DS4400 Notes

Xin Guan

1. Convex functions:

   A function $f : \mathbb{R}^d \to \mathbb{R}$ is convex iff $\forall \theta_1, \theta_2 \in \mathbb{R}^d$ and $\forall \alpha \in [0,1]$ we have $f(\alpha\theta_1 + (1-\alpha)\theta_2) \leq \alpha f(\theta_1) + (1-\alpha) f(\theta_2)$

   In the special case $(d = 1)$ $f : \mathbb{R} \to \mathbb{R}$, f is convex iff $\forall \theta, f''(\theta) \geq 0$

   When the function is convex, **local min** $\equiv$ **global min**. When the system is not convex, we might find only a **local min** but not a **global min**

2. Dealing with non convex function:

   In gradient descent:

   (a) use larger ρ in the beginning and gradually decrease ρ with interation.

   (b) Run SGD/GD with multiple random initializaitons $\theta_1^{(0)}, \theta_2^{(0)} \ldots$ and keep the best solution.

3. $\min_\theta \sum_{i=1}^N (y_i - \theta^T x_i)^2 \triangleq J(\theta)$

   In linear regression, $J(\theta)$ is convex.

4. Robustness of Regression to outliers:

   (a) Run outlier detection algorithm, remove detected outliers, then run Linear Regression on remaining points.

   (b) Robust Regression cost function.
   $\min_\theta \sum_{i=1}^N e_i^2$, $e_i \triangleq y_i - \theta^T x_i$
   e^2 is extremly unhappy with large errors.

   we might use $|e|$ to replace the function. This might be more tolerance. Then, $\min_\theta \sum_{i=1}^N |y_i - \theta^T x_i|$

5. Exercise: D = $\{(x_1, y_1 = 100) \ldots (x_10, y_10 = 100), (x_{11}, y_{11} = 0), (x_{12}, y_{12} = 0)\}$

   e^2: $10(\theta - 100)^2 + 2\theta^2 \to$
   $\frac{\partial}{\partial \theta} = 20(\theta - 100) + 4\theta = 0 \to$
   $\theta = 83.3$

   $|e| : \min_\theta \sum_{i=1}^{12} |\theta - y_i| = 10|\theta - 100| + 2\theta$
   $(\theta \leq 100) = \min_\theta 10(100 - \theta) + 2\theta$
   $= 1000 - 8\theta \to \theta = 100$
   $(\theta \geq 100) = \min_\theta 10(\theta - 100) + 2\theta$
   $= 12\theta - 1000 \to \theta = 100$

6. How to solve l1-norms cost functions?

   (a) No closed form

   (b) we need to be careful with gradient descent

   (c) We need to use convex programming toolboxs (convex optimizations)

7. Huber loss funct

$$l_\delta(e) = \begin{cases} \frac{1}{2}e^2 & |e| \leq \delta \\ \delta|e| - \frac{\delta^2}{2} & |e| \geq \delta \end{cases}$$

$$\frac{\partial l_\delta(e)}{\partial e} = \begin{cases} e & -\delta \leq e le \delta \\ \delta & e > \delta \\ -\delta & e < \delta \end{cases}$$

   in huber loss function, we don't have closed form solution but we can run gredient descent now.

8. Definition: Overfitting:

   Learning a system from traning data that does very well on training data itself (e.g, very low regression error on traning data), but performs poorly on test data.

9. Definition: Overfitting in Linear Regression

   $\Phi^T \Phi \theta = \Phi^T Y$
   $\Rightarrow \theta^* = (\Phi^T \Phi)^{-1} \Phi^T Y$

   $\text{rank}(\Phi^T \Phi) \leq \min\{rk(\Phi^T), rk(\Phi)\} = rk(\Phi) \leq \min\{N, d\}$

   $\Phi^T \Phi$ is $d \times d$ matrix, then rank is $\leq d$.

   Therefore, when $N < d$ it is not invertible which means we have multiple solutions and results in overfitting.