day-9-623

February 21, 2024

[1]:
```
# To check whether the mail is spam or ham mail
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

raw_mail_data = pd.read_csv("mail_data.csv")
raw_mail_data.head()
```

[1]:

	Category	Message
0	ham	Go until jurong point, crazy.. Available only …
1	ham	Ok lar… Joking wif u oni…
2	spam	Free entry in 2 a wkly comp to win FA Cup fina…
3	ham	U dun say so early hor… U c already then say…
4	ham	Nah I don't think he goes to usf, he lives aro…

[2]:
```
# To check and Nan filled in column
raw_mail_data.isna().sum()
```

[2]:
```
Category    0
Message     0
dtype: int64
```

[3]:
```
# To convert categorical field to numerical field
raw_mail_data['Category'].replace({'spam': 0, 'ham': 1},inplace = True)
raw_mail_data['Category']
```

[3]:
```
0       1
1       1
2       0
3       1
4       1
       ..
5567    0
5568    1
```

```
5569    1
5570    1
5571    1
Name: Category, Length: 5572, dtype: int64
```

[4]:
```
Y = raw_mail_data['Category']
X = raw_mail_data['Message']
X
```

[4]:
```
0       Go until jurong point, crazy.. Available only …
1                           Ok lar… Joking wif u oni…
2       Free entry in 2 a wkly comp to win FA Cup fina…
3       U dun say so early hor… U c already then say…
4       Nah I don't think he goes to usf, he lives aro…
                              …
5567    This is the 2nd time we have tried 2 contact u…
5568                 Will ü b going to esplanade fr home?
5569    Pity, * was in mood for that. So…any other s…
5570    The guy did some bitching but I acted like i'd…
5571                           Rofl. Its true to its name
Name: Message, Length: 5572, dtype: object
```

[5]:
```
# Convert to train and test split
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.2,␣
 ↪random_state=101)
x_train.shape
x_test.shape
y_train.shape
y_test.shape
```

[5]:
```
(1115,)
```

[6]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
```

[7]:
```
# Term Frequency Inverse Document Frequency
# It is a numerical representation of text document to capture the important␣
 ↪words in a collection of document.

feature_extraction = TfidfVectorizer()
x_train_feature = feature_extraction.fit_transform(x_train)
print(x_train_feature)
```

```
  (0, 4051)     0.396157711684648
  (0, 1218)     0.41535475634567975
  (0, 2818)     0.3749072721051035
  (0, 4767)     0.3119675917719436
  (0, 4837)     0.25509075047050056
```

```
(0, 4241)     0.3074030730595157
(0, 7387)     0.3637597414412867
(0, 7454)     0.25117871060511393
(0, 6863)     0.27871207078957255
(1, 2242)     0.2731600013202997
(1, 4546)     0.1797626942701284
(1, 5101)     0.21506873016833877
(1, 1593)     0.12127402897698365
(1, 6859)     0.17868082050185075
(1, 4461)     0.17514382420666957
(1, 7520)     0.2704784073749078
(1, 2172)     0.19735337010539045
(1, 3353)     0.24193633044383703
(1, 2188)     0.4333079074596793
(1, 3903)     0.1253617819600689
(1, 4873)     0.12248039075734285
(1, 3771)     0.1017977122975705
(1, 2195)     0.2731600013202997
(1, 5193)     0.1930341585252274
(1, 6336)     0.16594599980071684
:      :
(4453, 978)   0.13038053107248646
(4453, 4981)  0.14930203217779517
(4453, 7557)  0.1649992190976192
(4453, 7735)  0.09860253452103822
(4454, 2488)  0.3918446404234906
(4454, 1093)  0.4561498799631511
(4454, 7731)  0.3512725514873791
(4454, 7534)  0.2992212406303698
(4454, 5045)  0.3302004569135163
(4454, 3287)  0.35325691614792176
(4454, 2181)  0.2467213138355372
(4454, 6850)  0.15643754120122397
(4454, 3217)  0.24479147470432788
(4454, 3903)  0.21533587919848488
(4455, 3852)  0.537749125665614
(4455, 6104)  0.537749125665614
(4455, 2365)  0.44488109595659847
(4455, 1012)  0.45215351323911285
(4455, 7735)  0.13888760426229121
(4456, 2304)  0.5141795063447279
(4456, 4298)  0.5117584469570492
(4456, 7542)  0.4647300982182721
(4456, 7741)  0.28640686727058395
(4456, 1623)  0.2917871547948553
(4456, 4981)  0.3009651569703866
```

[8]:
```
x_test_feature = feature_extraction.transform(x_test)
print(x_test_feature)
```

```
  (0, 7454)     0.14375663059192606
  (0, 7437)     0.23049175807606267
  (0, 7433)     0.19319739895482224
  (0, 7248)     0.21205092807606885
  (0, 7122)     0.23771919255711657
  (0, 6891)     0.22223387476929624
  (0, 6323)     0.13592012780448376
  (0, 6113)     0.3719741430514692
  (0, 4241)     0.17593541232128582
  (0, 3802)     0.29066384377400534
  (0, 3783)     0.12197849102641574
  (0, 3655)     0.33262148717027007
  (0, 3436)     0.19079462603984626
  (0, 3062)     0.28477659918353654
  (0, 1419)     0.33262148717027007
  (0, 1350)     0.24281895578727183
  (0, 1296)     0.23049175807606267
  (1, 7741)     0.1015077001468157
  (1, 7493)     0.15496224153903065
  (1, 6976)     0.24405761207190935
  (1, 6765)     0.2600553793420984
  (1, 6525)     0.24405761207190935
  (1, 6332)     0.24405761207190935
  (1, 6239)     0.2383477617859644
  (1, 5917)     0.2039420019794958
  :     :
  (1112, 1456)  0.5779409557479678
  (1113, 7735)  0.13336100828685096
  (1113, 6336)  0.2887204539857919
  (1113, 3636)  0.6899776924068045
  (1113, 3617)  0.4923121701324622
  (1113, 2671)  0.3769444756355365
  (1113, 1623)  0.19577465190174592
  (1114, 7534)  0.18081029567036122
  (1114, 6863)  0.1477293702063055
  (1114, 5979)  0.27563749983919444
  (1114, 5725)  0.29370530168290404
  (1114, 5029)  0.23922006355845896
  (1114, 4841)  0.23449708446420803
  (1114, 4233)  0.2835300433614311
  (1114, 4105)  0.22166118852846142
  (1114, 3414)  0.2590135639128196
  (1114, 3391)  0.3213447866830497
  (1114, 2494)  0.19714470907984988
```

```
  (1114, 2307)  0.25112102039058287
  (1114, 1593)  0.1258779855332514
  (1114, 1379)  0.20036236298270874
  (1114, 1347)  0.21226331981483268
  (1114, 1298)  0.2637365430070705
  (1114, 951)   0.15049713502005194
  (1114, 907)   0.27563749983919444
```

[9]:
```
# Check the type of y_train and y_test
y_train.dtype
```

[9]:
```
dtype('int64')
```

[10]:
```
y_train = y_train.astype('int')
y_test = y_test.astype('int')
y_train.dtype
y_test.dtype
```

[10]:
```
dtype('int32')
```

[11]:
```
# Create instance for Logistic Regression
model = LogisticRegression()
model
```

[11]:
```
LogisticRegression()
```

[12]:
```
model.fit(x_train_feature, y_train)
```

[12]:
```
LogisticRegression()
```

[13]:
```
# Process of testing the model
prediction = model.predict(x_test_feature)
prediction
```

[13]:
```
array([1, 0, 1, …, 1, 1, 1])
```

[14]:
```
acc = accuracy_score(prediction, y_test)
print("Accuracy of the above model is =", acc)
```

```
Accuracy of the above model is = 0.9659192825112107
```

[15]:
```
# Real time verification
raw_mail_data['Message'][467]
```

[15]:
```
"They don't put that stuff on the roads to keep it from getting slippery over
there?"
```

[16]:
```
raw_mail_data['Category'][467]
```

[16]:
```
1
```

[17]:
```
input_mail = ["They don't put that stuff on the roads to keep it from getting␣
 ↪slippery over there?"]
```

[18]:
```
input_data_feature = feature_extraction.transform(input_mail)
print(input_data_feature)
```

```
  (0, 6970)     0.1155113743711798
  (0, 6874)     0.25874476624317705
  (0, 6867)     0.22904289265064137
  (0, 6850)     0.14105228485794397
  (0, 6847)     0.17897524361135556
  (0, 6580)     0.31331961904563294
  (0, 5850)     0.43824825263330897
  (0, 5553)     0.3263480439403955
  (0, 5072)     0.28872100799671646
  (0, 4981)     0.17975713282600092
  (0, 3937)     0.2801438185026489
  (0, 3783)     0.16856122294492457
  (0, 3141)     0.30424995422062356
  (0, 3035)     0.21157213644849557
  (0, 2411)     0.24225870597293092
```

[19]:
```
prediction = model.predict(input_data_feature)
if prediction[0]==1:
    print("It's Ham Mail")
else:
    print("It's Spam Mail")
```

```
It's Ham Mail
```

[20]:
```
pip install selenium
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: selenium in
c:\users\tanut\appdata\roaming\python\python311\site-packages (4.18.1)
Requirement already satisfied: urllib3[socks]<3,>=1.26 in
c:\programdata\anaconda3\lib\site-packages (from selenium) (1.26.16)
Requirement already satisfied: trio~=0.17 in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from selenium)
(0.24.0)
Requirement already satisfied: trio-websocket~=0.9 in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from selenium)
(0.11.1)
Requirement already satisfied: certifi>=2021.10.8 in
c:\programdata\anaconda3\lib\site-packages (from selenium) (2023.7.22)
Requirement already satisfied: typing_extensions>=4.9.0 in
```

```
c:\users\tanut\appdata\roaming\python\python311\site-packages (from selenium)
(4.9.0)
Requirement already satisfied: attrs>=20.1.0 in
c:\programdata\anaconda3\lib\site-packages (from trio~=0.17->selenium) (22.1.0)
Requirement already satisfied: sortedcontainers in
c:\programdata\anaconda3\lib\site-packages (from trio~=0.17->selenium) (2.4.0)
Requirement already satisfied: idna in c:\programdata\anaconda3\lib\site-
packages (from trio~=0.17->selenium) (3.4)
Requirement already satisfied: outcome in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from
trio~=0.17->selenium) (1.3.0.post0)
Requirement already satisfied: sniffio>=1.3.0 in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from
trio~=0.17->selenium) (1.3.0)
Requirement already satisfied: cffi>=1.14 in c:\programdata\anaconda3\lib\site-
packages (from trio~=0.17->selenium) (1.15.1)
Requirement already satisfied: wsproto>=0.14 in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from trio-
websocket~=0.9->selenium) (1.2.0)
Requirement already satisfied: PySocks!=1.5.7,<2.0,>=1.5.6 in
c:\programdata\anaconda3\lib\site-packages (from
urllib3[socks]<3,>=1.26->selenium) (1.7.1)
Requirement already satisfied: pycparser in c:\programdata\anaconda3\lib\site-
packages (from cffi>=1.14->trio~=0.17->selenium) (2.21)
Requirement already satisfied: h11<1,>=0.9.0 in
c:\users\tanut\appdata\roaming\python\python311\site-packages (from
wsproto>=0.14->trio-websocket~=0.9->selenium) (0.14.0)
Note: you may need to restart the kernel to use updated packages.
```

[21]:
```
pip install --upgrade urllib3==1.26.16
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: urllib3==1.26.16 in
c:\programdata\anaconda3\lib\site-packages (1.26.16)
Note: you may need to restart the kernel to use updated packages.
```

[24]:
```
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.common.by import By
s = Service(r"C:\Users\tanut\OneDrive\Desktop\chromedriver.exe")
webD = webdriver.Chrome(service=s)
# webD.get("https://books.toscrape.com/catalogue/category/books_1/page-10.html")
webD.get("https://books.toscrape.com/")
```

[ ]:
```
# #to retrive the elements from webpage
# #-->find element by name attribute
# ele=webD.find_element_by_name('element name')
```

```
# #-->find element by link
# ele=webD.find_element_by_link('link')

# #-->find element by tagname
# ele=webD.find_element_by_tag_name('link')

# #-->find element by xpath
# ele=webD.find_element_by_tag_xpath('xpath')

# #-->find element by classname
# ele=webD.find_element_by_class_name('classname')

# #-->find element by id
# ele=webD.find_element_by_id('id')
```

[ ]:
```
# to scrap the title
ele = webD.find_element(By.CLASS_NAME, "col-sm-8.h1")
print(ele)
```

[ ]:
```
#!To scrap the title
#col-sm-8-h1
#titl=webD.find_element_by_class_name("col-sm-8-h1")
# a=webD.find_element_by_tag_name('a')
a=webD.find_element(By.TAG_NAME,'a')
print(a.text)
```

[ ]:
```

```