The University of Texas at Austin
Optimization

HOMEWORK 9

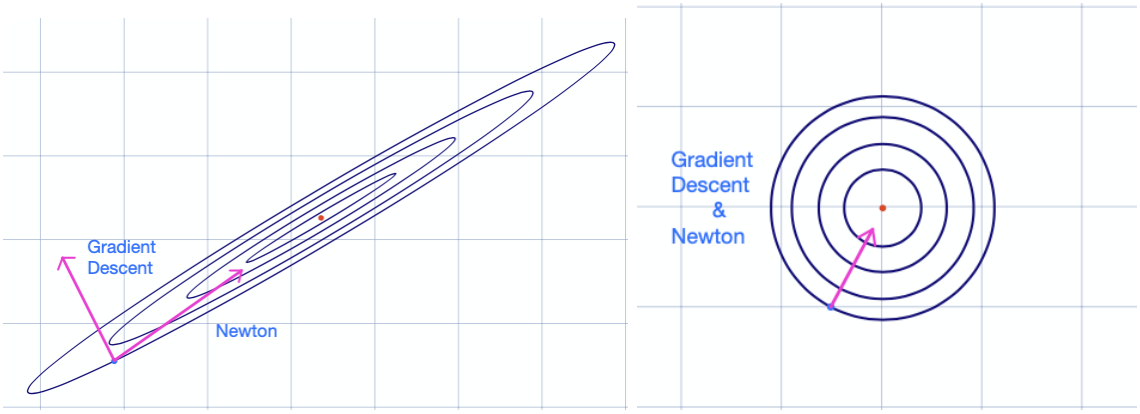
Constantine Caramanis, Sujay Sanghavi

Submitting solutions: *Please submit your solutions as a single pdf file. If you have code or figures, please include these in the pdf.*

1. Consider the convex unconstrained problem:

$$\min : \ f(\boldsymbol{x}),$$

where $f(\boldsymbol{x})$ is a strongly convex function. One of our motivation for Newton's method was based on the idea of finding the "best" affine transformation possible at each step of a gradient-descent algorithm. We illustrate the key idea in Figure 1. Gradient descent, by its definition, moves along the negative gradient at the current point. The gradient (and hence negative gradient) always points in the direction (locally) perpendicular to the level set. In the first example, due to the elongation of the level sets, the gradient is pointing in a direction almost perpendicular to the direction of the optimal solution. In the second example, there is no elongation, and the level sets are spherical, centered on the optimal solution. Here, gradient descent points directly towards the optimal solution. We see that in both examples, Newton's method is not affected by the local elongation of the level sets.



(a) Gradient Descent and Newton updates can be very different – nearly orthogonal – in the event that the level sets are (locally) very elongated.

(b) Gradient Descent and Newton updates coincide in the setting where the level sets are (locally) spherical.

Figure 1: Newton vs Gradient Descent

You will explore this explicitly in this exercise and the next.

(a) Consider the ellipse given by:

$$\mathcal{E} = \left\{ \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \ : \ \boldsymbol{x}^\top Q \boldsymbol{x} \leq 1 \right\},$$

where

$$Q = \begin{bmatrix} 1/9 & 0 \\ 0 & 1 \end{bmatrix}.$$

Plot the ellipse $\mathcal{E}$.

(b) Now consider the affine transformation (change of coordinates) given by

$$A\boldsymbol{y} = \boldsymbol{x},$$

where

$$A = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}.$$

Plot the ellipse in the new $\boldsymbol{y}$-coordinates.

(c) Now consider the ellipse given by

$$\mathcal{E} = \left\{ \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} : \boldsymbol{x}^\top Q \boldsymbol{x} \leq 1 \right\},$$

where

$$Q = \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}.$$

Plot the ellipse $\mathcal{E}$.

(d) Find a change of coordinates so that the ellipse in the new coordinates is a sphere. That is, find a matrix A so that the new ellipse in the $\boldsymbol{y}$-coordinates is spherical.

2. We will apply the same ideas to Gradient descent and Newton's method. Consider again a convex function $f(\cdot)$, and our standard unconstrained convex optimization problem.

   For A an invertible matrix, consider the change of coordinates $A\boldsymbol{y} = \boldsymbol{x}$, and accordingly, the function

   $$g(\boldsymbol{y}) = f(A\boldsymbol{y}).$$

   - Consider some starting point $\boldsymbol{x}_0$, and the sequence produced by gradient descent on $f(x)$ starting from $\boldsymbol{x}_0$, and using step-size η_k at iteration k. Define also $\boldsymbol{y}_0$ given by $A\boldsymbol{y}_0 = \boldsymbol{x}_0$, and consider the sequence produced by performing gradient descent on $g(\boldsymbol{y})$ starting from $\boldsymbol{y}_0$, and with the same step size. Show that in general these will not be the same by providing a specific example of a function $f(\cdot)$ and a matrix A, and demonstrating that the trajectories are not the same.
   - Now repeat this for Newton's method, where the updates $\{\boldsymbol{x}_k\}$ and $\{\boldsymbol{y}_k\}$ are generated by using *undamped* Newton's method on $f(\boldsymbol{x})$ and $g(\boldsymbol{y})$, respectively. Show that $A\boldsymbol{y}_k = \boldsymbol{x}_k$ for all k. Show this *in general*, not through example.

3. Consider the ℓ_1-regularized regression problem

$$\min_{\boldsymbol{x}} \quad \frac{1}{2}\|A\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_1 \tag{1}$$

   Where $\boldsymbol{x} \in \mathbb{R}^d$, $\boldsymbol{y} \in \mathbb{R}^n$. Show that a point $\bar{x}$ is an optimum of this problem if and only if there exists a $\boldsymbol{z} \in \mathbb{R}^d$ such that both the following hold:
   (a) $-A'(\boldsymbol{y} - A\bar{\boldsymbol{x}}) + \lambda \boldsymbol{z} = 0$
   (b) For every $i \in [d]$, $z_i = sign(\bar{x}_i)$ if $\bar{x}_i \neq 0$, and $|z_i| \leq 1$ if $\bar{x}_i = 0$.

4. Consider the Lasso problem from the previous exercise: given a $n \times d$ matrix A, and a vector $\boldsymbol{y} \in \mathbb{R}^n$, and a positive scalar λ:

$$\min_{\boldsymbol{x}} : \ \frac{1}{2}\|A\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_1.$$

As we have discussed extensively, this problem is convex. But as it is the sum of a smooth term ($\|A\boldsymbol{x} - \boldsymbol{y}\|_2^2$) and a non-smooth term ($\|\boldsymbol{x}\|_1$) the overall objective is not smooth. We have spent some time developing a composite-optimization approach to solve this, namely, the proximal gradient method. The goal of the proximal gradient approach is to exploit the fact that the first term is smooth, and the second term has an "easy" to evaluate prox function. On the other hand, since it is convex, we could just use the sub-gradient method to solve directly, without treating the two terms separately.

   (a) Write down the subgradient method for this problem. Be as explicit as possible when you write out the subgradient.

   (b) Now write the proximal gradient update. Use a fixed step size, $\eta_t = \eta$. What is a good choice of η?

5. For the same setting as above, generate random data, and solve the problem numerically, plotting the rate of convergence $\|\boldsymbol{x}_t - \boldsymbol{x}^*\|_2$. I recommend: choose $n = 50$ and $d = 200$, and let $\boldsymbol{x}^*$ a vector with only 5 non-zero values (say, choose them to be -10 or 10). Choose the matrix A by selecting each of its $n \times d$ entries uniformly at random from a standard Gaussian distribution ($A_{ij} \sim N(0, 1)$). Then, let $\boldsymbol{e}$ be a random n-dimensional vector, where each entry is generated according to $N(0, 0.1)$. And finally, set

$$\boldsymbol{y} = A\boldsymbol{x}^* + \boldsymbol{e}.$$

The game now is to pretend we do not know $\boldsymbol{x}^*$, and to try to estimate it by solving the Lasso problem. That is,

$$\hat{\boldsymbol{x}} = \arg\min : \ \frac{1}{2}\|A\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_1.$$

Use sub-gradient and prox grad method to solve this, and compare the rate of convergence for each method by plotting $\|\boldsymbol{x}_t - \boldsymbol{x}^*\|_2$ vs iteration.