

CAAM 551 Problem set 3

Due Fri. 9 Oct. 09

Problem 1

Least Squares Regularization

The following exercises will lead you through a procedure for choosing a regularization parameter via the trust region subproblem

$$\min\{\|\mathbf{b} - \mathbf{A}\mathbf{x}\| : \|\mathbf{x}\| \leq \Delta\}$$

where $\Delta > 0$ is a bound on the norm of the solution \mathbf{x} . (Here $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{x} \in \mathbb{R}^n$ with $m \geq n$) In all cases the norm $\|\cdot\|$ is the 2-norm.

In class we established that this problem could be solved by numerically computing the solution to

$$\text{Compute } \lambda \geq 0 \text{ s.t. } \frac{1}{\|\mathbf{x}_\lambda\|} = \frac{1}{\Delta}$$

where $\mathbf{x}_\lambda \equiv (\mathbf{A}^T \mathbf{A} + \lambda \mathbf{I})^{-1} \mathbf{A}^T \mathbf{b}$. (unless $\|\mathbf{x}_0\| < \Delta$ and the solution is then \mathbf{x}_0 with $\lambda = 0$)

We shall first work with a test problem using the following generating code

```
m = 100; n = 4;
s = [.5 , .8, 1 , 1.5];
S = diag(s);
[U,R] = qr(randn(m,n), 0); [V,R] = qr(randn(n),0);
%
% generate A with singular values
% specified by s
%
A = U*S*V';
x = ones(n,1);
b = A*x;
delta = norm(x)/(1.2);
```

- i) Write a Matlab code to graph $\phi(\lambda) \equiv \frac{1}{\|\mathbf{x}_\lambda\|}$ over the interval $\lambda \in [-2.5, 1]$. On the same plot (using hold on) plot the horizontal line $y(\lambda) = \frac{1}{\Delta}$ over the same interval and also plot the points $(-s(j)^2, 0)$ indicated by (red) * and the point $(0, \frac{1}{\|\mathbf{x}_0\|})$ indicated by a (green) o . Plot this several times and observe the graph. What do you notice? Why does the curve change? Please turn in one of the graphs and also the Matlab code with this HW.

- ii) Let $[\mathbf{Q}, \mathbf{R}] = \text{qr}(\mathbf{A}, 0)$; i.e. $\mathbf{A} = \mathbf{Q}\mathbf{R}$ (short form), and let $\mathbf{c} = [\mathbf{b}^T \mathbf{Q}, \mathbf{0}]^T$. Prove that \mathbf{x}_λ is the solution to the following least squares problem

$$\min \left\| \begin{bmatrix} \mathbf{R} \\ \sqrt{\lambda} \mathbf{I} \end{bmatrix} \mathbf{x}_\lambda - \begin{bmatrix} \mathbf{c} \\ \mathbf{0} \end{bmatrix} \right\|.$$

- iii) Let

$$\mathbf{Q}_\lambda \mathbf{R}_\lambda = \begin{bmatrix} \mathbf{R} \\ \sqrt{\lambda} \mathbf{I} \end{bmatrix}$$

be the short form QR-factorization. Give a formula for computing \mathbf{x}_λ for any non-negative value of λ that involves $\mathbf{Q}_\lambda, \mathbf{R}_\lambda$ and \mathbf{c} . Please discuss the importance of this formula if one wishes to compute \mathbf{x}_λ for many values of λ when $m \gg n$ (e.g. think $m = 100,000$ and $n = 100$). Please include a (somewhat) detailed discussion of the cost of this approach vs the alternative of using the SVD.

- iv) Prove that the following Matlab code segment will implement one step of Newton's method for solving $\phi(\lambda) = \frac{1}{\Delta}$:

```
M = [R ; sqrt(lamda)*I];
[Q1,R1] = qr(M,0);
x = (R1)\(Q1'*c);
w = (R1')\x;
psi = (norm(x) - delta)/delta;
dlamda = psi*(x'*x)/(w'*w);
lambda = lambda + dlamda;
```

(assume \mathbf{c}, \mathbf{R} are as defined above and that $\mathbf{I} = \text{eye}(n)$).

- v) Please use the code segment in item *iv* above to implement Newton's method for solving $\phi(\lambda) = \frac{1}{\Delta}$. Your function should have the following specification:

```
function [x, lambda] = minphi(A,b,delta,tol);
```

It should accept the input $\mathbf{A} = \mathbf{A}, \mathbf{b} = \mathbf{b}, \text{delta} = \Delta$ and tol (a user specified tolerance). It should return the solution $\mathbf{x} = \mathbf{x}_\lambda$ and the value $\text{lambda} = \lambda$ with

$$\left| \frac{\|\mathbf{x}_\lambda\| - \Delta}{\Delta} \right| \leq \text{tol}.$$

Please include a listing of your code with this HW.

- vi) Run your code on the test problem defined above with $\text{tol} = 10*\text{eps}$ and report the values `lambda` and `dlambda` side by side for each of the Newton iterates (This should be about 5 iterates if things are working properly). Please display the numbers for `lambda` in format `long e`.

What do you notice about the iterates? Is the expected quadratic rate of convergence obtained?

- vii) Repeat the computation in items *i* and *vi* with $\mathbf{s} = [.01, .8, 1, 1.5]$ and then again with $\mathbf{s} = [.001, .8, 1, 1.5]$. What happens to the iterates? Explain what you observe.
- viii) In all cases, the iterates (starting with $\lambda_1 = 0$ are monotone increasing: $\lambda_1 < \lambda_2 < \dots$ (at least they should be). Why does this always happen? Do you ever need *safeguarding* to assure convergence and termination of the iteration?

Problem 2

The regularization procedure can be generalized to use a weighted norm:

$$\min \|b - Ax\|_2 \quad \text{s.t.} \quad \|Wx\|_2 \leq \Delta$$

we assume $W \in \mathbf{R}^{n \times n}$ is nonsingular. Set up the Lagrangian for this problem and derive the optimality conditions (i.e. the parameterized normal equations). Explain how to use the QR decomposition (similar to Problem 1) in finding the optimal λ for this formulation.

Problem 3

Please do the Problem 1, page 126 of Saad.