objective: use principal component analysis

In [1]:
```
#step1 : import necessary library
import pandas as pd
import numpy as ny
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import datasets
```

In [2]:
```
#step2: load dataset
cancer_data=datasets.load_breast_cancer()
```

In [3]:
```
# step3: convert data into dataframe
cancer_df=pd.DataFrame(cancer_data.data,columns=cancer_data.feature_names)
```

In [4]:
```
cancer_df.head()
```

Out[4]:

| an ıre | mean perimeter | mean area | mean smoothness | mean compactness | mean concavity | mean concave points | mean symmetry | mean fractal dimension | ... | v ra |
|---|---|---|---|---|---|---|---|---|---|---|
| 38 | 122.80 | 1001.0 | 0.11840 | 0.27760 | 0.3001 | 0.14710 | 0.2419 | 0.07871 | ... | 2 |
| 77 | 132.90 | 1326.0 | 0.08474 | 0.07864 | 0.0869 | 0.07017 | 0.1812 | 0.05667 | ... | 2 |
| 25 | 130.00 | 1203.0 | 0.10960 | 0.15990 | 0.1974 | 0.12790 | 0.2069 | 0.05999 | ... | 2 |
| 38 | 77.58 | 386.1 | 0.14250 | 0.28390 | 0.2414 | 0.10520 | 0.2597 | 0.09744 | ... | 1 |
| 34 | 135.10 | 1297.0 | 0.10030 | 0.13280 | 0.1980 | 0.10430 | 0.1809 | 0.05883 | ... | 2 |

ımns

In [5]:
```
cancer_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 30 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   mean radius              569 non-null    float64
 1   mean texture             569 non-null    float64
 2   mean perimeter           569 non-null    float64
 3   mean area                569 non-null    float64
 4   mean smoothness          569 non-null    float64
 5   mean compactness         569 non-null    float64
 6   mean concavity           569 non-null    float64
 7   mean concave points      569 non-null    float64
 8   mean symmetry            569 non-null    float64
 9   mean fractal dimension   569 non-null    float64
 10  radius error             569 non-null    float64
 11  texture error            569 non-null    float64
 12  perimeter error          569 non-null    float64
 13  area error               569 non-null    float64
 14  smoothness error         569 non-null    float64
 15  compactness error        569 non-null    float64
 16  concavity error          569 non-null    float64
 17  concave points error     569 non-null    float64
 18  symmetry error           569 non-null    float64
 19  fractal dimension error  569 non-null    float64
 20  worst radius             569 non-null    float64
 21  worst texture            569 non-null    float64
 22  worst perimeter          569 non-null    float64
 23  worst area               569 non-null    float64
 24  worst smoothness         569 non-null    float64
 25  worst compactness        569 non-null    float64
 26  worst concavity          569 non-null    float64
 27  worst concave points     569 non-null    float64
 28  worst symmetry           569 non-null    float64
 29  worst fractal dimension  569 non-null    float64
dtypes: float64(30)
memory usage: 133.5 KB
```

In [6]:
```
# perform statistical analysis
cancer_df.describe()
```

Out[6]:

| | mean radius | mean texture | mean perimeter | mean area | mean smoothness | mean compactness | mean concavity |
|---|---|---|---|---|---|---|---|
| **count** | 569.000000 | 569.000000 | 569.000000 | 569.000000 | 569.000000 | 569.000000 | 569.000000 |
| **mean** | 14.127292 | 19.289649 | 91.969033 | 654.889104 | 0.096360 | 0.104341 | 0.088799 |
| **std** | 3.524049 | 4.301036 | 24.298981 | 351.914129 | 0.014064 | 0.052813 | 0.079720 |
| **min** | 6.981000 | 9.710000 | 43.790000 | 143.500000 | 0.052630 | 0.019380 | 0.000000 |
| **25%** | 11.700000 | 16.170000 | 75.170000 | 420.300000 | 0.086370 | 0.064920 | 0.029560 |
| **50%** | 13.370000 | 18.840000 | 86.240000 | 551.100000 | 0.095870 | 0.092630 | 0.061540 |
| **75%** | 15.780000 | 21.800000 | 104.100000 | 782.700000 | 0.105300 | 0.130400 | 0.130700 |
| **max** | 28.110000 | 39.280000 | 188.500000 | 2501.000000 | 0.163400 | 0.345400 | 0.426800 |

8 rows × 30 columns

In [7]:
```
# standardize the data
from sklearn.preprocessing import StandardScaler
ss=StandardScaler()
ss.fit(cancer_df)
```

Out[7]:
```
StandardScaler()
```

In [8]:
```
data_transform=ss.transform(cancer_df)
```

In [10]:
```
data_transform[0]
```

Out[10]:
```
array([ 1.09706398, -2.07333501,  1.26993369,  0.9843749 ,  1.56846633,
        3.28351467,  2.65287398,  2.53247522,  2.21751501,  2.25574689,
        2.48973393, -0.56526506,  2.83303087,  2.48757756, -0.21400165,
        1.31686157,  0.72402616,  0.66081994,  1.14875667,  0.90708308,
        1.88668963, -1.35929347,  2.30360062,  2.00123749,  1.30768627,
        2.61666502,  2.10952635,  2.29607613,  2.75062224,  1.93701461])
```

In [11]:
```
cancer_df.head()
```

Out[11]:

| | mean radius | mean texture | mean perimeter | mean area | mean smoothness | mean compactness | mean concavity | mean concave points | mean symmetry |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 17.99 | 10.38 | 122.80 | 1001.0 | 0.11840 | 0.27760 | 0.3001 | 0.14710 | 0.2419 |
| **1** | 20.57 | 17.77 | 132.90 | 1326.0 | 0.08474 | 0.07864 | 0.0869 | 0.07017 | 0.1812 |
| **2** | 19.69 | 21.25 | 130.00 | 1203.0 | 0.10960 | 0.15990 | 0.1974 | 0.12790 | 0.2069 |
| **3** | 11.42 | 20.38 | 77.58 | 386.1 | 0.14250 | 0.28390 | 0.2414 | 0.10520 | 0.2597 |
| **4** | 20.29 | 14.34 | 135.10 | 1297.0 | 0.10030 | 0.13280 | 0.1980 | 0.10430 | 0.1809 |

5 rows × 30 columns

In [13]:
```
data_transform.shape
```

Out[13]:
```
(569, 30)
```

In [14]:
```
#create principal components
from sklearn.decomposition import PCA

#create an instance of PCA
pca=PCA(n_components=2)

# apply
x_pca=pca.fit_transform(data_transform)
```

In [15]:
```
# result is x_pca
x_pca.shape
```

Out[15]:
```
(569, 2)
```

In [18]:
```
# visualize data - create a dataframe
pca_df=pd.DataFrame(x_pca,columns=['PC1','PC2'])
```

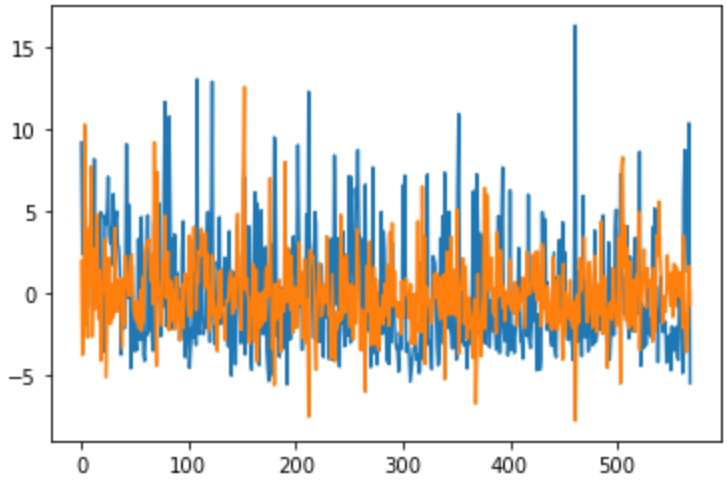
In [19]:
```
pca_df.head()
```

Out[19]:

| | PC1 | PC2 |
|---|---|---|
| **0** | 9.192837 | 1.948583 |
| **1** | 2.387802 | -3.768172 |
| **2** | 5.733896 | -1.075174 |
| **3** | 7.122953 | 10.275589 |
| **4** | 3.935302 | -1.948072 |

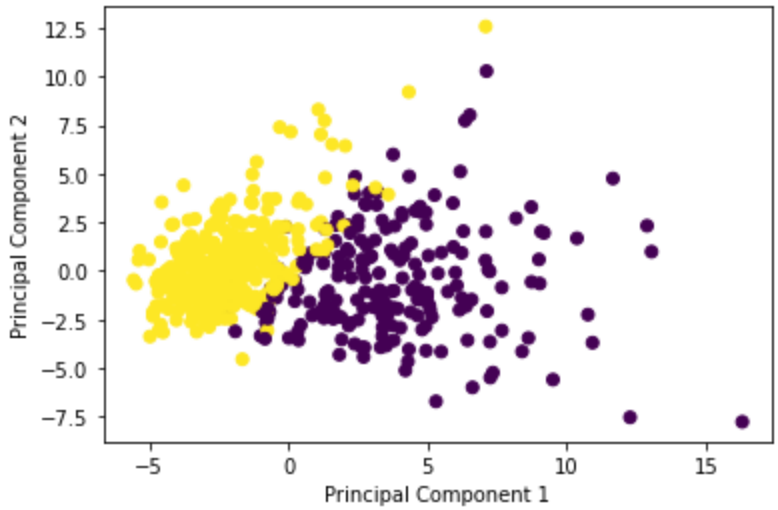
In [22]:
```
# draw a scatter plot of principal component
plt.plot(pca_df)
```

Out[22]:
```
[<matplotlib.lines.Line2D at 0x1ccbd74b0a0>,
 <matplotlib.lines.Line2D at 0x1ccbd74b0d0>]
```




In [27]:
```
plt.scatter(x='PC1',y='PC2',data=pca_df,c=cancer_data.target)
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
```

Out[27]:
```
Text(0, 0.5, 'Principal Component 2')
```




In [ ]: