24. Nonlinear programming

- Overview
- Example: making tires
- Example: largest inscribed polygon
- Example: navigation using ranges

Laurent Lessard (www.laurentlessard.com)

First things first

The labels **nonlinear** or **nonconvex** are not particularly informative or helpful in practice.

- Throughout the course we studied properties of linear constraints, convex quadratics, even MIPs. We can't expect there to be a rigorous science for "everything else".
- It doesn't really make sense to define something as not having a particular property.
- "I'm an ECE professor" is a very informative statement. But using the label "non-(ECE professor)" is virtually meaningless. It could be a student, a horse, a tomato,...

Important categories

- **Continuous vs discrete:** As with LPs, the presence of binary or integer constraints is an important feature.
- **Smoothness:** Are the constraints and the objective function differentiable? twice-differentiable?
- **Qualitative shape:** Are there many local minima?
- **Problem scale:** A few variables? hundreds? thousands?

This sort of information is very useful in practice. It helps you decide on an appropriate solution approach.

This lecture: examples!

- It doesn't make sense to enumerate all the tips and trick for solving nonlinear/nonconvex problems. Too many!
- Instead, we will look at a few specific examples in detail. Each example will highlight some important lessons about dealing with nonconvex/nonlinear problems.

Example: making tires

- Tires are made by combining rubber, oil, and carbon.
- Tires must have a hardness of between 25 and 35.
- Tires must have an elasticity of at least 16.
- Tires must have a tensile strength of at least 12.
- To make a set of four tires, we require 100 pounds of total product (rubber, oil, and carbon).
  - At least 50 pounds of carbon.
  - Between 25 and 60 pounds of rubber.

Example: making tires

- Chemical Engineers tell you that the tensile strength, elasticity, and hardness of tires made of r pounds of rubber, h pounds of oil, and c pounds of carbon are...
  - Tensile strength $= 12.5 - 0.1h - 0.001h^2$
  - Elasticity $= 17 + .35r - 0.04h - 0.002r^2$
  - Hardness $=$ $34 + 0.1r + 0.06h - 0.3c + 0.01rh + 0.005h^2 + 0.001c^{1.95}$
- The Purchasing Department says rubber costs \$.04/pound, oil costs \$.01/pound, and carbon costs \$.07/pound.

Example: making tires

$$
\begin{array}{rl}
\underset{r,h,c}{\text{minimize}} & 0.04r + 0.01h + 0.07c \\
\text{total:} & r + h + c = 100 \\
\text{tensile:} & 12.5 - 0.1h - 0.001h^2 \geq 12 \\
\text{elasticity:} & 17 + .35r - 0.04h - 0.002r^2 \geq 16 \\
\text{hardness:} & 25 \leq 34 + 0.1r + 0.06h - 0.3c \\
& \qquad + 0.01rh + 0.005h^2 + 0.001c^{1.95} \leq 35 \\
& 25 \leq r \leq 60, \quad h \geq 0, \quad c \geq 50
\end{array}
$$

- Problem is smooth and continuous. Julia: Tires.ipynb
- Fairly typical of something you might encounter in practice. Can we simplify it? Can we learn something useful?

Example: making tires

$$
\begin{aligned}
\underset{r,h,c}{\text{minimize}} \quad & 0.04r + 0.01h + 0.07c \\
\text{total:} \quad & r + h + c = 100 \\
\text{tensile:} \quad & 12.5 - 0.1h - 0.001h^2 \geq 12 \\
\text{elasticity:} \quad & 17 + .35r - 0.04h - 0.002r^2 \geq 16 \\
\text{hardness:} \quad & 25 \leq 34 + 0.1r + 0.06h - 0.3c \\
& \qquad + 0.01rh + 0.005h^2 + 0.001c^{1.95} \leq 35 \\
& 25 \leq r \leq 60, \quad h \geq 0, \quad c \geq 50
\end{aligned}
$$

- Optimal solution is: $(r_\star, h_\star, c_\star) = (45.23, 4.77, 50)$.
- Only tensile constraint is tight!
- Does this mean elasticity and hardness don't matter?

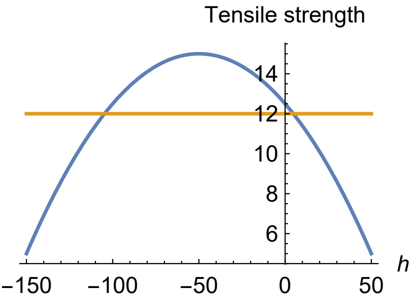
Example: making tires

$$
\begin{aligned}
\underset{r,h,c}{\text{minimize}} \quad & 0.04r + 0.01h + 0.07c \\
\text{total:} \quad & r + h + c = 100 \\
\text{tensile:} \quad & 12.5 - 0.1h - 0.001h^2 \geq 12 \\
\text{elasticity:} \quad & 17 + .35r - 0.04h - 0.002r^2 \geq 16 \\
\text{hardness:} \quad & 25 \leq 34 + 0.1r + 0.06h - 0.3c \\
& \qquad + 0.01rh + 0.005h^2 + 0.001c^{1.95} \leq 35 \\
& 25 \leq r \leq 60, \quad h \geq 0, \quad c \geq 50
\end{aligned}
$$

- Tensile constraint only depends on h.
- Can we simplify it?

Example: making tires

Tensile constraint: $12.5 - 0.1h - 0.001h^2 \geq 12$



- Since $h \geq 0$, only a small range of h is admissible
- If we solve for equality (quadratic formula), the positive solution is $h = 4.77$

We can replace the tensile constraint by $0 \leq h \leq 4.77$.

Example: making tires

$$
\begin{aligned}
\underset{r,h,c}{\text{minimize}} \quad & 0.04r + 0.01h + 0.07c \\
\text{total:} \quad & r + h + c = 100 \\
\text{tensile:} \quad & 0 \le h \le 4.77 \\
\text{elasticity:} \quad & 17 + .35r - 0.04h - 0.002r^2 \ge 16 \\
\text{hardness:} \quad & 25 \le 34 + 0.1r + 0.06h - 0.3c \\
& \qquad + 0.01rh + 0.005h^2 + 0.001c^{1.95} \le 35 \\
& 25 \le r \le 60, \quad c \ge 50
\end{aligned}
$$

- We can't independently choose r, h, c...
- Let's eliminate r. Replace r by $(100 - h - c)$.

Example: making tires

Objective function: $0.04r + 0.01h + 0.07c$

$$= 0.04(100 - h - c) + 0.01h + 0.07c$$
$$= 4 - 0.03h + 0.03c$$

Elasticity and hardness: (similar substitutions)

$$32 + 0.05c - 0.002c^2 + 0.01h - 0.004ch - 0.002h^2 \geq 16$$
$$25 \leq 44 + 0.96h - 0.4c - 0.01ch - 0.005h^2 + 0.001c^{1.95} \leq 35$$

Original bounds: $25 \leq r \leq 60$ and $c \geq 50$.

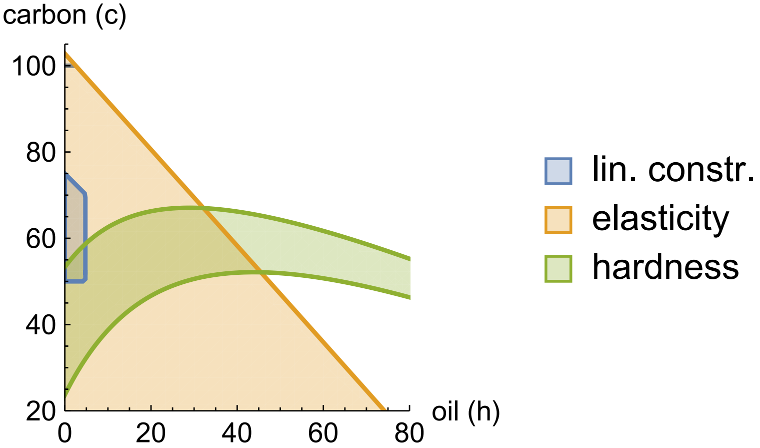
$$\iff 25 \leq 100 - h - c \leq 60 \text{ and } c \geq 50$$
$$\iff 40 \leq h + c \leq 75 \text{ and } c \geq 50$$
$$\iff 50 \leq h + c \leq 75 \text{ and } c \geq 50$$

Example: making tires

$$
\begin{aligned}
\underset{h,c}{\text{minimize}} \quad & 4 - 0.03h + 0.03c \\
\text{tensile:} \quad & 0 \le h \le 4.77 \\
\text{bound:} \quad & 50 \le h + c \le 75, \quad c \ge 50 \\
\text{elasticity:} \quad & 32 + 0.05c - 0.002c^2 + 0.01h \\
& \qquad - 0.004ch - 0.002h^2 \ge 16 \\
\text{hardness:} \quad & 25 \le 44 + 0.96h - 0.4c - 0.01ch \\
& \qquad - 0.005h^2 + 0.001c^{1.95} \le 35
\end{aligned}
$$

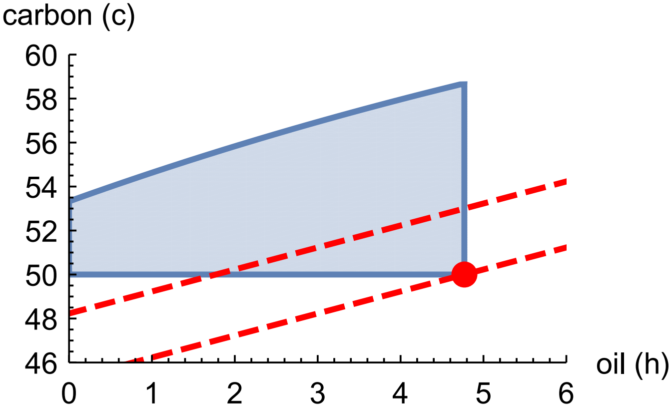
- tensile constraint is now linear
- elasticity constraint is a convex quadratic
- Only two variables! Let's draw a picture...

Example: making tires



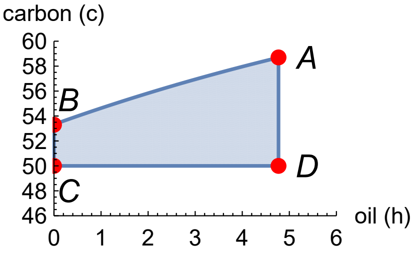
- Feasible region is quite small. Let's zoom in...

Example: making tires



- Objective is to minimize $4 - 0.03h + 0.03c$
- Solution doesn't involve hardness or elasticity constraints.

Example: making tires



- Objective function is:
  $(p_h - p_r)h + (p_c - p_r)c$
  where p_i is the price of i.
- Normal vector for objective:
  $n = \begin{bmatrix} p_h - p_r \\ p_c - p_r \end{bmatrix}$

Simple solution:

- Is rubber the cheapest ingredient? if so, choose **C**.
- Otherwise: is rubber the most expensive? if so, choose **A**.
- Otherwise: is oil cheaper than carbon? if so, choose **D**.
- Is rubber cheaper than the avg price of carbon and oil?
  if so, choose **B**. Otherwise, choose **A**.

Making tires, what did we learn?

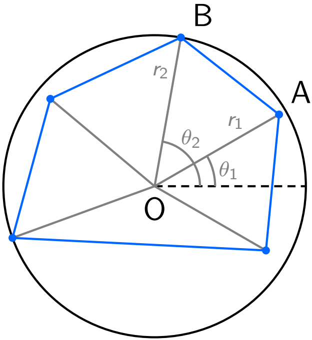
- Sometimes constraints that look complicated aren't actually complicated.
- Sometimes a constraint won't matter. You can examine dual variables to quickly check which constraints are active.
- If you can draw a picture, draw a picture!
- Complicated-looking problems can have simple solutions.

Example: largest inscribed polygon

What is the polygon (n sides) of maximal area that can be completely contained inside a circle of radius 1?

- A pretty famous problem. The solution is a regular polygon. All sides have equal length with vertices on the unit circle.
- How can we solve this using optimization?

Example: largest inscribed polygon



First model

Express the vertices of the polygon in polar coordinates (r_i, θ_i) where the origin is the center of the circle and angles are measured with respect to $(1, 0)$.

- What are the constraints?
- How do we compute the area?

- We must have $r_i \leq 1$ to ensure all points are inscribed.
- Calculate the area one triangle at a time. For example, triangle (OAB) has area $\frac{1}{2} r_1 r_2 \sin(\theta_2 - \theta_1)$.
- Is this enough? Let's see... Polygon.ipynb

Example: largest inscribed polygon

Model

$$\begin{aligned} \max_{r,\theta} \quad & \frac{1}{2}\sum_{i=1}^{n} r_i r_{i+1} \sin(\theta_{i+1} - \theta_i) \\ \text{s.t.} \quad & 0 \le r_i \le 1 \end{aligned}$$

Result

Solution is incorrect!

- Adding $\theta_i \ge 0$ doesn't help.
- Adding $\theta_i \le 2\pi$ doesn't help.
- Adding $\theta_1 = 0$ doesn't help.

- can obtain a single-point solution
- can obtain polygons that cross each other
- can obtain other suboptimal polygons

The reason is **local maxima**. More on this later...

Example: largest inscribed polygon

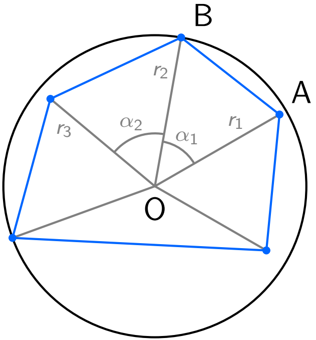
Model 1 finalized:

By assigning an order to the angles, we obtain the model:

$$
\begin{aligned}
\underset{r,\theta}{\text{maximize}} \quad & \frac{1}{2}\sum_{i=1}^{n} r_i r_{i+1} \sin(\theta_{i+1} - \theta_i) \\
\text{subject to:} \quad & 0 \le r_i \le 1 \\
& 0 = \theta_1 \le \theta_2 \le \cdots \le \theta_n \le 2\pi
\end{aligned}
$$

This model produces the correct solution!

Example: largest inscribed polygon



Second model

This time use *relative angles*. α_i is the angle between a pair of adjacent edges. This automatically encodes ordering!

- What are the constraints?
- How do we compute the area?

- We must have $r_i \leq 1$ to ensure all points are inscribed.
- Angles must sum to the full circle: $\alpha_1 + \cdots + \alpha_n = 2\pi$.
- Calculate the area one triangle at a time. For example, triangle (OAB) has area $\frac{1}{2} r_1 r_2 \sin(\alpha_i)$.

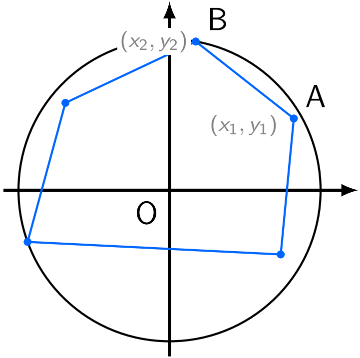
Example: largest inscribed polygon

Model 2 finalized:

$$
\begin{aligned}
\underset{r,\alpha}{\text{maximize}} \quad & \frac{1}{2}\sum_{i=1}^{n} r_i r_{i+1} \sin(\alpha_i) \\
\text{subject to:} \quad & 0 \le r_i \le 1 \\
& \alpha_1 + \cdots + \alpha_n = 2\pi \\
& \alpha_i \ge 0
\end{aligned}
$$

This model produces the correct solution as well!

Example: largest inscribed polygon



Third model

This time use cartesian coordinates. Each point is described by (x_i, y_i).

- What are the constraints?
- How do we compute the area?

- We must have $x_i^2 + y_i^2 \leq 1$ to ensure all points are inscribed.
- Calculate the area one triangle at a time. For example, triangle (OAB) has area $\frac{1}{2}\left|x_1 y_2 - y_1 x_2\right|$.

Example: largest inscribed polygon

Model

$$\begin{aligned} \max_{x,y} \quad & \frac{1}{2}\sum_{i=1}^{n}(x_i y_{i+1} - y_i x_{i+1}) \\ \text{s.t.} \quad & x_i^2 + y_i^2 \le 1 \end{aligned}$$

Result

Solution is zero...

- Changing initial values sometimes produces the correct answer.
- Fails frequently for larger n.

Reasons for failure

- again we have multiple local minima.
- area formula only works if vertices are consecutive!
- can fix this by ensuring $x_i y_{i+1} - y_i x_{i+1} > 0$ always holds

Example: largest inscribed polygon

Model 3 finalized:

$$\begin{aligned} \underset{x,y}{\text{maximize}} \quad & \frac{1}{2}\sum_{i=1}^{n}(x_i y_{i+1} - y_i x_{i+1}) \\ \text{subject to:} \quad & x_i^2 + y_i^2 \leq 1 \\ & x_i y_{i+1} - y_i x_{i+1} \geq 0 \quad \forall i \text{ (cyclic)} \end{aligned}$$

This model produces the correct solution provided we don't initialize the solver at zero.

Polygons, what did we learn?

- The choice of variables matters!
- Constraints can be added to remove unwanted symmetries or to avoid pathological cases (in the mathematical sense). e.g. our area formula fails if the vertices aren't consecutive.
- Local maxima/minima (extrema) are a problem!
- Can avoid local extrema by carefully choosing initial values. Choosing random values can work too.

Local minima

Mathematical definition: A point $\tilde{x}$ is a local minimum of f if there exists some $R > 0$ such that $f(\tilde{x}) \le f(x)$ whenever x satisfies $\|x - \tilde{x}\| \le R$.

Practical definition: A point $\tilde{x}$ is a local minimum of f if your solver thinks the answer is $\tilde{x}$ but it really isn't.

These definitions are **not** equivalent! Solvers will often claim victory when the point found isn't a minimum at all!

Example: $\left\{ \begin{array}{cc} \text{minimize} & -x^4 \\ \text{subject to:} & |x| \le 1 \end{array} \right\}$

Local minima

The solver will usually identify a local minimum if:

- changing any of the variables independently doesn't improve the objective. For example:

$$\begin{aligned} \max_{r,\theta} \quad & \frac{1}{2}\sum_{i=1}^{n} r_i r_{i+1} \sin(\theta_{i+1} - \theta_i) \\ \text{s.t.} \quad & 0 \le r_i \le 1 \end{aligned}$$

  - If we start with all variables zero, the objective remains zero if we change a single r_i or θ_i.
  - If all r_i are the same and all θ_i are the same, changing any of the r_i has no effect. Also, changing a single θ_i creates a cancellation so still no effect.

Local minima

The solver will usually identify a local minimum if:

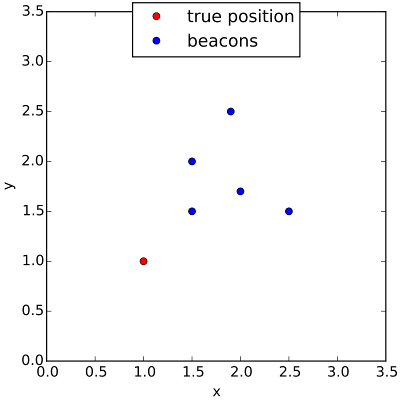
- all partial derivatives are zero at the particular point.
  For example: if $f(x, y)$ is the objective and $(\tilde{x}, \tilde{y})$ satisfies:

$$\frac{\partial f}{\partial x}(\tilde{x}, \tilde{y}) = \frac{\partial f}{\partial y}(\tilde{x}, \tilde{y}) = 0$$

  This was the case with the $-x^4$ example. It also happens with $-x^2$ and x^3, which is actually an inflection point.

Why does this happen? It has to do with how solvers work. We'll learn more about this in the next lecture.

Example: navigation using ranges



- There is a set of n beacons with known positions (x_i, y_i).
- We can measure our distance to each of the beacons. The measurements will be noisy.
- We would like to find our true position $(u_\star, v_\star)$ based on the beacon distances.

Example by L. Vandenberghe, UCLA, EE133A

Example: navigation using ranges

- The distance we measure to beacon i will be given by:

$$\rho_i = \sqrt{(x_i - u_\star)^2 + (y_i - v_\star)^2} + w_i$$

These are the measurements (w_i is noise).

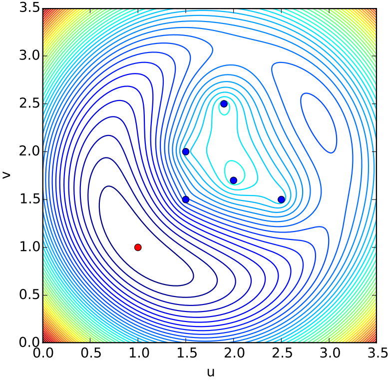
- Suppose we *think* we are at (u, v). We can compare the actual measurements to the hypothetical expected measurements by using a squared difference:

$$r(u, v) = \sum_{i=1}^{n} \left(\sqrt{(x_i - u)^2 + (y_i - v)^2} - \rho_i \right)^2$$

- Minimizing r is called nonlinear least squares. If the measurements are linear $y_i = a_i^\mathsf{T} x + w_i$ then r would simply be $\|Ax - y\|^2$, which is the conventional least-squares cost.

Example: navigation using ranges

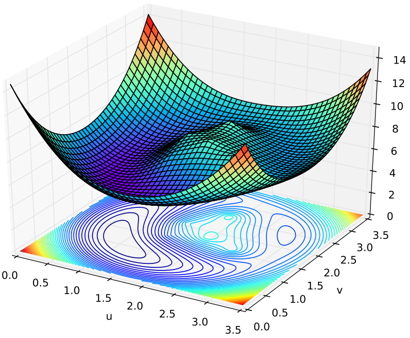
$$\underset{u,v}{\text{minimize}} \quad r(u,v) = \sum_{i=1}^{n} \left(\sqrt{(x_i - u)^2 + (y_i - v)^2} - \rho_i \right)^2$$



- In the noise-free measurement case, we have two local minima: $(1,1)$ and $(2.91, 2.32)$.
- There are three local maxima.
- In the noisy measurement case, we will never get an error of zero, so it's difficult to know when we've found the true position!

Example by L. Vandenberghe, UCLA, EE133A

Example: navigation using ranges



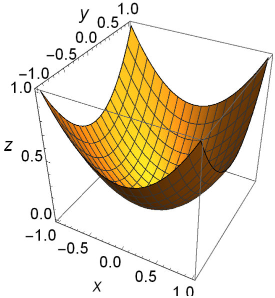
- Julia code: Navigation.ipynb
- Changing start values for the solver affects which minimum value is found.
- In the noisy measurement case, we will never get an error of zero, so it's difficult to know when we've found the true position!

- Solver struggles with finding the local **maxima** for this function. This is because the derivative of $r(u, v)$ is not defined at the beacon locations (where some of the maxima lie).
- Example: compare minimizing $\sqrt{x^2 + y^2}$ versus $\frac{1}{2}(x^2 + y^2)$.

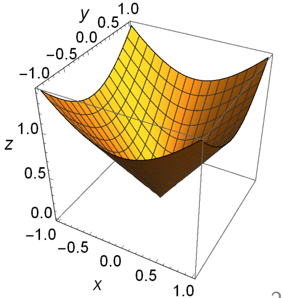
Example by L. Vandenberghe, UCLA, EE133A

Difficult derivatives

- Consider $f(x, y) = \frac{1}{2}(x^2 + y^2)$.
- A paraboloid with a smooth minimum.
- Easy to optimize because $\|\nabla f\|$ tells you how close you are. $\|\nabla f\| = \sqrt{x^2 + y^2}$. Small gradient $\iff$ close to optimality.



- Consider $f(x, y) = \sqrt{x^2 + y^2}$.
- A cone with a sharp minimum.
- Difficult to optimize because $\|\nabla f\|$ is not informative. $\|\nabla f\| = 1$. Hard to gauge distance to optimality.



Navigation & NLLS, what did we learn?

- Standard least squares is a convex problem. So there is a single local minimum which is also a global minimum (in the overdetermined case).
- In nonlinear least squares (NLLS), there may be multiple local and global minima.
- The solver may still struggle in certain cases, and this is related to gradients (more on this later).
- Again: draw a picture, it helps!