Dimensionality_Reduction_Using_Feature_Extraction

```
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn import datasets

digits = datasets.load_digits()

X = StandardScaler().fit_transform(digits.data)
pca = PCA(n_components = 0.99 , whiten = True )

X_pca = pca.fit_transform(X)

print('original number of features',X.shape[1])
print('reduced number of features',X_pca.shape[1])
```

```
     original number of features 64
     reduced number of features 54
     original number of features 64
     reduced number of features 54
```

```
# Load libraries
from sklearn.decomposition import PCA, KernelPCA
from sklearn.datasets import make_circles
# Create linearly inseparable data
X, _ = make_circles (n_samples=1000, random_state=1, noise=0.1, factor=0.1)
# Apply kernal PCA with radius basis function (RBF) kernel
kpca = KernelPCA (kernel="rbf", gamma=15, n_components=1)
X_kpca = kpca.fit_transform(X)
print('Original number of features:', X.shape[1])
print('Reduced number of features:', X_kpca.shape[1])
```

```
     Original number of features: 2
     Reduced number of features: 1
```

```
from sklearn import datasets
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis

iris = datasets.load_iris()
x = iris.data
y = iris.target

lda = LinearDiscriminantAnalysis(n_components=1)

X_lda= lda.fit(x, y).transform(X)

print('Original number of features:', X.shape[1])
print('Reduced number of features:', X_lda.shape[1])
print('Reduced number of features: ', X_lda.shape[1])

     Original number of features: 4
     Reduced number of features: 1
```

```
lda.explained_variance_ratio_

     array([0.9912126])
```

Start coding or generate with AI.