

Practical 8

Implementation of Naïve Bayes Algorithm on Jupyter Notebook using Python.

```
import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

from sklearn.datasets import load_iris

iris = load_iris()

dir(iris)
```

#Output : Fig 8.1

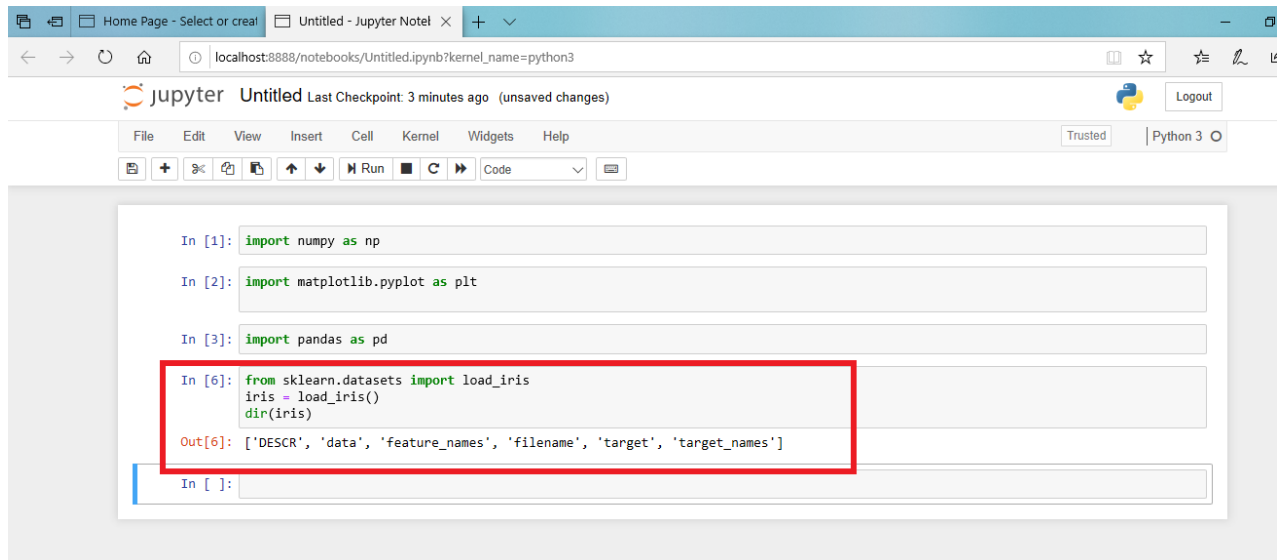
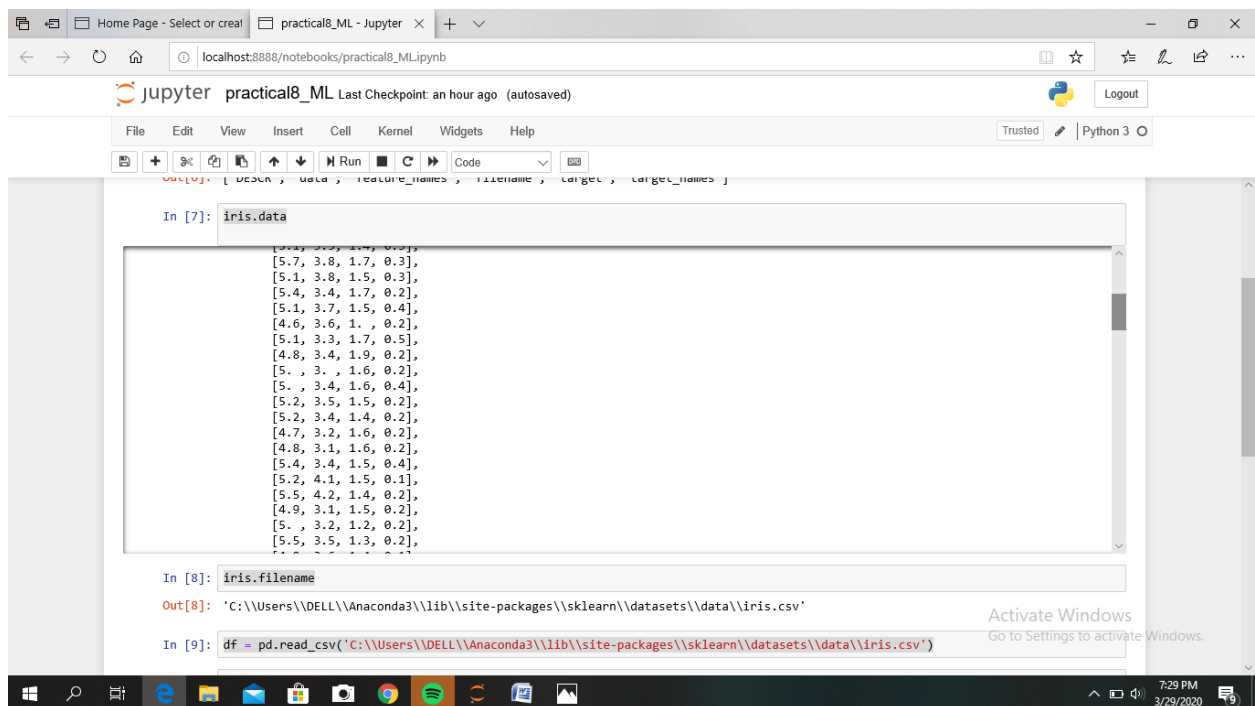
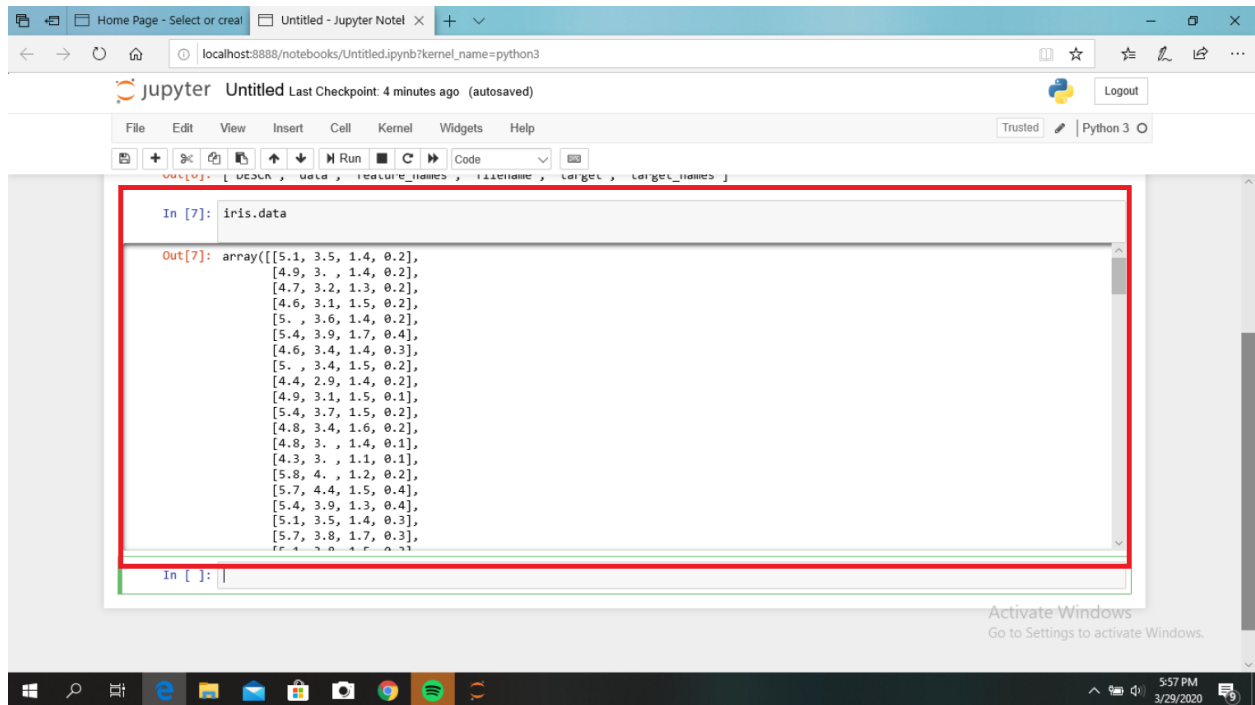


Fig8.1 dir(iris)

```
iris.data
```

#Output : Fig8.2



The screenshot shows a Jupyter Notebook interface with the following code and output:

```

In [7]: iris.data
Out[7]:
[[4.9, 2.4, 3.3, 1. ],
 [6.6, 2.9, 4.6, 1.3],
 [5.2, 2.7, 3.9, 1.4],
 [5. , 2. , 3.5, 1. ],
 [5.9, 3. , 4.2, 1.5],
 [6. , 2.2, 4. , 1. ],
 [6.1, 2.9, 4.7, 1.4],
 [5.6, 2.9, 3.6, 1.3],
 [6.7, 3.1, 4.4, 1.4],
 [5.6, 3. , 4.5, 1.5],
 [5.8, 2.7, 4.1, 1. ],
 [6.2, 2.2, 4.5, 1.5],
 [5.6, 2.5, 3.9, 1.1],
 [5.9, 3.2, 4.8, 1.8],
 [6.1, 2.8, 4. , 1.3],
 [6.3, 2.5, 4.9, 1.5],
 [6.1, 2.8, 4.7, 1.2],
 [6.4, 2.9, 4.3, 1.3],
 [6.6, 3. , 4.4, 1.4],
 [6.8, 2.8, 4.8, 1.4],
 ...

```

Below the first output, the following code is shown:

```

In [8]: iris.filename
Out[8]: 'C:\\Users\\DELL\\Anaconda3\\lib\\site-packages\\sklearn\\datasets\\data\\iris.csv'

In [9]: df = pd.read_csv('C:\\Users\\DELL\\Anaconda3\\lib\\site-packages\\sklearn\\datasets\\data\\iris.csv')

```

The screenshot shows the continuation of the Jupyter Notebook with the following code and output:

```

In [7]: iris.data
Out[7]:
[[7.9, 3.8, 6.4, 2. ],
 [6.4, 2.8, 5.6, 2.2],
 [6.3, 2.8, 5.1, 1.5],
 [6.1, 2.6, 5.6, 1.4],
 [7.7, 3. , 6.1, 2.3],
 [6.3, 3.4, 5.6, 2.4],
 [6.4, 3.1, 5.5, 1.8],
 [6. , 3. , 4.8, 1.8],
 [6.9, 3.1, 5.4, 2.1],
 [6.7, 3.1, 5.6, 2.4],
 [6.9, 3.1, 5.1, 2.3],
 [5.8, 2.7, 5.1, 1.9],
 [6.8, 3.2, 5.9, 2.3],
 [6.7, 3.3, 5.7, 2.5],
 [6.7, 3. , 5.2, 2.3],
 [6.3, 2.5, 5. , 1.9],
 [6.5, 3. , 5.2, 2. ],
 [6.2, 3.4, 5.4, 2.3],
 [5.9, 3. , 5.1, 1.8]]

```

The same code for In [8] and In [9] is repeated below.

Fig.8.2 iris.data

iris.filename

#Output: Fig.8.3

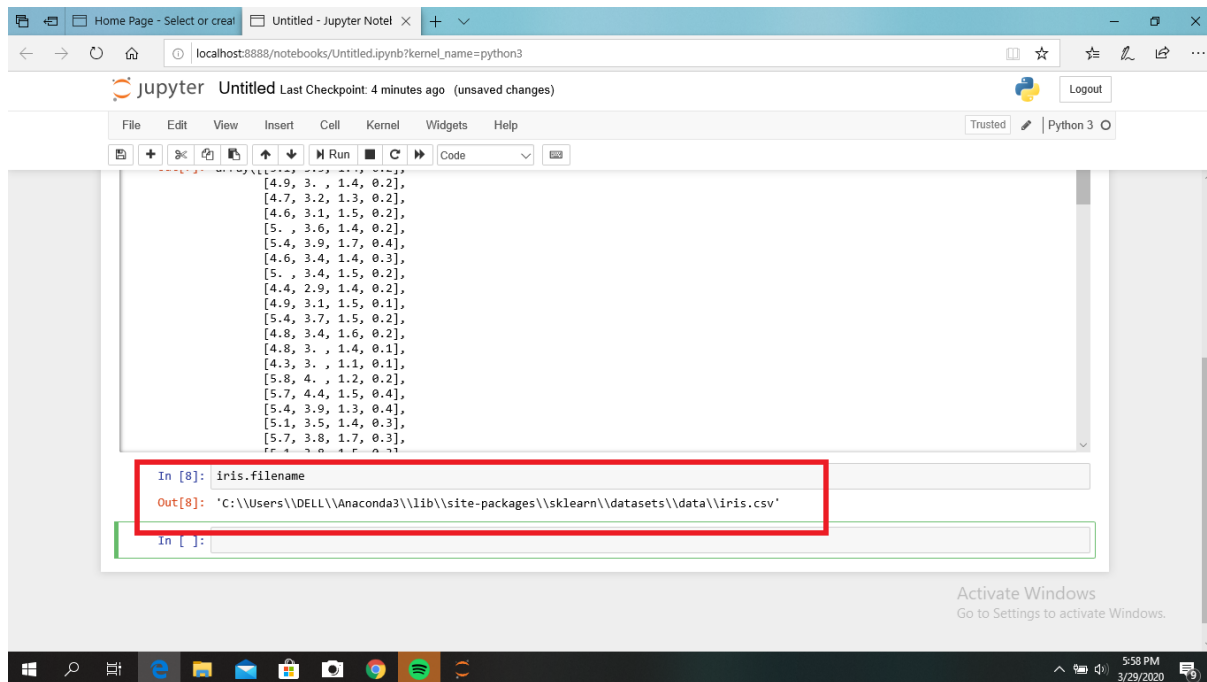


Fig. 8.3 iris.filename

```
df = pd.read_csv('C:\\Users\\DELL\\Anaconda3\\lib\\site-packages\\sklearn\\datasets\\data\\iris.csv')

from sklearn.datasets import load_iris

iris = load_iris()

x = iris.data

y = iris.target

from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.4,random_state = 1)

from sklearn.naive_bayes import GaussianNB

model = GaussianNB()

model.fit(x_train,y_train)
```

#Output : Fig.8.4

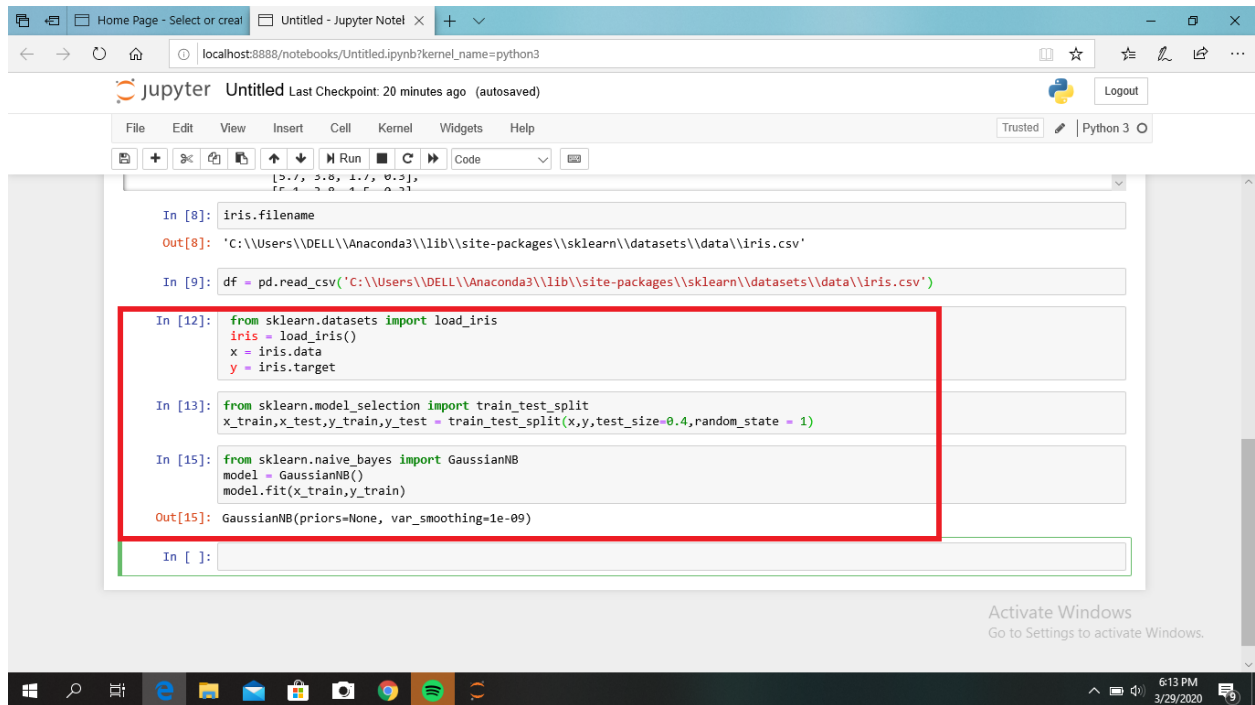


Fig.8.4 Model fitting

```
y_pred = model.predict(x_test)
```

```
from sklearn.metrics import accuracy_score
```

```
print(f'Guassian Naive Bayes model accuracy(in %) := {accuracy_score(y_test,y_pred)*100} %')
```

```
res = model.predict([[6.5,3.0,5.2,2.0]])
```

```
print (f'Result = {iris.target_names[res[0]]}')
```

Output :

The screenshot shows a Jupyter Notebook window titled 'practical8_ML'. The browser address bar shows 'localhost:8888/notebooks/practical8_ML.ipynb'. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and code execution. The code cells are as follows:

```
In [13]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.4,random_state = 1)

In [15]: from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
model.fit(x_train,y_train)

Out[15]: GaussianNB(priors=None, var_smoothing=1e-09)

In [16]: y_pred = model.predict(x_test)

In [18]: from sklearn.metrics import accuracy_score
print(f'Gaussian Naive Bayes model accuracy(in %) := {accuracy_score(y_test,y_pred)*100} %')
res = model.predict([[6.5,3.0,5.2,2.0]])
print (f'Result = {iris.target_names[res[0]]}')

Gaussian Naive Bayes model accuracy(in %) := 95.0 %
Result = virginica
```

The output of the final cell (In [18]) is displayed below the code, showing the model's accuracy and the predicted class name for a specific input. The text 'Gaussian Naive Bayes model accuracy(in %) := 95.0 %' and 'Result = virginica' is highlighted with a red box. The Windows taskbar at the bottom shows the time as 7:02 PM on 3/29/2020.

Fig.8.5 Result Output-Displaying target name as result