

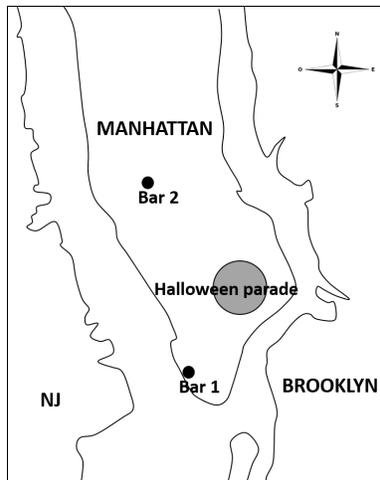
Instructor: A.A. Ahmadi

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

Due at 1:30pm in class, on November 10th, 2016

Problem 1: Halloween in Manhattan¹

It's October 31st and you want to be close to the Halloween parade in Manhattan but also not too far from either of two bars in town where your crush is supposed to show up tonight. You don't know which one (s)he is going to go to yet, but you are checking his/her Facebook status non-stop.



Bar 1 is located at coordinates $(-2, -3)$, bar 2 is in location $(-3, 4)$, and the parade is happening in a circle of radius 1 km centered at $(0, 0)$. All the coordinates are in kilometers (km).

1. Write a convex optimization problem that finds an optimal location for you to wait at, which (i) minimizes the amount of walking you have to do in worst case to get to the right bar, and (ii) allows you to be within 0.5 km walking-distance of the parade. Note that in Manhattan *you can only walk West↔East or North↕South*. Justify why the problem is convex, and solve it using CVX. Give us the coordinates of the optimal solution.

¹Beware that parts of this problem are different than what you may have seen in the practice midterm.

2. Using MATLAB, plot a circle of radius 1 centered at $(0,0)$ representing the parade. The goal is now to plot on the same drawing the set of points S within 0.5 km walking-distance of the parade. We help you do this by breaking down the question into a couple of steps.
 - (i) Consider the half-line that starts at the origin and forms an angle α with the \vec{x} axis. Write a convex optimization problem that finds the point on this line that is furthest away from the origin while still belonging to S .
 - (ii) For $\alpha = 0 : \frac{2\pi}{100} : 2\pi$, solve the problem above using CVX and store the resulting optimal point. Now draw the set S using the function `fill`. Add to this drawing the optimal location that you should wait at from part 1.

Problem 2: Theory-applications split in a course. (Courtesy of Stephen Boyd)

A professor teaches a course with 24 lectures, labeled $i = 1, \dots, 24$. The course involves some interesting theoretical topics, and many practical applications of the theory. The professor must decide how to split each lecture between theory and applications. Let T_i and A_i denote the fraction of the i th lecture devoted to theory and applications, for $i = 1, \dots, 24$. (We have $T_i \geq 0$, $A_i \geq 0$, and $T_i + A_i = 1$.)

A certain amount of theory has to be covered before the applications can be taught. We model this in a crude way as

$$A_1 + \dots + A_i \leq \phi(T_1 + \dots + T_i), \quad i = 1, \dots, 24, \quad (1)$$

where $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is a given nondecreasing function. We interpret $\phi(u)$ as the cumulative amount of applications that can be covered, when the cumulative amount of theory covered is u . We will use the simple form $\phi(u) = a \max\{0, u - b\}$ with $a, b > 0$, which means that no applications can be covered until b lectures of the theory is covered; after that, each lecture of theory covered opens the possibility of covering a lectures on applications.

The theory-applications split affects the emotional state of students differently. We let s_i denote the emotional state of a student after lecture i , with $s_i = 0$ meaning neutral, $s_i > 0$ meaning happy, and $s_i < 0$ meaning unhappy. Careful studies have shown that s_i evolves via a linear recursion (dynamics)

$$s_i = (1 - \theta)s_{i-1} + \theta(\alpha T_i + \beta A_i), \quad i = 1, \dots, 24,$$

with $s_0 = 0$. Here α and β are parameters (naturally interpreted as how much the student likes or dislikes theory and applications, respectively), and $\theta \in [0, 1]$ gives the emotional

volatility of the student (i.e., how quickly he or she reacts to the content of recent lectures). The student's *cumulative emotional state* (CES) is by definition $s_1 + \dots + s_{24}$. This is a measure of his/her overall happiness throughout the semester.

Now consider a specific instance of the problem, with course material parameters $a = 2$, $b = 3$, and three groups of students, with emotional dynamics parameters given as follows:

	Group 1	Group 2	Group 3
θ	0.05	0.1	0.3
α	-0.1	0.8	-0.3
β	1.4	-0.3	0.7

Your job is to plan (four different) theory-applications splits that respectively maximize the CES of the first group, the CES of the second group, the CES of the third group, and, finally, the minimum of the cumulative emotional states of all three groups. (Hint: you would need to appropriately reformulate constraint (1) to end up with a convex optimization problem.) Report the numerical values of the CES for each group, for each of the four theory-applications splits (i.e., fill out the following table):

	Group 1	Group 2	Group 3
Plan 1			
Plan 2			
Plan 3			
Plan 4			

For each of the four plans, plot T_i as well as the emotional state s_i for all three groups, versus i . (So you should have four figures with four curves on each.) These plots show you how the emotional states of the students change as the amount of theory varies.

Problem 3: Newton fractals

The sensitivity of Newton's method to initial conditions is beautifully demonstrated using plots over the complex plane known as *Newton fractals*. You may have seen a picture of Newton fractals in lecture notes, and now your task in this problem is to produce the Newton fractal associated with the critical points of $f(z) = z^5 - 5z$. The steps below are only meant to help you do this—there is no grade assigned to them.

1. Note that z is a complex number throughout this exercise. Verify that the critical points of f , i.e., the roots of f' are $z_1 = 1, z_2 = -1, z_3 = i, z_4 = -i$.

2. Discretize $[-1, 1] \times [-1, 1]$ using intervals of length 0.0031. We recommend that you define the sequence of points $x = -1 : 0.0031 : 0.999$, and $y = -0.999 : 0.0031 : 1$ to avoid certain numerical issues. For each point (x_j, y_l) in your discrete grid, apply Newton's method with $z_{jl} = x_j + iy_l$ as its initial point.

Hint: Consider using the `meshgrid` function to create your grid (which will contain 645^2 points on the complex plane). To run the Newton method, we recommend using matrix operations in MATLAB instead of for-loops. Finally, you may set the maximum number of iterations for Newton's method to 200 for simplicity.

3. Map each of the critical points of f to some color code; e.g., $z_1 \leftrightarrow 1, z_2 \leftrightarrow 2, z_3 \leftrightarrow 3, z_4 \leftrightarrow 4$. Then, to each initial condition (i.e., to each (x_j, y_l) on the grid) assign one of the four color codes based on the root that the iterations are converging to (up to some tolerance error, say, $\epsilon = 0.01$). Depending on how you discretize, for some initial conditions the algorithm may not converge. In that case, assign color code 0 to that particular (x_j, y_l) . You will obtain a 645×645 matrix of color codes representing your Newton fractal. Plot it using the `imagesc` function.

Submit a print out of your code and plot. To get credit, your code must produce the plot.

Problem 4: Finding the global minimum of a univariate function.

In this problem, we would like to find the global minimum of the function

$$f(x) = (x - 3)^4 + (2x - 1)^2 - 10.$$

1. Is f convex?
2. *Bisection method:* Starting from the interval $[1, 4]$, carry out two steps of the bisection method to minimize f . What is the interval that you end up with?
3. *Newton's method:* Starting from the initial point $x_0 = 1$, carry out two steps of the Newton method to minimize f . What is the point that you end up with?

For both bisection and Newton, you have the freedom of doing the calculations by hand or by Matlab.