Assignment #11: Least Squares, Diagonalization of Symmetric Matrices, Quadratic Forms, Optimization, Singular Value Decomposition

Due date: Friday, May 6, 2016 (10:10am)

Name: ____________________________________________

Section Number
Assignment #11: Least Squares, Diagonalization of Symmetric Matrices, Quadratic Forms, Optimization, Singular Value Decomposition

Due date: Friday, May 6, 2016 (10:10am)
For full credit you must show all of your work.

1. Find the least-squares quadratic that best fits the 4 points \((2, 2), (-2, 8), (1, 0), (-1, 4)\). What is the least squares error in this approximation?

2. Let \(A = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 2 & 0 \\ -1 & 0 & 1 \end{bmatrix}\).

   a. Write out the characteristic equation of \(A\). Solve for the roots of this equation.

   b. Explain how, without doing any computation, you can be certain of the dimensions of each of the eigenspaces corresponding to each of the eigenvalues of \(A\). What are these dimensions?

   c. Working by hand, find the eigenspace corresponding to each of the eigenvalues of \(A\).

   d. Express \(A\) as the product \(QDQ^T\), where \(D\) is diagonal and \(Q^T = Q^{-1}\).

3. Suppose \(B = \begin{bmatrix} 3 & 0 & -1 \\ 0 & -2 & 0 \\ -1 & 0 & 3 \end{bmatrix}\).

   a. Is \(B\) invertible?

   b. Use Matlab to compute the orthogonal diagonalization of \(B\).

   c. Express \(B\) as the weighted sum of three matrices \(M_1, M_2, M_3\), where \(M_1M_2 = 0, M_1M_3 = 0, M_2M_3 = 0\), and \(M_iM_i = M_i\) for \(i = 1, 2, 3\).

4. Show that for any vector \(v\) in \(\mathbb{R}^3\), the \(3 \times 3\) matrix \(vv^T\):

   i. is symmetric

   ii. has rank = 1

   iii. projects \(x\) onto the subspace of \(\mathbb{R}^3\) spanned by \(v\). Specifically: \(vv^T(x) = (x \cdot v)v\).

   b. Show that for any unit vector \(u\) in \(\mathbb{R}^3\), the \(3 \times 3\) matrix \(uu^T\) has \(u\) as an eigenvector, with the corresponding eigenvalue = 1.

   c. Show that for any \(m \times n\) matrix \(A\), the matrix products \(A^TA\) and \(AA^T\) are symmetric.

5. Construct the matrix of the quadratic form \(Q(x) = 7x_1^2 + 12x_1x_2 - 2x_2^2\) and classify the quadratic form. Then, use a change-of-variable to express the quadratic form without using any cross-product terms. What are the principal axes of this quadratic form?
6. For what values of $a$ will the matrix $A = \begin{bmatrix} a & 2 \\ 2 & 1 \end{bmatrix}$ be positive definite? Are there any values of $a$ for which this matrix will be negative definite? Please explain.

7. For what values of $b$ will the matrix $B = \begin{bmatrix} 1 & b \\ b & -2 \end{bmatrix}$ be negative definite? Are there any values of $b$ for which this matrix will be indefinite? Please explain.

8. a. Prove that for any arbitrary $m \times n$ matrix $A$, the matrix $A^T A$ is guaranteed to be positive semi-definite.

   b. Under what conditions will the product $A^T A$ of an arbitrary $m \times n$ matrix $A$ fail to be positive definite? (Characterize the matrices $A$ for which this outcome will occur.)

9. What is the range $\{\min, \max\}$ of the values that can be taken by the quadratic form $Q(x)$ of the matrix $A = \begin{bmatrix} 7 & -4 & -2 \\ -4 & 1 & -4 \\ -2 & -4 & 7 \end{bmatrix}$? At what unit vector(s) $x$ will $Q(x)$ take its maximum value? At what unit vector(s) $x$ will $Q(x)$ take its minimum value? Find a vector $x$ for which $Q(x) = 0$. (Hint: consider the relative distance between the minimum and maximum eigenvalues of $A$ and interpolate accordingly between vectors in the associated eigenspaces.)

10. Let $A = \begin{bmatrix} 1 & 2 & 0 \\ 0 & -2 & 1 \end{bmatrix}$. $A$ is a linear transformation that maps vectors $x$ in $\mathbb{R}^3$ into vectors $b$ in $\mathbb{R}^2$. Consider the set of all possible vectors $b = Ax$, where $x$ is of unit length. What is the longest vector $b$ in this set, and what unit length vector $x$ is used to obtain it?

11. Working by hand, compute the singular value decomposition of the following matrices. (This will be good practice for the final exam).

   a. $A = \begin{bmatrix} 2 & 3 \\ 4 & 6 \end{bmatrix}$  
   b. $B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix}$  
   c. $C = \begin{bmatrix} 2 & 1 & -2 \\ 2 & -2 & 1 \end{bmatrix}$

12. The singular value decomposition of an $m \times n$ matrix $A = U \Sigma V^T$ can be expressed as a sum of $r$ distinct matrices $B_1 \ldots B_r$, where $r = \min(m, n)$ and $B_i = u_i \sigma_i v_i^T$, where $u_i$ is the $i^{th}$ column of $U$, $\sigma_i$ is the $i^{th}$ diagonal element of $\Sigma$ and $v_i^T$ is the $i^{th}$ row of $V$. If we treat an image as an $m \times n$ matrix, and decompose it using a singular value decomposition, then the sum of all of the components would be equal to the original image. But what is the result if we take the sum of only the first few components? This Matlab exercise provides an opportunity for you to explore that question. Please follow the steps below:

   a. Use the command $A = \text{imread('testpat1.png')}$; to load the cameraman image into a matrix $A$. $A$ will have dimensions 512 x 512, and its entries will be 8-bit unsigned integers.
b. Use the command
\[ [U, S, V] = \text{svd(double(A))}; \]
to obtain the singular value decomposition of \( A \), where \( A = USV^T \). Note that you have to convert the values of \( A \) from \textit{int} to \textit{double} before you can to apply Matlab’s \text{svd()} function to it.

c. In a loop of increasing values of \( k \), for example \( k = 1:2:100 \), reconstruct a “reduced” version of \( A \) using only the first \( k \) components of the singular value decomposition. Specifically, you need to form the product \( a = usv^T \) where \( u \) is a \( 512 \times k \) matrix containing the first \( k \) columns of \( U \), \( s \) is a \( k \times k \) matrix containing the first \( k \) diagonal values in \( S \), and \( v^T \) is a \( k \times 512 \) matrix containing the first \( k \) rows of \( V^T \). You can use a command like: \( u = U(:, i) \) to extract the \( i \)th column from \( U \), and a command like: \( u = U(:, i:j) \) to extract a range of columns between the \( i \)th and the \( j \)th. Be careful to extract the correct portions of each component matrix. In your loop, use the commands \text{figure} \ and \text{imshow(uint8(a))} \ to display the series of reconstructed images. (You need to convert the values of the matrix \( a \) back to 8-bit unsigned integers before they can be displayed as an image.)

d. Report your impressions. How many components do you need to use before the image becomes recognizable? After how many components do you stop seeing much difference in the result?

e. Extract and plot the singular values in matrix \( S \). You can do this using the command:
\[ \text{plot(1:length(diag(S)),diag(S));} \]
Do you notice a relationship between the sizes of the singular values and the pace of improvement in the quality of the reduced image?

Please pass in: a printout of your code; your answers to the questions above, and one of the reduced images that you obtained, along with an annotation of the number of components you used to construct it (the value \( k \)).

13. Consider the table of data below:

<table>
<thead>
<tr>
<th>input</th>
<th>1</th>
<th>5</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>3</td>
<td>11</td>
<td>6</td>
<td>8</td>
<td>15</td>
<td>11</td>
</tr>
</tbody>
</table>

a. Normalize this data by converting it to mean-deviation form, then construct the sample covariance matrix. What is the variance of \textit{input}? of \textit{output}? What is the total variance?

b. Find the principal components of this data. (You can use Matlab; please round your answer to 4 digits after the decimal point.)

c. Define a new variable “efficiency” that defines a linear relationship between input and output: \( \text{efficiency} = (x)\text{input} + (y)\text{output} \), with \( x^2 + y^2 = 1 \), such that \textit{efficiency} describes most of the variance in the data.

d. How much of the total variance in the data is explained by \textit{efficiency}?

e. Use the command \text{scatter(input, output)} to plot the data in Matlab. Define two endpoints \( p1 = (xmin, y(xmin)) \), \( p2 = (xmax, y(xmax)) \) along the orthogonal regression line determined by the first principal component of the data and use the sequence of commands \text{hold on} \ and \text{plot(x, y)} \ [where \( x \) is the vector of \( x \)-coordinates and \( y \) is the vector of \( y \)-coordinates of the line to be plotted] to superimpose the regression line onto the data.

f. Compute the best-fitting line to the normalized data using the least-squares method and add it to the plot. Explain how these two different lines can both represent a “best fit” to the data.