

FILE SEGREGATION

ML PROJECT

By – BHAVYA SEHGAL

INTRODUCTION

PURPOSE

OVERVIEW

EXAMPLES

APPLICATIONS



NAIVE BAYE'S

Bayes' Theorem is stated as:

$$P(h|d) = (P(d|h) * P(h)) / P(d)$$

Where

- $P(h|d)$ is the probability of hypothesis h given the data d . This is called the posterior probability.
- $P(d|h)$ is the probability of data d given that the hypothesis h was true.
- $P(h)$ is the probability of hypothesis h being true (regardless of the data). This is called the prior probability of h .
- $P(d)$ is the probability of the data (regardless of the hypothesis).



PROCESS STEPS



COLLECTION
OF DATA SET
FROM SK
LEARN



DEFINING
CATEGORIES

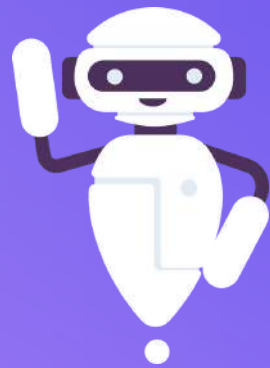


TESTING &
TRAINING
DATA

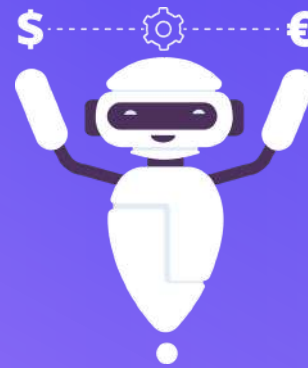


DISPLAY OF
RESULTS AND
HEAT MAP

CATEGORIES OF DATASETS



'alt.atheism',
'comp.graphics',
'comp.os.ms-
windows.misc',
'comp.sys.ibm.pc.ha
rdware',
'comp.sys.mac.hard
ware', 'misc.forsale',



'rec.autos',
'rec.motorcycles',
'rec.sport.baseball',
'rec.sport.hockey',
'sci.crypt',
'sci.electronics',
'sci.med',



'sci.space',
'soc.religion.christia
n', 'talk.politics.guns',
'talk.politics.mideast
' , 'talk.politics.misc',
'talk.religion.misc'
'comp.windows.x',

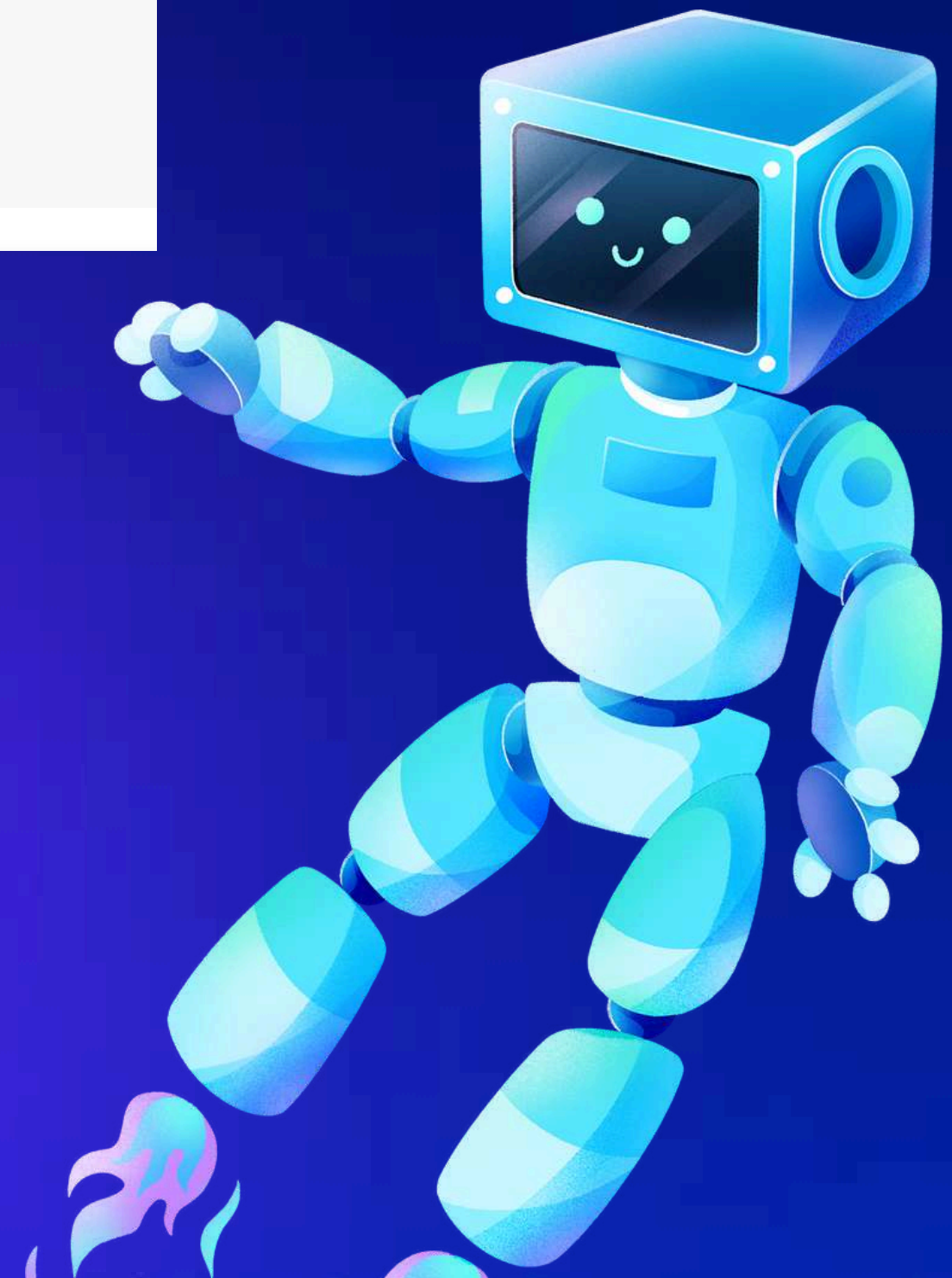
WALKAROUND OF THE CODE





```
%matplotlib inline  
import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns;sns.set()
```

IMPORTING MODULES AND PACKAGES





FETCHING DATASETS FROM SKLEARN



```
from sklearn.datasets import fetch_20newsgroups  
data = fetch_20newsgroups()  
target_names = data.target_names  
target_names
```



✓
0s



```
# defining all categories
categories = ['alt.atheism',
             'comp.graphics',
             'comp.os.ms-windows.misc',
             'comp.sys.ibm.pc.hardware',
             'comp.sys.mac.hardware',
             'comp.windows.x',
             'misc.forsale',
             'rec.autos',
             'rec.motorcycles',
             'rec.sport.baseball',
             'rec.sport.hockey',
             'sci.crypt',
             'sci.electronics',
             'sci.med',
             'sci.space',
             'soc.religion.christian',
             'talk.politics.guns',
             'talk.politics.mideast',
             'talk.politics.misc',
             'talk.religion.misc']

# training the data on these categories
train = fetch_20newsgroups(subset = 'train' , categories=categories)

# testing the data for these categories
test = fetch_20newsgroups(subset = 'test' , categories=categories)

# printing the training data
print(train.data[5])
```

DEFINING ALL CATEGORIES AND TRAINING AND TESTING DATA

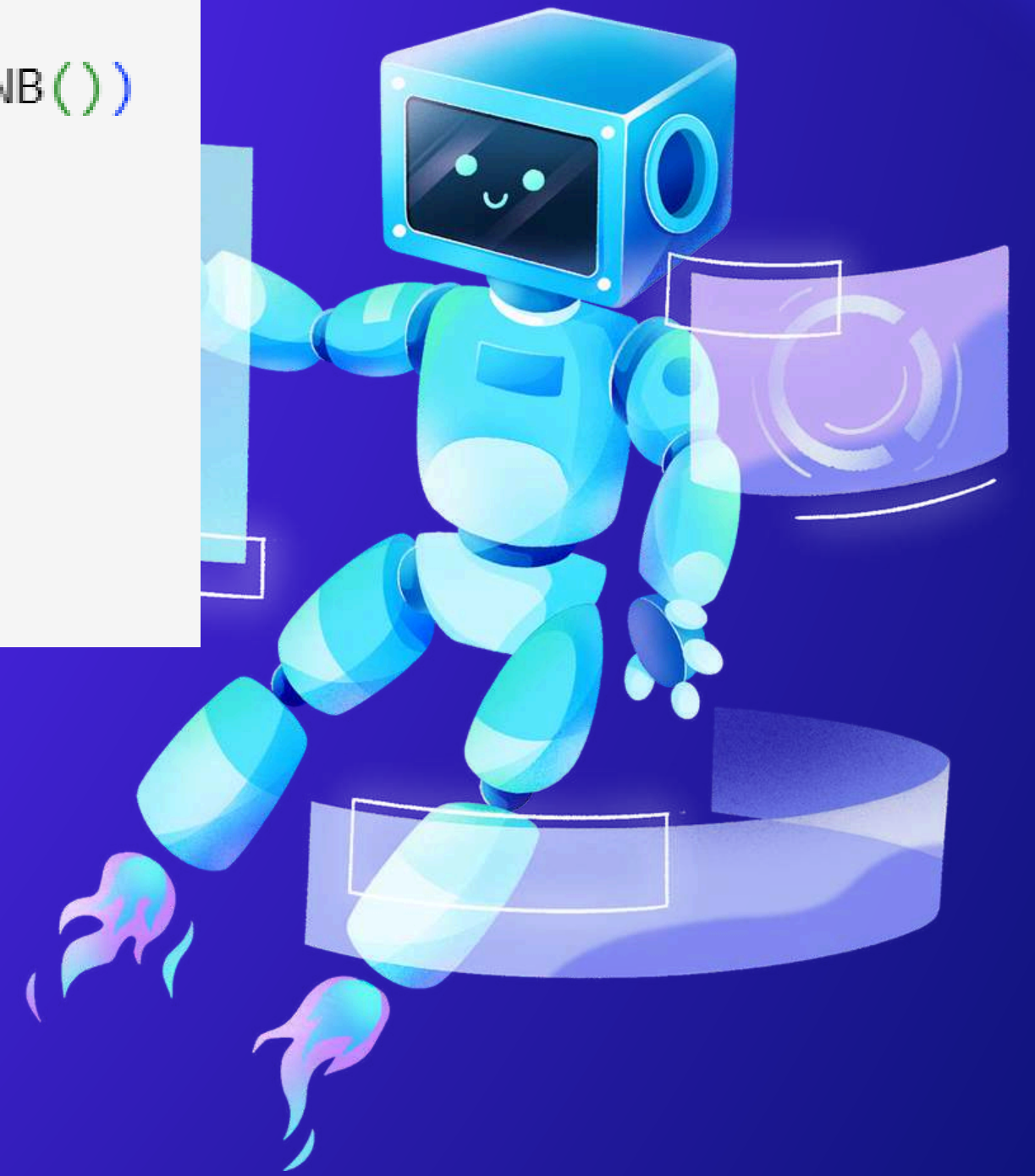


```
from sklearn.feature_extraction.text import TfidfVectorizer  
from sklearn.naive_bayes import MultinomialNB  
from sklearn.pipeline import make_pipeline
```

```
# creating a model based on multinomial bayes  
model = make_pipeline(TfidfVectorizer() , MultinomialNB())
```

```
# training the model with the training data  
model.fit(train.data , train.target)
```

```
# Creating labels for the test data  
labels = model.predict(test.data)
```



PREDICTING CATEGORIES


✓ [7] # predicting category on new data based on trained model

```
def predict_category( s, train=train , model = model):  
    pred = model.predict([s])  
    return train.target_names[pred[0]]
```

✓ 0s [8] predict_category('sending load to international space station')

'sci.space'

✓ 0s [12] file = open('/content/bhavya.txt.txt' , 'r')
ten = file.read()

✓ 0s  predict_category(ten)

'soc.religion.christian'

Text(109.44999999999996, 0.5, 'predicted label')

predicted label

| | | | | | | | | | | | | | | | | | | | |
|--------------------------|-----|-----|----|----|----|-----|----|----|----|----|---|----|----|----|----|----|---|----|----|
| alt.atheism | 661 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 2 | 0 | 0 | 2 | 33 |
| comp.graphics | 025 | 214 | 5 | 3 | 21 | 1 | 1 | 0 | 0 | 0 | 2 | 4 | 3 | 2 | 0 | 0 | 1 | 0 | 2 |
| comp.os.ms-windows.misc | 0 | 15 | 58 | 11 | 8 | 17 | 3 | 0 | 0 | 0 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| comp.sys.ibm.pc.hardware | 1 | 12 | 45 | 0 | 23 | 133 | 1 | 3 | 1 | 0 | 0 | 0 | 17 | 1 | 1 | 0 | 1 | 0 | 0 |
| comp.sys.mac.hardware | 0 | 9 | 3 | 17 | 9 | 82 | 12 | 0 | 0 | 1 | 0 | 0 | 5 | 1 | 0 | 0 | 0 | 0 | 0 |
| comp.windows.x | 1 | 18 | 9 | 1 | 0 | 29 | 81 | 0 | 0 | 0 | 0 | 1 | 0 | 3 | 3 | 0 | 0 | 0 | 0 |
| misc.forsale | 0 | 1 | 0 | 3 | 3 | 12 | 7 | 14 | 2 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | 2 | 0 | 0 |
| rec.autos | 0 | 2 | 2 | 6 | 8 | 0 | 19 | 64 | 10 | 4 | 1 | 3 | 8 | 0 | 2 | 0 | 0 | 0 | 1 |
| rec.motorcycles | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 3 | 7 | 0 | 0 | 0 | 7 | 2 | 1 | 1 | 1 | 0 | 0 |
| rec.sport.baseball | 1 | 5 | 3 | 0 | 3 | 1 | 4 | 2 | 0 | 35 | 4 | 0 | 1 | 3 | 0 | 0 | 1 | 1 | 0 |
| rec.sport.hockey | 1 | 2 | 2 | 2 | 1 | 0 | 6 | 2 | 0 | 22 | 8 | 0 | 2 | 4 | 1 | 0 | 0 | 0 | 1 |
| sci.crypt | 3 | 4 | 12 | 5 | 19 | 16 | 23 | 5 | 4 | 4 | 0 | 1 | 8 | 7 | 8 | 11 | 6 | 0 | 10 |
| sci.electronics | 0 | 4 | 1 | 13 | 8 | 0 | 12 | 1 | 0 | 0 | 0 | 12 | 3 | 55 | 1 | 0 | 0 | 0 | 0 |
| sci.med | 6 | 0 | 0 | 0 | 0 | 1 | 6 | 1 | 0 | 0 | 0 | 0 | 32 | 9 | 22 | 1 | 0 | 0 | 1 |
| sci.space | 3 | 6 | 6 | 5 | 2 | 4 | 3 | 3 | 0 | 2 | 1 | 0 | 11 | 6 | 35 | 2 | 1 | 0 | 7 |
| soc.religion.christian | 23 | 15 | 23 | 3 | 8 | 10 | 9 | 3 | 8 | 9 | 5 | 3 | 15 | 52 | 19 | 92 | 6 | 24 | 35 |
| talk.politics.guns | 4 | 4 | 2 | 1 | 3 | 2 | 3 | 4 | 2 | 1 | 0 | 1 | 2 | 6 | 4 | 0 | 4 | 3 | 11 |
| talk.politics.mideast | 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 4 | 0 | 0 | 1 | 4 | 5 |
| talk.politics.misc | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 11 | 29 |
| talk.religion.misc | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 35 |

alt.atheism
comp.graphics
comp.os.ms-windows.misc
comp.sys.ibm.pc.hardware
comp.sys.mac.hardware
comp.windows.x
misc.forsale
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
sci.crypt
sci.electronics
sci.med
sci.space
soc.religion.christian
talk.politics.guns
talk.politics.mideast
talk.politics.misc
talk.religion.misc

true label

HEAT MAP