Name:________________________________

PRINCETON UNIVERSITY

ORF 363/COS 323
Final Exam, Fall 2016

JANUARY 18, 2017

Instructor:
A.A. Ahmadi

AIs:
B. El Khadir, G. Hall, Z. Li,
K. Wang, J. Ye, J. Zhang

1. Please write out and sign the following pledge on top of the first page of your exam:
"I pledge my honor that I have not violated the Honor Code or the rules specified by the instructor during this examination."

2. Don't forget to write your name on the exam. Make a copy of your solutions and keep it.

3. The exam is not to be discussed with *anyone* except possibly the professor and the TAs. You can only ask *clarification questions*, and only as *public* (and preferably non-anonymous) questions on Piazza. No emails.

4. You are allowed to consult the lecture notes, your own notes, the reference books of the course as indicated on the syllabus, the problem sets and their solutions (yours and ours), the midterm and its solutions (yours and ours), the practice midterm and final exams and their solutions, all Piazza posts, but *nothing else*. You can only use the Internet in case you run into problems related to MATLAB or CVX (or if you want to recognize paintings).

5. You are allowed to refer to facts proven in the notes or problem sets without reproving them.

6. For all problems involving MATLAB or CVX, show your code. The MATLAB output that you present should come from your code.

7. Unless you have been granted an extension because of overlapping finals, the exam is to be turned in on Friday (January 20, 2017) at 9 AM in the instructor's office (Sherrerd 329). If you cannot make it on Friday and decide to turn in your exam sooner, or if your deadline is different under the rules of the exam, you have to drop your exam off in the ORF 363 box of the ORFE undergraduate lounge (Sherrerd 123). If you do that, you need to *write down the date and time on the first page of your exam and sign it.* You can also submit the exam electronically on Blackboard as a single PDF file.

8. Good luck!

Grading

Problem 1	20 pts	
Problem 2	20 pts	
Problem 3	20 pts	
Problem 4	20 pts	
Problem 5	20 pts	
TOTAL	100	

Problem 1: Feeling confident about the exam? Read on.

Consider the following problem (that may sound familiar): you have T hours to take a final exam which has n problems worth $p_1, p_2, \ldots, p_n$ points respectively. You have glanced over the problems and (because you know your own knowledge of the subject so well) have determined the questions will take $t_1, t_2, \ldots t_n$ hours of your time respectively. The catch is that you do not have the time to do all problems in the T hours allotted to you (i.e., $\sum_{i=1}^n t_i > T$), and so you would like to know how to allocate your time to get the highest grade possible. The optimization problem you need to solve for this purpose of course depends on the grading scheme:

- *no partial credit*— in this case, you would need to solve the integer program

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & \sum_{i=1}^n p_i x_i \\ \text{s.t.} \quad & \sum_{i=1}^n t_i x_i \leq T \\ & x_i \in \{0,1\}, \forall i = 1, \ldots, n. \end{aligned} \tag{1}$$

  Here, each entry x_i of x specifies whether you should work on question i ($x_i = 1$) or whether you should not ($x_i = 0$) as there is no point in working on part of a problem.

- *partial credit*— under this model, if you spend a fraction x_i of your required time t_i on problem i, you get a proportional $x_i p_i$ points on it. Hence, to maximize your grade, you have to solve the following problem:

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & \sum_{i=1}^n p_i x_i \\ \text{s.t.} \quad & \sum_{i=1}^n t_i x_i \leq T \\ & 0 \leq x_i \leq 1, \forall i = 1, \ldots, n. \end{aligned} \tag{2}$$

  Interestingly, notice that problem (2) is the linear programming (LP) relaxation of problem (1).

1. Consider first the case where the graders are incredibly wonderful people and give partial credit.[1] Your task is to show in part 1.b below that the optimal solution to LP

[1]Coincidentally, ORF 363 AIs will be giving partial credit on this final.

(2) has a simple and intuitive structure: you rank the problems in decreasing order of their "bang for the buck" (i.e., p_i/t_i). Then, starting from the top, you do the problems one by one until you reach a problem that you cannot finish. You spend the rest of your time on completing that problem partially.

(a) Use CVX to solve an instance of problem (2), where the final is a 48-hour take-home containing 5 problems, each worth 20 points, and respectively requiring 20 hours, 8 hours, 16 hours, 15 hours, and 10 hours of your time. Is the solution you obtain in agreement with the description we gave above?

(b) We assume for simplicity that there are no ties among the ratios p_i/t_i and that $0 < t_i \leq T$, $\forall i$. Suppose without loss of generality that $p_1/t_1 > p_2/t_2 > \ldots > p_n/t_n$. Since we know $\sum_{i=1}^n t_i > T$, there must exist a positive integer $r < n$ such that

$$t_1 + \ldots + t_r \leq T < t_1 + \ldots + t_r + t_{r+1}.$$

Under these conditions, use the optimality condition for convex optimization problems (seen in class) to show that the solution to (2) will be given by the vector $x^* = (x_1^*, \ldots, x_n^*)^T$, where $x_i^* = 1$ for $i = 1, \ldots, r$, $x_{r+1}^* = (T - \sum_{k=1}^r t_k)/t_{r+1}$, and $x_i^* = 0$ for $i = r+2, \ldots, n$.

2. Consider now the case where no partial credit will be offered. Unfortunately, the binary constraint on x makes problem (1) difficult to solve. You decide to go with the following strategy instead: you sort the problems on the exam by decreasing ratio p_i/t_i. You start on the problem with the highest p_i/t_i and work down the list with the following rule: if you have enough time to do a problem in its entirety, you do it. Otherwise, you skip over to the next problem. Show with an example (i.e., an instance $p_1, \ldots, p_n, t_1, \ldots, t_n, T$ satisfying the assumptions of part 1.b) that, in general, this strategy does not produce the optimal solution to (1).

Problem 2: Hands off my paintings

You are the newly appointed head of Security at the Princeton Art Museum and your task is to protect five particularly valuable paintings from theft at nighttime. These paintings are all put on the same wall with the following positions (see figure):

Painting i	x_1^i	y_1^i	x_2^i	y_2^i
1	40	335	160	237
2	70	235	172	85
3	190	385	310	259
4	320	365	440	219
5	280	185	400	90

Here, the origin is taken to be at the bottom left corner of the wall, (x_1^i, y_1^i) are the coordinates of the top left corner of painting i, and (x_2^i, y_2^i) are the coordinates of its bottom right one.

Figure 1: The museum wall

Because you are lazy and don't want to stand in front of the paintings all night, you install a detector that monitors them on the wall and triggers an alarm as soon as one of them moves. The detector's coverage area is a disk and the amount of energy it consumes is proportional to the area of the disk. (Recall that a disk of center $u \in \mathbb{R}^2$ and radius $r > 0$ is given by $B(u, r) = \{z \in \mathbb{R}^2 \mid ||z - u||_2 \leq r\}$.)

1. Use CVX to find a disk of minimum area that covers the *entirety* of all paintings. Justify your approach. Report the center and the radius of this disk.

2. Plot the circle corresponding to the boundary of the optimal disk on top of the paintings. You can use the function `Circledraw_wall` that is provided and the following code to load the image of the wall and draw a circle centered at (x, y) with radius r:

```
wall = imread('wall_with_paintings.png');
imshow(wall)
```

```
hold on
Circledraw_wall(x, y, r, 'red')
```

3. Name the artists who painted the paintings that touch the boundary of the optimal disk. (We are sure our ORF 363 students know their painters, but just in case, the following names may help: Matisse, Da Vinci, Monet, Picasso, Van Gogh, and Google.)

4. What is the largest integer k for which the following statement is true?
"The boundary of the minimum-area disk containing any set of $m > 1$ points in $\mathbb{R}^2$ must touch at least k of them."
Justify your answer.

Problem 3: Least absolute deviations

For a set of observations $\{(x_i, y_i)\}_{i=1}^m$ where $(x_i, y_i) \in \mathbb{R}^2$, linear regression aims to find $(a, b) \in \mathbb{R}^2$ such that $ax_i + b$ best predicts y_i (in some sense). How much we lose by picking one pair (a, b) over another is encoded in the notion of a *loss function*. This is a function $L : \mathbb{R}^2 \to \mathbb{R}$ mapping each pair (a, b) to a real number, with lower values corresponding to better predictions. The best predictor $(\hat{a}, \hat{b})$ is hence chosen to be an optimal solution to the problem $\underset{(a,b)\in\mathbb{R}^2}{\min.} L(a, b)$.

In class, we have seen one popular loss function, namely the *least squares loss function*

$$L_{LS}(a, b) = \sum_{i=1}^{m} [y_i - (ax_i + b)]^2.$$

In this problem, we study another popular loss function

$$L_{LAD}(a, b) = \sum_{i=1}^{m} |y_i - (ax_i + b)|,$$

known as the *least absolute deviation loss function*.

1. Show that $L_{LAD}(a, b)$ is a convex function.

2. Show that the problem $\underset{(a,b)\in\mathbb{R}^2}{\min.} L_{LAD}(a, b)$ can be reformulated as a linear program. (*Hint*: You may want to introduce new decision variables.)

3. Below are a set of observations $\{(x_i, y_i)\}_{i=1}^7$:

x_i	0.75	1.36	2.81	4.32	5.41	5.49	6.76
y_i	1.69	12.40	4.86	7.19	8.15	8.65	9.58

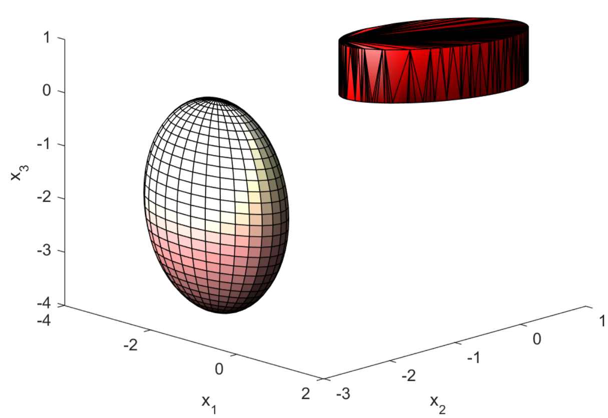
(a) Plot these points on a plane. You should see that all of them except for one are close to a line. This distinguished point is called an *outlier* and can correspond, e.g., to a faulty measurement.

(b) Use CVX to find the least squares estimate, i.e., the minimizer of the least squares loss function. Use CVX and your linear programming formulation to find the least absolute deviation estimate, i.e., the minimizer of the least absolute deviation loss function. Report the optimal solutions. Plot the two fitted lines that correspond to these solutions, together with the points, on one figure.

(c) Which method is more robust against the effect of the outlier? Give a brief and intuitive explanation as to why this is.

Problem 4: Distance between an ellipsoid and an elliptic cylinder

The distance between two sets S_1 and S_2 is the closest distance between any two points one taken from each set.



1. Consider an elliptic cylinder S_1, given by

$$S_1 = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 \mid \begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & 1 & 0.5 \\ x_2 & 0.5 & 1 \end{pmatrix} \succeq 0, \quad 0 \leq x_3 \leq 1 \right\},$$

and an ellipsoid S_2, given by $S_2 = \{x \in \mathbb{R}^3 \mid x^T Q x + b^T x + c \leq 1\}$, where

$$Q = \begin{pmatrix} 4/9 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1/4 \end{pmatrix}, \quad b = \begin{pmatrix} 16/9 & 16 & 1 \end{pmatrix}^T, \quad c = 169/9.$$

Formulate the problem of finding the distance between S_1 and S_2 as a convex optimization problem. Use CVX to find this distance and report the value.

2. Using the commands `hold on` and `plot3`, plot on the MATLAB figure (given in `distance_computation.fig`) the two points that achieve the minimum distance computed in the previous question as well as a line segment connecting them.

3. An extremist claims that for any two bounded[2] convex sets S_1 and S_2 which do not intersect, there exists an extreme point in S_1 or in S_2 where the minimum distance between S_1 and S_2 is achieved. Prove the extremist wrong. (A correct picture is enough.)

Problem 5: Minimizing homogeneous polynomials

We are given a homogeneous polynomial

$$f(x_1, x_2) = 2x_1^4 - 6x_1^3 x_2 + 5x_1^2 x_2^2 - 2b x_1 x_2^3 - a x_2^4, \tag{3}$$

where a and b are parameters. Find the largest scalar a for which there exists a scalar b such that the optimal value of the problem of minimizing f over $\mathbb{R}^2$ is finite (i.e., not $-\infty$). (*Hint:* First show that if the optimal value is finite, then it must be equal to zero.)

[2] A set $S \subset \mathbb{R}^n$ is bounded if there exists a scalar M such that $||x||_2 \leq M$, for all $x \in S$.