Geometry Meets Machine Learning

Instructor: Hao Su

Rough Plan

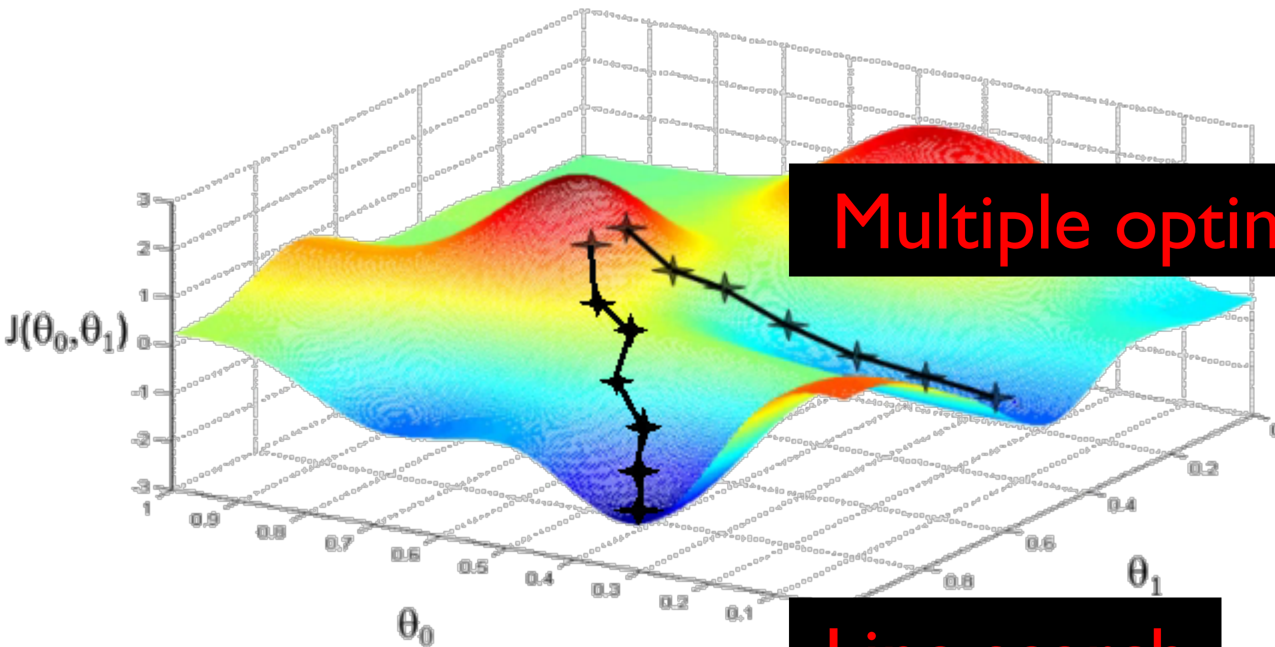
- Linear problems
- Unconstrained optimization
- Equality-constrained optimization

Unconstrained Optimization

$$\min_{x} f(x)$$

Unstructured.

Basic Algorithms



$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

Gradient descent

Basic Algorithms

$$\lambda_0 = 0, \lambda_s = \frac{1}{2}(1 + \sqrt{1 + 4\lambda_{s-1}^2}), \gamma_s = \frac{1 - \lambda_2}{\lambda_{s+1}}$$

$$y_{s+1} = x_s - \frac{1}{\beta} \nabla f(x_s)$$

$$x_{s+1} = (1 - \gamma_s) y_{s+1} + \gamma_s y_s$$

Inverse quadratic convergence on convex problems!

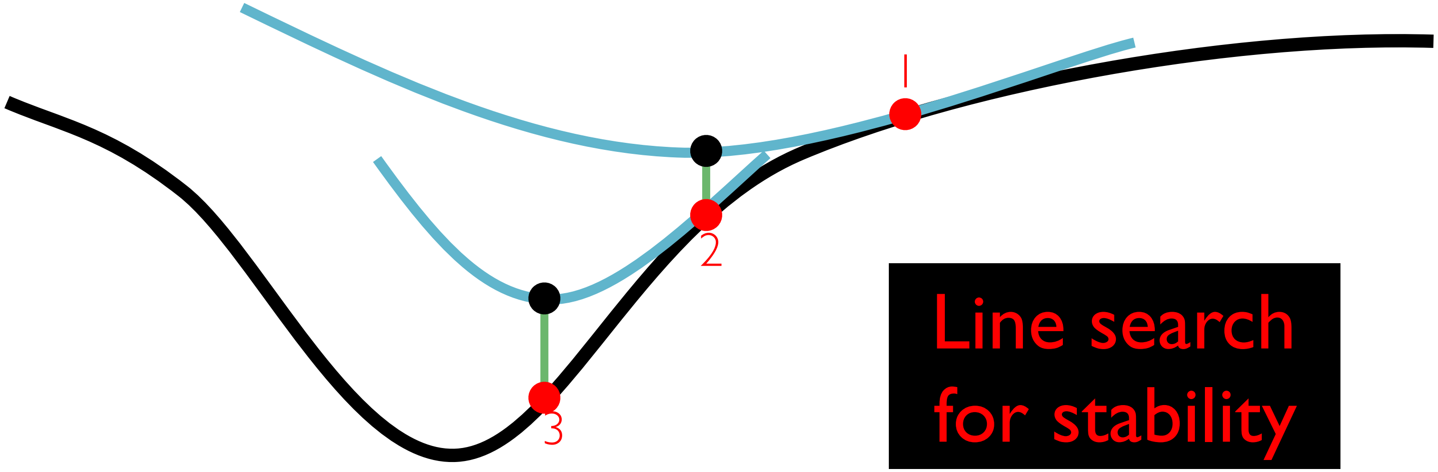
(Nesterov 1983)

$$f(X(t)) - f^\star \leq O\left(\frac{\|x_0 - x^\star\|^2}{t^2}\right)$$

Accelerated gradient descent

Basic Algorithms

$$x_{k+1} = x_k - [Hf(x_k)]^{-1} \nabla f(x_k)$$



Line search for stability

Newton's Method

Basic Algorithms

- (Often sparse) approximation from previous samples and gradients
- Inverse in closed form!

$$x_{k+1} = x_k - M_k^{-1} \nabla f(x_k)$$

Hessian approximation

Quasi-Newton: BFGS and friends

Example: Shape Interpolation

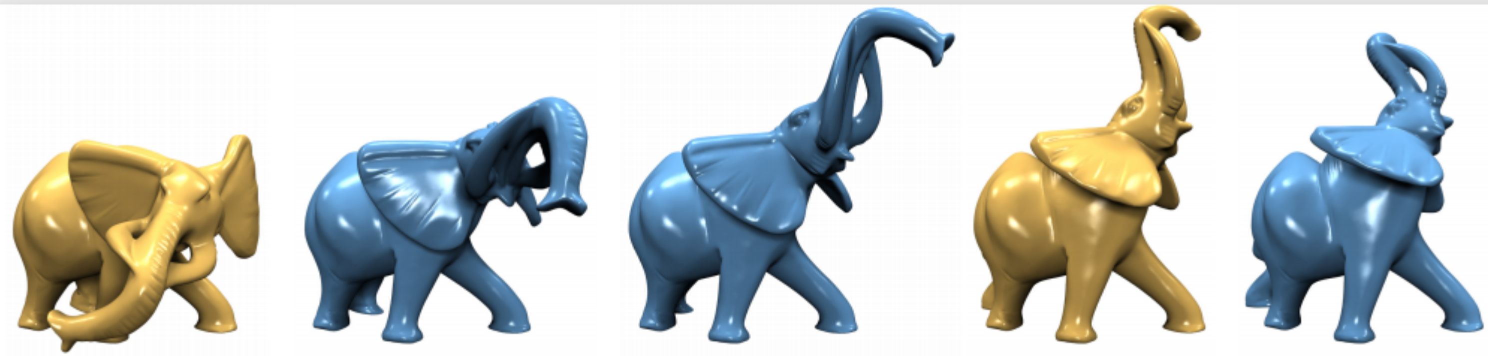


Figure 5: *Interpolation and extrapolation of the yellow example poses. The blending weights are* 0, 0.35, 0.65, 1.0, *and* 1.25.

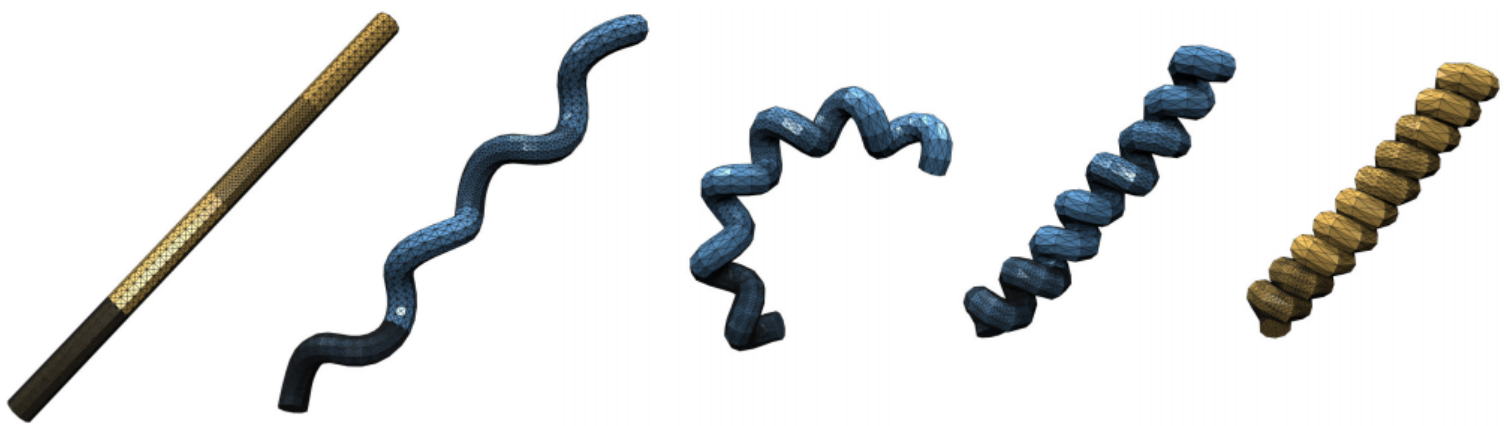


Figure 6: *Interpolation of an adaptively meshed and strongly twisted helix with blending weights* 0, 0.25, 0.5, 0.75, 1.0.

Fröhlich and Botsch. "Example-Driven Deformations Based on Discrete Shells." CGF 2011.

Interpolation Pipeline

Roughly:

1. Linearly interpolate edge lengths and dihedral angles.

$$\ell_e^* = (1-t)\ell_e^0 + t\ell_e^1$$

$$\theta_e^* = (1-t)\theta_e^0 + t\theta_e^1$$

2. Nonlinear optimization for vertex positions.

$$\min_{x_1,\ldots,x_m} \lambda \sum_e w_e(\ell_e(x) - \ell_e^*)^2 + \mu \sum_e w_b(\theta_e(x) - \theta_e^*)^2$$

Sum of squares: Gauss-Newton

Software

- Matlab: `fminunc` or `minfunc`
- C++: `libLBFGS`, `dlib`, others

Typically provide functions for function and gradient (and optionally, Hessian).

Try several!

Some Tricks

Lots of small elements: $\|x\|_2^2 = \sum_i x_i^2$

Lots of zeros: $\|x\|_1 = \sum_i |x_i|$

Uniform norm: $\|x\|_\infty = \max_i |x_i|$

Low rank: $\|X\|_* = \sum_i \sigma_i$

Mostly zero columns: $\|X\|_{2,1} = \sum_j \sqrt{\sum_i x_{ij}^2}$

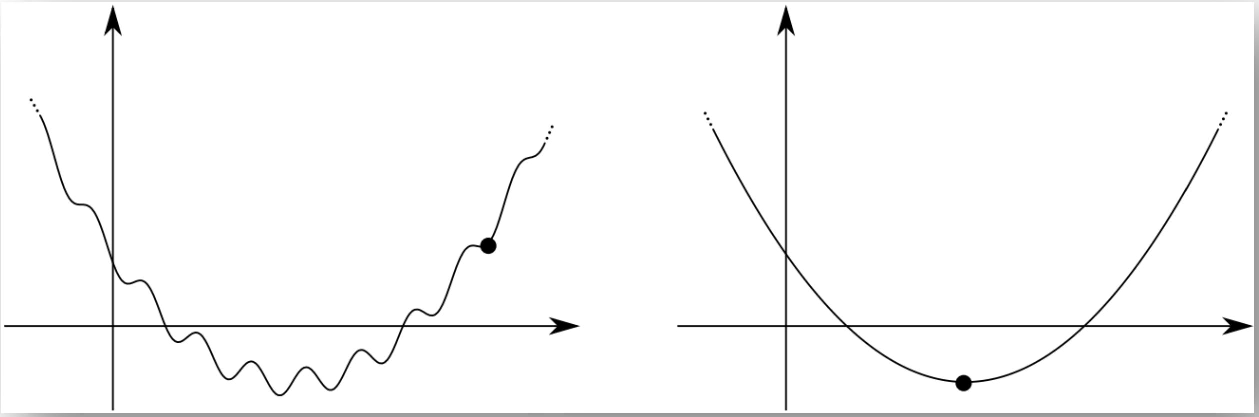
Smooth: $\int \|\nabla f\|_2^2$

Piecewise constant: $\int \|\nabla f\|_2$

???: Early stopping

Regularization

Some Tricks

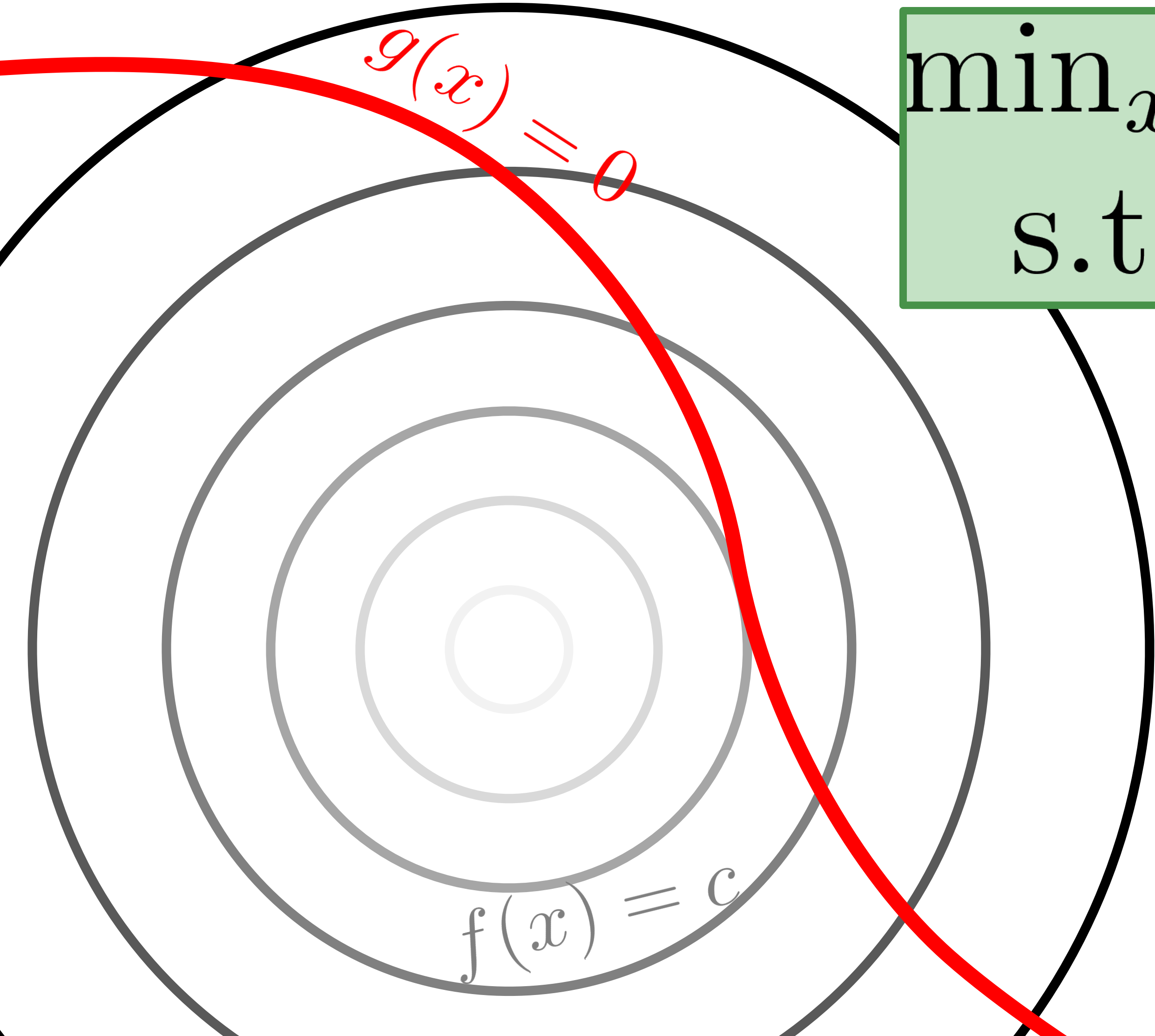


Multiscale/graduated optimization

Rough Plan

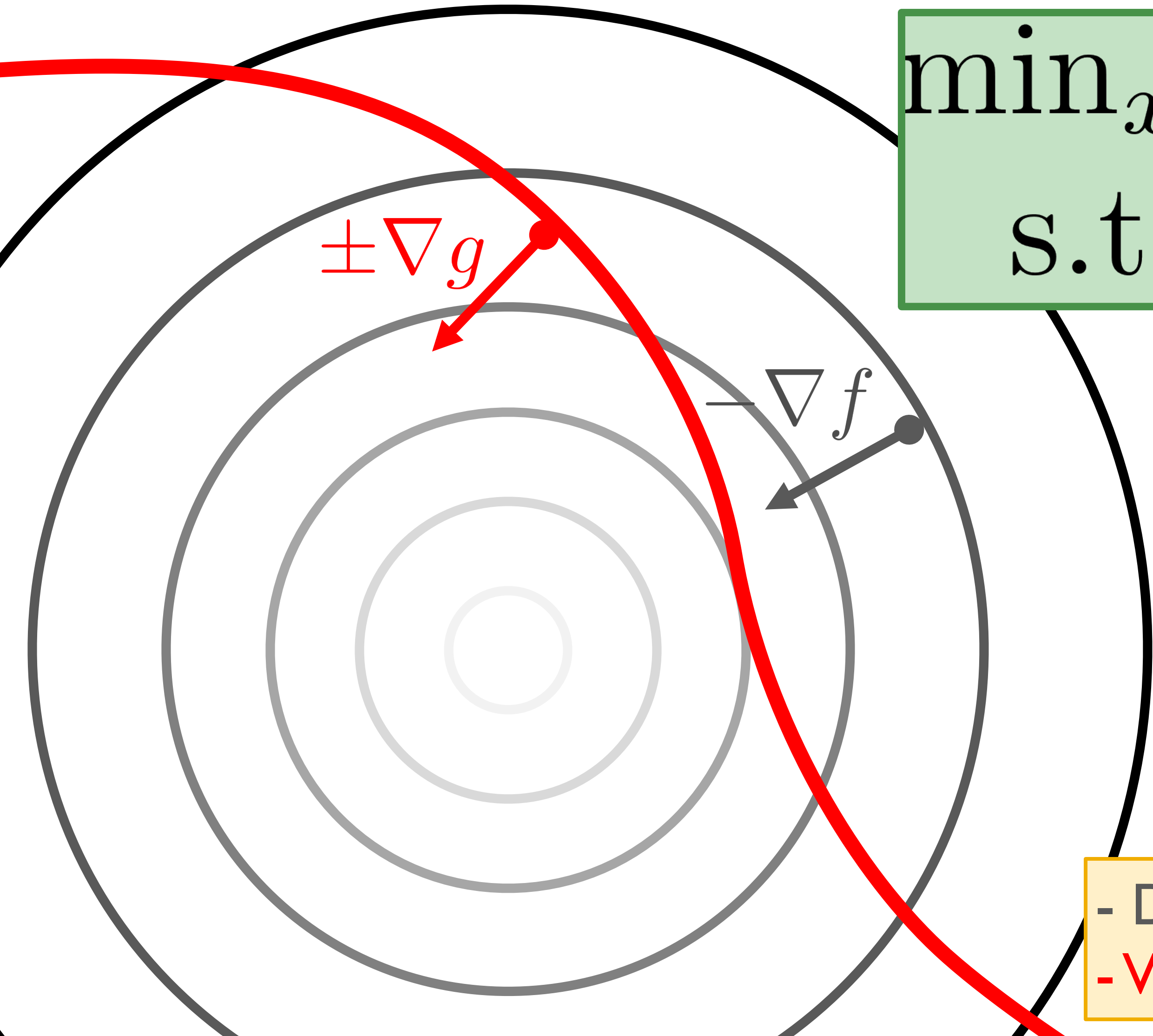
- Linear problems
- Unconstrained optimization
- Equality-constrained optimization

Lagrange Multipliers: Idea



$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

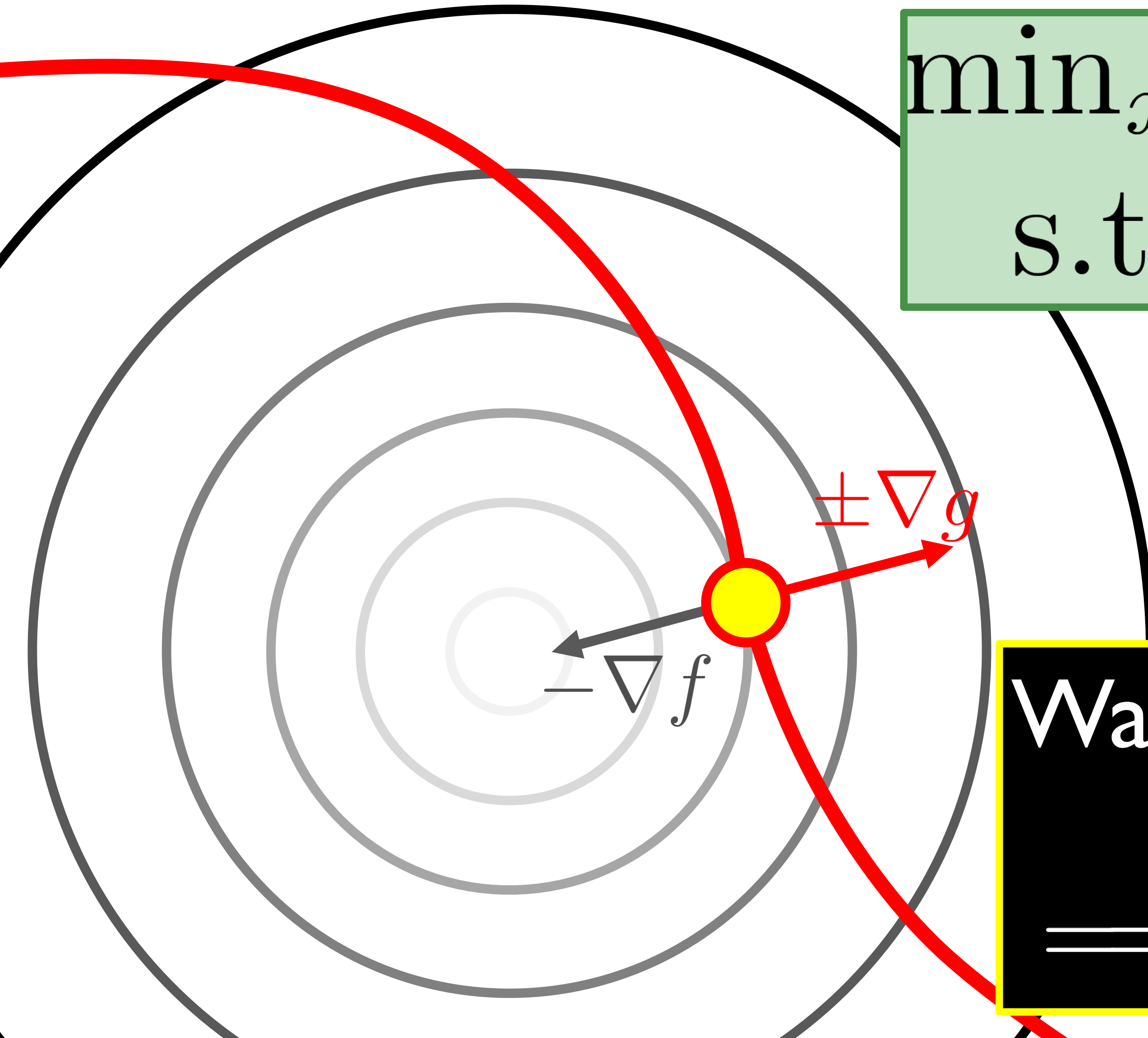
Lagrange Multipliers: Idea



$$\min_x \quad f(x)$$
$$\text{s.t.} \quad g(x) = 0$$

- Decrease *f*:
- Violate constraint:

Lagrange Multipliers: Idea



$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

Want:

$$\nabla f \| \nabla g$$

$$\implies \nabla f = \lambda \nabla g$$

Example: Symmetric Eigenvectors

$$f(x) = x^\top A x \implies \nabla f(x) = 2Ax$$

$$g(x) = \|x\|_2^2 \implies \nabla g(x) = 2x$$

$$\implies Ax = \lambda x$$

Use of Lagrange Multipliers

Turns constrained optimization into

unconstrained root-finding.

$$\nabla f(x) = \lambda \nabla g(x)$$

$$g(x) = 0$$

Many Options

- **Reparameterization**

  Eliminate constraints to reduce to unconstrained case

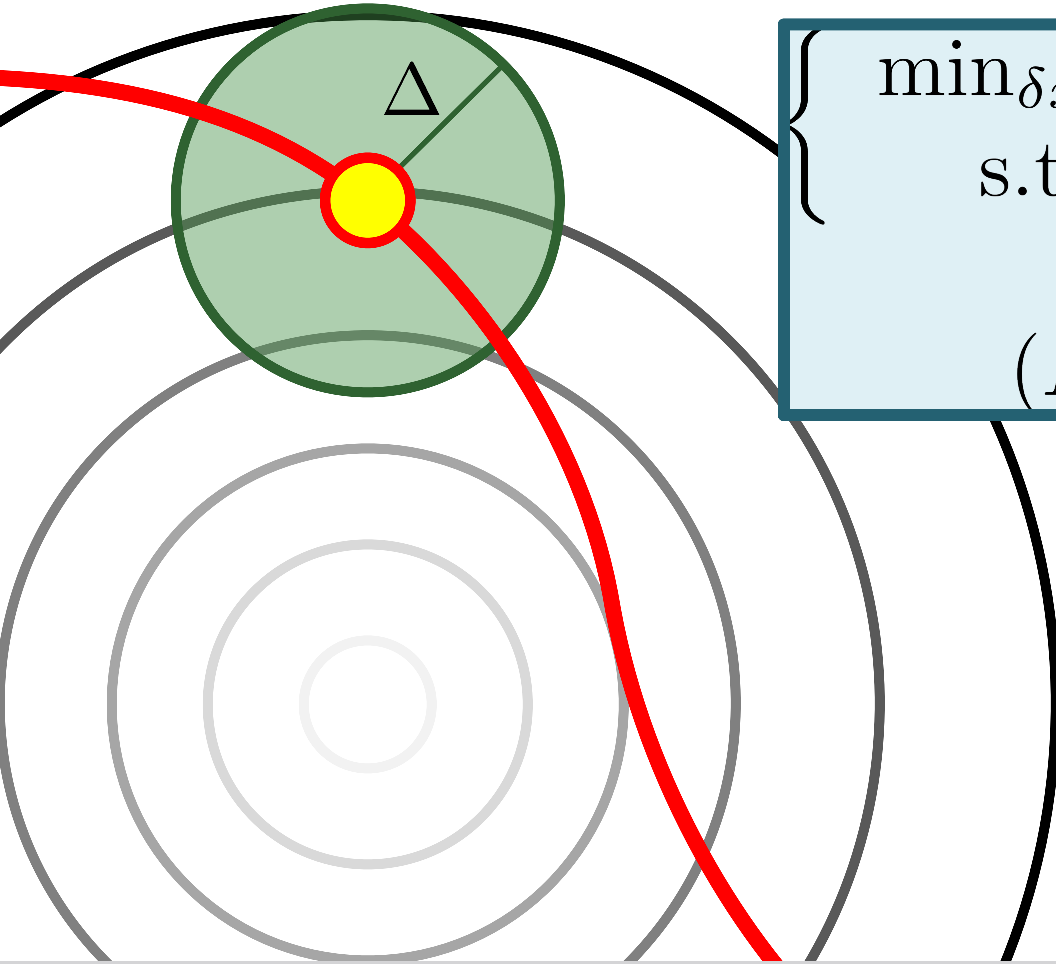
- **Newton's method**

  Approximation: quadratic function with linear constraint

- **Penalty method**

  Augment objective with barrier term, e.g. $f(x) + \rho\,|\,g(x)\,|$

Trust Region Methods



$$\left\{ \begin{array}{ll} \min_{\delta x} & \frac{1}{2}\delta x^\top H \delta x + w^\top x \\ \text{s.t.} & \|\delta x\|_2^2 \leq \Delta \end{array} \right\}$$

$$\downarrow$$

$$(H + \lambda I)\delta x = -w$$

Fix (or adjust) *damping parameter* .

Example: Levenberg-Marquardt

Aside:

Convex Optimization Tools

versus

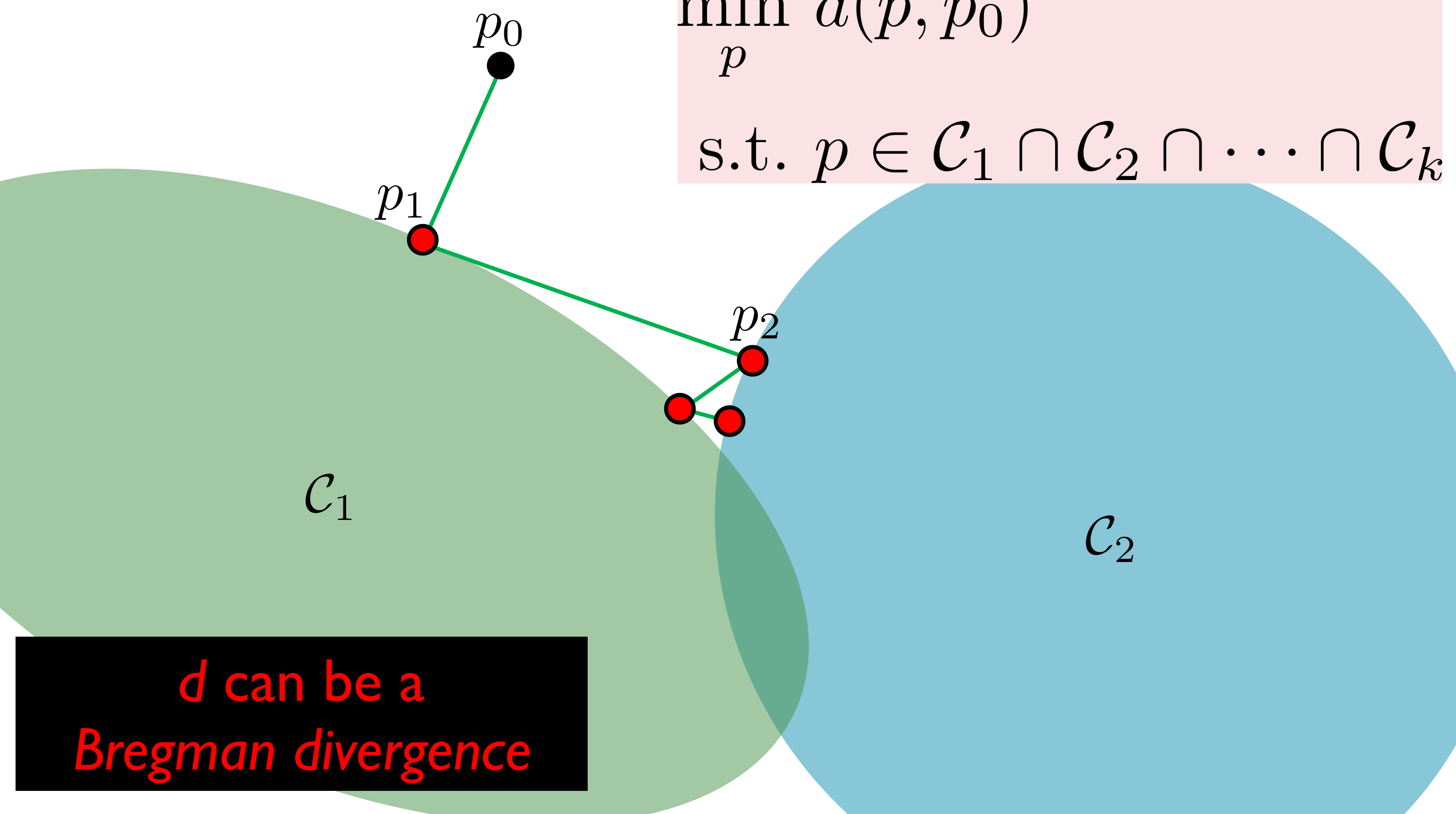
Sometimes work for non-convex problems...

Try lightweight options

Alternating Projection

$$\min_{p} \; d(p, p_0)$$

$$\text{s.t. } p \in \mathcal{C}_1 \cap \mathcal{C}_2 \cap \cdots \cap \mathcal{C}_k$$



d can be a *Bregman divergence*

Augmented Lagrangians

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

$$\downarrow$$

$$\begin{aligned} \min_x \quad & f(x) + \frac{\rho}{2}\|g(x)\|_2^2 \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

Does nothing when constraint is satisfied

Add constraint to objective

Alternating Direction Method of Multipliers (ADMM)

$$\begin{aligned} \min_{x,z} \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c \end{aligned}$$

$$\Lambda_\rho(x, z; \lambda) = f(x) + g(z) + \lambda^\top(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2$$

$$x \leftarrow \arg\min_x \Lambda_\rho(x, z, \lambda)$$

$$z \leftarrow \arg\min_z \Lambda_\rho(x, z, \lambda)$$

$$\lambda \leftarrow \lambda + \rho(Ax + Bz - c)$$

https://web.stanford.edu/~boyd/papers/pdf/admm_slides.pdf