Statistical Learning with Sparsity

Generalizations of loss functions and lasso penalties

Boen Jiang

Applied Statistics @ Fudan University

September 16, 2025

Lasso: recap

Least Absolute Shrinkage and Selection Operator (lasso)

Recall that the lasso estimate is defined by

$$\hat{\beta}^{\text{lasso}}(\lambda) = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{2N} \|\boldsymbol{y} - \boldsymbol{X}\beta\|_2^2 + \lambda \|\beta\|_1 \right\},$$

where $\lambda \geq 0$ is a tuning parameter that controls the amount of shrinkage.

For the red part of the objective function, we can generalize the square loss to other loss functions:

- Negative log-likelihood for GLMs
- Negative log-partial-likelihood for Cox models
- Hinge loss for SVM

For the blue part of the objective function, we can generalize the lasso penalty to other penalties:

- Elastic net
- Group lasso
- Fused lasso

Part I

- Negative log-likelihood for GLMs
- Negative log-partial-likelihood for Cox models
- Hinge loss for SVM

Binary response variable

- For simplicity, we can still use the linear model for a binary outcome
- Linear probability model $y_i = x_i^\top \beta + \varepsilon_i, \quad E(\varepsilon_i \mid x_i) = 0$

$$P(y_i = 1 \mid x_i) = E(y_i \mid x_i) = x_i^\top \beta.$$

- Easy interpretation

$$\frac{\partial P(y_i = 1 \mid x_i)}{\partial x_{ij}} = \beta_j$$

However, there are two defects:

- Heteroskedasticity $\quad \mathrm{Var}(y_i \mid x_i) = x_i^\top \beta \left(1 - x_i^\top \beta\right).$
- Not natural for binary outcome because probability is bounded between zero and one.

GLMs

A generalized linear model is made up of a linear predictor

$$\eta_i = \beta_0 + \beta_1 x_{1i} + \ldots + \beta_p x_{pi}$$

and two functions:

- link function that describes how the mean, $\mathbb{E}(Y_i) = \mu_i$, depends on the linear predictor
$$g(\mu_i) = \eta_i$$
- variance function that describes how the variance, $\mathrm{Var}(Y_i)$, depends on the mean
$$\mathrm{Var}(Y_i) = \phi V(\mu)$$
where the dispersion parameter ϕ is a constant

Link functions

We can use a monotone transformation to force the linear predictor to lie within the interval $[0, 1]$:

$$P(y_i = 1|x_i) = g(x_i^\top \beta).$$

Here, the inverse of g is called the **link function**.
There are some canonical choices of g:

- Logit link: $g(t) = \frac{e^t}{1+e^t} = \frac{1}{1+e^{-t}}$, c.f. standard logistic distribution
- Probit link: $g(t) = \Phi(t)$, c.f. standard normal distribution
- Complementary log-log link: $g(t) = 1 - e^{-e^t}$, c.f. standard log-Weibull distribution
- Cauchit link: $g(t) = \frac{1}{\pi}\arctan(t) + \frac{1}{2}$, c.f. standard Cauchy distribution

Modelling Binomial Data

Suppose

$$Y_i \sim \text{Binomial}\left(n_i, p_i\right)$$

and we wish to model the proportions Y_i/n_i. Then

$$\mathbb{E}\left(Y_i/n_i\right) = p_i \quad \text{Var}\left(Y_i/n_i\right) = \frac{1}{n_i} p_i \left(1 - p_i\right)$$

So our variance function is

$$V\left(\mu_i\right) = \mu_i \left(1 - \mu_i\right)$$

Our link function must map from $(0, 1) \to (-\infty, \infty)$. A common choice is

$$g\left(\mu_i\right) = \text{logit}\left(\mu_i\right) = \log\left(\frac{\mu_i}{1 - \mu_i}\right)$$

Modelling Poisson Data

Suppose

$$Y_i \sim \text{ Poisson } (\lambda_i)$$

Then

$$\mathbb{E}(Y_i) = \lambda_i \quad \text{Var}(Y_i) = \lambda_i$$

So our variance function is

$$V(\mu_i) = \mu_i$$

Our link function must map from $(0, \infty) \to (-\infty, \infty)$. A natural choice is

$$g(\mu_i) = \log(\mu_i)$$

One-parameter Canonical Exponential Family

- Canonical exponential family for $k = 1, y \in \mathbb{R}$

$$f_\theta(y) = \exp\left(\frac{y\theta - b(\theta)}{\phi} + c(y, \phi)\right)$$

  for some known functions $b(\cdot)$ and $c(\cdot, \cdot)$.
- If ϕ is known, this is a one-parameter exponential family with θ being the canonical parameter.
- If ϕ is unknown, this may/may not be a two-parameter exponential family. ϕ is called dispersion parameter.
- We assume that ϕ is known.
- The function g that links the mean μ to the canonical parameter θ is called Canonical Link:

$$g(\mu) = \theta$$

Expectation

Note that

$$\ell(\theta) = \frac{Y\theta - b(\theta)}{\phi} + c(Y;\phi),$$

Therefore

$$\frac{\partial \ell}{\partial \theta} = \frac{Y - b'(\theta)}{\phi}$$

It yields

$$0 = \mathbb{E}\left(\frac{\partial \ell}{\partial \theta}\right) = \frac{\mathbb{E}(Y) - b'(\theta)}{\phi}$$

which leads to

$$\mathbb{E}(Y) = \mu = b'(\theta)$$

Variance

On the other hand we have

$$\frac{\partial^2 \ell}{\partial \theta^2} + \left(\frac{\partial \ell}{\partial \theta}\right)^2 = -\frac{b''(\theta)}{\phi} + \left(\frac{Y - b'(\theta)}{\phi}\right)^2$$

and from the previous result,

$$\frac{Y - b'(\theta)}{\phi} = \frac{Y - \mathbb{E}(Y)}{\phi}$$

Together, with the second identity, this yields

$$0 = -\frac{b''(\theta)}{\phi} + \frac{\mathrm{var}(Y)}{\phi^2}$$

which leads to

$$\mathrm{var}(Y) = V(Y) = b''(\theta)\phi$$

Negative log-likelihood

Now we consider minimize negative log-likelihood with a penalty:

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ -\frac{1}{N}\mathcal{L}\left(\beta_0, \beta; \boldsymbol{y}, \boldsymbol{X}\right) + \lambda \|\beta\| \right\}$$

where the type of norm is specified in the problem. We consider the linear model as an example of GLM. Assuming $Y|X = x \sim \mathcal{N}(\mu(x), \sigma^2)$. Then:

$$\mathcal{L}\left(\beta_0, \beta; \boldsymbol{y}, \boldsymbol{X}\right) = -\sum_{i=1}^{N} \frac{(y_i - \beta_0 - \beta x_i)^2}{2\sigma^2} + c = -\frac{\|\boldsymbol{y} - \beta_0 - \beta \boldsymbol{X}\|_2^2}{2\sigma^2 N} + c$$

where c is a constant that does not depend on β_0 and β.
Hence, negative log-likelihood is equivalent to the square error loss in this case.

Remarks on negative log-likelihood

Why? Under regular conditions, the Fisher information matrix is positive definite, so the negative log-likelihood is a convex function of the parameter.

An example for classification

Suppose we take $Y_i \in \{+1, -1\}$, namely,
$P(Y_i = 1) = \pi_i, P(Y_i = -1) = 1 - \pi_i$, and

$$\operatorname{logit}(\pi_i) = \beta_0 + \beta_1^T x_i, i = 1, ..., n.$$

Then the likelihood function is

$$\mathcal{L}(\pi | y_1, ..., y_n) = \prod_{i=1}^{n} \frac{1}{1 + \left(\frac{1-\pi_i}{\pi_i}\right)^{y_i}}.$$

After some trivial algebras, we can get the following negative log-likelihood function with a penalty:

$$\frac{1}{N} \sum_{i=1}^{N} \log\left(1 + e^{-y_i\left(\beta_0 + \beta^\top x_i\right)}\right) + \lambda \|\beta\|_1.$$

Here, $y_i\left(\beta_0 + \beta^\top x_i\right)$ can be interpreted as the **margin** of the *i*-th observation, for which positive values indicate correct classification and negative values indicate incorrect classification.

Case study: document classification

- The 20 Newsgroups data set is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. It was originally collected by Ken Lang, for his **Newsweeder: Learning to filter netnews** paper, though he does not explicitly mention this collection. The 20 newsgroups collection has become a popular data set for experiments in text applications of machine learning techniques, such as **text classification** and **text clustering**.
- Koh, Kim, Boyd (2007) analysis the data set by logistic regression interior point method.
- Freidman, Hastie, Tibshirani (2010) use this data to illustrate their `glmnet` via coordinate descent.
- A team at Renmin U and Tsinghua U (2022) developed a Bayesian method for classification and summerization, their work published on Journal of Machine Learning Research.

Case study: document classification

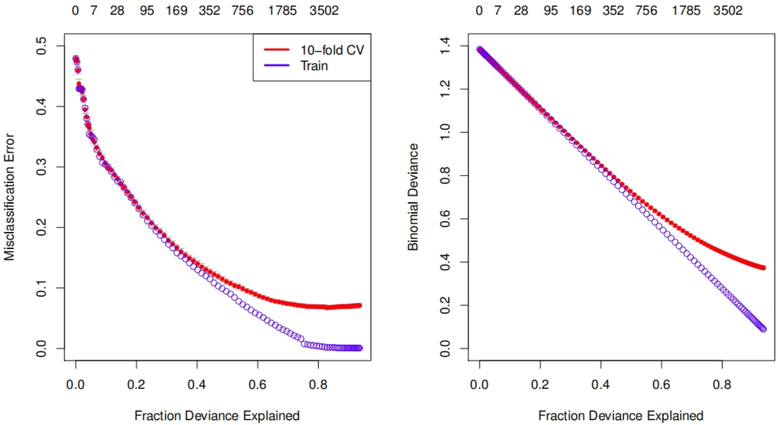
Table 1: Class Groups, Classes and the Numbers of Documents in Each Class

Class Group	Class Number	Class Name	No. Training	No. Test
Computer Science	1	comp.graphics	584	389
	2	comp.os.ms-windows.misc	591	394
	3	comp.sys.ibm.pc.hardware	590	392
	4	comp.sys.mac.hardware	578	385
	5	comp.windows.x	593	395
For Sale	6	misc.forsale	585	390
Auto & Sports	7	rec.autos	594	396
	8	rec.motorcycles	598	398
	9	rec.sport.baseball	597	397
	10	rec.sport.hockey	600	399
Science	11	sci.crypt	595	396
	12	sci.electronics	591	393
	13	sci.med	594	396
	14	sci.space	593	394
Politics	15	talk.politics.guns	546	364
	16	talk.politics.mideast	564	376
	17	talk.politics.misc	465	310
Religion	18	alt.atheism	480	319
	19	soc.religion.christian	599	398
	20	talk.religion.misc	377	251

Case study: document classification

- We have $N = 11314$ documents that we want to classify into two different groups ($Y \in \{-1, +1\}$).
- The features are defined as the set of **trigrams** (with some restrictions). In NLP, trigrams mean a sequence of three adjacent elements from a string of tokens. We have $p = 777811$ features in total.
- Each document contains an average of 425 nonzero features. So this is a sparse problem.
- We want to perform ℓ_1 regularized logistic regression.

Case study: document classification



You can see that overfitting occurs when λ is too small, or equivalently, fraction deviance explained is too large, namely, the model is too "saturated".

Case study: document classification

The **fraction deviance explained** $\left(D^2_\lambda\right)$ is then defined by:

$$D^2_\lambda = \frac{\mathrm{Dev}_{\mathrm{null}} - \mathsf{Dev}_\lambda}{\mathsf{Dev}_{\mathrm{null}}}$$

$$R^2 = \frac{\mathrm{SS}_{\mathrm{tot}} - \mathrm{SS}_{\mathrm{res}}}{SS_{\mathrm{tot}}}$$

Deviance: (Dev_λ) is defined as minus twice the difference in the log-likelihood for a model fit with parameter λ and the "saturated model"(having $\hat{y} = y_i$).

Remarks on GOF statistics

It also reminds me conditional MSE in Guorong Dai's setup:

$$\frac{\mathbb{E}\{(Y - g(U))^2 | S = s\}}{\mathbb{E}\{(Y - d(X))^2 | S = s\}}.$$

So when we comparing two models, a natural choice is to find a proper ratio of the "errors" of two models.

Computational techniques

- With current estimate $\left(\tilde{\beta}_0, \tilde{\beta}\right)$, we form the quadratic function:

$$Q\left(\beta_0, \beta\right) = \frac{1}{2N} \sum_{i=1}^{N} w_i \left(z_i - \beta_0 - \beta^T x_i\right)^2 + C\left(\tilde{\beta}_0, \tilde{\beta}\right),$$

- C denotes a constant independent of $\left(\beta_0, \beta\right)$, z_i and w_i are defined as:

$$z_i = \tilde{\beta}_0 + \tilde{\beta}^T x_i + \frac{y_i - \tilde{p}\left(x_i\right)}{\tilde{p}\left(x_i\right)\left(1 - \tilde{p}\left(x_i\right)\right)}, \quad \text{and} \quad w_i = \tilde{p}\left(x_i\right)\left(1 - \tilde{p}\left(x_i\right)\right)$$

  where $\tilde{p}\left(x_i\right)$ is the current estimate for $\Pr\left(Y = 1 \mid X = x_i\right)$
- Each outer loop then amounts to a weighted lasso regression

Multiple outcomes

Setting

$Y \in \{1, ..., K\}$ for $K > 2$ classes. There are two natural ways for reduction to binary classification in general:

- **OvO (One versus One):** all $\binom{K}{2}$ pairs of classes samples are used to fit $\binom{K}{2}$ binary classifiers, then the predicted class is the one which is predicted the most.
- **OvA (One versus All):** treat all other classes as a single negative class.

Drawbacks

- OvO: computationally exhaustive and cases where same amount of votes for more classes.
- OvA: imbalance amounts positive and negative observations.

Multinomial distribution

- Suppose we have nomial variable: a categorical variable that does not have intrinsic ordering or ranking, e.g., gender, colors, marital status, race, blood types
- Multinomial distribution

$$y \sim \text{ Multinomial}\left(\pi_1, \ldots, \pi_K\right), \quad \sum_{k=1}^{K} \pi_k = 1$$

  y takes values in $\{1, ..., K\}$ with probability $P(y = k) = \pi_k$.
- We want to model y_i based on covariates x_i.

$$y_i \mid x_i \sim \text{ Multinomial } \left[1; \left\{\pi_1\left(x_i\right), \ldots, \pi_K\left(x_i\right)\right\}\right],$$

$$\sum_{k=1}^{K} \pi_k\left(x_i\right) = 1 \text{ for all } x_i.$$

$$\pi_{y_i}\left(x_i\right) = \prod_{k=1}^{K} \left\{\pi_k\left(x_i\right) \text{ if } y_i = k\right\} = \prod_{k=1}^{K} \left\{\pi_k\left(x_i\right)\right\}^{\mathbb{I}(y_i=k)}.$$

Multinomial logistic/softmax regression

- View the category K as the reference level, we model the ratios of the probabilities of category k and K

$$\log \frac{\pi_k(x_i)}{\pi_K(x_i)} = \beta_{0k} + x_i^\top \beta_k \quad (k = 1, \ldots, K-1)$$

$$\pi_k(x_i) = \pi_K(x_i)\, e^{\beta_{0k} + x_i^\top \beta_k}$$

- Denote $\beta_K = 0 \rightsquigarrow$ reference level

$$\begin{aligned}
\sum_{k=1}^{K} \pi_k(x_i) = 1 \Longrightarrow \pi_K(x_i) \sum_{k=1}^{K} e^{\beta_{0k} + x_i^\top \beta_k} &= 1 \\
\Longrightarrow \pi_K(x_i) &= 1 / \sum_{k=1}^{K} e^{\beta_{0k} + x_i^\top \beta_k} \\
\Longrightarrow \pi_k(x_i) &= \frac{e^{\beta_{0k} + x_i^\top \beta_k}}{\sum_{l=1}^{K} e^{\beta_{0l} + x_i^\top \beta_l}}
\end{aligned}$$

Multinomial logistic with penalty

- In traditional multinomial regression, one category must be chosen as a reference group (i.e., have $\boldsymbol{\beta}_k$ set to $\mathbf{0}$) or else the problem is not identifiable $\rightsquigarrow \{\beta_{kj} + c_j\}_{k=1}^K$ and $\{\beta_{kj}\}_{k=1}^K$ produce the same likelihood

Instead of traditional multinomial logistic regression, we can consider the following over specified version:

$$P(Y = k \mid X = x) = \frac{e^{\beta_{0k} + \beta_k^T x}}{\sum_{\ell=1}^K e^{\beta_{0\ell} + \beta_\ell^T x}}.$$

- we regularize the coefficients, and the regularized solutions are not equivariant under base/reference changes,
- the regularization automatically eliminates the redundancy
- The penalty term is $\lambda \sum_{k=1}^K \|\beta_k\|_1$

Multinomial logistic with vectorization

- Log-likelihood form is

$$\log P\left(Y = y_i \mid x_i\right) = \beta_{0,y_i} + \beta_{y_i}^\top x_i - \log\left(\sum_{k=1}^{K} e^{\beta_{0k}+\beta_k^\top x_i}\right).$$

- Let $r_{ik} := \mathbb{I}\left(y_i = k\right), \boldsymbol{R}_{N\times K} := \left(r_{ij}\right)$
  - Such vectors and matrices can be stored efficiently by only storing the nonzero values, and then row and column indices of where they occur $\rightsquigarrow$ Compressed Column Storage (CCS), ...
- We have $\beta_{0,y_i} + \beta_{y_i}^\top x_i = \sum_{k=1}^{K} r_{ik}\left(\beta_{0k} + \beta_k^\top x_i\right)$
- Hence

$$\log P\left(Y = y_i \mid x_i\right) = \sum_{k=1}^{K} r_{ik}\left(\beta_{0k} + \beta_k^\top x_i\right) - \log\left(\sum_{k=1}^{K} e^{\beta_{0k}+\beta_k^\top x_i}\right)$$

Multinomial logistic with penalty

Then we can write the log-likelihood in the more explicit form

$$\frac{1}{N}\sum_{i=1}^{N} w_i \left[\sum_{k=1}^{K} r_{ik}\left(\beta_{0k}+\beta_k^T x_i\right) - \log\left\{\sum_{k=1}^{K} e^{\beta_{0k}+\beta_k^T x_i}\right\}\right]$$

- The weights w_i are used to adjust the contribution of each observation to the likelihood, $w_i = 1$ by default.
- Then for any candidate solution $\{\tilde{\beta}_{kj}\}_{k=1}^{K}$, the criterion to resolve the choice of c_j is the penalty term.

$$c_j = \arg\min_{c\in\mathbb{R}} \left\{\sum_{k=1}^{K} \left|\tilde{\beta}_{kj} - c\right|\right\} = \text{median}\left\{\tilde{\beta}_{1j},\ldots,\tilde{\beta}_{Kj}\right\},$$

for $j = 1, \ldots, p$.

Case study: Handwritten digits

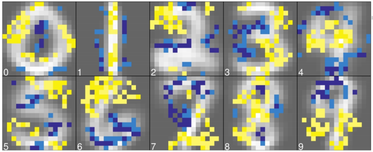
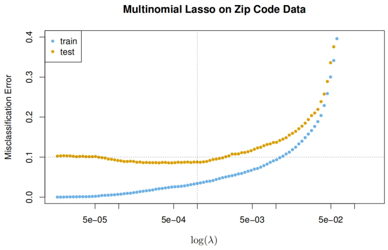


Figure: Source: Trevor Hastie, Robert Tibshirani, and Martin Wainwright. Statistical learning with sparsity: the Lasso and generalizations. CRC Press, 2015, page 38.

- Handwritten Digit Recognition with a Back-Propagation Network is published in 1989.
- LeNet-5: The first part includes two convolutional layers and two pooling layers which are placed alternatively. The second part consists of three fully connected layers.

Case study: Handwritten digits

- We have $N = 7921$ gray-scale images of $p = 256$ pixels representing handwritten digits from 0 yo 9, namely, $Y \in \{0, ..., 9\}$.
- Each one of the p features represents the intensity in a $[0, 1]$-scale of the corresponding pixel (0 black, 1 white).
- We can fit a 10-classes lasso multinomial model.
- Deep networks indeed have better performance.



Computational techniques

- Linear predictor:

$$Z_{ik} = \beta_{0k} + \beta_k^\top x_i, \quad p_{ik} = \Pr\left(Y = k \mid x_i\right) = \frac{e^{Z_{ik}}}{\sum_{m=1}^{K} e^{Z_{im}}}.$$

- Log-likelihood (without penalty):

$$\mathcal{L}(\beta) = \sum_{i=1}^{N}\sum_{k=1}^{K} r_{ik} Z_{ik} - \sum_{i=1}^{N} \log\left(\sum_{m=1}^{K} e^{Z_{im}}\right).$$

- Lasso problem:

$$\min_{\beta} -\frac{1}{N}\mathcal{L}(\beta) + \lambda \sum_{k=1}^{K} \|\beta_k\|_1 .$$

- We use coordinate descent (BCD) algorithm to solve it (profiling)

Computational techniques

- Fix other parameters $\left\{\tilde{\beta}_{0k'}, \tilde{\beta}_{k'}\right\}_{k' \neq k}$, update (β_{0k}, β_k)
- Let $\ell_{ik} = y_{ik} Z_{ik} - \log\left(\sum_{j=1}^{K} e^{Z_{ij}}\right)$ and $p_k(x_i) = \frac{e^{Z_{ik}}}{\sum_{j=1}^{K} e^{Z_{ij}}}$
    - $\frac{\partial \ell_{ik}}{\partial Z_{ik}} = y_{ik} - p_k(x_i)$
    - $\frac{\partial^2 \ell_{ik}}{\partial Z_{ik}^2} = -p_k(x_i)(1 - p_k(x_i)) =: -w_{ik}$
- Heuristically, by Taylor expansion, we can derive a quadratic objective function as

$$Q_k(\beta_{0k}, \beta_k) = -\frac{1}{2N} \sum_{i=1}^{N} w_{ik} \left(h_{ik} - \beta_{0k} - \beta_k^T x_i\right)^2 + C\left(\left\{\tilde{\beta}_{0k}, \tilde{\beta}_k\right\}_{k=1}^{K}\right),$$

where C denotes a constant independent of (β_{0k}, β_k), and

$$h_{ik} = \tilde{\beta}_{0k} + \tilde{\beta}_k^T x_i + \frac{y_{ik} - \tilde{p}_k(x_i)}{\tilde{p}_k(x_i)(1 - \tilde{p}_k(x_i))} \text{ and } w_{ik} = \tilde{p}_k(x_i)(1 - \tilde{p}_k(x_i))$$

Count outcomes

- A random variable Y is Poisson(λ) if its probability mass function is

$$P(Y = k) = e^{-\lambda}\frac{\lambda^k}{k!}, \quad (k = 0, 1, 2, \ldots)$$

- If $Y \sim \text{Poisson}(\lambda)$, then $\mathbb{E}(Y) = \text{Var}(Y) = \lambda$.
- If $Y_1, \ldots, Y_K$ are mutually independent with $Y_k \sim \text{Poisson}\,(\lambda_k)$

$$Y_1 + \cdots + Y_K \sim \text{Poisson}(\lambda),$$

$$(Y_1, \ldots, Y_K) \mid Y_1 + \cdots + Y_K = n \sim \text{Multinomial}\left(n, \left(\frac{\lambda_1}{\lambda}, \ldots, \frac{\lambda_K}{\lambda}\right)\right),$$

where $\lambda = \lambda_1 + \cdots + \lambda_K$.

Some extensions of Poisson distribution

- The Poisson distribution restricts that the mean must be the same as the variance. It cannot capture the feature of over-dispersed data with variance larger than the mean.
- **Negative binomial distribution:** a scale-mixture of Poisson.

$$P(Y = k) = \frac{\Gamma(k+\theta)}{\Gamma(k+1)\Gamma(\theta)} \left(\frac{\theta}{\mu+\theta}\right)^{\theta} \left(\frac{\mu}{\mu+\theta}\right)^{k}, \quad (k = 0, 1, 2, \dots)$$

  with $\mathbb{E}(Y) = \mu$ and $\mathrm{Var}(Y) = \mu + \mu^2/\theta$.
- **Zero-inflated Poisson:** a mixture of Poisson and point mass at zero.

$$P(Y = k) = \begin{cases} p + (1-p)e^{-\lambda}, & \text{if } k = 0 \\ (1-p)e^{-\lambda}\frac{\lambda^k}{k!}, & \text{if } k = 1, 2, \dots \end{cases}$$

$$\mathbb{E}(Y) = (1-p)\lambda \text{ and } \mathrm{Var}(Y) = (1-p)\lambda(1+p\lambda)$$

Poisson regression model / log-linear model

- $$\begin{cases} Y_i \mid X_i & \sim \text{Poisson}\left(\lambda_i\right), \\ \lambda_i & = \lambda\left(X_i, \beta\right) = e^{\beta_0 + X_i^\top \beta}. \end{cases}$$
  $$\mathbb{E}\left(Y_i \mid X_i\right) = \text{var}\left(Y_i \mid X_i\right) = e^{\beta_0 + X_i^\top \beta}.$$
- This model is also called log-linear model
  $$\log \mathbb{E}\left(Y_i \mid X_i\right) = \beta_0 + X_i^\top \beta$$
- Interpretation: conditional log mean ratio
  $$\log \frac{\mathbb{E}\left(Y_i \mid \ldots, X_{ij} + 1, \ldots\right)}{\mathbb{E}\left(Y_i \mid \ldots, X_{ij}, \ldots\right)} = \beta_j$$

Poisson regression with penalty

- Likelihood:

$$L(\beta) = \prod_{i=1}^{n} e^{-\lambda_i} \frac{\lambda_i^{y_i}}{y_i!} \propto \prod_{i=1}^{n} e^{-\lambda_i} \lambda_i^{y_i}$$

$$\log L(\beta) = \sum_{i=1}^{n} (-\lambda_i + y_i \log \lambda_i) = \sum_{i=1}^{n} \left(-e^{\beta_0 + x_i^\top \beta} + y_i(\beta_0 + x_i^\top \beta) \right).$$

- The penalty term is $\lambda \|\beta\|_1$.

$$-\frac{1}{N} \sum_{i=1}^{N} \left\{ y_i \left(\beta_0 + \beta^\top x_i \right) - e^{\beta_0 + \beta^\top x_i} \right\} + \lambda \|\beta\|_1$$

- Take derivatives on β_0 and set it to zero $\rightsquigarrow \frac{1}{N} \sum_{i=1}^{N} e^{\hat{\beta}_0 + \hat{\beta}^\top x_i} = \bar{y}$

Poisson regression for rates modeling

- If observation windows have different lengths T_i, then

$$\mathbb{E}\left[y_i \mid X_i = x_i\right] = T_i \mu\left(x_i\right)$$

  where $\mu\left(x_i\right)$ rate per unit time interval.

- 6 months $\sim$ yearly visit to doctor has $T = 6$.
-

$$\log \mathbb{E}[Y \mid X = x, T] = \underbrace{\log T}_{\text{“offset”}} + \beta_0 + \beta^\top x$$

- The terms $\log T_i$ for each observation require no fitting, and are called an offset.

Case study: distribution smoothing

- N count variables $\{y_k\}_{k=1}^N$ coming from a N-cell multinomial distribution.
- $\boldsymbol{r} = \{r_k\}_{k=1}^N = \left\{y_k / \sum_{k=1}^N y_k\right\}_{k=1}^N$ vector of proportions.
- Issue: $\boldsymbol{r}$ could be sparse. Want to regularize it toward a more stable distribution $\boldsymbol{u} = \{u_k\}_{k=1}^N$.
-

$$\underset{\boldsymbol{q}\in\mathbb{R}^N, q_k \geq 0}{\text{minimize}} \quad \underbrace{\sum_{k=1}^N q_k \log\left(\frac{q_k}{u_k}\right)}_{\text{Kullback-Leibler divergence}} \quad \text{such that } \|\boldsymbol{q} - \boldsymbol{r}\|_\infty \leq \delta, \sum_{k=1}^N q_k = 1$$

- We want a distribution $\boldsymbol{q}$ which is approximately equal to our observed proportions but at the same time as close as possible to a nominal distribution $\boldsymbol{u}$.

Case study: distribution smoothing

Why this problems falls in Poisson model framework?

- The previous minimization problem $\underset{\boldsymbol{q}\in\mathbb{R}^N, q_k\geq 0}{\text{minimize}} \sum_{k=1}^N q_k \log\left(\frac{q_k}{u_k}\right)$ such that $\|\boldsymbol{q}-\boldsymbol{r}\|_\infty \leq \delta, \sum_{k=1}^N q_k = 1$ has Lagrange dual

$$\underset{\beta_0,\boldsymbol{\alpha}}{\text{maximize}} \left\{ \sum_{k=1}^N r_k \left[\log u_k + \beta_0 + \alpha_k - u_k e^{\beta_0+\alpha_k}\right] - \delta\|\boldsymbol{\alpha}\|_1 \right\}$$

- This is equivalent to fitting a Poisson model with offset $\log u_k$, individual parameter α_k and design matrix $X = \mathbb{I}_{N\times N}$.

Case study: distribution smoothing

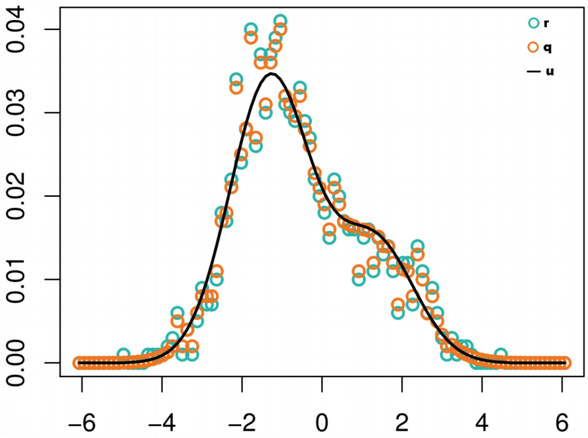


Figure: Source: Trevor Hastie, Robert Tibshirani, and Martin Wainwright. Statistical learning with sparsity

Time-to-event data

- Survival analysis in biostatistics
  - Outcome denotes the Survival time or the time to the recurrent of the disease
- Duration analysis in econometrics
  - Outcome denotes the weeks unemployed or days until the next arrest after being released from incarceration
- Time-to-event-data
  - Non-negative
  - May be censored, resulting in inadequate tail information

Survival function

- Medical studies interested in time to death T of sick patients, usually characterized by the survival function $S(t) := \mathbb{P}(T > t)$, the probability of surviving beyond a certain time t.
- Some patients drop out the study or die because of unrelated causes: we call this situation a censoring time C.
- $Y := \min(C, T)$ is the observed outcome variable, together with an indicator $\delta := \mathbb{I}\{Y = T\}$ of whether the patient died correctly (because of the studied illness).

Hazard function and Cox model

- **Hazard function**: Instantaneous probability of death at time t, given survival up till t.

$$h(t) = \lim_{\delta \to 0} \frac{\mathbb{P}(Y \in \{t, t+\delta\} \mid Y \geq t)}{\delta} = \frac{f(t)}{S(t)}$$

  where $f(t)$ density of T.
- **Cox's model** treats special cases of hazard functions:

$$h(t|x) = h_0(t) e^{\beta^\top x}$$

  where x represents e.g. gene expressions and $h_0(t)$ is **baseline hazard**: hazard for one individual with $x = 0 \rightsquigarrow$ semiparametric
- The coefficient β represents the multiplicative effect of the covariates on the hazard

$$\frac{h(t|\cdots, x_j + 1, \cdots)}{h(t|\cdots, x_j, \cdots)} = e^{\beta_j} \rightsquigarrow \text{hazard ratio (HR)}$$

The hazard of hazard ratio (Hernán, 2010)

- Treatment Z_i and time-to-event outcome Y_i.
- Hazard: the event rate at time t conditional on survival until time t or later
$$\lim_{\Delta t \to 0} \frac{P(t \leq Y < t + \Delta t | Y \geq t)}{\Delta t}.$$
- Survival analysis assumes models for hazard, e.g., Cox models, additive hazard models, etc.
- Hazard ratio between the treatment and control compares
$$\lim_{\Delta t \to 0} \frac{P(t \leq Y(1) < t + \Delta t | Y(1) \geq t)}{\Delta t}, \quad \lim_{\Delta t \to 0} \frac{P(t \leq Y(0) < t + \Delta t | Y(0) \geq t)}{\Delta t}$$
which compares the populations $\{i : Y_i(1) \geq t\}$ and $\{i : Y_i(0) \geq t\}$.
- Hazard ratio has a built-in selection bias.

Risk sets and partial likelihood

- Go back to the Cox model.

$$h(t|x) = h_0(t)e^{\beta^\top x}$$

- Denote by $R_i := \{j \mid y_j \geq y_i\}$ the risk set of subject i (individuals which are still in the study when subject i dies).
- Cox (1972; 1975) proposed the partial likelihood. He argued that only a part of the likelihood is useful when we are interested in the parameter β only
- The partial likelihood of subject i is given by

$$\frac{h(y_i|x_i)}{\sum_{j\in R_i} h(y_j|x_j)} = \frac{e^{\beta^\top x_i}}{\sum_{j\in R_i} e^{\beta^\top x_j}}$$

Note that baseline hazard h_0 has no effect here, does not depend on actual death times

Interpretation of Partial Likelihood

- Let the "event" = "die" for convenience
- The probability that a subject with covariate value $X_{(j)}$ dies at time t_j, given that one and only one subject in the risk set R_j dies at time t_j, is

$$
\begin{aligned}
& P(\text{subject with } X_{(j)} \text{ dies at time } t_j | \text{dies in } R_j \text{ but don't know who}) \\
= & \frac{P(\text{subject with } X_{(j)} \text{ dies at time } t_j \mid \text{survives to time } t_j)}{P(\text{dies at time } t_j \text{ and know in the risk set } R_j \mid \text{in the set } R_j \text{ survive to time } t_j)} \\
= & \frac{h(t_j|X_{(j)})}{\sum_{i \in R_j} h(t_j|X_i)} \\
= & \frac{e^{\beta^\top X_{(j)}}}{\sum_{i \in R_j} e^{\beta^\top X_i}}.
\end{aligned}
$$

- We sequentially consider in time order $t_1, t_2, \dots, t_k$ and then we obtain the likelihood $L_p(\beta)$

Cox model with penalty

The log-partial-Likelihood is

$$\ell_p(\beta; \boldsymbol{x}, \boldsymbol{\delta}) = \underbrace{\sum_{\delta_i=1}}_{\text{died "correctly"}} \log\left[\frac{e^{\beta^\top x_i}}{\sum_{j\in R_i} e^{\beta^\top x_j}}\right]$$

with corresponding ℓ_1-penalized CPH problem:

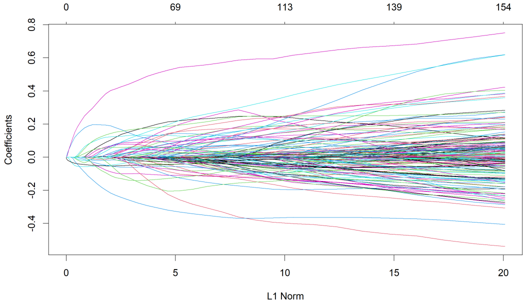
$$\underset{\beta}{\text{minimize}} \left\{ -\sum_{\delta_i=1} \log\left[\frac{e^{\beta^\top x_i}}{\sum_{j\in R_i} e^{\beta^\top x_j}}\right] + \lambda\|\beta\|_1 \right\}$$

Case study: lymphoma data

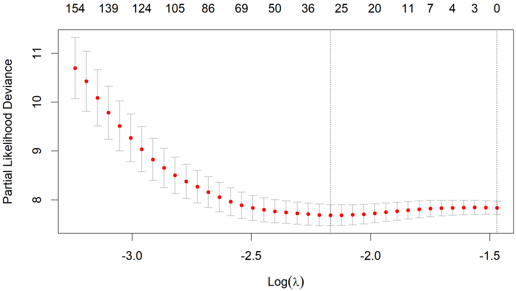
- We want to estimate the hazard function S for $N = 240$ Lymphoma patients with $p = 7399$ variables measuring gene expressions. 102 of these samples are right censored, i.e.

$$Y = \min(T, C) = C$$

- They use the ℓ_1-penalized CPH problem to find $\hat{\beta}(\lambda_{\min})$.



Case study: lymphoma data



- Simon, Friedman, Hastie and Tibshirani (2011) give details for an algorithm based on coordinate-descent
- Implemented in the R package `glmnet`

Kaplan-Meier estimator

We use the Kaplan-Meier estimator of survivor function $S(t)$: let $\hat{\eta}(x) := \hat{\beta}\left(\lambda_{\min}\right)^{\top} x$, then

$$\widehat{S}(t) = \prod_{i: y_i \leq t} \left(1 - \frac{e^{\hat{\eta}(x_i)}}{\sum_{j \in R_i} e^{\hat{\eta}\left(x_j\right)}}\right)$$

is an estimate of $S(t)$. We use these in the following plot.
In computing the score $\hat{\eta}\left(x_i\right)^{(k)}$ for the observations in fold k, we use the coefficient vector $\widehat{\beta}^{(-k)}$ computed with those observations omitted. Doing this for all K folds, we obtain the "pre-validated" dataset $\left\{\left(\hat{\eta}\left(x_i\right)^{(k)}, y_i, \delta_i\right)\right\}_{i=1}^{N}$.

Pre-validation

- The Cox model is a semi-parametric model, and the proportional hazard assumption is crucial.
- The proportional hazard assumption is not always satisfied, and the Cox model may not be the best choice.
- Pre-validation is a method to check the proportional hazard assumption.
- The idea is to split the data into two parts,
    - one part $\rightsquigarrow x_i$
    - the other part $\rightsquigarrow \hat{\beta}^{(k)}$
    - repeat for $k = 1, ..., K$ folds
    - obtain the pre-validated dataset $\{(\hat{\eta}(x_i)^{(k_i)}, y_i, \delta_i)\}_{i=1}^{N}$
- If the proportional hazard assumption holds, the estimated coefficients should be similar.

Case study: lymphoma data

- Log-rank test is used to formally test whether the survival curves are statistically different

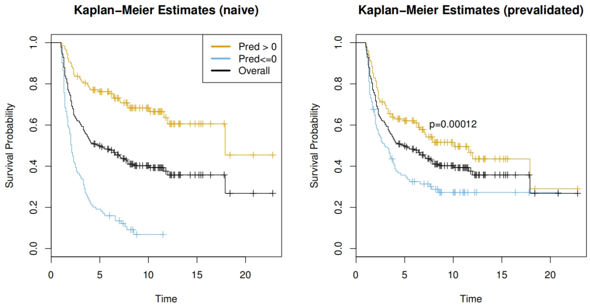
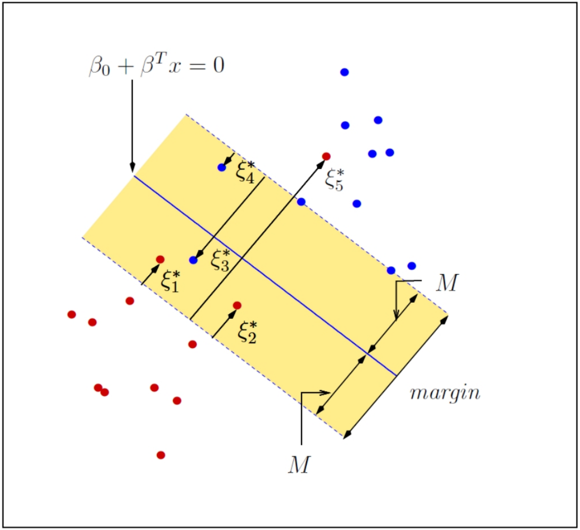


Figure 3.7 *The black curves are the Kaplan–Meier estimates of $S(t)$ for the Lymphoma data. In the left plot, we segment the data based on the predictions from the Cox proportional hazards lasso model, selected by cross-validation. Although the tuning parameter is chosen by cross-validation, the predictions are based on the full training set, and are overly optimistic. The right panel uses prevalidation to build a prediction on the entire dataset, with this training-set bias removed. Although the separation is not as strong, it is still significant. The spikes indicate censoring times. The p-value in the right panel comes from the log-rank test.*

Support vector machine



SVM: A toy model

- Suppose we have a classification rule : $\{x : f(x) = \beta_0 + x^T\beta\}$, here $x = (x_1, x_2)^T \in \mathbb{R}^2$.
- Consider the following lines:

$$\begin{aligned} l_1 &: \beta_0 + \beta_1 x_1 + \beta_2 x_2 = 1, \\ l_{-1} &: \beta_0 + \beta_1 x_1 + \beta_2 x_2 = -1 \\ l_0 &: \beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0. \end{aligned}$$

- Then, by geometry,

$$d(l_0, l_1) = d(l_0, l_{-1}) = \frac{1}{\sqrt{\beta_1^2 + \beta_2^2}} = \frac{1}{\|\beta\|_2}.$$

- So if the dataset is linear separable, by setting the closest point x_i to the classification boundary l_0 as $f(x_i) = 0$, we have $\exists \beta_0, \beta$ such that

$$f(x_i) = \begin{cases} > 0, & \text{if } y_i = +1, \\ < 0, & \text{if } y_i = -1. \end{cases}$$

SVM geometric interpretation

Consider the Boundary $B = \{x \in \mathbb{R}^p \mid f(x) = 0\}$, where

$$f(x) = \beta_0 + \beta^\top x$$

Then the distance between the boundary and the point x_0 is

$$\operatorname{dist}(x_0, B) = \inf_{z \in B} \|z - x_0\|_2 = \frac{|f(x_0)|}{\|\beta\|_2}$$

So we find that the optimal separating plane $f^*(x) = 0$ has margin

$$M_2^* = \max_{\beta_0, \beta} \left\{ \min_{i \in \{1, \ldots, n\}} \frac{y_i f(x_i, \beta_0, \beta)}{\|\beta\|_2} \right\}$$

SVM with slack variable ξ_i

- We allow some points to be misclassified, and introduce a slack variable $\xi = (\xi_1, ..., \xi_n), \xi_i \geq 0$ for each observation.
- $y_i(\beta_0 + x_i^T\beta)$ represents the "distance" of x_i to the classification boundary $\rightsquigarrow$ a linear classifier is scale invariant, the solution coefficients can be rescaled, WLOG, we can set $\|\beta\|_2 = 1$
- Foundamentally, we want $y_i(\beta_0 + x_i^T\beta) \geq M$,
- But we allow some points to be misclassified, so we relax the constraint to $y_i(\beta_0 + x_i^T\beta) \geq M(1 - \xi_i)$.
- $\sum_{i=1}^N \xi_i \leq C$ will introduce a bias-variance trade-off:
  - $C = 0$: hard margin SVM, no misclassification allowed,
  - C large: wide margin, introduce large bias and low variance in classification,
  - C small: narrow margin, introduce small bias and high variance in classification.

Optimization problem of SVM

- Denote M as half width of the yellow part in the illustrator as the **margin** of the classifier.
- Objective:

$$\max_{\beta_0,\beta,\{\xi_i\}_{i=1}^N} \quad M$$

- Constraints: $y_i \underbrace{\left(\beta_0 + \beta^\top x_i\right)}_{f(x_i,\beta_0,\beta)} \geq M\left(1-\xi_i\right), \forall i$, where

  $\xi_i \geq 0, \forall i, \sum_{i=1}^N \xi_i \leq C, \|\beta\|_2 = 1$
- Note that by the constraints, we have

$$\xi_i \geq 1 - y_i f\left(x_i, \beta_0, \beta\right), \quad \xi_i \geq 0.$$

  so $\sum_{i=1}^N \xi_i \geq \sum_{i=1}^N [1 - y_i(\beta_0 + \beta^T x_i)]_+$.

Optimization problem of SVM

- By writing the Lagrangian equivalent of the original minimization problem (SVM), we get:

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left[1 - y_i f\left(x;\beta_0,\beta\right)\right]_+ + \lambda \|\beta\|_2^2 \right\}$$

  Decreasing λ corresponds to decreasing C and $f(x_i;\beta_0,\beta) = \beta_0 + \beta^T x_i$.
- Hinge loss $\left[1 - y_i f\left(x;\beta_0,\beta\right)\right]_+$ is zero when if x_i lies on the correct side of the margin. For data on the wrong side of the margin, the function's value is proportional to the distance from the margin.
- ℓ_1 penalized version:

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left[1 - y_i f\left(x_i;\beta_0,\beta\right)\right]_+ + \lambda \|\beta\|_1 \right\}.$$

RKHS and kernel trick

Kernel

Let $\mathcal{X}$ be a non-empty set. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a kernel if there exists an $\mathbb{R}$-Hilbert space and a map $\phi : \mathcal{X} \to \mathcal{H}$ such that $\forall x, x' \in \mathcal{X}$,

$$k\left(x, x'\right) := \langle \phi(x), \phi\left(x'\right) \rangle_{\mathcal{H}}.$$

RKHS

Let $\mathcal{H}$ be a Hilbert space of $\mathbb{R}$-valued functions defined on a non-empty set $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a reproducing kernel of $\mathcal{H}$, and $\mathcal{H}$ is a reproducing kernel Hilbert space, if k satisfies

- $\forall x \in \mathcal{X}, \quad k(\cdot, x) \in \mathcal{H}$,
- $\forall x \in \mathcal{X}, \forall f \in \mathcal{H}, \langle f, k(\cdot, x) \rangle_{\mathcal{H}} = f(x)$ (the reproducing property).

Examples of kernels

- If $K(x,y) = (\langle x,y\rangle + c)^2 = (x_1y_1 + x_2y_2 + c)^2 = \langle\Phi(x),\Phi(y)\rangle$, then $\Phi(x) = (x_1, x_2, \sqrt{2c}x_1, \sqrt{2c}x_2, c)^T$.
- There are many other kernels,
  - Polynomial kernel: $K(x,y) = (\langle x,y\rangle + c)^d$,
  - Gaussian kernel: $K(x,y) = \exp(-\|x-y\|^2/2\sigma^2)$,
  - Laplacian kernel: $K(x,y) = \exp(-\|x-y\|/\sigma)$,
  - Sigmoid kernel: $K(x,y) = \tanh(\kappa\langle x,y\rangle + \theta)$.
- The linear SVM can be generalized using a kernel to create nonlinear boundaries.
- $\|\beta\|_2 \rightsquigarrow \|\beta\|_{\mathcal{H}_K}$
- We'll revisit this variation in Group Lasso

SVM vs logistic regression

- Penalized logistic:

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \underbrace{\log\left(1 + e^{-y_i f(x_i,\beta_0,\beta)}\right)}_{\text{logistic loss}} + \lambda\|\beta\| \right\}$$

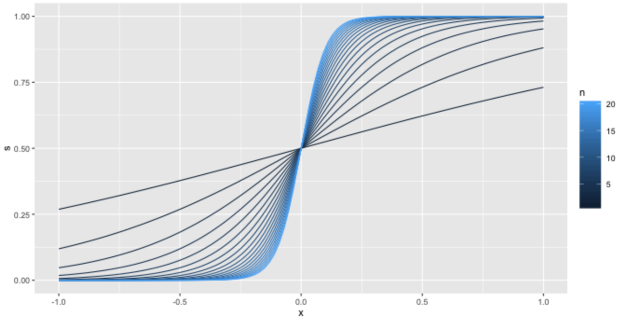
- Penalized svm:

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \underbrace{\left[1 - y_i f\left(x;\beta_0,\beta\right)\right]_+}_{\text{hinge loss}} + \lambda\|\beta\| \right\}$$

- Data is separable: there exists a hyperplane that separates the two cases. In this cases logistic regression has a problem:

$$P(Y = 1 \mid X = x) = \frac{e^{\beta_0 + \beta^\top x}}{1 + e^{\beta_0 + \beta^\top x}}$$

Problem of logistic regression



Problem: When $p \gg N$, the points are almost always separable.

Relationship between SVM and logistic regression

Consider the problem

$$\underset{\beta_0,\beta}{\text{minimize}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \log\left(1 + e^{-y_i f(x_i,\beta_0,\beta)}\right) + \lambda \|\beta\|_2^2 \right\}$$

Let $\left(\tilde{\beta}_0(\lambda), \tilde{\beta}(\lambda)\right)$ be the solution, then Rosset et al. (2004) showed that

$$M_2^* = \lim_{\lambda \to 0} \left\{ \min_{i \in \{1,\ldots,N\}} \frac{y_i f\left(x_i, \tilde{\beta}_0(\lambda), \tilde{\beta}(\lambda)\right)}{\|\tilde{\beta}(\lambda)\|_2} \right\}$$

So for $\lambda \to 0$ we have that the ℓ_2-regularized logistic regression corresponds to the SVM solution.

Relationship between SVM and logistic regression

In particular, if $\left(\breve{\beta}_0, \breve{\beta}\right)$ solve the SVM problem for $C = 0$, then we have that:

$$\lim_{\lambda \to 0} \frac{\tilde{\beta}(\lambda)}{\|\tilde{\beta}(\lambda)\|_2} = \breve{\beta}$$

Note that the division by the ℓ_2 norm of $\tilde{\beta}(\lambda)$ makes sure that the solution on the SVM problem does not blow up.

- As $\lambda \to 0$, logistic regression and SVM solutions coincide
- SVM leads to a more stable numerical method for computing the solution in the solution is most dense
- Logistic regression is more useful in the sparser part of the solution path

Part II

- Elastic net
- Group lasso/overlap group lasso
- Fused lasso

Case study: comparison of lasso and elastic net on highly correlated variables

In microarray studies, **groups of genes in the same biological pathway** tend to be expressed (or not) together, and hence measures of their expression tend to be strongly correlated.

Simulation setup

1. 2 sets of 3 variables, pairwise correlations around 0.97 in each group
2. sample size: $N = 100$,
3. data are simulated as follows:

$$
\begin{aligned}
&Z_1, Z_2 \sim N(0,1) \quad \text{independent} \\
&Y = 3Z_1 - 1.5Z_2 + 2\epsilon, \quad \epsilon \sim N(0,1) \\
&X_j(j = 1,2,3) = Z_1 + \xi/5, \quad \xi_j \sim N(0,1) \\
&X_j(j = 4,5,6) = Z_2 + \xi/5, \quad \xi_j \sim N(0,1)
\end{aligned}
$$

Case study: comparison of lasso and elastic net on highly correlated variables

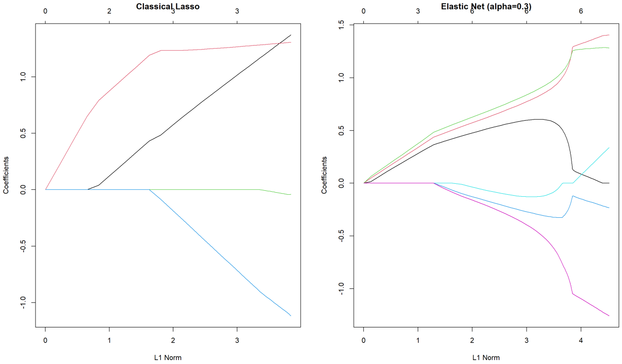
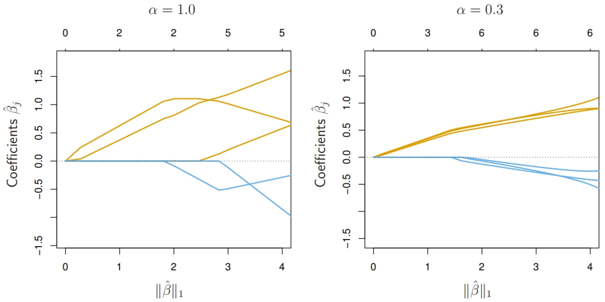


Figure: The lasso estimates ($\alpha = 1$), as shown in the left panel, exhibit somewhat erratic behavior as the regularization parameter λ is varied. In the right panel, the elastic net with ($\alpha = 0.3$) includes all the variables, and the correlated groups are pulled together.

Case study: comparison of lasso and elastic net on highly correlated variables



- lasso estimates exhibit erratic behavior as λ varies: one variable is excluded and the correlations among variables are not clear
- elastic net includes all variables and correlated groups are pulled together, sharing values approximately equally.
- the difference between my plot and the book's plot is due to the random seed.

Elastic net

Recap, the elastic net problem is defined as

$$\underset{(\beta_0,\beta)\in\mathbb{R}\times\mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \beta_0 - x_i^T\beta\right)^2 + \lambda\left[\frac{1}{2}(1-\alpha)\|\beta\|_2^2 + \alpha\|\beta\|_1\right] \right\}$$

- Denote R as the objective function of elastic net, then for $\beta_j > 0$,

$$\begin{aligned}
\frac{\partial R}{\partial \beta_j} &= \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \beta_0 - \boldsymbol{x}_{\boldsymbol{j}}^\top\boldsymbol{\beta}\right)(-x_{ij}) + \lambda\left[(1-\alpha)\beta_j + \alpha\,\text{sgn}\left(\beta_j\right)\right] \\
&= \frac{1}{N}\sum_{i=1}^{N}(\underbrace{y_i - \beta_0 - \sum_{k\neq j} x_{ik}\beta_k}_{=r_{ij}} - x_{ij\beta_j})(-x_{ij}) + \ldots \\
&= -\frac{1}{N}\sum_{i=1}^{N} r_{ij}x_{ij} + \frac{1}{N}\sum_{i=1}^{N} x_{ij}^2\beta_j + \lambda\left[(1-\alpha)\beta_j + \alpha\,\text{sgn}\left(\beta_j\right)\right].
\end{aligned}$$

Elastic net

- Setting the derivatives to zero yields

$$\left[\frac{1}{N}\sum_{i=1}^{N} x_{ij}^2 + \lambda(1-\alpha)\right]\beta_j + \alpha\lambda\,\mathrm{sgn}(\beta) = \frac{1}{N}\sum_{i=1}^{N} r_{ij}x_{ij}$$

  As we did in lasso case, here coordinate descent have a close form update for each β_j.

$$\widehat{\beta}_j = \frac{\mathcal{S}_{\lambda\alpha}\left(\sum_{i=1}^{N} r_{ij}x_{ij}\right)}{\sum_{i=1}^{N} x_{ij}^2 + \lambda(1-\alpha)},$$

  where $r_{ij} = y_i - \tilde{\beta}_0 - \sum_{k\neq j} x_{ik}\hat{\beta}_k$ and $\mathcal{S}_\mu(z) := \mathrm{sign}(z)(z-\mu)_+$.
- In practice, group structure may not be as evident as the previous ‘ideal’ model, this example does capture the main idea of elastic net.
- By adding ridge penalty to lasso penalty, elastic net automatically controls for strong within-group correlations.

Elastic net

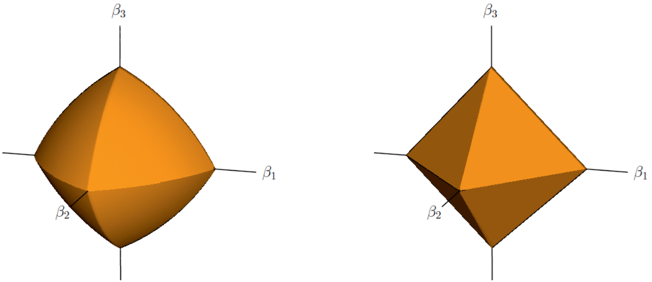


Figure 4.2 *The elastic-net ball with* $\alpha = 0.7$ *(left panel) in* $\mathbb{R}^3$*, compared to the* ℓ_1 *ball (right panel). The curved contours encourage strongly correlated variables to share coefficients (see* ~~*Exercise 4.2 for details)*~~ the following pages for details

Why does elastic net promote grouping?

Grouping effect (Zou and Hastie, 2005)

Given data $(\boldsymbol{y}, \boldsymbol{X})$ and penalty parameters (λ_1, λ_2), the response $\boldsymbol{y}$ is centered and the predictor $\boldsymbol{X}$ are standardized. Let $\hat{\beta}(\lambda_1, \lambda_2)$ be the naive elastic net estimate. Suppose that $\hat{\beta}_i(\lambda_1, \lambda_2)\hat{\beta}_j(\lambda_1, \lambda_2) > 0$. Then

$$D_{\lambda_1,\lambda_2}(i,j) = \frac{1}{\|\boldsymbol{y}\|_2} \left| \hat{\beta}_i(\lambda_1, \lambda_2) - \hat{\beta}_j(\lambda_1, \lambda_2) \right| \leq \frac{1}{\lambda_2} \sqrt{2(1-\rho)}$$

where $\rho = \boldsymbol{x}_i^T \boldsymbol{x}_j$, the sample correlation.

The unitless quantity $D_{\lambda_1,\lambda_2}(i,j)$ describes the difference between the coefficient paths of predictors i and j. If $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are highly correlated, i.e. $\rho \approx 1$ (if $\rho \approx -1$ then consider $-\boldsymbol{x}_j$), grouping effect says that the difference between the coefficient paths of predictor i and predictor j is almost 0

Grouping effect

Consider the overall minimization

$$\min_{\beta} \underbrace{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{\beta}\|^2}_{L(\boldsymbol{\beta})}.$$

Then the optimality at β_i and β_j is

$$\begin{cases} \frac{\partial L}{\partial \beta_i} = -2\boldsymbol{x_i}^T\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\right) + 2\lambda_2\beta_i + \lambda_1 \operatorname{sgn}\left(\beta_i\right) = 0 \\ \frac{\partial L}{\partial \beta_j} = -2\boldsymbol{x_j}^T\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\right) + 2\lambda_2\beta_j + \lambda_1 \operatorname{sgn}\left(\beta_j\right) = 0 \end{cases}$$

Substracting the two equations, we have

$$2\left(\boldsymbol{x_j} - \boldsymbol{x_i}\right)^\top\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\right) + 2\lambda_2\left(\beta_i - \beta_j\right) + \lambda_1\left(\underbrace{\operatorname{sgn}\left(\beta_i\right) - \operatorname{sgn}\left(\beta_j\right)}_{=0,\text{ by assumption}}\right) = 0$$

Grouping effect (cont'd)

By the previous equation, we have

$$(\beta_i - \beta_j) = \frac{1}{\lambda_2} \left(\boldsymbol{x_i} - \boldsymbol{x_j}\right)^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$$

Then, we have

$$(\beta_i - \beta_j)^2 \leq \frac{1}{\lambda_2^2} \|\boldsymbol{x_i} - \boldsymbol{x_j}\|^2 \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 \quad \text{(by Cauchy)}$$

Then we can bound the inequality by parts. For $\|\boldsymbol{x_i} - \boldsymbol{x_j}\|^2$, by centered dataset, we have $\|\boldsymbol{x_i}\|^2 = 1, i = 1, ..., p$ and $\boldsymbol{x_i}^T \boldsymbol{x_j} = \rho$, so

$$\|\boldsymbol{x_i} - \boldsymbol{x_j}\|^2 = \|\boldsymbol{x_i}\|^2 + \|\boldsymbol{x_j}\|^2 - 2\boldsymbol{x_i}^T \boldsymbol{x_j} = 2(1 - \rho).$$

Grouping effect (cont'd)

For $\|\boldsymbol{y} - \boldsymbol{X}\beta\|^2$, by optimization,

$$\|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\|_2^2 + \lambda_1\|\hat{\boldsymbol{\beta}}\|_1 + \lambda_2\|\hat{\boldsymbol{\beta}}\|_2^2 = L(\hat{\boldsymbol{\beta}}) \leq L(0) = \|\boldsymbol{y}\|_2^2.$$

then,

$$\|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\|_2^2 \leq \|\boldsymbol{y}\|_2^2 - \lambda_1\|\hat{\boldsymbol{\beta}}\|_1 - \lambda_2\|\hat{\boldsymbol{\beta}}\|_2^2 \leq \|\boldsymbol{y}\|_2^2.$$

Combining all the upper bounds, we have

$$|\beta_i - \beta_j| \leqslant \frac{1}{\lambda_2}\sqrt{2(1-p)}\|y\|_2,$$

which completes the proof.

Quadratic Form

Assume the design matrix has two centered and standardized columns, with inner products given by

$$x_1^\top x_1 = x_2^\top x_2 = 1, \quad x_1^\top x_2 = \rho, \quad -1 \leq \rho \leq 1.$$

For the coefficient vector $\beta = (\beta_1, \beta_2)^\top$, the squared error can be written as a quadratic form

$$L(\beta) = \frac{1}{2}\|y - X\beta\|_2^2 = \frac{1}{2}\left(\beta - \hat{\beta}_{\mathrm{OLS}}\right)^\top G \left(\beta - \hat{\beta}_{\mathrm{OLS}}\right) + \text{ const },$$

where the Gram matrix is

$$G = X^\top X = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

and the contours of the loss function, centered at $\hat{\beta}_{\mathrm{OLS}}$, are given by the equation

$$\left(\beta - \hat{\beta}_{\mathrm{OLS}}\right)^\top G \left(\beta - \hat{\beta}_{\mathrm{OLS}}\right) = \text{ const }.$$

Eigenvalues

Perform eigen-decomposition of G. Solve for eigenvalues μ:

$$\det(G - \mu I) = \det \begin{pmatrix} 1-\mu & \rho \\ \rho & 1-\mu \end{pmatrix} = (1-\mu)^2 - \rho^2 = 0.$$

The solutions are

$$\mu_1 = 1 + \rho, \quad \mu_2 = 1 - \rho.$$

The corresponding (unnormalized) eigenvectors can be chosen as

$$v^{(1)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{(in the direction } y = x\text{)} , \quad v^{(2)} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad (y = -x) .$$

Ellipse

In the eigenvector basis, let $u = Q^\top \left(\beta - \hat{\beta}_{\mathrm{OLS}}\right)$, where Q is composed of the unit eigenvectors as columns. The quadratic form becomes

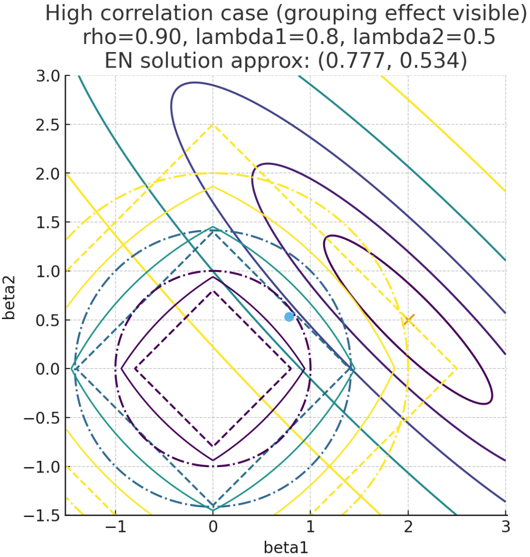
$$(\beta - \hat{\beta})^\top G(\beta - \hat{\beta}) = \mu_1 u_1^2 + \mu_2 u_2^2.$$

Fixing the level $= c$ (i.e., the contour), we obtain the ellipse equation

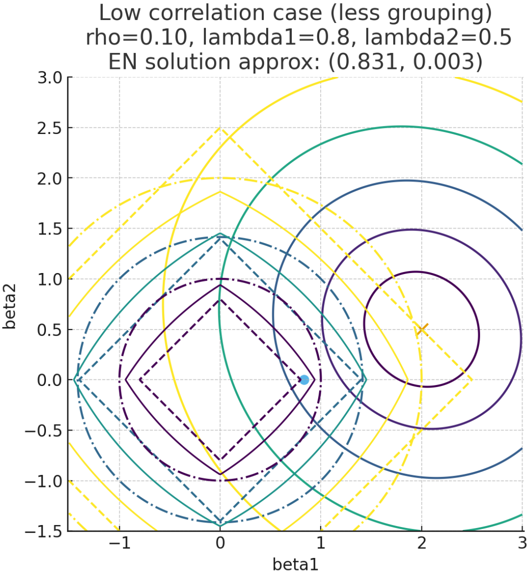
$$\frac{u_1^2}{c/\mu_1} + \frac{u_2^2}{c/\mu_2} = 1.$$

From this, the semi-axis length in the i-th eigenvector direction is $\sqrt{c/\mu_i}$. Thus, the smaller the eigenvalue, the longer the semi-axis in the corresponding direction. Noting that $\mu_2 = 1 - \rho \approx 0$, the semi-axis in the $y = -x$ direction is very long, and the loss function changes slowly in this direction. In other words, if the two variables are highly correlated, their coefficients can vary over a wide range with almost no change in the loss function value.

Grouping effect: an illustration



Grouping effect: an illustration



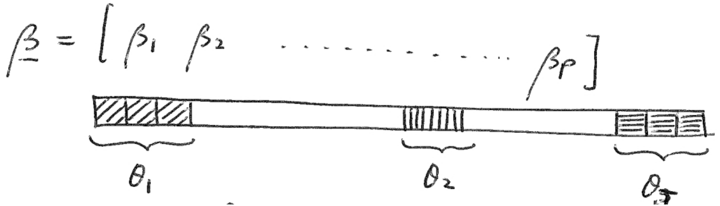
Group lasso

An example for group lasso

Genes and proteins often lie in known pathways, an investigator may be more interested in which pathway are related to an outcome than whether particular individual genes are.

- Groups of covariates should be selected into or out of a model together
- Desirable to have all coefficients within a group become nonzero (or zero) simultaneously

We use group lasso penalty (Yuan and Lin, 2006) for such situations.



Group lasso

- Consider linear regression model involving J groups of covariates, where $j = 1, ..., J$
- Vector $\boldsymbol{Z_j} = (X_{j1}, ..., X_{jp_j})^T \in \mathbb{R}^{p_j}$ represents the covariates in group j
  - p_j does not need to be the identical for all j
- Goal: predict real-valued response $Y \in \mathbb{R}$ based on collection of covariates $(Z_1, ..., Z_J)$
- Linear model $\mathbb{E}(Y|Z)$ takes the form $\theta_0 + \sum_{j=1}^{J} \boldsymbol{Z_j}^T \theta_j$, where $\theta_j \in \mathbb{R}^{p_j}$.

Group lasso

Given a collection of N samples $\{(y_i, z_{i1}, z_{i2}, \ldots, z_{iJ})\}_{i=1}^{N}$ the group lasso solves the convex problem:

$$\underset{\theta_0 \in \mathbb{R}, \theta_j \in \mathbb{R}^{p_j}}{\text{minimize}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(y_i - \theta_0 - \sum_{j=1}^{J} z_{ij}^T \theta_j \right)^2 + \lambda \sum_{j=1}^{J} \|\theta_j\|_2 \right\}$$

Where $\|\theta_j\|_2$ is the Euclidean norm.
This is group generalization of the lasso with properties:

- Depending on $\lambda \geq 0$ either the entire vector $\hat{\theta}_j$ will be zero, or all its elements will be nonzero.
- When $p_j = 1, j = 1, ..., J$, then we have $\|\theta_j\|_2 = |\theta_j|$, so reduces to the ordinary lasso.

Lasso vs group lasso

If $\boldsymbol{\beta}$ is individually sparse, then very likely all $\|\boldsymbol{\theta}_j\|_2$ will have value. But if $\boldsymbol{\beta}$ is group sparse, then only a few groups $\|\boldsymbol{\theta}_j\|_2$ is activated.

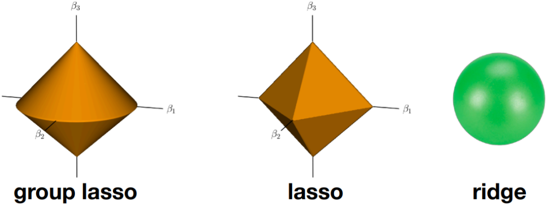
Lasso penalty:

- $|\beta|_1 = \sum_{j=1}^{p} |\beta_j|$,
- $\hat{\beta}$ is sparse.

Group lasso penalty:

- $\boldsymbol{\beta} = (\boldsymbol{\theta_1}, ..., \boldsymbol{\theta_J})$
- By notation abuse, denote $\|\boldsymbol{\beta}\|_{2,1} = \sum_{j=1}^{p} \|\theta\|_2$, denote $\boldsymbol{\theta} = (\|\theta_1\|_1, ..., \|\theta_J\|_2)$.
- $\hat{\boldsymbol{\theta}}$ is sparse.

An unit ball example for group lasso penalty



- Left unit ball can be characterized as $\sqrt{\beta_1^2 + \beta_2^2} + |\beta_3| \leq 1$
- Middle unit ball can be characterized as $|\beta_1| + |\beta_2| + |\beta_3| \leq 1$
- Right unit ball can be characterized as $\sqrt{\beta_1^2 + \beta_2^2 + \beta_3^2} \leq 1$

Multilevel factors in regression

- A predictor can be a categorical factor with multiple levels.
- Example: one continuous predictor X and a 3-level factor $G \in \{g_1, g_2, g_3\}$.
- Model for the mean:

$$\mathbb{E}(Y \mid X, G) = X\beta + \sum_{k=1}^{3} \theta_k \mathbb{I}_k[G]$$

- Interpretation: regression in X with different intercepts depending on G.

Dummy variable representation

- Introduce dummy vector $Z = (Z_1, Z_2, Z_3)$, where $Z_k = \mathbb{I}_k[G]$.
- Model becomes:

$$\mathbb{E}(Y \mid X, Z) = X\beta + Z^T\theta, \quad \theta = (\theta_1, \theta_2, \theta_3).$$

- If G has no predictive power $\Rightarrow \theta = 0$.
- Otherwise, coefficients in θ are typically nonzero.

General form with multiple factors

- With multiple group variables $G_1, \ldots, G_J$:

$$\mathbb{E}(Y \mid X, G_1, \ldots, G_J) = \beta_0 + X^T\beta + \sum_{j=1}^{J} Z_j^T\theta_j$$

- Variable selection often done at **group level**, not individual coefficients.
- **Group Lasso** is designed for this purpose.

Aliasing and coding issues

- In unpenalized regression, dummy variables in a set sum to one

$$\mathbb{E}(Y \mid X, G_1, \ldots, G_J) = \beta_0 + X^T \beta + \sum_{j=1}^{J} Z_j^T \theta_j$$

- This causes aliasing with the intercept (unidentifiable)
- Usual fix: use contrasts (e.g., sum-to-zero coding)
- With group lasso:
  - No aliasing concern due to ℓ_2 penalties
  - Penalty enforces coefficients in a group to sum to zero

Group-lasso and factors

- Problem:

$$\min_{\theta_0,\{\theta_j\}} \ \frac{1}{2}\sum_{i=1}^{N}\Big(y_i - \theta_0 - \sum_j z_{ij}^T\theta_j\Big)^2 + \lambda\sum_j \|\theta_j\|_2.$$

- If group j is a factor coded by dummies with $\sum_k z_{ij,k} = 1$ for each i, then

$$\text{at optimum} \quad \mathbf{1}^\top\theta_j = 0$$

- Proof idea: for any scalar c replace $\theta_0 \mapsto \theta_0 + c$, $\theta_j \mapsto \theta_j - c\mathbf{1}$. This leaves residuals unchanged but changes the penalty. Choosing $c = \frac{\mathbf{1}^\top\theta_j}{p_j}$ minimizes $\|\theta_j - c\mathbf{1}\|_2$ and forces the group coefficients to sum to zero. So optimal θ_j must have zero sum.

Multitask learning with group lasso

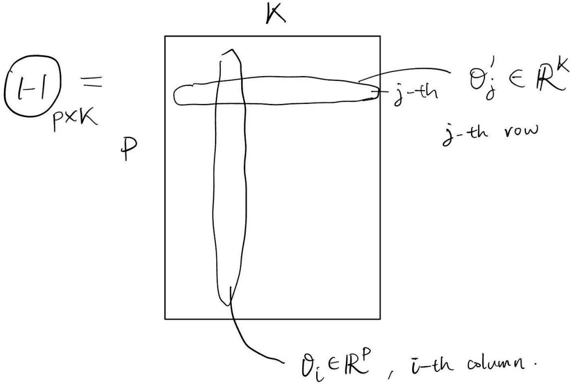
- Multivariate response $\boldsymbol{Y} \in \mathbb{R}^{N\times K}$, predictor matrix $\boldsymbol{X} \in \mathbb{R}^{N\times p}$.
- Matrix of coefficients $\boldsymbol{\Theta} \in \mathbb{R}^{p\times K}$ with, matrix of errors $\boldsymbol{E} \in \mathbb{R}^{N\times K}$.
- $Y \in \mathbb{R}^K$ may be correlated, for example, taking Y as K movies ratings of N users, then the ratings of different movies are correlated.
- Goal: estimate $\boldsymbol{\Theta}$ by minimizing the following objective function:

$$\underset{\boldsymbol{\Theta}\in\mathbb{R}^{p\times K}}{\text{minimize}} \left\{ \frac{1}{2}\|\mathbf{Y} - \mathbf{X}\boldsymbol{\Theta}\|_F^2 + \lambda \left(\sum_{j=1}^{p} \left\|\theta_j'\right\|_2 \right) \right\}$$

where $\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}$. θ_j' is the jth row of $\boldsymbol{\Theta}$, which means the coefficients of the jth task.

Multitask learning with group lasso

- If we only use $\sum_{j=1}^{p}\sum_{k=1}^{K}|\boldsymbol{\Theta}_{jk}|$, namely, $\|\boldsymbol{\Theta}\|_1$, then we have no way of controlling group sparsity.
- $\sum_{j=1}^{p}\left\|\theta_j'\right\|_2$ can do group sparsity because once the j^{th} row is actuated, all elements on the row are activated.



Computation for group lasso

We ignore the intercept and rewrite the optimization problem as follows:

$$\min_{(\theta_1,\ldots,\theta_j)} \left\{ \frac{1}{2} \left\| \boldsymbol{y} - \sum_{j=1}^{J} \boldsymbol{Z_j} \boldsymbol{\theta_j} \right\|_2^2 + \lambda \sum_{j=1}^{J} \left\| \boldsymbol{\theta_j} \right\|_2 \right\}.$$

By taking sub-derivative on the objective function and letting the derivative to be zero, we have the following estimating equation:

$$-\boldsymbol{Z_j}^T \left(\boldsymbol{y} - \sum_{j=1}^{J} \boldsymbol{Z_j} \hat{\theta}_j \right) + \lambda \hat{s}_j = 0,$$

where $\hat{s}_j \in \partial \|\hat{\theta}_j\|_2 = \begin{cases} \frac{\hat{\theta}_j}{\|\theta_j\|_2}, & \text{if } \hat{\theta}_j \neq 0, \\ \text{any } \boldsymbol{v} \text{ s.t. } \|\boldsymbol{v}\|_2 \leq 1, & \text{if } \hat{\theta}_j = 0. \end{cases}$

Computation for group lasso

Denote $\boldsymbol{r_j} = \boldsymbol{y} - \sum_{k \neq j} \boldsymbol{Z_k} \hat{\boldsymbol{\theta}}_{\boldsymbol{k}}$, then we have the estimating equation

$$-\boldsymbol{Z_j}^T \left(\boldsymbol{r_j} - \boldsymbol{Z_j} \hat{\theta}_j \right) + \lambda \hat{s}_j = 0.$$

By coordinate descent lemma, we have

$$\hat{\theta}_j = \begin{cases} \left(\boldsymbol{Z}_j^T \boldsymbol{Z}_j + \frac{\lambda}{\|\hat{\theta}_j\|_2} \boldsymbol{I} \right)^{-1} \boldsymbol{Z}_j^T \boldsymbol{r}_j, & \text{if } \|\boldsymbol{Z_j}^T \boldsymbol{r_j}\|_2 \geq \lambda, \\ \mathbf{0}, & \text{otherwise.} \end{cases}$$

This is not a closed form solution, one can use iterative methods to solve the equation, or add some assumptions on $\boldsymbol{Z_j}$, for example, if $\boldsymbol{Z_j}$ is orthogonal, then the solution is closed form.

Computation for group lasso

- We have

$$\hat{\theta}_j = \left(\boldsymbol{Z}_j^T \boldsymbol{Z}_j + \frac{\lambda}{\left\| \widehat{\theta}_j \right\|_2} \boldsymbol{I} \right)^{-1} \boldsymbol{Z}_j^T \boldsymbol{r}_j \cdot I \left(\|\boldsymbol{Z}_{\boldsymbol{j}}^T \boldsymbol{r}_{\boldsymbol{j}}\|_2 \geq \lambda \right)$$

- Further assume that $\boldsymbol{Z}_{\boldsymbol{j}}$ is orthonormal, then $\boldsymbol{Z}_{\boldsymbol{j}}^T \boldsymbol{Z}_{\boldsymbol{j}} = \boldsymbol{I}$, we have

$$\widehat{\theta}_j = \left(1 - \frac{\lambda}{\left\| \boldsymbol{Z}_j^T \boldsymbol{r}_j \right\|_2} \right)_+ \boldsymbol{Z}_j^T \boldsymbol{r}_j$$

Computation for group lasso with dummy variables

- For a factor with p_j levels, the dummy matrix Z_j has columns that are level indicators.
- Then
$$Z_j^T Z_j = \mathrm{diag}(n_{j,1}, \ldots, n_{j,p_j}),$$
where $n_{j,k}$ = number of observations in level k.
- Without normalization, groups with many observations produce larger values of $\|Z_j^T r\|_2$ and are more easily selected.
- Normalization ($Z_j^T Z_j = I$) $\rightsquigarrow$ standardize dummy columns (divide by $\sqrt{n_{j,k}}$)

Sparse group lasso

- When a group is included in a group-lasso fit, all the coefficients in that group are nonzero.
- We want sparsity both with respect to which groups are selected, and which coefficients are nonzero within a group $\rightsquigarrow$ variation of group lasso, called sparse group lasso

An example for sparse group lasso

Although a biological pathway may be implicated in the progression of a particular type of cancer, not all gene in the pathway need be active.

Sparse group lasso

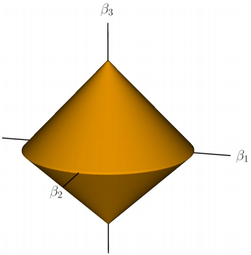
In order to achieve within-group sparsity, augment with additional ℓ_1-penalty, leading to the convex program:

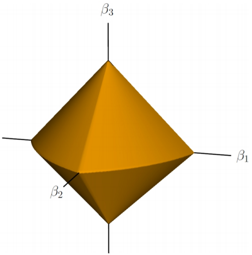
$$\underset{\{\theta_j \in \mathbb{R}^{p_j}\}_{j=1}^{J}}{\text{minimize}} \left\{ \frac{1}{2} \left\| \boldsymbol{y} - \sum_{j=1}^{J} \boldsymbol{Z}_j \theta_j \right\|_2^2 + \lambda \sum_{j=1}^{J} \left[(1-\alpha) \|\theta_j\|_2 + \alpha \|\theta_j\|_1 \right] \right\},$$

where $\alpha \in [0, 1]$.

- $\alpha = 0$, reduces to the group lasso.
- $\alpha = 1$, reduces to the lasso.

Sparse group lasso constraint region





- Left unit ball can be characterized as $\sqrt{\beta_1^2 + \beta_2^2} + |\beta_3| \leq 1$.
- Right unit ball can be characterized as $(1-\alpha)\sqrt{\beta_1^2 + \beta_2^2} + \alpha(|\beta_1| + |\beta_2|) + (1-\alpha)|\beta_3| + \alpha|\beta_3| \leq 1$.

Overlap group lasso

- Sometimes variables can belong to more than one group.
- Genes can belong to more than one biological pathway
- For example, we divide 5 variables into 2 groups,

$$\boldsymbol{Z}_1 = (X_1, X_2, X_3), \quad \boldsymbol{Z}_2 = (X_3, X_4, X_5).$$

  - If we simply replicate variable X_3, then use group lasso $\rightsquigarrow$ X_3 will be selected to the model with higher probability
  - If we simply replicate parameter then use sparse group lasso $\rightsquigarrow$ X_3 will be selected to the model only when both two groups are selected
- So **replicate variable** is preferred.

Overlap group lasso

- $\nu_j \in \mathbb{R}^p$ is a vector which is zero everywhere except in those positions corresponding to member of the group j.
- $\mathcal{V}_i \subseteq \mathbb{R}^p$ subspace of possible vectors.
- For $X = (X_1, \ldots, X_p)$ the coefficient vector is given by $\beta = \sum_{j=1}^{J} \nu_j$
- The overlap group lasso solves the problem:

$$\text{minimize}_{\nu_j \in \mathcal{V}_j, j=1,\ldots,J} \left\{ \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{X} \left(\sum_{j=1}^{J} \nu_j \right) \right\|_2^2 + \lambda \sum_{j=1}^{J} \|\nu_j\|_2 \right\}$$

- Jacob et al., (2009) showed that, the equivalent optimization problem can be put in the form:

$$\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\beta\|_2^2 + \lambda \Omega_{\mathcal{V}}(\beta) \right\}, \quad \Omega_{\mathcal{V}}(\beta) := \inf_{\substack{\nu_j \in \mathcal{V}_j \\ \beta = \sum_{j=1}^{J} \nu_j}} \sum_{j=1}^{J} \|\nu_j\|_2$$

Basis functions

- Suppose for the moment that we have just a single feature X and we are interested in estimating $\mathbb{E}(Y \mid X) = f(X)$
- A common approach for extending the linear model $f(X) = X\beta$ is to augment X with additional, known functions of X :

$$f(X) = \sum_{m=1}^{M} \beta_m h_m(X),$$

  where the $\{h_m\}$ are called basis functions
- Because the basis functions $\{h_m\}$ are prespecified and the model is linear in the new variables, ordinary least squares approaches can be used (at least in low-dimensional settings)

Basis functions

- Polynomial regression
  - Not only does this introduce bias, but it also results in extremely high variance near the edges of the range of x
  - Runge's phenomenon
- *local* basis functions, which ensure that a given observation affects only the nearby fit, not the fit of the entire line $\rightsquigarrow$ splines
  - Cubic splines
  - Natural cubic splines

Additive models

Additive models

Additive models are based on approximating the regression function by sums of the form:

$$f(x) = f(x_1, \ldots, x_J) \approx \sum_{j=1}^{J} f_j(x_j), \quad f_j \in \mathcal{F}_j, \quad j = 1, \ldots, J$$

- $\mathcal{F}_j$ are fixes set of univariate function classes
- Each $\mathcal{F}_j$ assumed to be a subset of $L^2(\mathbb{P}_j)$
- $\mathbb{P}_j$ is the distribution of covariate X_j equipped with squared $L^2(\mathbb{P}_j)$ norm

$$\|f_j\|_2^2 := \mathbb{E}\left[f_j^2(X_j)\right]$$

- Some theoretical results need $\mathcal{F}$ to be the Sobelev class of functions on $[a, b]$. (Buhlmann and van de Geer, 2010)

Additive models

Best additive approximation to regression function $\mathbb{E}(Y \mid X = x)$ solves problem:

$$\underset{f_j \in \mathcal{F}_j j=1,\ldots,J}{\text{minimize}} \, \mathbb{E}\left[\left(Y - \sum_{j=1}^{J} f_j\left(X_j\right)\right)^2\right], \mathcal{F}_j \subseteq L^2\left(\mathbb{P}_j\right), j = 1, \ldots, J$$

The optimal solution $\left(\tilde{f}_1, \ldots, \tilde{f}_J\right)$ is characterized by the **backfitting equations**:

$$\tilde{f}_j\left(x_j\right) = \mathbb{E}\left[Y - \sum_{k \neq j} \tilde{f}_k\left(X_k\right) \mid X_j = x_j\right], \text{ for } j = 1, \ldots, J$$

or $\tilde{f}_j = \mathcal{P}_j(Y - \sum_{k \neq j} \tilde{f}_k(X_k))$

Partial Residuals and Backfitting for Linear Models

The general form of a linear regression model is

$$\mathbb{E}[Y \mid \vec{X} = \vec{x}] = \beta_0 + \vec{\beta} \cdot \vec{x} = \sum_{j=0}^{p} \beta_j x_j$$

Suppose we don't condition on all of $\vec{X}$ but just one component of it, say X_k. What is the conditional expectation of Y ?

$$\begin{aligned}
\mathbb{E}\left[Y \mid X_k = x_k\right] &= \mathbb{E}\left[\mathbb{E}\left[Y \mid X_1, X_2, \ldots X_k, \ldots X_p\right] \mid X_k = x_k\right] \\
&= \mathbb{E}\left[\sum_{j=0}^{p} \beta_j X_j \mid X_k = x_k\right] \\
&= \beta_k x_k + \mathbb{E}\left[\sum_{j \neq k} \beta_j X_j \mid X_k = x_k\right]
\end{aligned}$$

Partial Residuals and Backfitting for LM/GAMs

Rearranging gives

$$\begin{aligned}\beta_k x_k &= \mathbb{E}\left[Y \mid X_k = x_k\right] - \mathbb{E}\left[\sum_{j\neq k} \beta_j X_j \mid X_k = x_k\right] \\ &= \mathbb{E}\left[Y - \left(\sum_{j\neq k} \beta_j X_j\right) \mid X_k = x_k\right]\end{aligned}$$

The expression in the expectation is the k^{th} partial residual. Let's introduce a symbol for this, say $Y^{(k)}$.

$$\beta_k x_k = \mathbb{E}\left[Y^{(k)} \mid X_k = x_k\right]$$

$\rightsquigarrow$ Gauss-Seidel type of algorithm for fitting linear models.
"a popular version of coordinate descent is known as backfitting and is used to fit generalized additive models"

Sparse additive models (SPAM)

- For $\lambda \geq 0$ type of k best sparse approximation:

$$\underset{\substack{f_j \in \mathcal{F}_j = 1,\ldots,J \\ |S| = k}}{\text{minimize}}\ \mathbb{E}\left[\left(Y - \sum_{j \in S} f_j\left(X_j\right)\right)^2\right]$$

  where $S \subset \{1, \ldots, J\} \rightsquigarrow$ 0-norm constraint on the number of nonzero components

- SPAM combines ideas from sparse linear modeling and additive nonparametric regression

$$\underset{f_j \in \mathcal{F}_j = 1,\ldots,J}{\text{minimize}} \left\{ \mathbb{E}\left[\left(Y - \sum_{j=1}^{J} f_j\left(X_j\right)\right)^2\right] + \lambda \sum_{j=1}^{J} \left\|f_j\right\|_2 \right\}, \quad \|f\|_2 = \sqrt{\mathbb{E}\left[f^2\left(X_j\right)\right]}$$

  This idea was originally proposed by Ravikumar et al. (2009)

COSSO

- COSSO method uses combination of the ℓ_1-norm with the Hilbert norm:

$$\|f\|_{\mathcal{H},1} := \sum_{j=1}^{J} \|f_j\|_{\mathcal{H}}$$

- COSSO's objective function is given by

$$\min_{f_j \in \mathcal{H}_j j=1,\dots,J} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{J} f_j\left(x_{ij}\right) \right)^2 + \lambda_{\mathcal{H}} \sum_{j=1}^{J} \|f_j\|_{\mathcal{H}_j} \right\}$$

- By Pythagorean theorem, the j^{th} coordinate function $\widehat{f}_j$ in any optimal COSSO solution can be written in the form $\widehat{f}_j(\cdot) = \sum_{i=1}^{N} \widehat{\theta}_{ij} \mathcal{R}_j\left(\cdot, x_{ij}\right)$, for a suitably chosen weight vector $\widehat{\boldsymbol{\theta}}_j \in \mathbb{R}^N \rightsquigarrow$ dimension reduction
- Gram matrix $\boldsymbol{R}_j \in \mathbb{R}^{N\times N}$ with entries $\left(\boldsymbol{R}_j\right)_{ii'} = \mathcal{R}_j\left(x_{ij}, x_{i'j}\right)$

COSSO with RKHS

- Let $\mathcal{H}_j = \text{span}\{\mathcal{R}_j(\cdot, x_{ij}), i = 1, \ldots, N\}$, we have

$$\begin{aligned}\|\hat{f}_j\|^2_{\mathcal{H}_j} &= \left\langle \sum_{i=1}^{N} \widehat{\theta}_{ij}\mathcal{R}_j(\cdot, x_{ij}), \sum_{i'=1}^{N} \widehat{\theta}_{i'j}\mathcal{R}_j(\cdot, x_{i'j}) \right\rangle \\ &= \sum_{i=1}^{N}\sum_{i'=1}^{N} \widehat{\theta}_{ij}\widehat{\theta}_{i'j}\mathcal{R}_j(x_{ij}, x_{i'j}) = \widehat{\boldsymbol{\theta}}_j^T \boldsymbol{R}_j \widehat{\boldsymbol{\theta}}_j\end{aligned}$$

- The COSSO optimization problem can be written as

$$\text{minimize}_{\boldsymbol{\theta}_j \in \mathbb{R}^N, j=1,\ldots,J} \left\{ \frac{1}{N} \left\| \boldsymbol{y} - \sum_{j=1}^{J} \boldsymbol{R}_j\boldsymbol{\theta}_j \right\|_2^2 + \tau \sum_{j=1}^{J} \sqrt{\boldsymbol{\theta}_j^T \boldsymbol{R}_j \boldsymbol{\theta}_j} \right\}$$

Computational techniques for COSSO

- Introducing $\gamma \in \mathbb{R}^J$, an equivalent formulation of COSSO is

$$\min_{\substack{f_j \in \mathcal{H}_j, j=1,\ldots J \\ \gamma \geq 0}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{J} f_j(x_{ij}) \right)^2 + \sum_{j=1}^{J} \frac{1}{\gamma_j} \|f_j\|_{\mathcal{H}_j}^2 + \lambda \sum_{j=1}^{J} \gamma_j \right\}$$

- if we set $\lambda = \tau^2/4 \rightsquigarrow$ equivalent to original COSSO formulation

alternates between two steps

1. For γ_j fixed, the problem results in an additive-spline fit
2. With the fitted additive spline fixed, updating the vector of coefficients $\gamma = (\gamma_1, \ldots, \gamma_J)$ amounts to a nonnegative lasso problem. $\boldsymbol{g}_j := \boldsymbol{R}_j \boldsymbol{\theta}_j / \gamma_j \in \mathbb{R}^N$, where $\boldsymbol{f}_j = \boldsymbol{R}_j \boldsymbol{\theta}_j$

$$\min_{\gamma \geq 0} \left\{ \frac{1}{N} \|\boldsymbol{y} - \boldsymbol{G}\gamma\|_2^2 + \lambda \|\gamma\|_1 \right\}$$

Multiple Penalization

- Multiple ways of enforcing sparsity for a nonparametric problem. (SPAM backfitting, COSSO).
- SPAM backfitting base on a combination of ℓ_1-norm: $\|f\|_{N,1} := \sum_{j=1}^{J} \|f_j\|_N$ with $\|f_j\|_N^2 := \frac{1}{N}\sum_{j=1}^{J} f_j^2\left(x_{ij}\right)$
- COSSO method uses combination of the ℓ_1-norm with the Hilbert norm:
$$\|f\|_{\mathcal{H},1} := \sum_{j=1}^{J} \|f_j\|_{\mathcal{H}}$$

Instead of focusing on only one regularizer, one might consider the more general family of estimators

$$\min_{\substack{f_j \in \mathcal{H}_j \\ j=1,\dots,J}} \left\{ \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \sum_{j=1}^{J} f_j\left(x_{ij}\right)\right)^2 + \lambda_{\mathcal{H}}\sum_{j=1}^{J}\|f_j\|_{\mathcal{H}_j} + \lambda_N \sum_{j=1}^{J}\|f_j\|_N \right\}$$

Why Multiple Penalization?

Two Penalties in Sparse Additive Models

$$L(f) = \frac{1}{2n} \sum_{i=1}^{n} \big(y_i - \bar{y} - f(x_i)\big)^2 + \lambda_n \|f\|_{n,1} + \rho_n \|f\|_{H,1}$$

- $\|f\|_{n,1} = \sum_{j=1}^{d} \|f_j\|_{L^2(n)}$
  - Encourages **sparsity**: many f_j are set to zero.
  - Controls **model selection** when $d \gg n$.
- $\|f\|_{H,1} = \sum_{j=1}^{d} \|f_j\|_{H_j}$
  - Encourages **smoothness**: prevents overfitting in each f_j.
  - Controls **function complexity** via RKHS norms.

Impact of Each Penalty

Impact of Each Penalty

- **Only sparsity penalty**: selects variables but risks wiggly, overfit functions.
- **Only smoothness penalty**: yields smooth fits but many irrelevant f_j remain nonzero.
- **Both penalties**: balance variable selection & smooth estimation, yielding minimax-optimal rates.

Fused lasso

The fused LASSO (signal approximator) solves the problem

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \left\{ \sum_{i=1}^{n} (Y_i - \theta_i)^2 + \lambda_1 \sum_{i=1}^{n} |\theta_i| + \lambda_2 \sum_{i=2}^{n} |\theta_i - \theta_{i-1}| \right\}.$$

More generally one can use the penalty

$$\lambda_2 \sum_{i \sim j} |\theta_i - \theta_j| ,$$

where $\sim$ is a relation depending on the problem at hand.

- Fused term : $\sum_{i=2}^{n} |\theta_i - \theta_{i-1}| \rightsquigarrow$ encourages neighboring coefficients θ_i to be similar
- Lasso term : $\sum_{i=1}^{n} |\theta_i| \rightsquigarrow$ encourages sparsity in the coefficients θ_i

Variation of fused lasso

Consider the Fused LASSO with a constant term θ_0, which means:

$$\min_{\theta_0,\theta} L(\theta_0,\theta) \triangleq \min_{\theta_0,\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i-\theta_0-\theta_i)^2+\lambda_1\sum_{i=1}^{n}|\theta_i|+\lambda_2\sum_{i=2}^{n}|\theta_i-\theta_{i-1}|,$$

Then we can get $\hat{\theta}_0$ by directly differentiate L by θ_0, get:

$$\hat{\theta}_0=\frac{1}{N}\sum_{i=1}^{N}y_i-\frac{1}{N}\sum_{i=1}^{N}\hat{\theta}_i.$$

Variation of fused lasso (cont'd)

We conclude that

$$\text{Median}\left(\theta_i, 1 \leq i \leq N\right) = 0.$$

Since if $\text{Median}\left(\theta_i, 1 \leq i \leq N\right) = m \neq 0$, we can change all of the θ_i to $\theta_i - m$, $1 \leq i \leq N$ and change θ_0 to $\theta_0 + m$, making the median to 0 and get:

$$\begin{aligned}
& L\left(\theta_0 + m, \theta - m \cdot 1\right) \\
& = \sum_{i=1}^{N} \left(y_i - (\theta_0 + m) - (\theta_i - m)\right)^2 + \lambda_1 \sum_{i=1}^{N} \left|\theta_i - m\right| + \lambda_2 \sum_{i=2}^{N} \left|\theta_i - \theta_{i-1}\right|,
\end{aligned}$$

which makes the loss function smaller.

Variation of fused lasso (cont'd)

Consider the Fused LASSO with a more general type:

$$\min_{\beta_0,\beta}\left\{\frac{1}{2}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda_1\sum_{j=1}^{p}|\beta_j|+\lambda_2\sum_{j=2}^{p}|\beta_j-\beta_{j-1}|\right\}.$$

We have $\hat{\beta}_0$ satisfies:

$$\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)=0\Rightarrow\hat{\beta}_0=\frac{1}{N}\sum_{i=1}^{N}y_i-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{p}x_{ij}\beta_j,$$

if x_{ij} and y_i are centered then

$$\hat{\beta}_0=\bar{y}-\sum_{j=1}^{p}\bar{x}_{\cdot j}\beta_j=0-0=0$$

we can actually omit β_0, independent with the choice of $\beta_i, 1\le i\le p$.

Computational techniques for fused lasso

Lemmas for fused lasso

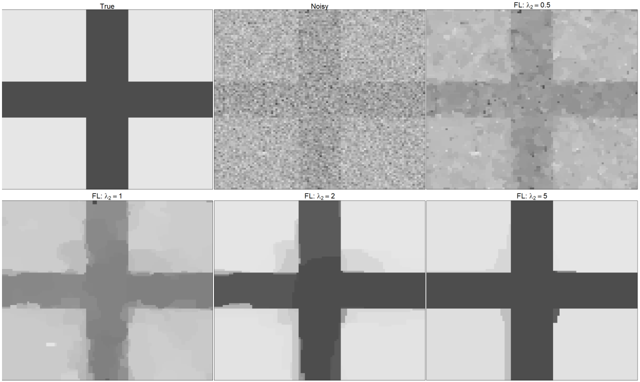
1. $\widehat{\theta}_i\left(\lambda_1, \lambda_2\right) = \mathcal{S}_{\lambda_1}\left(\widehat{\theta}_i\left(0, \lambda_2\right)\right)$ for each $i = 1, \ldots, N$.
2. Suppose that for some value of λ and some index $i \in \{1, \ldots, N-1\}$, the optimal solution satisfies $\widehat{\theta}_i(\lambda) = \widehat{\theta}_{i+1}(\lambda)$. Then for all $\lambda' > \lambda$, we also have $\widehat{\theta}_i\left(\lambda'\right) = \widehat{\theta}_{i+1}\left(\lambda'\right)$.

Hence we can focus on the optimization problem:

$$\underset{\boldsymbol{\theta} \in \mathbb{R}^N}{\text{minimize}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(y_i - \theta_i\right)^2 + \lambda \sum_{i=2}^{N} \left|\theta_i - \theta_{i-1}\right| \right\}.$$

- reparametrize
- start from $\lambda = 0$ and increase λ until all θ_i are fused
- use lagrangian duality to solve the problem

Case study: total variation denoising



- The idea here is that there exists a “true” image, but we only see a noisy image, from which we would like to recover the true image.

Trend Filtering

The fused LASSO is a special case of trend filtering, which fits piecewise polynomials to data. The idea is to minimize a criterion of the form

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{2} \sum_{i=1}^{n} (y_i - \beta_i)^2 + \lambda \left\| D^{(k+1)} \beta \right\|_1$$

where $D^{(k+1)}$ is the $(k+1)$ th order difference operator.
Specifically, the second difference operator is given by

$$D^{(2)} = \begin{bmatrix} -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ \vdots & & & & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 2 & -1 \end{bmatrix}$$

And the loss function is

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{2} \sum_{i=1}^{n} (y_i - \beta_i)^2 + \lambda \sum_{i=1}^{n-2} |\beta_i - 2\beta_{i+1} + \beta_{i+2}|$$

Change Point Detection

The penalty is assumed that the observations occur at evenly spaced points. For arbitrary input points $x_1 < x_2 < \cdots < x_n$, we can also use the following penalty [1] :

$$\frac{1}{2}\sum_{i=1}^{N}(y_i - \theta_i)^2 + \lambda \sum_{i=1}^{n-2}\left|\frac{\theta_i - \theta_{i+1}}{x_i - x_{i+1}} - \frac{\theta_{i+1} - \theta_{i+2}}{x_{i+1} - x_{i+2}}\right|$$

It encourages the slopes of the adjacent linear segments is the same, leading to piecewise linear fits.

- Here x_i stands for the time stamp/input feature of observation y_i.

[1] Tibshirani, R.J. (2014). Adaptive piecewise polynomial estimation via trend filtering. Annals of Statistics, 42(1), 285-323.

Isotonic regression

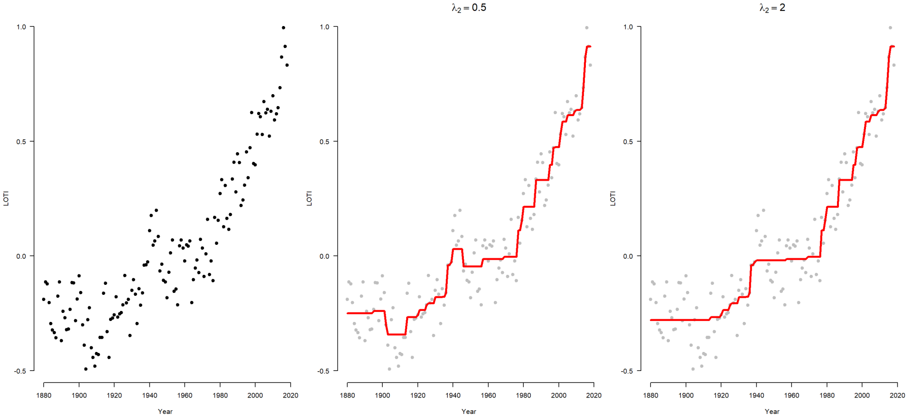
- Classical isotonic regression problem is to fit a nondecreasing sequence to a given sequence of observations $(y_1, \ldots, y_N)$ by solving the optimization problem

$$\underset{\boldsymbol{\theta} \in \mathbb{R}^N}{\text{minimize}} \left\{ \sum_{i=1}^{N} (y_i - \theta_i)^2 \right\} \text{ subject to } \theta_1 \leq \theta_2 \leq \ldots \leq \theta_N$$

- Nearly isotonic regression is a natural relaxation, in which we introduce a regularization parameter $\lambda \geq 0$, and instead solve the penalized problem

$$\underset{\boldsymbol{\theta} \in \mathbb{R}^N}{\text{minimize}} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \theta_i)^2 + \lambda \sum_{i=1}^{N-1} (\theta_i - \theta_{i+1})_+ \right\}$$

Case study: global warming



SCAD

- A variety of nonconvex penalties have been proposed: one of the earliest and most influential was the smoothly clipped absolute deviations (SCAD) penalty:

$$P_{\lambda,\gamma}(\theta) = \begin{cases} \lambda|\theta| & \text{if } |\theta| \leq \lambda \\ \frac{2\gamma\lambda|\theta| - \theta^2 - \lambda^2}{2(\gamma-1)} & \text{if } \lambda < |\theta| < \gamma\lambda \\ \frac{\lambda^2(\gamma+1)}{2} & \text{if } |\theta| \geq \gamma\lambda \end{cases}$$

  for $\gamma > 2$

- Note that SCAD coincides with the lasso until $|\theta| = \lambda$, then smoothly transitions to a quadratic function until $|\theta| = \gamma\lambda$, after which it remains constant for all $|\theta| > \gamma\lambda$

MCP

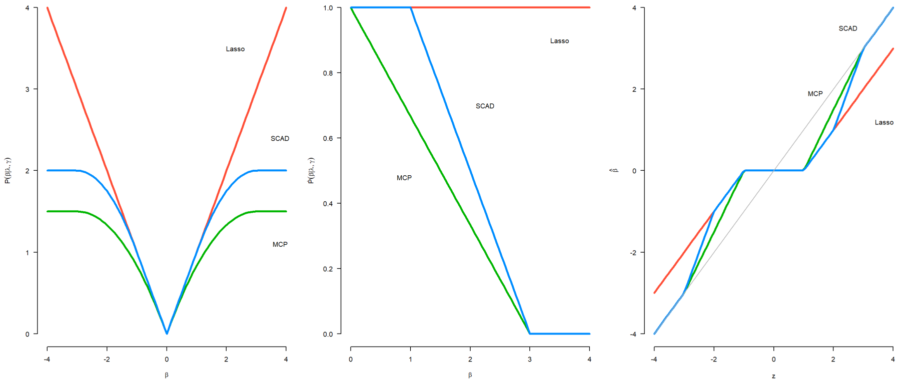
- The MC+ penalty on each coordinate is defined by

$$P_{\lambda,\gamma}(\theta) := \int_0^{|\theta|} \left(1 - \frac{x}{\lambda\gamma}\right)_+ dx = \begin{cases} |\theta| - \frac{|\theta|^2}{2\lambda\gamma}, & |\theta| < \lambda\gamma, \\ \frac{\lambda\gamma}{2}, & |\theta| \geq \lambda\gamma. \end{cases}$$

- For squared-error loss we pose the (nonconvex) optimization problem

$$\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\beta\|_2^2 + \sum_{j=1}^{p} P_{\lambda,\gamma}\left(\beta_j\right) \right\},$$

SCAD, MCP and lasso in 1 dimension



Questions or comments?

References

- Statistical Learning with Sparsity
- Linear models and extensions, Peng Ding
- The Elements of Statistical Learning
- Course slides for sparse learning in ETH Zurich
- Purdue ECE695Notes
- U Iowa High-Dimensional Data Analysis (BIOS 7240) course notes
- Survival analysis lecture notes, SYSU