

10. Naive Bayes Classifier

October 17, 2018

Saumay Agrawal
16BCE1151

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

```
/anaconda3/lib/python3.6/importlib/_bootstrap.py:219: RuntimeWarning: numpy.dtype size changed
return f(*args, **kwargs)
/anaconda3/lib/python3.6/importlib/_bootstrap.py:219: RuntimeWarning: numpy.dtype size changed
return f(*args, **kwargs)
```

```
In [2]: # Finds the probabilities for all the target labels
```

```
def NBC(condition, target, data):

    n = df.shape[0]

    classes = df[target].unique()
    prob = {}
    for i in classes:
        p = df.loc[df[target]==i].shape[0]/n
        prob[i]= round(p, 3)

    for feature in condition.keys():
        for i in classes:
            df2 = df.loc[df[target]==i]
            df3 = df2.loc[df[feature]==condition[feature]]
            x = df3.shape[0]
            y = df2.shape[0]
            if x==0:
                x += 1
                y += 1
            p = x/y
            prob[i] *= p
            prob[i] = round(prob[i], 3)

    result = None
```

```

max_prob = 0

print('Probabilities:\n', prob)

for i in classes:
    if prob[i]>max_prob:
        result = i
        max_prob = prob[i]

print('\nCondition classified in:\n', result)

```

In [3]: # Reading data for part 1

```

df = pd.read_csv('nbc.csv')
print(df.head(14))

```

	Age	Income	Student	Credit_rating	Buys_computer
0	<=30	High	No	Fair	No
1	<=30	High	No	Excellent	No
2	31...40	High	No	Fair	Yes
3	>40	Medium	No	Fair	Yes
4	>40	Low	Yes	Fair	Yes
5	>40	Low	Yes	Excellent	No
6	31...40	Low	Yes	Excellent	Yes
7	<=30	Medium	No	Fair	No
8	<=30	Low	Yes	Fair	Yes
9	>40	Medium	Yes	Fair	Yes
10	<=30	Medium	Yes	Excellent	Yes
11	31...40	Medium	No	Excellent	Yes
12	31...40	High	Yes	Fair	Yes
13	>40	Medium	No	Excellent	No

In [4]: # Setting necessary conditions for classification

```

condition = {
    'Age': '<=30',
    'Income': 'Medium',
    'Student': 'Yes',
    'Credit_rating': 'Fair'
}

target = 'Buys_computer'

NBC(condition, target, df)

```

```

Probabilities:
{'No': 0.007, 'Yes': 0.029}

```

Condition classified in:
Yes

In [5]: # Reading data for part 2

```
df = pd.read_csv('data2.csv')
df.head(10)
```

```
Out[5]:
```

	TDP	Nifty	Sidhu	BJP	Sensex	Sixer	Congress	Century	Category
0	4	0	3	5	1	0	6	0	Politics
1	0	5	0	2	6	0	1	0	Business
2	0	0	6	1	0	4	1	2	Sports
3	4	1	0	1	1	0	6	0	Politics
4	0	0	0	0	0	5	0	6	Sports
5	0	4	0	2	6	0	0	1	Business
6	5	0	0	3	0	0	5	0	Politics

In [6]: # Modifying from continuous to categorical data

```
df = df.applymap(str)
for attr in df.columns[:-1]:
    df.loc[df[attr] > '3', attr] = 'high'
    df.loc[df[attr] <= '3', attr] = 'low'
df.head(10)
```

```
Out[6]:
```

	TDP	Nifty	Sidhu	BJP	Sensex	Sixer	Congress	Century	Category
0	high	low	low	high	low	low	high	low	Politics
1	low	high	low	low	high	low	low	low	Business
2	low	low	high	low	low	high	low	low	Sports
3	high	low	low	low	low	low	high	low	Politics
4	low	low	low	low	low	high	low	high	Sports
5	low	high	low	low	high	low	low	low	Business
6	high	low	low	low	low	low	high	low	Politics

In [7]: # Setting necessary conditions for classification

```
condition = {
    'TDP': 'low',
    'Nifty': 'low',
    'Sidhu': 'low',
    'BJP': 'low',
    'Sensex': 'high',
    'Sixer': 'low',
    'Congress': 'low',
    'Century': 'low',
}

target = 'Category'
```

```
NBC(condition, target, df)
```

Probabilities:

```
{'Politics': 0.004, 'Business': 0.095, 'Sports': 0.008}
```

Condition classified in:

```
Business
```