Assignment 4: Linear Least Squares Problems

ACM 106a: Introductory Methods of Computational Mathematics (Fall 2013)

Due date: Thursday, October 31, 2013

1 The Pseudoinverse

Suppose that $A \in \mathbf{R}^{m\times n}$ with $m > n$ and full column rank ($\text{rank}(A) = n$). Prove the following properties of the pseudoinverse $A^\dagger = (A^TA)^{-1}A^T$:

(a) $X = A^\dagger$ minimizes

$$\min_{X\in\mathbf{R}^{n\times n}} \|AX - I\|_F$$

Hint: try using the definition of the Frobenius norm of M as the square root of the sum of the squares of the vector 2-norms of the columns of M to decompose the matrix least squares objective as the sum of n linear least squares objectives.

(b) $(AA^\dagger)A = A$

(c) $A^T(AA^\dagger) = A^T$.

2 The underdetermined problem

Suppose that $A \in \mathbf{R}^{m\times n}$ such that $m < n$. Suppose further that A has full row rank ($\text{rank}(A) = m$). Recall that the underdetermined linear system $Ax = b$ admits an $(n-m)$-dimensional subspace of solutions. Show that the minimum 2-norm solution of $Ax = b$ is equal to

$$x = A^\dagger b$$

where $A^\dagger = A^T(AA^T)^{-1}$.

Hint: The matrix A will have a nontrivial null space because it does not have n linearly independent columns. Any solution x can be written in the form $x = x_1 + x_2$ where $x_1 \in \text{Col}(A)$ and $x_2 \in \text{Null}(A)$. Using either the QR decomposition or SVD of A^T you can derive an analytic expression for x_1 and the subspace of solutions. Choosing the minimum norm solution from this subspace completes the proof.

3 Polynomial approximation

In this problem, we will fit polynomials to the function $f(t) = \sin(4t)$ in the interval $[0, 1]$. To generate your observations, evaluate $b_i = \sin(4t_i)$ at 51 equally spaced t_i on the interval $[0, 1]$; the vector of inputs t can be generated using the Matlab function "linspace" and the vector of observations is given by $b = \sin(4 * t)$ (sine and scalar multiplication are already vectorized). Recall from class that the coefficients x of the degree-$(m-1)$ polynomial best approximating our "unknown" function (with respect to the 2-norm) are given by the solution of the linear least squares problem

$$x = \arg\min_{x\in\mathbf{R}^m} \|Ax - b\|_2,$$

where $A \in \mathbf{R}^{n\times m}$ is the matrix with (i, j)th entry equal to $A_{ij} = t_i^{j-1}$.

Compute the coefficients of the polynomials of degree $m - 1$ for (i) $m = 3$, (ii) $m = 7$, and (iii) $m = 12$ approximating $f(t) = \sin(4t)$ by solving the least squares problem $\|Ax - b\|_2$ using each of the following methods:

(a) Solving the normal equations using a Cholesky factorization.

(b) Using a Householder-based QR factorization of A.

You may use the Matlab command "chol" to obtain the necessary Cholesky decompositions. For the QR factorization, you should write your own Matlab functions called "householder" and "getQb" to perform the necessary triangularization of A and compute $Q^T b$, respectively, using the version of the Householder QR algorithm that avoids any explicit computation of the projection matrices P_i discussed in class (see p.121 in Demmel for more details). You may solve any necessary triangular linear systems by using "\" in Matlab; be careful to ensure that the system is lower or upper triangular before calling backslash.

For each method and choice of m, plot the function f and the approximating polynomials obtained. Print the solution coefficient vector x, the norm of the residual $r = b - Ax$, and time required to obtain the approximation for each method and m. Explain carefully any observed differences regarding the quality of solutions and timing for the two methods.

Hint: it may be useful to consider the condition number $\kappa_2(A)$ of A used in each approximation and the dependency of error in the two least squares algorithms on $\kappa_2(A)$. To test your code and check your solutions you may use Matlab's built-in polynomial fitting function "polyfit" and the "\" command's least squares solver (which uses a QR decomposition).

Submission Instructions:

- Assignments are due at the **start** of class (1pm) on the due date.
- Write your name and ID# clearly on all pages, and underline your last name.
- **Matlab files:** please submit a single zip/rar/etc file with file name in the format **Lastname_Firstname_ID#_A4** to **homework.acm106a@gmail.com** that decompresses to a single folder of the same name containing the following:
  - A single thoroughly commented Matlab script file containing all commands used for each assigned programming/simulation problem. File names should have format **Lastname_Firstname_ID#_A4_P#** :

    e.g. **Ames_Brendan_12345678_A4_P4** for problem 4.

  - All Matlab functions used to perform any assigned programming/simulation problems with appropriate file names. Please add a comment to the beginning of each file with the format **Lastname_Firstname_ID#_A4_P#**
  - A diary file of your session for each programming/simulation problem with file name in the format **Lastname_Firstname_ID#_A4_P#_DIARY.txt**. Please also submit a hard copy of the diary and any relevant derivations, pseudocode, etc. you may want considered for partial credit with your submitted solution sets at the beginning of class.