

NaiveBayes_ML_07

March 15, 2022

1 *Naive Bayes*

```
[ ]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
phool = pd.read_csv('iris.csv')
phool.head()
```

```
[ ]:      sepal.length  sepal.width  petal.length  petal.width  variety
0           5.1           3.5           1.4           0.2    Setosa
1           4.9           3.0           1.4           0.2    Setosa
2           4.7           3.2           1.3           0.2    Setosa
3           4.6           3.1           1.5           0.2    Setosa
4           5.0           3.6           1.4           0.2    Setosa
```

```
[ ]: X = phool.iloc[:, :-1]
y = phool.iloc[:, -1:]
```

```
[ ]: X.head()
```

```
[ ]:      sepal.length  sepal.width  petal.length  petal.width
0           5.1           3.5           1.4           0.2
1           4.9           3.0           1.4           0.2
2           4.7           3.2           1.3           0.2
3           4.6           3.1           1.5           0.2
4           5.0           3.6           1.4           0.2
```

```
[ ]: y.head()
```

```
[ ]:      variety
0    Setosa
1    Setosa
2    Setosa
3    Setosa
4    Setosa
```

```
[ ]: from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
model.fit(X,y)
```

C:\Users\Sartaj\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\utils\validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
y = column_or_1d(y, warn=True)

```
[ ]: GaussianNB()
```

```
[ ]: from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
X_train, X_test, y_train, y_test = train_test_split(X,y, train_size=0.2,
↳random_state=0)
model.fit(X_train,y_train)
y_predict = model.predict(X_test)
y_predict
```

C:\Users\Sartaj\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\utils\validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
y = column_or_1d(y, warn=True)

```
[ ]: array(['Virginica', 'Versicolor', 'Setosa', 'Virginica', 'Setosa',
'Virginica', 'Setosa', 'Versicolor', 'Versicolor', 'Versicolor',
'Versicolor', 'Versicolor', 'Versicolor', 'Versicolor',
'Versicolor', 'Setosa', 'Versicolor', 'Versicolor', 'Setosa',
'Setosa', 'Virginica', 'Versicolor', 'Setosa', 'Setosa',
'Versicolor', 'Setosa', 'Setosa', 'Versicolor', 'Versicolor',
'Setosa', 'Virginica', 'Versicolor', 'Setosa', 'Versicolor',
'Virginica', 'Versicolor', 'Setosa', 'Versicolor', 'Versicolor',
'Versicolor', 'Virginica', 'Setosa', 'Versicolor', 'Versicolor',
'Virginica', 'Versicolor', 'Versicolor', 'Versicolor',
'Virginica', 'Setosa', 'Setosa', 'Virginica', 'Versicolor',
'Setosa', 'Setosa', 'Versicolor', 'Setosa', 'Virginica',
'Versicolor', 'Setosa', 'Versicolor', 'Virginica', 'Versicolor',
'Setosa', 'Virginica', 'Virginica', 'Virginica', 'Virginica',
'Setosa', 'Setosa', 'Virginica', 'Virginica', 'Setosa',
'Virginica', 'Setosa', 'Virginica', 'Virginica', 'Setosa',
'Setosa', 'Virginica', 'Setosa', 'Setosa', 'Setosa', 'Versicolor',
'Virginica', 'Virginica', 'Setosa', 'Setosa', 'Setosa',
```

```
'Versicolor', 'Versicolor', 'Setosa', 'Setosa', 'Versicolor',
'Setosa', 'Virginica', 'Versicolor', 'Virginica', 'Versicolor',
'Setosa', 'Versicolor'], dtype='<U10')
```

```
[ ]: score = accuracy_score(y_test,y_predict)*100
print(format(score, ".2f"))
```

93.33

```
[ ]: from sklearn import metrics
cm = metrics.confusion_matrix(y_test,y_predict)
cm
```

```
[ ]: array([[40,  0,  0],
          [ 0, 39,  0],
          [ 0,  8, 33]], dtype=int64)
```

```
[ ]: plt.figure(figsize=(10,6))
sns.heatmap(cm,annot=True, fmt=".3f", linewidths=.5, square=True,
            cmap='Spectral')
plt.xlabel("Actual Labels")
plt.ylabel("Predicted labels")
all_sample = "Guassians Naive Bayes model accuracy:{0}".format(score, ".2f")
plt.title(all_sample, size=15)
```

```
[ ]: Text(0.5, 1.0, 'Guassians Naive Bayes model accuracy:93.33333333333333')
```

Guassians Naive Bayes model accuracy:93.33333333333333

