University of Toronto
Department of Electrical and Computer Engineering

ECE367 MATRIX ALGEBRA AND OPTIMIZATION

Problem Set #5
Autumn 2018

Prof. S. C. Draper **Due:** In class on Tuesday, 20 Nov. 2018

Homework policy: Problem sets must be turned in in class on the due date. Late problem sets will not be accepted. See information sheet for full discussion of problem set policy. Some problems are drawn from the course text "Optimization Models" by Giuseppe Calafiore and Laurent El Ghaoui, abbreviated as "OptM", others from the text "Introduction to Applied Linear Algebra", abbreciated "IALA", by Stephen Boyd and Lieven Vandenberghe. An electronic version of the latter is available on the authors' websites. In both cases we indicate the appropriate problem number and name.

Grading: Grading is not uniform. Typically the numerical/design questions are expected to take longer and, correspondingly, are more heavily weighted in the final grade.

Optional problems: Optional problems (if there are any) are located at the end of the problem set and are marked "optional". Although you are not required to do these problems, you are expected to review and understand the solutions to them.

Problem Set #5 problem categories: A quick categorization by topic of the problems in this problem set is as follows:

- Linear programs: 5.1-5.2
- Linear and quadratic programs: 5.3-5.4
- Applications in control and finance: 5.5-5.6

Problem 5.1 (An optimal breakfast)
OptM Problem 9.5. In your solution both state the form of the optimization problem you want to solve and solve it, e.g., using Matlab and CVX. In your solution specify:

(a) The optimal variable x^*.

(b) The optimum value p^*.

(c) Remember to attach your code.

To help get you started we include below a snippet of `CVX` code in `MATLAB` that solves the "meat and potatoes" example from class. (See PS01 for instructions on getting the `CVX` toolbox.)

```
%%% Example done in class
%%% Two variables: pounds purchased of  meat or potatoes
%%% Cost vector c: $1 per pound mean and $0.25 per pound potatoes
%%% Data matrix D: grams carbs/fiber/protein (rows) per pound meat/potatoes (cols)
%%% Constraint vector R: daily reqirements grams carbs / fiber / protein

c = [1 0.25];
D = [40 200; 5 40; 100 20];
R = [400; 40; 200];

cvx_begin
variable x(2)
cost = c*x;
carbs = D(1,:)*x;
fibre = D(2,:)*x;
protein = D(3,:)*x;

minimize(cost)
subject to
carbs >= 400;
fibre >= 40;
protein >= 200;
x >= 0;
cvx_end
```

Problem 5.2 (Standard form LPs)
In class we introduced linear programs to be mathematical programs of the form (in the objective $c^T x + d$ we also included the bias term d which we ignore now):

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax = b, \\ & Gx \geq h \end{aligned} \tag{1}$$

A "standard form" linear program is of the form:

$$\begin{aligned} \underset{\bar{x}}{\text{minimize}} \quad & \bar{c}^T \bar{x} \\ \text{subject to} \quad & \bar{A}\bar{x} = \bar{b}, \\ & \bar{x} \geq 0 \end{aligned} \tag{2}$$

Any LP of the form (1) we introduced in class can be manipulated into an "equivalent" linear program of the standard form (2). An "equivalent" program is a different mathematical program from which the optimal p^* and x^* of the original program can be recovered. Two insights underlie the transformation: (i) any $x \in \mathbb{R}$ can be expressed as $x = x^+ - x^-$ where both $x^+ \geq 0$ and $x^- \geq 0$, (ii) any inequality $a \leq b$ can be transformed into an equality constraint through the introduction of a nonnegative "slack" variable $s \geq 0$ as $a + s = b$.

(a) Use observations (i) and (ii) to show how one can manipulate any LP of form (1) into an "equivalent" LP of form (2), specifying $\bar{c}$, $\bar{A}$, and $\bar{b}$.

(b) Show how you can recover p^* and x^* from $\bar{x}^*$.

(Note: Following observations (i) and (ii) the reformulation should follow very quickly. This is not suppose to be a trickly question. But, some software packages require standard form, so it can be important to know how quickly to reformulate your problem.)

Problem 5.3 (Formulating problems as LPs and QPs)
OptM Problem 9.1. Do the problem for objective functions f_1, f_2, f_4, and f_5. (I.e., skip objective function f_3.)

Problem 5.4 (A slalom problem)
OptM problem 9.2. In particular, do the following:

(a) State an optimization problem that solves for the minimal path. Let (x_i, z_i) denote the (horizontal, vertical) coordinates of your skier at time i. Note that x_i is given but you need to solve for z_i. (This is what is meant by part 1.)

(b) In `MATLAB` (or python, etc.) design a routine that solves the optimization problem you state in part (a) using numerical tools, e.g., `CVX`, using the data tabulated in Table 9.3. (This is equivalent to part 2.) Turn in your code.

(c) Report your (x_i^*, z_i^*) for $i \in \{0, 1, \ldots 6\}$ that you found in part (b).

(d) Report the optimal (minimal) path length p^* that you found in part (b).

(Note: It may be helpful–though it's not crucial–to know that one can define constraints recursively in `MATLAB` + `CVX` using a `for` loop.)

Problem 5.5 (Optimal control of a unit mass, new norms)
In an earlier assignment you solved an optimal control problem with a quadratic (i.e., $\|\cdot\|_2^2$) objective where the force applied was piece-wise constant, specified by a force vector $p = (p_1, \ldots, p_{10})$, and the goal was to move a mass from being at rest at the origin at time zero to being at rest at the unit position at time 10. In this problem we consider the same setup but with two different objectives, the ℓ_1 and ℓ_∞ norms, i.e.,

$$\|p\|_1 = \sum_i |p_i|, \quad \text{and} \quad \|p\|_\infty = \max_i |p_i|.$$

The ℓ_1 norm can serve as a proxy for fuel consumption, the ℓ_∞ norm tries to minimize the *peak* force used.

(a) First consider the problem of minimizing $\|p\|_1$ using the same setup as in the previous assignment. Find the optimal solution using, e.g., `MATLAB` and the `CVX` toolbox. (If using `MATLAB` without `CVX` you may find the `MATLAB` command `linprog` useful.) Plot the optimal force, position, and velocity. What do you observe about the form of the optimization variables at the optimum, how do they contrast to those of the ℓ_2 solution? This solution is sometimes called *bang-bang* control. Does that terminology seem to match what you have observed about the solution?

(b) Repeat part (a) for the $\|\cdot\|_\infty$ minimization problem. How does the ℓ_∞ solution compare to the ℓ_1 and ℓ_2 solutions? Does the solution make sense?

(c) Include your code with your assignment.

Further, you can also repeat the second part of the earlier problem – where the mass also is required to be at the origin at time 5 – for the new norms. But this is optional.

Problem 5.6 (Portfolio Design)
In this problem we consider a classic approach to investment known as "Markowitz portfolio optimization." The idea is that there is often a risk/reward tradeoff in investing that must be managed. The QP that we discuss in this problem is one way to manage for that risk.

Say there are n stocks in which you can invest. We consider a single investment period (for convenience one year). Your must determine an investment strategy which boils down to an allocation of your funds p across the n stocks. Normalizing your wealth to one unit, $\sum_{i=1}^{n} p_i = 1$ (you must invest all your portfolio) and $p_i \geq 0$ for all i (you can't "short" stocks).

There are two pieces of knowledge you have: the expected return of each of the n stocks, and the variability in those returns. As we next discuss, the former is parameterized by the vector $\bar{x}$ and the latter by the matrix Σ. If you research tells you that the annual mean return of stock i is 35% then, if you invest \$1 in stock i on 1 January, your expected investment on 31 December will be worth $\bar{x}_i = \$1.35$. Of course there is variability about this return and we denote by x_i the actual value of your investment, so $E[x_i] = \bar{x}_i$. The variability is denoted by the variance in the stock $E[(x - x_i)^2] = \Sigma_{ii}$. Stocks are correlated so their joint variability is encapsulated by their covariance $E[(x - x_i)(x - x_j)] = \Sigma_{ij}$. We stack the covariance into the $n \times n$ matrix Σ.

For this problem we consider four stocks, $n = 4$. The returns of the 4 stocks are shown on the left-hand table below, which the covariance in the stocks is shown in the right-hand table below. In other words, suppose that you invest \$1 in each stock at the start of the year. Then, $\mathbb{E}[x] = \overline{x} = [1.1 \;\; 1.35 \;\; 1.25 \;\; 1.05]^T$, and $\mathbb{E}[(x - \overline{x})(x - \overline{x})^T] = \Sigma$.

IBM	10%
Google	35%
Apple	25%
Intel	5%

	IBM	Google	Apple	Intel
IBM	0.2	−0.2	−0.12	0.02
Google	−0.2	1.4	0.02	0
Apple	−0.12	0.02	1	−0.4
Intel	0.02	0	−0.4	0.2

We wish to design a portfolio (i.e., the proportion of money invested in each company) to minimize the variance of the investment subject to some fixed minimum expected return $r_{\min}$. The variance of an investment allocation p is $p^T \Sigma p$. (Observe that this expression already arose in the course when we discussed the "sample variance" in the derivation of principal components analysis.)

(a) Formulate the optimization problem as a quadratic programming problem. Plot the tradeoff curve between the variance and the expected return $r_{\min}$ as you vary the lower bound on expected return $r_{\min}$ to plot the risk-return tradeoff curve from one extreme to the other..

(`MATLAB` hint: If you are not using `CVX` you may find the `MATLAB` routine `quadprog` useful.)

(b) Plot the composition of the portfolio as you move from one extreme of the risk-return tradeoff curve to the other extreme. Comment on the benefit of diversification.