In [29]:

```
from sklearn.datasets import load_iris
D_iris = load_iris()['data']
D_iris.shape
```

Out[29]:

```
(150, 4)
```

In [35]:

```
import numpy as np

d = np.array([[1,1.5],[1,2],[3,4],[-1,-1],[-1,1],[1,-2],[2,2],[2,3]])
d.shape
```

Out[35]:

```
(8, 2)
```

In [ ]:

```
######### QUESTION 2 ###########
```

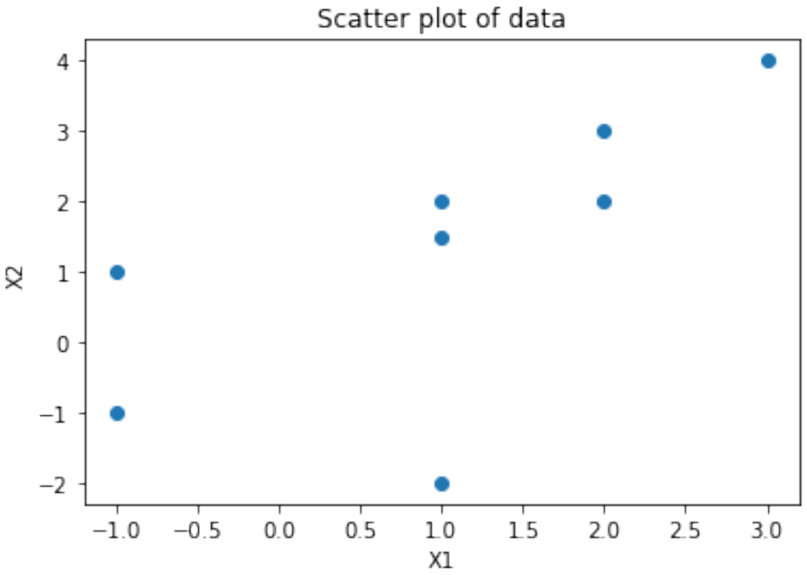
In [31]:

```
import matplotlib.pyplot as plt
plt.scatter(d[:,0], d[:,1])
plt.xlabel('X1')
plt.ylabel('X2')
plt.title('Scatter plot of data')

######### A ###########
```

Out[31]:

```
Text(0.5, 1.0, 'Scatter plot of data')
```



In [ ]:

```
######### B ###########
```

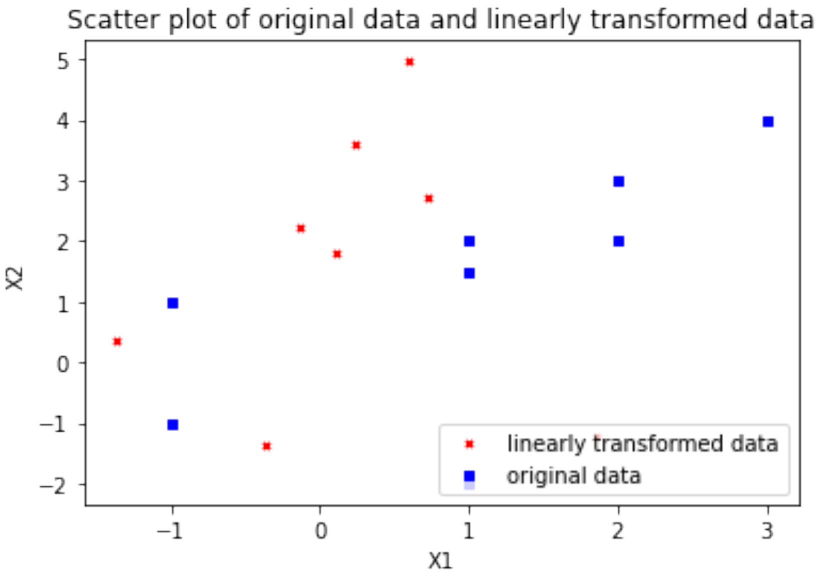
In [49]:
```
import math
m = np.array([[math.sqrt(3)/2,-1/2],[1/2,math.sqrt(3)/2]])
linearly_transformed_data = m.dot(d.T)
```

Out[49]:
```
array([[ 0.8660254, -0.5      ],
       [ 0.5      ,  0.8660254]])
```

In [71]:
```
fig = plt.figure()
ax = fig.add_subplot(111)
ax.scatter(linearly_transformed_data[0,:], linearly_transformed_data[1,:], s=
ax.scatter(d[:,0], d[:,1], s=10, c='b', marker='s', label='original data')
plt.xlabel('X1')
plt.ylabel('X2')
plt.legend(loc='lower right')
plt.title('Scatter plot of original data and linearly transformed data')

######### C ###########
```

Out[71]:
```
Text(0.5, 1.0, 'Scatter plot of original data and linearly transformed data')
```



In [65]:
```
######### D ###########
multi_d_mean = np.mean(d, axis=0)
multi_d_mean
```

Out[65]:
```
array([1.    , 1.3125])
```

In [66]:
```
######### E ###########
centered_data = d - multi_d_mean
centered_data
```

Out[66]:

```
array([[ 0.    ,  0.1875],
       [ 0.    ,  0.6875],
       [ 2.    ,  2.6875],
       [-2.    , -2.3125],
       [-2.    , -0.3125],
       [ 0.    , -3.3125],
       [ 1.    ,  0.6875],
       [ 1.    ,  1.6875]])
```

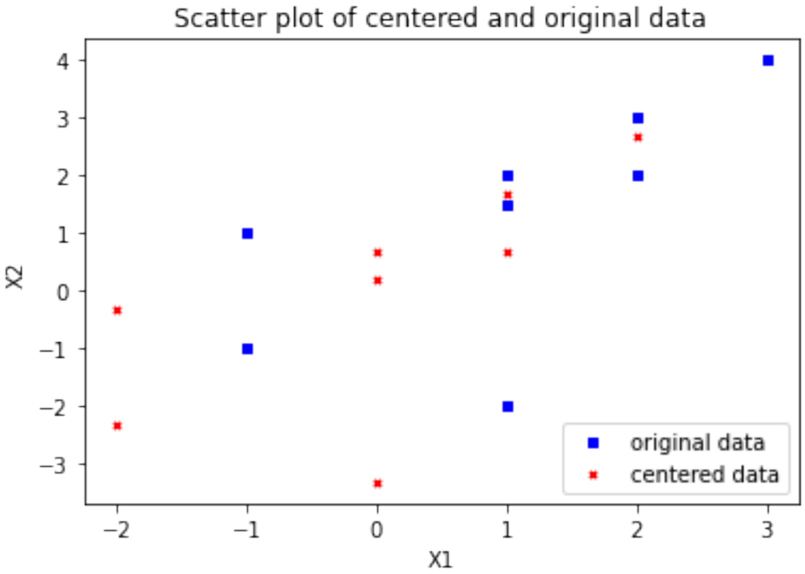
In [72]:

```
fig = plt.figure()
ax = fig.add_subplot(111)
ax.scatter(d[:,0], d[:,1], s=10, c='b', marker='s', label='original data')
ax.scatter(centered_data[:,0], centered_data[:,1], s=10, c='r', marker='x', l
plt.xlabel('X1')
plt.ylabel('X2')
plt.legend(loc='lower right')
plt.title('Scatter plot of centered and original data')

######### F ###########
```

Out[72]:

```
Text(0.5, 1.0, 'Scatter plot of centered and original data')
```



In [73]:

```
######### G ###########
np.cov(d.T, ddof=1)
```

Out[73]:

```
array([[2.        , 1.85714286],
       [1.85714286, 3.92410714]])
```

In [83]:

```
######### H ###########
np.cov(centered_data.T, ddof=1)
```

Out[83]:

```
array([[2.        , 1.85714286],
       [1.85714286, 3.92410714]])
```

In [82]:

```
######### I ###########
from sklearn.feature_selection import VarianceThreshold

from sklearn.preprocessing import MinMaxScaler
d_normalized = MinMaxScaler().fit_transform(d)
np.cov(d_normalized.T, ddof=1)
threshold = .06
feature_selector = VarianceThreshold(threshold=threshold)
feature_selector.fit_transform(d_normalized)
```

Out[82]:

```
array([[0.5       , 0.58333333],
       [0.5       , 0.66666667],
       [1.        , 1.        ],
       [0.        , 0.16666667],
       [0.        , 0.5       ],
       [0.5       , 0.        ],
       [0.75      , 0.66666667],
       [0.75      , 0.83333333]])
```

In [ ]: