Introduction to (Mathematical) Optimization

AJ Friend, Nick Henderson
(w/ material from Stephen Boyd and Steven Diamond)

June 28, 2015

Optimization

Optimization

Optimization introducing a change to a system to achieve a better (or best) outcome

Optimization

Optimization introducing a change to a system to achieve a better (or best) outcome

Optimized there does not exist a (known) change to a system to achieve a better outcome

Mathematical Optimization

Optimization introducing a change to a system to achieve a better (or best) outcome

Optimized there does not exist a (known) change to a system to achieve a better outcome

Mathematical Optimization

Optimization introducing a change to a system to achieve a better (or best) outcome

Optimized there does not exist a (known) change to a system to achieve a better outcome

- system: mathematical model
- change: change to input variables (parameters)
- outcome: a measure of performance of the model, objective function

The Raptor Problem

See other slides

Optimization in one variable

$$\text{minimize} \quad f(x) \in C^2 : \mathbf{R} \to \mathbf{R}$$

- x is real number variable
- $f(x)$ is the objective function, we typically want this to be twice continuously differentiable. This means both the first and second derivative are continuous in x
- We are looking for a point x^* such that $f'(x) = 0$ and $f''(x) > 0$. Note that this is a *local* optimizer

Optimization in one variable

Key words

- *domain*: space for input variable x
- *range*: space for output $f(x)$
- *critical point*: $f'(x) = 0$
- *local minimizer*: $f'(x) = 0$ and $f''(x) > 0$
- *local maximizer*: $f'(x) = 0$ and $f''(x) < 0$
- *saddle point*: $f'(x) = 0$ and $f''(x) = 0$
- *global minimizer*: x^* such that $f(x^*) \leq f(x)$ for all x in domain

Optimization in two variables

$$\text{minimize} \quad f(x) \in C^2 : \mathbf{R}^2 \to \mathbf{R}$$

- x is a 2-dimensional vector of real variables
- $f(x)$ is the objective function
  - First derivative or gradient of f is written $\nabla f(x)$
  - Second derivate or Hessian of f is written $\nabla^2 f(x)$
- We are looking for a point x^* such that $\nabla f(x) = 0$ and $\nabla^2 f(x) \succeq 0$. Note that this is a *local* optimizer

The gradient $\nabla f(x)$

Vector of variables:

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Gradient of f:

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix}$$

The Hessian $\nabla^2 f(x)$

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}$$

Surface plots and contours

- show local minimizer/maximizer critical point
- show saddle point

Higher dimensions

- can't easily visualize
- need analysis

Algorithms

- Basic loop
- Descent condition
- Gradient descent
- Newton's method

Show example on rosenbrock function

- gradient descent vs. newton's method

Two very important optimization problems

- linear least squares
- non-linear least squares

Constraints

- basic idea of constraints
- work through example from multivariate calculus
- introduce idea of multipliers
- equality constraints
- inequality constraints
- linear constraints
- nonlinear constraints

Penalty and barrier methods

- introduce a penalty into the objective to penalize constraint violation

Linear programming

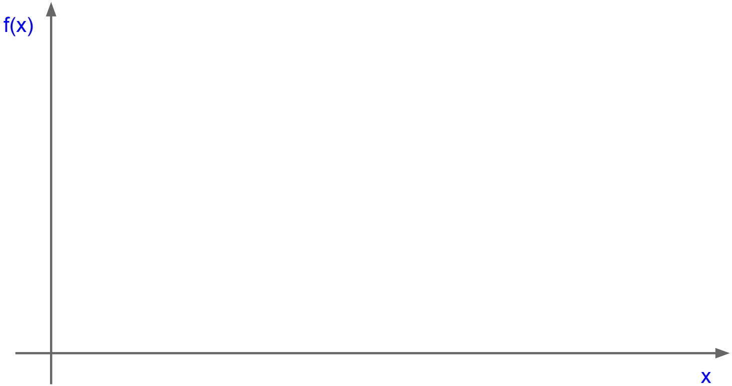
Discrete variables

- Mixed integer programming
- Scheduling problems

What's next

- Nonlinear programming
- Convex modelling
- Study of algorithms
- Modeling languages
- Automated differentiation

Test image 1



Test image 2