In [1]:

```
!pip install scikit-learn matplotlib numpy
```

```
Requirement already satisfied: scikit-learn in ./opt/anaconda3/lib/pyt
hon3.9/site-packages (0.24.2)
Requirement already satisfied: matplotlib in ./opt/anaconda3/lib/pytho
n3.9/site-packages (3.4.3)
Requirement already satisfied: numpy in ./opt/anaconda3/lib/python3.9/
site-packages (1.20.3)
Requirement already satisfied: threadpoolctl>=2.0.0 in ./opt/anaconda
3/lib/python3.9/site-packages (from scikit-learn) (2.2.0)
Requirement already satisfied: scipy>=0.19.1 in ./opt/anaconda3/lib/py
thon3.9/site-packages (from scikit-learn) (1.7.1)
Requirement already satisfied: joblib>=0.11 in ./opt/anaconda3/lib/pyt
hon3.9/site-packages (from scikit-learn) (1.1.0)
Requirement already satisfied: cycler>=0.10 in ./opt/anaconda3/lib/pyt
hon3.9/site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: pyparsing>=2.2.1 in ./opt/anaconda3/li
b/python3.9/site-packages (from matplotlib) (3.0.4)
Requirement already satisfied: kiwisolver>=1.0.1 in ./opt/anaconda3/li
b/python3.9/site-packages (from matplotlib) (1.3.1)
Requirement already satisfied: pillow>=6.2.0 in ./opt/anaconda3/lib/py
thon3.9/site-packages (from matplotlib) (8.4.0)
Requirement already satisfied: python-dateutil>=2.7 in ./opt/anaconda
3/lib/python3.9/site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six in ./opt/anaconda3/lib/python3.9/si
te-packages (from cycler>=0.10->matplotlib) (1.16.0)
```

In [2]:

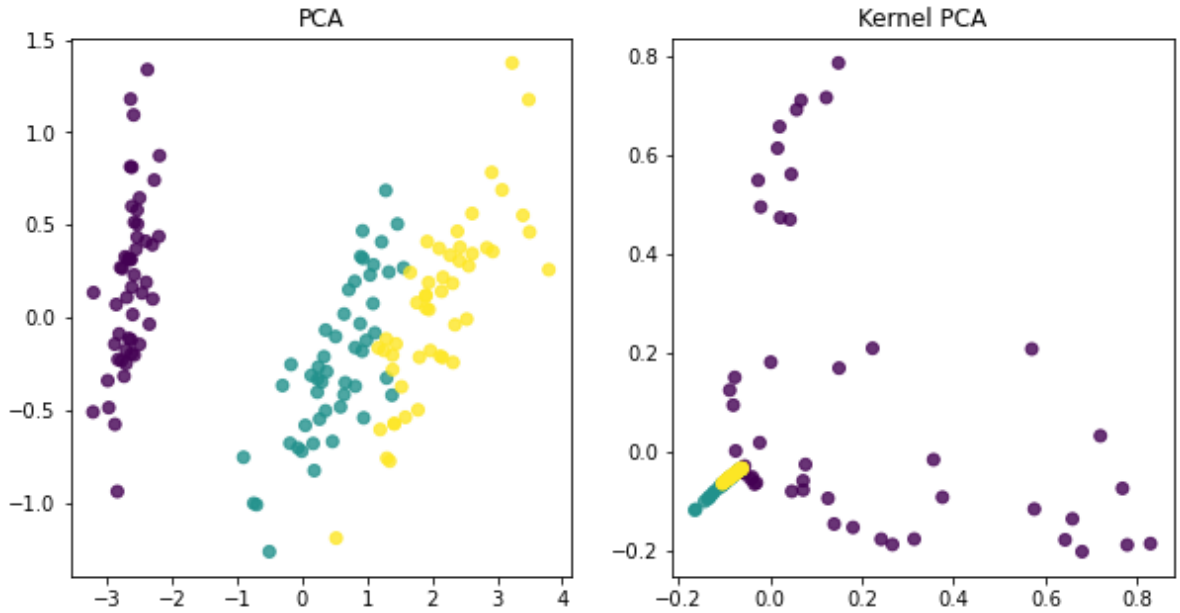
```
from sklearn.datasets import load_iris
iris = load_iris()
X = iris.data
y = iris.target
```

In [3]:

```
from sklearn.decomposition import PCA, KernelPCA
# PCA
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
# Kernel PCA
kpca = KernelPCA(n_components=2, kernel='rbf', gamma=15)
X_kpca = kpca.fit_transform(X)
```

In [4]:

```
import matplotlib.pyplot as plt
plt.figure(figsize=(10, 5))
plt.subplot(1, 2, 1)
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y, alpha=0.8)
plt.title('PCA')
plt.subplot(1, 2, 2)
plt.scatter(X_kpca[:, 0], X_kpca[:, 1], c=y, alpha=0.8)
plt.title('Kernel PCA')
plt.show()
```



In [5]:

```
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA, KernelPCA
import matplotlib.pyplot as plt
```

In [6]:

```
iris = load_iris()
X = iris.data
y = iris.target
```

In [7]:

```
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
```

In [8]:

```
kpca = KernelPCA(n_components=2, kernel='rbf', gamma=15)
X_kpca = kpca.fit_transform(X)
```

In [9]:

```
plt.figure(figsize=(10, 5))
```

Out[9]:

```
<Figure size 720x360 with 0 Axes>
```

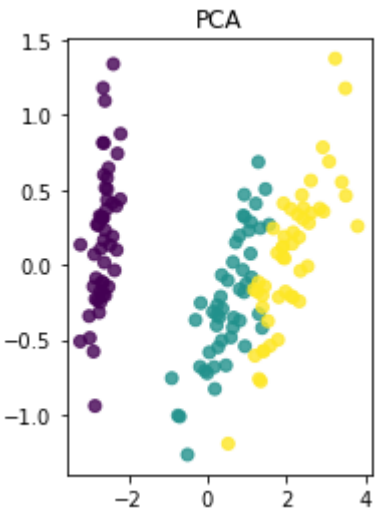
```
<Figure size 720x360 with 0 Axes>
```

In [10]:

```
plt.subplot(1, 2, 1)
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y, alpha=0.8)
plt.title('PCA')
```

Out[10]:

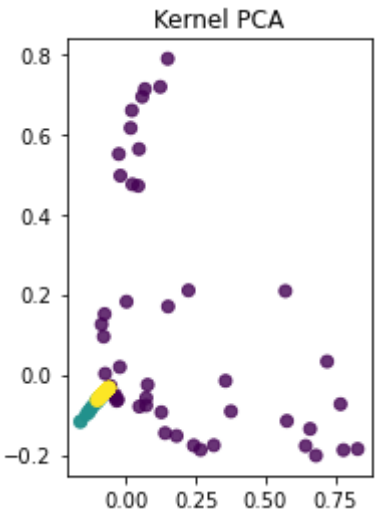
```
Text(0.5, 1.0, 'PCA')
```



In [11]:

```
plt.subplot(1, 2, 2)
plt.scatter(X_kpca[:, 0], X_kpca[:, 1], c=y, alpha=0.8)
plt.title('Kernel PCA')

plt.show()
```



In [ ]: