Homework 2 - Due Mon. Feb. 13th
AS.110.446, EN.550.416

Dr. Mauro Maggioni — **Office**: 405 Krieger Hall
Web page: www.math.jhu.edu/˜ mauro — **E-mail**: myfirstname.maggioni@jhu.edu

Homework Policies

As in the first homework set.

Exercises

Exercise 1 (10pts). We work in $\mathbb{R}^n$, and fix a vector $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_+$, i.e. with $\alpha_1, \ldots, \alpha_n > 0$.

- Show that
$$\langle v, w\rangle_\alpha = \sum_{i=1}^{n} \alpha_i v_i w_i$$
defines an inner product on $\mathbb{R}^n$.
- Find an $n \times n$ matrix A_α (that depends on α) such that you may write the inner product above as $v^T A_\alpha w$ (thinking of vectors as column vectors).
- What is the norm $|| \cdot ||_\alpha$ associated with this inner product?
- Draw a sketch of the unit sphere $\{||v||_\alpha = 1\}$ corresponding to this norm, for some fixed α and for, say, $n = 2$.
- (extra question, no credit) What happens if at least one of the α_i's is 0?

Exercise 2 (25pts). An $n \times n$ matrix A is called positive-definite if A is symmetric and $\langle Av, v\rangle \geq 0$ for all $v \in \mathbb{R}^n$, and $\langle Av, v\rangle = 0$ implies $v = 0$ (here $\langle \cdot, \cdot\rangle$ is the standard Euclidean inner product).

- Give at least 3 non-trivial and significantly different examples of 2×2 positive definite matrices.
- Was the matrix A_α in the previous exercise positive definite?
- If A is a positive definite $n \times n$ matrix, define $\langle v, w\rangle_A := \langle Av, w\rangle$ (where $\langle \cdot, \cdot\rangle$ is the standard Euclidean inner product). Show that $\langle \cdot, \cdot\rangle$ is an inner product in $\mathbb{R}^n$.

Exercise 3 (15pts). Let $A = \begin{bmatrix} 2 & 4 \\ 0 & 4 \end{bmatrix}$. Compute $||A||_1$, $||A||_2$, $||A||_\infty$.

Exercise 4 (50pts). See the script `Homework2.m` below, also available on the Wiki, also included below: follow the instructions in the script, completing it. Note the questions marked with ?????? and do Part I and Part II. See the wiki for the code (which requires a few additional scripts to load the data). Please remember to include a print out of your code, figures, etc... as per homework policies.

```
%% Add path to data set and helpers to load data.
% Change to the appropriate directory
addpath('../Data');

%% Load all handwritten digits in a data set
[X,Y]       = LoadMNIST;
nClasses    = length(unique(Y));

%% Let's look at some random data points, visualized as images
idxs        = randperm(size(X,2),32); ...
                                                                                 ...
    % Picks 32 random numbers between 1 and N=number of points
bigFig;
for k = 1:length(idxs)
    subplot(ceil(length(idxs)/8),8,k);
    imagesc(reshape(X(:,idxs(k)),[28,28])); ...
                                                                                 ...
        % Each column vector in X is a 28x28 image
    colormap(gray);
end

%% Compute the mean of the data
Xmean    = mean(X,2);% ??????      % Compute the mean
% ?????? Display the mean
Xsubmean = bsxfun(@minus,X,Xmean);                               % Subtract the mean to ...
    obtain centered data. NOTE: what's the computational complexity of this command? ...
    Would you have computed Xsubmean in other ways? Would they be faster?
% Let's look at some random digits after subtracting the mean: make sure you ...
    understand why the images lookk like the way they do
idxs = randperm(size(Xsubmean,2),32);
bigFig;
for k = 1:length(idxs)
    subplot(ceil(length(idxs)/8),8,k);
    imagesc(reshape(Xsubmean(:,idxs(k)),[28,28]));
    colormap(gray);
end
set(gcf, 'Name', 'Centered Data');

%% Principal Component Analysis
[U,S,V] = svd(Xsubmean);                                         % ?????: This will ...
    fail: why? If it does not fail, you will wait for a very large time...
[U,S,¬] = svd(Xsubmean,'econ');                                  % ?????: Run this ...
    instead: how is it different from the above?
% Displays some of the singular vectors
bigFig;h1=gcf;
scrsz = ...
    get(groot,'ScreenSize');h2=figure('Position',[scrsz(3)*(1/8+3/4-3/24),scrsz(4)*(3/4),scrsz(3)*3/24,scrsz(
for k = 1:10
    Xproj = U(:,k:k+2)'*X;
    figure(h1);
    scatter3(Xproj(1,:),Xproj(2,:),Xproj(3,:),10,Y,'filled');colorbar;
    xlabel(num2str(k));ylabel(num2str(k+1));zlabel(num2str(k+2));
    figure(h2);                          % NOTE: this helps in understanding the ...
        coordinates being used
    for l = 1:3
        subplot(1,3,l); imagesc(reshape(U(:,k+l-1),[28,28]));colormap(gray);
    end
    pause;
end

%% Part I : Project onto singular vectors
% - For each value of k=1,2,3,4,8,16,32,64,128
% - Project the (centered) data onto the first k singular vectors ...
    {U(:,1),U(:,2),...,U(:,k)}
```

Figure 1: Matlab code of `Homework2.m`

```
% - Compute the length of each centered data point, the length of its projection onto ...
    the first k singular vectors, and the
%   length of the difference between the original point and this projection. What is ...
    the relationship between these 3 lengths?
% - Produce a figure with three rows, where in the first row you show some of an ...
    original data point, the second row shows the
%   same data point projected onto the first k singular vectors plus the mean, and the ...
    third row shows the difference between the original
%   data point and its projection in column two. Do this for several (random) data ...
    points, and for several values of k. Comment
%   on how the approximations and the errors look like, how they decrease, for which ...
    value of k the approximated digits become recognizable, ...
% - Plot the singular values (on the diagonal of S above). You may want to switch to ...
    log_{10} scale of the singular values. What
%   do they tell you about the error (measured how) incurred by approximating the data ...
    by k singular values?




%% Part II : Principal Component Analysis for each class
% - Pick two classes, e.g. {1,3}
% - Compute PCA for the data in the first class, yielding principal vectors U_1 and ...
    principal values S_1, and separately for
%   the second class, yielding principal vectors U_2 and principal values S_2.
% - Plot S_1 and S_2 on the same plot, comment on their appearance, and try to provide ...
    an explanation of similarities/differences
% - Compare the mean squared error for:
%           approximating points in class 1 with their projection onto U_1(:,1:k)?
%           approximating points in class 2 with their projection onto U_1(:,1:k)?
%           approximating points in class 1 with their projection onto U_2(:,1:k)?
%           approximating points in class 2 with their projection onto U_2(:,1:k)?
% - Can you design a simple rule for assigning a point (of unknown class) to either ...
    class 1 or class 2 based on how well it i
%   approximated by its projection onto U_1 or U_2? (be careful and don't forget about ...
    the appropriate means to subtract)
% - How do the results above change upon changing k?
% - Think about and describe all of the above geometrically
```

Figure 2: Matlab code of `Homework2.m`