

Assignment 10

Principal component analysis (PCA)

EECE-6822 Machine Learning

Cris Ababei
Electrical and Computer Engr., Marquette University

1. Objective

The objectives of this activity include: (1) run several code examples that illustrate PCA; and (2) conduct an investigation on the use of PCA based embeddings for a face identification example.

2. Prerequisite Readings

Murphy

- PCA, Singular value decomposition: Murphy 20.1
- Kernel PCA: Murphy 20.4.6

Geron

- Ch.8: Dimensionality reduction

Raschka

- Ch.5: Compressing data via dimensionality reduction

3. Code Examples

Example 1: Dimensionality reduction, PCA ←

This is the example code from **Ch.8** from Aurelian Geron's book.

[*B3-Geron] Aurelien Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly, 2022.

Open in your google colab and go through the code. You should read first the chapter itself from the book, before going through the code.

https://github.com/ageron/handson-ml3/blob/main/08_dimensionality_reduction.ipynb

The chapter discusses the curse of dimensionality. Then, it considers the two main approaches to dimensionality reduction (projection and manifold learning). It presents then several of the most popular dimensionality reduction techniques, including PCA.

Example 2: Compressing data via dimensionality reduction ←

This code example is from **Ch.5** from:

[*B3-Raschka] Sebastian Raschka, Yuxi Liu, and Vahid Mirjalili, *Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python*, Packt Publishing, 2022.

The source code (both Python code and Jupyter Notebook) is located at the GitHub repository. You should read first the chapter itself from the book, before going through the code.

<https://github.com/rasbt/machine-learning-book/tree/main/ch05>

To execute the code, I found the easiest usually to run directly the Python code (same code as in the notebook) in Anaconda Spyder. However, because the source code was developed with earlier versions of various libraries, you may need to fix little things here and there to make it work.

The first part of this chapter provides a simple Python implementation of principal component analysis (PCA) for unsupervised data compression; it uses the wine dataset. Then, it presents PCA's implementation from SciKit-Learn. Notably, it presents a discussion of how to assess feature contributions (i.e., loadings).

Example 3: PCA examples

These are examples suggested by K. Murphy on his github repository for the textbook:

<https://github.com/probml/pyprobml/tree/master/notebooks/book1/20>

Look on the above page for the notebooks that have PCA in their file name and study them.

Example 4: mlm tutorials

Here, we look at several tutorials from **machinelearningmastery (mlm)**. Read the following tutorials and run the code where applicable:

-- How to Calculate Principal Component Analysis (PCA) from Scratch in Python

<https://machinelearningmastery.com/calculate-principal-component-analysis-scratch-python/>

-- Principal Component Analysis for Dimensionality Reduction in Python

<https://machinelearningmastery.com/principal-components-analysis-for-dimensionality-reduction-in-python/>

--6 Dimensionality Reduction Algorithms With Python

<https://machinelearningmastery.com/dimensionality-reduction-algorithms-with-python/>

--Face Recognition using Principal Component Analysis ←

<https://machinelearningmastery.com/face-recognition-using-principal-component-analysis/>

4. Assignment

In this assignment, you must change the example demoed in the previous lecture 6, in class:

Code 1: `mlm-face-recognition-system-using-facenet-in-keras-and-an-svm-classifier.py`

In that example we saw a “face identification” pipeline, working with a dataset with images of 5 famous people, that essentially used:

1--**MTCNN** model to do face-detection and cropping of those detected faces into 160x160 images.

2--**FaceNet** model to pre-process detected faces to create face embeddings

3--**Linear SVM** to classify any given face-embedding as one of the 5 people

Your task here is to replace the use of **FaceNet** model (that generated the embeddings) with the **PCA**, which was demoed in this lecture 7:

Code 2: `mlm-face-recognition-using-principal-component-analysis.py`

Instead of using as input into “3--Linear SVM” the embeddings created by “2--FaceNet”, you must use the “**weights**” (calculated in step #6 inside `mlm-face-recognition-using-principal-component-analysis.py`). So, the objective of this assignment is to investigate if we could use the PCA’s weights as “embeddings” in Code 1 and still get good performance; your analysis will answer that question, by reporting the new performance and comparing it to the initial one achieved in Code 1 example. **Note** that we have 50 weights in this example for any given image.

NOTES:

1--Note that you will have to apply the PCA code from Code 2 to the dataset from Code 1!

2—Initially, try to keep in Code 1 the size of images as it is now: `required_size=(160, 160)`. If it turns out to be too large, then you can change it to match the size that code 2 works with, i.e., 112x92 pixels.

3—Note also that you may need to convert images from jpeg to pgm format to be able to reuse the code from Code 2 with minimal changes.

5. Deliverables

You must write (typed) a report and upload it as a PDF file on D2L. The report should be named “**hw10_report_LastName.pdf**”. You should also create a .zip archive with all your code and implementations of all parts of the assignment. Upload also this archive .zip file with the name “**hw10_implementation_code_LastName.zip**” to D2L. Hence, your D2L should contain two items: the report and the .zip file. **Do not include the report inside the .zip and upload only the .zip. They should be two separate items!**

The report should include the following sections and subsections. Make sure section titles are in bold font and pages are numbered.

- 1) **Title + course info + your name**
- 2) **Summary.** Describe in one paragraph what the objective of the assignment is.
- 3) **Description of Experiments and Discussion.** Describe the experiments you did. All tables and figures should be numbered and should have captions. All plots in all figures should have axes labels and titles. Present a meaningful discussion with the interpretation of the results you obtained. Explain if you expected the results or not; discuss the intuition behind it.
- 4) **Conclusion.** Present your conclusions; highlight what are your main takeaways that you learned from this assignment. Describe what issues you encountered and how you solved them.
- 5) **References.** Include all references that you used, as a numbered list. Cite them in the report itself; do not just list them! If your report has References that are not numbered and cited in the report, points will be deducted!