Assignment 1: Dimensionality Reduction and Visualization

Due: February 15 2022 11:59PM

Three datasets (Iris, Cho, Iyer) with cluster labels are provided. In each dataset file, the last column is the cluster label, and the remaining columns are attributes.

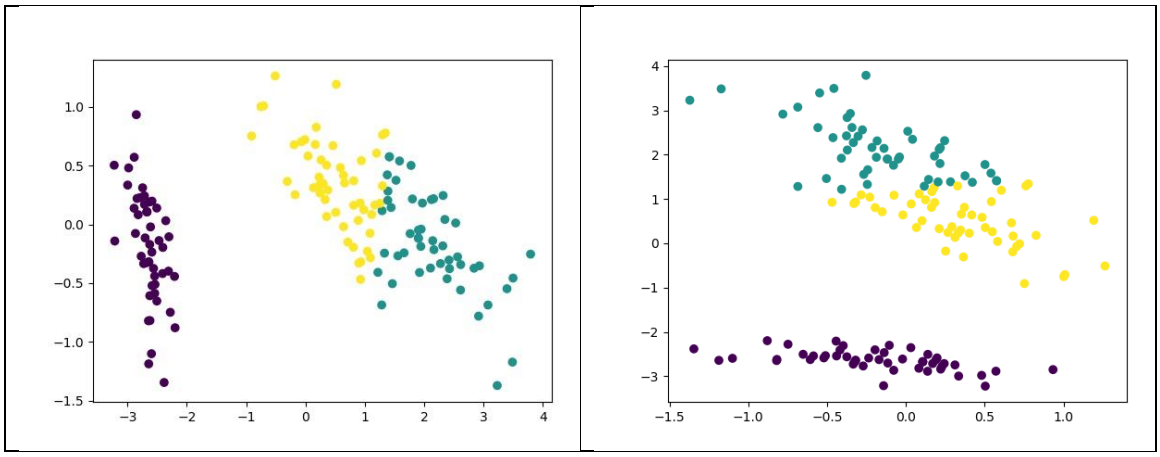
In this assignment, you need to implement Principle Component Analysis (PCA) algorithm by yourself to map high-dimensional data to 2 dimensions, and plot the 2-dimensional data points. You are not allowed to call existing PCA libraries directly.

Please take the following steps:

1. The template based on Python is provided. In the template, necessary libraries are imported. You only need to complete the required functions (pca and plot) in that template. Please do not change the input and output provided in the template. The details of the template are explained as follows:
There are four functions in the template: ***loadDataSet, pca, plot*** and ***the main function***.
The loadDataSet function is to load the dataset from the csv file. The input of this function is the filename of the dataset and the outputs are the data matrix (*dataMat*) and corresponding labels (*labelMat*). Each row in the *dataMat* represents an observation and each column in the *dataMat* represents an attribute. Each entry of *labelMat* is the label corresponding to each row of *dataMat*.
You need to implement PCA algorithm in the pca function. The input of the pca function is *dataMat* obtained from the *loadDataSet* function and the number of dimensions after PCA transformation which is set to be 2. The output of the pca function is the two-dimensional data(*lowDDataMat*) after PCA transformation.
In the plot function you need to plot all observations as scatter plots and color the data points according to their labels. You also need to save the figure. The input to the plot function is the data matrix after PCA transformation (*lowDDataMat*) obtained from the pca function, the label vector (*labelMat*) obtained from the loadDataSet function and the name of the saved figure (figname).
The aforementioned functions are called in the main function. To run the template, you can use the command line and then type the following command:
Python pca_template.py [filename]
You can also run the template in an IDE such as PyCharm and specify the configuration.
The parameter filename is an optional parameter to specify the name of the data file you want to read. If it is not specified, the default value ('iris_with_cluster.csv') will be used.

2. Apply PCA on the Iris dataset and get the two-dimensional data points. Draw them in a scatter plot, and color them according to their cluster labels. Compare the scatter plot with the given plots below and see if you get the plot correctly. If your plot matches either of the following plots, it is correct. Note that it is NOT permitted to use an existing

function/package that directly achieves the final results. If you are not sure about whether it is OK to use a certain function, please post your question on Piazza.



Iris Data

3. If Step 2 works fine, then apply PCA on the Cho and Iyer datasets and draw a scatter plot for each dataset following the same procedure discussed in Step 2.

4. Prepare your submission. Your final submission should be a zip file named as Assignment1.zip. In the zip file, you should include:

- The codes you implemented (i.e., the python script).
- Report: A Word or PDF file named as Assignment1.doc or Assignment1.docx or Assignment1.pdf. The report should consist of the following parts: 1) Two scatter plots obtained by running PCA on Cho and Iyer datasets. Please label them properly by the dataset names and have tick marks along both axes in each plot. 2) The codes of PCA and plot drawing.

5. Submit the zip file under Assignment 1 on Brightspace.

Please refer to Course Syllabus for late submission policy and academic integrity policy. This assignment must be done independently. Running your submitted code should be able to reproduce the plots in the report.