9/1 Course Intro

Me: CI @ NYU Courant

My info: sampotter.github.io ← Syllabus

Office: WWH 1104

Office hours: TBD

TA: Mariana Martinez-Aguilar

Office hours: TBD

Recitation: 3:30 – 4:45 CIWW (starting 9/9)

- The course: Linear & Nonlinear Optimization

- What is optimization? modeling + math + algorithms + implementation + application...

- What is modeling?

application/problem $\xrightarrow{\text{model the problem}}$ model ← (choose among several models?)

model $\xrightarrow{\text{analyze the model}}$ math

math → numerical methods ← decide what discretizations are appropriate, what algorithms to use, how to get it onto a computer...

numerical methods → application/problem ← "run the model"/ "solve the problem".. feed back into your decision-making loop..

- This is the basic process... can be very open-ended

- Optimization is a type of modeling focused on solving extremal problems: find the biggest ~~ob~~ or smallest object among a set...

Variations: find the thing which minimizes or maximizes this function over a set of possible argument...

- minimize the cost/loss/regret
- maximize the efficiency/~~[illegible]~~ performance

← jargon!

- More jargon: optimize = train = learn

ML mumbo jumbo

- How we write it mathematically:

$$(1) \quad \left[\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in X \end{array}\right.$$

← cost function

← domain of optimization problem

Note: $\min_{x \in X} f(x) = -\max_{x \in X} -f(x)$] inline notation

so it is OK to just talk about minimization problems.

- (1) is very abstract. There are many ways of making it concrete. ~~[illegible]~~ When we make (1) concrete, we have identified a model (see prev. page).

We will focus on ~~three~~ four big (and very important — really, the most common) families of models of optimization problems:

a) linear programs

b) dynamic programs ← as we go...

c) unconstrained nonlinear programs

d) constrained nonlinear programs

We will also learn about how convexity relates to all this.

- This is a math class, so we will start by explaining some of this terminology.

- Historical aside: the word programming is used to describe different subdisciplines of optimization (a.k.a. "mathematical programming"); e.g. "linear programming", "dynamic programming". Nowadays, the word programming is more frequently used to talk about computer programming and has taken on a new meaning.

- Linear programming: for linear programs (LPs), the cost function is linear, and the domain is described by a set of linear equality constraints and linear inequality constraints. More jargon... what does it mean?

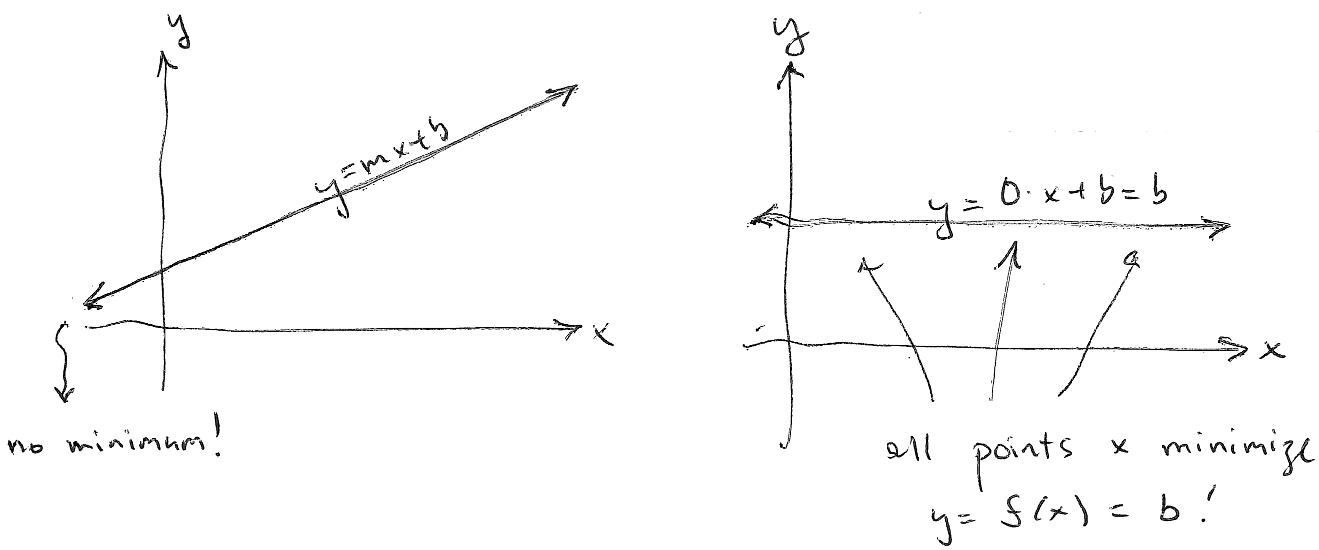
Let's discuss our setting a little more... Our goal will be to minimize (or maximize) a scalar-valued function over some domain. Single-variable calculus gives us tools that are good enough for optimizing over $\mathbb{R}$ (the real numbers). So we will focus on cost functions $\underline{\underline{f}}: \mathbb{R}^n \to \mathbb{R}$ mapping a vector $\underline{x} \in \mathbb{R}^n$ to a scalar $y = f(\underline{x})$.

For LPs, since $f(x)$ is linear, then it has the form $f(x) = \sum_{i=1}^{n} c_i x_i + d = c^T x + d$.

- Note: play fast and loose w/ scalar vs. vector notation... you will learn to be able to differentiate based on ctx.

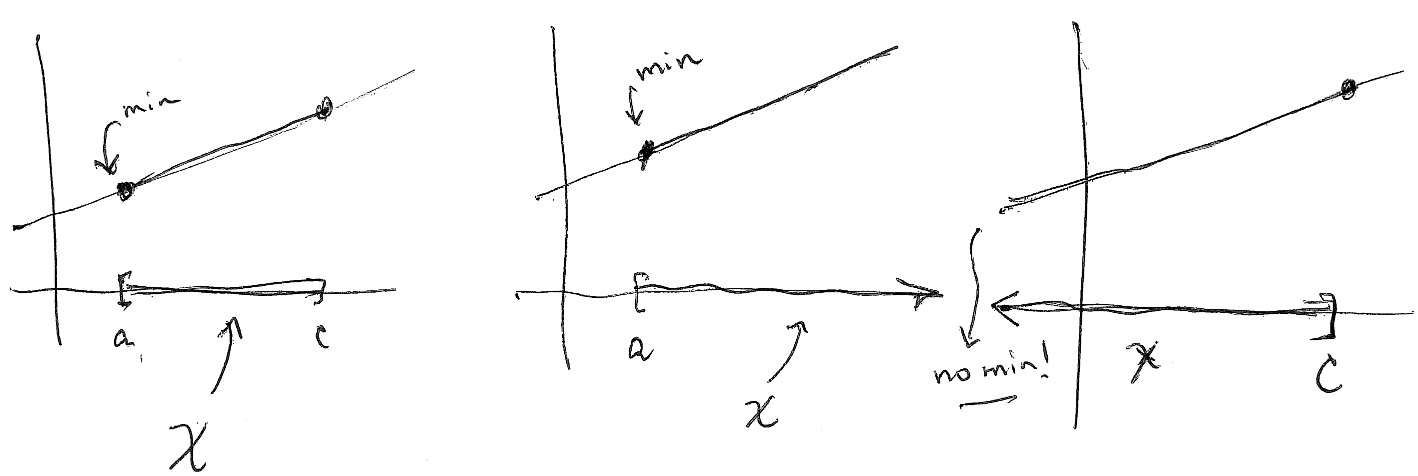
This gives the cost function. In 1D, we get $y = mx + b$, a line. What happens if we try to minimize it?

Two cases:



We can make this more interesting by including some inequality constraints:

$$\min \; f(x) = mx + b$$
$$\text{st} \quad a \le x \le c \qquad (\text{i.e. "} x \in X \text{" from before})$$



Again, still not that interesting... Let's try $\mathbb{R}^n$, $n > 1$...
Say, $n = 2$...

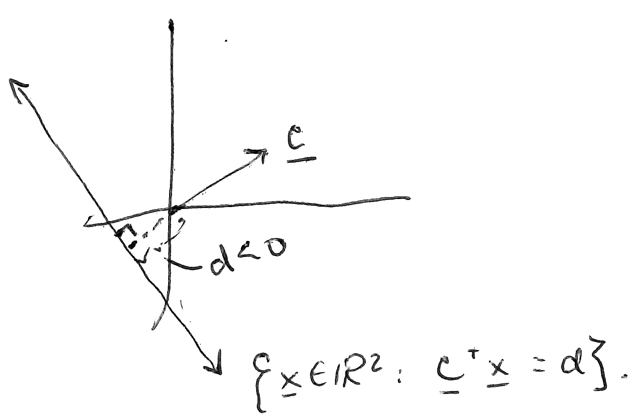
- $\underline{\text{LP w/ } n = 2}$: What does this look like?

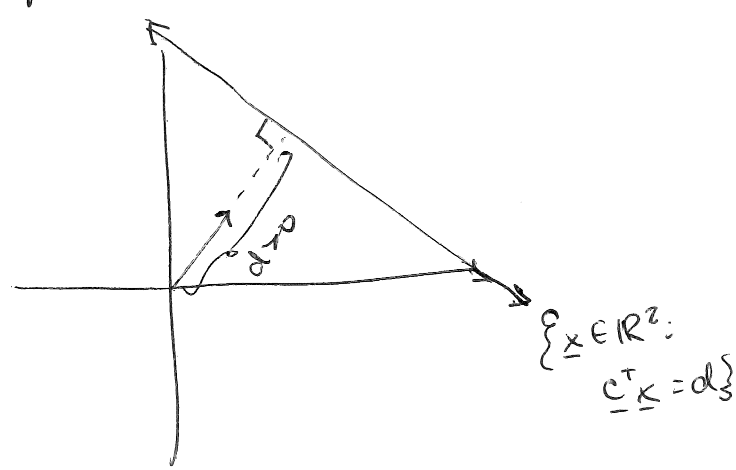
$$\text{minimize} \quad m_1 x_1 + m_2 x_2 \quad \Big(= \underbrace{f(x_1, x_2)}_{\text{cost function}} \Big)$$

$$\text{subject to} \quad \left.\begin{aligned} a_{11} x_1 + a_{12} x_2 &\le b_1 \\ a_{21} x_1 + a_{22} x_2 &\le b_2 \\ &\vdots \\ a_{k1} x_1 + a_{k2} x_2 &\le b_k \end{aligned}\right] \quad \underline{\text{What are these?}}$$

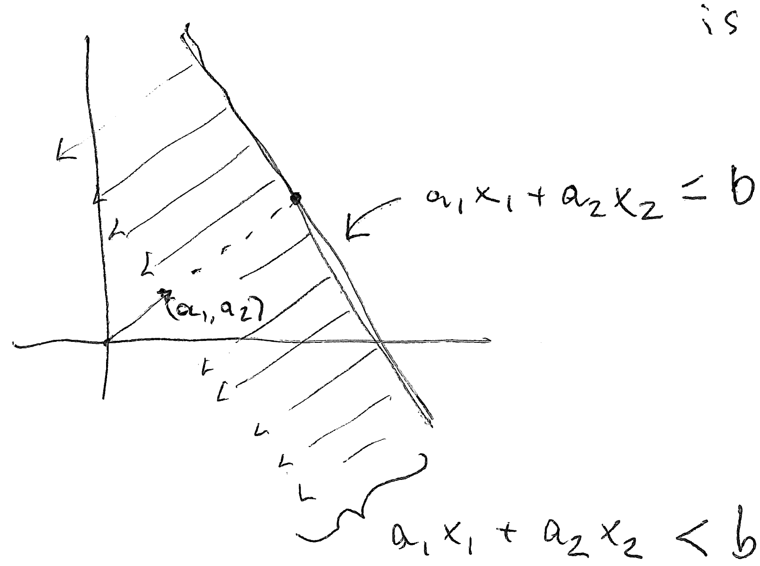
Recall from multivariable calculus the implicit equation for a line in $\mathbb{R}^2$: $\underline{c_1 x_1 + c_2 x_2 = d}$.

Let $\underline{c} = (c_1, c_2)$, $\underline{x} = (x_1, x_2)$ so to get $\underline{c}^T \underline{x} = d$, and assume $\|\underline{c}\|_2$ (the 2-norm of $\underline{c}$, its length) is just $\underline{1}$. So $\underline{c}$ is a unit vector. Then "$\underline{c}^T \underline{x}$" $= \text{comp}_{\underline{c}} \underline{x}$, "the component of $\underline{x}$ in the $\underline{c}$ direction", or the scalar projection of $\underline{x}$ onto $\underline{c}$. So, it must be the same for all $\underline{x}$ which satisfy the equation, hence:

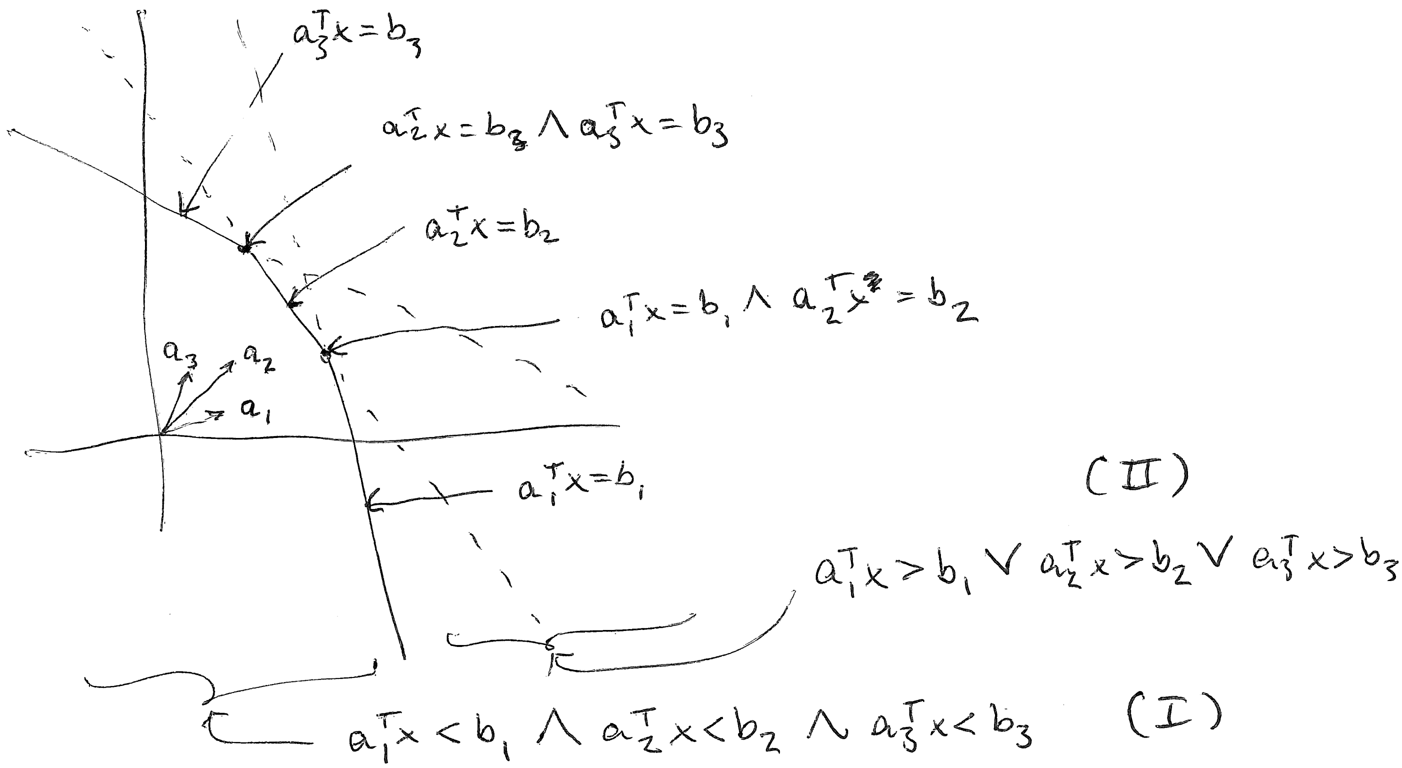




- So, from this, we can see that: $\{(x_1,x_2) \in \mathbb{R}^2 : a_1x_1 + a_2x_2 \leq b\}$ is a half-space:



- If we have multiple such constraints in effect at once:



~~Observations:~~ Questions:

1) any points x such that $a_1^Tx = b_1 \wedge a_2^Tx = b_2 \wedge a_3^Tx = b_3$?
2) we can partition region II into $2^3 - 1$ subsets whose interiors correspond to one "inequality combination"?

Observations : 1 :

~~(1) A system of linear inequalities defines a polyhedral region~~

~~(1)~~ We can stack the inequalities into a matrix. How?

Consider "$a_1^T x = b_1 \wedge a_2^T x = b_2 \wedge a_3^T x = b_3$"

again... This is the same as solving each equation simultaneously... :

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \\ a_{31}x_1 + a_{32}x_2 &= b_3 \end{aligned} \iff \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

or : $Ax = b, \quad A \in \mathbb{R}^{3\times 2}, \quad x \in \mathbb{R}^{2(\times 1)}, \quad b \in \mathbb{R}^{3(\times 1)}$

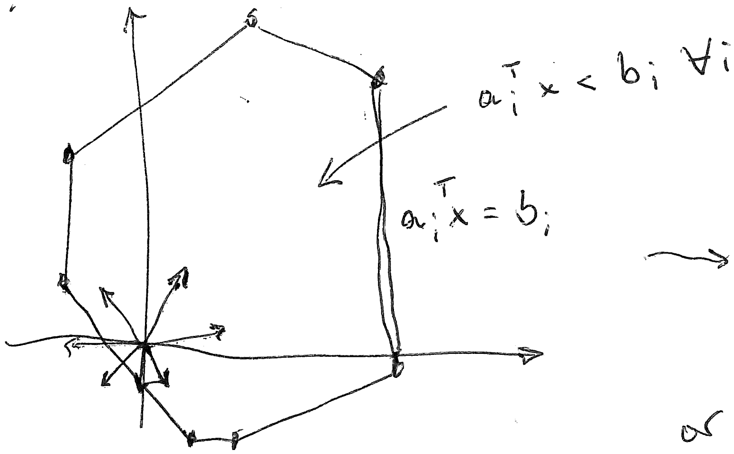
OK. Well now we know why we can't find a point :

1) geometric reason : 3 lines in $\mathbb{R}^2$ in "general position" do not intersect in a single point (rather, there are $\binom{3}{2}$ intersections)

2) linear algebra reason : overdetermined linear system where (from the picture), it is clear that no equations ~~rows~~ are linearly dependent on any others

Observation 2 : a system of linear inequalities forms a polyhedral region ...

Eg:



~~matrix~~ componentwise inequality for vectors

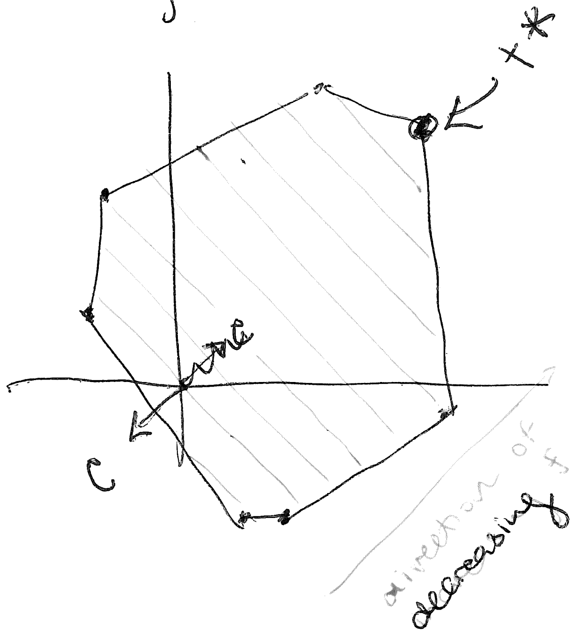
$\rightarrow \boxed{Ax \preceq b}$

or: $\boxed{\mathcal{X} = \{x \in \mathbb{R}^2 : Ax \preceq b\}}$

- OK... but this is a class about optimization... ~~but~~ what function are we optimizing?

Recall: for an LP, minimize linear functions: $f(x) = c^T x$

~~we already saw~~ We already saw that the level sets of $f(x)$ are lines in $\mathbb{R}^2$... so if we ~~plot~~ draw a few of these, along w/ our region $\mathcal{X}$:



Recall also that:

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right)$$

$$= (c_1, c_2) = c$$

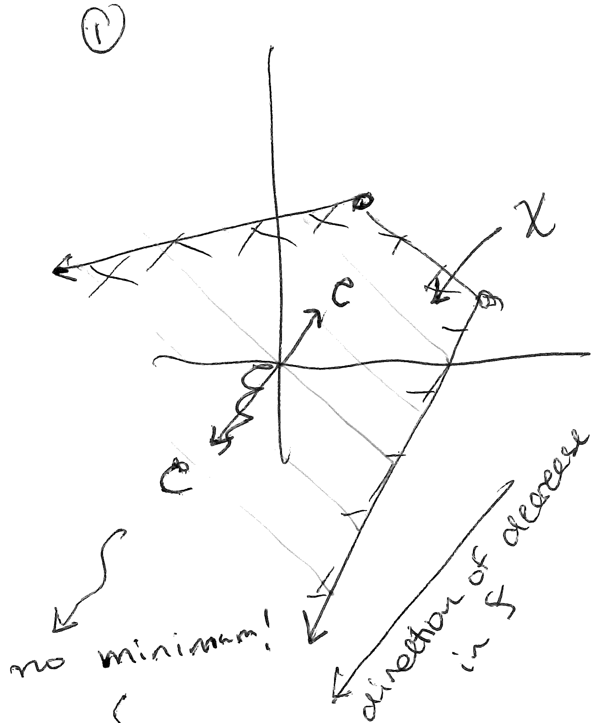
and that the gradient of a function f gives the direction of steepest increase in f at a point.

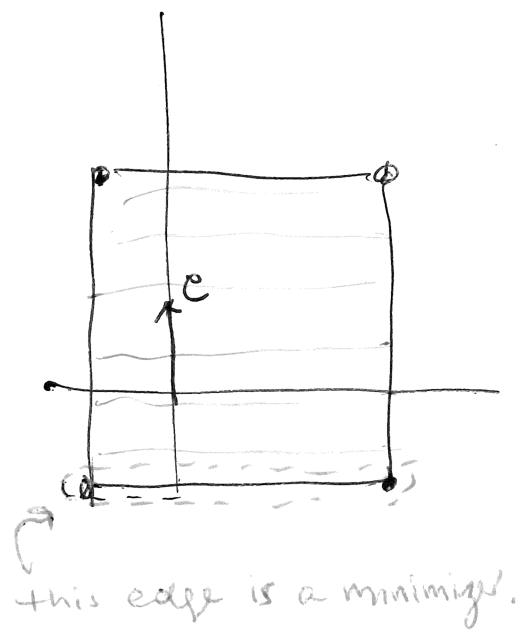
Let $x^* = \arg\min_{x \in \mathcal{X}} f(x)$.

the minimizing argument in the LP

- In this simple case, we can see exactly where x^* must be. It is a vertex of the polygon

- Two cases to watch out for:





$$\min_{x \in \mathcal{X}} f(x) = -\infty !$$

- Linear programs have MANY applications. The ones which are most important are in many variables. This discussion gives a basic picture of LPs and builds some intuition, but you can't visualize an LP ~~with~~ where $\mathcal{X} \subseteq \mathbb{R}^{1,000,000}$.

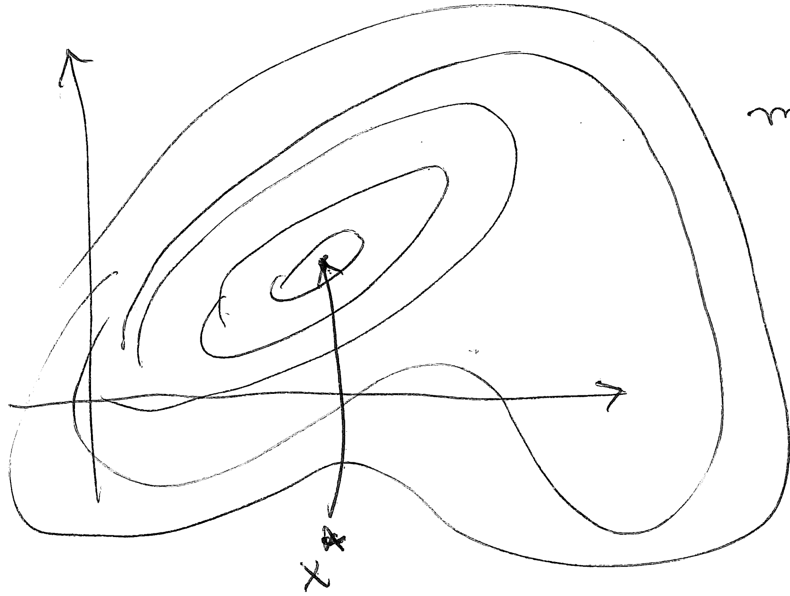
- Note: a polyhedral set in $\mathbb{R}^n$, $n > 3$, $\mathcal{X} = \{x \in \mathbb{R}^n : Ax \leq b\}$ is called a (convex) polytope.

  come back to this...

- We will learn about different algorithms for solving LPs and study some geometry of LPs.

- This is one part of the class... What else will we do?

- ~~Assa~~ Unconstrained nonlinear optimization:



$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \ f(x)$$

nonlinear

Example

bakery:

dozen bagels: 5c flour, 2 eggs, 1c sug

dozen muffins: 4c fl, 4 eggs, 2 sug

bagel: $10/dozen

muff: $12 dozen

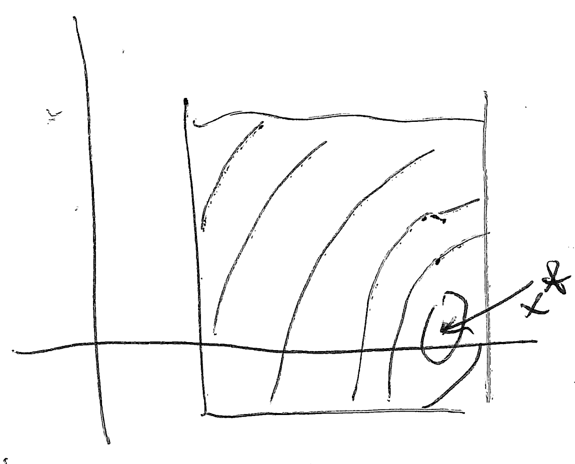
has: 50c sugar, 30 eggs, 20c sugar

bake ⇒ max profits

$$\max \ 10b + 12m$$
$$\text{st} \quad 5b + 4m \le 50$$
$$2b + 4m \le 30$$
$$1b + 2m \le 20$$
$$b \ge 0, \ m \ge 0$$

plot? $Ax \le \cancel{B}\, c$?

- Constrained nonlinear optimization:



- Combinatorial optimization:

shortest paths in graphs, e.g.