Lecture 2 – Part 1
Optimization

(January 16, 2015)

Mu Zhu
University of Waterloo

Need for Optimization

- $\mathbb{E}(y|\boldsymbol{x})$, $\mathbb{P}(y|\boldsymbol{x})$ — want to "go after" them
- first, model
  - some examples last week
- then, estimate
  - didn't discuss last week
  - often requires solving an optimization problem

Ex I: Linear Regression

$$\min_{\boldsymbol{\beta}} \quad \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}\right)^2 \quad = \quad \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2$$

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_n^{\mathrm{T}} \end{bmatrix}$$

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$$

Ex II: Logistic Regression

$$y_i \sim \text{Bernoulli}(p_i)$$

$$p_i \equiv \mathbb{P}(y_i = 1 | \boldsymbol{x}_i)$$

$$\log \frac{p_i}{1-p_i} = \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta} \quad \text{or} \quad p_i = \frac{\exp(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta})}{1 + \exp(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta})}$$

$$\mathcal{L}(p_i; y_i) = \prod_{i=1}^{n} p_i^{y_i} (1-p_i)^{1-y_i}$$

Ex II: Logistic Regression

- log-likelihood

$$\begin{aligned}
\ell(p_i; y_i) &= \log \mathcal{L}(p_i; y_i) \\
&= \sum_{i=1}^{n} y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \\
&= \sum_{i=1}^{n} y_i \log \frac{p_i}{1 - p_i} + \log(1 - p_i) \\
\ell(\boldsymbol{\beta}; y_i, \boldsymbol{x}_i) &= \sum_{i=1}^{n} y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta} - \log(1 + e^{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta}})
\end{aligned}$$

- to estimate $\boldsymbol{\beta}$,

$$\max_{\boldsymbol{\beta}} \quad \ell(\boldsymbol{\beta}; y_i, \boldsymbol{x}_i)$$

by Newton-Raphson

Ex II: Logistic Regression

$$\ell'(\boldsymbol{\beta}) = \sum_{i=1}^{n} y_i \boldsymbol{x}_i - \left[\frac{e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}}}{1 + e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}}} \right] \boldsymbol{x}_i = \sum_{i=1}^{n} (y_i - p_i)\boldsymbol{x}_i$$

$$= \begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n \end{bmatrix} \begin{bmatrix} y_1 - p_1 \\ \vdots \\ y_n - p_n \end{bmatrix} = \boldsymbol{X}^{\mathrm{T}}(\boldsymbol{y} - \boldsymbol{p})$$

Ex II: Logistic Regression

$$
\begin{aligned}
& \ell''(\boldsymbol{\beta}) \\
= & -\sum_{i=1}^{n} \boldsymbol{x}_i \left[\frac{(e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}})(1+e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}}) - (e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}})(e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}})}{(1+e^{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta}})^2} \right] \boldsymbol{x}_i^{\mathrm{T}} \\
= & -\sum_{i=1}^{n} \boldsymbol{x}_i \left[p_i - p_i^2 \right] \boldsymbol{x}_i^{\mathrm{T}} \\
= & -\begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n \end{bmatrix} \begin{bmatrix} p_1(1-p_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & p_n(1-p_n) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_n^{\mathrm{T}} \end{bmatrix} \\
= & -\boldsymbol{X}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{X}
\end{aligned}
$$

Ex II: Logistic Regression

$$
\begin{aligned}
\boldsymbol{\beta}_{new} &= \boldsymbol{\beta}_{old} - \left[\ell''(\boldsymbol{\beta}_{old})\right]^{-1}\left[l'(\boldsymbol{\beta}_{old})\right] \\
&= \boldsymbol{\beta}_{old} + \left[\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right]^{-1}\left[\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{y}-\boldsymbol{p})\right] \\
&= \underbrace{\left[\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right]^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}}_{\text{weighted least squares}}\left[\boldsymbol{X}\boldsymbol{\beta}_{old} + \boldsymbol{W}^{-1}(\boldsymbol{y}-\boldsymbol{p})\right]
\end{aligned}
$$

(both $\boldsymbol{W}$ and $\boldsymbol{p}$ depend on $\boldsymbol{\beta}_{old}$)

$w_{ii} \equiv p_i(1-p_i)$ max at $p_i = 1/2$ and min at $p_i = 0$ or $p_i = 1$

$\Downarrow$

estimates influenced mostly by points near decision boundary

Enter "Big Data"

- if all this sounds easy, think about $\boldsymbol{x} \in \mathbb{R}^d$ and d is large relative to n
- end up estimating too much with too little
- resulting estimate cannot be very good
- in statistically terms, $\mathbb{V}\text{ar}(\widehat{\boldsymbol{\beta}})$ inflated
- to reduce variance, introduce bias

Penalized Regression

- let $\boldsymbol{x}_1, ..., \boldsymbol{x}_d$ denote columns of $\boldsymbol{X}$
- suppose $\boldsymbol{y}, \boldsymbol{x}_1, ..., \boldsymbol{x}_d$ all standardized ($\mathbf{1}^{\mathrm{T}}\boldsymbol{y} = 0$, $\|\boldsymbol{y}\| = 1$, etc)
- bias each β_j by introducing penalty $J(\beta_j)$

$$\min_{\boldsymbol{\beta}} \quad \frac{1}{2}\|\boldsymbol{y} - (\beta_1\boldsymbol{x}_1 + ... + \beta_d\boldsymbol{x}_d)\|^2 + \sum_{j=1}^{d} J(\beta_j)$$

A Class of Penalty Functions

$$J(\beta_j) = \lambda|\beta_j|^{\alpha} \quad \Rightarrow \quad \sum_{j=1}^{d} J(\beta_j) = \lambda \sum_{j=1}^{d} |\beta_j|^{\alpha} \equiv \lambda \|\boldsymbol{\beta}\|_{\alpha}$$

	α	$\|\cdot\|_{\alpha}$	difficulty	algorithm
ridge regression	2	ℓ_2	convex	analytic (exercise)
LASSO	1	ℓ_1	convex	coordinate descent
subset selection	0	ℓ_0	NP-hard	heuristic

$$|\beta_j|^0 = \begin{cases} 0, & \beta_j = 0 \\ 1, & \beta_j \neq 0 \end{cases} \quad \Rightarrow \quad \sum |\beta_j|^0 = \sum I(\beta_j \neq 0)$$

Bias-Variance Trade-off

Exercise Show that, under typical model assumptions,

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad \mathbb{E}(\boldsymbol{\varepsilon}) = \boldsymbol{0}, \quad \text{and} \quad \mathbb{V}\text{ar}(\boldsymbol{\varepsilon}) = \sigma^2\boldsymbol{I},$$

(i) the "usual" estimate, $\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$, is unbiased;

(ii) the ridge regression estimate — call it $\widehat{\boldsymbol{\beta}}_\lambda$ — is biased, but $\exists$ orthonormal matrix $\boldsymbol{V}$ such that

$$\mathbb{V}\text{ar}(\widehat{\boldsymbol{\beta}}_\lambda) = \sigma^2\boldsymbol{V}\boldsymbol{D}_\lambda\boldsymbol{V}^{\mathrm{T}}, \quad \mathbb{V}\text{ar}(\widehat{\boldsymbol{\beta}}) = \sigma^2\boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^{\mathrm{T}},$$

and

$$\boldsymbol{D}_\lambda(j,j) \leq \boldsymbol{D}(j,j) \quad \forall \quad j.$$

(Hint: Use the singular value decomposition of $\boldsymbol{X}$.)

Ex III: LASSO

- coordinate descent — at each iteration, solve a univariate problem,

$$\min_{\beta_j} \quad L(\beta_j) \quad = \quad \frac{1}{2}\|\boldsymbol{z} - \beta_j \boldsymbol{x}_j\|^2 + \lambda|\beta_j| + c_j,$$

where

$$\boldsymbol{z} \equiv \boldsymbol{y} - \sum_{k \neq j} \beta_k \boldsymbol{x}_k, \quad c_j = \sum_{k \neq j} \lambda|\beta_k|,$$

while fixing all β_k, $k \neq j$.

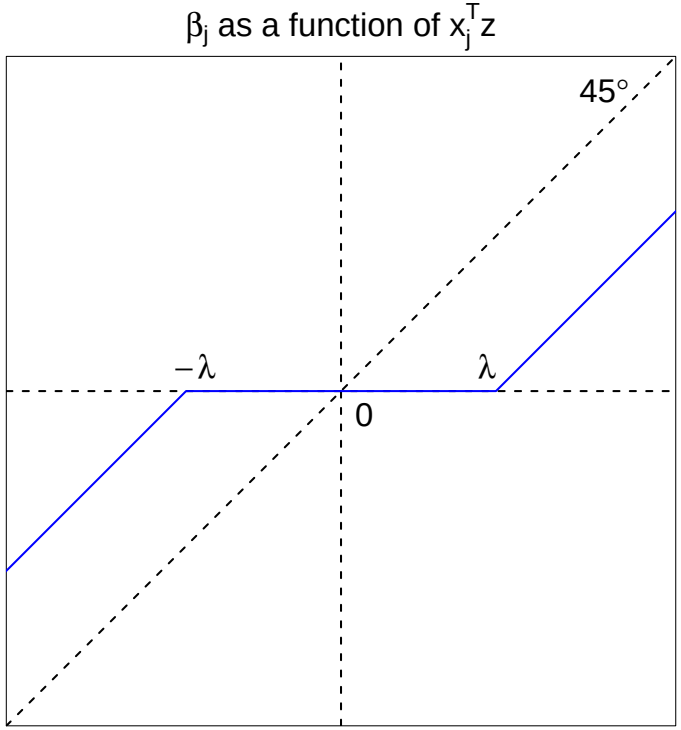
- cycle through $j = 1, 2, ..., d, 1, 2, ..., d, ...$

Ex III: LASSO

$$\frac{d}{d\beta_j} L(\beta_j) = 0$$

$$\beta_j \underbrace{(\boldsymbol{x}_j^{\mathrm{T}} \boldsymbol{x}_j)}_{\|\boldsymbol{x}_j\|^2 = 1} + \lambda \cdot \mathrm{sgn}(\beta_j) = \boldsymbol{x}_j^{\mathrm{T}} \boldsymbol{z}$$

$$\beta_j = \begin{cases} \boldsymbol{x}_j^{\mathrm{T}} \boldsymbol{z} - \lambda, & \beta_j > 0; \\ \boldsymbol{x}_j^{\mathrm{T}} \boldsymbol{z} + \lambda, & \beta_j < 0 \end{cases}$$



solution will typically contain many zeros, i.e., be sparse

selection effect of the LASSO

The ℓ_1 Penalty

D. L. Donoho (2006), "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution", *Communications on Pure and Applied Mathematics* **59**, pp. 797–829.

- ℓ_1-problem as a convex relaxation of ℓ_0-problem

Ex IV: Graphical LASSO

- from $\boldsymbol{x}_i \overset{iid}{\sim} \mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, estimate $\boldsymbol{\Omega} \equiv \boldsymbol{\Sigma}^{-1}$ (hard for d large)
- recall linear discriminant analysis requires $\boldsymbol{\Sigma}^{-1}$ (last week)
- for $\boldsymbol{x} = (x_1, ..., x_d)^{\mathrm{T}} \sim \mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, variables x_j and x_k conditionally independent given all other variables if and only if $\boldsymbol{\Omega}_{jk} = 0$ (Dempster, 1972; *Biometrics*)
- for Gaussian graphical models, $\exists$ edge between nodes j and k if and only if $\boldsymbol{\Omega}_{jk} \neq 0$

Ex IV: Graphical LASSO

$$\widehat{\boldsymbol{\mu}} = \bar{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i, \quad \boldsymbol{S} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})^{\mathrm{T}}$$

$$\mathcal{L}(\boldsymbol{\Sigma}) = \prod_{i=1}^{n}\frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}|}}\exp\left[-\frac{1}{2}(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})\right]$$

$$\begin{aligned}\ell(\boldsymbol{\Omega}) &= \text{const} + \frac{n}{2}\log|\boldsymbol{\Sigma}^{-1}| - \frac{1}{2}\sum_{i=1}^{n}\text{tr}\left[(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}})\right] \\ &= \text{const} + \frac{n}{2}\log|\boldsymbol{\Omega}| - \frac{n}{2}\text{tr}\,(\boldsymbol{\Omega}\boldsymbol{S})\end{aligned}$$

Ex IV: Graphical LASSO

- maximize ℓ_1-penalized likelihood (Friedman, Hastie & Tibshirani, 2008; *Biostatistics*):

$$\max_{\boldsymbol{\Omega} \succeq 0} \quad \underbrace{\log |\boldsymbol{\Omega}| - \operatorname{tr}(\boldsymbol{\Omega}\boldsymbol{S})}_{\ell(\boldsymbol{\Omega})} - \lambda \|\boldsymbol{\Omega}\|_1$$

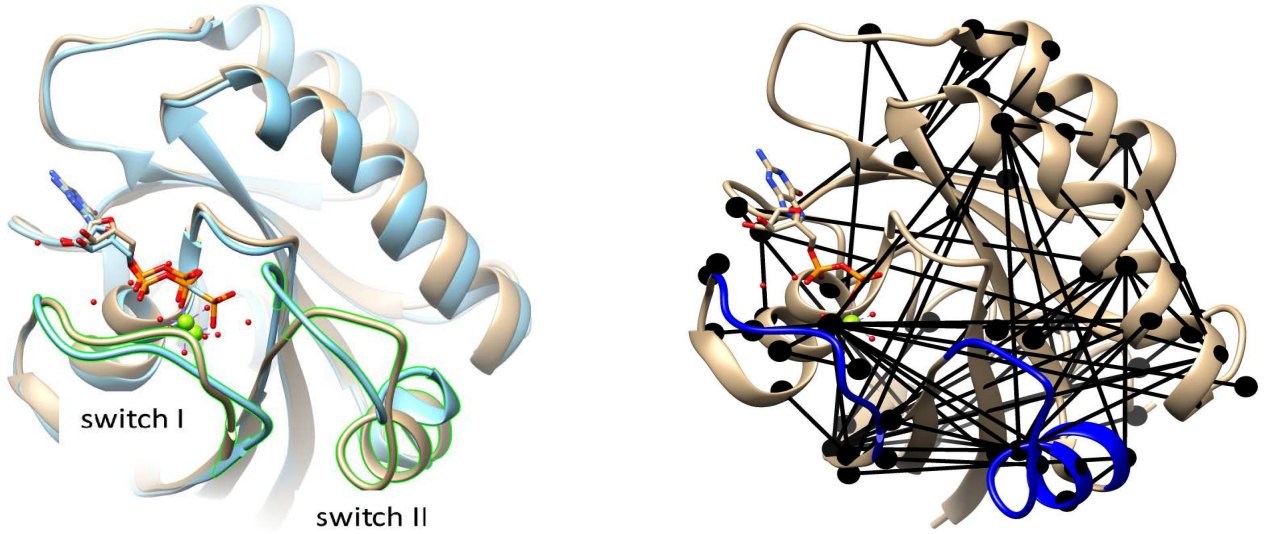
  where

$$\|\boldsymbol{\Omega}\|_1 = \sum_{j,k} |\boldsymbol{\Omega}_{jk}|$$

  by coordinate descent (one row/column at a time)

- resulting graphical model is sparse in the number of edges

Ex IV: Graphical LASSO



H-Ras. *Left*: Inactive form (4q21) & active form (6q21). Regions labelled "Switch I" & "Switch II" are known to undergo major conformational changes between 4q21 and 6q21. *Right*: Estimated graphical model showing conditional dependence structures, obtained by analysing 4q21 alone. (L. Soltan-Ghoraie, F. Burkowski & M. Zhu)

Ex V: The Netflix Problem (Part 1)

- high-profile, million-dollar Netflix contest (2006-09)
- rating matrix $\boldsymbol{R}$ where

$$r_{ui} = \text{rating of item } i \text{ by user } u$$

- set $\mathbb{T} = \{(u, i) : r_{ui} \text{ observed}\}$
- want to predict the missing entries of $\boldsymbol{R}$

Ex V: The Netflix Problem (Part 1)

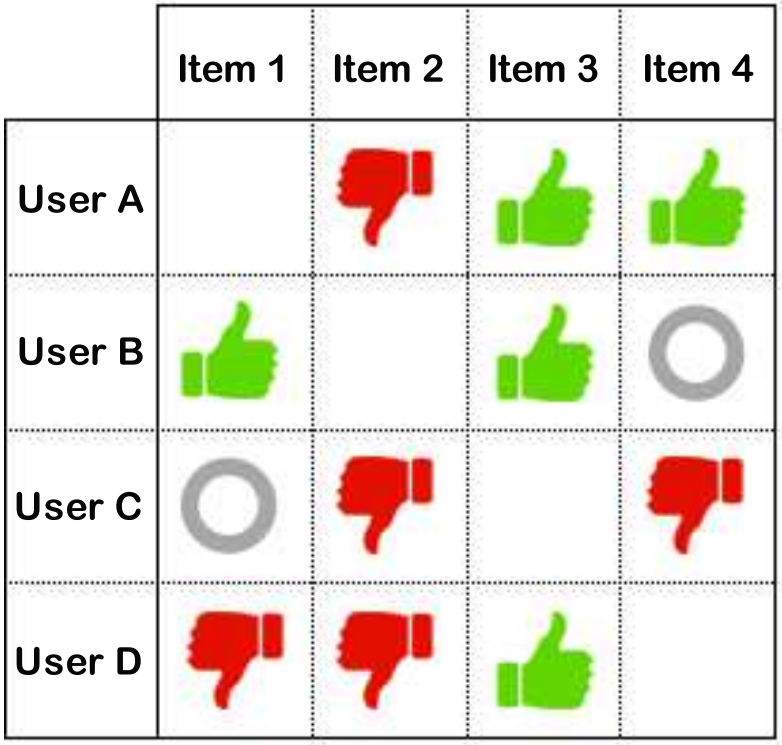


Illustration of the Rating Matrix $\boldsymbol{R}$

Ex V: The Netflix Problem (Part 1)

- want to solve

$$\min_{\widehat{\boldsymbol{R}}} \quad \text{rank}(\widehat{\boldsymbol{R}}) \quad \text{s.t.} \quad \widehat{r}_{ui} = r_{ui} \text{ for } (u,i) \in \mathbb{T}$$

- philosophy: only a few factors affect user preferences, so rank of rating matrix must be low

- but problem is NP-hard

Ex V: The Netflix Problem (Part 1)

- instead, solve convex relaxation,

$$\min_{\widehat{\boldsymbol{R}}} \quad \|\widehat{\boldsymbol{R}}\|_* \quad \text{s.t.} \quad \widehat{r}_{ui} = r_{ui} \text{ for } (u,i) \in \mathbb{T}$$

  where $\|\cdot\|_*$ denotes the nuclear norm of a matrix

- let $\sigma_1, \sigma_2, ...$ be singular values of $\boldsymbol{A}$; then,

$$\|\boldsymbol{A}\|_* = \sum |\sigma_j|^1$$

  whereas

$$\text{rank}(\boldsymbol{A}) = \sum I(\sigma_j \neq 0) = \sum |\sigma_j|^0$$

- a matter of ℓ_1 vs ℓ_0

Ex V: The Netflix Problem (Part 1)

E. J. Candès & B. Recht (2009), "Exact matrix completion via convex optimization", *Found Comput Math* **9**, pp. 717–772.

B. Recht, M. Fazel, & P. A. Parrilo (2010), "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization", *SIAM Review* **52**, pp. 471–501.

- under certain conditions, $\min \|\cdot\|_* \Leftrightarrow \min \text{rank}(\cdot)$
- $\min \|\cdot\|_* \Leftrightarrow$ a semi-definite program

Ex VI: The Netflix Problem (Part 2)

- explicit parameterization

$$\boldsymbol{R} \approx \boldsymbol{P}\boldsymbol{Q}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{p}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{p}_n^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_1 & \cdots & \boldsymbol{q}_m \end{bmatrix}$$

- $\boldsymbol{p}_u, \boldsymbol{q}_i \in \mathbb{R}^K$ $(K \ll n, m)$ are latent coordinates

- just estimate $\boldsymbol{p}_u, \boldsymbol{q}_i$ and predict missing entries with $\widehat{r}_{ui} = \boldsymbol{p}_u^{\mathrm{T}} \boldsymbol{q}_i$

- in reality, user- and item-effects are removed prior to doing this

Ex VI: The Netflix Problem (Part 2)

$$\min_{\boldsymbol{p}_u, \boldsymbol{q}_i} \quad \sum_{(u,i)\in\mathbb{T}} \left(r_{ui} - \boldsymbol{p}_u^{\mathrm{T}}\boldsymbol{q}_i\right)^2 + \lambda\left[\sum_u \|\boldsymbol{p}_u\|^2 + \sum_i \|\boldsymbol{q}_i\|^2\right]$$

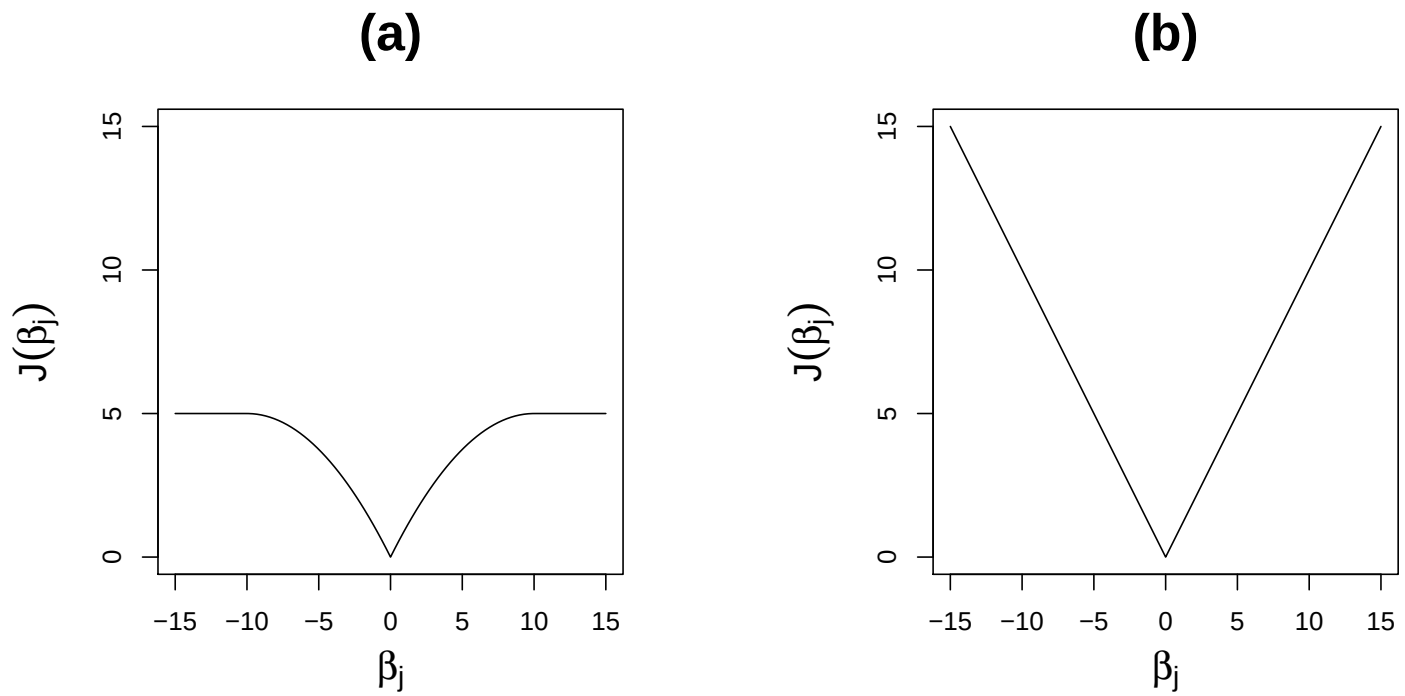
- coordinate descent still applies (over $\boldsymbol{p}_1, ..., \boldsymbol{p}_n$, $\boldsymbol{q}_1, ..., \boldsymbol{q}_m$, ...)
- strictly speaking, blockwise coordinate descent
- each step convex, but overall nonconvex (both $\boldsymbol{p}_u$, $\boldsymbol{q}_i$ unknown)
- many traps (e.g., local solutions, saddle points)

Ex VII: MCP

For the penalized regression problem, Zhang (2010; *Ann. Stat.*) proposed the so-called minimax concave penalty (MCP):

$$J(\beta_j) = \lambda \int_0^{|\beta_j|} \left(1 - \frac{x}{\gamma\lambda}\right)_+ dx = \begin{cases} \lambda|\beta_j| - \frac{\beta_j^2}{2\gamma}, & |\beta_j| \le \gamma\lambda; \\ \frac{1}{2}\gamma\lambda^2, & |\beta_j| > \gamma\lambda, \end{cases}$$

instead of $J(\beta_j) = \lambda|\beta_j|^\alpha$.



(a) MCP ($\lambda = 1$, $\gamma = 10$); (b) LASSO ($\lambda = 1$).

Ex VII: MCP

- for $|\beta_j|$ large,
  - $J(\cdot)$ constant "beyond a certain point" — reduces bias
- for $|\beta_j|$ small,
  - $J(\cdot)$ still "like the LASSO" — keeps sparsity
- in between,
  - smooth interpolation that minimizes maximal concavity
- coordinate descent applies,
  - but many traps (e.g., local solutions, saddle points)

Summary

- key ideas:
  - introduce bias to reduce variance; penalty functions
  - ℓ_1 norm; nuclear norm; nonconvex penalties
- specific methods:
  - ridge; LASSO; MCP
  - graphical LASSO
  - matrix completion; matrix factorization
  - Newton-Raphson; coordinate descent
- application areas:
  - proteins
  - recommender systems

Next ...

- lecture @ 2 pm by Professor S. Vavasis on optimization
- a short, 10-minute break
- research with my students (mostly, the Netflix Problem)