The University of Texas at Austin
Optimization

HOMEWORK 6

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. Compute the Fenchel Transform of the following two functions:

   (a)
   $$f(\boldsymbol{x} = \sum_{i=1}^{n} x_i \log x_i.$$

   (b)
   $$f(\boldsymbol{x} = -\sum_{i=1}^{n} \log x_i.$$

2. Let $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ be any function, not necessarily convex. Show that $f^{**}(\boldsymbol{x}) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x}$.

3. Recall that one of the parameters of gradient descent is the step size chosen. Consider the step-size $\eta_t = 2^{-t}$. Show that this is not a good choice, by giving a example of a convex function and initialization point, where gradient descent fails to approach an optimal solution using such a step size rule.

4. Implementing Gradient Descent. In this question you will implement gradient descent for an easy problem: convex quadratic functions. Consider the problem:
   $$\min : \ f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top Q\boldsymbol{x} + \boldsymbol{q}^\top \boldsymbol{x},$$
   where Q is a (strictly) positive definite matrix, i.e., it is symmetric and all its eigenvalues are strictly positive.

   (a) Compute the gradient: $\nabla f(\boldsymbol{x})$.

   (b) Implement gradient descent using three stepsize choices: $\eta_t = 1/t$, $\eta_t = 1/\sqrt{t}$, and $\eta_t = \eta$ (in other words, the last one is for fixed η).

   (c) Randomly generate data (make sure that your matrix Q is symmetric and strictly positive definite) and plot your results for the first two choices. For the third, find values of η for which gradient descent converges, and for which it diverges.

5. Consider the optimization problem:

$$\min: \; f_1(x_1) + f_2(x_2) + f_3(x_3) + g(x_1 + x_2 + x_3),$$

where f_1, f_2, f_3 and g are all convex functions of a single variable. In this exercise, we will get a better hint as to the potential benefits of dual algorithms.

(a) The lectures explained how to compute the dual of this problem, in the general case when the objective is of the form: $f(\boldsymbol{x}) + g(A\boldsymbol{x})$. In this case, $f(\cdot)$ has special structure:

$$f(\boldsymbol{x}) = f(x_1, x_2, x_3) = f_1(x_1) + f_2(x_2) + f_3(x_3),$$

and A as well has specified form. Using these, derive the dual optimization problem. It is a univariate optimization problem.

(b) Now, write the dual update. As explained in the lecture, computing the update itself requires solving an optimization problem. Write down the update explicitly, and show that it can be computed by solving four univariate optimization problems.