Convex Optimization

Madeleine Udell

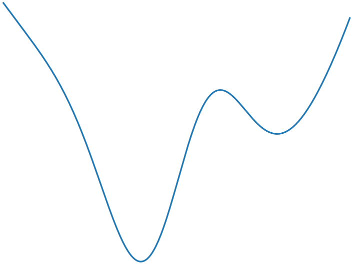
Operations Research and Information Engineering
Cornell University

Schonfeld Quantitative, September 26 2017

What is an optimization problem?

the optimization contract:

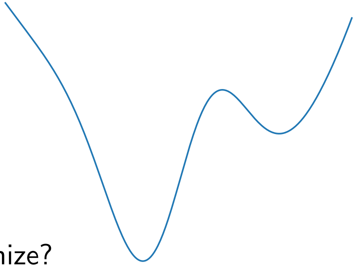
- you give me a function, and I'll find you its minimum



What is an optimization problem?

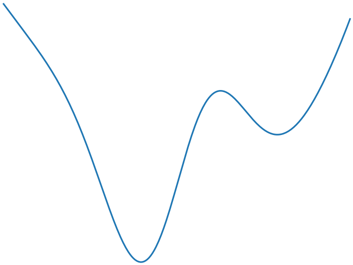
the optimization contract:

- you give me a function, and I'll find you its minimum



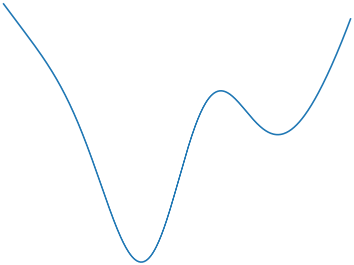
- why optimize?
  - fit a model to data (*e.g.*, understand customer preferences)
  - make predictions (*e.g.*, image recognition)
  - maximize revenue (*e.g.*, airline pricing)
  - maximize investment returns (*e.g.*, quant finance)
  - design a control system (*e.g.*, autopilot)
  - ...

How to find the minimum



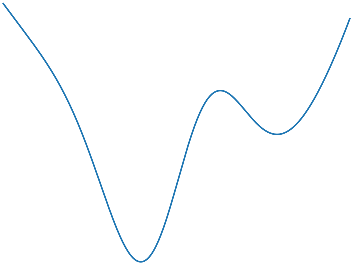
how to find the minimum?

How to find the minimum



how to find the minimum?

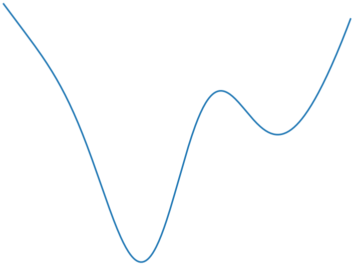
How to find the minimum



how to find the minimum?

- set derivative to zero

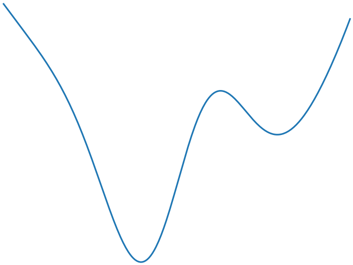
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula

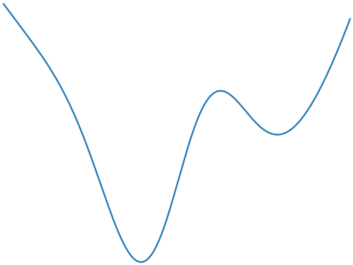
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop

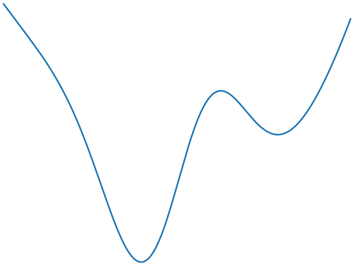
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable

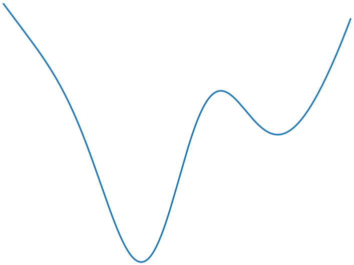
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable
- other fancy optimization methods

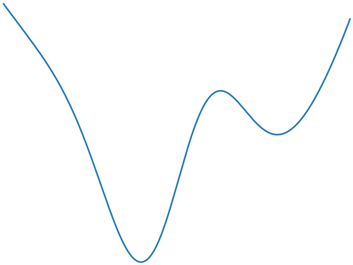
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable
- other fancy optimization methods . . . where they apply

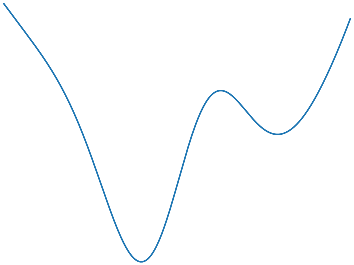
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable
- other fancy optimization methods . . . where they apply
- point

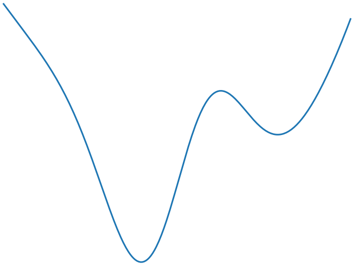
How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable
- other fancy optimization methods . . . where they apply
- point . . . if I can plot problem in 2D or 3D

How to find the minimum



how to find the minimum?

- set derivative to zero . . . derivative has an analytical formula
- gradient descent / backprop . . . if f is differentiable
- other fancy optimization methods . . . where they apply
- point . . . if I can plot problem in 2D or 3D

key question: what do I know about the problem?

What is an optimization problem?

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m_1 \\ & h_i(x) = 0, \quad i = 1, \dots, m_2 \end{array}$$

- problem variable $x \in \mathbf{R}^n$
- objective f_0
- inequality constraints f_i
- equality constraints h_i

Convex optimization

general optimization problems are hard

- local minima, saddle points
- can take exponential time to solve

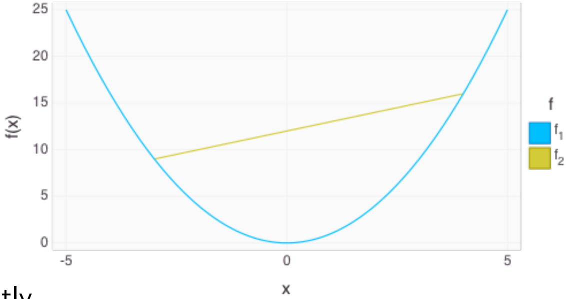
convex optimization problems are easy

- all minima are global
- fast, global convergence
- can prove we found a solution
- models a wide variety of real problems

Convexity

f is **convex** if for all $\theta \in [0, 1]$

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y)$$



equivalently,

- f has nonnegative (upward) curvature
- the graph of f never lies above its chords
- $f'' \geq 0$ (if f is differentiable)

Convex optimization

convex optimization problem:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m_1 \\ & h_i(x) = 0, \quad i = 1, \ldots, m_2 \end{array}$$

where

- objective f_0 is convex
- inequality constraints f_i are convex
- equality constraints h_i are affine

Convex in action

demo:

https://github.com/madeleineudell/schoenfeld/Convex-tutorial.ipynb

Portfolio optimization

demo:

`https://github.com/madeleineudell/schoenfeld/Convex-tutorial.ipynb`

Disciplined convex programming

Disciplined convex programming (DCP) [Grant, Boyd & Ye, 2006] provides a set of simple inductive rules to verify convexity:

- $f \circ g(x)$ is convex in x if
  - f is convex nondecreasing and g is convex
  - f is convex nonincreasing and g is concave

  cf., the chain rule:

$$(f \circ g)''(x) = f''(g(x))(g(x))^2 + f'(g(x))g''(x)$$

Disciplined convex programming

Disciplined convex programming (DCP) [Grant, Boyd & Ye, 2006] provides a set of simple inductive rules to verify convexity:

- $f \circ g(x)$ is convex in x if
  - f is convex nondecreasing and g is convex
  - f is convex nonincreasing and g is concave

  cf., the chain rule:

  $$(f \circ g)''(x) = f''(g(x))(g(x))^2 + f'(g(x))g''(x)$$

- $f \circ g(x)$ is concave in x if $-f \circ g(x)$ is convex
- $f \circ g(x)$ is affine if it is both convex and concave

A function is **DCP** if its convexity (or concavity) can be inferred from these composition rules.

Disciplined convex programming

Disciplined convex programming (DCP) [Grant, Boyd & Ye, 2006] provides a set of simple inductive rules to verify convexity:

- $f \circ g(x)$ is convex in x if
  - f is convex nondecreasing and g is convex
  - f is convex nonincreasing and g is concave

  cf., the chain rule:

$$(f \circ g)''(x) = f''(g(x))(g(x))^2 + f'(g(x))g''(x)$$

- $f \circ g(x)$ is concave in x if $-f \circ g(x)$ is convex
- $f \circ g(x)$ is affine if it is both convex and concave

A function is **DCP** if its convexity (or concavity) can be inferred from these composition rules.

(**N.B.** a function that is not DCP may still be convex)

DCP: demo

demo:

`https://github.com/madeleineudell/schoenfeld/Convex-tutorial.ipynb`

Structure determines solvers

How should we solve this problem?

- LP solver?
- conic solver?
- nonlinear derivative based solver?
- operator splitting?

Structure determines solvers

How should we solve this problem?

- LP solver?
- conic solver?
- nonlinear derivative based solver?
- operator splitting?

What do we know about this problem's structure?

Structure

useful kinds of structure:

- is the problem convex?
- is the problem representable in some standard form?
  - convex: LP, QP, SOCP, SDP, ...
  - nonconvex: MILP, MISOCP ...
- is the problem smooth?
- ...

Optimization in Julia

model specifies structure; solvers exploit structure

- model (*e.g.*, JuMP or Convex)
- glue (MathProgBase)
- solvers (*e.g.*, GLKP, Gurobi, Mosek, ECOS, ...)

JuliaOpt curates all these solvers: `http://www.juliaopt.org/`

Convex optimization modeling tools

many choices for convex optimization modeling!

- Convex.jl (Julia; this talk)
- CVX (Matlab)
- CVXPY (Python)
- CVXR (R)

Summary

convex optimization provides

- flexible modeling
- fast, efficient, reliable algorithms
- guaranteed solutions