In [3]:

```
import numpy as np
```

In [9]:

```
import matplotlib.pyplot as plt
```

In [4]:

```
mu, sigma = 0, 1 # mean and standard deviation
```

In [5]:

```
X1=np.random.normal(mu, sigma, 100)
```

In [6]:

```
mu, sigma = 1, 1 # mean and standard deviation
```

In [7]:

```
X2 = np.random.normal(mu, sigma, 100)
```

In [10]:

```
plt.scatter(X1,X2, s=50)
```

Out[10]:

```
<matplotlib.collections.PathCollection at 0x1d50aaf76d8>
```

In [11]:

```
X1
```

Out[11]:

```
array([-0.442029  ,  0.51526306, -1.52512206,  0.01717074,  2.03050915,
        1.05366165,  0.68688603, -0.53496755,  0.78132981,  1.60468426,
       -0.13229154, -0.95596895,  0.16728181, -0.59431096, -0.85564853,
       -1.33918025, -1.07321027,  1.05898074,  0.81386053, -0.69219292,
       -0.74532086,  0.10392746, -0.18460106, -2.42007258,  0.8400526 ,
       -0.62798278, -1.00484156,  0.7945313 ,  0.92592336, -0.44801546,
       -1.37992822,  0.29032444, -0.13000293, -1.5751543 , -0.49528978,
       -0.20639441, -0.82054991, -0.02048624,  0.36125336,  1.68325412,
        0.95198224, -0.75611268,  0.38828091, -0.70725185, -0.73996954,
       -0.03633231,  0.66028096, -0.3799243 ,  0.21918721, -1.10873732,
       -1.04154024, -0.13287654,  0.08235272,  0.59034988,  0.01659933,
        0.14178655,  0.59781923,  1.07514802,  2.39361581,  0.47029999,
        2.05776824, -0.41084877,  2.11115171,  0.55405354, -0.70851851,
       -2.20866757,  0.65713795,  1.34333866, -1.01644294, -1.04469413,
        0.19724087,  0.14295933, -0.16063246,  0.15159122, -0.86839529,
        0.25458823,  0.38508419,  1.50140323, -0.48275925,  0.19389845,
        0.71131525,  0.2432477 ,  1.75868906,  0.21662399,  0.55484557,
       -1.36908676,  0.40036225, -0.00251392, -0.06098125, -0.39780534,
        1.70405847, -0.4744607 , -1.5998999 ,  0.92604539,  1.06050572,
       -0.95179071,  0.8677339 , -1.19623424,  1.78826868,  0.03969604])
```

In [12]:

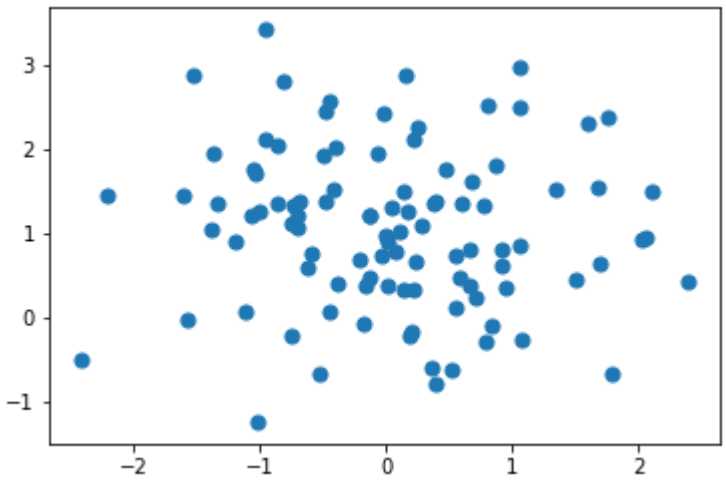
```
X2
```

Out[12]:

```
array([ 0.05816465, -0.61368729,  2.87645794,  0.38411024,  0.93607121,
        2.96204935,  1.61014103, -0.67129617,  1.33039328,  2.29917236,
        1.21122096,  3.42179699,  1.25367606,  0.76559338,  2.0476228 ,
        1.35291621,  1.20147596,  2.49923553,  2.50808804,  1.36634755,
       -0.21613582,  1.01728026, -0.08389593, -0.49598373, -0.08889899,
        0.5859071 ,  1.25160535, -0.28153964,  0.6228939 ,  2.56082756,
        1.03931596,  1.10064508,  1.20688914, -0.02107043,  1.92163062,
        0.69601856,  2.80377501,  2.42852916, -0.59951916,  1.54964655,
        0.35113967,  1.11389495, -0.7746565 ,  1.21659671,  1.31758635,
        0.72774693,  0.37620367,  0.40951937,  0.33490568,  0.08149154,
        1.71491549,  0.47245764,  0.78277993,  0.46770338,  0.9073791 ,
        1.48735795,  1.34484642, -0.25131762,  0.41640271,  1.75472391,
        0.94086233,  1.51654401,  1.50120595,  0.12748528,  1.06488744,
        1.45954301,  0.80754148,  1.51055267, -1.24265279,  1.75811566,
       -0.17921674,  0.3296375 ,  0.38555415,  2.87161942,  1.3435167 ,
        2.25391503,  1.3430142 ,  0.45530261,  1.38621775, -0.22085678,
        0.22897653,  0.67306257,  2.37210638,  2.11759353,  0.74346784,
        1.94372659,  1.36908055,  0.96086483,  1.95485379,  2.01537022,
        0.65052077,  2.43526228,  1.44075757,  0.80133716,  0.86499685,
        2.10953468,  1.80717181,  0.90184987, -0.67129379,  1.3138181 ])
```

In [13]:

```
plt.show()
```



In [14]:

```
C1=np.vstack((X1, X2)).T
```

In [15]:

```
y_C1 = np.zeros(C1.shape[0])
```

In [16]:

```
mu, sigma = 0, 1 # mean and standard deviation
```

In [17]:

```
X1=np.random.normal(mu, sigma, 100)
```

In [18]:

```
mu, sigma = -2, 1 # mean and standard deviation
```

In [19]:

```
X2 = np.random.normal(mu, sigma, 100)
```

In [20]:

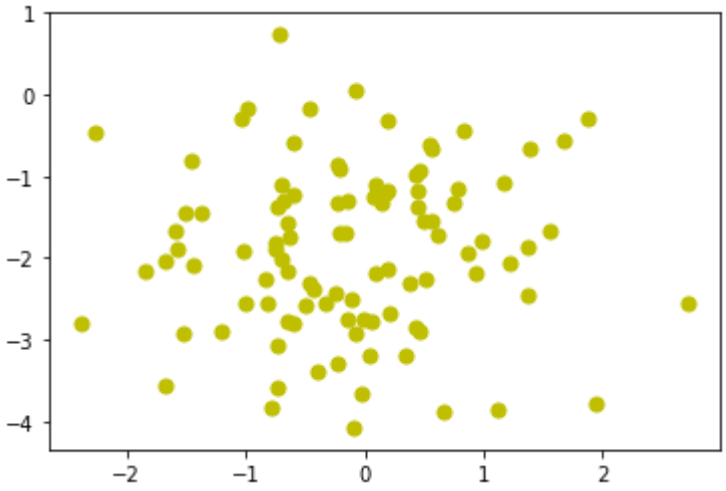
```
plt.scatter(X1,X2, s=50, c='y')
```

Out[20]:

```
<matplotlib.collections.PathCollection at 0x1d50abd4c50>
```

In [21]:

```
plt.show()
```



In [25]:

```
C2=np.vstack((X1, X2)).T
```

In [26]:

```
y_C2 = np.ones(C2.shape[0])
```

In [27]:

```
from sklearn.model_selection import train_test_split
```

In [28]:

```
X_train, X_test, y_train, y_test = train_test_split(C1, y_C1, test_size=0.30)
```

In [29]:

```
from sklearn.naive_bayes import GaussianNB
```

In [30]:

```
GNB = GaussianNB()
```

In [31]:

```
GNB.fit(X_train,y_train)
```

Out[31]:

```
GaussianNB(priors=None)
```

In [32]:

```
predict = GNB.predict(X_test)
```

In [33]:

```
from sklearn.metrics import accuracy_score
print('Accuracy: %.2f' %accuracy_score(y_test, predict))
```

```
Accuracy: 1.00
```

In [34]:

```
X_train, X_test, y_train, y_test = train_test_split(C2, y_C2, test_size=0.30)
```

In [35]:

```
GNB.fit(X_train,y_train)
```

Out[35]:

```
GaussianNB(priors=None)
```

In [36]:

```
predict = GNB.predict(X_test)
```

In [37]:

```
print('Accuracy: %.2f' %accuracy_score(y_test, predict))
```

```
Accuracy: 1.00
```

In [39]:

```
from sklearn.linear_model import LogisticRegression
```