

Sparse Linear Model

Some current challenges . . . are high dimensional data, sparsity, semi-supervised learning, the relation between computation and risk, and structured prediction.¹

¹John Lafferty and Larry Wasserman. Challenges in statistical machine learning. *Statistica Sinica*. Volume 16, Number 2, pp. 307-323, 2006.

1. Sparsity: where and why
2. Sparsity: how
3. Sparsity: algorithms
4. Sparsity: applications and extensions

Sparsity: where and why

1. Compressive sensing²

A signal processing technique for efficiently acquiring and reconstructing a signal, by finding solutions to underdetermined linear systems.

²<http://terrytao.wordpress.com/2007/04/13/compressed-sensing-and-single-pixel-cameras/>

Sparsity: where and why

1. Compressive sensing²

A signal processing technique for efficiently acquiring and reconstructing a signal, by finding solutions to underdetermined linear systems.

Traditional Compression

- Based on the acquired images,...
- Detect and eliminate redundancy

²<http://terrytao.wordpress.com/2007/04/13/compressed-sensing-and-single-pixel-cameras/>

Sparsity: where and why

1. Compressive sensing ²

A signal processing technique for efficiently acquiring and reconstructing a signal, by finding solutions to underdetermined linear systems.

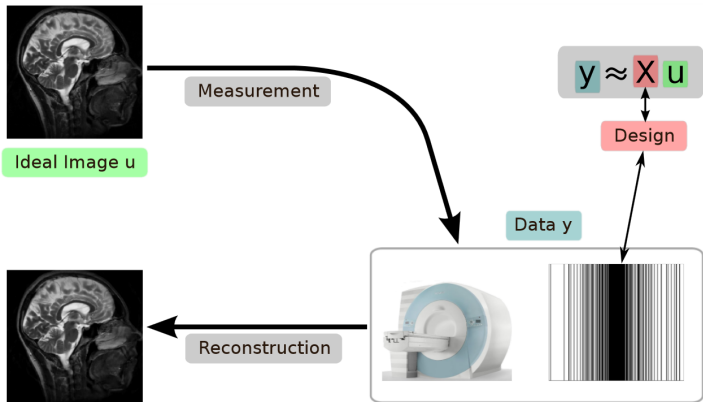
Traditional Compression

- Based on the acquired images,...
- Detect and eliminate redundancy

Fundamental Change of CS

- Based on the signal's sparsity
- Sensing compressed signal

²<http://terrytao.wordpress.com/2007/04/13/compressed-sensing-and-single-pixel-cameras/>

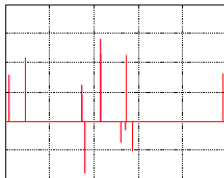
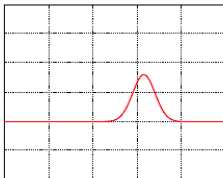
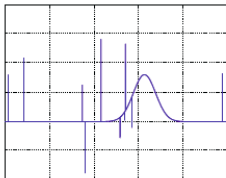


3

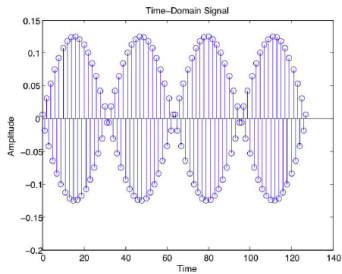
³http://mlss.tuebingen.mpg.de/2013/seeger_slides.pdf

Sparse Representations:

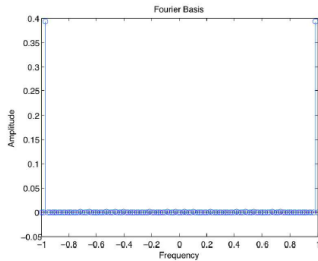
Consider the following signal (blue) and a dictionary $D = [I, G]$:



Some signals are inherently sparse:



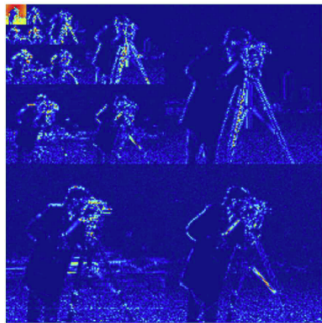
(a)



(b)



(a)



(b)

So why sparsity? ⁴

⁴<http://www-stat.stanford.edu/~candes/stats330/index.shtml>

So why sparsity? ⁴

- **Simplicity:** describe the signal with fewest elements.

⁴<http://www-stat.stanford.edu/~candes/stats330/index.shtml>

So why sparsity? ⁴

- **Simplicity:** describe the signal with fewest elements.
- **Meaningful:** the representation, by itself, describes the signal meaningful.

⁴<http://www-stat.stanford.edu/~candes/stats330/index.shtml>

So why sparsity? ⁴

- **Simplicity:** describe the signal with fewest elements.
- **Meaningful:** the representation, by itself, describes the signal meaningful.
- **Parsimony:** If a collection of signals enjoy K -sparse representation in a dictionary, then this space of signals has only K degrees of freedom

⁴<http://www-stat.stanford.edu/~candes/stats330/index.shtml>

2. LASSO

$$\hat{\beta}^{\text{lasso}} = \arg \min_{\beta} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

Why do we need lasso?

- LSE: $\hat{\beta} = (X^T X)^{-1} X^T y$ ($X^T X$ may be singular, e.g. **small** N , **large** p)
- **Prediction accuracy:** LSE is unbiased but may has high variance.
(avoid overfitting)
- **Interpretation:** determine a smaller subset that exhibits the strongest effects.

3. Sparsity and feature selection

By pursuing a sparse solution, we do make feature selection.

3. Sparsity and feature selection

By pursuing a sparse solution, we do make feature selection.

Variable-based Approach

- Low variance
- Low correlation or MI with the output
- ...

3. Sparsity and feature selection

By pursuing a sparse solution, we do make feature selection.

Variable-based Approach

- Low variance
- Low correlation or MI with the output
- ...

MLE or Bayesian

- Can we do feature/model selection by MLE?
- Can Bayesian work?

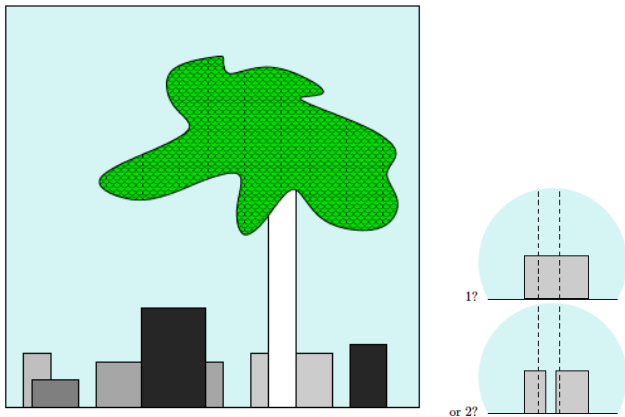
Bayesian Variable Selection

1. Why can Bayesian do model/feature selection but cannot MLE? ⁵

⁵Please reference "Information theory, inference and Learning Algorithms"

Bayesian Variable Selection

1. Why can Bayesian do model/feature selection but cannot MLE? ⁵



⁵Please reference "Information theory, inference and Learning Algorithms"

1.1 From a view of model comparison

Given two models, H_1, H_2 (suppose H_1 is 'simpler') and they can both fit to the experimental data D , which one is preferred?

$$\frac{P(H_1|D)}{P(H_2|D)} = \frac{P(H_1)}{P(H_2)} \frac{P(D|H_1)}{P(D|H_2)}$$

1.1 From a view of model comparison

Given two models, H_1, H_2 (suppose H_1 is 'simpler') and they can both fit to the experimental data D , which one is preferred?

$$\frac{P(H_1|D)}{P(H_2|D)} = \frac{P(H_1)}{P(H_2)} \frac{P(D|H_1)}{P(D|H_2)}$$

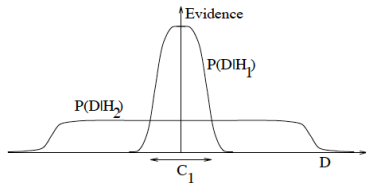
- Aesthetic and empirical motivations: specify a higher prior to the simpler model

1.1 From a view of model comparison

Given two models, H_1, H_2 (suppose H_1 is 'simpler') and they can both fit to the experimental data D , which one is preferred?

$$\frac{P(H_1|D)}{P(H_2|D)} = \frac{P(H_1)}{P(H_2)} \frac{P(D|H_1)}{P(D|H_2)}$$

- Aesthetic and empirical motivations: specify a higher prior to the simpler model
- The second factor (evidence) also favors the simpler model.



1.2 MLE will always choose the most complex model

- More complex model **can** always fit the data better.
- But over-parameterized models generalize poorly.

1.2 MLE will always choose the most complex model

- More complex model **can** always fit the data better.
- But over-parameterized models generalize poorly.

1.3 Evaluating the evidence

$$P(D|H_i) = \int P(D|w, H_i)P(w|H_i)dw$$

By Laplace's method and in terms of the fact that the posterior has a strong peak at the most probable parameters w_{mp}

$$P(D|H_i) \approx P(D|w_{MP}, H_i) \times P(w_{MP}|H_i)\sigma_{w|D}$$

Why Bayesian variable selection?

- Equipped with natural measures of uncertainty, such as the posterior probability of each model and the marginal inclusion probabilities of the individual predictors.
- Model uncertainty can be incorporated into prediction through model averaging, which usually improves prediction.

Example: R package "BMA" can be used for Bayesian variable selection.

```
summary(bicreg(X,y))
```

Variable	Post mean	Post sd.	Prob non-zero
INDUS	-0.13	0.25	0.28
NOX	-0.34	0.26	0.51
RM	3.45	0.41	1.00
TAX	-0.24	0.34	0.43
PT	-0.68	0.33	0.86
LSTAT	-1.93	0.38	1.00

2. Bayesian variable selection

Define an indicator for every variable j :

$$\gamma_j = \begin{cases} 1 & \text{if } j \text{ is relevant} \\ 0 & \text{otherwise} \end{cases}$$

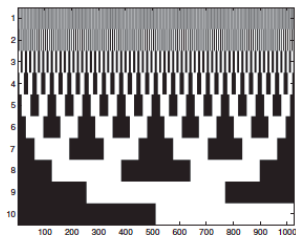
Bayesian wants to compute the posterior over models:

(D is the observed dataset)

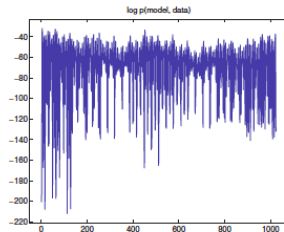
$$p(\gamma|D) = \frac{p(D|\gamma)p(\gamma)}{\sum_{\gamma'} p(D, \gamma')}$$

A simulating experiment: $N = 20, y_i \sim \mathcal{N}(\mathbf{w}^T \mathbf{x}, \sigma^2), \mathbf{x} \in R^{10}, \sigma = 1$

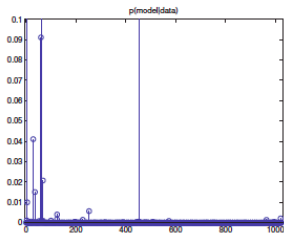
$$\mathbf{w} = [0, -1.67, 0.13, 0, 0, 1.19, -0.04, 0.33, 0]$$



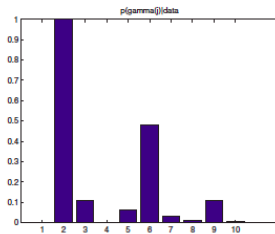
(a)



(b)



(c)



(d)

Two fundamental observations:

- For finite data sets, some variables (coefficients are small relative to the noise) are hard to detect and there will usually be considerable posterior uncertainty.
- There're 2^p possible models. So it'll be impossible to compute the full posterior in general, and even finding summaries, such as MAP.

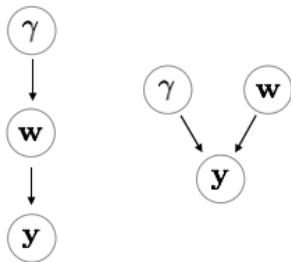
- We like and use Gaussian:
 - Fast, simple, well understood
- But we need sparsity ...

$$p(\gamma|D) \propto p(D|\gamma)p(\gamma) = p(\mathbf{y}|\mathbf{X}, \gamma)p(\gamma)$$

$$y_i \sim \mathcal{N}(\mathbf{w}^{\mathbf{T}} \mathbf{x}, \sigma^2)$$

$$p(\gamma|D) \propto p(D|\gamma)p(\gamma) = p(\mathbf{y}|\mathbf{X}, \gamma)p(\gamma)$$

$$y_i \sim \mathcal{N}(\mathbf{w}^T \mathbf{x}, \sigma^2)$$



2.1 The spike and slab model

- Bernoulli prior on the bit vector γ :

$$p(\gamma; \pi_0) = \prod_{j=1}^p \text{Ber}(\gamma_j; \pi_0) = \pi_0^{||\gamma||_0} (1 - \pi_0)^{p - ||\gamma||_0}$$

$$\log p(\gamma; \pi_0) = -\lambda ||\gamma||_0 + \text{const}$$

- Prior on \mathbf{w} : $p(\mathbf{w}|\gamma, \sigma^2)$ (Spike and slab model)

$$p(\mathbf{w}_j|\gamma_j, \sigma^2) = \begin{cases} \delta_0(w_j) & \text{if } \gamma_j = 0 \\ \mathcal{N}(w_j; 0, \sigma^2 \sigma_w^2) & \text{if } \gamma_j = 1 \end{cases}$$

- Prior on \mathbf{w} : $p(\mathbf{w}|\gamma, \sigma^2)$ (Spike and slab model)

$$p(\mathbf{w}_j|\gamma_j, \sigma^2) = \begin{cases} \delta_0(w_j) & \text{if } \gamma_j = 0 \\ \mathcal{N}(w_j; 0, \sigma^2 \sigma_w^2) & \text{if } \gamma_j = 1 \end{cases}$$

- Likelihood:

$$p(D|\gamma) = \iint p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \gamma) p(\mathbf{w}|\gamma, \sigma^2) p(\sigma^2) d\mathbf{w} d\sigma^2$$

$$p(D|\gamma, \sigma^2) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathbf{C}_\gamma) \quad \mathbf{C}_\gamma = \sigma^2 \mathbf{X}_\gamma \Sigma_\gamma \mathbf{X}_\gamma^T + \sigma^2 \mathbf{I}_N$$

Note this marginal likelihood can be approximated by BIC score:

$$\log p(D|\gamma) \approx \log p(\mathbf{y}|\mathbf{X}, \hat{\mathbf{w}}_\gamma, \sigma^2) - \frac{\|\gamma\|_0}{2} \log N$$

- "Degree of Freedom": $\|\gamma\|_0$

Note this marginal likelihood can be approximated by BIC score:

$$\log p(D|\gamma) \approx \log p(\mathbf{y}|\mathbf{X}, \hat{\mathbf{w}}_\gamma, \sigma^2) - \frac{\|\gamma\|_0}{2} \log N$$

- "Degree of Freedom": $\|\gamma\|_0$

Now we can write out the overall objective:

$$\log p(\gamma|D) \approx \log p(\mathbf{y}|\mathbf{X}, \hat{\mathbf{w}}_\gamma, \sigma^2) - \frac{\|\gamma\|_0}{2} \log N - \lambda \|\gamma\|_0 + \text{const}$$

2.2 The Bernoulli-Gaussian model

$$\begin{aligned}y_i | \mathbf{x}_i, \mathbf{w}, \gamma, \sigma^2 &\sim \mathcal{N}(\sum_j \gamma_j w_j x_{ij}, \sigma^2) \\ \gamma_j &\sim \text{Ber}(\pi_0) \\ w_j &\sim \mathcal{N}(0, \sigma_w^2)\end{aligned}$$

It isn't difficult to know:

$$\begin{aligned}f(\gamma, \mathbf{w}) &\triangleq -2\sigma^2 \log p(\gamma, \mathbf{w}, \mathbf{y} | \mathbf{X}) \\ &= \|\mathbf{y} - \mathbf{X}(\gamma \cdot \times \mathbf{w})\|^2 + \frac{\sigma^2}{\sigma_w^2} \|\mathbf{w}\|^2 + \lambda \|\gamma\|_0 + \text{const}\end{aligned}$$

2.2 The Bernoulli-Gaussian model

$$\begin{aligned}y_i | \mathbf{x}_i, \mathbf{w}, \gamma, \sigma^2 &\sim \mathcal{N}(\sum_j \gamma_j w_j x_{ij}, \sigma^2) \\ \gamma_j &\sim \text{Ber}(\pi_0) \\ w_j &\sim \mathcal{N}(0, \sigma_w^2)\end{aligned}$$

It isn't difficult to know:

$$\begin{aligned}f(\gamma, \mathbf{w}) &\triangleq -2\sigma^2 \log p(\gamma, \mathbf{w}, \mathbf{y} | \mathbf{X}) \\ &= \|\mathbf{y} - \mathbf{X}(\gamma \cdot \times \mathbf{w})\|^2 + \frac{\sigma^2}{\sigma_w^2} \|\mathbf{w}\|^2 + \lambda \|\gamma\|_0 + \text{const}\end{aligned}$$

Now consider the case $\sigma_w \rightarrow \infty$ (a uniform distribution of w):

$$f(\gamma, \mathbf{w}) = \|\mathbf{y} - \mathbf{X}_\gamma \mathbf{w}_\gamma\|^2 + \lambda \|\gamma\|_0$$



Karl Broman

@kwbroman



Following

That the 1st 11 socks in the laundry are each distinct suggests there are a lot more socks.

Reply Retweet Favorite More



FAVORITES

6



2:20 PM - 17 Oct 2014

l_0 regularization

$$\min \quad f(\mathbf{w}) \triangleq ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_0$$

- Is l_0 a norm?

l_0 regularization

$$\min \quad f(\mathbf{w}) \triangleq ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_0$$

- Is l_0 a norm?
 - **pseudo-norm**
- How to solve?

l_0 regularization

$$\min \quad f(\mathbf{w}) \triangleq ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_0$$

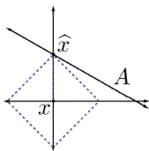
- Is l_0 a norm?
 - **pseudo-norm**
- How to solve?
 - **NP-hard. (heuristic or approximation)**

Heuristic Strategies for l_0 Optimization

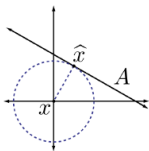
The basic idea is to **search** through the space of models.

- How to search while we cannot explore the full space(2^p)?
 - Greedy search
 - Stochastic search
- How to efficiently evaluate every model?

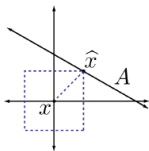
From l_0 to l_1 Regularization



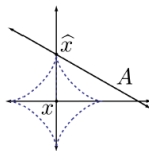
(a)



(b)



(c)



(d)

- Bayesian view: Laplace prior

$$p(\mathbf{w}|\lambda) = \prod_{j=1}^p \text{Lap}(w_j; 0, \frac{1}{\lambda}) \propto \prod_{j=1}^p e^{-\lambda|w_j|}$$

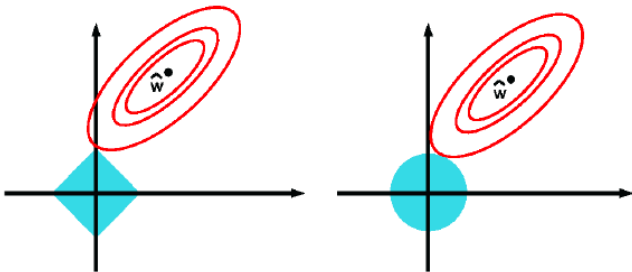
$$f(\mathbf{w}) = -\log p(D|\mathbf{w}) - \log p(\mathbf{w}; \lambda) = \text{NLL}(\mathbf{w}) + \lambda \|\mathbf{w}\|_1$$

- In the case of linear regression:

$$\min_{\mathbf{w}} \text{RSS}(\mathbf{w}) + \lambda \|\mathbf{w}\|_1$$

Why l_1 yields sparse solutions?

$$\begin{aligned} \min_{\mathbf{w}} \quad & \text{RSS}(\mathbf{w}) = \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 \\ \text{s.t.} \quad & \|\mathbf{w}\|_1 \leq t \end{aligned}$$



LASSO Solutions

- I. General theory
- II. Regularization path - LARS
- III. General l_1 -norm optimization - ADMM

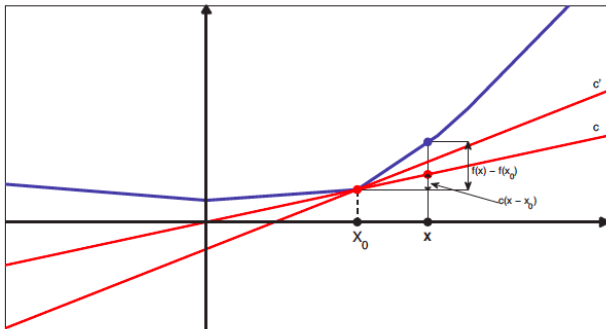
Subgradient

Definition:

Let f be a convex function. A vector g is called a subgradient of function f at point $x_0 \in \text{dom } f$ if:

$$\forall x \in \text{dom } f, f(x) \geq f(x_0) + \langle g, x - x_0 \rangle$$

And the set of all subgradients of f at x_0 , denoted as $\partial f(x_0)$ is called the subdifferential set of f at x_0 .



Consider the function $f(x) = |x|, x \in R^1$.

$$\partial f(x) = \begin{cases} \{-1\} & \text{if } x < 0 \\ [-1, 1] & \text{if } x = 0 \\ \{1\} & \text{if } x > 0 \end{cases}$$

Subgradients and Optimization

We have $f(x^*) = \min_{x \in \text{dom } f} f(x)$ if and only if:

$$0 \in \partial f(x^*)$$

Consider a simpler problem than LASSO:

$$\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{w}\|^2 + \lambda \|\mathbf{w}\|_1 \quad \mathbf{w} \in R^p$$

- Component separability.

$$\|\mathbf{y} - \mathbf{w}\|_2^2 = (y_1 - w_1)^2 + \cdots + (y_n - w_n)^2$$

- Consider the n scalar minimization problem.

$$w_i^+ = \arg \min_{w_i} (\lambda |w_i| + (y_i - w_i)^2)$$

- The subdifferential of $f(w_i) = \lambda|w_i| + (y_i - w_i)^2$ is:

$$\partial f(w_i) = \begin{cases} \{-\lambda + 2(w_i - y_i)\} & w_i < 0 \\ \{\lambda + 2(w_i - y_i)\} & w_i > 0 \\ [-\lambda + 2(w_i - y_i), \lambda + 2(w_i - y_i)] & w_i = 0 \end{cases}$$

- So we can get the solution:

$$w_i^+ := S_{\frac{\lambda}{2}}(y_i) = \text{sign}(y_i) \max(|y_i| - \frac{\lambda}{2}, 0)$$

where S is a **soft thresholding** operator:

$$S_{\tau}(a) = \begin{cases} a - \tau & a > \tau \\ a + \tau & a < -\tau \\ 0 & |a| < \tau \end{cases}$$

Now let's consider LASSO:

$$\min_{\mathbf{w}} \quad ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_1$$

$$\begin{aligned} & \frac{\partial}{\partial w_j} \text{RSS}(\mathbf{w}) \\ = & 2(\mathbf{X}\mathbf{w} - \mathbf{y})^T \mathbf{x}_{\cdot j} \\ = & 2 \sum_{i=1}^N x_{ij}^2 w_j - 2 \sum_{i=1}^N x_{ij} (y_i - \mathbf{w}_{-j}^T \mathbf{x}_{i,-j}) \\ = & a_j w_j - c_j \end{aligned}$$

$$\begin{aligned} & \partial_{w_j} f(\mathbf{w}) \\ = & (a_j w_j - c_j) + \lambda \partial_{w_j} ||\mathbf{w}||_1 \\ = & \begin{cases} \{a_j w_j - c_j - \lambda\} & \text{if } w_j < 0 \\ [-c_j - \lambda, -c_j + \lambda] & \text{if } w_j = 0 \\ \{a_j w_j - c_j + \lambda\} & \text{if } w_j > 0 \end{cases} \end{aligned}$$

Now let's consider LASSO:

$$\min_{\mathbf{w}} \quad ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_1$$

$$\begin{aligned} & \frac{\partial}{\partial w_j} \text{RSS}(\mathbf{w}) && \partial_{w_j} f(\mathbf{w}) \\ = & 2(\mathbf{X}\mathbf{w} - \mathbf{y})^T \mathbf{x}_{\cdot j} && = (a_j w_j - c_j) + \lambda \partial_{w_j} ||\mathbf{w}||_1 \\ = & 2 \sum_{i=1}^N x_{ij}^2 w_j - 2 \sum_{i=1}^N x_{ij} (y_i - \mathbf{w}_{-j}^T \mathbf{x}_{i,-j}) && = \begin{cases} \{a_j w_j - c_j - \lambda\} & \text{if } w_j < 0 \\ [-c_j - \lambda, -c_j + \lambda] & \text{if } w_j = 0 \\ \{a_j w_j - c_j + \lambda\} & \text{if } w_j > 0 \end{cases} \\ = & a_j w_j - c_j \end{aligned}$$

LASSO: least absolute selection and shrinkage operator?

$$\hat{w}_j = S_{\frac{\lambda}{a_j}}\left(\frac{c_j}{a_j}\right) = \text{sign}\left(\frac{c_j}{a_j}\right) \max\left(\left|\frac{c_j}{a_j}\right| - \frac{\lambda}{a_j}, 0\right)$$

- Select a subset of variables and shrinks all the coefficients by penalizing the absolute values.

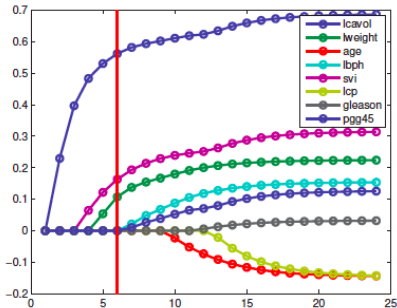
LASSO Algorithm(1)

Algorithm 13.1: Coordinate descent for lasso (aka shooting algorithm)

```
1 Initialize  $\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$ ;  
2 repeat  
3   for  $j = 1, \dots, D$  do  
4      $a_j = 2 \sum_{i=1}^n x_{ij}^2$ ;  
5      $c_j = 2 \sum_{i=1}^n x_{ij} (y_i - \mathbf{w}^T \mathbf{x}_i + w_j x_{ij})$  ;  
6      $w_j = \text{soft}(\frac{c_j}{a_j}, \frac{\lambda}{a_j})$ ;  
7 until converged;
```

LASSO Algorithm(2) – LARS

- Regularization path



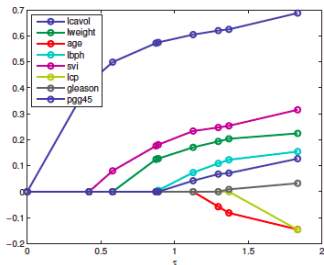
Firstly, consider the λ_{\max} that makes all coefficients zeroes.

$$\hat{w}_j = S_{\frac{\lambda}{a_j}}\left(\frac{c_j}{a_j}\right) = \text{sign}\left(\frac{c_j}{a_j}\right) \max\left(\left|\frac{c_j}{a_j}\right| - \frac{\lambda}{a_j}, 0\right)$$

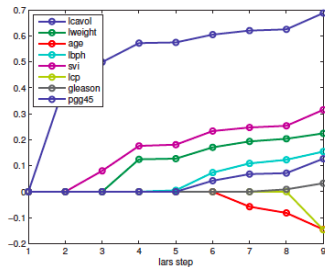
Set $c_j \in [-\lambda, \lambda]$, and:

$$\lambda_{\max} = \|\mathbf{X}^T \mathbf{y}\|_{\infty} = \max_j |\mathbf{y}^T \mathbf{x}_{\cdot j}|$$

LARS(Least Angle Regression and Shrinkage) can find the whole lasso path.



(a)

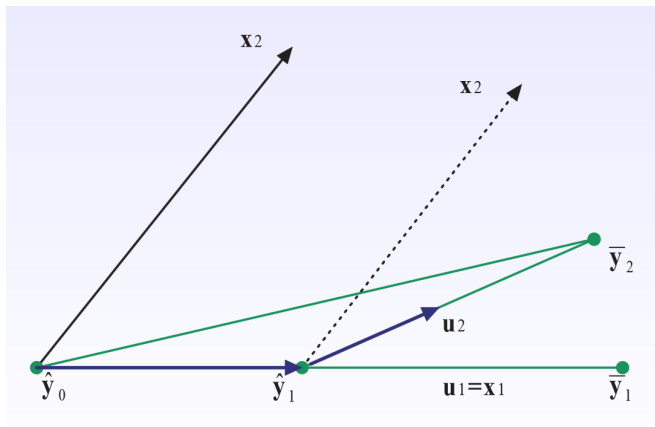


(b)

Least Angle Regression ⁶

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \dots, \beta_p = 0$.
2. Find the predictor \mathbf{x}_j most correlated with \mathbf{r} .
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor \mathbf{x}_k has as much correlation with the current residual as does \mathbf{x}_j .
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor \mathbf{x}_l has as much correlation with the current residual.
5. Continue in this way until all p predictors have been entered. After $\min(N - 1, p)$ steps, we arrive at the full least-squares solution.

⁶Please reference "The Elements of Statistical Learning"



LARS:LASSO modification

- 4a. If a non-zero coefficient hits zero, drop its variable from the active set of variables and recompute the current joint least squares direction.

LASSO Algorithm(3) – ADMM ⁷

- Alternating Direction Method of Multipliers
- A simple but powerful algorithm that is suited to **large-scale distributed convex optimization**
- Follow a decomposition-coordination procedure

⁷Please reference "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers"

Problem ADMM is suited:

$$\begin{array}{ll}\min & f(x) + g(z) \\ \text{s.t.} & Ax + Bz = c\end{array}$$

General convex optimization problem: (f is a convex function and \mathcal{C} is a convex set)

$$\begin{array}{ll}\min & f(x) \\ \text{s.t.} & x \in \mathbf{C}\end{array}$$

Define g as an indicator function of \mathcal{C} , so the above problem can be rewritten as:

$$\begin{array}{ll}\min & f(x) + g(z) \\ \text{s.t.} & x - z = 0\end{array}$$

ADMM Algorithm

$$\begin{array}{ll}\min & f(x) + g(z) \\ \text{s.t.} & Ax + Bz = c\end{array}$$

- Define the **augmented Lagrangian** as:

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2} \|Ax + Bz - c\|_2^2$$

- Iteration: (simultaneously update primal and dual variables)

$$x^{k+1} := \arg \min_x L_\rho(x, z^k, y^k)$$

$$z^{k+1} := \arg \min_z L_\rho(x^{k+1}, z, y^k)$$

$$y^{k+1} := y^k + \rho(Ax^{k+1} + Bz^{k+1} - c)$$

ADMM for LASSO

$$\min_{\mathbf{w}} \quad ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{w}||_1$$

- Rewrite is in ADMM form:

$$\begin{aligned} \min \quad & \frac{1}{2} ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{z}||_1 \\ \text{s.t.} \quad & \mathbf{w} - \mathbf{z} = 0 \end{aligned}$$

- Argumented Lagrangian:

$$L_{\rho}(\mathbf{w}, \mathbf{z}, \mathbf{t}) = \frac{1}{2} ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda ||\mathbf{z}||_1 + \mathbf{t}^T (\mathbf{w} - \mathbf{z}) + \frac{\rho}{2} ||\mathbf{w} - \mathbf{z}||^2$$

$$L_{\rho}(\mathbf{w}, \mathbf{z}, \mathbf{t}) = \frac{1}{2} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \lambda \|\mathbf{z}\|_1 + \mathbf{t}^T(\mathbf{w} - \mathbf{z}) + \frac{\rho}{2} \|\mathbf{w} - \mathbf{z}\|^2$$

- Update \mathbf{w} given $\mathbf{z}^k, \mathbf{t}^k$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{X}^T(\mathbf{X}\mathbf{w} - \mathbf{y}) + \mathbf{t} + \rho(\mathbf{w} - \mathbf{z}) = 0$$

$$\mathbf{w}^{k+1} = (\mathbf{X}^T\mathbf{X} + \rho\mathbf{I})^{-1}(\mathbf{X}^T\mathbf{y} - \mathbf{t}^k + \rho\mathbf{z}^k)$$

- Update \mathbf{z}^k given $\mathbf{w}^{k+1}, \mathbf{t}^k$

$$\text{Subgradient: } \lambda \partial \|\mathbf{z}\|_1 - \mathbf{t} + \rho(\mathbf{z} - \mathbf{w})$$

$$\mathbf{z}^{k+1} = S_{\frac{\lambda}{\rho}}(\mathbf{w}^{k+1} + \mathbf{t}^k)$$

- Update \mathbf{t}^{k+1} given \mathbf{w}^{k+1} and \mathbf{z}^{k+1}

$$\mathbf{t}^{k+1} = \mathbf{t}^k + \rho(\mathbf{w}^{k+1} - \mathbf{z}^{k+1})$$

Comparison of Different Linear Regression Methods⁸

- OLS
- Least Absolute Deviations
- Subset selection
- Ridge Regression
- LASSO
- PCA regression
- Partial Least Square (PLS)
- ...

⁸For details, "The Elements of Statistical Learning"

Minimize different norms of residuals

- Ordinary least square

$$\min_{\mathbf{w}} \quad ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2$$

- Least Absolute Deviations (Robust linear regression)

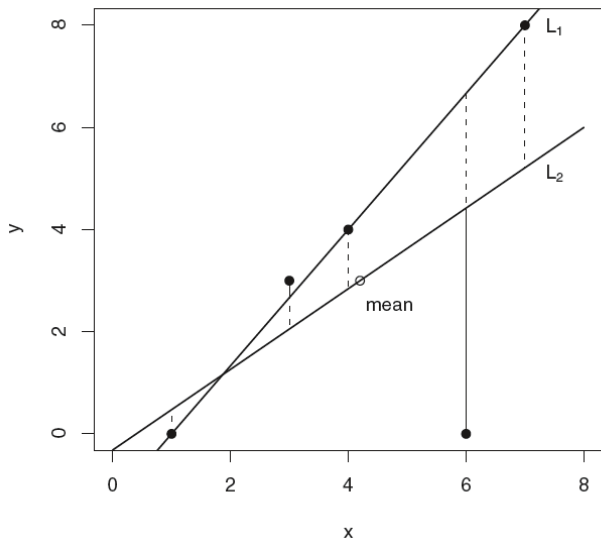
$$\min_{\mathbf{w}} \quad ||\mathbf{y} - \mathbf{X}\mathbf{w}||_1$$

Consider the 1D case:

Given a series of data points $y_1, \dots, y_N \in R$,

$$x_1 = \arg \min_x \sum_{i=1}^N (y_i - x)^2$$

$$x_2 = \arg \min_x \sum_{i=1}^N |y_i - x|$$



Solutions to LAD

- Linear programming
- ADMM
- IRLS (Iteratively reweighted least square)
-

ADMM for LAD

$$\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_1 \quad \Leftrightarrow \quad \min_{\mathbf{z}} \|\mathbf{z}\|_1 \quad \text{s.t.} \mathbf{X}\mathbf{w} - \mathbf{z} = \mathbf{y}$$

- Argumented Lagrangian:

$$L_{\rho}(\mathbf{w}, \mathbf{z}, \mathbf{t}) = \|\mathbf{z}\|_1 + \mathbf{t}^T(\mathbf{X}\mathbf{w} - \mathbf{z} - \mathbf{b}) + \frac{\rho}{2} \|\mathbf{X}\mathbf{w} - \mathbf{z} - \mathbf{y}\|^2$$

- ADMM procedure:

$$\mathbf{w}^{k+1} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T (\mathbf{z}^k + \mathbf{y} - \frac{1}{\rho} \mathbf{t}^k)$$

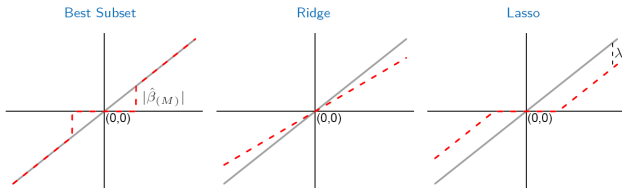
$$\mathbf{z}^{k+1} = S_{\frac{1}{\rho}}(\mathbf{X}\mathbf{w}^{k+1} - \mathbf{y} + \frac{1}{\rho} \mathbf{t}^k)$$

$$\mathbf{t}^{k+1} = \mathbf{t}^k + \rho(\mathbf{X}\mathbf{w}^{k+1} - \mathbf{z}^{k+1} - \mathbf{y})$$

Subset selection, Ridge Regression, LASSO

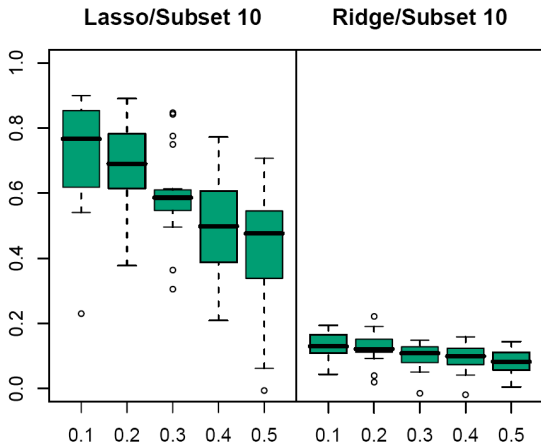
Consider all the features of \mathbf{X} are orthonormal, ($\mathbf{X}^T \mathbf{X} = \mathbf{I}$)

Estimator	Formula
Best subset (size M)	$\hat{\beta}_j \cdot I(\hat{\beta}_j \geq \hat{\beta}_{(M)})$
Ridge	$\hat{\beta}_j / (1 + \lambda)$
Lasso	$\text{sign}(\hat{\beta}_j)(\hat{\beta}_j - \lambda)_+$



"Bet on Sparsity"

Example: Suppose a design matrix \mathbf{X} with independent Gaussian variables ($N = 50, p = 300$) and the response variable is given by $\mathbf{y} = \mathbf{X}\beta + \epsilon$. (β has only 10 nonzero coefficients.)



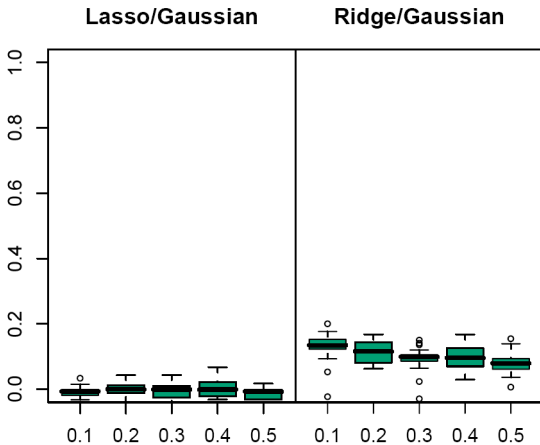
x -axis:

Noise-to-signal Ratio

y -axis:

Explained variance

All coefficients β are nonzero.



Friedman et al, 2004

Using a procedure that does well in sparse problems, since no procedure does well in dense problems.

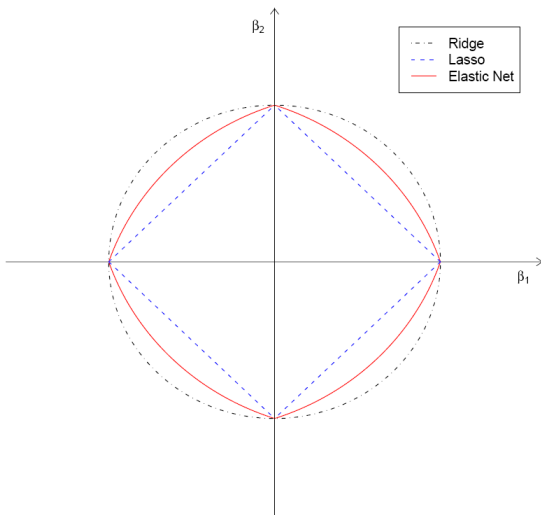
The limitations of LASSO

- If $p > N$, the LASSO selects at most n variables. The number of selected features is bounded by the number of samples.
- Grouped variables: LASSO fails to do grouped selection. It tends to select one variable from a group and ignore the others.

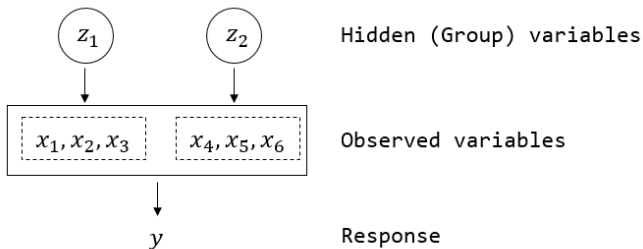
Elastic Net

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} ||\mathbf{y} - \mathbf{X}\mathbf{w}||^2 + \lambda_2 ||\mathbf{w}||^2 + \lambda_1 ||\mathbf{w}||_1$$

- The l_1 part of penalty generates a sparse model.
- The quadratic part of the penalty
 - Remove the limitation on the number of selected variables
 - Encourage grouping effect
 - Stabilize the l_1 regularization path



A simple illusion: elastic net vs. LASSO

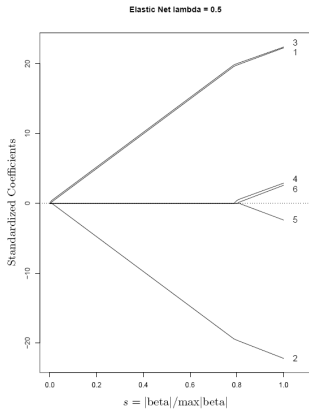
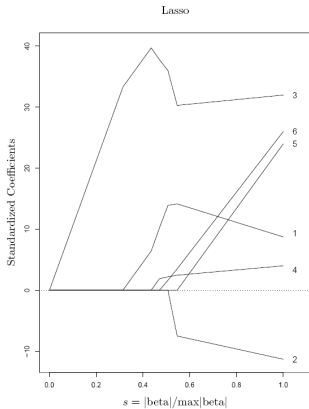


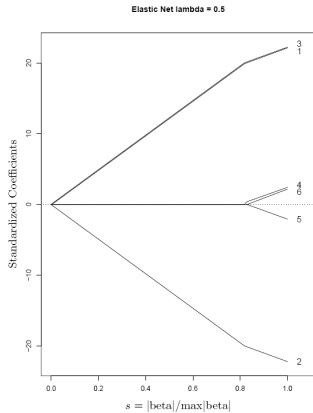
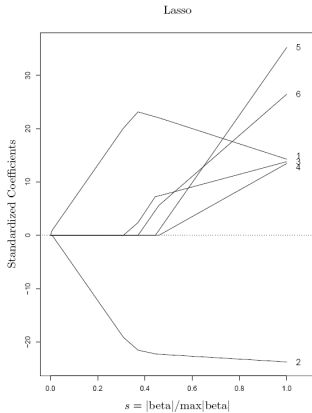
- Independent hidden factors: $\mathbf{z}_1 \sim U(0, 20)$, $\mathbf{z}_2 \sim U(0, 20)$
- Observed variables:

$$\mathbf{x}_1 = \mathbf{z}_1 + \epsilon_1, \mathbf{x}_2 = -\mathbf{z}_1 + \epsilon_2, \mathbf{x}_3 = \mathbf{z}_1 + \epsilon_3$$

$$\mathbf{x}_4 = \mathbf{z}_2 + \epsilon_4, \mathbf{x}_5 = -\mathbf{z}_2 + \epsilon_5, \mathbf{x}_6 = \mathbf{z}_2 + \epsilon_6$$

- Response variables: $\mathbf{y} = \mathbf{z}_1 + 0.2\mathbf{z}_2 + \mathcal{N}(0, 1)$





Effective Degree of Freedom

- Effective df describes the model complexity.
- df is very useful in estimating the prediction accuracy of the fitted model.

Effective df for linear regression methods

- Suppose $\hat{u} = S\mathbf{y}$ (linear smoothers), then $df(\hat{u}) = \text{Tr}(S)$
 - Simple least squares: $p + 1$ (number of features + 1)
 - Ridge regression:

$$df = \text{Tr}(\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T) = \text{Tr}(\mathbf{H}_\lambda) = \sum_{j=1}^p \frac{\sigma_j^2}{\sigma_j^2 + \lambda}$$

- Non-linear case: difficult
 - LASSO: the number of nonzero coefficients
 - Elastic net: $df = E[\hat{df}]$

$\hat{df} = \text{Tr}(\mathbf{H}_{\lambda_2}(\mathcal{A}))$ where \mathcal{A} is the active set

$$\mathbf{H}_{\lambda_2}(\mathcal{A}) = \mathbf{X}_{\mathcal{A}}(\mathbf{X}_{\mathcal{A}}^T\mathbf{X}_{\mathcal{A}} + \lambda_2\mathbf{I})^{-1}\mathbf{X}_{\mathcal{A}}^T$$

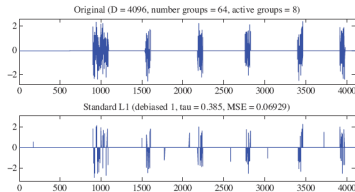
A Brief History of l_1 regularization

- LASSO for linear regression: in statistics (Tibshirani, 1995)
- Basis Pursuit: in signal processing (Chen, Donoho and Saunders, 1996)
- Extension to generalized linear models: e.g., logistic regression and so on.
- Structured sparsity: e.g., fused lasso, group lasso, elastic net, graphical lasso and so on.
- Compressive sensing: near exact recovery of sparse signals in very high dimensions (Donoho 2004, Candes and Tao 2005)
- Low-rank approximation: from vectors to matrices

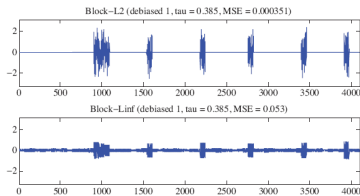
Group LASSO

$$J(\mathbf{w}) = \text{NLL}(\mathbf{w}) + \sum_{g=1}^G \lambda_g \|\mathbf{w}_g\|_2$$

- But we need to know the groups first.



(a)



(b)

Graph-regularized LASSO

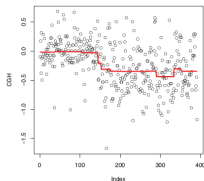
The penalty becomes:

$$\lambda_1 \|\beta\|_1 + \lambda_2 \beta^T \mathbf{L} \beta$$

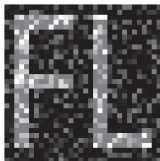
- $\mathbf{L} = \mathbf{I}$: elastic net
- \mathbf{L} can be a adjacency matrix of a graph

$$\beta^T \mathbf{L} \beta = \sum_{i,j} \mathbf{L}_{ij} \beta_i \beta_j$$

Fused LASSO



(a)



(b)



(c)

Basic idea: in addition to being sparse, neighboring coefficients are similar to each other.

$$\lambda_1 \|\mathbf{w}\|_1 + \lambda_2 \sum_{i=1}^{p-1} |w_{i+1} - w_i|$$

Graphical LASSO

Gaussian Graphical Model(Gaussian-Markov Random Field)

1. If all variables have a multivariate Gaussian distribution with
 - mean μ
 - covariance matrix Σ (positive definite)
 - concentration matrix $K = \Sigma^{-1}$

the ij th component of Σ^{-1} is zero \Rightarrow variables i and j are **conditionally independent**, given the other variables.

2. Undirected graph: no link means conditionally independent.
3. Parameter estimation and model selection:
estimating parameters and identifying zeros in K

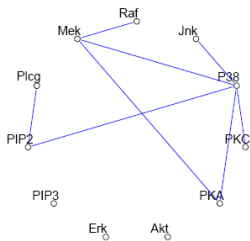
Lasso for covariance selection

$$-\log \det(K) + \text{Tr}(KS_n) + \lambda \sum_{i \neq j} |k_{ij}|$$

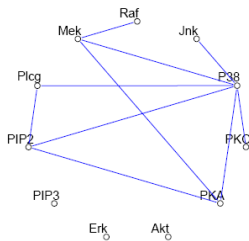
$$\min \quad -\log \det(K) + \text{Tr}(KS_n)$$

$$s.t. \quad \sum_{i \neq j} |k_{ij}| \leq t$$

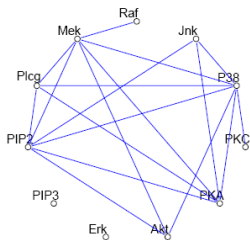
$\lambda = 36$



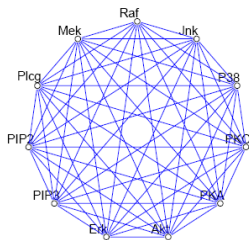
$\lambda = 27$



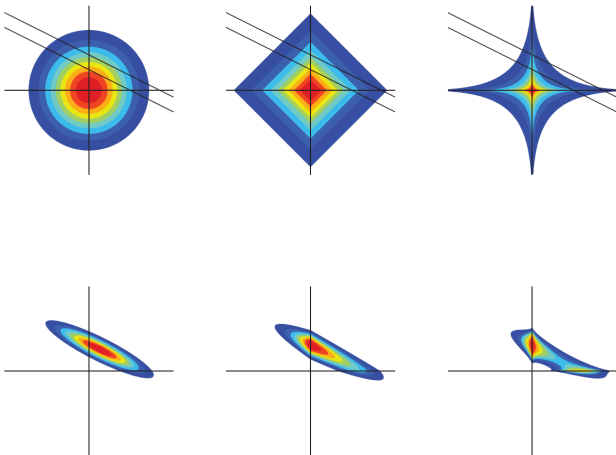
$\lambda = 7$



$\lambda = 0$



Non-convex Regularizers

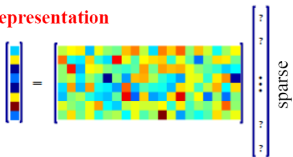


Sparsity and Low-rank

Two low-dimensional representations?

Sparse Representation

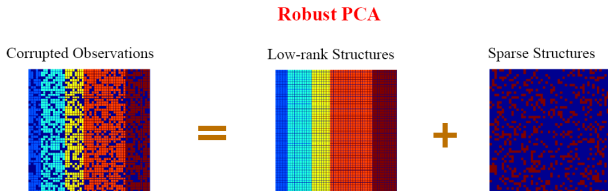
Underdetermined system

$$y = Ax$$


The diagram illustrates the equation $y = Ax$ for an underdetermined system. On the left, y is represented by a vertical vector of 12 colored squares (blue, yellow, red, green). To its right is an equals sign, followed by a matrix A of 12 rows and 16 columns, each cell containing a small colored square. To the right of the matrix is a vertical vector x of 16 elements, with the top 4 elements shown as colored squares and the rest as ellipses, followed by the word "sparse" written vertically.

Robust PCA

Corrupted Observations Low-rank Structures Sparse Structures



The diagram illustrates the Robust PCA equation. It shows three heatmaps: "Corrupted Observations" (a noisy heatmap with vertical bands of color), "Low-rank Structures" (a heatmap with distinct vertical bands of color), and "Sparse Structures" (a heatmap with sparse, scattered colored pixels on a dark background). These are connected by an equals sign and a plus sign, representing the equation: Corrupted Observations = Low-rank Structures + Sparse Structures.

Underdetermined system

Sparse representations reflect low-dimensional structure

$$\mathbf{y} = \begin{bmatrix} -4 \\ -5 \\ 3 \end{bmatrix} \quad \mathbf{A} = \begin{bmatrix} 1 & 4 & 1 & 1 & 6 \\ -2 & 1 & -4 & 2 & -3 \\ 3 & 3 & 2 & -2 & 1 \end{bmatrix}$$

1. Non-sparse

$$\mathbf{x} = \begin{bmatrix} 4 \\ -1 \\ 3 \\ 5 \\ -2 \end{bmatrix}$$

2. Sparse

$$\mathbf{x} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \\ -1 \end{bmatrix}$$

$$\begin{array}{ccc}
 \mathbf{y} & \approx & \mathbf{A} \mathbf{x} \\
 \begin{array}{c} \text{(Patches of) ...} \\ \text{input image} \end{array} & & \begin{array}{c} \text{Learned dictionary} \\ \mathbf{x} \text{ coefficients} \end{array}
 \end{array}$$

See [Elad+Bryt '08], [Horev et. Al., '12] ... Image: [Aharon+Elad '05]

Robust PCA

- Incomplete data
- Outliers
 - intra-sample outlier
 - sample outlier

PCA with Missing Entries

Incomplete data point x :

- x_i -entry is missing: x is known only up to a line in R^D :

$$\begin{aligned}x \in L &= \{[x_1, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_D]^T, x_i \in R\} \\&= \{x_{-i} + x_i e_i, x_i \in R\}\end{aligned}$$

- m missing entries: partition x into observed part and unobserved part, then x is known only up to a m -dimensional affine subspace:

$$x \in L = \left\{ \begin{bmatrix} 0 \\ x_O \end{bmatrix} + \begin{bmatrix} I_m \\ 0 \end{bmatrix} x_U, x_U \in R^m \right\}$$

Let $X \in R^{D \times N}$ be a matrix whose columns are drawn from a low-dimensional subspace of R^D of dimensions $d \ll D$. The observed subset of X is indexed by:

$$\Omega = \{(i, j) : x_{ij} \text{ is observed} \}$$

Define a map $\mathcal{P}_\Omega : R^{D \times N} \rightarrow R^{D \times N}$

$$\mathcal{P}_\Omega(X) = \begin{cases} x_{ij} & \text{if } (i, j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

It has been shown we may complete the missing entries in X by searching for a complete matrix $A \in R^{D \times N}$ that of low-rank and coincides with X in Ω .

$$\min_A \text{rank}(A) \quad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

Convex relaxation:

$$\min_A \|A\|_* \quad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

where $\|A\|_* = \sum \sigma_i(A)$ is the nuclear norm (the sum of singular values) which is the convex envelop of the rank function.

When the matrix is incoherent, the locations of known entries are sampled uniformly at random, and the number of known entries is sufficiently large, the minimizer to this problem is unique and equal to the matrix X for most matrices X .

Matrix Completion

movies

	2		1			4				5	
	5		4			?		1		3	
		3		5		2					
4			?			5		3		?	
		4		1	3				5		
			2				1	?			4
	1					5		5		4	
		2		?	5		?		4		
	3		3		1		5		2		1
	3				1				2		3
	4			5	1			3			
		3				3	?			5	
2	?		1		1						
		5			2	?		4		4	
	1		3		1	5		4		5	
1		2			4				5	?	

users

PCA with Corrupted Data

Some of the entries of the data points have been corrupted by gross errors:

$$x_{ij} = x_{ij}^0 + e_{ij} \text{ or } x_j = x_j^0 + e_j \text{ or } X = X^0 + E$$

- Detect and correct the errors:

$$\Omega = \{(i, j) : e_{ij} \neq 0\}$$

Robust PCA by Convex Optimization

The given data matrix X is generated as the sum of two matrices:

$$X = L_0 + E_0$$

- L_0 : the ideal low-rank matrix (principle components)
- E_0 : intra-sample outliers, which is sparse.

$$\min_{L, E} \text{rank}(L) + \lambda \|E\|_0 \quad \text{s.t. } X = L + E$$

1. $D \times N$ equations and $2D \times N$ unknowns
2. There're many trivial cases. For example, $x_{11} = 1, x_{ij} = 0$ (X is both low-rank and sparse).
3. Cost function: non-convex and non-differentiable.

Under certain conditions on L_0 and E_0 , the correct solution to decompose $X \rightarrow (L_0, E_0)$ can be found by solving the following convex optimization problem:

$$\min_{L,E} \|L\|_* + \lambda \|E\|_1 \quad \text{s.t. } X = L + E$$

- $\|L\|_* = \sum_i \sigma_i(L)$
- $\|E\|_1 = \sum_{i,j} |e_{ij}|$

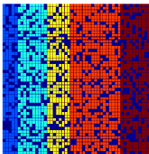
In some cases, there're also small dense noise in the data.

$$X = L + E + Z$$

where Z is a Gaussian matrix that models small Gaussian noise in the given data.

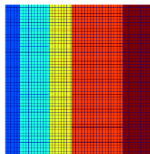
$$\min_{L, E} \|L\|_* + \lambda \|E\|_1 \quad \text{s.t.} \quad \|X - L - E\|_2^2 \leq \epsilon^2$$

Observation Matrix



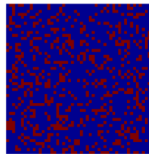
=

Low-rank Structures

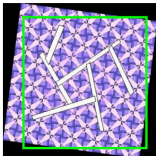


+

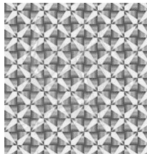
Sparse Component



Things can be difficult ...



$\circ \tau =$



+

