

PR Assignment-2

G HAREESH
18MCMB15
M.TECH (IT)

1st question

def standerize (data):

```
    mean = np.mean(data,axis=0)
```

```
    std = np.std(data,axis=0)
```

```
    return (data - mean) / std
```

def getpca(k,data,get_eig = False):

```
    data = standerize(data)
```

```
    cov = np.cov(data.T)
```

```
    eig_vals , eig_vecs = np.linalg.eig(cov)
```

```
    eig_pairs = [(np.abs(eig_vals[i]),eig_vecs[:,i]) for i in range(len(eig_vals))]
```

```
    eig_pairs.sort(key = lambda k:k[0] , reverse = True)
```

```
    mat = np.hstack([eig_pairs[i][1][:,np.newaxis] for i in range(k)])
```

```
    if get_eig == True:
```

```
        return eigen_vals , data.dot(mat)
```

```
    return data.dot(mat)
```

def getlda(k,data,datalabels):

```

labels = np.unique(datalabels)

data = standerdize(data)

mean_vecs = []

mean_labels = []

for label in labels:

mean_vecs.append(np.mean(data[datalabels==label] , axis=0))

mean_labels.append(label)

d = data.shape[1]

S_W = np.zeros((d,d))

for label,mv in zip(labels,mean_vecs):

class_scatter = np.zeros((d,d))

for row in data[datalabels==label]:

row, mv = row.reshape(d, 1), mv.reshape(d, 1)
class_scatter += (row-mv).dot((row-mv).T)
S_W += class_scatter

"""for label,mv in zip(range(1, 4), mean_vecs):

class_scatter = np.cov(X_train_std[y_train==label].T)

S_W += class_scatter"""

```

```

mean_overall = np.mean(data, axis=0)

mean_overall = mean_overall.reshape(d,1)

S_B = np.zeros((d, d))

for label, mean_vec in zip(mean_labels,mean_vecs):

n = data[datalabels == label].shape[0]

mean_vec = mean_vec.reshape(d,1)

S_B += n * (mean_vec - mean_overall).dot((mean_vec - mean_overall).T)

eigen_vals, eigen_vecs = np.linalg.eig(np.linalg.inv(S_W) @ (S_B))

eigen_pairs = [(np.abs(eigen_vals[i]), eigen_vecs[:,i]) for i in range(len(eigen_vals))]
eigen_pairs = sorted(eigen_pairs,key=lambda k: k[0], reverse=True) mat =
np.hstack([eigen_pairs[i][1][:,np.newaxis] for i in range(k)])

return data.dot(mat)

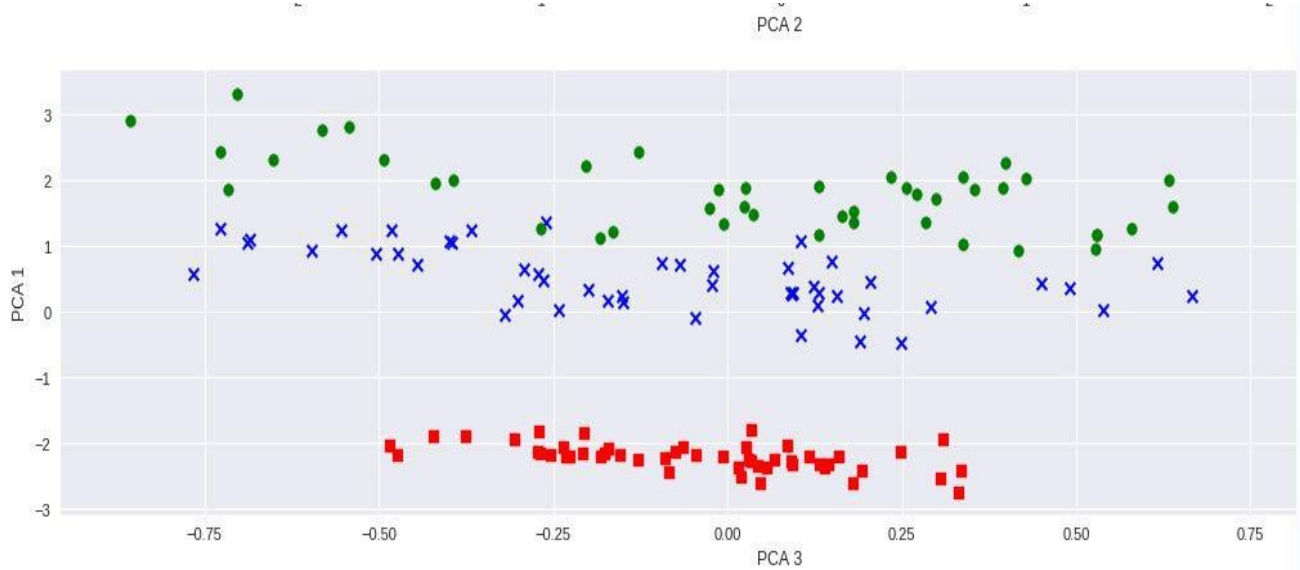
```

2nd Question:

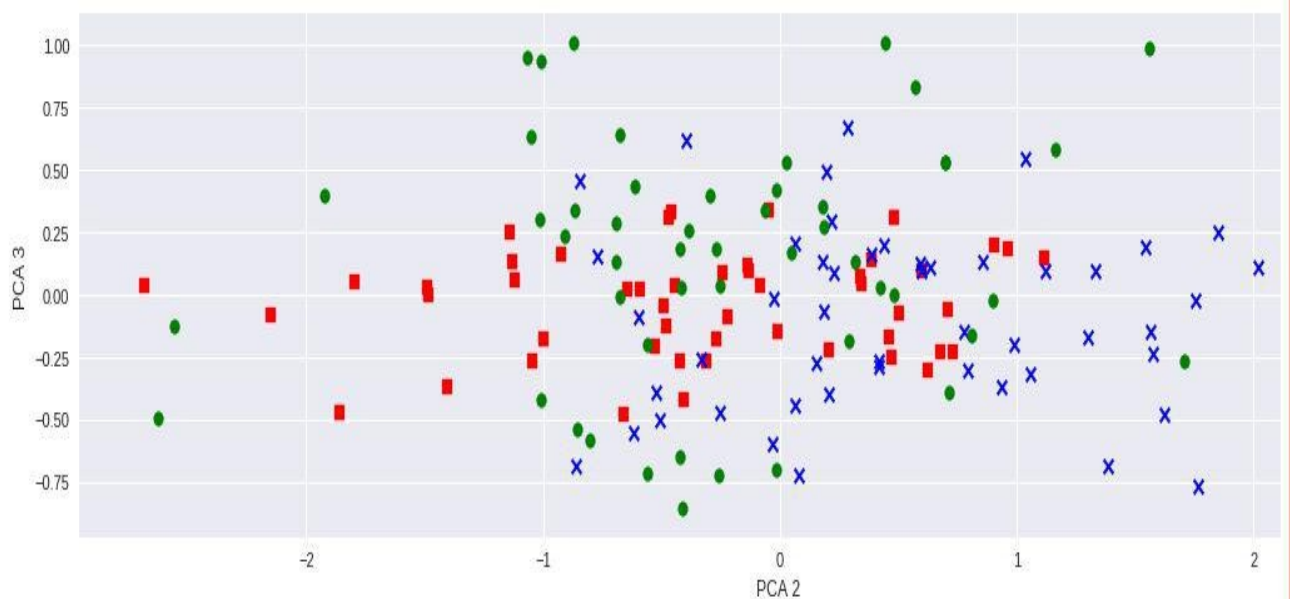
Results :

(Snapshots of the Results TO view them completely Open Notebook)

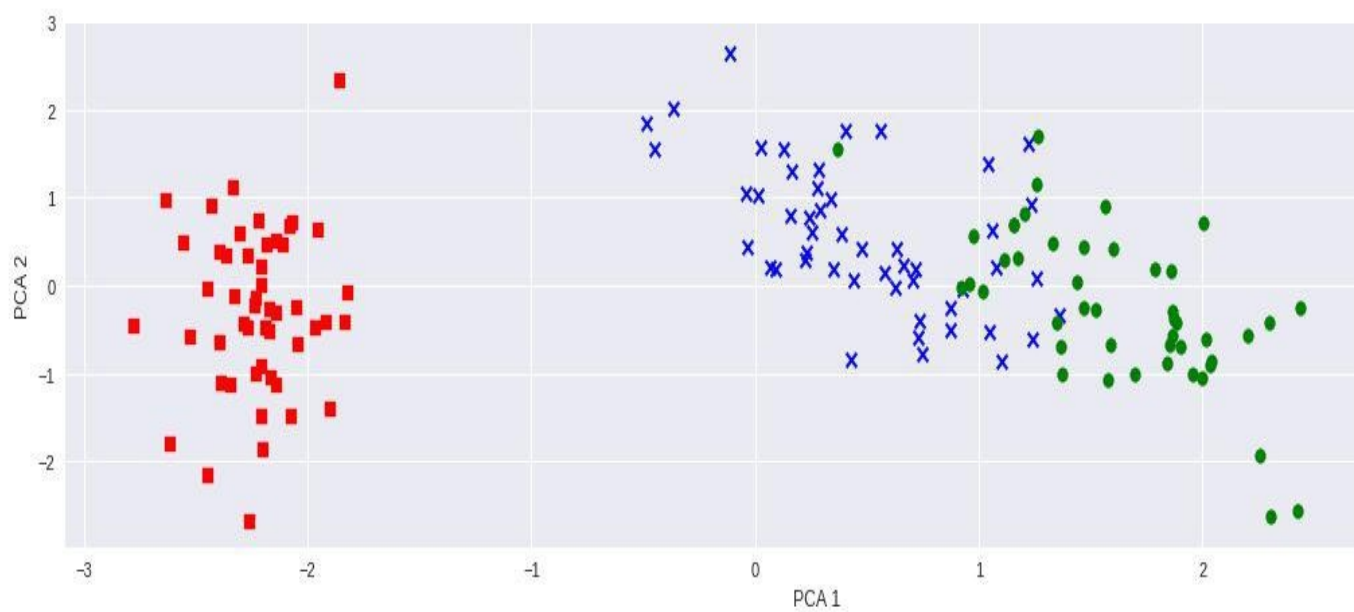
PCA1 vs PCA3



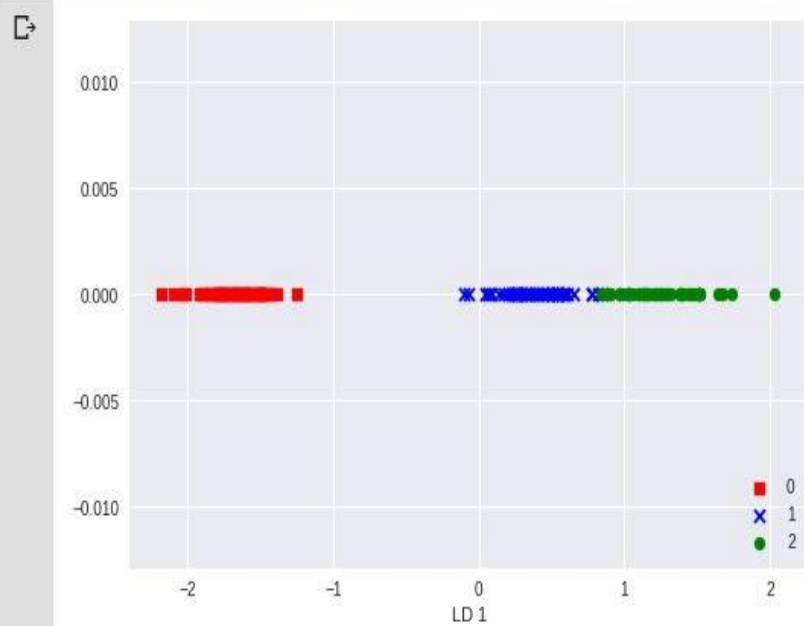
PCA2 vs PCA3



PCA1 vs PCA2

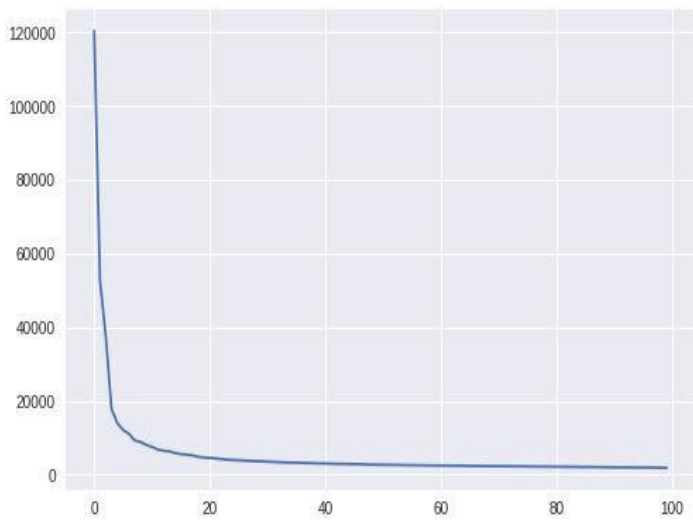


LDA :



Arcene :

plot.show()



3rd question

Libraries Used:

```
import numpy as np
```

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

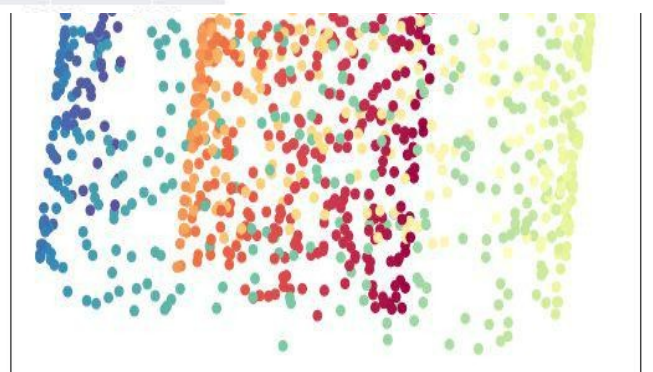
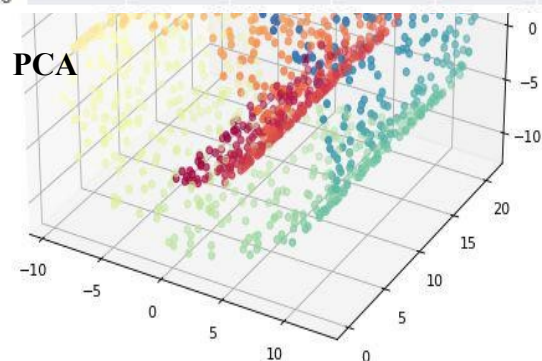
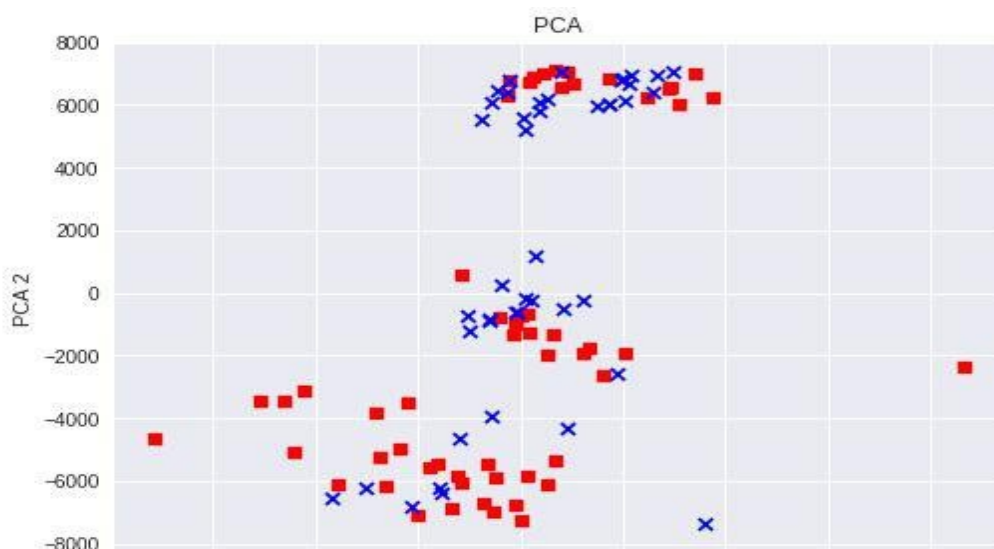
Dataset Used:

UCI Arcene cancer dataset

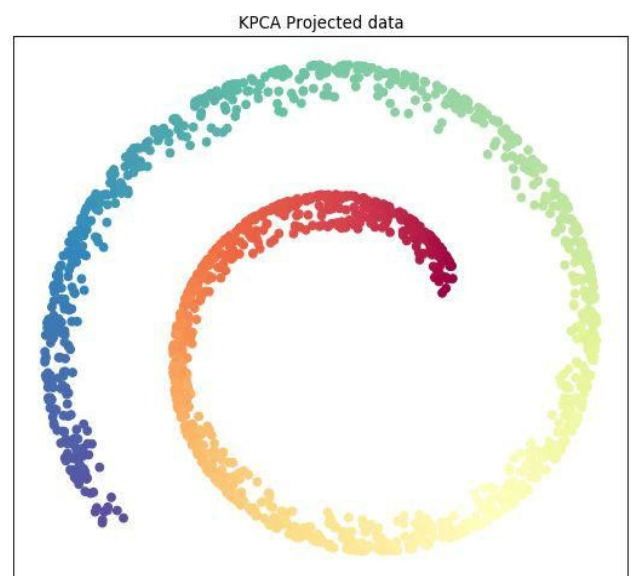
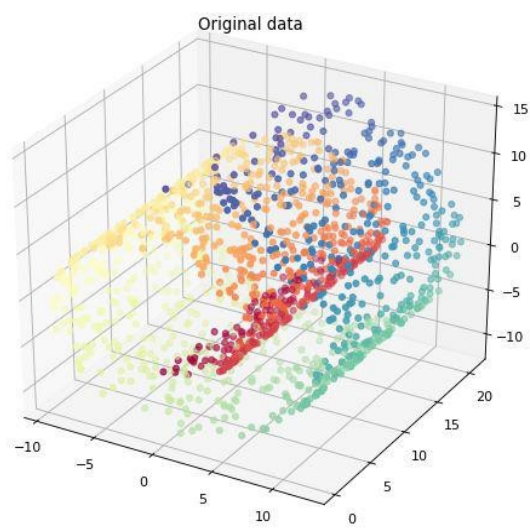
Analysis of components with respect to variance: No of Components required for getting 0.85 variances are : 61 No of Components required for getting 0.90 variances are : 73 No of Components required for getting 0.95 variances are : 86 No of Components required for getting 0.99 variances are : 97

Results:

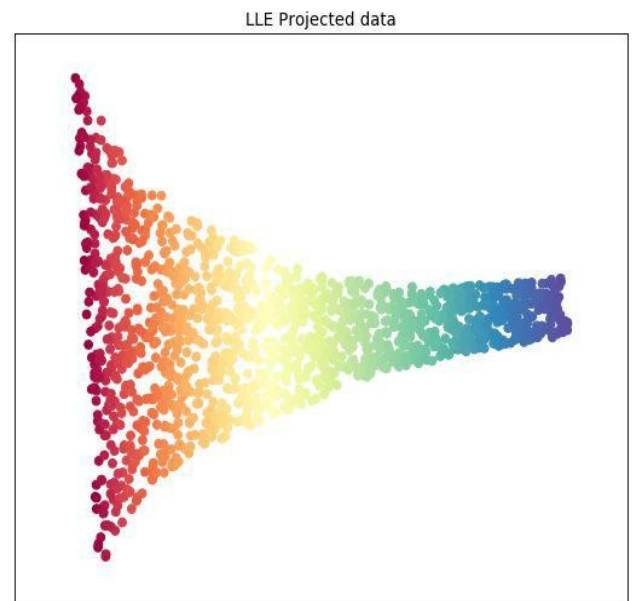
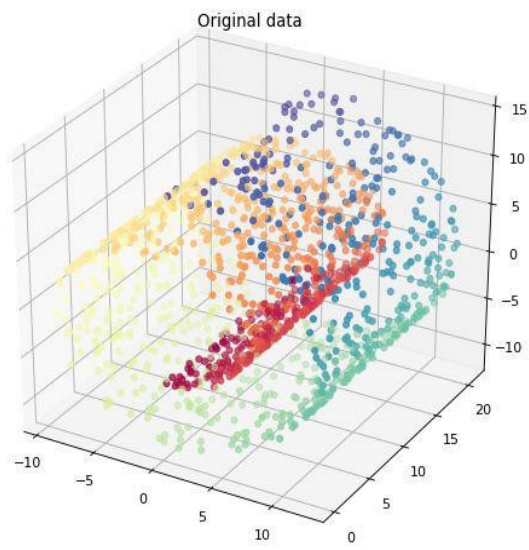
Pca two principal component



KPCA



LLE

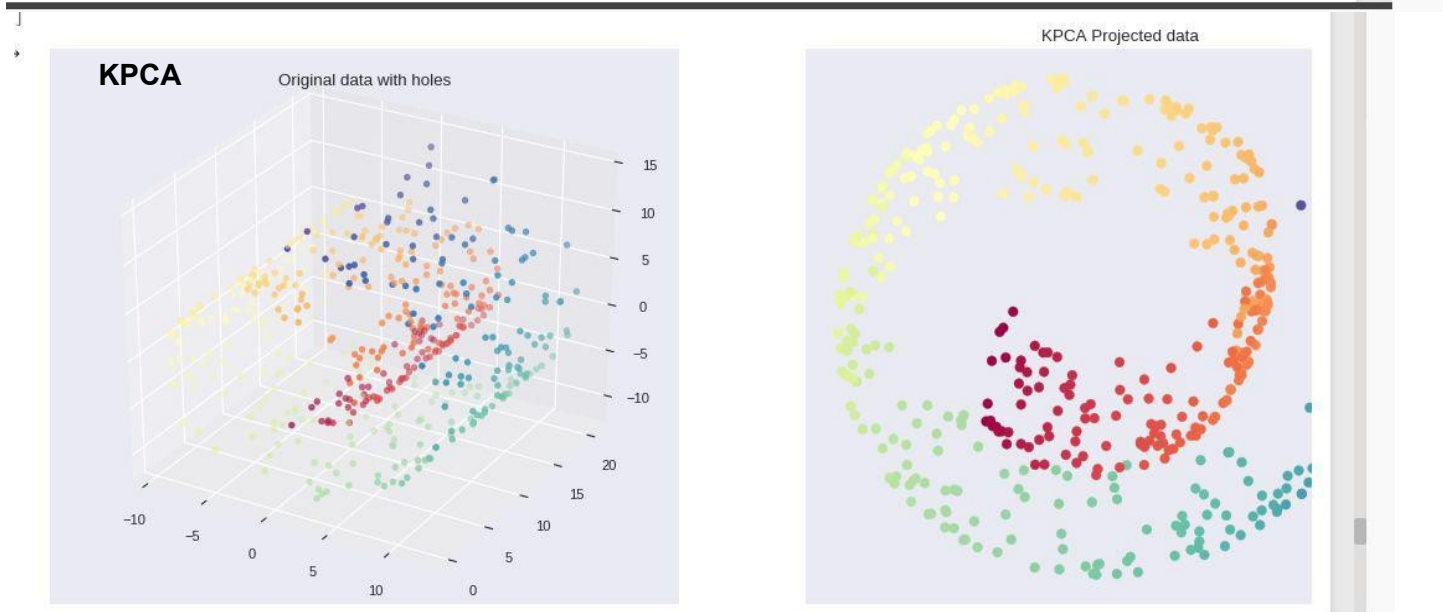
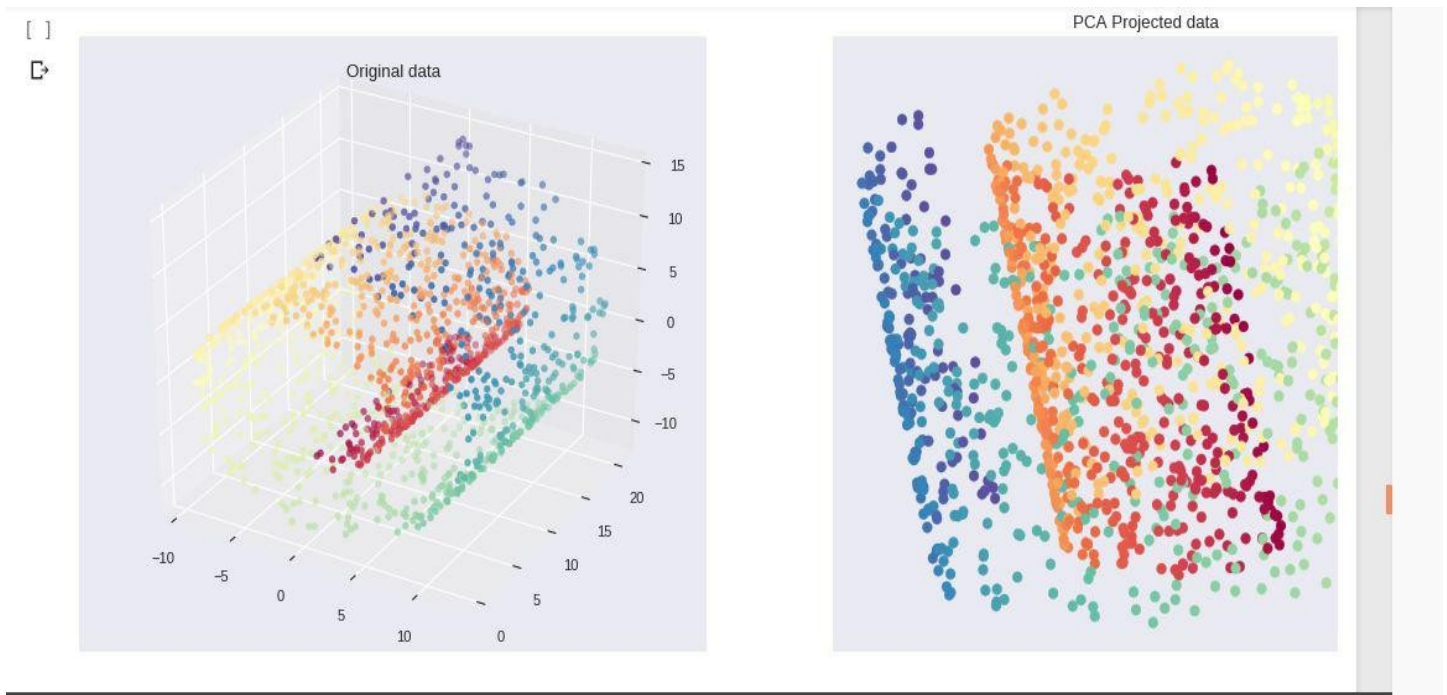


4th Question

Dataset Used:
3D swissroll dataset

Results :

PCA



LLE

