DSC 190 - "Super Homework"

Due: Wednesday, June 8

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.

On Midterm 02, there was a question along the following lines. Suppose you have a data set of points X in $\mathbb{R}^{100}$ and wish to use PCA to reduce the dimensionality to 50. Consider these two approaches:

- Approach 1: Run PCA once to go directly from $\mathbb{R}^{100}$ to $\mathbb{R}^{50}$, constructing a new data set Z_1.
- Approach 2: First run PCA with $k = 75$ to create an intermediate data set Z' of points in $\mathbb{R}^{75}$, then run PCA with $k = 50$ on Z' to create a new data set Z_2.

Is there any difference between the two approaches?

The correct answer is: no, there is not. That is, $Z_1 = Z_2$. The exam's answer key gave an intuitive geometric explanation of the answer – here we will derive it more rigorously.

In this problem, assume that X is an $n \times d$ matrix of n data points in $\mathbb{R}^d$; furthermore, assume the data are centered. Let C be the covariance matrix of the original data. Let C' be the covariance matrix of Z' (the intermediate data in approach #2). Let U_{75} be a 100×75 matrix consisting of the top 75 eigenvectors of C, and let U_{50} be a 100×50 matrix consisting of the top 50 eigenvectors of C. Then the new PCA features in approach 1 are $Z_1 = XU_{50}$, and the intermediate PCA features in approach 2 are $Z' = XU_{75}$.

Throughout this problem you may assume for simplicity that all eigenvalues are unique.

a) Recall that C' is the covariance matrix of Z', the intermediate data in approach #2. Show that C' is a diagonal matrix.

Hint: $C' = (Z')^T Z'$. Also remember that for general matrices AB, $(AB)^T = B^T A^T$.

b) The data set Z_2 is computed by multiplying the intermediate data set Z' by a 75×50 matrix U' consisting of the top 50 eigenvectors of the covariance matrix C'.

Argue that U' is the matrix where entry $u'_{ii} = 1$ and all other entries are zero. That is, it is a kind of rectangular identity matrix.

c) Using what we have learned above, show that $Z_2 = XU_{50}$, and is therefore equal to Z_1.

Hint: $Z_2 = Z'U'$. Start by substituting for both U' and Z'.

Problem 2.

As a data scientist you will have the opportunity to work on problems that are of great importance to society. This is not one of those problems.

The *menu-match* dataset consists of 646 images of food from three different restaurants: an Asian restaurant, an Italian restaurant, and a soup restaurant. The data set was constructed by employees of Microsoft Research[1].

The data set is available at the following link:

[1] `http://neelj.com/projects/menumatch/`

`https://f000.backblazeb2.com/file/dsc-data/menu-match.npz`

The file is in compressed numpy format; it can be loaded with 'np.load'. Once loaded, it behaves like a dictionary with four keys: `X_train`, `X_test`, `y_train`, and `y_test`, corresponding to the training data, test data, training labels, and test labels, respectively. For example, to get the training data:

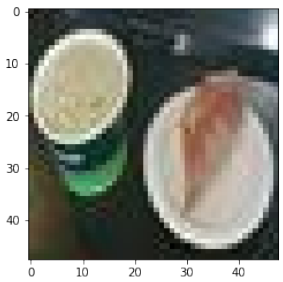
```
>>> data = np.load("menu-match.npz")
>>> data['X_train']
```

The training data is a `480 x 48 x 48 x 3` tensor, since it consists of 480 images, each `48 x 48` pixels with 3 colors.

Note that the given labels are strings: "a" for Asian restaurant, "i" for Italian, and "s" for soup.

You may display an image with `matplotlib` using `plt.imshow`. For example:

```
>>> # plot test example #30
>>> plt.imshow(data['X_test'][30])
```



Using tensorflow, train a convolutional neural network to predict whether a given image is from the Asian restaurant or not (thus turning the problem into a binary classification problem). You will need to determine the network architecture, but the example code given in lecture and in the discussion is a good starting point. Report the test accuracy – your model should be able to get above 70% of the test set correct (preferably higher!). Show your code.