**Math 160 (De Loera)** **Homework 2**
**Due date: May 2, 2016**

## INSTRUCTIONS

**All homeworks will have many problems, both theoretical and practical. Programming exercises need to be submitted via SMARTSITE using the assignment boxes. Other methods of submission without prior approval will receive zero points.**

**Write legibly preferably using word processing if your hand-writing is unclear. Be organized and use the notation appropriately. Show your work on every problem. Correct answers with no support work will not receive full credit.**

1. Classification of images via their SVD decomposition properties

   You will write an algorithm in MATLAB for the classification of handwritten digits. For this you can download the following files from `https://www.math.ucdavis.edu/~deloera/TEACHING/MATH160/guessdigit-project.zip`

   NOTE: Data comes in compressed form, to open you type (after downloading), unzip guessdigit-project.zip. Inside the directory that will open up you will find 5 files:

```
ima2.m        -- Code Displays an image vector in the right orientation

azip.mat        -- the matrix of training digits data

dzip.mat        -- command dzip(i) tells you the (correct) digit you have in column
                           i of the matrix azip

dtest.mat       -- tells you the (correct answer) of test digits

testzip.mat     -- the test digits data
```

   Use the training set, and compute the SVD of each class matrix (classes are those matrices that represent the same digit). Use the first few (5, 10, 20) singular vectors as basis of a class and classify unknown test digits according to how well they can be represented in terms of the respective bases (use the relative residual vector in the least squares problem as a measure). Here are some specific tasks.

   - Write your code to do classification, it brakes the training data in classes, computes the SVD of each class and uses that to make predictions. It takes in a test data point and makes a prediction.
   - Tune the algorithm for accuracy of classification. Give a table or graph of the percentage of correctly classified digits as a function of the number of basis vectors. Graph the situation for 5, 10, 20 basis vectors. Display the results in a table (or tables).

- Check the singular values of the different classes. Is it reasonable to use different numbers of basis vectors for different classes? If so, perform a few experiments to find out if it really pays off to use fewer basis vectors in one or two of the classes (i.e., do you get different/same outcome?).
- Check if all digits are equally easy or difficult to classify. Also look at one of the difficult ones, and see that in many cases they are very badly written. What is the most difficult digit to read for the computer? Does it help to increase the number of singular vectors you used? Write comments at the very end of your code with your thoughts.

2. Let $A$ be an $m \times n$ matrix and $x \in R^n$, and $b \in R^m$ vectors. Prove that the set $P = \{x : Ax \leq b\}$ is a convex set.

3. Let $C$ be a nonempty subset of $R^n$, and let $\lambda_1$ and $\lambda_2$ be positive scalars. Show that if $C$ is a convex set, then $(\lambda_1 + \lambda_2)C = \lambda_1 C + \lambda_2 C$. Show by example that this need not be true when $C$ is not convex.

4. Show that the image and the inverse image of a cone under a linear transformation is a cone. Show that a subset $C$ is a convex cone if and only if it is closed under addition and positive scalar multiplication.

5. Show that for $x, y$ positive scalar real numbers $ye^{x/y} = \max_{a>0} a(x+y) - y \cdot a \cdot ln(a)$. Use this to prove that the function $ye^{x/y}$ is convex inside the positive orthant. Let $f(x) = \ln(e^{x_1} + \ldots e^{x_n})$. Is this convex?

6. In this problem you need to test whether the following functions are convex or not:

   - The function $s_k : R^n \to R$ which is defined as $s_k(x) = \sum_{i=1}^{k} x_{[i]}$ where $x_{[i]}$ is the $i$-th largest component of the vector $x$. HINT: Explore what happens with examples $n = 3, k = 2$.
   - For $n = 2k - 1$ odd Consider the function $\phi : R^n R$ with

     $$\phi(x) = \frac{1}{n} \sum_{i=1}^{n} |x_i - med(x)|$$

     where $med(x)$ is the median of the components of $x$. HINT: Use $s_k$.
   -

7. You need to write a SCIP model to solve the following challenge:

   Your problem is packing $m$ of spheres in a box of minimal area. The spheres have a given radius $r_i$, and the problem is to determine the precise location of the centers $x_i$. The constraints in this problem are that the spheres should not overlap, and should be contained in a square of center 0 and half-size R. The objective is to minimize the area of the containing box.

   - Show that two spheres of radius $r_1$, $r_2$ and centers $x_1$, $x_2$ respectively do not intersect if and only if $||x_1 - x_2||_2$ exceeds a certain number, which you will determine.
   - Formulate the sphere packing problem as an optimization problem. Write it in SCIP. Is the formulation you have found convex? Can you solve it for 5 spheres being packed (you can choose sizes of the spheres)?

8. Using MATLAB, do the following clustering or K-means experiments. (**important** if you have a personal copy of MATLAB on your laptop computer, MATLAB k-means algorithm may not be accessible because it is a part of the statistics toolbox, which your copy may not include. Run help kmeans in your MATLAB session. Then you can see whether your MATLAB has this kmeans function. **If not, you need do this experiments in our computer room.**)

   You need to download the *breastcancer.mat* from `https://www.math.ucdavis.edu/~deloera/TEACHING/MATH160/breastcancer.mat` load this file into your MATLAB. This file contains a matrix called data of size $683 \times 11$, which represents 9 measurements (various geometric features) taken from images of 683 cells in breasts. $data(j, 1)$ is the Id. number of the jth cell, data(j,2:10) contains those 9 measurements of the jth cell, and data(j,11) contains the diagnosis by the doctors, i.e., benign if this value is 2; malignant if this value is 4.

   **(a)** Run the K-means algorithm kmeans in MATLAB with K = 2.

```
>> idx = kmeans(data(:,2:10), 2);
```

   The output idx contains the cluster number of each data point, and since you used K = 2, the cluster value of the jth cell, i.e., idx(j) is either 1 or 2. Create now the table of classification results like the one shown below:

| | M | B |
|---|---|---|
| M | 222 | 17 |
| B | 9 | 435 |

In the above table, 222 malignant cells were correctly classified as "malignant',' but 17 malignant cells were erroneously classified as "benign." On the other hand, 435 benign cells were correctly classified as "benign," but 9 benign cells were erroneously classified as "malignant."

[Hint: `2*idx` gives you the array whose entries are either 2 or 4 so that that is easier to compare with the ground truth diagnosed by the doctors. However, one thing you need to pay attention to is that the value 2 in `2*idx` may or may not correspond to the "benign" class because $K$-means algorithm with $K$ outputs just two clusters without any label information. It is up to you to decide which cluster should match the "benign" class and which cluster should match the "malignant" class. You should pick the matching/labeling that gives you lower misclassification rates.]

**(b)** Compute the *false-positive rate* and *false-negative rate* of your classification results. These are defined as:

$$\text{false-positive rate} := \frac{\text{Number of benign cells incorrectly classified as "malignant"}}{\text{Total number of benign cells}}$$

$$\text{false-negative rate} := \frac{\text{Number of malignant cells incorrectly classified as "benign"}}{\text{Total number of malignant cells}}$$

Hence if the table shown above is your result, then these rates are:

$$\text{false-positive rate} = \frac{9}{444} = 0.0203.$$
$$\text{false-negative rate} = \frac{17}{239} = 0.0711.$$

**(c)** Repeat the above classification experiments using $K$-means algorithm 9 more times. Then, generate the classification table of $2 \times 2$ consisting of the average rates of these 10 experiments (the first one in Part (a) and these 9 experiments in Part (c)). Then, compute the average false-positive and false-negative rates of these 10 experiments. Report these results.

9. **Using Math to decide the ranking of an event:** An exam with $m$ question is given to $n$ students of MATH 16000. The instructor collects all grades in an $n \times m$ matrix $G$, with $G_{ij}$ the grade obtained by student $i$ on question $j$. The professor would like to assign a *difficulty score* to each question based on the available data, rather than use the subjective perception of students.

   From the theory of SVD's we know $G$ can be decomposed as a sum of rank-many rank one matrices. Suppose that $G$ is well approximated by a rank one matrix $sq^T$ with $s \in R^n$ and $q \in R^m$ with non-negative components. Can you use this fact to give a difficulty score? What is the possible meaning of the vector $s$?