Lecture 14: Variable Selection - Beyond LASSO

Hao Helen Zhang

Extension of LASSO

- To achieve oracle properties,
  - L_q penalty with $0 < q < 1$, SCAD penalty (Fan and Li 2001; Zhang et al. 2007).
  - Adaptive LASSO (Zou 2006; Zhang and Lu 2007; Wang et al. 2007)

$$J_\lambda(\boldsymbol{\beta}) = \lambda \sum_{j=1}^{p} w_j |\beta_j|.$$

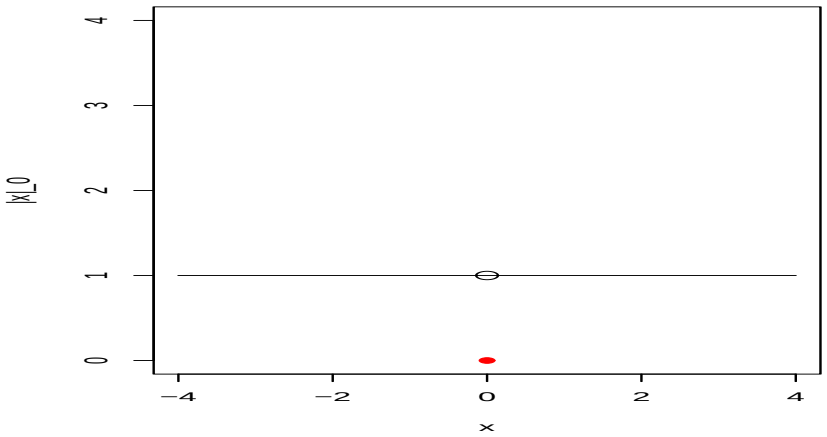
  - Nonnegative garotte (Breiman, 1995)
- To handle correlated predictors,
  - Elastic net (Zou and Hastie 2005), adaptive elastic net

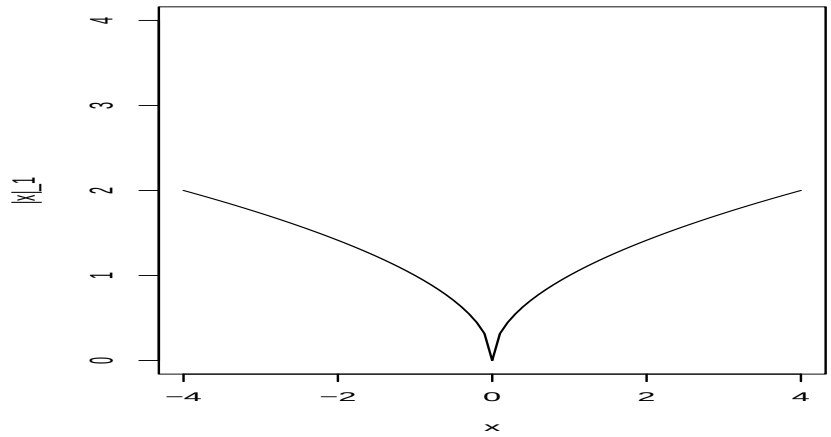
$$J_\lambda = \sum_{j=1}^{p} [\lambda|\beta_j| + (1-\lambda)\beta_j^2], \lambda \in [0,1].$$

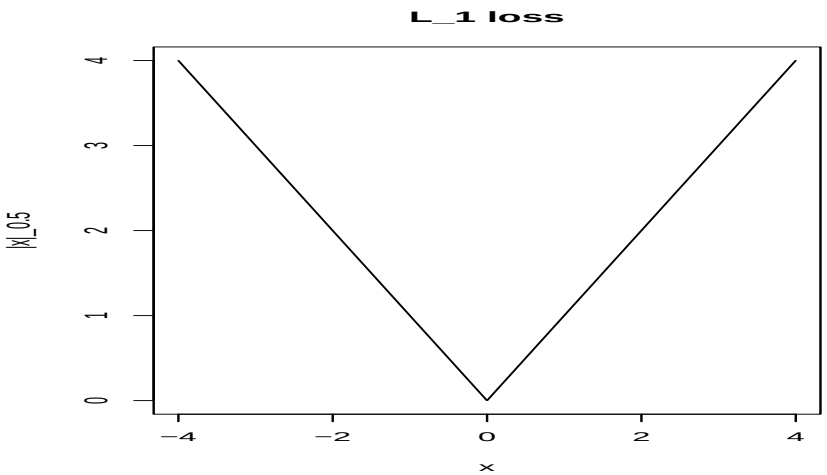
  - OSCAR (Bondell and Reich, 2006)
  - Adaptive elastic net (Zou and Zhang 2009)

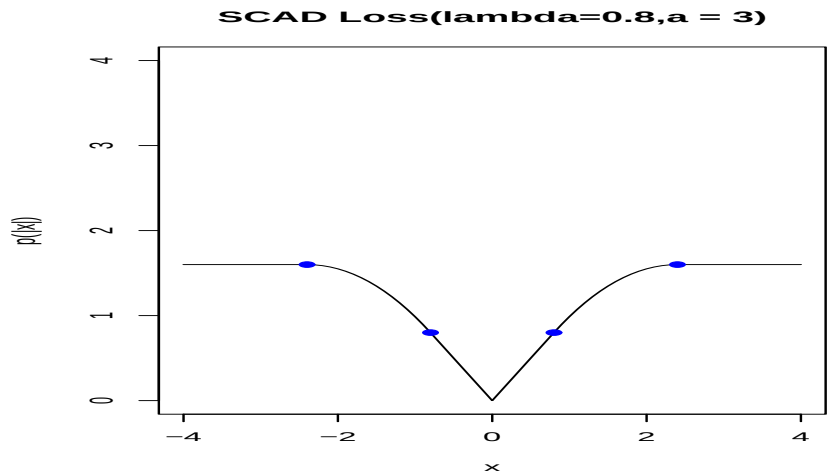
- For group variable selection,
  - group LASSO (Yuan and Lin 2006)
  - supnorm penalty (Zhang et al. 2008)
- For nonparametric (nonlinear) models
  - LBP (Zhang et al. 2004),
  - COSSO (Lin and Zhang 2006; Zhang 2006; Zhang and Lin 2006)
  - group lasso based methods

Various Penalty Functions









About L_q Penalty Methods

- $q = 0$ corresponds variable selection, the penalty is the number of nonzero coefficients or model size
- For $q \leq 1$, more weights imposed on the coordinate directions
- $q = 1$ is the smallest q such that the constraint region is convex
- q can be fractions. Maybe estimate q from the data?

Motivations:

- LASSO imposes the same penalty on the coefficients
- Ideally, small penalty should be applied to large coefficients (for small bias), while large penalty applied to small coefficients (for better sparsity)
- LASSO is not an oracle procedure

Methods with oracle properties include:

- SCAD, weighted lasso, adaptive lasso, nonnegative garrote, etc.

What is A Good Penalty

A good penalty function should lead to the estimator

- "nearly" unbiased
- sparsity: threshold small coefficients to zero (reduced model complexity)
- continuous in data, robust against small perturbation)

How to make these happen? (Fan and Li 2002)

- sufficient condition for unbiasedness: $J'(|\beta|) = 0$ for large $|\beta|$ (penalty bounded by a constant).
- necessary and sufficient condition for sparsity: $J(|\beta|)$ is singular at 0

Most of the oracle penalties are concave, suffering from multiple local minima numerically.

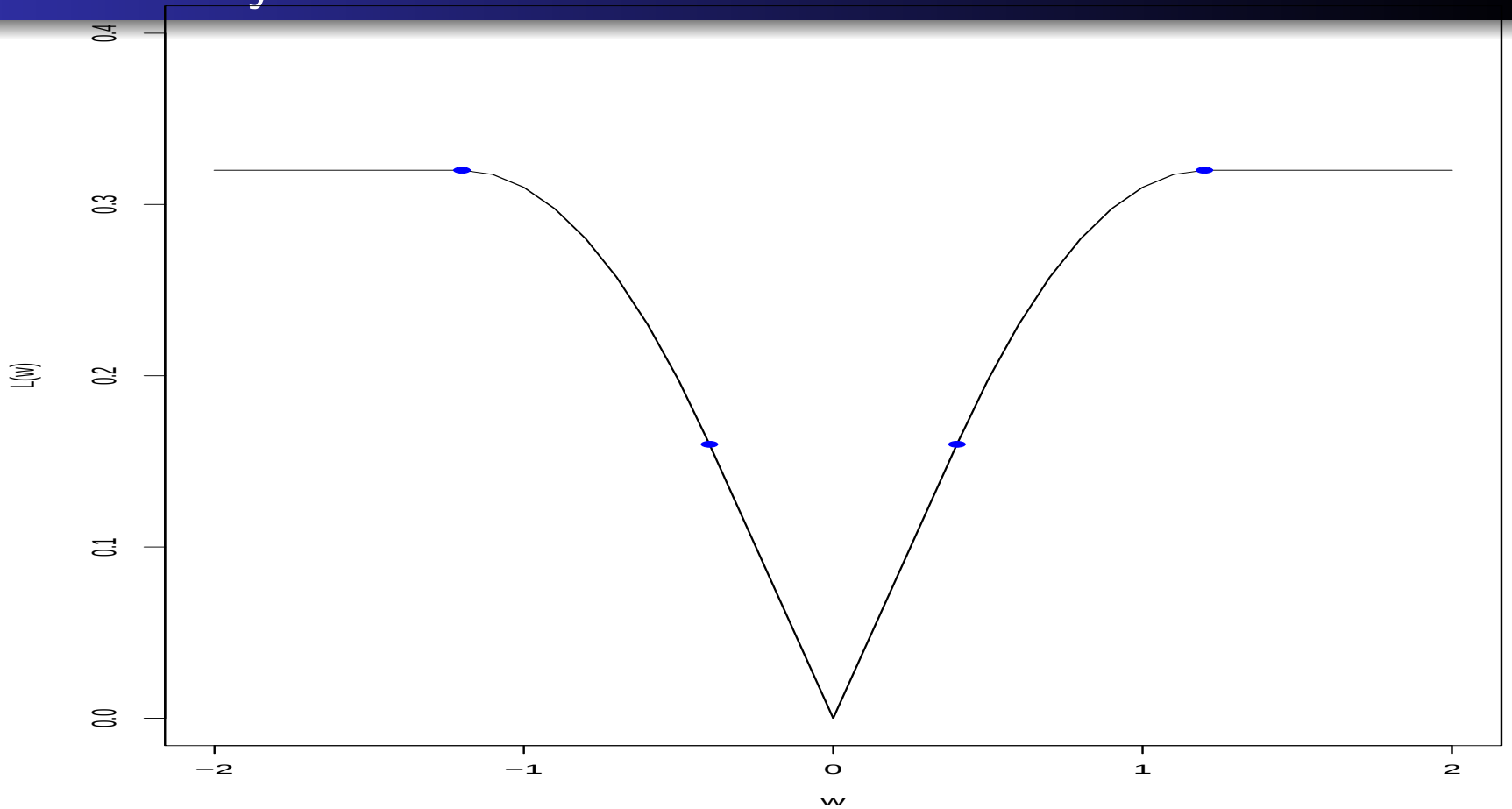
Smoothly Clipped Absolute Deviation Penalty

$$p_\lambda(\beta) = \begin{cases} \lambda|\beta| & \text{if} \quad |\beta| \le \lambda, \\ -\frac{(|\beta|^2 - 2a\lambda|\beta| + \lambda^2)}{2(a-1)} & \text{if} \quad \lambda < |\beta| \le a\lambda, \\ \frac{(a+1)\lambda^2}{2} & \text{if} \quad |\beta| > a\lambda, \end{cases}$$

where $a > 2$ and $\lambda > 0$ are tuning parameters.

- A quadratic spline function with two knots at λ and $a\lambda$.
- singular at the origin; continuous first-order derivative.
- constant penalty on large coefficients; not convex

SCAD Penalty



Computational Issues

The SCAD penalty is not convex

- Local Quadratic Approximation (LQA; Fan and Li, 2001)
- Local Linear Approximation (LLA; Zou and Li, 2008)

Nonnegative Garotte

Breiman (1995): solve

$$\begin{aligned} \min_{c_1,\cdots,c_p} \quad & \|\mathbf{y} - \sum_{j=1}^{p} \boldsymbol{x}_j \hat{\beta}_j^{ols} c_j\|^2 + \lambda_n \sum_{j=1}^{p} c_j \\ \text{subject to} \quad & c_j \geq 0, \quad j = 1, \cdots, p, \end{aligned}$$

and denote the solution as $\hat{c}_j$'s. The garotte solution

$$\hat{\beta}_j^{garotte} = \hat{c}_j \hat{\beta}_j^{ols}, \quad j = 1, \cdots, p.$$

- A sufficiently large λ_n shrinks some c_j to exact 0 (sparsity)
- The nonnegative garotte estimate is selection consistent. See also Yuan and Lin (2005).

Adaptive LASSO

Proposed by Zou (2006), Zhang and Lu (2007), Wang et al. (2007)

$$\widehat{\boldsymbol{\beta}}^{alasso} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda \sum_{j=1}^{p} w_j |\beta_j|,$$

where $w_j \geq 0$ for all j.

Adaptive choices of weights

- The weights are data-dependent and adaptively chosen from the data.
- Large coefficients receive small weights (and small penalties)
- Small coefficients receive large weights (and penalties)

How to Compute Weights

Need some good initial estimates $\tilde{\beta}_j$'s.

- In general, we require $\tilde{\beta}_j$'s to be root-n consistent
- The weights are calculated as

$$w_j = \frac{1}{|\tilde{\beta}_j|^\gamma}, \quad j = 1, \cdots, p.$$

  $\gamma > 0$ is another tuning parameter.

- For example, $\tilde{\beta}_j$'s can be chosen as $\widehat{\boldsymbol{\beta}}^{ols}$.
- Alternatively, if collinearity is a concern, one can use $\widehat{\boldsymbol{\beta}}^{ridge}$.

As the sample size n grows,

- the weights for zero-coefficient predictors get inflated (to ∞);
- the weights for nonzerocoefficient predictors converge to a finite constant.

Oracle Properties of Adaptive LASSO

Assume the first q variables are important. And

$$\frac{1}{n} X_n^T X_n \to C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix},$$

where C is a positive definite matrix.

Assume the initial estimators $\tilde{\beta}$ are root-n consistent. Suppose $\lambda_n/\sqrt{n} \to 0$ and $\lambda_n n^{(\gamma-1)/2} \to \infty$.

- Consistency in variable selection:

$$\lim_n P(\hat{\mathcal{A}}_n = \mathcal{A}_0) = 1,$$

  where $\hat{\mathcal{A}}_n = \{j : \hat{\beta}_{j,n}^{alasso} \neq 0\}$.
- Asymptotic normality:

$$\sqrt{n}(\widehat{\beta}_n^{alasso} - \beta_{\mathcal{A}_0,0}) \to_d N(\mathbf{0}, \sigma^2 C_{11}^{-1}).$$

Additional Comments

- The oracle properties are closely related to the super-efficiency phenomenon (Lehmann and Casella 1998).
- Adaptive lasso simultaneously unbiasedly (asymptotically) estimate large coefficients and small threshold estimates.
- The adaptive lasso uses the same rational behind SCAD.

Computational Advantages of Adaptive LASSO

The adaptive lasso

- has continuous solutions (continuous in data)
- is a convex optimization problem
- can be solved by the same efficient algorithm LARS to obtain the entire solution path.

By contrast,

- The bridge regression (Frank and Friedman 1993) uses the L_q penalty.
- The bridge with $0 < q < 1$ has the oracle properties (Knight and Fu, 2000), but the bridge with $q < 1$ solution is not continuous. (See the picture)

Fit Adaptive LASSO using LARS algorithm

Algorithm:

1. Define $\boldsymbol{x}_j^{**} = \boldsymbol{x}_j / w_j$ for $j = 1, \cdots, p$.
2. Solve the lasso problem for all λ_n,

$$\widehat{\boldsymbol{\beta}}^{**} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \sum_{j=1}^{p} \boldsymbol{x}_j^{**} \beta_j\|^2 + \lambda_n \sum_{j=1}^{p} |\beta_j|.$$

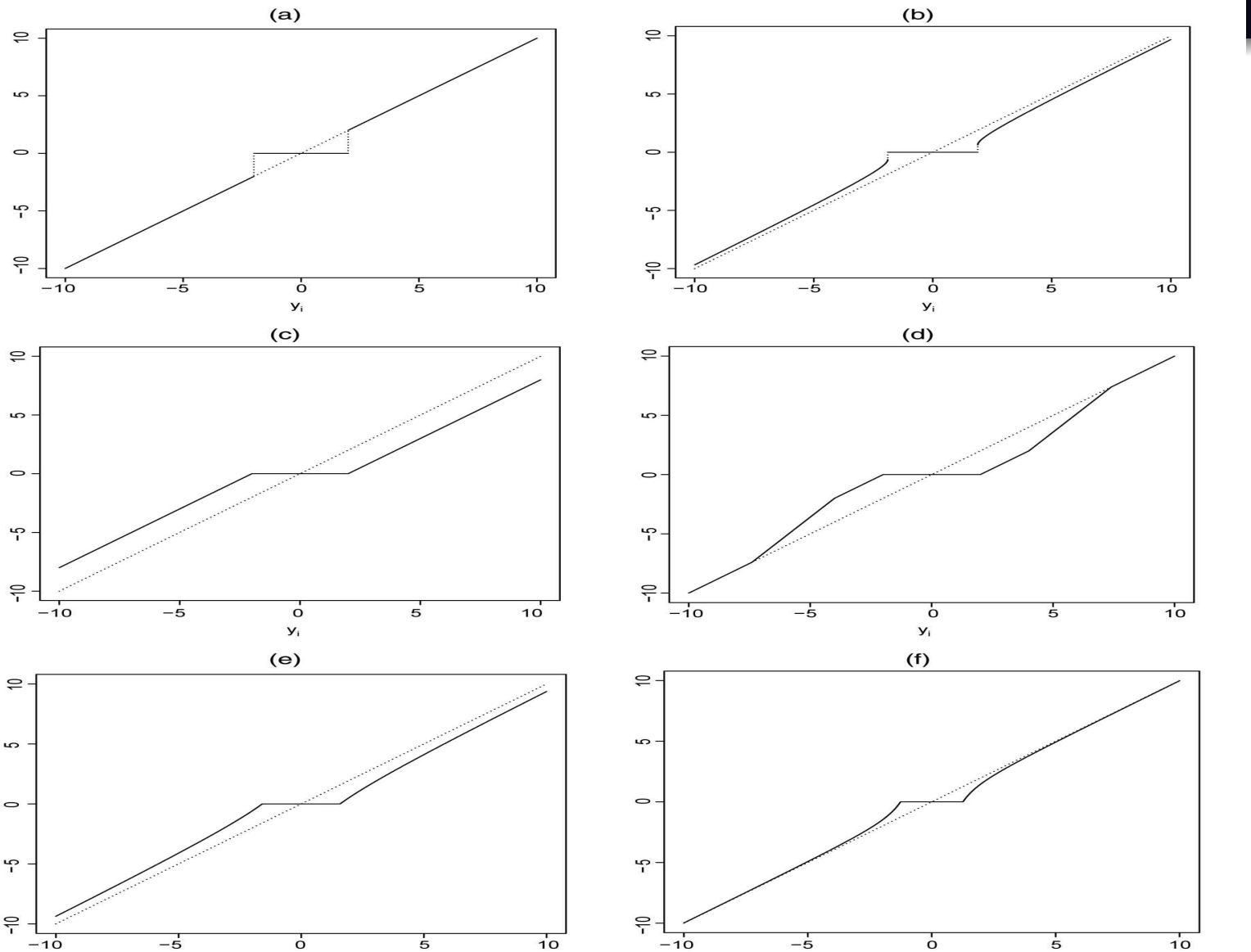
3. Output

$$\hat{\beta}_j^{alasso} = \hat{\beta}_j^{**} / w_j, \quad j = 1, \cdots, p.$$

The LARS algorithm is used to compute the entire solution path of the lasso in Step 2.

- The computational cost is of order $O(np^2)$, which is the same order of computation of a single OLS fit.

Adaptive Solution Plot



Connection of Nonnegative Garotte and Adaptive LASSO

Let $\gamma = 1$. The garotte can be reformulated as

$$\begin{aligned} \min_{\beta} \quad & \|\mathbf{y} - \sum_{j=1}^{p} \boldsymbol{x}_j \beta_j\|^2 + \lambda_n \sum_{j=1}^{p} \frac{|\beta_j|}{|\hat{\beta}_j^{ols}|} \\ \text{subject to} \quad & \beta_j \hat{\beta}_j^{ols} \geq 0, \quad j = 1, \cdots, p, \end{aligned}$$

- The nonnegative garotte can be regarded as the adaptive lasso ($\gamma = 1$) with additional sign constraint.
- Zou (2006) showed that, if $\lambda \to \infty$ and $\lambda_n/\sqrt{n} \to 0$, then then the nonnegative garotte is consistent for variable selection consistent.

Adaptive LASSO for Generalized Linear Models (GLMs)

Assume the data has the density of the form

$$f(y|\mathbf{x},\theta) = h(y)\exp(y\theta - \phi(\theta)), \quad \theta = \boldsymbol{x}^T\boldsymbol{\beta},$$

which belongs the exponential family with canonical parameter θ. The adaptive lasso solves the penalized log-likelihood

$$\widehat{\boldsymbol{\beta}}_n^{alasso} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \Big(-y_i(\mathbf{x}_i^T\boldsymbol{\beta}) + \phi(\mathbf{x}_i^T\boldsymbol{\beta})\Big) + \lambda_n \sum_{j=1}^{p} w_j|\beta_j|,$$

where

$$w_j = 1/|\hat{\beta}_j^{mle}|^\gamma, \quad j = 1, \cdots, p,$$

- $\hat{\beta}_j^{mle}$ is the maximum likelihood estimator.

Adaptive LASSO for GLMs

For logistic regression, the adaptive lasso solves

$$\widehat{\boldsymbol{\beta}}_n^{alasso} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \Big(-y_i(\mathbf{x}_i^T \boldsymbol{\beta}) + \log(1 + e^{\mathbf{x}_i^T \boldsymbol{\beta}}) \Big) + \lambda_n \sum_{j=1}^{p} w_j |\beta_j|.$$

For Poisson log-linear regression models, the adaptive lasso solves

$$\widehat{\boldsymbol{\beta}}_n^{alasso} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \Big(-y_i(\mathbf{x}_i^T \boldsymbol{\beta}) + \exp(\mathbf{x}_i^T \boldsymbol{\beta}) \Big) + \lambda_n \sum_{j=1}^{p} w_j |\beta_j|.$$

- Zou (2006) show that, under some mild regularity conditions, $\widehat{\boldsymbol{\beta}}_n^{alasso}$ enjoys oracle properties if λ_n is chosen appropriately.
- The limiting covariance matrix of $\widehat{\boldsymbol{\beta}}_n^{alasso}$ is I_{11}, the Fisher information with the true sub-model known.

Adaptive LASSO for High Dimensional Linear Regression

Huang, Ma, and Zhang (2008) considers the case $p \to \infty$ as $n \to \infty$. If a reasonable initial estimator is available, they show

- Under appropriate conditions, $\widehat{\boldsymbol{\beta}}^{alasso}$ is consistent in variable selection, and the nonzero coefficient estimators have the same asymptotic distribution they would have if the true model were known (oracle properties)
- Under a partial orthogonality condition, i.e., unimportant covariates are weakly correlated with important covariates, marginal regression can be used to obtain the initial estimator and assures the oracle property of $\widehat{\boldsymbol{\beta}}^{alasso}$ even when $p > n$.
- For example, margin regression initial estimates work in the essentially uncorrelated case

$$|\text{corr}(X_j, X_k)| \leq \rho_n = O(n^{-1/2}), \quad \forall j \in \mathcal{A}_0, k \in \mathcal{A}_0^c.$$

Correlated Predictors

Motivations:

- If $p > n$ or $p \gg n$, the lasso can select at most n variables before it saturates, because of the nature of the convex optimization problem.
- If there exist high pairwise correlations among predictors, lasso tends to select only one variable from the group and does not care which one is selected.
- For usual $n > p$ situations, if there are high correlations between predictors, it has been empirically observed that the prediction performance of the lasso is dominated by ridge regression (Tibshirani, 1996).

Methods: elastic net, OSCAR, adaptive elastic net

Motivations of Elastic Net

A typical microarray data set has many thousands of predictors (genes) and often fewer than 100 samples.

- Genes sharing the same biological 'pathway' tend to have high correlations between them (Segal and Conklin, 2003).
- The genes belonging to a same pathway form a group.

The ideal gene selection method should do two things:

- eliminate the trivial genes
- automatically include whole groups into the model once one gene among them is selected ('grouped selection').

The lasso lacks the ability to reveal the grouping information.

Elastic net

Zou and Hastie (2005) combine the lasso and the ridge

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda_2\|\boldsymbol{\beta}\|^2 + \lambda_1\|\boldsymbol{\beta}\|_1.$$

The l_1 part of the penalty generates a sparse model.

The quadratic part of the penalty

- removes the limitation on the number of selected variables;
- encourages grouping effect;
- stabilizes l_1 regularization path.

The new estimator is called **Elastic Net**. The solution is

$$\widehat{\boldsymbol{\beta}}^{enet} = (1 + \lambda_2)\widehat{\boldsymbol{\beta}},$$

which removes the double shrinkage effect.

Elastic Net Regularization

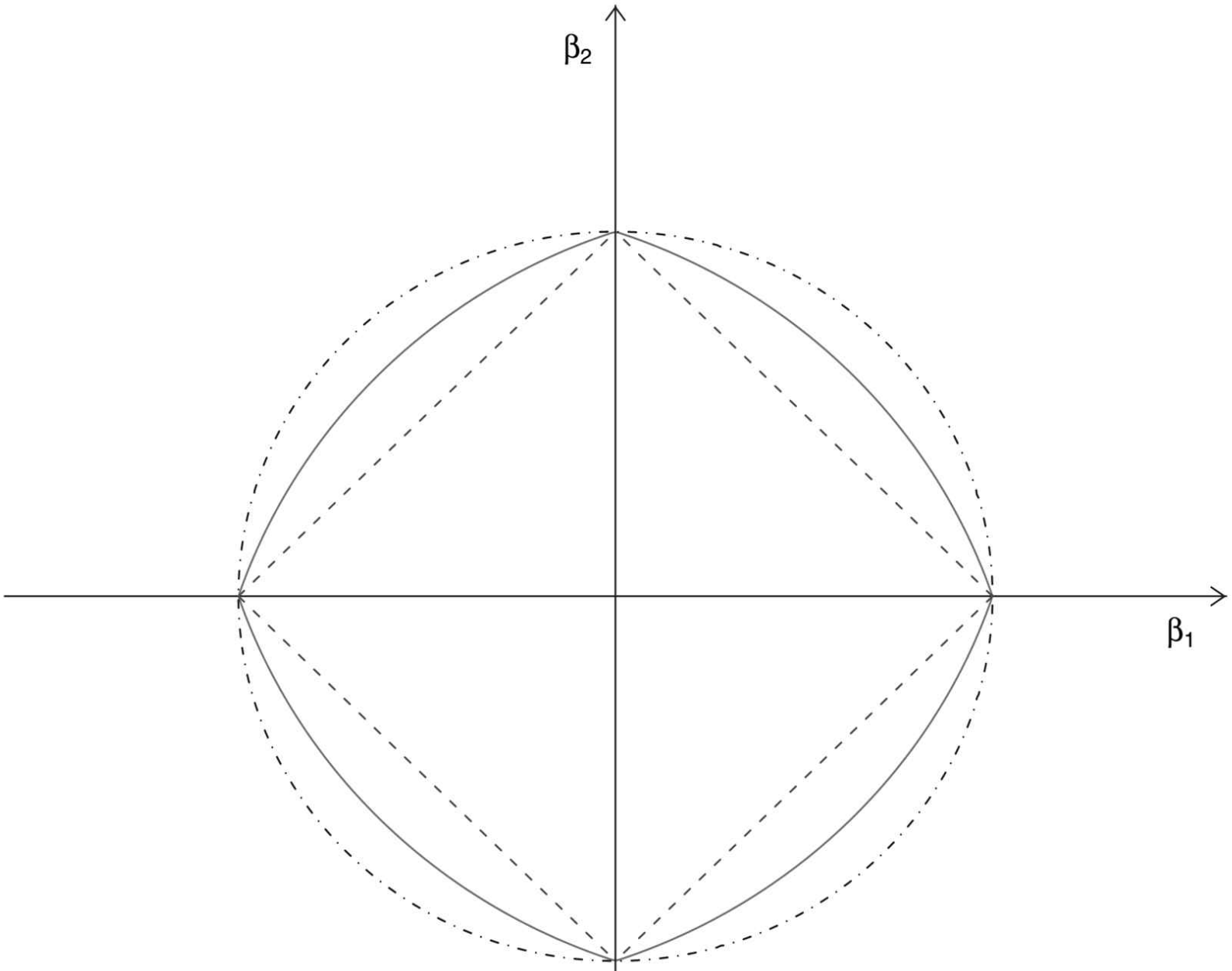
The elastic net penalty function

- has singularity at the vertex (necessary for sparsity)
- has strict convex edges (encouraging grouping)

Properties: the elastic net solution

- simultaneously does automatic variable selection and continuous shrinkage
- can select groups of correlated variables, retaining 'all the big fish'.

The elastic net often outperforms the lasso in terms of prediction accuracy in numerical results.


β2
β1

Strictly Convex Penalty vs LASSO penalty

Consider

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda J(\boldsymbol{\beta}),$$

where $J(\boldsymbol{\beta}) > 0$ for $\boldsymbol{\beta} \neq \mathbf{0}$.

Assume $\mathbf{x}_i = \mathbf{x}_j, i, , j \in \{1, \cdots, p\}$.

- If J is strictly convex, then $\hat{\beta}_i = \hat{\beta}_j$ for any $\lambda > 0$.
- If $J(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_1$, then $\hat{\beta}_i \hat{\beta}_j > 0$, and $\widehat{\boldsymbol{\beta}}^*$ is another minimzer, where

$$\hat{\beta}_k^* = \begin{cases} \hat{\beta}_k & \text{if } k \neq i, k \neq j, \\ (\hat{\beta}_i + \hat{\beta}_j) \cdot s & \text{if } k = i, \\ (\hat{\beta}_i + \hat{\beta}_j) \cdot (1 - s) & \text{if } k = j, \end{cases}$$

for any $s \in [0, 1]$.

Comments

There is a clear distinction between strictly convex penalty functions and the lasso penalty.

- Strict convexity guarantees the grouping effect in the extreme situation with identical predictors.
- In contrast the lasso does not even have a unique solution.
- The elastic net penalty with $\lambda_2 > 0$ is strictly convex, thus enjoying the grouping effect.

Results on the Grouping Effect

Given data $(\mathbf{y}, X)$ and parameters (λ_1, λ_2), the response $\mathbf{y}$ is centred and the predictors X are standardized. Let $\widehat{\boldsymbol{\beta}}^{enet}(\lambda_1, \lambda_2)$ be the elastic net estimate. Suppose $\hat{\beta}_i^{enet}(\lambda_1, \lambda_2)\hat{\beta}_j^{enet}(\lambda_1, \lambda_2) > 0$, then

$$\frac{1}{\|\mathbf{y}\|_1}|\hat{\beta}_i^{enet}(\lambda_1, \lambda_2) - \hat{\beta}_j^{enet}(\lambda_1, \lambda_2)| < \frac{\sqrt{2}}{\lambda_2}\sqrt{1-\rho_{ij}},$$

where $\rho_{ij} = \mathbf{x}_i^T \mathbf{x}_j$ is the sample correlation.

- The closer to 1 ρ_{ij} is, the more grouping in the solution.
- The larger λ_2 is, the more grouping in the solution.

Effective Degrees of Freedom

- Effective df describes the model complexity.
- df is very useful in estimating the prediction accuracy of the fitted model.
- df is well studied for linear smoothers: $\hat{\mathbf{y}} = S\mathbf{y}$, and

$$df(\hat{\mathbf{y}}) = tr(S).$$

For the l_1 related methods, the non-linear nature makes the analysis difficult.

Elastic net: degrees of freedom

For the elastic net, an unbiased estimate for df is

$$\widehat{df}(\hat{\beta}^{enet}) = \text{Tr}(H_{\lambda_2}(\mathcal{A})),$$

where $\mathcal{A} = \{j : \hat{\beta}_j^{enet} \neq 0, j = 1, \cdots, p\}$ is the active set (containing all the selected variables), and

$$H_{\lambda_2}(\mathcal{A}) = X_{\mathcal{A}}(X_{\mathcal{A}}^T X_{\mathcal{A}} + \lambda_2 I)^{-1} X_{\mathcal{A}}^T.$$

- Special case for the lasso: $\lambda_2 = 0$ and

$$\widehat{df}(\hat{\beta}^{lasso}) = \text{the number of nonzero coefficients in } \hat{\beta}^{lasso}.$$

Computing and Tuning Elastic Net

Solution path algorithm:

- The elastic net solution path is piecewise linear.
- Given a fixed λ_2, a stage-wise algorithm called LARS-EN efficiently solves the entire elastic net solution path.
- The computational effort is same as that of a single OLS fit.
- R package: elasticnet

Two-dimensional cross validation

Adaptive Elastic Net

Zou and Zhang (2009) propose

$$\widehat{\boldsymbol{\beta}}^{aenet} = (1+\lambda_2)\arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda_2\|\boldsymbol{\beta}\|^2 + \lambda_1 \sum_{j=1}^{p} w_j|\beta_j|,$$

where $w_j = (|\hat{\beta}_j^{enet}|)^{-\gamma}$ for $j = 1, \cdots, p$ and $\gamma > 0$ is a constant.

Main results:

- Under weak regularity conditions, the adaptive elastic-net has the oracle property.
- Numerically, the adaptive elastic-net deals with the collinearity problem better than the other oracle-like methods and has better finite sample performance.

Motivation of Group Lasso

The problem of group selection arises naturally in many practical situations. For example, In multifactor ANOVA,

- each factor has several levels and can be expressed through a group of dummy variables;
- the goal is to select important main effects and interactions for accurate prediction, which amounts to the selection of groups of derived input variables.

Variable selection amounts to the selection of important factors (groups of variables) rather than individual derived variables.

Another Motivation: Nonparametric Regression

Additive Regression Models

- Each component in the additive model may be expressed as a linear combination of a number of basis functions of the original measured variable.
- The selection of important measured variables corresponds to the selection of groups of basis functions.

Variable selection amount to the selection of groups of basis functions (corresponding to a same variable).

Group Selection Setup

Consider the general regression problem with J factors:

$$\mathbf{y} = \sum_{j=1}^{J} X_j \beta_j + \epsilon,$$

- $\mathbf{y}$ is an $n \times 1$ vector
- $\epsilon \sim N(0, \sigma^2 I)$
- X_j is an $n \times p_j$ matrix corresponding to the jth factor
- β_j is a coefficient vector of size $p_j, j = 1, \ldots, J$.

Both $\mathbf{y}$ and X_j's are centered. Each X_j is orthonormalized, i.e.,

$$X_j^T X_j = I_{p_j}, \quad j = 1, ..., J,$$

which can be done through GramSchmidt orthonormalization.
Denoting $X = (X_1, X_2, \ldots, X_J)$ and $\beta = (\beta_1^T, \beta_2^T, \ldots, \beta_J^T)^T$.

Group LASSO

Yuan and Lin (2006) propose

$$\widehat{\beta}^{glasso} = \arg\min_{\beta} \|\mathbf{y} - \sum_{j=1}^{J} X_j \beta_j\|^2 + \lambda \sum_{j=1}^{J} \sqrt{p_j} \|\beta_j\|,$$

where

- $\lambda \geq 0$ is a tuning parameter.
- $\|\beta_j\| = \sqrt{\beta_j^T \beta_j}$ is the empirical L_2 norm.
- The group lasso penalty encourages sparsity at the factor level.
- When $p_1 = \cdots = p_J = 1$, the group lasso reduces to the lasso.

Degree of Freedom for Group LASSO

$$\widehat{df} = \sum_{j=1}^{J} I(\|\widehat{\beta}_j^{glasso}\| > 0) + \sum_{j=1}^{J} \frac{\|\widehat{\beta}_j^{glasso}\|}{\|\widehat{\beta}_j^{ols}\|}(p_j - 1).$$

- $\lambda \geq 0$ is a tuning parameter.
- The group lasso penalty encourages sparsity at the factor level.
- When $p_1 = \cdots = p_J = 1$, the group lasso reduces to the lasso.

Motivation of Adaptive Group LASSO

The group lasso suffers estimation inefficiency and selection inconsistency. To remedy this, Wang and Leng (2006) propose

$$\widehat{\boldsymbol{\beta}}^{aglasso} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \sum_{j=1}^{J} X_j \boldsymbol{\beta}_j\|^2 + \lambda \sum_{j=1}^{J} w_j \|\boldsymbol{\beta}_j\|,$$

- The weight $w_j = \|\tilde{\boldsymbol{\beta}}_j\|^{-\gamma}$ for all $j = 1, \cdots, J$.
- The adaptive group LASSO has oracle properties. In other words, $\widehat{\boldsymbol{\beta}}^{aglasso}$ is root-n consistency, model selection consistency, asymptotically normal and efficient.

Function ANOVA Decomposition

Similar to classical ANOVA, any multivariate function f is decomposed as

$$f(\boldsymbol{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k) + \cdots + f_{1\cdots p}(x_1, \cdots, x_p)$$

- Side conditions guarantee uniqueness of decomposition (Wahba 1990, Gu 2002)

Product Domain

Consider a multivariate function $f(\boldsymbol{x}) = f(x_1, \ldots, x_p)$.

- Let $\mathcal{X}_j$ be the domain for x_j, i.e., $x_j \in \mathcal{X}_j$
- $\mathcal{X} = \prod_{j=1}^{p} \mathcal{X}_j$ is the product domain for $\mathbf{x} = (x_1, \ldots, x_p)$

Examples:

- continuous domain: $\mathcal{X}_j = [0, 1]$, and $\mathcal{X} = [0, 1]^p$.
- discrete domain: $\mathcal{X}_j = \{1, 2, \cdots, K\}$

Averaging Operator: A_j

A_j is the average operator on $\mathcal{X}_j$ that averages out x_j from the active argument list

- A_j satisfies $A_j^2 = A_j$.
- A_j is constant on $\mathcal{X}_j$, but not necessarily an overall constant function

Examples: $p = 2$.

- Example 1. $\mathcal{X}_1 = \{1, \ldots, K_1\}$ and $\mathcal{X}_2 = \{1, \ldots, K_2\}$
  - $A_1 = f(1, x_2), A_2 = f(x_1, 1)$.
  - $A_j = \sum_{x_j=1}^{K_j} f(x_1, x_2)/K_j$ for $j = 1, 2$
- Example 2: $\mathcal{X}_1 = \mathcal{X}_2 = [0, 1], \mathcal{X} = [0, 1]^2$
  - $A_1 = f(0, x_2), A_2 = f(x_1, 0)$.
  - $A_j = \int_0^1 f(x_1, x_2) dx_j$ for $j = 1, 2$

Multiway ANOVA Decomposition

$$
\begin{aligned}
f(\boldsymbol{x}) &= \{\prod_{j=1}^{p}(I - A_j + A_j)\}f = \sum_{\mathcal{S}}\{\prod_{j\in\mathcal{S}}(I - A_j)\prod_{j\notin\mathcal{S}} A_j\}f \\
&= \sum_{\mathcal{S}} f_{\mathcal{S}} \\
&= \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k) + \cdots + f_{1\cdots p}(x_1, \cdots, x_p)
\end{aligned}
$$

where

- $\mathcal{S} \in \{1, \ldots, p\}$ enlists the active arguments in $f_{\mathcal{S}}$
- the summation is over all of the 2^p subsets of $\{1, \ldots, p\}$.

ANOVA Interpretation

- $\beta_0 = \prod_{j=1}^{p} A_j(f)$ is a constant (overall mean).
- $f_j = f_j = (I - A_j) \sum_{k \neq j} A_k(f)$ is the x_j main effect.
- $f_{jk} = f_{j,k} = (I - A_j)(I - A_k) \sum_{l \neq k,j} A_l(f)$ is x_j-x_k interaction

Side conditions:

$$A_j f_{\mathcal{S}} = 0, \quad \forall j \in \mathcal{S}.$$

Example: $p = 2$.

$$\begin{aligned}
\beta_0 &= A_1 A_2 f, \\
f_1 &= (I - A_1) A_2 f = A_2 f - A_1 A_2 f = A_2 f - \beta_0, \\
f_2 &= (I - A_2) A_1 f = A_1 f - A_1 A_2 f = A_1 f - \beta_0, \\
f_{12} &= f(x_1, x_2) - f_1(x_1) - f_2(x_2) + \beta_0.
\end{aligned}$$

Discrete Domain Example: $p = 2$

Domain $\mathcal{X}_1 = \{1, \ldots, K_1\}$ and $\mathcal{X}_2 = \{1, \ldots, K_2\}$.

- Example: $A_1 = f(1, x_2)$ and $A_2 = f(x_1, 1)$.
  - $\beta_0 = A_1 A_2 f = f(1, 1)$
  - $f_1 = (I - A_1)A_2 f = f(x_1, 1) - f(1, 1)$
  - $f_2 = (I - A_2)A_1 f = f(1, x_2) - f(1, 1)$
  - $f_{12} = (I - A_1)(I - A_2)f = f(x_1, x_2) - f(x_1, 1) - f(1, x_2) + f(1, 1)$

- Example: $A_j = \sum_{x_j=1}^{K_j} f(x_1, x_2)/K_j$ for $j = 1, \ldots, 2$
  - $\beta_0 = A_1 A_2 f = f_{\cdot\cdot}$
  - $f_1 = (I - A_1)A_2 f = f_{x_1 \cdot} - f_{\cdot\cdot}$
  - $f_2 = (I - A_2)A_1 f = f_{\cdot x_2} - f_{\cdot\cdot}$
  - $f_{12} = (I - A_1)(I - A_2)f = f(x_1, x_2) - f_{x_1 \cdot} - f_{\cdot x_2} + f_{\cdot\cdot}$,

  $f_{\cdot\cdot}$ is the overall mean, $f_{x_1 \cdot} = \sum_{x_2=1}^{K_2} f(x_1, x_2)/K_2$ is the marginal average over x_2, $f_{\cdot x_2} = \sum_{x_1=1}^{K_1} f(x_1, x_2)/K_1$ is the marginal average over x_1.

Continuous Domain Example: $p = 2$

Domain $\mathcal{X}_1 = \mathcal{X}_2 = [0, 1]$.

- Example: $A_1 = f(0, x_2)$ and $A_2 = f(x_1, 0)$.
  - $\beta_0 = A_1 A_2 f = f(0, 0)$
  - $f_1 = (I - A_1)A_2 f = f(x_1, 0) - f(0, 0)$
  - $f_2 = (I - A_2)A_1 f = f(0, x_2) - f(0, 0)$
  - $f_{12} = (I - A_1)(I - A_2)f = f(x_1, x_2) - f(x_1, 0) - f(0, x_2) + f(0, 0)$

- Example: $A_j = \int_0^1 f(x_1, x_2) dx_j$ for $j = 1, \ldots, 2$
  - $\beta_0 = A_1 A_2 f = \int_0^1 \int_0^1 f(x_1, x_2) dx_1 x_2$
  - $f_1 = (I - A_1)A_2 f = \int_0^1 f(x_1, x_2) dx_2 - \beta_0$
  - $f_2 = (I - A_2)A_1 f = \int_0^1 f(x_1, x_2) dx_1 - \beta_0$
  - $f_{12} = (I - A_1)(I - A_2)f$
    $= f(x_1, x_2) - \int_0^1 f dx_1 - \int_0^1 f dx_2 + \int_0^1 \int_0^1 f(x_1, x_2) dx_1 x_2$

Truncated Models

- Additive models

$$f(\boldsymbol{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x_j),$$

  Claim X_j as unimportant if the function $f_j = 0$

- Two-way interaction model

$$f(\boldsymbol{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k).$$

  The interaction effect between X_j and X_k is unimportant if $f_{jk} = 0$.

Variable Selection for Additive Models

$$Y_i = \mu + \sum_{j=1}^{p} f_j(X_{ij}) + \epsilon_i,$$

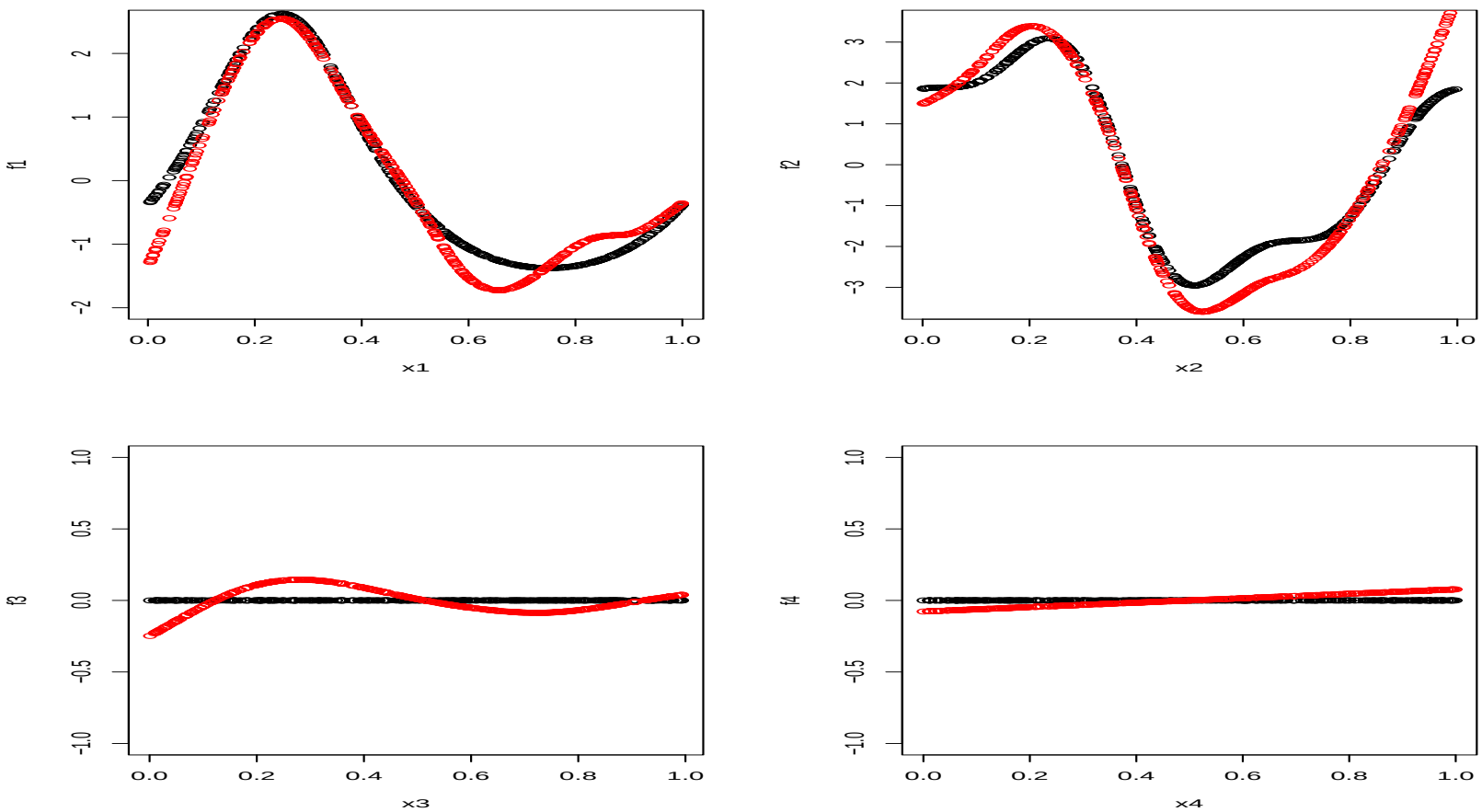
where

- μ is the intercept
- the component f_j's are unknown smooth functions
- ϵ_i is an unobserved random variable with mean 0 and finite variance σ^2.

Suppose some of the additive components f_j's are zero. The goals are

- to distinguish the nonzero components from the zero components
- to and estimate the nonzero components consistently

(Traditional) Smoothing Spline Fits



Traditional splines do not produce sparse function estimates, as zero functions are not estimated as exactly zero.

Existing Methods for Variable Selection for Additie Models

- Multivariate Adaptive Regression Splines (MARS) (Friedman 1991)
- Classification and Regression Tree (CART, Brieman 1985)
- Basis Pursuit (Zhang et al. 2004); Component Selection and Smoothing Operator (COSSO, Lin and Zhang, 2006)
- Group LASSO (Huang), Group SCAD (Wang et al. 2007)
- Sparse Additive Models (SpAM, Ravikuma 2007)

Example: to identify the correct sparse model structure

$$Y = f_1(X_1) + f_2(X_2) + 0(X_3) + 0(X_4) + \epsilon.$$

Review of Basis-Expansion Methods

Assume

$$f_j(x_j) = \sum_{k=1}^{m_n} \beta_{jk}\phi_k(x_j), \quad j = 1, \cdots, p,$$

where $\{\phi_k : 1 \leq k \leq m_n\}$ is the set of m_n basis functions.

- MARS (Friedman, 1991) builds models by sequentially searching/adding basis functions, followed by basis pruning (mimic stepwise selection)

Likelihood Basis Pursuit (LBP)

An early attempt to generalize LASSO to multivariate spline regression context. Consider additive models with a basis expansion:

$$f(\boldsymbol{x}) = \sum_{j=1}^{p} f_j(x_j) = \sum_{j=1}^{p} \sum_{k=1}^{m_j} \beta_{jk} \phi_k(x_j).$$

The LASSO penalty is imposed on basis coefficients

$$\min \sum_{i=1}^{n} [y_i - f(\boldsymbol{x}_i)]^2 + \lambda \sum_{j=1}^{p} \sum_{k=1}^{m_j} |\beta_{jk}|.$$

See Zhang et al. (2004).

Properties of LBP

- Each X_j is associated with multiple coefficients β_{lj}.
- To claim X_j as an unimportant variable, one needs $\beta_{jk} = 0$ for all $k = 1, \cdots, m_j$.
- Sequential testing was suggested (based on Monte Carlo bootstrap).

Group-LASSO Methods

Huang et al. (2010) considers the adaptive group Lasso by

$$\min_{\mu,\boldsymbol{\beta}_1,\cdots,\boldsymbol{\beta}_p} \sum_{i=1}^{n} \left(y_i - \mu - \sum_{j=1}^{p} \sum_{k=1}^{m_n} \beta_{jk}\phi_k(x_{ij}) \right)^2 + \lambda \sum_{j=1}^{p} w_j \|\boldsymbol{\beta}_j\|_2,$$

where $\boldsymbol{\beta}_j = (\beta_{j1}, \cdots, \beta_{jm_n})^T$.

- ϕ_k's are normalized B-spline basis functions.

Two Important Issues

- Can we do nonparametric variable selection via function shrinkage operator?
- Can the estimator achieve the nonparametric oracle properties?

Answer:
COSSO (Lin and Zhang, 2006), Adaptive COSSO.

COmponent Smoothing and Selection Operator

COSSO is a soft-thresholding operator on the function components to achieve sparse function estimation:

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} [y_i - f(\boldsymbol{x}_i)]^2 + \lambda J(f),$$

where $J(f)$ is the sum of RKHS norms (of the components)

$$J(f) = \sum_{j=1}^{p} \|P^j f\|_{\mathcal{F}}.$$

- P^j is the projection operator of f into the function space of X_j
- $J(f)$ is continuous in f; $J(f)$ is convex

See Lin and Zhang (2006) and Zhang and Lin (2006) for details.

Model Selection Consistency of COSSO

The solution $\hat{f}$ is a linear combination of p components

$$\hat{f}(\boldsymbol{x}) = b + \sum_{j=1}^{p} \hat{\theta}_j \Big[\sum_{i=1}^{n} c_i R_j(\boldsymbol{x}_i, \boldsymbol{x}) \Big].$$

Theorem 2 Let $\mathcal{F}$ be the second order Sobolev space of periodic functions. Under the tensor product design, when $\lambda \to 0, n\lambda^2 \to \infty$,

- If $f_j = 0$, then with probability tending to 1, $\hat{\theta}_j = 0$ (implying $\hat{f}_j$ is dropped from the model).
- If $f_j \neq 0$, then with probability tending to 1, $\hat{\theta}_j > 0$.

Example 1

- Dimension $p = 10$, sample size $n = 100$.
- Generate $y = f(\boldsymbol{x}) + \epsilon$, with $\epsilon \sim N(0, 1.74)$
- True regression function

$$f(\boldsymbol{x}) = 5g_1(x_1) + 3g_2(x_2) + 4g_3(x_3) + 6g_4(x_4).$$

- $\text{var}\{5g_1(X_1)\} = 2.08$, $\text{var}\{3g_2(X_2)\} = 0.80$, $\text{var}\{4g_3(X_3)\} = 3.30$, $\text{var}\{6g_4(X_4)\} = 9.45$.
- MARS is done in R with function "mars" in the "mda" library
- The magnitude of the functional component is measured by the empirical L_1 norm: $n^{-1}\sum_{i=1}^{n}|\hat{f}_j(x_{ij})|$ for each j.

Table 1. The average *ISE*s over 100 runs of Example 1 .

	Comp. symm.		
	$t = 0$	$t = 1$	$t = 3$
COSSO(GCV)	0.93 (0.05)	0.92 (0.04)	0.97 (0.07)
COSSO(5CV)	0.80 (0.03)	0.97 (0.05)	1.07 (0.06)
MARS	1.57 (0.07)	1.24 (0.06)	1.30 (0.06)
	AR(1)		
	$\rho = -0.5$	$\rho = 0$	$\rho = 0.5$
COSSO(GCV)	0.94 (0.05)	1.04 (0.07)	0.98 (0.07)
COSSO(5CV)	1.03 (0.06)	1.03 (0.06)	0.98 (0.05)
MARS	1.32 (0.07)	1.34 (0.07)	1.36 (0.08)

Selection Frequency of Variables

For the independent case, we have

	Variable									
	1	2	3	4	5	6	7	8	9	10
COSSO(GCV)	100	100	100	100	14	11	18	15	11	13
COSSO(5CV)	100	94	100	100	1	1	3	2	4	2
MARS	100	100	100	100	35	35	34	39	28	35

Nonparametric Oracle Property

A nonparametric regression estimator is *nonparametric-oracle* if

(1) $\|\hat{f} - f\| = O_p(n^{-\alpha})$ at the optimal rate $\alpha > 0$.

(2) $\hat{f}_j = 0$ for all $j \in U$ with probability tending to 1

The COSSO is not oracle:

- For consistent estimation, λ approaches to zero at a rate $n^{-4/5}$, which is faster than needed for consistent model selection.

Can we *oraclize* COSSO?

Adaptive (weighted) COSSO

Recently we proposed

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} [y_i - f(\boldsymbol{x}_i)]^2 + \lambda \sum_{j=1}^{p} w_j \| P^j f \|_{Fc}.$$

- $0 < w_j \leq \infty$ are pre-specified weights

Role of weights: adaptively adjusting the penalty on individual components based on their relative importance

- Large weights are imposed on unimportant components $\Longrightarrow$ sparsity
- Small weights are imposed on important components $\Longrightarrow$ small bias

Choices of Weights

Recall that $\|g\|_{L_2}^2 = \int [g(x)]^2 dx$.

- Given an initial estimate $\tilde{f}$, we construct the weights as

$$w_j = \|P^j \tilde{f}\|_{L^2}^{-\gamma}, \quad j = 1, \cdots, p.$$

  where $\gamma > 0$ is an extra parameter which can be adaptively chosen.

$\tilde{f}$ should be a good estimator (say, consistent) to accurately indicate the relative importance of different function components.

- In practice, use the traditional smoothing spline or COSSO fit as $\tilde{f}$ to construct weights.
- Adaptive COSSO estimator is np-oracle (Storlie, et al. 2011)