In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

In [22]:
```
from sklearn.datasets import load_boston
```

In [23]:
```
boston = load_boston()
```

In [24]:
```
boston.keys()
```

Out[24]:
```
dict_keys(['data', 'target', 'feature_names', 'DESCR', 'filename'])
```

```
Proceedings on the Tenth International Conference of Machine Learning, 236-24
3, University of Massachusetts, Amherst. Morgan Kaufmann.
```

In [26]:
```
df=pd.DataFrame(boston['data'],columns=boston['feature_names'])
```

In [27]:
```
df.head(10)
```

Out[27]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LST
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5
5	0.02985	0.0	2.18	0.0	0.458	6.430	58.7	6.0622	3.0	222.0	18.7	394.12	5
6	0.08829	12.5	7.87	0.0	0.524	6.012	66.6	5.5605	5.0	311.0	15.2	395.60	12
7	0.14455	12.5	7.87	0.0	0.524	6.172	96.1	5.9505	5.0	311.0	15.2	396.90	19
8	0.21124	12.5	7.87	0.0	0.524	5.631	100.0	6.0821	5.0	311.0	15.2	386.63	29
9	0.17004	12.5	7.87	0.0	0.524	6.004	85.9	6.5921	5.0	311.0	15.2	386.71	17

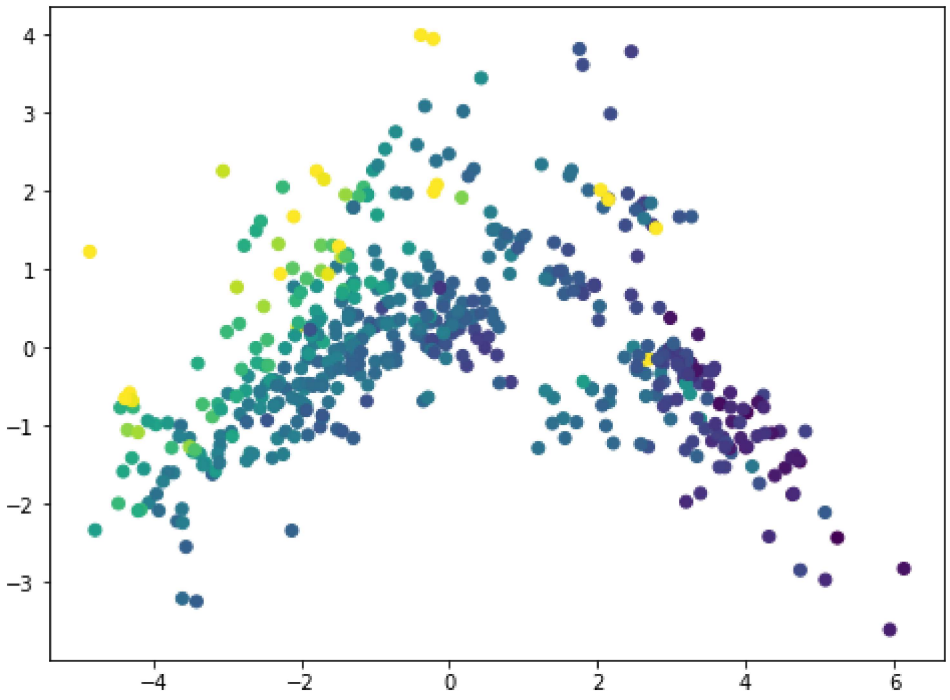
In [28]:
```
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()
scaler.fit(df)
StandardScaler()
scaled_data=scaler.transform(df)
from sklearn.decomposition import PCA
pca=PCA(n_components=2)
pca.fit(scaled_data)
PCA(n_components=2)
n_pca=pca.transform(scaled_data)
scaled_data.shape
scaled_data
```

Out[28]:
```
array([[-0.41978194,  0.28482986, -1.2879095 , ..., -1.45900038,
         0.44105193, -1.0755623 ],
       [-0.41733926, -0.48772236, -0.59338101, ..., -0.30309415,
         0.44105193, -0.49243937],
       [-0.41734159, -0.48772236, -0.59338101, ..., -0.30309415,
         0.39642699, -1.2087274 ],
       ...,
       [-0.41344658, -0.48772236,  0.11573841, ...,  1.17646583,
         0.44105193, -0.98304761],
       [-0.40776407, -0.48772236,  0.11573841, ...,  1.17646583,
         0.4032249 , -0.86530163],
       [-0.41500016, -0.48772236,  0.11573841, ...,  1.17646583,
         0.44105193, -0.66905833]])
```

In [29]:
```
n_pca
```

Out[29]:
```
array([[-2.09829747,  0.77311267],
       [-1.45725167,  0.59198522],
       [-2.07459756,  0.5996394 ],
       ...,
       [-0.31236047,  1.15524643],
       [-0.27051907,  1.04136157],
       [-0.12580322,  0.76197804]])
```

In [31]:
```
plt.figure(figsize=(8,6))
plt.scatter(n_pca[:,0],n_pca[:,1],c=boston['target'])
plt.xlabel='first principle component'
plt.ylabel='second principle component'
```



In [ ]:
```

```