# Gradient Descent

Yu-Xiang Wang
CS292A

(Based on Ryan Tibshirani's 10-725)

# Last time: Convex Optimization Basics

- Formulation and Reformulation of a problem
- Convex Relaxation
- Canonical convex programs

- Linear program (LP): takes the form

$$\begin{aligned} \min_{x} \quad & c^T x \\ \text{subject to} \quad & Dx \leq d \\ & Ax = b \end{aligned}$$

- Quadratic program (QP): like LP, but with quadratic criterion
- Semidefinite program (SDP): like LP, but with matrices
- Conic program: the most general form of all

# Today: Gradient Methods and Online Learning

- The Gradient Descent Algorithm.
- How to choose learning rates?
- Convergence analysis
  1. Gradient Descent for Smooth Convex Problems
  2. Gradient Descent for Smooth and Strongly Convex Problems
- Useful extensions
  1. Gradient Descent for Nonconvex Functions
  2. Restricted Strong Convexity
  3. Polyak-Łojasiewicz Condition and its friends.
- Example: Gradient boosting

# Gradient descent

Consider unconstrained, smooth convex optimization

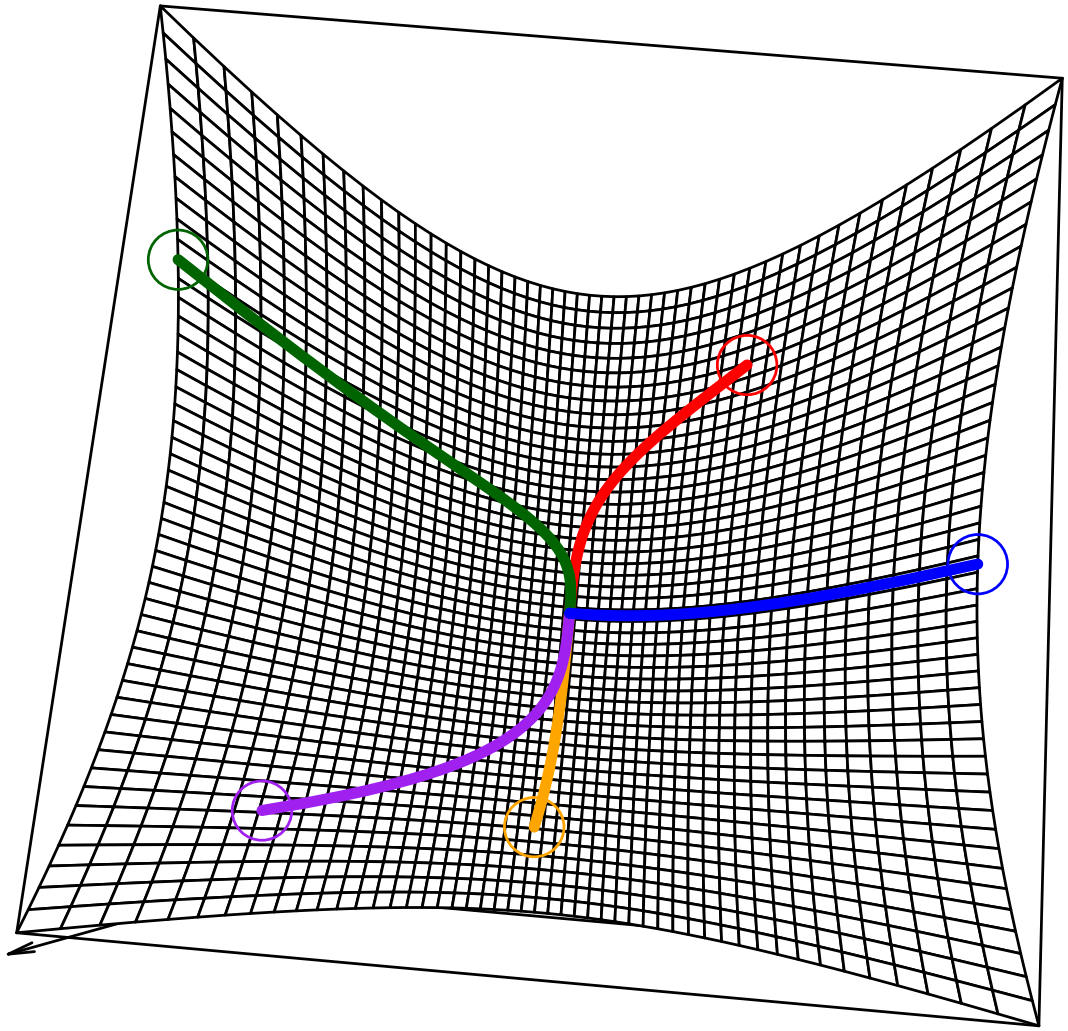
$$\min_x \ f(x)$$

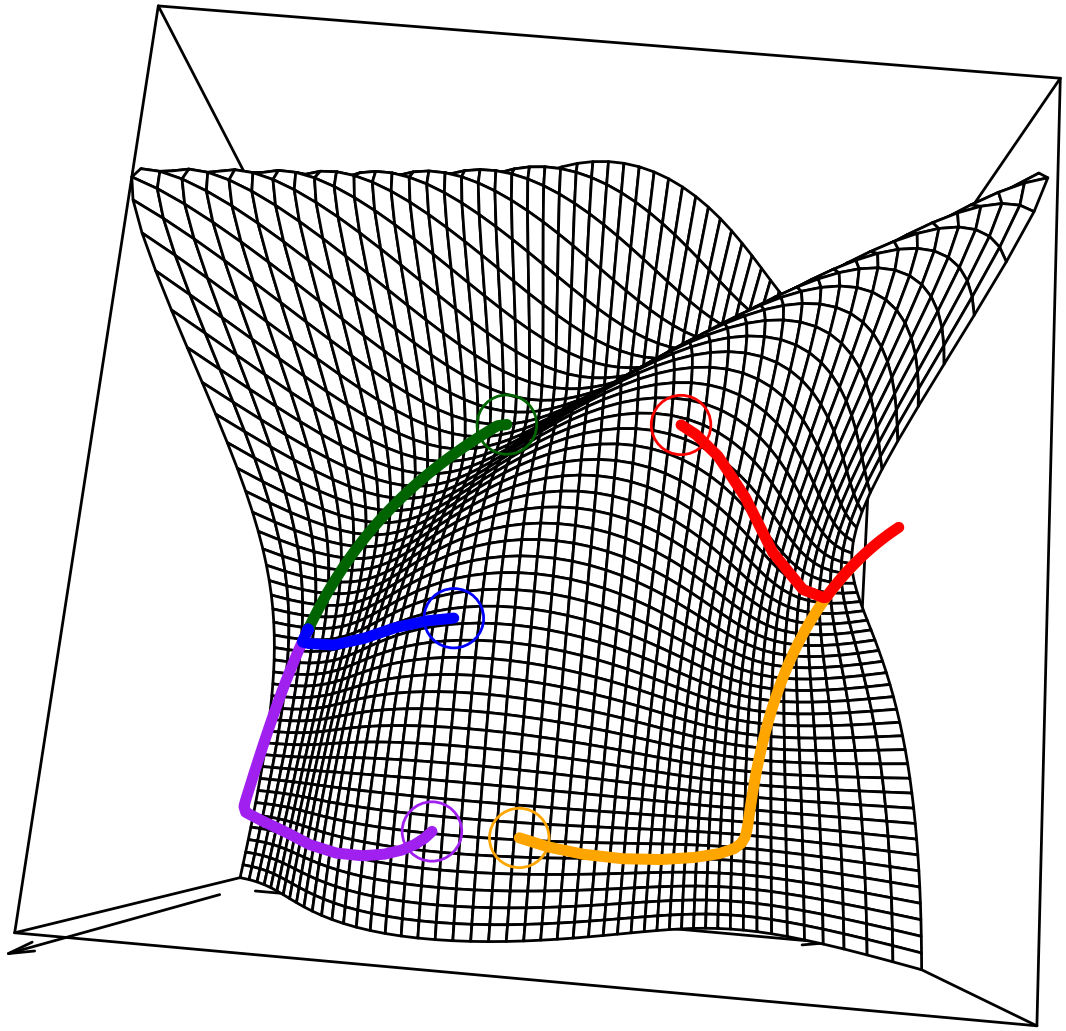
i.e., $f$ is convex and differentiable with $\mathrm{dom}(f) = \mathbb{R}^n$. Denote the optimal criterion value by $f^\star = \min_x \ f(x)$, and a solution by $x^\star$

Gradient descent: choose initial point $x^{(0)} \in \mathbb{R}^n$, repeat:

$$x^{(k)} = x^{(k-1)} - t_k \cdot \nabla f(x^{(k-1)}), \quad k = 1, 2, 3, \ldots$$

Stop at some point





# Gradient descent interpretation

At each iteration, consider the expansion

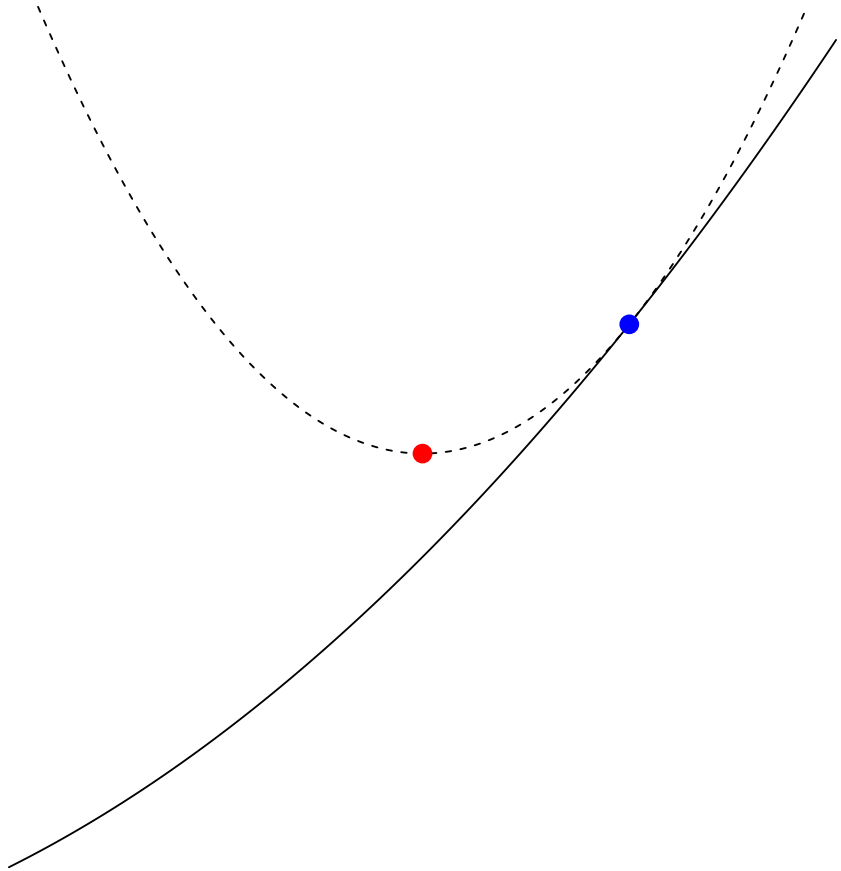
$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t}\|y - x\|_2^2$$

Quadratic approximation, replacing usual Hessian $\nabla^2 f(x)$ by $\frac{1}{t} I$

| | |
|---|---|
| $f(x) + \nabla f(x)^T (y - x)$ | linear approximation to $f$ |
| $\frac{1}{2t}\|y - x\|_2^2$ | proximity term to $x$, with weight $1/(2t)$ |

Choose next point $y = x^+$ to minimize quadratic approximation:

$$x^+ = x - t\nabla f(x)$$

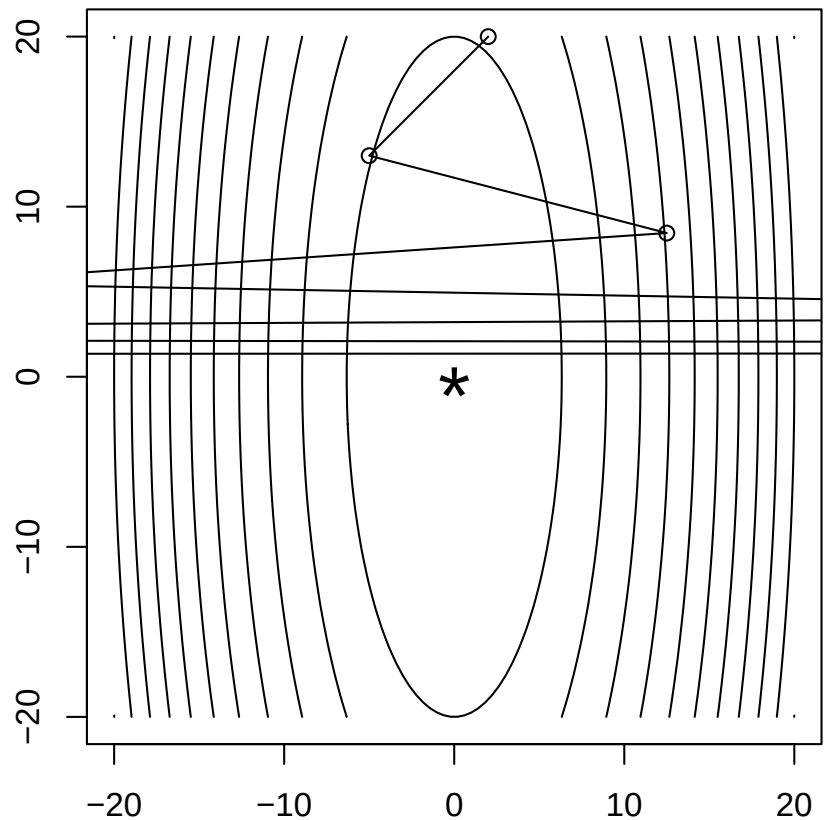


Blue point is $x$, red point is

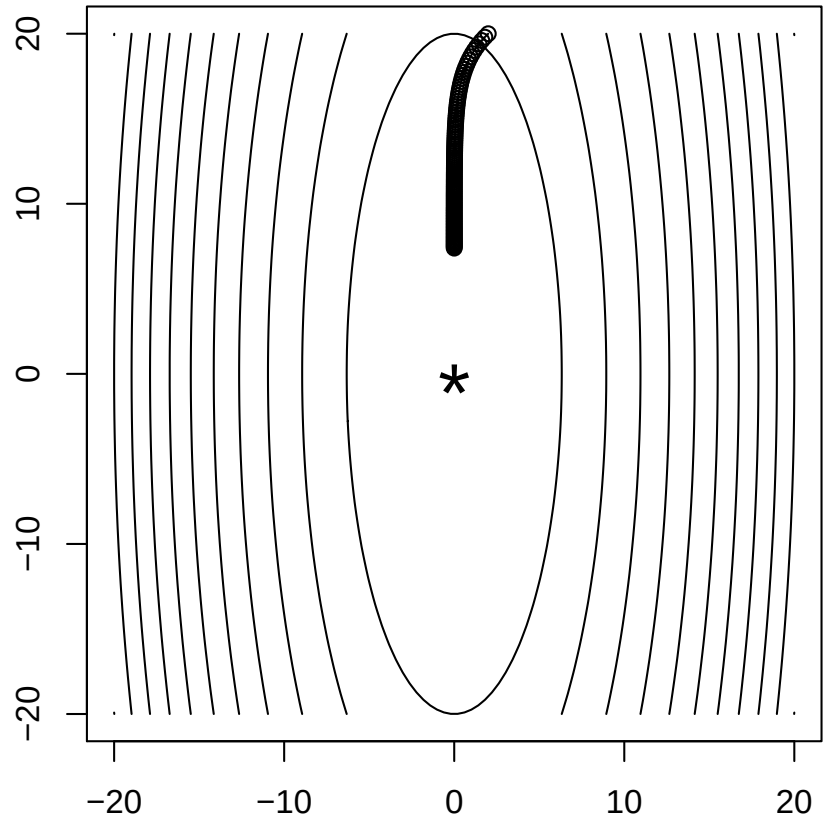
$$x^+ = \operatorname*{argmin}_y \; f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t} \|y - x\|_2^2$$

# Fixed step size

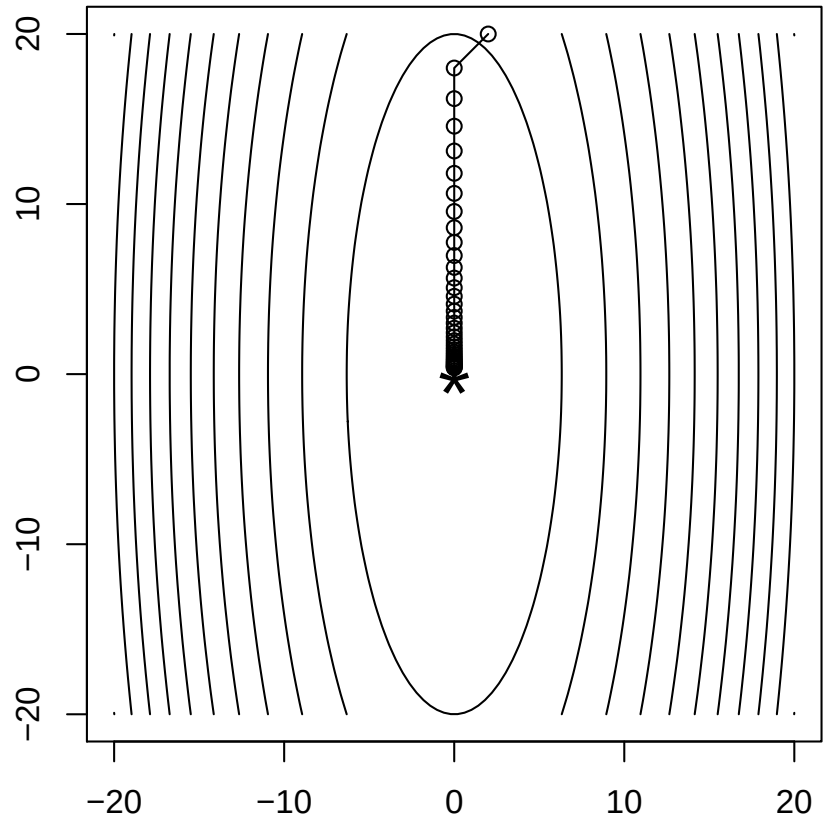
Simply take $t_k = t$ for all $k = 1, 2, 3, \ldots$, can diverge if $t$ is too big.
Consider $f(x) = (10x_1^2 + x_2^2)/2$, gradient descent after 8 steps:



Can be slow if $t$ is too small. Same example, gradient descent after 100 steps:



Converges nicely when $t$ is "just right". Same example, 40 steps:




Convergence analysis later will give us a precise idea of "just right"

# Backtracking line search

One way to adaptively choose the step size is to use backtracking line search:

- First fix parameters $0 < \beta < 1$ and $0 < \alpha \leq 1/2$
- At each iteration, start with $t = t_{\text{init}}$, and while

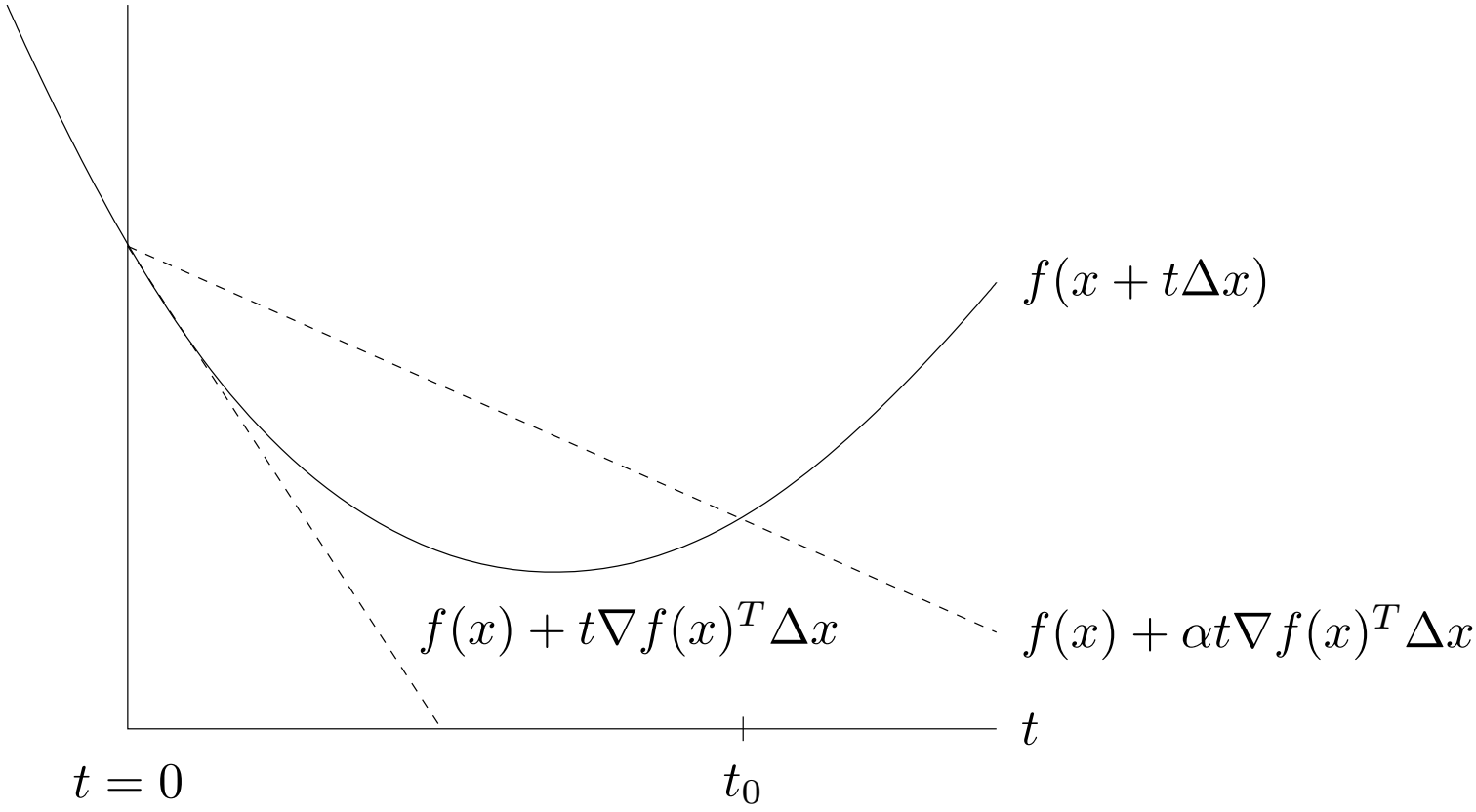
$$f(x - t\nabla f(x)) > f(x) - \alpha t \|\nabla f(x)\|_2^2$$

shrink $t = \beta t$. Else perform gradient descent update

$$x^+ = x - t\nabla f(x)$$

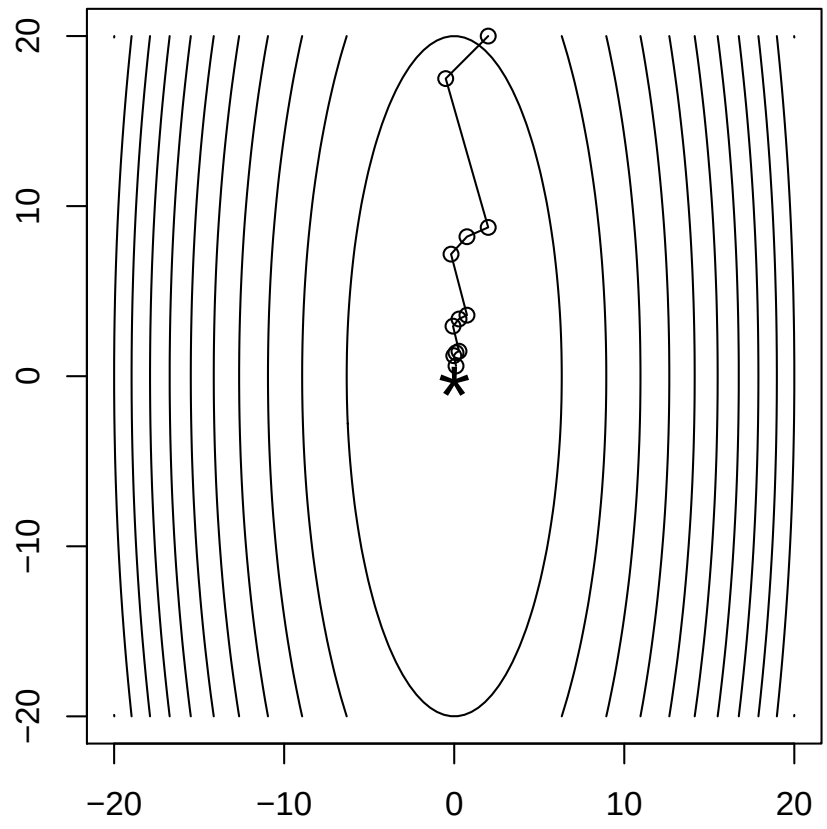
Simple and tends to work well in practice (further simplification: just take $\alpha = 1/2$)

# Backtracking interpretation




For us $\Delta x = -\nabla f(x)$

Setting $\alpha = \beta = 0.5$, backtracking picks up roughly the right step size (12 outer steps, 40 steps total),



# Exact line search

We could also choose step to do the best we can along direction of negative gradient, called exact line search:

$$t = \operatorname*{argmin}_{s \geq 0} \; f(x - s\nabla f(x))$$

Usually not possible to do this minimization exactly

Approximations to exact line search are typically not as efficient as backtracking, and it's typically not worth it

# Convergence analysis

Assume that $f$ convex and differentiable, with $\mathrm{dom}(f) = \mathbb{R}^n$, and additionally

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2 \quad \text{for any } x, y$$

I.e., $\nabla f$ is Lipschitz continuous with constant $L > 0$

**Theorem:** Gradient descent with fixed step size $t \leq 1/L$ satisfies

$$f(x^{(k)}) - f^\star \leq \frac{\|x^{(0)} - x^\star\|_2^2}{2tk}$$

and same result holds for backtracking, with $t$ replaced by $\beta/L$

We say gradient descent has convergence rate $O(1/k)$. I.e., it finds $\epsilon$-suboptimal point in $O(1/\epsilon)$ iterations

# Convergence under strong convexity

Reminder: strong convexity of $f$ means $f(x) - \frac{m}{2}\|x\|_2^2$ is convex for some $m > 0$

Assuming Lipschitz gradient as before, and also strong convexity:

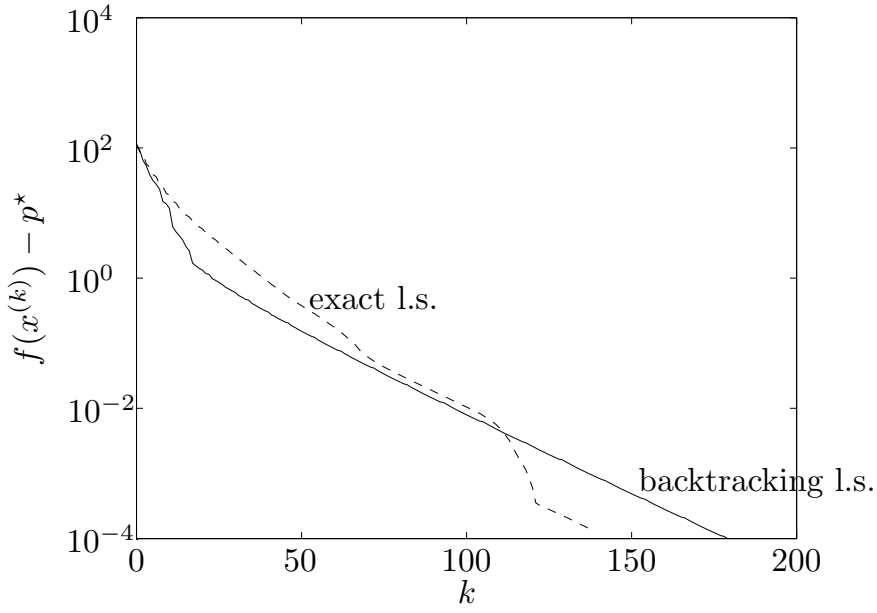
**Theorem:** Gradient descent with fixed step size $t \leq 2/(m+L)$ or with backtracking line search search satisfies

$$f(x^{(k)}) - f^\star \leq c^k \frac{L}{2}\|x^{(0)} - x^\star\|_2^2$$

where $0 < c < 1$

Rate under strong convexity is $O(c^k)$, exponentially fast! I.e., we find $\epsilon$-suboptimal point in $O(\log(1/\epsilon))$ iterations

Called linear convergence, because looks linear on a semi-log plot




(From B & V page 487)

Important note: contraction factor $c$ in rate depends adversely on condition number $L/m$: higher condition number $\Rightarrow$ slower rate

Affects not only our upper bound ... very apparent in practice too

# A look at the conditions

A look at the conditions for a simple problem, $f(\beta) = \frac{1}{2}\|y - X\beta\|_2^2$

Lipschitz continuity of $\nabla f$:

- This means $\nabla^2 f(x) \preceq LI$
- As $\nabla^2 f(\beta) = X^T X$, we have $L = \sigma_{\mathsf{max}}^2(X)$

Strong convexity of $f$:

- This means $\nabla^2 f(x) \succeq mI$
- As $\nabla^2 f(\beta) = X^T X$, we have $m = \sigma_{\mathsf{min}}^2(X)$
- If $X$ is wide (i.e., $X$ is $n \times p$ with $p > n$), then $\sigma_{\mathsf{min}}(X) = 0$, and $f$ can't be strongly convex
- Even if $\sigma_{\mathsf{min}}(X) > 0$, can have a very large condition number $L/m = \sigma_{\mathsf{max}}^2(X)/\sigma_{\mathsf{min}}^2(X)$

# Practicalities

Stopping rule: stop when $\|\nabla f(x)\|_2$ is small

- Recall $\nabla f(x^\star) = 0$ at solution $x^\star$
- If $f$ is strongly convex with parameter $m$, then

$$\|\nabla f(x)\|_2 \leq \sqrt{2m\epsilon} \implies f(x) - f^\star \leq \epsilon$$

Pros and cons of gradient descent:

- Pro: simple idea, and each iteration is cheap (usually)
- Pro: fast for well-conditioned, strongly convex problems
- Con: can often be slow, because many interesting problems aren't strongly convex or well-conditioned
- Con: can't handle nondifferentiable functions

# Can we do better?

Gradient descent has $O(1/\epsilon)$ convergence rate over problem class of convex, differentiable functions with Lipschitz gradients

First-order method: iterative method, which updates $x^{(k)}$ in

$$x^{(0)} + \mathrm{span}\{\nabla f(x^{(0)}), \nabla f(x^{(1)}), \ldots \nabla f(x^{(k-1)})\}$$

The span of the gradients is called a Krylov space.

**Theorem (Nesterov):** For any $k \leq (n-1)/2$ and any starting point $x^{(0)}$, there is a function $f$ in the problem class such that any first-order method satisfies

$$f(x^{(k)}) - f^\star \geq \frac{3L\|x^{(0)} - x^\star\|_2^2}{32(k+1)^2}$$

Can attain rate $O(1/k^2)$, or $O(1/\sqrt{\epsilon})$? Answer: yes (we'll see)!

# What about nonconvex functions?

Assume $f$ is differentiable with Lipschitz gradient as before, but now nonconvex. Asking for optimality is too much. So we'll settle for $x$ such that $\|\nabla f(x)\|_2 \leq \epsilon$, called $\epsilon$-stationarity

**Theorem:** Gradient descent with fixed step size $t \leq 1/L$ satisfies

$$\min_{i=0,\ldots,k} \|\nabla f(x^{(i)})\|_2 \leq \sqrt{\frac{2(f(x^{(0)}) - f^\star)}{t(k+1)}}$$

Thus gradient descent has rate $O(1/\sqrt{k})$, or $O(1/\epsilon^2)$, even in the nonconvex case for finding stationary points

This rate cannot be improved (over class of differentiable functions with Lipschitz gradients) by any deterministic algorithm[1]

[1] Carmon et al. (2017), "Lower bounds for finding stationary points I"

# Proof

Key steps:

- $\nabla f$ Lipschitz with constant $L$ means

$$f(y) \leq f(x) + \nabla f(x)^T(y-x) + \frac{L}{2}\|y-x\|_2^2 \quad \text{all } x, y$$

- Plugging in $y = x^+ = x - t\nabla f(x)$,

$$f(x^+) \leq f(x) - \left(1 - \frac{Lt}{2}\right)t\|\nabla f(x)\|_2^2$$

- Taking $0 < t \leq 1/L$, and rearranging,

$$\|\nabla f(x)\|_2^2 \leq \frac{2}{t}(f(x) - f(x^+))$$

- Summing over iterations

$$\sum_{i=0}^{k} \|\nabla f(x^{(i)})\|_2^2 \leq \frac{2}{t}(f(x^{(0)}) - f(x^{(k+1)})) \leq \frac{2}{t}(f(x^{(0)}) - f^\star)$$

- Lower bound sum by $(k+1)\min_{i=0,\ldots,k}\|\nabla f(x^{(i)})\|_2^2$, conclude

□

# Generalization of Strong Convexity

Linear convergence can be obtained under conditions weaker strong convexity.

1. Restricted Strong Convexity
2. Polyak-Łojasiewicz Condition

# Restricted Strong Convexity and Restricted Secant Inequality

1. (From HW1) An equivalent definition of $m$-strong convexity:

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq m\|x - y\|^2 \ \forall x, y.$$

2. **Restricted Strong Convexity** (Negahban, Ravikumar, Wainwright, Yu, NIPS2009), (Zhang and Cheng, 2014, Optimization Letter): (1) $f$ is convex; (2)$f$ obeys the Restricted Secant Inequality:

$$\langle \nabla f(x) - \nabla f(x_{\text{prj}}), x - x_{\text{prj}} \rangle \geq m\|x - x_{\text{prj}}\|^2 \ \forall x,$$

where $x_{\text{prj}}$ is the projection of $x$ to the solution set $\mathcal{X}^*$.

# RSI Functions are not necessarily convex

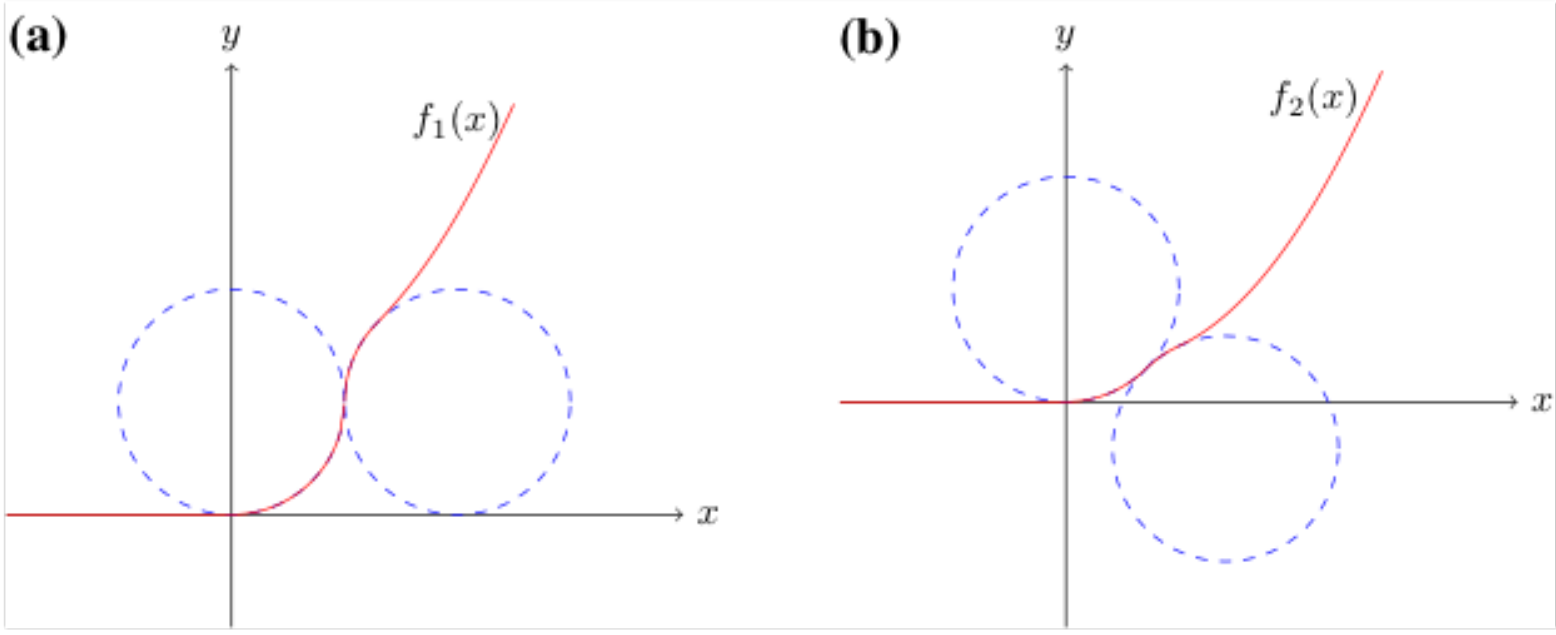
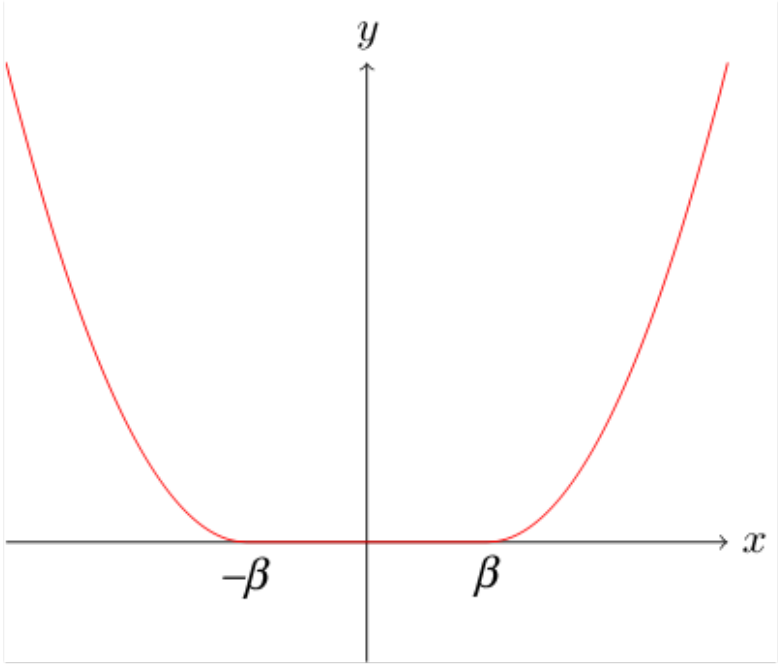
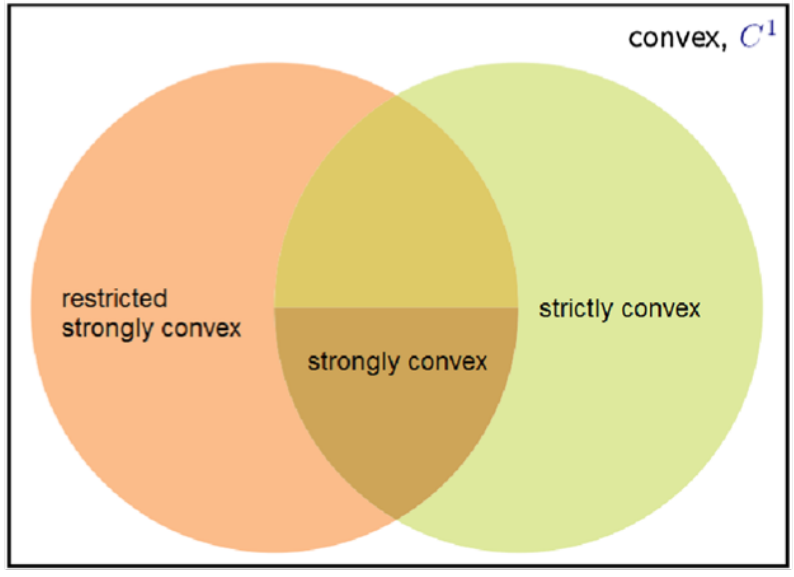



**Fig. 1** Non-convex functions satisfying RSI

(Examples taken from Zhang and Cheng, 2014 )

# RSC functions are not necessarily strictly convex.




(Illustration taken from Zhang and Cheng, 2014 )

1. If $g$ is RSC with parameter $m$, and $A \in \mathbb{R}^{m \times n}$ $(m \leq n)$ obeys that $g(Ax)$ RSC with $m' = m\lambda_{\min}(AA^T)$.
2. This fact was used to prove linear convergence of Lasso ( Is it strictly convex or not?) to statistical accuracy. (Agarwal, Negahban, Wainwright 2009, 2012)

# RSC is Sufficient and Necessary for Global Linear Convergence

**Theorem:** (Theorem 2 of Zhang and Cheng, 2014) If function $f$ is $L$-Smooth and $m$-RSC, then the gradient descent algorithm with fixed step size $t \leq 1/L$ converges linearly with

$$\|x^{(k+1)} - x_{\text{prj}}^{(k+1)}\| \leq (1 - m/L)^{1/2} \|x^{(k)} - x_{\text{prj}}^{(k)}\|.$$

Conversely, assuming $f$ has the unique solution $x^*$ and the gradient descent algorithm achieves a linear convergence rate from any $x^{(0)}$ with contraction ratio $1 - \delta$ for some $0 < \delta < 1$, then $f$ is RSC with parameter $m > 0$.

- How to prove sub-optimality convergence using this?

# Other conditions for linear convergence

1. Polyak-Łojasiewicz Condition (Polyak, 1963)

$$\frac{1}{2}\|\nabla f(x)\|^2 \geq m(f(x) - f^*),\ \forall x$$

2. Quadratic Growth Condition (Anitescu, 2000)

$$f(x) - f^* \geq \frac{m}{2}\|x_{\text{prj}} - x\|^2,\ \forall x$$

3. Error Bounds condition (Luo and Tseng, 1993)

$$\|\nabla f(x)\| \geq m\|x_{\text{prj}} - x\|,\ \forall x$$

**Theorem:** (Karimi, Nutini, Schmidt, ECML2016) For smooth functions $(RSI) \rightarrow (EB) \equiv (PL) \rightarrow (QG)$. If in addition, $f$ is convex, then $(RSC) \equiv (PL) \equiv (QC) \equiv (EB)$.

# Combine the PL condition and the Descent Lemma

1. Descent Lemma for $L$-smooth functions:

$$f(x^{(k+1)}) - f(x^{(k)}) \leq -\frac{1}{2L}\|\nabla f(x^{(k)})\|^2.$$

2. Apply the PL condition

$$f(x^{(k+1)}) - f(x^{(k)}) \leq -\frac{m}{L}(f^{(k)} - f^*).$$

3. Subtract $f^*$ on both sides

$$f(x^{(k+1)}) - f^* \leq (1 - \frac{m}{L})(f^{(k)} - f^*).$$

4. Linear convergence!

**Question**: Show that a $m$-strongly convexity $\Rightarrow$ $m$-PL condition!

# Gradient boosting

*The Annals of Statistics*
2001, Vol. 29, No. 5, 1189–1232

## 1999 REITZ LECTURE

## GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE[1]

BY JEROME H. FRIEDMAN

*Stanford University*

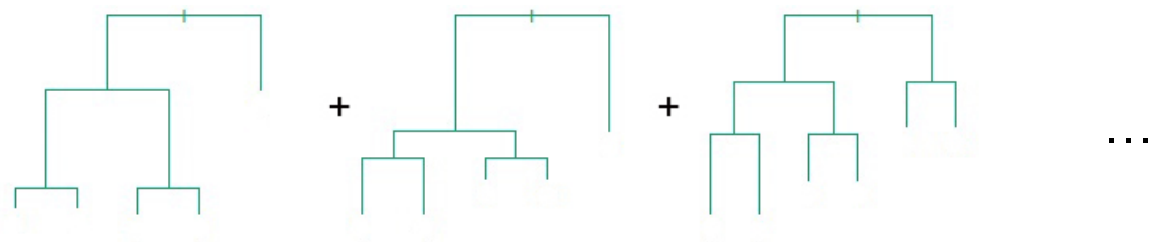
Function estimation/approximation is viewed from the perspective of numerical optimization in function space, rather than parameter space. A connection is made between stagewise additive expansions and steepest-descent minimization. A general gradient descent "boosting" paradigm is developed for additive expansions based on any fitting criterion. Specific algorithms are presented for least-squares, least absolute deviation, and Huber-$M$ loss functions for regression, and multiclass logistic likelihood for classification. Special enhancements are derived for the particular case where the individual additive components are regression trees, and tools for interpreting such "TreeBoost" models are presented. Gradient boosting of regression trees produces competitive, highly robust, interpretable procedures for both regression and classification, especially appropriate for mining less than clean data. Connections between this approach and the boosting methods of Freund and Shapire and Friedman, Hastie and Tibshirani are discussed.

Given responses $y_i \in \mathbb{R}$ and features $x_i \in \mathbb{R}^p$, $i = 1, \ldots n$

Want to construct a flexible (nonlinear) model for response based on features. Weighted sum of trees:

$$u_i = \sum_{j=1}^{m} \beta_j \cdot T_j(x_i), \quad i = 1, \ldots n$$

Each tree $T_j$ inputs $x_i$, outputs predicted response. Typically trees are pretty short



Pick a loss function $L$ to reflect setting. For continuous responses, e.g., could take $L(y_i, u_i) = (y_i - u_i)^2$

Want to solve

$$\min_{\beta} \sum_{i=1}^{n} L\Big(y_i, \sum_{j=1}^{M} \beta_j \cdot T_j(x_i)\Big)$$

Indexes all trees of a fixed size (e.g., depth $= 5$), so $M$ is huge. Space is simply too big to optimize

Gradient boosting: basically a version of gradient descent that is forced to work with trees

First think of optimization as $\min_u \ f(u)$, over predicted values $u$, subject to $u$ coming from trees

Start with initial model, a single tree $u^{(0)} = T_0$. Repeat:

- Compute negative gradient $d$ at latest prediction $u^{(k-1)}$,

$$d_i = -\left[\frac{\partial L(y_i, u_i)}{\partial u_i}\right]\Bigg|_{u_i = u_i^{(k-1)}}, \quad i = 1, \ldots n$$

- Find a tree $T_k$ that is close to $a$, i.e., according to

$$\min_{\text{trees } T} \sum_{i=1}^{n} (d_i - T(x_i))^2$$

  Not hard to (approximately) solve for a single tree

- Compute step size $\alpha_k$, and update our prediction:

$$u^{(k)} = u^{(k-1)} + \alpha_k \cdot T_k$$

  Note: predictions are weighted sums of trees, as desired

# References and further reading


- S. Boyd and L. Vandenberghe (2004), "Convex optimization", Chapter 9
- T. Hastie, R. Tibshirani and J. Friedman (2009), "The elements of statistical learning", Chapters 10 and 16
- Y. Nesterov (1998), "Introductory lectures on convex optimization: a basic course", Chapter 2
- L. Vandenberghe, Lecture notes for EE 236C, UCLA, Spring 2011-2012
- H. Zhang and L. Cheng. "Restricted strong convexity and its applications to convergence analysis of gradient-type methods in convex optimization." Optimization Letters 9.5 (2015): 961-979.
- H. Karimi, J. Nutini and M. Schmidt. "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition." in ECML-PKDD. (2016).