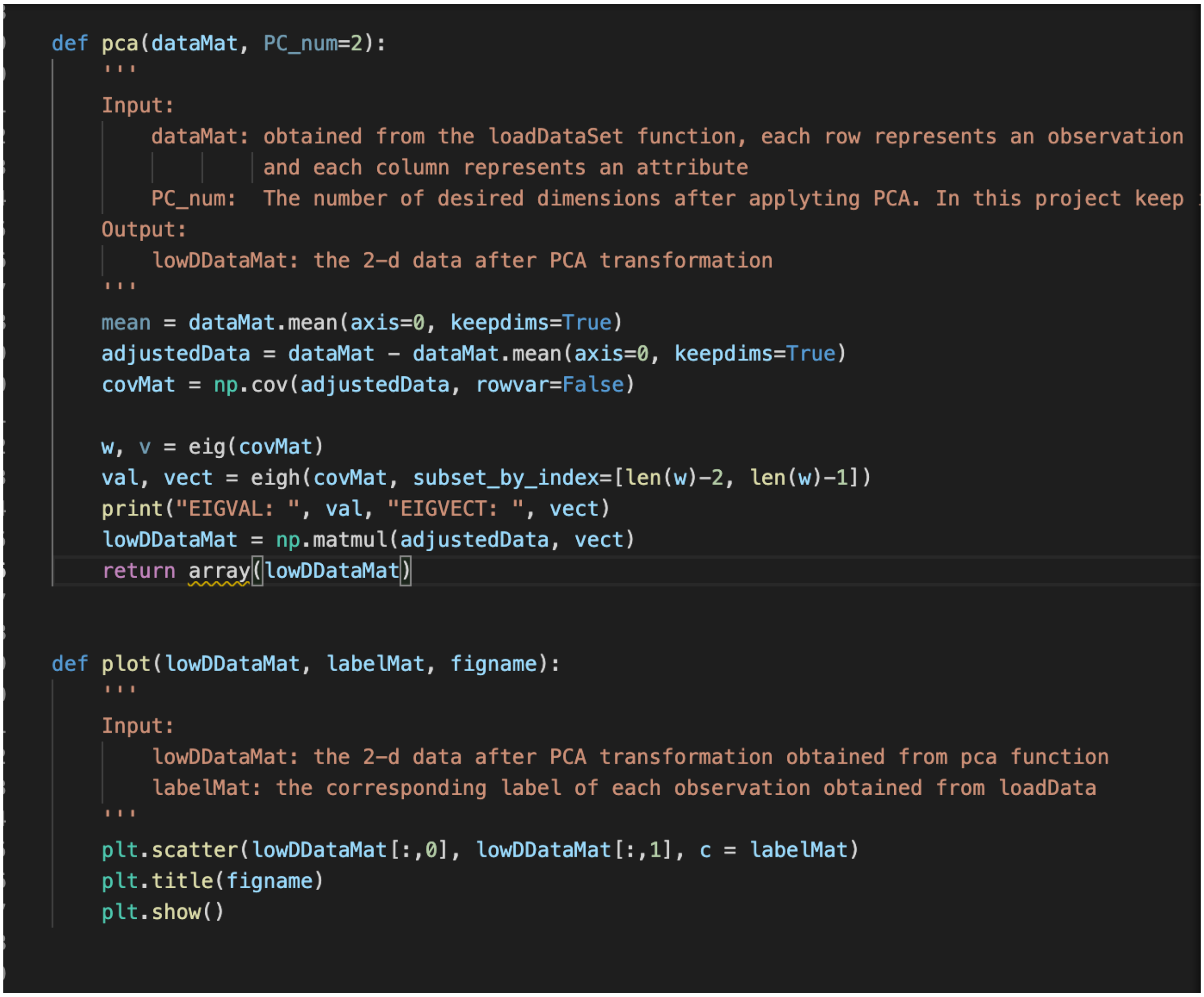
Vivek Khanolkar



```
def pca(dataMat, PC_num=2):
    '''
    Input:
        dataMat: obtained from the loadDataSet function, each row represents an observation
                 and each column represents an attribute
        PC_num:  The number of desired dimensions after applyting PCA. In this project keep
    Output:
        lowDDataMat: the 2-d data after PCA transformation
    '''
    mean = dataMat.mean(axis=0, keepdims=True)
    adjustedData = dataMat - dataMat.mean(axis=0, keepdims=True)
    covMat = np.cov(adjustedData, rowvar=False)

    w, v = eig(covMat)
    val, vect = eigh(covMat, subset_by_index=[len(w)-2, len(w)-1])
    print("EIGVAL: ", val, "EIGVECT: ", vect)
    lowDDataMat = np.matmul(adjustedData, vect)
    return array(lowDDataMat)


def plot(lowDDataMat, labelMat, figname):
    '''
    Input:
        lowDDataMat: the 2-d data after PCA transformation obtained from pca function
        labelMat: the corresponding label of each observation obtained from loadData
    '''
    plt.scatter(lowDDataMat[:,0], lowDDataMat[:,1], c = labelMat)
    plt.title(figname)
    plt.show()
```

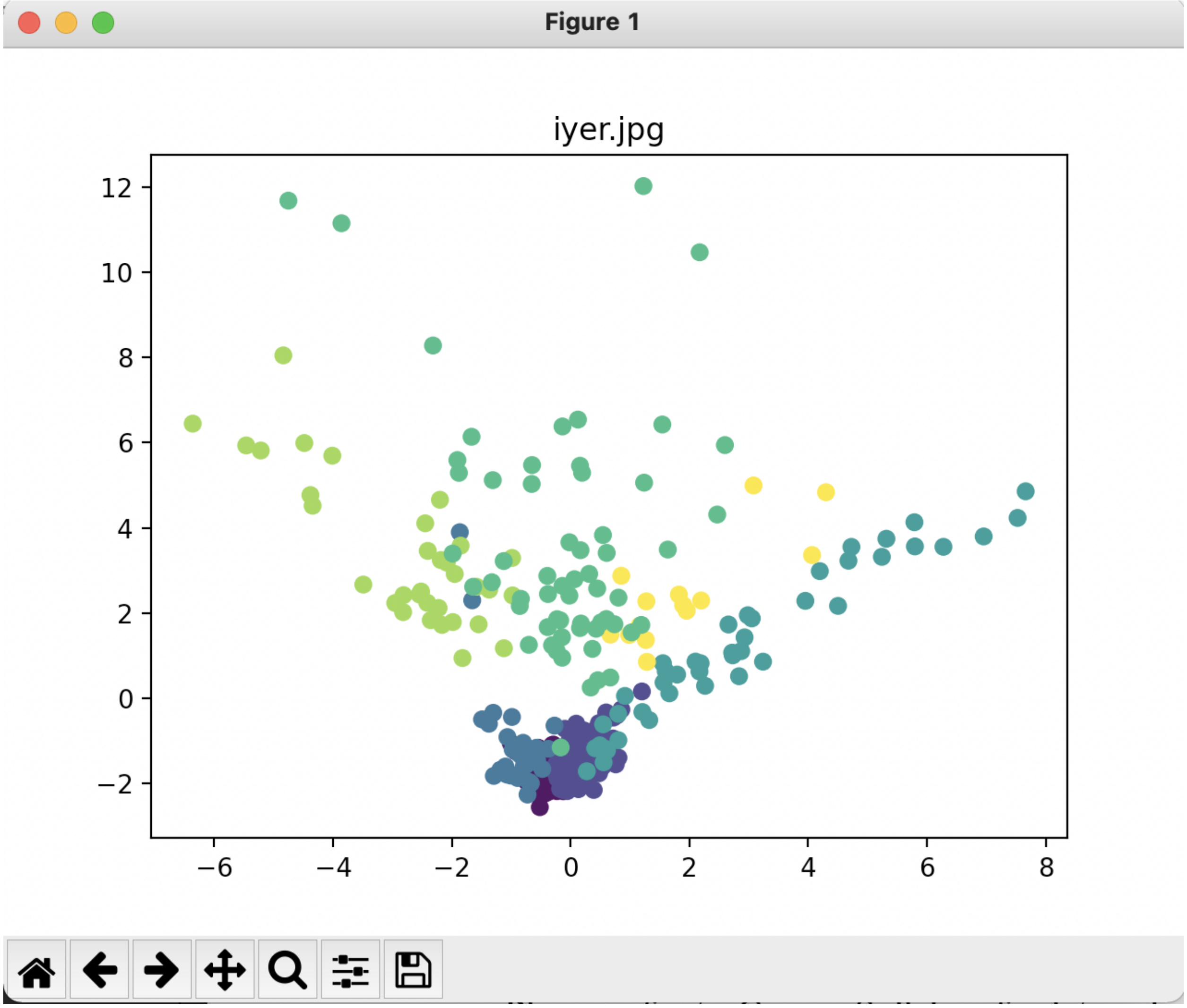


Figure 1
iyer.jpg
12
10
8
6
4
2
0
−2
−6
−4
−2
0
2
4
6
8

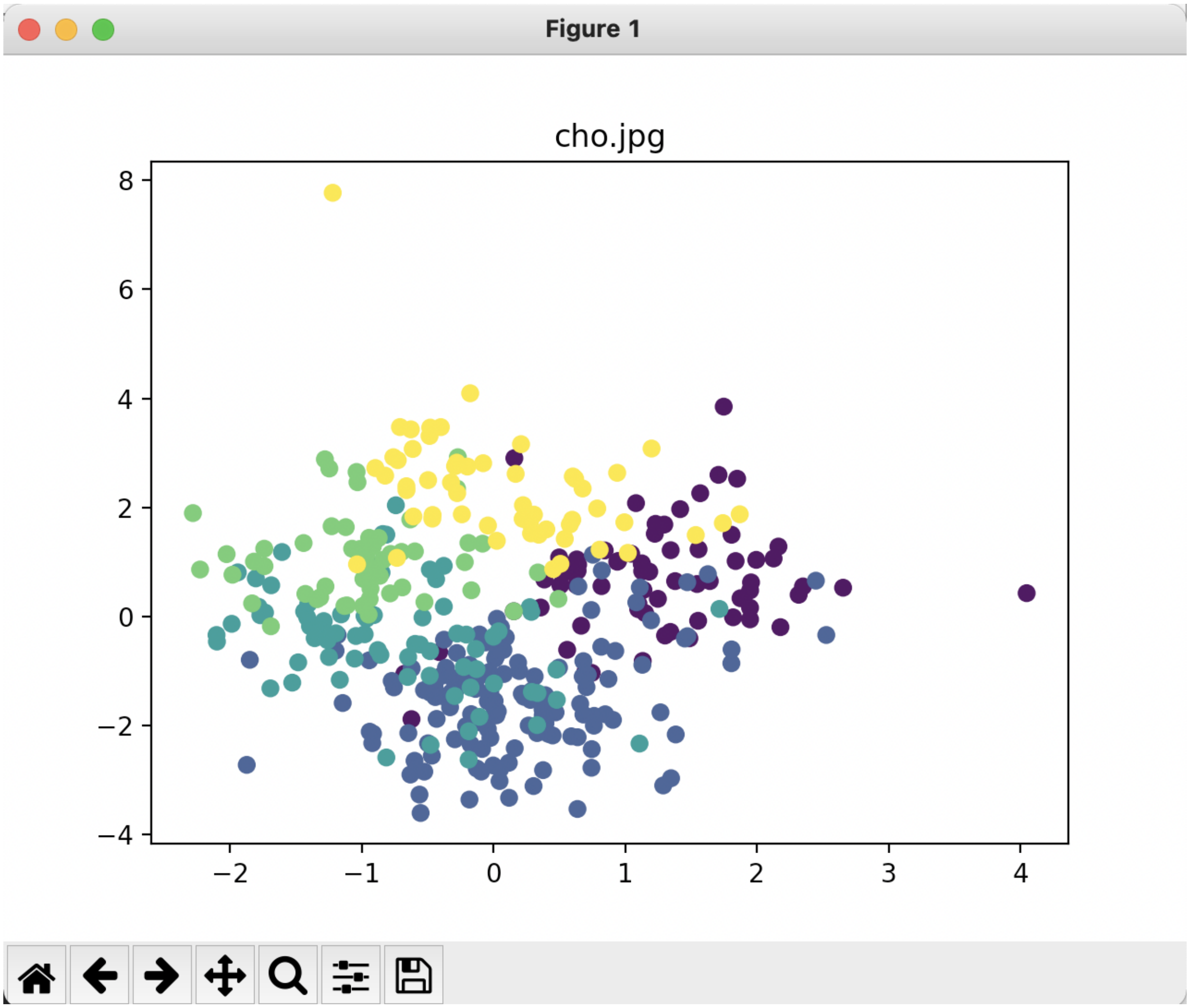


Figure 1
cho.jpg
8
6
4
2
0
−2
−4
−2
−1
0
1
2
3
4