Regularization techniques

Jitkomut Songsiri

Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University

CUEE

January 27, 2023

Outline

Overview of regularization

Overview

we provide a concept of estimation with two objectives:

$$\underset{x}{\text{minimize}} \quad f(x) := g(x) + \gamma h(x)$$

- x is model parameter
- g is a loss function that indicates **model fitting**
- h is a **regularization function** that affects solution properties (aka **penalty**)
- $\gamma > 0$ is a penalty weight controlling a balance between model quality and regularization of x

we will layout the ideas by demostrating with a quadratic loss first

when g is a least-squares loss function

Overview

typyical characteristic of least-squares solutions to

$$\underset{\beta}{\text{minimize}} \quad \|y - X\beta\|_2, \quad y \in \mathbf{R}^N, \quad \beta \in \mathbf{R}^p$$

- entries in the solution β are nonzero
- if $p \gg N$, LS estimate is not unique

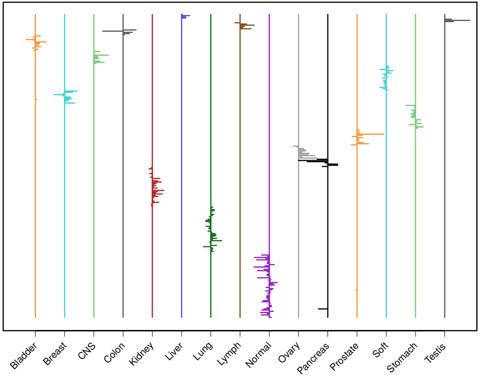
one can **regularize** the estimation process by solving

$$\underset{\beta}{\text{minimize}} \, \|y - X\beta\|_2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

- regard that $\|\beta\|_1 \leq t$ is our budget on the norm of parameter
- using ℓ_1 norm and small t yield a **sparse** solution

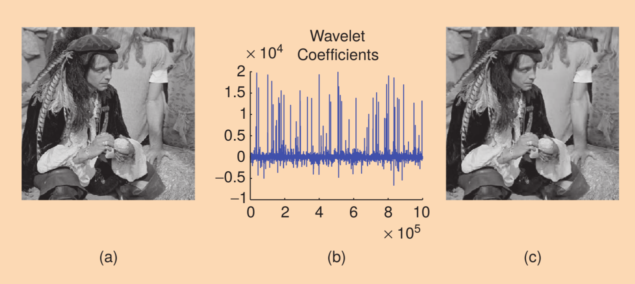
Example: 15-class gene expression cancer

example: 15-class gene expression cancer data



feature weights estimated from a lasso-regularized multinomial classifier (sparse)

Example: image reconstruction by wavelet representation



- zeroing out the wavelet coefficient but keeping the largest 25,000 ones
- relatively few wavelet coefficients capture most of the signal energy
- the difference between the original image (left) and the reconstructed image (right) are hardly noticeable

Why regularizations are needed?

reasons for alternatives to the least-squares estimate

- **prediction accuracy:**
  - LS estimate has low bias but large variance
  - shrinking some entries of β to zero introduces some bias but reduce the variance of β
  - when making predictions on new data set, it may improve the overall prediction accuracy
- **interpretation:** when having a large number of predictors, we often would like to identify a *smaller* subset of β that exhibit *strongest* effects

ℓ_2 regularization

ℓ_2-regularized least-squares

adding the 2-norm penalty to the objective function

$$\underset{\beta}{\text{minimize}} \quad \|y - X\beta\|_2^2 + \gamma\|\beta\|_2^2$$

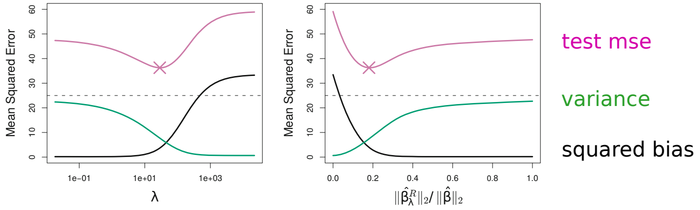
- seek for an approximate solution of $X\beta \approx y$ with small norm
- also called **Tikhonov regularized least-squares** or **ridge regression**
- $\gamma > 0$ controls the trade off between the fitting error and the size of x
- has the analytical solution for any $\gamma > 0$:

$$\beta = (X^T X + \gamma I)^{-1} X^T y$$

  (no restrictions on shape, rank of X)
- interpreted as a MAP estimation with the log-prior of the Gaussian

MSE of ridge regression

test mse versus regularization parameter λ



- as λ increases, we have a trade-off between bias and variance
- variance drops significantly as λ from 0 to 10 with little increase in bias; this leads MSE to decrease
- MSE at $\lambda = \infty$ is as high as MSE at $\lambda = 0$ but the minimum MSE is acheived at intermediate value of λ

Similar form of ℓ_2-regularized LS

the ℓ_2-norm is an inequality constraint:

$$\underset{\beta}{\text{minimize}} \quad \|y - X\beta\|_2 \quad \text{subject to} \quad \beta_1^2 + \cdots + \beta_p^2 \leq t$$

- t is specified by the user
- t serves as a budget of the sum squared of β
- the ℓ_2-regularized LS on page 10 is the Lagrangian form of this problem
- for every value of γ on page 10 there is a corresponding t such that the two formulations have the same estimates of β

Practical issues

some concerns on implementing ridge regression

- the ℓ_2 penalty on β should NOT apply to the intercept β_0 since β_0 measures the mean value of the response when $x_1, \ldots, x_p$ are zero
- ridge solutions are **not equivariant** under scaling of inputs: $\tilde{x}_j = \alpha_j x_j$

$$\tilde{X} = \begin{bmatrix} \alpha_1 x_1 & \alpha_2 x_2 & \cdots & \alpha_p x_p \end{bmatrix} \triangleq XD$$

- $\hat{\beta}_j$ depends on λ and the scaling of other predictors

$$\hat{\beta} = (D^T X^T X D + \gamma I)^{-1} D^T X^T y$$

- it is best to apply ℓ_2 regularization after **standardizing** X

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}} \quad \text{(all predictors are on the same scale)}$$

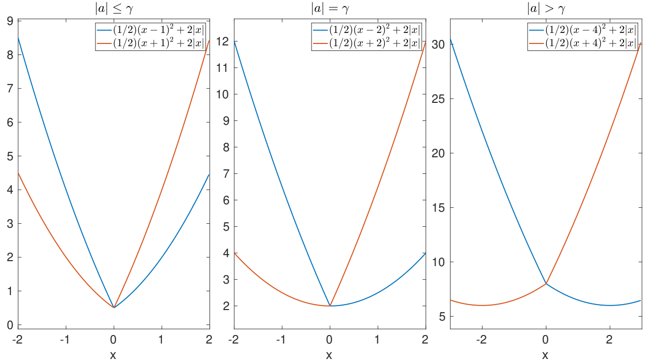
ℓ_1 regularization

Scalar ℓ_1-regularized least-squares

Idea: adding $|x|$ to a minimization problem introduces a sparse solution

consider a scalar problem:

$$\underset{x}{\text{minimize}} \quad f(x) = (1/2)(x-a)^2 + \gamma|x|$$



Optimal solution

to derive the optimal solution, we consider the two cases:

- if $x \geq 0$ then $f(x) = (1/2)(x - (a - \gamma))^2$ (parabola with center at $a - \gamma$)

  $$x^\star = a - \gamma, \quad \text{provided that } a \geq \gamma$$

  if $a < \gamma$, then $x^\star = 0$ (because parabola f is centered at $a - \gamma$ which is negative)
- if $x \leq 0$ then $f(x) = (1/2)(x - (a + \gamma))^2$

  $$x^\star = a + \gamma, \quad \text{provided that } a \leq -\gamma$$

  if $a \geq -\gamma$ then $x^\star = 0$ (because parabola f is centered at $a + \gamma$ which is positive)

conclusion: when $|a| \leq \gamma$ then $x^\star$ must be zero the optimal solution to minimization of $f(x) = (1/2)(x-a)^2 + \gamma|x|$ is

$$x^\star = \begin{cases} (|a| - \gamma)\mathbf{sign}(a), & |a| > \gamma \\ 0, & |a| \leq \gamma \end{cases}$$

meaning: if γ is large enough, x^* will be zero

generalization to vector case: $x \in \mathbf{R}^n$

$$\underset{x}{\text{minimize}} \quad f(x) = (1/2)\|x-a\|^2 + \gamma\|x\|_1$$

the optimal solution has the same form

$$x^\star = \begin{cases} (|a| - \gamma)\mathbf{sign}(a), & |a| > \gamma \\ 0, & |a| \leq \gamma \end{cases}$$

where all operations are done in *elementwise*

ℓ_1-regularized least-squares

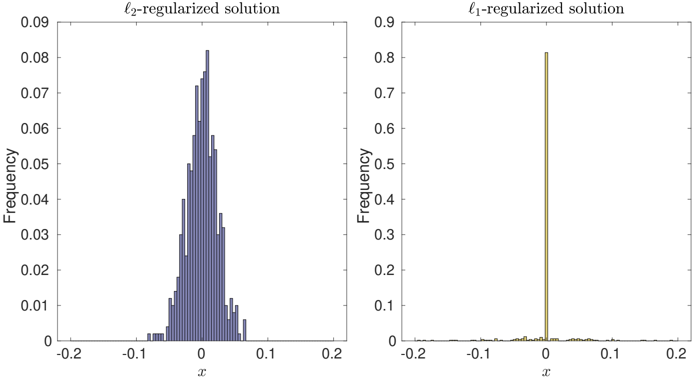
adding the ℓ_1-norm penalty to the least-square problem

$$\underset{\beta}{\text{minimize}} \quad (1/2)\|y - X\beta\|_2^2 + \gamma\|\beta\|_1, \quad y \in \mathbf{R}^N, \quad \beta \in \mathbf{R}^p$$

- a convex heuristic method for finding a sparse β that gives $X\beta \approx y$
- also called **Lasso** or **basis pursuit**
- a nondifferentiable problem due to $\|\cdot\|_1$ term
- no analytical solution, but can be solved efficiently
- interpreted as a MAP estimation with the log-prior of the Laplacian distribution

Example

$X \in \mathbf{R}^{m \times n}, b \in \mathbf{R}^m$ with $m = 100, n = 500, \gamma = 0.2$



- solution of ℓ_2 regularization is more widely spread
- solution of ℓ_1 regularization is concentrated at zero

Similar form of ℓ_1-regularized LS

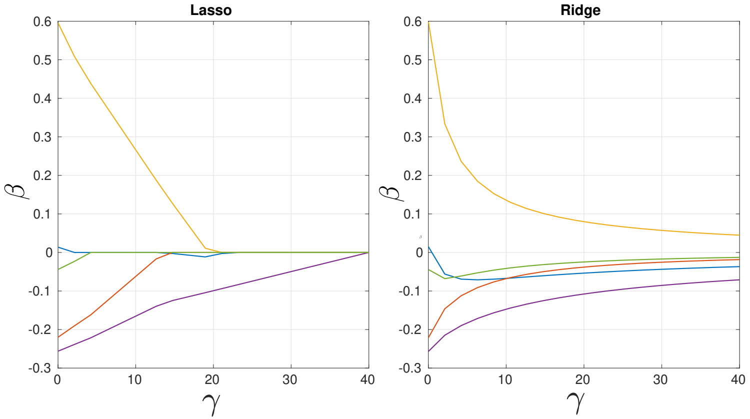
the ℓ_1-norm is an inequality constraint:

$$\underset{\beta}{\text{minimize}} \quad \|y - X\beta\|_2 \quad \text{subject to} \quad \|\beta\|_1 \leq t$$

- t is specified by the user
- t serves as a budget of the sum of absolute values of x
- the ℓ_1-regularized LS on page 18 is the Lagrangian form of this problem
- for each t where $\|\beta\|_1 \leq t$ is active, there is a corresponding value of γ that yields the same solution from page 18

Solution paths of regularized LS

solve the regularized LS when $n = 5$ and vary γ (penalty parameter)



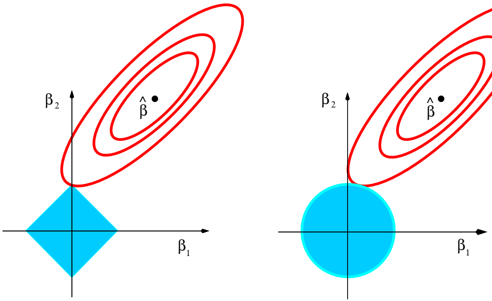
- for lasso, many entries of β are exactly zero as γ varies
- for ridge, many entries of β are nonzero but converging to small values

Contour of quadratic loss and constraints

both regularized LS problems has the objective function: minimize$_\beta$ $\|y - X\beta\|_2^2$

but with different constraints:

ridge: $\beta_1^2 + \cdots + \beta_p^2 \leq t$ **lasso:** $|\beta_1| + \cdots + |\beta_p| \leq t$



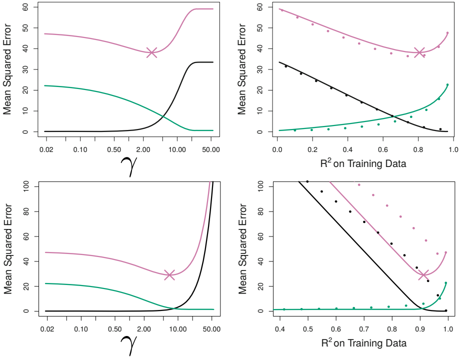
the contour can hit a corner of ℓ_1-norm ball where some β_k must be zero

Comparing ridge and lasso

left: as γ increases, lasso estimate gives a trade-off in variance and bias

true beta is dense

true beta is sparse



test mse

variance

squared bias

dotted: ridge

solid: lasso

- plot test MSE against R^2 on training data to compare the two models
- dense ground-truth: minimum MSE of ridge is smaller than that of lasso
- sparse ground-truth: lasso tends to outperform ridge in term of bias, variance and MSE

Subgradient calculus for computing lasso

standardized *one-predictor lasso* formulation:

$$\underset{\beta}{\text{minimize}} \quad \frac{1}{2N}\sum_{i=1}^{N}(y_i - x_i\beta)^2 + \gamma|\beta|$$

standardization: $\frac{1}{N}\sum_i^N y_i = 0$, $\frac{1}{N}\sum_i x_i = 0$, and $\frac{1}{N}\sum_i x_i^2 = 1$

- the term $f(\beta) = |\beta|$ is non-differentiable at zero
- convex theory: g is a subgradient of f at x if it satisfies

$$f(y) \geq f(x) + g^T(y - x), \quad \forall y \in \mathbf{dom}\, f$$

  (which is similar to the first-order condition for a convex function)
- a subgradient is not unique; subgradient of $|\beta|$ is any number between -1 and 1 (or simply $\mathbf{sign}(\beta)$)
- a subgradient of $f(\beta) = \|\beta\|_1$ is g where $\|g\|_\infty \leq 1$

Optimality condition of scalar lasso

optimality condition (with subgradient g): use notation $\sum_i x_i y_i = \langle x, y \rangle$

$$\beta + \gamma g = \frac{1}{N}\langle x, y \rangle \qquad \text{(effect of } N \text{ is apparent)}$$

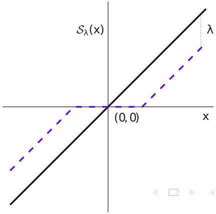
where $g = \mathbf{sign}(\beta)$ if $\beta \neq 0$ and $g \in [-1, 1]$ if $\beta = 0$

the optimality condition can be written as

$$\hat{\beta} = \begin{cases} \frac{1}{N}\langle x, y \rangle - \gamma, & \text{if } \frac{1}{N}\langle x, y \rangle > \gamma \\ 0, & \text{if } \frac{1}{N}\langle x, y \rangle \leq \gamma \\ \frac{1}{N}\langle x, y \rangle + \gamma, & \text{if } \frac{1}{N}\langle x, y \rangle < -\gamma \end{cases}$$

a lasso estimate can be expressed using **soft-thresholding operator**

$$\hat{\beta} = \mathcal{S}_\gamma\left(\frac{1}{N}\langle x, y \rangle\right), \quad S_\gamma(z) = \mathbf{sign}(z)(|z| - \gamma)_+$$



Properties of lasso formulation

lasso formulation: minimize$_\beta$ $(1/2)\|y - X\beta\|_2^2 + \gamma\|\beta\|_1$

- it is a quadratic programming (and hence, convex)
- when X is not full column rank (either $p \leq N$ with colinearity or $p \geq N$), the LS fitted values are unique but $\hat{\beta}$ is not
- when $\gamma > 0$ and if X are in general position (Hastie et.al 2015) then the lasso solutions are unique
- the optimality condition from the convex theory is

$$-X^T(y - X\beta) + \gamma g = 0$$

where $g = (g_1, \ldots, g_p)$ is a subgradient of $\|\cdot\|_1$

$$g_i = \mathbf{sign}(\beta_i) \quad \text{if } \beta_i \neq 0, \quad g_i \in [-1, 1] \quad \text{if } \beta_i = 0$$

Computing lasso estimate in practice

standardization: on the predictor matrix X ($\hat{\beta}$ would not depend on the units)

- each column is centered: $\frac{1}{N}\sum_{i=1}^{N} x_{ij} = 0$
- each column has unit variance: $\frac{1}{N}\sum_{i=1}^{N} x_{ij}^2 = 1$

standardization: on the response y (so that the intercept term β_0 is not needed)

- centered at zero mean: $\frac{1}{N}\sum_{i=1}^{N} y_i = 0$
- we can recover the optimal solutions for the uncentered data by

$$\hat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_j \hat{\beta}_j$$

where $\bar{y}$ and $\{\bar{x}_j\}_{j=1}^{p}$ are the original mean from the data

Standardized lasso formulation

$$\underset{\beta}{\text{minimize}} \quad \frac{1}{2N}\|y - X\beta\|_2^2 + \gamma\|\beta\|_1, \quad y \in \mathbf{R}^N, \beta \in \mathbf{R}^p$$

the factor N makes γ values comparable for different sample sizes

library packages for solving lasso problems:

- `lasso` in MATLAB: using ADMM algorithm
- `glmnet` with `lasso` option in R: using cyclic coordinate descent algorithm
- `scikit-learn` with `linear_model` in Python: using coordinate descent algorithm

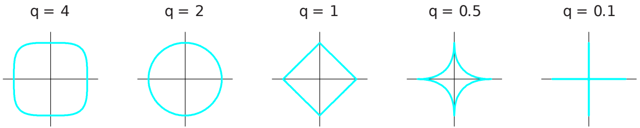
all above algorithms use the soft-thresholding operator

Generalizations of ℓ_1-regularized problems

ℓ_q regularization

for a fixed real number $q \geq 0$, consider

$$\underset{\beta}{\text{minimize}} \quad \frac{1}{2N}\|y - X\beta\|_2^2 + \gamma \sum_{j=1}^{p} |\beta_j|^q$$



- lasso for $q = 1$ and ridge for $q = 2$
- for $q = 0$, $\sum_{j=1}^{p} |\beta_j|^q$ counts the number of nonzeros in β (called **best subset selection**)
- for $q < 1$, the constraint region is *nonconvex*

Generalizations of ℓ_1-regularization

many variants are proposed for acheiving particular structures in solutions

- ℓ_1 regularization with other cost objectives
- elastic net: for highly correlated variables and lasso doesn't perform well
- group lasso: for acheiving sparsity in group
- fused lasso: for neighboring variables to be similar

Sparse methods

example of ℓ_1 regularization used with other cost objectives

$$\underset{\beta}{\text{minimize}} \quad f(\beta) + \gamma \|\beta\|_1$$

problems are in the form of minimizing some loss function with ℓ_1 penalty

- sparse logistic regression
- sparse Gaussian graphical model (graphical lasso)
- sparse PCA
- sparse SVM
- sparse LDA (linear discriminant analysis)

and many more (see Hastie et. al 2015)

Sparse logistic regression

a logistic model for binary y

$$\log \frac{P(y=1|x)}{P(y=0|x)} = \beta_0 + \beta^T x \quad \Rightarrow \quad P(y=1|x) = \frac{e^{\beta_0+\beta^T x}}{1+e^{\beta_0+\beta^T x}}$$

ℓ_1-regularized logistic regression:

$$\underset{\beta_0,\beta}{\text{maximize}} \quad \sum_{i=1}^{N} \left[y_i(\beta_0 + \beta^T x_i) - \log(1+e^{\beta_0+\beta^T x_i}) \right] - \gamma \sum_{j=1}^{p} |\beta_j|$$

- use the lasso term to shrink some regression coefficients toward zero
- typically, the intercept term β_0 is not penalized
- solved by `lassoglm` in MATLAB or `glmnet` in R

Sparse Gaussian graphical model

a problem of estimating a sparse inverse of covariance matrix of Gaussian variable

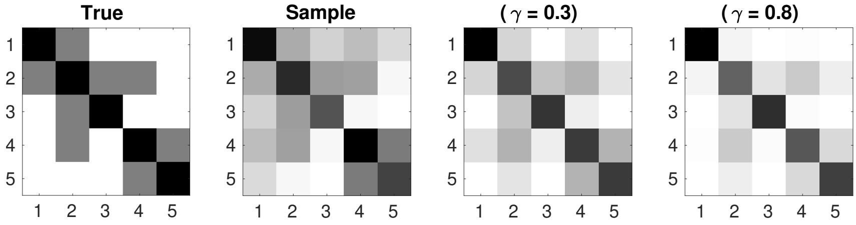
$$\underset{X}{\text{maximize}} \quad \log\det X - \mathbf{tr}(SX) - \gamma \|X\|_1 \qquad \textbf{(graphical lasso)}$$

where $\|X\|_1 = \sum_{ij} |X_{ij}|$

- known fact: if $Y \sim \mathcal{N}(0, \Sigma)$ then the zero pattern of Σ^{-1} gives a conditional independent structure among components of Y
- given samples of random vectors $y_1, y_2, \ldots, y_N$, we aim to estimate a sparse Σ^{-1} and use its sparsity to interpret relationship among variables
- S is the sample covariance matrix, computed from the data
- with a good choice of γ, the solution X gives an estimate of Σ^{-1}
- can be solved by `glasso` in R or `GraphicalLasso` class in Python Scikit-Learn

Example: Gaussian graphical model

5-dimensional Gaussian with sparse Σ^{-1}



- the ground-truth Σ^{-1} has a sparse structure
- it's hard to infer the structure from the sample covariance inverse using $N = 30$
- graphical lasso solutions depend on the penalty parameter
- the higher γ the sparser graph we get

Elastic net

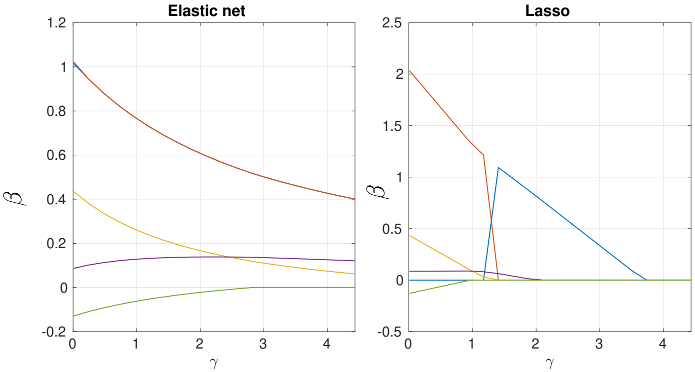
a combination between the ℓ_1 and ℓ_2 regularizations

$$\underset{\beta}{\text{minimize}} \;\; (1/2)\|y - X\beta\|_2^2 + \gamma \left\{ (1/2)(1-\alpha)\|\beta\|_2^2 + \alpha\|\beta\|_1 \right\}$$

where $\alpha \in [0,1]$ and γ are parameters

- when $\alpha = 1$ it's lasso and when $\alpha = 0$ it's a ridge regression
- used when we expect groups of very correlated variables (e.g. microarray, genes)
- strictly convex problem for any $\alpha < 1$ and $\gamma > 0$ (unique solution)

generate $X \in \mathbf{R}^{20\times 5}$ where β_1 and β_2 are highly correlated



- if $x_1 = x_2$, the ridge estimate of β_1 and β_2 will be equal (it can be proved)
- the blue and orange lines correspond to the variables β_1 and β_2
- the lasso does not reflect the relative importance of the two variables
- the elastic net selects the estimates of β_1 and β_2 together

Group lasso

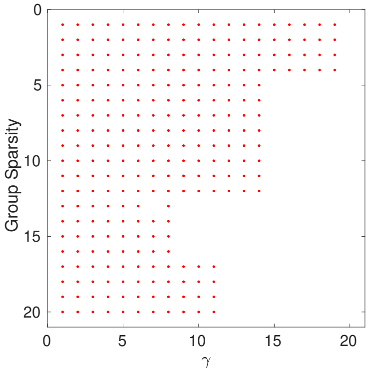
to have all entries in β within a *group* become zero simultaneously

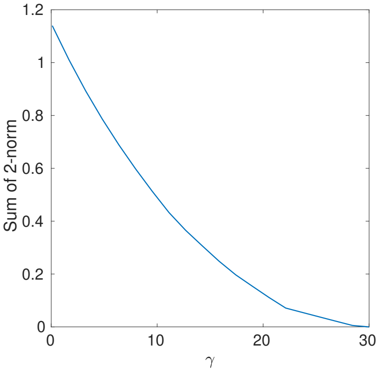
let $\beta = (\beta_1, \beta_2, \ldots, \beta_K)$ where $\beta_j \in \mathbf{R}^p$

$$\text{minimize} \;\; (1/2)\|y - X\beta\|_2^2 + \gamma \sum_{j=1}^{K} \|\beta_j\|_2$$

- the sum of ℓ_2 norm is a generalization of ℓ_1-like penalty
- as γ is large enough, either x_j is entirely zero or all its element is nonzero
- when $p = 1$, group lasso reduces to the lasso
- a nondifferentiable convex problem but can be solved efficiently

generate the problem with $\beta = (\beta_1, \beta_2, \ldots, \beta_5)$ where $\beta_i \in \mathbf{R}^4$





- as γ increases, some of partition β_i becomes entirely zero
- as the sum of 2-norm is zero, the entire vector β is zero

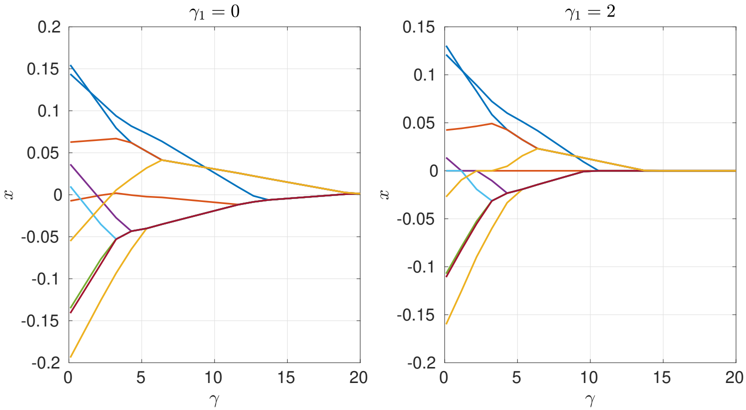
Fused lasso

to have neighboring variables similar and sparse

$$\underset{\beta}{\text{minimize}} \quad (1/2)\|y - X\beta\|_2^2 + \gamma_1 \|\beta\|_1 + \gamma_2 \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|$$

- the ℓ_1 penalty serves to shrink β_i toward zero
- the second penalty is ℓ_1-type encouraging some pairs of consecutive entries to be similar
- also known as **total variation denoising** in signal processing
- γ_1 controls the sparsity of β and γ_2 controls the similarity of neighboring entries
- a nondifferentiable convex problem but can be solved efficiently

generate $X \in \mathbf{R}^{100\times 10}$ and vary γ_2 with two values of γ_1



- as γ_2 increases, consecutive entries of β tend to be equal
- for a higher value of γ_1, some of the entries of β become zero

Sparse PCA

definition: given $Z \in \mathbf{R}^{N \times p}$, PCA finds a unit-norm $x \in \mathbf{R}^p$ such that

$$\mathbf{var}(Zx) = \mathbf{var} \begin{bmatrix} z_1^T x \\ \vdots \\ z_N^T x \end{bmatrix} = \frac{1}{N} \sum_{i=1}^{N} (z_i^T x)^2 = \frac{1}{N} \sum_{i=1}^{N} x^T z_i z_i^T x = x^T \left(\frac{Z^T Z}{N} \right) x$$

is at maximum (assume data in Z is normalized to zero mean)

- x is the right-singular vector of Z (or right eigenvector of $Z^T Z$) w.r.t $\sigma_{\max}(Z)$
- $y = Zx$ is called the **first principal component** of the data Z
- x is called the **principal component loading**
- the r-principal components are $Y = ZX$ where $X_{p \times r}$ is solved from

$$\underset{X}{\text{maximize}} \quad \mathbf{tr}(X^T Z^T Z X) \quad \text{subject to} \quad X^T X = I_r \tag{1}$$

(r columns of X are loadings and mutually orthogonal)

Sparse PCA

- PCA originally was defined as a *sequential procedure* to find r components; however, the optimization explains that the loadings vector in X *maximize* the total variance among all such collections
- each column of Y is a linear combination of data, $y_i = Zx_i$ where loading x_i gives the weight of such combination
- the problem (1) is non-convex due to the objective function and the quadratic constraint

SDP formulation of sparse PCA

let us call $\Sigma = (1/N)Z^TZ$ a sample covariance matrix and consider

$$\underset{x}{\text{maximize}} \quad x^T\Sigma x \quad \text{subject to } \|x\|_2 = 1, \quad \|x\|_0 \leq k \tag{2}$$

we look for the first principal loading that is promoted to be sparse

convex relaxation: define $X = xx^T$ [d'Aspremont et al 2007]

$$\underset{X}{\text{maximize}} \quad \mathbf{tr}(\Sigma X) \quad \text{subject to} \quad \mathbf{tr}(X) = 1, \quad \mathbf{1}^T|X|\mathbf{1} \leq k, \quad X \succeq 0$$

- $\mathbf{tr}(X) = 1$ is from the unit-norm constraint
- $\mathbf{1}^T|X|\mathbf{1} \leq k$ is a weaker convex constraint for the cardinality constraint
- $X \succeq 0$ is enforced due to the form of $X = xx^T$ which is psdf
- we have dropped the rank-1 constraint of X (making the problem a relaxation)

Sparse SVM

soft-margin SVM versus sparse SVM [Ghaoui 2014]

$$\begin{array}{lll} \text{minimize}_{w,b,z} & (1/2)\|w\|_2^2 + \lambda \mathbf{1}^T z \\ \text{subject to} & z \succeq 0 \\ & y_i(x_i^T w + b) \geq 1 - z_i, \end{array} \qquad \begin{array}{lll} \text{minimize}_{w,b,z} & \lambda \|w\|_1 + \frac{1}{N}\mathbf{1}^T z \\ \text{subject to} & z \succeq 0 \\ & y_i(x_i^T w + b) \geq 1 - z_i, \end{array}$$

for $i = 1, \ldots, N$

another common formulation of sparse SVM using hinge loss

$$\underset{w,b}{\text{minimize}}\, \lambda\|w\|_1 + \frac{1}{N}\sum_{i=1}^{N} \max(0, 1 - y_i(x_i^T w + b))$$

- use $\|w\|_1$ in the objective (instead of $\|\cdot\|_2$) to encourage a sparsity in w
- for such a sparse w, term $w^T x$ involves only a few entries in x (use less features)
- a soft-margin SVM is a quadratic program; sparse SVM can be cast as an linear program

Another sparse SVM formulation

one of several formulations of sparse SVM was proposed by A.B. Chan et al 2007

idea: use $\mathbf{card}(w) = r \Rightarrow \|w\|_1 \leq \sqrt{r}\|w\|_2$ to add an ℓ_1-norm constraint

$$\begin{array}{ll} \text{minimize} & t + \lambda \mathbf{1}^T z \\ \text{subject to} & y_i(x_i^T w + b) \geq 1 - z_i, \quad i = 1, 2, \ldots, N \\ & z \succeq 0, \\ & \|w\|_2^2 \leq t, \quad \|w\|_1^2 \leq rt \end{array}$$

with variables $w \in \mathbf{R}^n, b \in \mathbf{R}, z \in \mathbf{R}^N, t \in \mathbf{R}$

- we find a hyperplane with a large margin and the normal vector is also sparse
- the problem is QCQP (quadratically constrained quadratic program)

Summary

- ridge regression is used to shrink the coefficient so that it has small norm; making the solution has less variance
- lasso is used to shrink the coefficient toward zero; promoting simplicity in the solution interpretation
- both ℓ_2 and ℓ_1-regularized LS are convex; can be solved efficiently even when p is large

Regularizations from optimization point of views

Sparse estimation

why a problem of the form

$$\underset{x}{\text{minimize}} \quad f(x) := g(x) + \gamma \|x\|_1$$

produces sparse solutions? we will answer this by giving

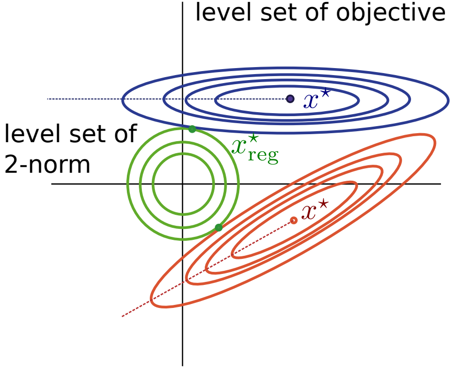
- interpretation of solution shrinkage (both ℓ_1 and ℓ_2)
- the analysis requires a quadratic approximation of g

we will also provide a meaningful connection between early stopping and ℓ_2 penalty

How ℓ_2 penalty affects the optimal solution

setting: minimize $f(x) = g(x) + (\gamma/2)\|x\|_2^2$ (parameter γ is also called weight decay)

- $x^\star$ is a minimizer of g (unpenalized objective)
- $x^\star_{\rm reg}$ is a minimizer of f (regularized objective)



along the dashed line is the direction that Hessian is small; hence, the objective does not increase much

ℓ_2 penalty has a **strong** effect on $x^\star_{\rm reg}$ in the direction of small Hessian (not a preference along this direction to improve objective)

the effect is like pulling $x^\star$ toward zero to explain the effect of ℓ_2 penalty, consider an approximation model

$$\hat{g}(x) = g(x^\star) + \underbrace{\nabla g(x^\star)^T}_{=0}(x - x^\star) + (1/2)(x - x^\star)^T H(x - x^\star)$$

where H (Hessian) can be assumed $\succeq 0$ near $x^\star$ (local minimum of g)

the zero-gradient of regularized objective: $\hat{f}(x) = \hat{g}(x) + (\gamma/2)\|x\|_2^2$ is approximately

$$\nabla f(x) \approx \nabla \hat{f}(x) = H(x - x^\star) + \gamma x = 0$$

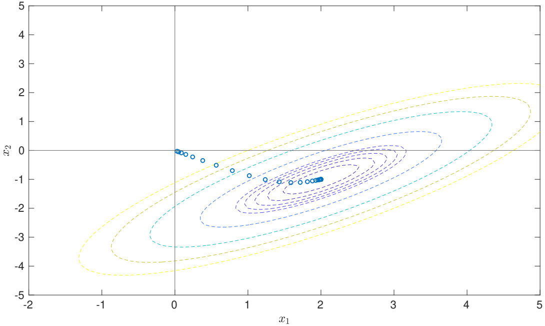
the regularized solution satisfies $x^\star_{\mathrm{reg}} = (H + \gamma I)^{-1} H x^\star$ or

$$x^\star_{\mathrm{reg}} = U(\Lambda + \gamma I)^{-1}\Lambda U^T x^\star, \quad \text{using} \;\; H = U\Lambda U^T$$

- if λ_i is so large that $\lambda_i/(\lambda_i + \gamma) \approx 1$, then the penalty effect on $u_i^T x^\star$ is small
- if $\lambda_i \leq \gamma$ then $\lambda_i/(\lambda_i + \gamma)$ is very small; $u_i^T x^\star$ is shrunk toward zero

Example

minimize $(x-x_c)^T H(x-x_c) + \|x\|_2^2$ with $x_c = (2,-1), H = \begin{bmatrix} 11 & -9 \\ -9 & 11 \end{bmatrix}$



$$U = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \triangleq \begin{bmatrix} u_1 & u_2 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 20 & 0 \\ 0 & 2 \end{bmatrix} \triangleq \mathbf{diag}(\lambda_1, \lambda_2)$$

$$x^\star = (H+\gamma I)^{-1} H x_c$$

$$= u_1 \frac{\lambda_1 (u_1^T x_c)}{\lambda_1 + \gamma} + u_2 \frac{\lambda_2 (u_2^T x_c)}{\lambda_2 + \gamma}$$

- vary $\gamma \in (10^{-4}, 10^3)$ in log-scale and compute $x^\star_{\mathrm{reg}}(\gamma)$ for each γ
- $x^\star_{\mathrm{reg}}(0) = x_c$ and $x^\star_{\mathrm{reg}}(\gamma) \to 0$ as γ increases (the regularizer pulls $x^\star_{\mathrm{reg}}$ toward zero)
- the regularizer has a strong effect on direction u_2 when $\lambda_2 \leq \gamma \leq \lambda_1$
- when $\gamma \geq \lambda_2 \geq \lambda_1$, the regularization affects on both directions

How ℓ_1 penalty affects the optimal solution

setting: minimize $f(x) = g(x) + \gamma \|x\|_1$ for $x \in \mathbf{R}^n$

- $x^\star$ is a minimizer of g (unpenalized objective)
- $x^\star_{\text{reg}}$ is a minimizer of f (regularized objective)
- approximate model: $\hat{g}(x) = g(x^\star) + (1/2)(x - x^\star)^T H (x - x^\star)$
- assume that H is diagonal and $\succeq 0$ (analysis is not simple for a general Hessian)

minimizing $\hat{f}(x) = \hat{g}(x) + \gamma \|x\|_1$ has optimality that zero is one of subgradients

$$0 \in \partial \hat{f}(x) = H(x - x^\star) + \gamma \mathbf{sign}(x) \Rightarrow H_i x - H_i x^\star + \gamma \mathbf{sign}(x_i) = 0$$

(using that $H = \mathbf{diag}(H_1, H_2, \ldots, H_n)$)

- at optimum if $x > 0$ then $x = x^\star - \gamma / H_i$
- at optimum if $x < 0$ then $x = x^\star + \gamma / H_i$

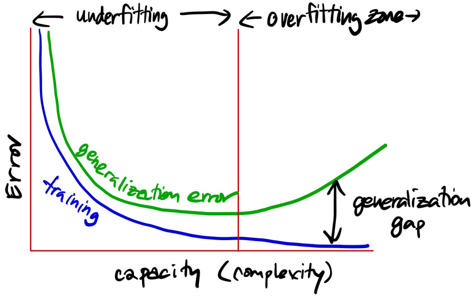
minimizing an **approximated ℓ_1-regularized** function has the analytical solution

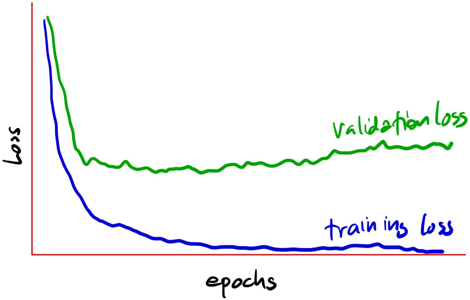
$$x^\star_{\mathrm{reg},i} = \mathbf{sign}(x^\star_i) \cdot \max\left(|x^\star_i| - \frac{\gamma}{H_i}, 0\right), \quad i = 1, 2, \ldots, n$$

- ℓ_1 regularized problem results in **sparse** solution (when γ is large enough)
- when H_i is large, the contribution of g to the regularized objective is overwhelmed in direction i (not preferable to move to that direction) – hence, the regularizer pushes $x^\star_{\mathrm{reg},i}$ to zero
- when $|x^\star_i| > \gamma/H_i$, the regularizer does not move the optimal solution to zero but just shifts it by a distance equal to γ/H_i

Early stopping

the training set loss decreases over time but validation set error may start to rise again





early stopping: return to use solution at the iteration with lowest validation error

- run validation error evaluation periodically during training – either in parallel by separate GPU or using small validation set compared to training set
- store the best solution in a seperate memory from training

Early stopping as a regularizer

early stopping is an unobtrusive form of regularization – no change in training process

- $x^\star$ is a minimizer of $f(x)$
- approximate model: $\hat{f}(x) = f(x^\star) + (1/2)(x - x^\star)^T H (x - x^\star)$ ($H \succeq 0$ at $x^\star$)
- assume to use gradient descent with learning rate ϵ and early stop at iteration τ

the gradient descent step for minimizing $\hat{f}$ is

$$x^+ = x - \epsilon \nabla \hat{f}(x) = x - \epsilon H(x - x^\star) \quad \Rightarrow \quad x^+ - x^\star = (I - \epsilon H)(x - x^\star)$$

use eigenvalue decomposition: $H = U \Lambda U^T$

$$U^T(x^+ - x^\star) = U^T(I - \epsilon U \Lambda U^T)(x - x^\star) = (I - \epsilon \Lambda) U^T (x - x^\star)$$

if $|\lambda(I - \epsilon\Lambda)| \leq 1$ (the matrix is stable), the iterations propragate as

$$U^T(x^{(\tau)} - x^\star) = (I - \epsilon\Lambda)^\tau U^T(x^{(0)} - x^\star)$$

assume that we initialize at $x^{(0)} = 0$ and we return the solution at iteration τ

$$U^T x^{(\tau)} = [I - (I - \epsilon\Lambda)^\tau] \, U^T x^\star$$

now compare with the ℓ_2 regularized solution

$$U^T x^\star_{\mathrm{reg}} = (\Lambda + \gamma I)^{-1} \Lambda U^T x^\star = \left[I - (\Lambda + \gamma I)^{-1}\gamma\right] U^T x^\star$$

(using matrix inversion lemma: $(I + A)^{-1} = I - (I + A)^{-1} A$)

early stopping and ℓ_2 regularization can be seen equivalent if

$$(I - \epsilon\Lambda)^\tau = (\Lambda + \gamma I)^{-1}\gamma$$

which means: τ, ϵ, γ are chosen to the relation above we can use the following facts

- power (and inverse) of a diagonal matrix is diagonal
- $\log(1+x) \approx x$ when x is small (Taylor approximation)

then taking the log transformation of $(I - \epsilon\Lambda)^\tau = (\Lambda + \gamma I)^{-1}\gamma$ gives

$$\tau \log(1 - \epsilon\lambda) = \log(1 + \lambda/\gamma)^{-1} \quad \text{when } \epsilon\lambda \ll 1 \text{ and } \lambda/\gamma \ll 1 \;\Rightarrow\; \tau\epsilon\lambda \approx \frac{\lambda}{\gamma}$$

conclusion: $\tau \approx \frac{1}{\epsilon\gamma}$ or equivalently $\gamma \approx \frac{1}{\tau\epsilon}$

- training iterations plays a role inversely proportional to penalty parameter
- parameter value corresponding to direction of significant curvature (of objective) are regularized less — parameter of that direction tends to learn early
- solving ℓ_2 problem involves finding a good γ – early stopping has an advantage that it determines the right amount of regularization by monitoring validation error instead

References

some figures and examples are taken from

- G.James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*, Springer, 2015
- T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd edition, Springer, 2009
- T. Hastie, R. Tibshirani and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, CRC Press, 2015
- G. Calafiore and L. El Ghaoui, *Optimization Models*, Cambridge University Press, 2014
- A. d'Aspremont, L.El. Ghaoui, M.I. Jordan and G.R.G. Lanckriet, *A Direct Formulation for Sparse PCA using Semidefinite Programing*, SIAM Review, Vol.49, No.3, 2007
- A.B. Chan, N. Vasconcelos, and G.R.G. Lanckriet, *Direct Convex Relaxations of Sparse SVM* Proceedings of the 24th Int. Conf. on Machine Learning (ICML), 2007
- I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, The MIT Press, 2016