

```
In [75]: #IMPORTING THE DEPENDENCIES

import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

In [76]: #LOADING THE DATA FROM CSV FILE TO PANDAS DATAFRAME
df = pd.read_csv('mail.csv')

In [77]: print(df)

      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...
...      ...
5567    spam  This is the 2nd time we have tried 2 contact u...
5568    ham      Will ù b going to esplanade fr home?
5569    ham  Pity, * was in mood for that. So...any other s...
5570    ham  The guy did some bitching but I acted like i'd...
5571    ham      Rofl. Its true to its name

[5572 rows x 2 columns]

In [78]: #REPLACING THE NULL VALUES WITH NULL STRING

data = df.where((pd.notnull(df)), '')

In [79]: #PRINTING THE FIRST 5 ROW OF THIS DATAFRAME

data.head(5)

Out[79]:      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...

In [80]: #PRINTING THE LAST 5 ROWS OF THE DATAFRAME

data.tail(5)

Out[80]:      Category      Message
5567    spam  This is the 2nd time we have tried 2 contact u...
5568    ham      Will ù b going to esplanade fr home?
5569    ham  Pity, * was in mood for that. So...any other s...
5570    ham  The guy did some bitching but I acted like i'd...
5571    ham      Rofl. Its true to its name

In [81]: #VIWING THE INFO ABOUT DATA
data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 2 columns):
 #   Column      Non-Null Count  Dtype
---  ---
 0   Category    5572 non-null   object
 1   Message     5572 non-null   object
dtypes: object(2)
memory usage: 87.2+ KB

In [82]: #CHECKING THE NUMBER OF ROWS AND COLUMNS OF THE DATASET

data.shape

Out[82]: (5572, 2)

In [83]: #LABLEING THE SPAM MAIL AS 0 AND HAM MAIL AS 1

data.loc[data['Category'] == 'spam', 'Cateogry',] = 0
data.loc[data['Category'] == 'ham', 'Cateogry',] = 1

In [84]: #SEPRATING THE DATA AS TEXTS AND LABELS

X= data['Message']

Y= data['Category']

In [85]: print(X)

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 ù 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

In [86]: print(Y)

0      ham
1      ham
2      spam
3      ham
4      ham
...
5567    spam
5568    ham
5569    ham
5570    ham
5571    ham
Name: Category, Length: 5572, dtype: object

In [87]: #SPLITING THE DATASET INTO TRAINING AND TESTING DATA

X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size= 0.2, random_state = 3)

In [88]: print(X.shape)
print(X_train.shape)
print(X_test.shape)

(5572,)
(4457,)
(1115,)

In [89]: print(Y.shape)
print(Y_train.shape)
print(Y_test.shape)

(5572,)
(4457,)
(1115,)

In [91]: #FEATURE EXTRACTION(TRANSFORMING THE TEXT DATA TO FEATURE VECTORS THAT CAN BE USED AS INPUT TO THE LOGISTIC REGRESSION)

feature_extraction = TfidfVectorizer(min_df = 1, stop_words = 'english', lowercase= 'True')

X_train_features = feature_extraction.fit_transform(X_train)
X_test_features = feature_extraction.transform(X_test)

Y_train = Y_train.astype('str')
Y_test = Y_test.astype('str')

In [92]: print(X_train)

3075      Don know. I did't msg him recently.
1787      Do you know why god created gap between your f...
1614      Thnx dude. u guys out 2nite?
4304      Yup i'm free...
3266      44 7732584351, Do you want a New Nokia 3510i c...
...
789      5 Free Top Polyphonic Tones call 087018728737,...
968      What do u want when i come back?.a beautiful n...
1667      Guess who spent all last night phasing in and ...
3321      Eh sorry leh... I din c ur msg. Not sad ahead...
1688      Free Top ringtone -sub to weekly ringtone-get ...
Name: Message, Length: 4457, dtype: object

In [93]: print(X_train_features)

(0, 5413)      0.6198254967574347
(0, 4456)      0.4168658090846482
(0, 2224)      0.413103377943378
(0, 3811)      0.34780165336891333
(0, 2329)      0.38783870336935383
(1, 4080)      0.18880584110891163
(1, 3185)      0.29694482957694585
(1, 3325)      0.31610586766078863
(1, 2957)      0.3398297002864083
(1, 2746)      0.3398297002864083
(1, 918)       0.22871581159877646
(1, 1839)      0.2784903590561455
(1, 2758)      0.3226407885943799
(1, 2956)      0.33036995955537024
(1, 1991)      0.33036995955537024
(1, 3046)      0.2503712792613518
(1, 3811)      0.17419952275504033
(2, 407)       0.509272536051008
(2, 3156)      0.4107239318312698
(2, 2404)      0.45287711070606745
(2, 6601)      0.6056811524587518
(3, 2870)      0.5864269879324768
(3, 7414)      0.8100020912469564
(4, 50)        0.23633754072626942
(4, 5497)      0.15743785051118356
:
:
(4454, 4602)   0.2669765732445391
(4454, 3142)   0.32014451677763156
(4455, 2247)   0.37052851863170466
(4455, 2469)   0.35441545511837946
(4455, 5646)   0.33545678464631296
(4455, 6810)   0.2973175715898277
(4455, 6091)   0.23103841516927642
(4455, 7113)   0.30536590342067704
(4455, 3872)   0.3108911491788658
(4455, 4715)   0.30714144758811196
(4455, 6916)   0.19636985317119715
(4455, 3922)   0.31287563163368587
(4455, 4456)   0.24920025316220423
(4456, 141)    0.292943737785358
(4456, 647)    0.30133182431707617
(4456, 6311)   0.30133182431707617
(4456, 5569)   0.4619395404299172
(4456, 6028)   0.21034888000987115
(4456, 7154)   0.24083218452280053
(4456, 7150)   0.3677554681447669
(4456, 6249)   0.17573831794959716
(4456, 6307)   0.2752760476857975
(4456, 334)    0.2220077711654938
(4456, 5778)   0.16243064490100795
(4456, 2870)   0.31523196273113385

In [94]: #LOGISTIC REGRESSION

model = LogisticRegression()

In [95]: #TRAINING THE LOGISTIC REGRESSION MODEL WITH TRAINING DATA

model.fit(X_train_features, Y_train)

Out[95]: LogisticRegression()

In [96]: #PREDICTION ON TRAINING DATA

prediction_on_training_data = model.predict(X_train_features)
accuracy_on_training_data = accuracy_score(Y_train, prediction_on_training_data)

In [97]: print('Acc on trainiing data : ', accuracy_on_training_data)

Acc on training data :  0.9670181736594121

In [98]: ##PREDICTION ON TEST DATA

prediction_on_test_data = model.predict(X_test_features)
accuracy_on_test_data = accuracy_score(Y_test, prediction_on_test_data)

In [99]: print('Acc on test data :', accuracy_on_test_data)

Acc on test data : 0.9659192825112107

In [100..: #BUILDING A PREDICTIVE SYSTEM

input_mail = ["Congratulations! Claim your free iPhone today"]

#CONVERT TEXT TO FEATURE VECTOR
input_data_features = feature_extraction.transform(input_mail)

#MAKING PREDICTION
prediction = model.predict(input_data_features)

print(prediction)

['spam']

In [ ]:
```