

# multinomial\_naivebayes

November 20, 2019

## 1 1. Multinomial Naive Bayes Classifier

```
[1]: import numpy as np
import pandas as pd

w = pd.read_csv("C:/Users/kshitij/Desktop/Desktop/AML/CSV Files/bayes_mn.csv")

w
```

```
[1]:
```

	Class	Beijing	Chinese	Macau	Shanghai	Japan	Tokyo
0	c	1	2	0	0	0	0
1	c	0	2	0	1	0	0
2	c	0	1	1	0	0	0
3	j	0	1	0	0	1	1
4	j	0	3	0	0	1	1

```
[2]: X = w.iloc[:,1:]
y = w.iloc[:,0]

y
```

```
[2]: 0    c
1    c
2    c
3    j
4    j
Name: Class, dtype: object
```

```
[3]: X
```

```
[3]:
```

	Beijing	Chinese	Macau	Shanghai	Japan	Tokyo
0	1	2	0	0	0	0
1	0	2	0	1	0	0
2	0	1	1	0	0	0
3	0	1	0	0	1	1
4	0	3	0	0	1	1

```
[4]: #Multinomial naive bayes classifier
from sklearn.naive_bayes import MultinomialNB
```

```
clf=MultinomialNB()  
clf.fit(X,y)
```

[4]: MultinomialNB(alpha=1.0, class\_prior=None, fit\_prior=True)

```
[5]: y_pred = clf.predict(X)  
from sklearn.metrics import accuracy_score  
print("Accuracy Score : {}".format(accuracy_score(y,y_pred)))  
prediction = clf.predict(np.array([[1,2,3,4,0,1]]))  
prediction
```

Accuracy Score : 1.0

[5]: array(['c'], dtype='<U1')

## 2 Analysis

- 2.1 First we need to form a dataset containing documents with words and classes to which they belong. For a multinomial naive bayes classifier, data taken considers the frequency of a word occurring in a document. In order to do this we preprocess the dataset with a preprocessing tool like Weka. After preprocessing, make use of MultinomialNB() model on training data.