ECE 497-3: Inverse Problems in Engineering
Homework #2

Due: Friday December 20, 2002

Motivation

In this homework, we will look at the singular value decomposition of a matrix as a method for examining least squares problems. In this homework, the attatched Matlab routines will do most of the work. You will need to set the "numeric format" (in the options window) to "long" for this homework.

1) Consider the following linear system of equations

$$\underline{y} = \begin{bmatrix} -0.2042 & 0.0559 & 0.3715 & -0.2077 \\ 0.1347 & 0.2208 & 0.5294 & 0.1336 \\ 0.4803 & 0.3889 & 0.6862 & 0.4670 \\ 0.6078 & 0.6088 & 1.2159 & 0.6081 \end{bmatrix} \underline{x}$$

Using Matlab, compute $\underline{y}$ for $\underline{x}^T = [1\ 1\ 1\ 1]$. Then, using this $\underline{y}$, compute $\underline{x}$ by determining the inverse of the above matrix.

It turns out that for *any* matrix $\mathbf{Z} \in R^{m\times n}$ we can always compute its *singular value decomposition*

$$\mathbf{Z} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

where, $\mathbf{U} \in R^{m\times m}$ and $\mathbf{U}$ is a *unitary* matrix ($\mathbf{U}^T = \mathbf{U}^{-1}$), $\mathbf{V} \in R^{n\times n}$ and $\mathbf{V}$ is a unitary matrix ($\mathbf{V}^T = \mathbf{V}^{-1}$), and $\mathbf{S} \in R^{m\times n}$. $\mathbf{S}$ is all zeros except for a submatrix, which will will denote Σ, of size $m \times m$ or $n \times n$ (depending on whether m or n is smaller) in the upper left corner. The entries in the submatrix Σ are called the *singular values* of the matrix and will b e denoted σ_i for the i^{th} singlular value. In general, $\sigma_i \geq \sigma_{i+1}$, that is, the singular values decrease as the mode number increases (this is important!) As you will see, these singular values contain a great deal of information about the matrix (and what may be going wrong).

2) Lets assume that

$$\begin{aligned} \underline{x} &= \mathbf{V}\underline{\alpha} \\ \underline{y} &= \mathbf{U}\underline{\beta} \end{aligned}$$

That is, $\underline{x}$ can be written as a linear combination of the columns of $\mathbf{V}$

$$\begin{aligned} \underline{x} &= \mathbf{V}\underline{\alpha} \\ &= [\underline{v}_1\ \underline{v}_2\ \ldots\ \underline{v}_n]\ \underline{\alpha} \\ &= \underline{v}_1\alpha_1 + \underline{v}_2\alpha_2 + \ldots\ + \underline{v}_n\alpha_n \end{aligned}$$

Since $\underline{x} \in R^n$ and the columns of $\mathbf{V}$ span R^n, we know such an expansion must exist. Similarly, we must be able to represent $\underline{y}$ as a linear combination of the columns of $\mathbf{U}$.

Now assume we have our linear problem $\underline{y} = \mathbf{Z}\underline{x}$.

Show that the singular value decomposition leads to

$$\underline{\alpha} = \mathbf{\Sigma}^{-1}\underline{\beta}$$

(Note, if there are more α_i than β_i, the extra α_i are set to zero. This just means that $\mathbf{V}$ spans a larger space than $\mathbf{U}$. Similarly if there are more β_i than α_i, the extra β_i are set to zero. This just means $\mathbf{U}$ spans a larger space than $\mathbf{V}$.) This can be written in terms of components as

$$\alpha_i = \beta_i/\sigma_i$$

You may assume that $\mathbf{S}$ *has full rank (.e.,* $\mathbf{S}^{-1}$ *exists) for this derivation, even if it may not.*

3) Now assume we have an overdetermined least squares solution

$$\underline{\hat{x}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\underline{y}$$

with the assumptions above, and writing $\mathbf{Z}$ in terms of its singular value decompostion, show that the solution of the overdetermined least squares problem

$$\underline{\hat{x}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\underline{y}$$

again leads to

$$\underline{\alpha} = \mathbf{\Sigma}^{-1}\underline{\beta}$$

or

$$\alpha_i = \beta_i/\sigma_i$$

You may assume that $\mathbf{S}$ *has full rank (.e.,* $\mathbf{S}^{-1}$ *exists) for this derivation, even if it may not.*

4) Now we need to get either the $\underline{\alpha}$, or, equivalently, the $\underline{\beta}$. The answer is fairly simple. Since we observe $\underline{y}$ and $\mathbf{U}$ is a unitary matrix, you should be able to show that

$$\underline{\beta} = \mathbf{U}^T\underline{y}$$

5) Now compute $\underline{x}$ using the singular value decomposition of $\mathbf{Z}$. (The Matlab routine will do this for you, it should be the same as in (1).)

6) Now assume we have a slight error in our measurement, a very likely thing in the "real" world. The Matlab routine will add a small amount of noise to the observation vector. It should usually change the $\underline{y}$ values by a few percent. Determine the new $\underline{y}$ when noise is added, and compute a new estimate of $\underline{x}$.

7) Are the estimates in (1) and (6) above close? Do these answers seem strange? **Run your program at least three times, with three different sets of noise and turn them**

in!

8) Now lets look at our orignal problem and what is going wrong. If there was no noise, then we would have

$$\begin{aligned} \underline{y} &= \beta_1 \underline{u}_1 + \beta_2 \underline{u}_2 + \beta_3 \underline{u}_3 + \beta_4 \underline{u}_4 \\ \underline{x} &= \frac{\beta_1}{\sigma_1}\underline{v}_1 + \frac{\beta_2}{\sigma_2}\underline{v}_2 + \frac{\beta_3}{\sigma_3}\underline{v}_3 + \frac{\beta_4}{\sigma_4}\underline{v}_4 \end{aligned}$$

However, since there is measurement error in $\underline{y}$, which we will define as $\underline{y} + \delta\underline{y}$, we have instead of the above,

$$\begin{aligned} \underline{y} + \delta\underline{y} &= (\beta_1 + \delta\beta_1)\underline{u}_1 + (\beta_2 + \delta\beta_2)\underline{u}_2 + (\beta_3 + \delta\beta_3)\underline{u}_3 + (\beta_4 + \delta\beta_4)\underline{u}_4 \\ \underline{x} + \delta\underline{x} &= \frac{\beta_1 + \delta\beta_1}{\sigma_1}\underline{v}_1 + \frac{\beta_2 + \delta\beta_2}{\sigma_2}\underline{v}_2 + \frac{\beta_3 + \delta\beta_3}{\sigma_3}\underline{v}_3 + \frac{\beta_4 + \delta\beta_4}{\sigma_4}\underline{v}_4 \end{aligned}$$

or, the deviation from the true answer $\underline{x}$ is

$$\delta\underline{x} = \frac{\delta\beta_1}{\sigma_1}\underline{v}_1 + \frac{\delta\beta_2}{\sigma_2}\underline{v}_2 + \frac{\delta\beta_3}{\sigma_3}\underline{v}_3 + \frac{\delta\beta_4}{\sigma_4}\underline{v}_4$$

Now, since by construction, the magnitude of the $\underline{v}_i$ are one, and the errors in the observation are assumed to be reasonably small, we need to look at the sizes of the singular values σ_i. As they get smaller, they will **amplify** the errors $\delta\beta_i$ more strongly. For our example problem, determine the ratios $\delta\beta_i/\sigma_i$.

9) Now what do we do with this newfound information? What we will try is called *Truncated singular value decomposition*, and you do what the name implies. That is, only use the eigenvector expansion for the first few terms (before the singular values become small), and set the coefficients for all of the higher terms to zero. (Of course, since in general one does not often actually know what the true answer is, this requires some experience and a close examination of the relative sizes of the singular values.)

For our problem, compute $\underline{x}$ omitting first the last term (i.e. the one with the smallest singular value), the last two terms, and the last three terms. Do your answers get better than using all of the terms?

(10) Repeat the above steps (1,5,6,7,8,9) for the problem

$$\underline{y} = \begin{bmatrix} -0.0505 & 0.0683 & 0.1175 & 0.0538 & 0.1709 \\ 0.2215 & 0.1729 & 0.0005 & 0.2794 & 0.2963 \\ 0.4663 & 0.4675 & -0.0623 & 0.4237 & 0.3403 \\ -0.1871 & 0.0701 & 0.2650 & 0.0460 & 0.3136 \\ 0.2109 & 0.0203 & 0.5130 & -0.0603 & 0.0717 \end{bmatrix} \underline{x}$$

<u>To Hand In</u>: Hand in your code for both parts (both matrices) as well as the outputs (3 outputs for each case). Also include some **discussion** of your results (just a paragraph). You will be partially graded on neatness and organization! **Think** about your results. We will be using SVD more in the future.