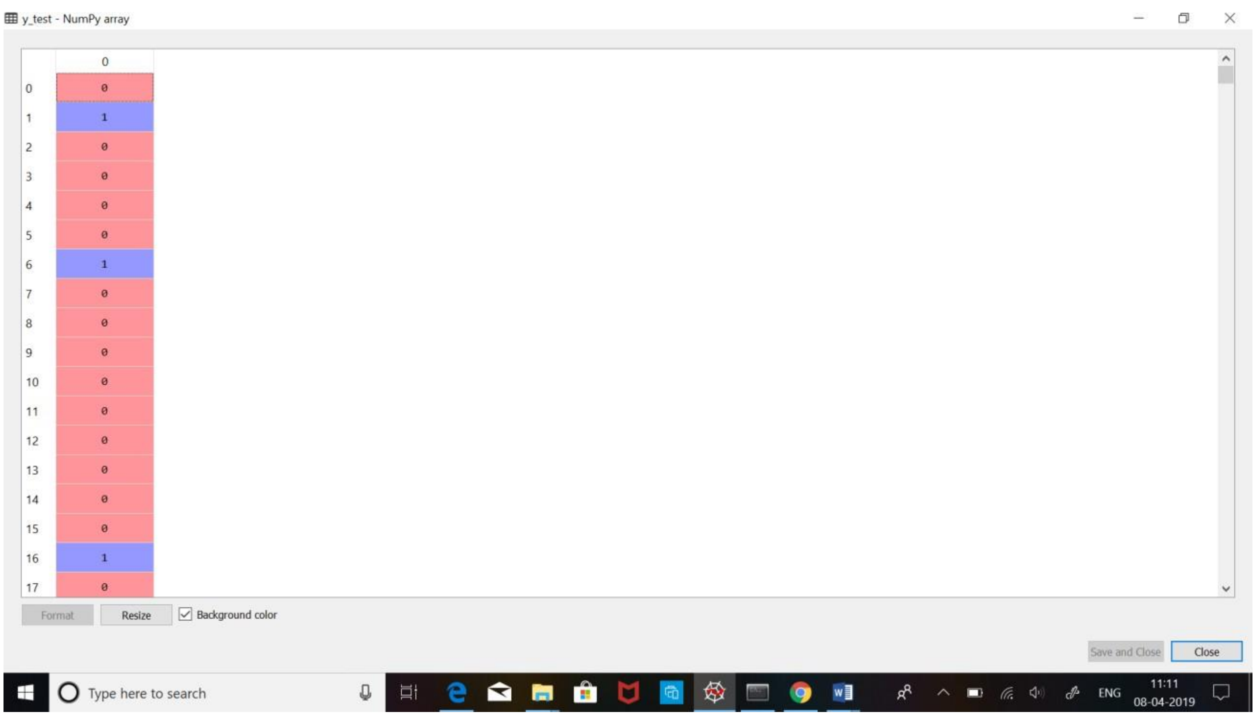
Format Resize Background color Column min/max Save and Close Close

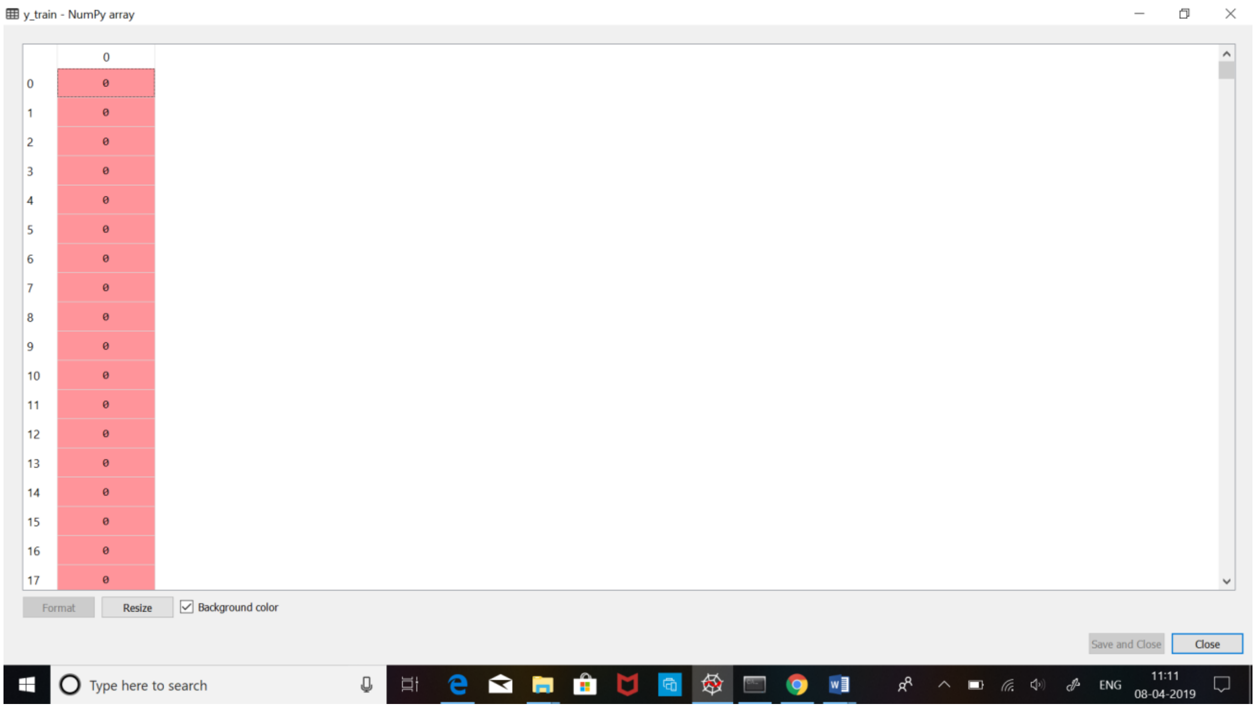

y - NumPy array

	0
0	0
1	0
2	1
3	0
4	0
5	1
6	0
7	0
8	1
9	1
10	0
11	1
12	1
13	0
14	0
15	1
16	0
17	0

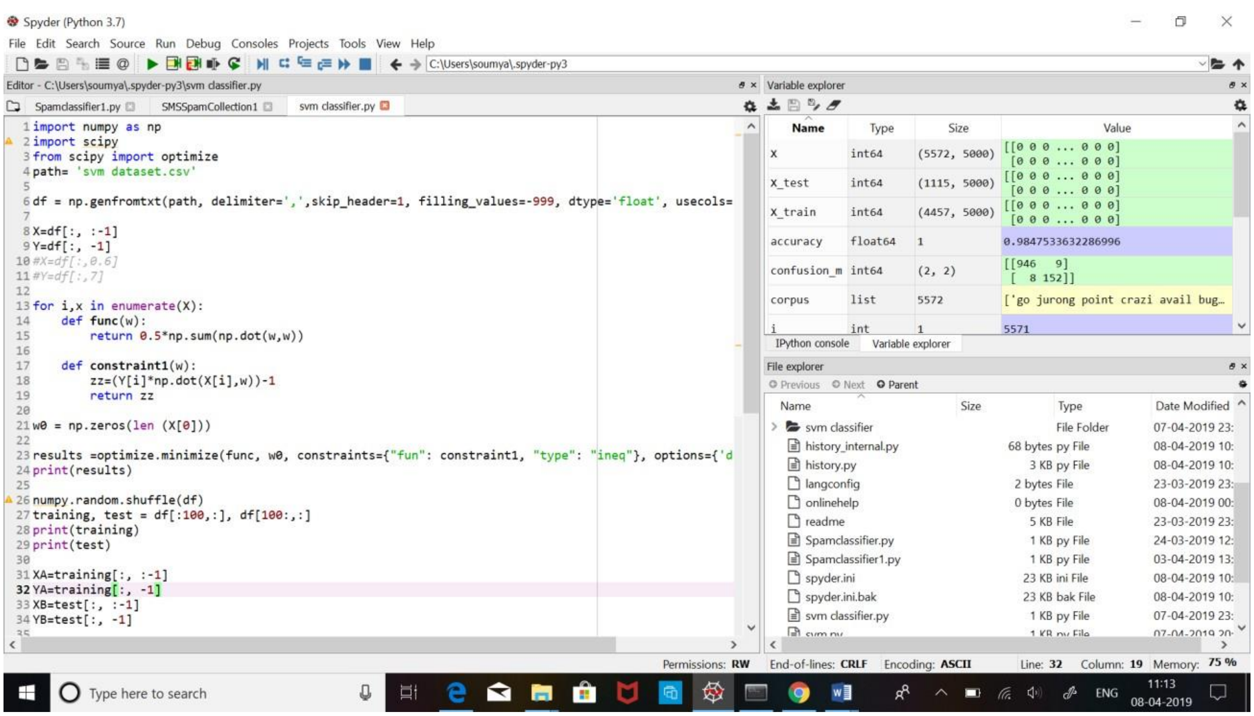
Format Resize Background color Save and Close Close

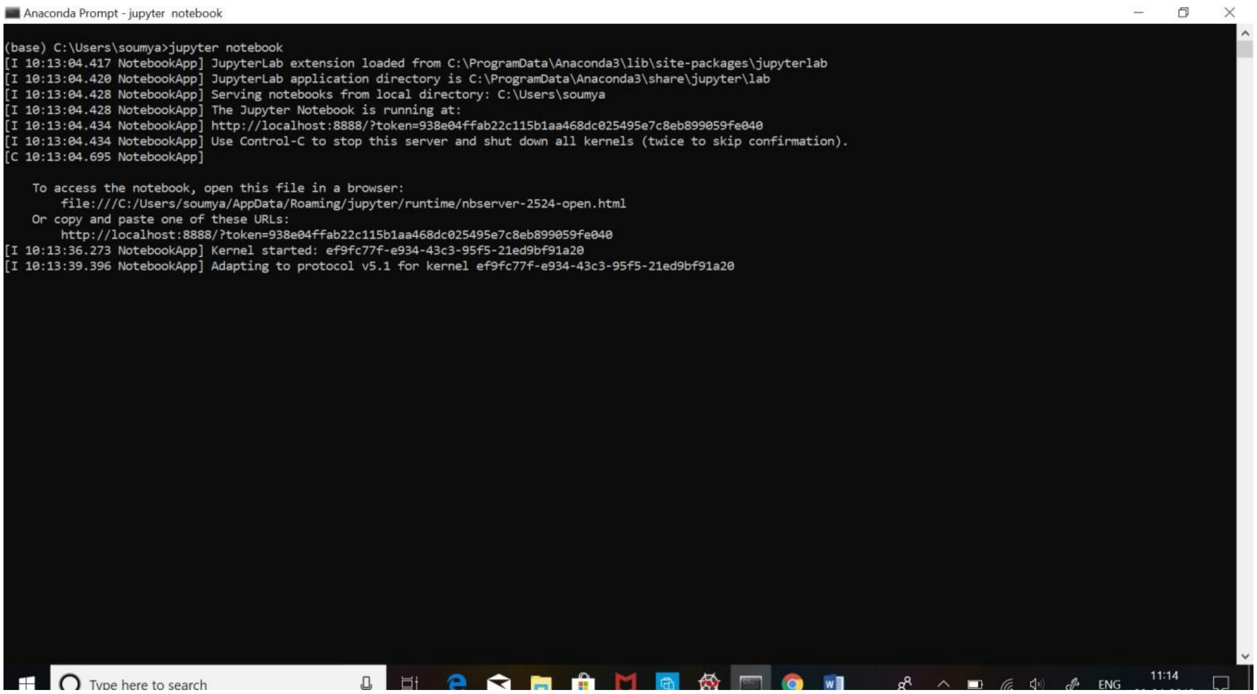

y_pred - NumPy array
0
0 0
1 1
2 0
3 0
4 0
5 0
6 1
7 0
8 0
9 0
10 0
11 0
12 0
13 0
14 0
15 0
16 1
17 0
Format
Resize
Background color
Save and Close
Close
Type here to search
ENG
11:10
08-04-2019


y_test - NumPy array
0
0 0
1 1
2 0
3 0
4 0
5 0
6 1
7 0
8 0
9 0
10 0
11 0
12 0
13 0
14 0
15 0
16 1
17 0
Format
Resize
Background color
Save and Close
Close
Type here to search
ENG
11:11
08-04-2019

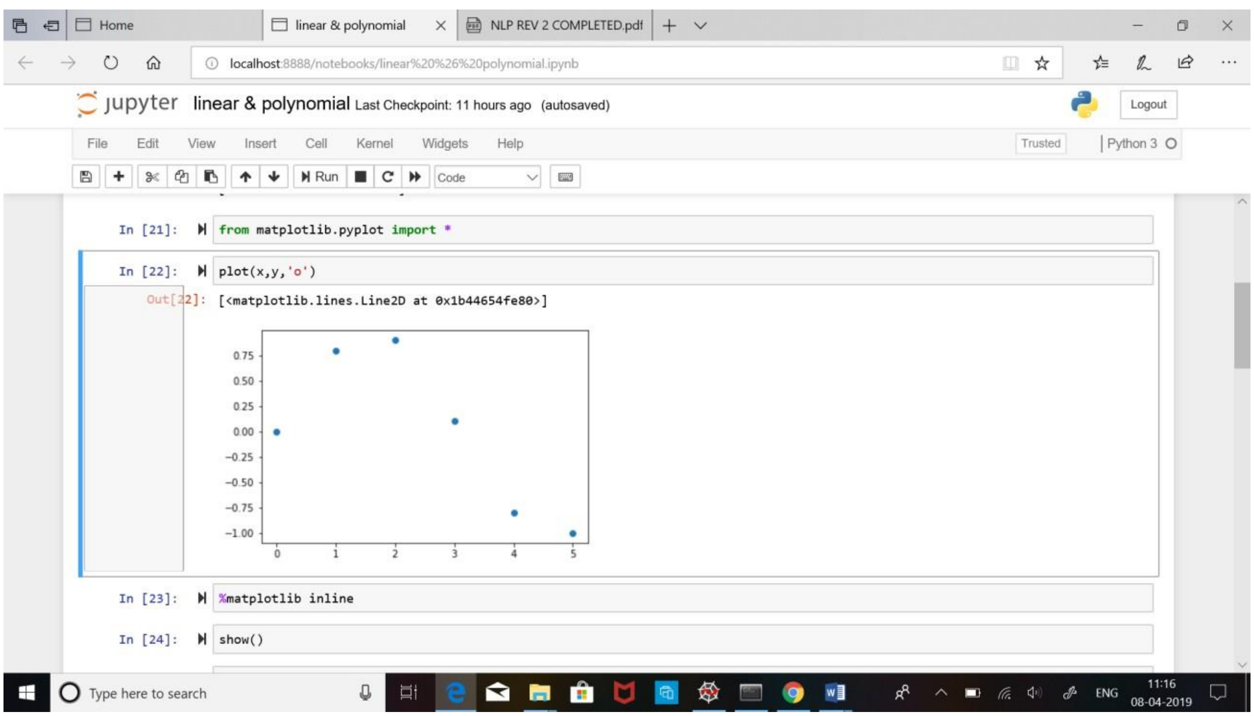


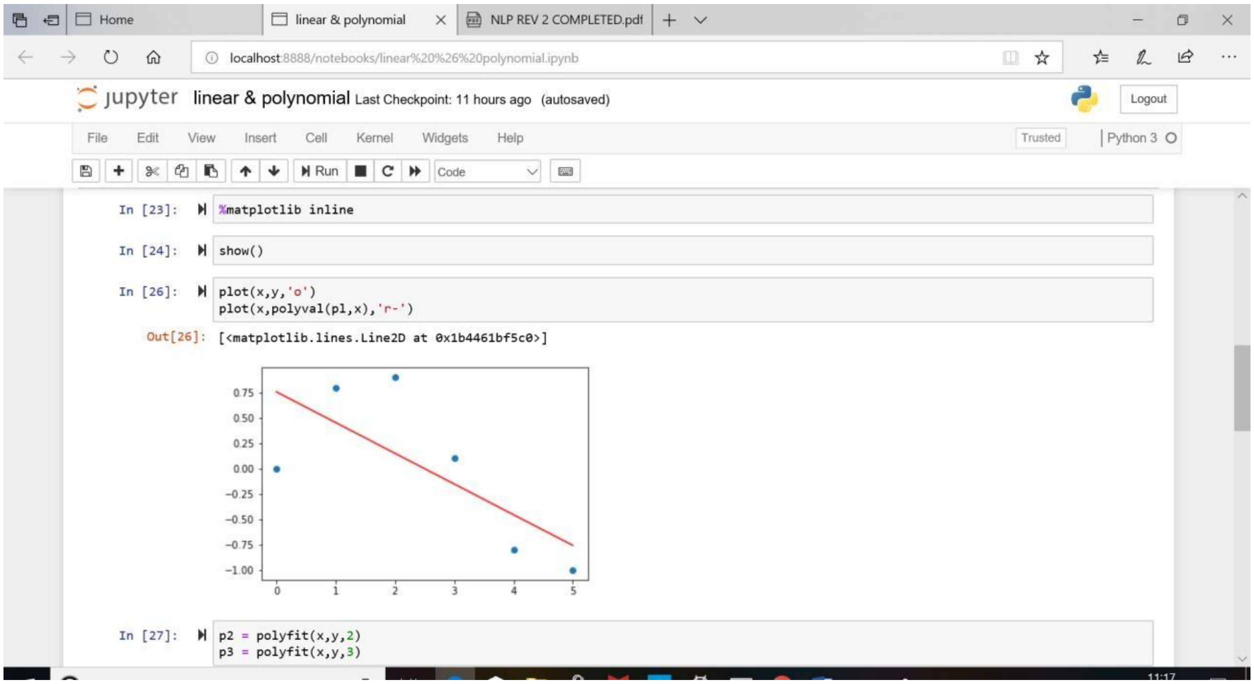
SVM CLASSIFIER:



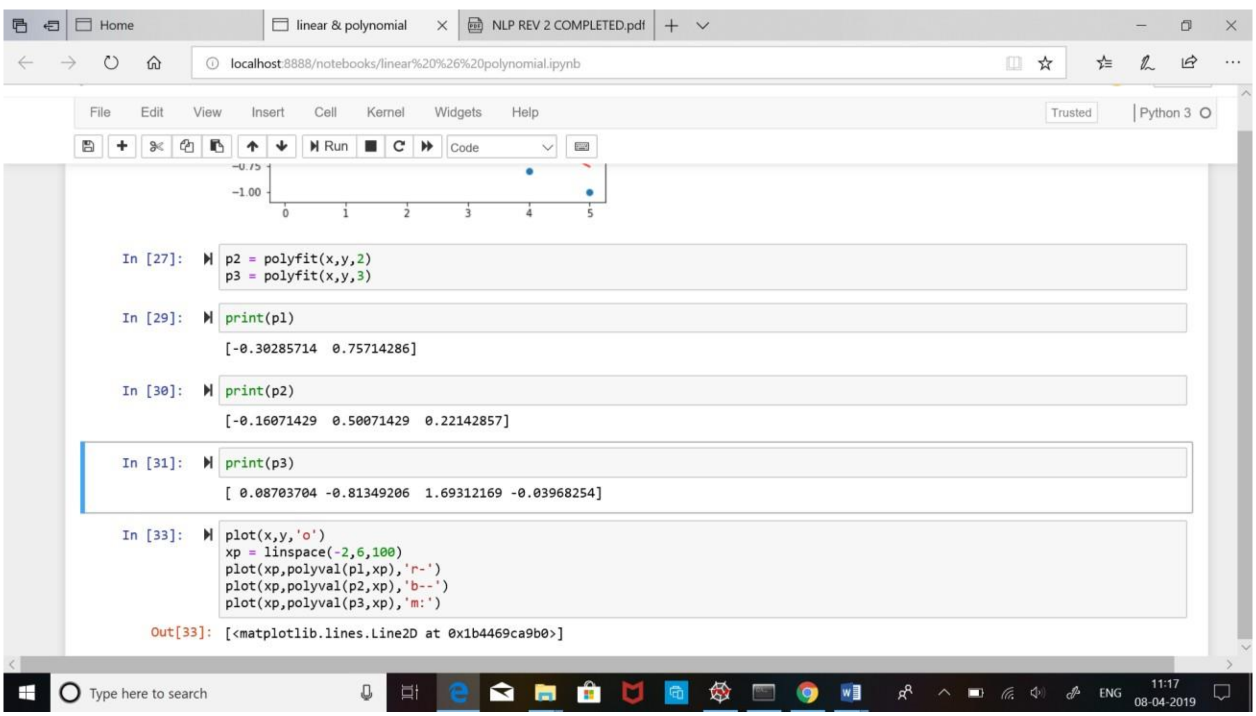


JUPYTER NOTEBOOK:

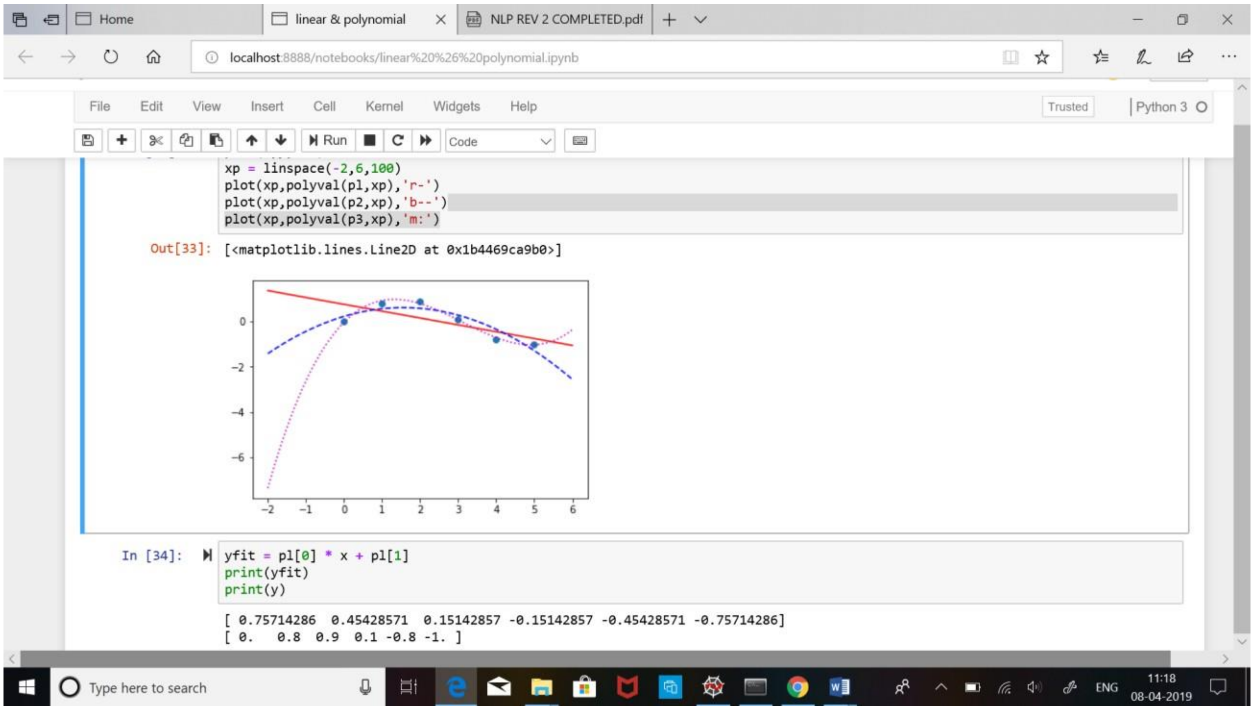


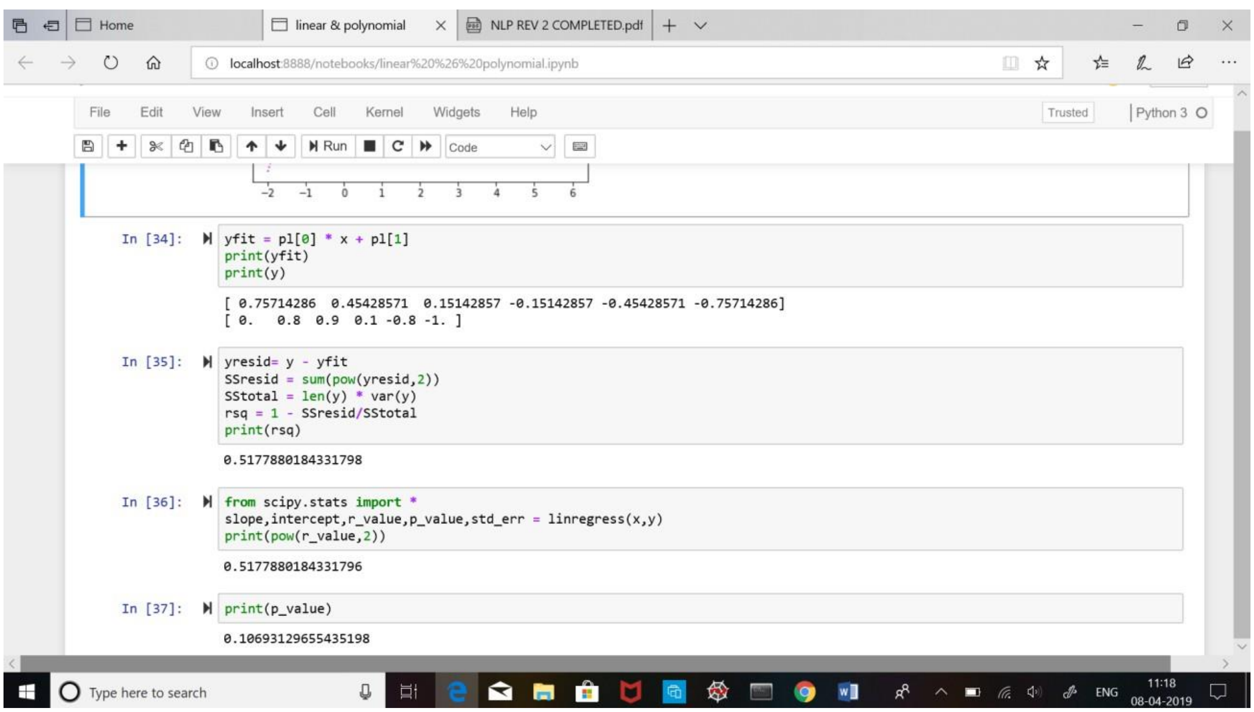


Home
linear & polynomial
NLP REV 2 COMPLETED.pdf
localhost:8888/notebooks/linear%20%26%20polynomial.ipynb
jupyter linear & polynomial Last Checkpoint: 11 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [23]: %matplotlib inline
In [24]: show()
In [26]: plot(x,y,'o')
plot(x,polyval(p1,x),'r-')
Out[26]: [<matplotlib.lines.Line2D at 0x1b4461bf5c0>]
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
1
2
3
4
5
In [27]: p2 = polyfit(x,y,2)
p3 = polyfit(x,y,3)



Home
linear & polynomial
NLP REV 2 COMPLETED.pdf
localhost:8888/notebooks/linear%20%26%20polynomial.ipynb
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
−1.00
0
1
2
3
4
5
In [27]: p2 = polyfit(x,y,2)
p3 = polyfit(x,y,3)
In [29]: print(p1)
[-0.30285714 0.75714286]
In [30]: print(p2)
[-0.16071429 0.50071429 0.22142857]
In [31]: print(p3)
[0.08703704 -0.81349206 1.69312169 -0.03968254]
In [33]: plot(x,y,'o')
xp = linspace(-2,6,100)
plot(xp,polyval(p1,xp),'r-')
plot(xp,polyval(p2,xp),'b--')
plot(xp,polyval(p3,xp),'m:')
Out[33]: [<matplotlib.lines.Line2D at 0x1b4469ca9b0>]
Type here to search
ENG
11:17
08-04-2019





IMPLEMENTATION :

PYTHON CODE FOR NAÏVE BAYES CLASSIFICATION:

```
# importing the Dataset

import pandas as pd

messages = pd.read_csv('smsspamcollection/SMSSpamCollection1', sep ='\t',
                       names=["label", "message"])

#Data cleaning and preprocessing
import re
import nltk
nltk.download('stopwords')

from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
corpus = []
for i in range(0, len(messages)):
    review = re.sub('[^a-zA-Z]', ' ', messages['message'][i])
    review = review.lower()
```

```
    review = review.split()

    review = [ps.stem(word) for word in review if not word in stopwords.words('english')]
    review = ' '.join(review)
    corpus.append(review)


#creating the Bag of Words model
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000)
X = cv.fit_transform(corpus).toarray()

y=pd.get_dummies(messages['label'])
y=y.iloc[:,1].values


#Train Test Split
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)

#Training model using Naive bayes classifier

from sklearn.naive_bayes import MultinomialNB
spam_detect_model = MultinomialNB().fit(X_train, y_train)

y_pred=spam_detect_model.predict(X_test)

from sklearn.metrics import confusion_matrix
confusion_m=confusion_matrix(y_test,y_pred)

from sklearn.metrics import accuracy_score
accuracy=accuracy_score(y_test,y_pred)
```

SUPPORT VECTOR MACHINE (SVM) CODE:

```
import numpy as np
```

```
import scipy
from scipy import optimize
path= 'svm dataset.csv'

df = np.genfromtxt(path, delimiter=',',skip_header=1, filling_values=-999, dtype='float', usecols=[0,1,2,3,4,5,6,7])

X=df[:, :-1]
Y=df[:, -1]
#X=df[:,0.6]
#Y=df[:,7]

for i,x in enumerate(X):
    def func(w):
        return 0.5*np.sum(np.dot(w,w))

    def constraint1(w):
        zz=(Y[i]*np.dot(X[i],w))-1
        return zz
```

```
w0 = np.zeros(len (X[0]))


results =optimize.minimize(func, w0, constraints={"fun": constraint1, "type":
"ineq"}, options={'disp': True})

print(results)


numpy.random.shuffle(df)

training, test = df[:100,:], df[100:,:]

print(training)

print(test)


XA=training[:, :-1]

YA=training[:, -1]

XB=test[:, :-1]

YB=test[:, -1]


for i,x in enumerate(XA):

    def func(w):
```

```
        return 0.5*np.sum(np.dot(w,w))


    def constraint1(w):

        zz = (YA[i]*np.dot(XA[i],w))-1

        return zz

w0 = np.zeros(len(XA[0]))


results =optimize.minimize(func, w0, constraints={"fun": constraint1, "type": "ineq"},options={'disp':True})

print(results)


w2 = results.x


for i,x in enumerate(XB):

    z3=(1-(np.dot(XB[i],w2)))


    if(z3 >= 1.0):

        z4 = 1.0

    elif (z3 <= -1.0):
```

```
z4= -1.0

z5=np.sum(z4-YB[i])/len(test)
print("The error value is", z5*100)
```

REFERENCE:

[1] C. Radulescu, M. Dinsoreanu, R. Potolea, “Identification of spam comments using Natural Language Processing Techniques”, Technical University of Cluj-Napoca, Romania, 2014, pp. 29-35.

[2] S. P. Algur, A. P. Patil, P. S. Hiremath, S. Shivashankar, “Conceptual level Similarity Measure based Review Spam Detection”, Dept. of P.G. Studies and Research in Computer Science, Gulbarga University, Gulbarga, Karnataka, India, 2010, pp. 416-423.

[3] D. Liang, H. Shen, “Detecting spam review by combing reviewer feature and relationship”, 2014 International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS), pp. 102-107.

[4] A. Gupta, R. Kaushal,“Improving Spam Detection in Online Social Networks”, Department of Information Technology Indira Gandhi Delhi Technical University for Women Kashmere Gate, Delhi, pp. 1-6.

[5] H. A. Najada, X Zhu, “iSRD: Spam Review Detection with Imbalanced Data Distribution”, Dept. of Computer & Electrical Engineering and Computer Science, Florida Atlantic University Boca Raton 33431 FL, USA, pp. 553-560.
[6] X. Yang, “One Methodology for Spam Review Detection Based on Review Coherence Metrics”, 2015 International Conference on Intelligent Computing and Internet of Things (IC1T), pp. 99-102.
[7] R. Patel, P. Thakkar, “Opinion Spam Detection Using Feature Selection”, 2014 Sixth International Conference on Computational Intelligence and Communication Networks, pp. 560-564.
[8] M. L. Ramprasad, M. G. Amudha, “Spammer detection and tagging based user generated video search system –a Survey”, ICICES2014 - S.A.Engineering College, Chennai, Tamil Nadu, India, pp. 6-14.
[9] H. Li, Z. Chen, B. Liu, X. Wei, J. Shao, “Spotting Fake Reviews via Collective PositiveUnlabeled Learning”, 2014 IEEE International Conference on Data Mining, pp. 899-904.
[10] Y. Lin, H. Wu, J. Zhang, X. Wang, A. Zhou, “Towards Online Anti-Opinion Spam: Spotting Fake Reviews from the Review Sequence”, 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pp. 261- 264.