```python
In [33]: import numpy as np
         import pandas as pd
         import matplotlib.pyplot as plt
```

```python
In [34]: header_list = ("x1", "x2", "x3", "y")
         df = pd.read_csv("data.csv", names = header_list)
         df.head()
```

Out[34]:

	x1	x2	x3	y
0	700	1500	2600	1
1	550	1550	2400	1
2	700	1500	2600	1
3	700	1600	2700	1
4	550	1600	2600	1

```python
In [35]: df.tail()
```

Out[35]:

	x1	x2	x3	y
866	500	1050	2900	6
867	500	1000	3000	6
868	500	1000	2800	6
869	500	900	2800	6
870	500	950	2700	6

```python
In [36]: x=df.drop("y",axis=1);
         y=df['y'];
```

```python
In [37]: x.head()
```

Out[37]:

	x1	x2	x3
0	700	1500	2600
1	550	1550	2400
2	700	1500	2600
3	700	1600	2700
4	550	1600	2600

```python
In [38]: y.head()
```

```
Out[38]: 0    1
         1    1
         2    1
         3    1
         4    1
         Name: y, dtype: int64
```

```python
In [39]: from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2)
```

```python
In [40]: from sklearn.naive_bayes import GaussianNB
```

```python
In [41]: model = GaussianNB()
```

```python
In [42]: model.fit(x_train, y_train)
```

```
Out[42]: GaussianNB()
```

```python
In [43]: y_preds = model.predict(x_test)
```

```python
In [44]: model.score(x_test,y_test)
```

```
Out[44]: 0.8171428571428572
```

```python
In [45]: from sklearn.metrics import recall_score, f1_score, accuracy_score, precision_score
```

```python
In [46]: accuracy_score(y_test, y_preds)
```

```
Out[46]: 0.8171428571428572
```

```python
In [48]: recall_score(y_test, y_preds, average = "macro")
```

```
Out[48]: 0.7933637197569087
```

```python
In [51]: precision_score(y_test,y_preds, average = "macro")
```

```
Out[51]: 0.8429969153653364
```

```python
In [52]: from sklearn.metrics import confusion_matrix, plot_confusion_matrix
```

```python
In [53]: confusion_matrix(y_test, y_preds)
```

```
Out[53]: array([[ 7,  3,  0,  0,  5,  0],
                [ 0, 17,  0,  0,  0,  1],
                [ 0,  0, 33,  0,  5,  0],
                [ 0,  0,  0, 32,  0,  2],
                [ 0,  0,  5,  0, 24,  3],
                [ 0,  0,  0,  7,  1, 30]], dtype=int64)
```

```
In [55]: plot_confusion_matrix(model,x_test,y_test)
```

Out[55]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x1355ad7e400>