In this assignment students have to transform iris data into 3 dimensions and plot a 3d chart with transformed dimensions and colour each data point with specific class.

In [14]:

```
import pandas as pd
from sklearn.datasets import load_iris
import matplotlib.pyplot as plt

data = load_iris()
df = pd.DataFrame(data['data'], columns=data['feature_names'])
df['target'] = data['target']
df.head()
```

Out[14]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

In [15]:

```
#Segregating independent and dependent variables

X = df.iloc[:,:].values
Y = df.iloc[:,4].values
```

In [16]:

```
#label encoding the Target values

from sklearn.preprocessing import LabelEncoder
label_encoder_y = LabelEncoder()
y = label_encoder_y.fit_transform(y)
```

In [17]:

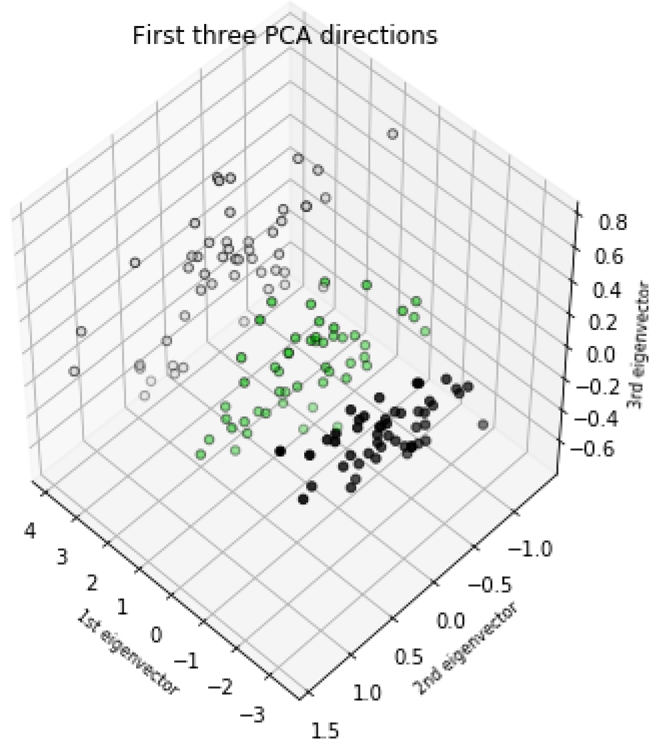
```
#feature scaling

from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_std = scaler.fit_transform(X)
```

In [18]:

```
#Reducing the dimensions for 3D

from sklearn.decomposition import PCA
pca = PCA(n_components=3)
X = pca.fit_transform(X)
```

In [30]:

```
#Visualising the 3D chart

from mpl_toolkits.mplot3d import Axes3D

fig = plt.figure(1, figsize=(5, 5))
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)
plt.cla()
ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=y, cmap=plt.cm.spectral, edgecolor='k')

ax.set_title("First three PCA directions")

ax.set_xlabel('1st eigenvector', fontsize = 8)
ax.set_ylabel('2nd eigenvector', fontsize = 8)
ax.set_zlabel('3rd eigenvector', fontsize = 8)

plt.show()
```


First three PCA directions
3rd eigenvector
2nd eigenvector
1st eigenvector