AQM Problem Set 3

Ismael Martinez

January 17, 2018

Problem 1 - Regularization and Condition Numbers

Part a.

Let $\Gamma = \gamma I$.

$$
\begin{aligned}
RSS &= \frac{1}{2}(y - X\beta)^T(y - X\beta) - \frac{1}{2}\Gamma\beta^T\beta \\
\frac{\partial RSS}{\partial \beta} &= -X^T(y - X\hat{\beta}) + \Gamma\hat{\beta} = 0 \\
&\implies -X^Ty + X^TX\hat{\beta} + \Gamma\hat{\beta} = 0 \\
&\implies X^TX + \Gamma\hat{\beta} = X^Ty \\
&\implies (X^TX + \Gamma)\hat{\beta} = X^Ty \\
&\implies \hat{\beta} = (X^TX + \Gamma)^{-1}X^Ty = (X^TX + \gamma I)^{-1}X^Ty
\end{aligned}
$$

Part b.

$$
\begin{aligned}
cond(A) &= \frac{\frac{A^{-1}\Delta\|b\|}{A^{-1}\|b\|}}{\frac{\Delta\|b\|}{\|b\|}} \\
&= \frac{\|A^{-1}\Delta b\|}{\|A^{-1}b\|} \cdot \frac{\|b\|}{\|\Delta b\|} \\
&= \frac{\|A^{-1}\Delta b\|}{\|\Delta b\|} \cdot \frac{\|b\|}{\|A^{-1}b\|} \\
&= \|A^{-1}\| \cdot \frac{1}{\|A^{-1}\|} \\
&= \|A^{-1}\| \cdot \|A\|
\end{aligned}
$$

By the Cauchy-Schwarz inequality, we get

$$cond(A) = \|A^{-1}\| \cdot \|A\| \geq \|A^{-1}A\| = 1$$

Part c.

We assume that $\Delta A = 0$ and $\Delta b \leq b$.

$$
\begin{aligned}
\frac{\|\Delta\beta\|}{\|\beta\|} &= \frac{\|\Delta(A^{-1}b)\|}{\|A^{-1}b\|} \\
&= \frac{\|\Delta A^{-1}\Delta b\|}{\|A^{-1}b\|} \\
&\leq 1 \leq cond(A)
\end{aligned}
$$

Part d.

A is a positive definite matrix. $B = A + \alpha I$, $\alpha \in [0,1]$. Looking at the eigenvalues of A and B, we see:

$$
\begin{aligned}
& Ax = \lambda x \\
\implies & Ax + \alpha I x = \lambda x + \alpha x \\
\implies & Bx = (\lambda + \alpha)x
\end{aligned}
$$

Since the eigenvalues of B are larger than those of A, and A has all positive eigenvalues from being positive definite, then B has all positive eigenvalues and B has an inverse.

$$cond(A) = \|A^{-1}\|\|A\| \geq \|A^{-1} + \alpha I\|\|A + \alpha I\| = cond(B)$$

Problem 2 - Elastic Net and Variable Selection

Part a.

We first define our functions and augmented data.

$$J_1(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_2\|\beta\|_2^2 + \lambda\|\beta\|$$

and

$$J_2(\widetilde{\beta}) = \|\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\widetilde{\beta}\|_2^2 + \lambda_1\|\widetilde{\beta}\|$$

where $c = (1 + \lambda_2)^{-\frac{1}{2}}$, $\beta = c\widetilde{\beta}$, and

$$\widetilde{\mathbf{X}} = c\begin{bmatrix} X \\ \sqrt{\lambda_2}\mathbf{I}_d \end{bmatrix}, \widetilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_{dx1} \end{bmatrix}$$

1.

Proof.

$$\begin{aligned} c(\underset{\widetilde{\beta}}{\operatorname{argmin}} J_2(\widetilde{\beta})) &= c[\underset{\widetilde{\beta}}{\operatorname{argmin}} \|\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\widetilde{\beta}\|_2^2 + c\lambda_1\|\widetilde{\beta}\|] \\ &= c[\underset{\beta}{\operatorname{argmin}} \frac{1}{c}[\sum^{N}(\mathbf{y} - c\mathbf{X}\frac{\beta}{c})^2 + \sum^{D}(0 - c\sqrt{\lambda_2}\frac{\beta}{c})^2 + c\lambda_1\sum^{D}|\frac{\beta}{c}|]] \\ &= \frac{c}{c}[\underset{\beta}{\operatorname{argmin}} \sum^{N}(\mathbf{y} - \mathbf{X}\beta)^2 + \lambda_2\sum^{D}\beta^2 + \lambda_1\|\beta\|] \\ &= \underset{\beta}{\operatorname{argmin}} J_1(\beta) \end{aligned}$$

$\square$

2.

$$J_{W_1}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T \mathbf{W}(\mathbf{y} - \mathbf{X}\beta) + \lambda_2 \|\beta\|_2^2 + \lambda_1 \|\beta\|$$

$$\text{where } \mathbf{W} = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 & \dots & 0 \\ 0 & \frac{1}{\sigma_2^2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\sigma_n^2} \end{bmatrix}$$

$$\hat{\beta} = \operatorname*{argmin}_{\beta} J_{W_1}(\beta)$$

3.

$$J_{W_2}(\widetilde{\beta}) = (\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\widetilde{\beta})^T \widetilde{\mathbf{W}}(\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\widetilde{\beta}) + \lambda_1 \|\widetilde{\beta}\|$$

$$\text{where } \widetilde{\mathbf{W}} = \begin{bmatrix} \mathbf{W} \\ \mathbf{I}_d \end{bmatrix}$$

$$\hat{\beta} = \operatorname*{argmin}_{\widetilde{\beta}} J_W(\widetilde{\beta})$$

4.

$$\hat{\beta} = \operatorname*{argmin}_{\beta} \beta^T \left(\frac{\mathbf{X}^T \mathbf{W} \mathbf{X} + \lambda_2 \mathbf{I}}{1 + \lambda_2} \right) \beta - 2\mathbf{y}^T \mathbf{W} \mathbf{X} \beta + \lambda_1 \|\beta\|_1$$

Part b.

$$J_W(\beta) = \beta^T \left(\frac{X^T W X}{1 + \lambda_2} \right) \beta - 2y^T W X \beta + \lambda_1 \|\beta\|_1$$

$$\frac{\partial J_W}{\partial \beta} = \beta^T \left(\left(\frac{X^T W X}{1 + \lambda_2} \right)^T + \left(\frac{X^T W X}{1 + \lambda_2} \right) \right) - 2y^T W X + \lambda_1 \partial \|\beta\|_1$$

$$\text{where } \partial \|\beta\|_1 = \begin{cases} -\lambda_1 & \text{if } \beta_j < 0 \\ [-\lambda_1 . \lambda_1] & \text{if } \beta_j = 0 \\ \lambda_1 & \text{if } \beta_j > 0 \end{cases}$$

Part c.

Run the file `GradientDescentMethods.py` which calls the `proximalGradientDescent` and `subgradientDescent.py` algorithms. By testing a range of λ_1 and λ_2 values, we see that the Proximal Gradient Descent method is much faster.

Part d.

1. Find the parallelized algorithm in `parallelizedGradientDescent.py`.
2. Find the Bootstrap CI Algorithm in `bootstrapBoston.py`

Part f.

2. We are clustering a set of normally distributed arrays of size 30, where each point is drawn from a $\mathcal{N}(0, \mathbf{\Sigma})$. It can be important to distinguish between points overlapped in multiple clusters as they show a level of uncertainty to whcih cluster a point belongs and allows us to view those points as having a probability of belonging to one or the other. When a point must be a part of a single cluster, having this non-overlapping approach is more useful.