*#Importing necessary library*

import numpy as np import pandas as pd

import matplotlib.pyplot as plt

data = pd.read\_csv("emails.csv") data.head()

text spam

1. Subject: naturally irresistible your corporate... 1
2. Subject: the stock trading gunslinger fanny i... 1
3. Subject: unbelievable new homes made easy im ... 1
4. Subject: 4 color printing special request add... 1
5. Subject: do not have money , get software cds ... 1

data.shape (5728, 2)

data['text'][0]

"Subject: naturally irresistible your corporate identity lt is really hard to recollect a company : the market is full of suqgestions and the information isoverwhelminq ; but a good catchy logo , stylish statlonery and outstanding website will make the task much easier . we do not promise that havinq ordered a iogo your company will automaticaily become a world ieader : it isguite ciear that without good products , effective business organization and practicable aim it will be hotat nowadays market ; but we do promise that your marketing efforts will become much more effective . here is the list of clear benefits : creativeness : hand - made , original logos , specially done to reflect your distinctive company image . convenience : logo and stationery are provided in all formats ; easy - to - use content management system letsyou change your website content and even its structure . promptness : you will see logo drafts within three business days . affordability : your marketing break - through shouldn ' t make gaps in your budget . 100 % satisfaction

guaranteed : we provide unlimited amount of changes with no extra fees for you to be surethat you will love the result of this collaboration

. have a look at our portfolio \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_

\_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ not interested . . . \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_

\_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_" data['spam'].value\_counts()

0 4360

1 1368

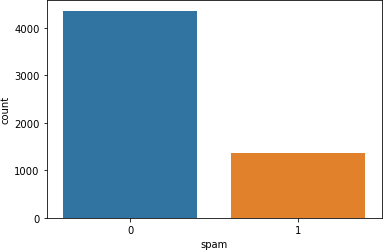
Name: spam, dtype: int64 import seaborn as sns

sns.countplot(data['spam'])

c:\python 3.7\lib\site-packages\seaborn\\_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

FutureWarning

<AxesSubplot:xlabel='spam', ylabel='count'>



data.duplicated().sum() 33

data.drop\_duplicates(inplace=True) data.duplicated().sum()

0

data.isnull().sum() text 0

spam 0 dtype: int64

data.shape (5695, 2)

5728 - 33

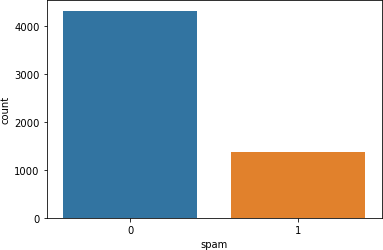
5695

sns.countplot(data['spam'])

c:\python 3.7\lib\site-packages\seaborn\\_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

FutureWarning

<AxesSubplot:xlabel='spam', ylabel='count'>



data['spam'].value\_counts()

0 4327

1 1368

Name: spam, dtype: int64

Separate in X and Y

X = data['text'].values y = data['spam'].values

y

array([1, 1, 1, ..., 0, 0, 0], dtype=int64)

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 =

0.2 , random\_state= 0)

X\_train.shape (4556,)

X\_test.shape (1139,)

y\_train.shape (4556,)

y\_test.shape

(1139,)

Preprocessing

from sklearn.feature\_extraction.text import CountVectorizer

cv = CountVectorizer()

x\_train = cv.fit\_transform(X\_train) x\_train.toarray()

array([[1, 0, 0, ..., 0, 0, 0],

[0, 0, 0, ..., 0, 0, 0],

[0, 0, 0, ..., 0, 0, 0],

...,

[0, 0, 0, ..., 0, 0, 0],

[0, 0, 0, ..., 0, 0, 0],

[0, 0, 0, ..., 0, 0, 0]], dtype=int64)

len(x\_train.toarray()) 4556

len(x\_train.toarray()[0])

33126

Training by ML Algorithm

from sklearn.naive\_bayes import MultinomialNB nb = MultinomialNB()

nb.fit(x\_train, y\_train) MultinomialNB()

x\_test = cv.transform(X\_test) len(x\_test.toarray())

1139

len(x\_test.toarray()[0]) 33126

y\_pred = nb.predict(x\_test)

from sklearn.metrics import accuracy\_score

print("Testing Accuracy:") accuracy\_score(y\_pred, y\_test)

Testing Accuracy: 0.990342405618964

print("Training Accuracy:") nb.score(x\_train,y\_train)

Training Accuracy: 0.995171202809482

Lets test using some emails

email = ['Hey, Jack whats up dude? Tomorrow please meet with me at my home.']

clean\_email = cv.transform(email) len(clean\_email.toarray()[0]) 33126

check = nb.predict(clean\_email)[0]

check

0

Evaluation Function

email = ['Hey i am Elon Musk. Get a brand new car from Tesla']

clean\_email = cv.transform(email) check = nb.predict(clean\_email)[0]

if check == 0:

print("This is a Ham Email!") else:

print("This is a Spam Email!") This is a Spam Email!