CODE

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Mar  9 12:23:57 2020

@author: M
"""
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

Xtest=pd.read_csv('XTEST.csv')
Xtrain=pd.read_csv('XTRAIN.csv')
Ytest=pd.read_csv('YTEST.csv')
Ytrain=pd.read_csv('YTRAIN.csv')
X = pd.concat([Xtrain, Xtest], axis = 0)
Y = pd.concat([Ytrain, Ytest], axis = 0)
X.head()
y2 = Y
y2 = np.asarray(y2)
y2 = pd.DataFrame(y2)
y2[0][367]
for i in range(0, len(y2)):
    if(y2[0][i] == 1):
        y2[0][i] = 'Walking'
    elif(y2[0][i] == 2):
        y2[0][i] = 'Walking Upstairs'
    elif(y2[0][i] == 3):
        y2[0][i] = 'Walking Downstairs'
    elif(y2[0][i] == 4):
        y2[0][i] = 'Sitting'
    elif(y2[0][i] == 5):
        y2[0][i] = 'Standing'
    elif(y2[0][i] == 6):
        y2[0][i] = 'Laying'
X.shape
#PCA algorithm being used for the dimensionality reduction process
from sklearn.decomposition import PCA
```

Editor - C:\Users\M\Desktop\UCI HAR Dataset (1)\UCI HAR Dataset\untitled2.py

```python
    elif(y2[0][i] == 4):
        y2[0][i] = 'Sitting'
    elif(y2[0][i] == 5):
        y2[0][i] = 'Standing'
    elif(y2[0][i] == 6):
        y2[0][i] = 'Laying'
X.shape
#PCA algorithm being used for the dimensionality reduction process
from sklearn.decomposition import PCA
pca = PCA(n_components=1)
pca.fit(Y)

y_trans = pca.transform(Y)
y_trans = pd.DataFrame(y_trans, columns = ['0'])
y_trans.head()
y_trans['label'] = y2[0]

y_trans.tail()
import matplotlib
sns.lmplot(x = '0', y = '0', data = y_trans, hue = 'label',  fit_reg = False, x_jitter = 4.25, y_jitter = 2.4, size = 15)
plt.xlabel('F1 Feature')
plt.ylabel('F2 Feature')
import matplotlib
matplotlib.rcParams.update({'font.size': 20})
plt.show()
plt.figure(figsize = (18, 10))
sns.countplot(x = 'label', data = y_trans)
plt.tight_layout()
plt.show()

#F1 strip plot
plt.figure(figsize = (15, 8))
sns.stripplot(x = 'label', y= '0', data = y_trans, jitter = True, dodge = True)
plt.tight_layout()
plt.xlabel('Class Label')
plt.ylabel(' Feature')
plt.show()
```