ECE 599 / CS 539 Large-Scale Convex/Nonconvex Optimization - Homework 3

Fall 2019
School of Electrical Engineering and Computer Science
Oregon State Univeristy

Due: Nov. 29, 2019

Q1 Answer the following questions.

a) Consider the linear program

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \end{array}$$

with A square and nonsingular. Show that the optimal value is given by

$$p^\star = \begin{cases} c^T A^{-1} b, & A^{-T} c \preceq 0 \\ -\infty & \text{otherwise} \end{cases}$$

(10%)

b) Formulate the following problems as an LP:

$$\begin{array}{ll} \text{miminize} & \|x\|_1 \\ \text{subject to} & \|Ax - b\|_\infty \leq 1, \end{array}$$

where $A \in \mathbb{R}^{m \times n}$. (10%)

c) Formulate the ℓ_4-norm approximation problem as an equivalent QCQP:

$$\text{minimize } \|Ax - b\|_4,$$

where $A \in \mathbb{R}^{m \times n}$. (10%)

d) Consider a robust variation of the (convex) quadratic program

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Ax \preceq b \end{array}$$

For simplicity we assume that only the matrix P is subject to errors, and the other parameters (q, r, A, b) are exactly known. The robust quadratic program is defined as

$$\begin{array}{ll} \text{minimize} & \sup_{P \in \mathcal{E}} (1/2)x^T P x + q^T x + r \\ \text{subject to} & Ax \preceq b. \end{array}$$

Express the robust QP as an SOCP problem given that

$$P = \{P_0 + \sum_{i=1}^{K} P_i u_i \mid \|u\|_2 \leq 1\},$$

where $P_i \in \mathbb{S}_+^n$ for $i = 0, 1, \ldots, K$. (10%)

Q2 Verify that $x^\star = (1, 1/2, -1)$ is optimal for the optimization problem

$$\begin{array}{c} \text{minimize } (1/2)x^T P x + q^T x + r \\ \text{subject to } -1 \leq x_i \leq 1, \ i = 1, 2, 3, \end{array}$$

where

$$P = \begin{bmatrix} 13, & 12, & -2 \\ 12, & 17, & 6 \\ -2, & 6, & 12 \end{bmatrix}, \ q = \begin{bmatrix} -22.0 \\ -14.5 \\ 13.0 \end{bmatrix}, \ r = 1.$$

(10%)

Q3 In this problem, we look at an optimization problem

$$\begin{aligned} \text{minimize } & \mathrm{Tr}(CX) \\ \text{subject to } & \mathrm{Tr}(AX) = 1 \\ & X \succeq 0 \\ & \mathrm{rank}(X) = 1. \end{aligned} \tag{1}$$

This is a very hard problem, as we have seen in class. The common trick of handling this problem is to discard $\mathrm{rank}(X) = 1$ and consider

$$\begin{aligned} \text{minimize } & \mathrm{Tr}(CX) \\ \text{subject to } & \mathrm{Tr}(AX) = 1 \\ & X \succeq 0. \end{aligned} \tag{2}$$

That is, the semidefinite relaxation (SDR). Normally solving (2) does not give a solution to (1), but there are some interesting exceptions. Let us consider the case where $C \succ 0$ and $A = aa^T$. Also assume that Slater's condition is satisfied by (2).

a) What are the KKT conditions? (Hint: read B&V book 5.9.2.) (10%)

b) Show that the solution

$$X^\star_{\mathsf{SDR}} = \arg\min \ (2)$$

satisfies

$$\mathrm{rank}(X^\star_{\mathsf{SDR}}) = 1.$$

That is, SDR solves the original **nonconvex problem** to global optimality. (10%)

Q4 Download the datasets `train_separable.mat` and `test_separable.mat` from the course website. Download `CVX` from `http://cvxr.com/cvx/` (or `http://www.cvxpy.org/en/latest/` if you use Python) and learn how to use it. Implement the following using `CVX`.

a) Apply the C-Hull formulation to train a classifer, i.e.,

$$\begin{aligned} \text{minimize}_{u,v} \ & \|Au - Bv\|_2^2 \\ \text{subject to } & 1^T u = 1, \ u \succeq 0 \\ & 1^T v = 1, \ v \succeq 0 \end{aligned}$$

Visualize the training data together with the classifier. Also visualize the testing data and the classifier in another figure, and report the classification error on the testing data using the true labels provided in `test_separable.mat` . (15%)

b) Repeat the above for `train_overlap.mat` and `test_overlap.mat` using the reduced C-Huall, i.e.,

$$\begin{aligned} \text{minimize}_{u,v} \ & \|Au - Bv\|_2^2 \\ \text{subject to } & \mathbf{1}^T u = 1, \ d\mathbf{1} \succeq u \succeq 0 \\ & \mathbf{1}^T v = 1, \ d\mathbf{1} \succeq v \succeq 0. \end{aligned}$$

Report the classification error on the testing data using $d = 0.9$. (15%)

(Please also submit your scripts. In case that your do not know what is C-Hull, check out the paper by Kristin P. Bennett, and Erin J. Bredensteiner, "**Duality and geometry in SVM classifiers**," ICML 2000. In addition, for background of classification, check out the slides of **Lecture 1**.)