DSC 140B - Homework 03

Due: Wednesday, April 26

Write your solutions to the following problems by either typing them up or handwriting them on another piece of paper. Unless otherwise noted by the problem's instructions, show your work or provide some justification for your answer. Homeworks are due via Gradescope at 11:59 PM.

Problem 1.

As a data scientist, there will be many times when you will be working with massive, high dimensional data sets consisting of hundreds of thousands or even millions of points. This is not one of those times.

In this problem, we'll work with the following data set of three points:

$$x^{(1)} = (1, 3)^T$$
$$x^{(2)} = (-3, -9)^T$$
$$x^{(3)} = (2, 6)^T$$

a) Compute the sample covariance[1] matrix by hand. Show the calculations for each entry of the matrix.

b) What is the top eigenvector of the covariance matrix? You do not need to calculate the eigenvector explicitly, but you should justify your answer.

Hint: plot the data.

c) What is the eigenvalue associated with the top eigenvector?

d) Reduce the dimensionality of each point above by carrying out PCA by hand. Be sure to use the normalized eigenvector. Show your calculations.

Hint: one of your new features should be equal to $-3\sqrt{10}$.

e) The result of PCA is a data set consisting of three numbers. Compute the variance of these three numbers.

Hint: the result should be familiar.

Problem 2.

Let $\vec{x}^{(1)}, \ldots, \vec{x}^{(n)}$ be a set of n *centered* data vectors in $\mathbb{R}^d$. If $\vec{u}$ is a unit vector, we compute the "variance in the direction of $\vec{u}$" by 1) reducing each data vector $\vec{x}^{(i)}$ to a single number $z^{(i)}$ by setting $z^{(i)} = \vec{x}^{(i)} \cdot \vec{u}$; and then 2) computing the variance of these new numbers, $z^{(1)}, \ldots, z^{(n)}$. That is, the "variance in the direction of $\vec{u}$" is defined to be:

$$\frac{1}{n}\sum_{i=1}^{n}(\vec{x}^{(i)} \cdot \vec{u})^2$$

Let C be the sample covariance matrix. Show that:

$$\frac{1}{n}\sum_{i=1}^{n}\left(\vec{x}^{(i)} \cdot \vec{u}\right)^2 = \frac{1}{n}\vec{u}^T C \vec{u}$$

[1] Use the version of the sample covariance defined in lecture, not the one that divides by $n - 1$.

That is, show that the variance in the direction of $\vec{u}$ is also computed by the vector-matrix-vector product $\vec{u}^T C\vec{u}$.

Hint: this is an exercise in vector and matrix algebra. It helps to remember that the covariance matrix, C, can be written as $\frac{1}{n}X^T X$, where X is the *data matrix*. It may help to define $\vec{v} = X\vec{u}$, and to recognize that the ith entry of $\vec{v}$ is $\vec{x}^{(i)} \cdot \vec{u}$. It is also helpful to remember that, for any vector $\vec{a}$, $\vec{a}^T\vec{a} = \|\vec{a}\|^2$, and that for any matrices/vectors A and B, $(AB)^T = B^T A^T$.

Problem 3.

The Olivetti faces data set contains a collection of images of peoples' faces. It can be downloaded at the following URL:

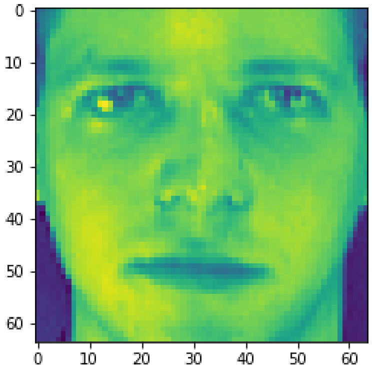
`https://f000.backblazeb2.com/file/dsc-data/faces.csv`

Each row in that file represents one face. It is a vector with 4096 entries, each entry recording the intensity of a pixel in a 64 × 64 image. The row can be reshaped and plotted using `matplotlib` to display it as an image. For example, the code below plots the first image in the data set.

```
import numpy as np
import matplotlib.pyplot as plt

faces = np.loadtxt("faces.csv", delimiter=',')
example_image = faces[0]
plt.imshow(example_image.reshape((64, 64)))
```

You should see this image:



The full set of images forms a point cloud in 4096-dimensional space. The directions in which this point cloud has the greatest variance are often human-interpretable, in that they tend to correspond to ways in which faces vary. For instance, one "dimension" along which faces vary is in the presence/absence of facial hair; another is in the presence/absence of a nose ring, etc.

Some of the people in this dataset are wearing glasses – but who? Since the eigenvectors of the covariance matrix correspond to directions of maximum variance, we can use them to find eyeglass wearers without using any labels whatsoever.

For this problem, you can use whatever programming language or package you'd like. But, as always, the tradeoff is that if you choose to use a package, you're expected to read the documentation to figure out how the package works.

a) Compute the **seventh** eigenvector $\vec{u}^{(7)}$ of the data set's sample covariance matrix (that is, the eigenvector with seventh largest eigenvalue). It, too, should be a vector in $\mathbb{R}^{4096}$. Reshape this vector using code like the above, and visualize it. You should see something like a face with eyeglasses. That is, this eigenvector is an "eyeglasses detector".

In your submission, show your code and the resulting image.

b) Take the dot product of eigenvector $\vec{u}^{(7)}$ with every image in the data set. Plot the 20 images whose dot product with the eigenvector is the largest in absolute value. We will grade your answer to this problem manually by inspecting your plots.

If you did everything right, you should see a lot of eyeglasses.

Again, for this problem you should show the resulting images and your code.