EECS 127/227AT **Fall 2020**
Homework 8
Homework 8 is due on Gradescope by Friday 11/6 at 11.59 p.m.

1 Median versus average

This question illustrates the connection between the median and the mean of a finite set of real numbers. It touches on problem transformation methods, so the most relevant sections of the textbooks would be Sec. 8.3.4 of the textbook of Calafiore and El Ghaoui and Sec. 4.2.4 of the textbook of Boyd and Vandenberghe.

For a given vector $v \in \mathbb{R}^n$, the average can be found as the solution to the optimization problem

$$\min_{x \in \mathbb{R}} \|v - x\mathbb{1}\|_2^2, \tag{1}$$

where $\mathbb{1}$ denotes the vector of ones in $\mathbb{R}^n$. Similarly, it turns out that the median (a median is any value x such that there is an equal number of values in v above and below x) can be found via the optimization problem

$$\min_{x \in \mathbb{R}} \|v - x\mathbb{1}\|_1. \tag{2}$$

We consider a robust version of problem (1) of finding the average, i.e.

$$\min_{x} \max_{u\,:\,\|u\|_\infty \le \lambda} \|v + u - x\mathbb{1}\|_2^2, \tag{3}$$

in which we assume that the components of v can be independently perturbed by a vector u each of whose components has magnitude bounded by a given number $\lambda \geq 0$.

(a) Is the robust problem (3) convex? You should be able to justify your answer based on the expression (3), without having to do any manipulations,

(b) Show that problem (3) can be expressed as

$$\min_{x \in \mathbb{R}} \sum_{i=1}^{n} \left(|v_i - x| + \lambda\right)^2.$$

(c) Express problem (2) as an LP. State precisely the variables and the constraints if any.

(d) Express problem (3) as a QP. State precisely the variables and the constraints, if any.

(e) Show that when λ is large the solution set of the problem in (3) approaches that of the median problem (2).

(f) It is often said that the median is a more robust notion of "middle" value of a finite set of real numbers than the average, when noise is present in the observations. Based on the previous part of this question, justify this statement.

2 A minimum time path problem

This question illustrates how to formulate an optimization problem as an SOCP. The problem studied in this question arises in optics. The relevant sections of the textbooks are Secs. 10.1 and 10.2 of the textbook of Calafiore and El Ghaoui and Sec. 4.4.2 of the textbook of Boyd and Vandenberghe.

Consider Figure 1, in which a point in 0 must move to reach point $p = [4 \;\; 2.5]^\top$, crossing three layers of fluids having different densities.

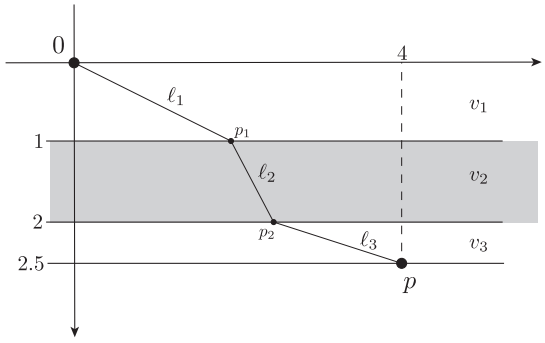

Figure 1: A minimum-time path problem.

In the first layer, the point must travel at speed v_1, while in the second layer and third layers it must travel at lower maximum speeds, respectively $v_2 = v_1/\eta_2$, and $v_3 = v_1/\eta_3$, with $\eta_2, \eta_3 > 1$. Assume $v_1 = 1$, $\eta_2 = 1.5$, $\eta_3 = 1.2$. You have to determine what is the fastest (i.e., minimum time) path from 0 to p.

Hint: You may use path leg lengths ℓ_1, ℓ_2, ℓ_3 as variables, and observe that, in this problem, equality constraints of the type $\ell_i =$ "something" can be equivalently substituted for by inequality constraints $\ell_i \geq$ "something" (explain why).

3 LASSO vs Soft-Margin SVM vs Ridge Regression

In this question we will compare three different methods of classification on a very simple classification problem in $\mathbb{R}^2$. Here is the link to the jupyter notebook:

a) Please fill out the code to perform ridge regression in the jupyter notebook, you may use `sklearn` or other packages of your choice.

b) Please fill out the code to perform LASSO in the jupyter notebook, you may use `sklearn` or other packages of your choice.

c) Please fill out the code to train a soft-margin support vector machine in the jupyter notebook, you may use `sklearn` or other packages of your choice.

d) Comment on your results for each part. Which model performed the worst and why? Which model performed the best and why? What values of λ did you try for parts a and b?

e) LASSO stands for Least Absolute Shrinkage and Selection Operator. In a sense, the shrinkage part comes from the fact that we penalize vectors with large ℓ_1 norms, but where does the "selection" part come from? In this problem we will try to understand LASSO better. Recall that LASSO is formally defined as:

$$\arg\min_{w \in \mathbb{R}^d} \|Xw - y\|_2^2 : \|w\|_1 \leq \lambda$$

which is equivalent to:

$$\arg\min_{w \in \mathbb{R}^d} \|Xw - y\|_2^2 + \lambda \|w\|_1,$$

in the sense that solving the former class of problems as the hyperparameter varies is equivalent to solving the latter class of problems as the hyperparameter varies.

Assume for simplicity that the data has been centered and whitened so that each feature has mean 0, variance 1, and the features are uncorrelated, i.e. $X^T X = nI$. Here $X \in R^{n \times n}, y \in \mathbb{R}^n$, and $w \in \mathbb{R}^n$.

i. First decompose the problem into n univariate problems over each element of w. Let X_i denote the i^{th} column of X.

ii. Assume that the optimal w_i, denoted by w_i^*, satisfies $w_i^* > 0$. What is w_i^* and what is the condition on $y^T X_i$ for this to be possible?

iii. Now assume that the $w_i^* < 0$. What is w_i^* and what is the condition on $y^T x_i$ for this to be possible?

iv. What can you conclude about w_i^* if $|y^T X_i| \leq \frac{\lambda}{2}$? How does the value of λ impact the individual entries of w^*?

4 Support vector machines

This question explores the role of the dual in maximum margin support vector machines. The relevant sections from the textbooks are Secs. 13.3 and 13.4 of the textbook of Calafiore and El Ghaoui.

Let $x_1, \ldots, x_n$ be vectors in $\mathbb{R}^n$, with each x_i having a classification $y_i \in \{-1, 1\}$. We wish to find $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$ such that the hyperplane

$$\mathcal{H} := \{x \in \mathbb{R}^n : w^T x + b = 0\} \tag{4}$$

solves the problem

$$\min_{w,b} \quad \|w\|_2 \tag{5}$$

$$\text{s.t.:} \quad y_i(w^\top x_i + b) \geq 1,\ i = 1, \ldots, n. \tag{6}$$

This is the *maximum margin SVM* problem, as posed in eqn. (13.11) of the textbook of Calfiore and El Ghaoui.

(a) Show that the problem (5) can be solved via the QP

$$\min_{w,b} \quad \frac{1}{2}\|w\|_2^2 \tag{7}$$

$$\text{subject to:} \quad y_i(w^\top x_i + b) \geq 1,\ i = 1, \ldots, n. \tag{8}$$

(b) Show that the Lagrangian dual to the primal QP in (7) can be written as the QP

$$\begin{aligned} \max_{\lambda} \quad & -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^\top x_j + \sum_{i=1}^{n} \lambda_i \\ \text{s.t.} \quad & \sum_{i=1}^{n} \lambda_i y_i = 0, \\ & \lambda_i \geq 0,\ i = 1, \ldots, n. \end{aligned} \tag{9}$$

(c) Show that strong duality holds.

(d) Since strong duality holds, we know that if (w^*, b^*) is primal optimal and λ^* is dual optimal then they must together satisfy the KKT conditions. Since the primal is a convex optimization problem, we also know that if (w, b) is any primal point and λ any dual point that together satisfy the KKT conditions, then (w, b) will be primal optimal and λ will be dual optimal (and we will learn that strong duality holds, but we already know this).

If we found a primal optimal pair (w^*, b^*) we would use it to classify a new vector $\hat{x} \in \mathbb{R}^n$ by computing $w^{*\top}\hat{x} + b^*$ and assigning it to the class $\text{sign}(w^{*\top}\hat{x} + b^*)$.

Suppose that (w^*, b^*) is a primal optimal pair and λ^* is dual optimal (Note that part of this assumption is that the primal problem is feasible, which will hold only if the data is linearly separable). Further, assume that the value of the primal problem (and hence of the dual problem, since we have strong duality) is strictly positive. Using the KKT conditions for the problem, show that we have

$$w^* := \sum_{i=1}^{n} \lambda_i^* y_i x_i, \tag{10}$$

$$b^* := -\frac{1}{2}\left(\max_{i:y_i=-1}\left(\sum_{j=1}^{n} \lambda_j^* y_j x_j^\top x_i\right) + \min_{i:y_i=1}\left(\sum_{j=1}^{n} \lambda_j^* y_j x_j^\top x_i\right)\right). \tag{11}$$

Remark: A consequence of this is that we can base our classification of new vectors $\hat{x}$ entirely on the dual optimal λ^*, i.e. on the sign of

$$\sum_{i=1}^{n} \lambda_i^* y_i x_i^\top \hat{x} - \frac{1}{2}\left(\max_{i:y_i=-1} (\sum_{j=1}^{n} \lambda_j^* y_j x_j^\top x_i) + \min_{i:y_i=1} (\sum_{j=1}^{n} \lambda_j^* y_j x_j^\top x_i) \right).$$

Remark: Typically only a few of the λ_i^* will be nonzero (these correspond to the data points that are on the margin of the classifier). The sum of terms in the formula for b^* needs to be computed only once after the dual problem is solved. Further, to compute this sum as well as the term that depends on the new vector $\hat{x}$ all we need to do is to compute inner products (either of the type $x_i^\top x_j$ or $x_i^T \hat{x}$). In many machine learning applications this can be done much more efficiently than it would appear from the dimension of the space in which the vectors live, because the vectors x_i, $1 \leq i \leq n$ and $\hat{x}$ themselves arise as the image of points in some other underlying space (often called the space of *observations* or *patterns*) and computing the inner product can be done much more efficiently in that space using a function of pairs in that space called the *kernel*, which is basically the inner product of the corresponding vectors—this observation is called the *kernel trick*. [1] The point of working directly with the dual, then, is that if we were to work directly with an optimal primal pair (w^*, b^*), this would require us to compute the inner product $w^{*\top} \hat{x}$, but typically w^* is not easily interpretable in terms of an underlying observation (or pattern) and therefore will not a priori allow an easy computation of the inner product $w^{*\top} \hat{x}$.

[1] A more accurate way of describing how this works in practice is that the kernel is what we start with, to measure dissimilarity of patterns (or observations) and, for kernels satisfying the property that it is possible to map the observations (or patterns) to vectors in such a way that the kernel for a pair of observations (or patterns) becomes the inner product of the corresponding vectors (these vector are called *feature vectors*), we can then, in effect, determine the maximum margin classifier in the vector space. This will then map back to a nonlinear classification rule in the space of patterns (or observations). See Fig. 13.12 of the textbook of Calafiore and El Ghaoui for an example of this.

5 Geometric programs

This question discusses the use of convex optimization methods in the study of geometric programs. The relevant sections of the textbooks are Sec. 9.7 of the textbook of Calafiore and El Ghaoui and Sec. 4.5 of the textbook of Boyd and Vandenberghe.

We have 3 functions,

$$\begin{aligned} f(x) &= \alpha x_1^{a_1} x_2^{a_2}, && \text{dom } f = \mathbb{R}^2_{++}; \\ g(x) &= \beta_1 x_1^{b_{11}} x_2^{b_{21}} + \beta_2 x_1^{b_{12}} x_2^{b_{22}}, && \text{dom } g = \mathbb{R}^2_{++}; \\ h(x) &= \gamma x_1^{c_1} x_2^{c_2}, && \text{dom } h = \mathbb{R}^2_{++}, \end{aligned}$$

where, $\alpha, \beta_1, \beta_2, \gamma, a_1, a_2, b_{11}, b_{21}, b_{12}, b_{22}, c_1, c_2 \in \mathbb{R}$ are constants and $\alpha, \beta_1, \beta_2, \gamma > 0$.

Consider the optimization problem (such a problem is called a *geometric program*) given by

$$\begin{aligned} \min_x \ & f(x) \\ \text{s.t. } & g(x) \leq 1 \ \text{ and } \ h(x) = 1. \end{aligned} \tag{12}$$

Note that the domain of the problem is $\mathbb{R}^2_{++}$.

(a) Is the problem as stated convex? Why or why not?

(b) A function of the form $\delta x_1^{d_1} \ldots x_n^{d_n}$ in the variables $x_1, \ldots, x_n$, with $d_i \in \mathbb{R}$ for $1 \leq i \leq n$ and $\delta > 0$ is called a *monomial* in the theory of geometric programming (note that this terminology is not consistent with the use of the term *monomial* in the theory of polynomials, where it would be required that each d_i should be a nonnegative integer and there would be no requirement of positivity on the coefficient δ). The domain of the monomial is $\mathbb{R}^n_{++}$.

Let $y_i := \log x_i$ for $1 \leq i \leq n$, the logarithm being to the natural base. Show that $f(x)$ is a monomial in the variables $x = (x_1, \ldots, x_n)$ then $y \to \log f(e^y)$ is a convex function of $y \in \mathbb{R}^n$ (here e^y is interpreted coordinatewise).

(c) The *log-sum-exp function* is the function lse : $\mathbb{R}^n \to \mathbb{R}$ given by

$$\text{lse}(x) := \log(\sum_{i=1}^{n} e^{x_i}),$$

where $x := \begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix}^T \in \mathbb{R}^n$. Here the logarithm is to the natural base. Note that the log-sum-exp function is well-defined for all $x \in \mathbb{R}^n$ because the argument of the logarithm will be strictly positive for all $x \in \mathbb{R}^n$, so we can take $\text{dom}(\text{lse}) = \mathbb{R}^n$.

Show that lse is a convex function on $\mathbb{R}^n$.

(d) Let $f_j(x)$, $1 \leq j \leq m$ be monomials in $x = (x_1, \ldots, x_n)$. Let $f(x) := \sum_{j=1}^m f_j(x)$. Let $y_i := \log x_i$ for $1 \leq i \leq n$. Show that $y \to \log f(e^y)$ is a convex function of $y \in \mathbb{R}^n$, where e^y is interpreted coordinatewise.

(e) Based on the observations in the three preceding parts of this question, convert the geometric program in (12) to a convex problem in standard form.