

Fast Sparse Regression and Classification

Jerome H. Friedman

Stanford University

PREDICTION (Regression/Classification)

y = outcome/response variable

$\mathbf{x} = \{x_1, \dots, x_n\}$ predictors

Goal: $\hat{y} = F(\mathbf{x})$

Want good $F(\mathbf{x})$

LINEAR MODEL

$$F(\mathbf{x}; \mathbf{a}) = a_0 + \sum_{j=1}^n a_j x_j$$

a_0 = intercept

$\{a_j\}_1^n$ = coefficients

ACCURACY

Cost for error: $L(y, F)$

$$L(y, F) = (y - F)^2, |y - F| \quad y \in R$$

$y \in \{-1, 1\}$:

$$L(y, F) = \log(1 + e^{-yF}) \quad \text{logistic reg.}$$

$$L(y, F) = (1 - yF)_+ \quad \text{SVM}$$

any $-\log(\text{likelihood})$

many many more

PREDICTION RISK

$$R(\mathbf{a}) = E_{\mathbf{x},y}L(y, F(\mathbf{x}; \mathbf{a}))$$

Optimal solution: $\mathbf{a}^* = \arg \min_{\mathbf{a}} R(\mathbf{a})$

$p(\mathbf{x}, y)$ unknown $\Rightarrow \mathbf{a}^*$ unknown

STATISTICAL (MACHINE) LEARNING

Training data: $\{y_i, \mathbf{x}_i\}_1^N \sim p(\mathbf{x}, y)$

$$\hat{R}(\mathbf{a}) = \frac{1}{N} \sum_{i=1}^N L \left(y_i, a_0 + \sum_{j=1}^n a_j x_{ij} \right)$$

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} \hat{R}(\mathbf{a})$$

If not $N \gg n$, not very good!

$$R(\hat{\mathbf{a}}) \gg R(\mathbf{a}^*) \quad (\text{high variance})$$

REGULARIZATION (biased learning)

$$\hat{\mathbf{a}}(\lambda) = \arg \min_{\mathbf{a}} [\hat{R}(\mathbf{a}) + \lambda \cdot P(\mathbf{a})]$$

$P(\mathbf{a}) \geq 0$ penalty

$0 \leq \lambda \leq \infty$ strength parameter

$\lambda \uparrow \implies$ bias \uparrow & variance \downarrow

$\hat{\mathbf{a}}(\lambda) \sim$ 1-dim. path of solutions $\in S^{n+1}$

$S^{n+1} =$ parameter space

MODEL SELECTION (λ)

$$\lambda^* = \arg \min_{0 \leq \lambda \leq \infty} R(\hat{\mathbf{a}}(\lambda))$$

Model selection criterion:

$$\tilde{R}(\mathbf{a}) = \text{surrogate for } R(\mathbf{a})$$

depends on $L(y, F)$ & $P(\mathbf{a})$

$$\hat{\lambda} = \arg \min_{0 \leq \lambda \leq \infty} \tilde{R}(\hat{\mathbf{a}}(\lambda))$$

$$\hat{\mathbf{a}}(\hat{\lambda}) = \text{selected model}$$

Cross-validation: any $L(y, F)$ & $P(\mathbf{a})$

PENALTY SELECTION

$\mathbf{a}^* = \text{point in } S^{n+1}$

Choose penalty that induces paths that

on average come close to \mathbf{a}^*

$$\{y_i, \mathbf{x}_i\}_1^N \sim p(\mathbf{x}, y)$$

Depends on \mathbf{a}^*

Choose $P(\mathbf{a})$ based on knowledge of \mathbf{a}^*

SPARSITY

Fraction of non influential variables

$$S(\mathbf{a}) = \#(|a_k| = \text{small})/n$$

Assumption: $\hat{\mathbf{a}} \simeq \mathbf{a}^* \Rightarrow S(\hat{\mathbf{a}}) \simeq S(\mathbf{a}^*)$

Choose $P(\mathbf{a})$ s.t. $S(\hat{\mathbf{a}}(\lambda^*)) \simeq S(\mathbf{a}^*)$

Don't know $S(\mathbf{a}^*)$?

Family of penalties $P_\gamma(\mathbf{a})$: γ regulates $S(\hat{\mathbf{a}})$

bridging sparse \rightarrow dense solutions

Model selection to jointly estimate (γ, λ)

("bridge regression": Frank & Friedman 1993)

POWER FAMILY

$$P_\gamma(\mathbf{a}) = \sum_{j=1}^n |a_j|^\gamma$$

With $L(y, F) = (y - F)^2$:

$\gamma = 2$: ridge-regression (dense)

$\gamma = 1$: lasso (moderately sparse)

$\gamma = 0$: (all) subsets selection (sparsest)

$0 \leq \gamma \leq 2$ bridges subset \rightarrow ridge

Note: $\gamma \geq 1 \Rightarrow$ convex, $\gamma < 1 \Rightarrow$ non convex

Generalized Elastic Net (convex)

$1 \leq \beta \leq 2$ (bridges lasso \rightarrow ridge):

Elastic Net (Zou & Hastie 2005)

$$P_{\beta}(\mathbf{a}) = \sum_{j=1}^n (\beta - 1) a_j^2 / 2 + (2 - \beta) |a_j|$$

Generalized Elastic Net (non convex)

$0 \leq \beta < 1$ (bridges subset selection \rightarrow lasso):

$$P_{\beta}(\mathbf{a}) = \sum_{j=1}^n \log((1 - \beta) |a_j| + \beta)$$

$0 \leq \beta \leq 2$ bridges subset \rightarrow ridge

Better statistical & computational properties

Method works for both + many more

BRIDGE REGRESSION

(1) Repeatedly solve:

$$\hat{\mathbf{a}}_{\beta}(\lambda) = \arg \min_{\mathbf{a}} [\hat{R}(\mathbf{a}) + \lambda \cdot P_{\beta}(\mathbf{a})]$$

$$0 \leq \beta \leq 2, \quad 0 \leq \lambda \leq \infty$$

(2) $(\hat{\beta}, \hat{\lambda}) \leftarrow$ model selection criterion

(3) $\hat{\mathbf{a}}_{\hat{\beta}}(\hat{\lambda}) =$ solution

Big challenge: fast enough algorithm for (1)

Especially for $P_{\beta}(\mathbf{a}) =$ non convex

DIRECT PATH SEEKING

Goal: rapidly produce path \simeq given $P(\mathbf{a})$

without repeatedly optimizing

EXAMPLES

$$L(y, F) = (y - F)^2:$$

PLS \simeq ridge-regression ($\beta = 2$)

LAR \simeq lasso ($\beta = 1$)

Stepwise \simeq all-subsets ($\beta = 0$)

Any convex $L(y, F)$:

Gradient boosting \simeq lasso ($\beta = 1$)

Want bridge regression: $0 \leq \beta \leq 2$

Generalized Path Seeking (GPS)

Fast algorithm for:

(1) any convex $L(y, F)$ (some non convex)

(2) any $P(\mathbf{a})$ s.t. $\frac{\partial P(\mathbf{a})}{\partial |a_j|} \geq 0$

i.e. $P(\mathbf{a})$ monotone $\uparrow |a_j|$

EXAMPLES

power family

generalized elastic net family (*)

SCAD, MC_+ family

grouped lasso, grouped bridge family, CAP

many many more

Definitions

$\nu \geq 0$: path length

$\Delta\nu > 0$: small increment

$$g_j(\nu) = - \left[\frac{\partial \hat{R}(\mathbf{a})}{\partial a_j} \right]_{\mathbf{a}=\hat{\mathbf{a}}(\nu)} \quad (\text{loss} + \text{data})$$

$$p_j(\nu) = \left[\frac{\partial P(\mathbf{a})}{\partial |a_j|} \right]_{\mathbf{a}=\hat{\mathbf{a}}(\nu)} \quad (\text{penalty})$$

$$\lambda_j(\nu) = g_j(\nu) / p_j(\nu) \quad (\text{combination})$$

- 1 Initialize: $\nu = 0; \{\hat{a}_j(0) = 0\}_1^n$
- 2 Loop {
- 3 Compute $\{\lambda_j(\nu)\}_1^n$
- 4 $S = \{j \mid \lambda_j(\nu) \cdot \hat{a}_j(\nu) < 0\}$
- 5 if ($S = \text{empty}$) $j^* = \arg \max_j |\lambda_j(\nu)|$
- 6 else $j^* = \arg \max_{j \in S} |\lambda_j(\nu)|$
- 7 $\hat{a}_{j^*}(\nu + \Delta\nu) = \hat{a}_{j^*}(\nu) + \Delta\nu \cdot \text{sign}(\lambda_{j^*}(\nu))$
- 8 $\{\hat{a}_j(\nu + \Delta\nu) = \hat{a}_j(\nu)\}_{j \neq j^*}$
- 9 $\nu \leftarrow \nu + \Delta\nu$
- 10 } Until $\lambda(\nu) = 0$

THEOREM

$\hat{\mathbf{a}}(\lambda) = \text{exact path}$

$\hat{\mathbf{a}}(\nu) = \text{GPS path}$

If for all $\lambda > \lambda_0$

all $\{\hat{a}_j(\lambda)\}_1^n$ are continuous and monotone

Then for all $\lambda > \lambda_0$

$\hat{\mathbf{a}}(\nu) = \hat{\mathbf{a}}(\lambda)$, as $\Delta\nu \rightarrow 0$

i.e. GPS produces exact path

Otherwise: $\hat{a}_j(\nu) \simeq \hat{a}_j(\lambda)$

When $\hat{a}_j(\lambda)$ becomes non monotone:

$\hat{a}_j(\nu)$ tends to slightly delay becoming non monotone

(smoother more stable paths)

When $\hat{a}_j(\lambda)$ discontinuous ($\gamma < 1$, $\beta < 1/2$):

$\hat{a}_j(\nu) =$ continuous

\sim interpolates between $\hat{a}_j(\lambda)$ discontinuities

(smoother more stable paths)

Regression: under-determined example

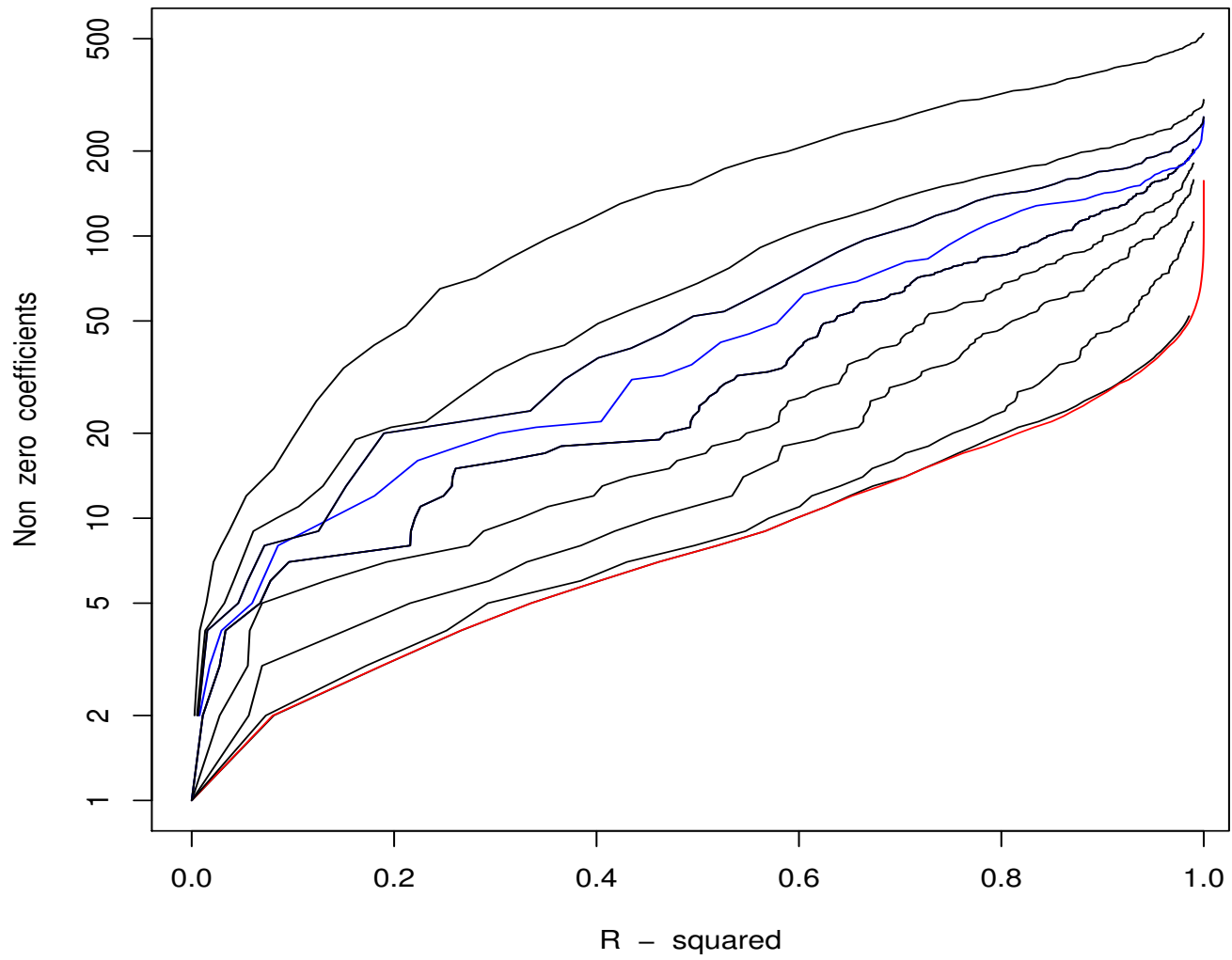
$$n = 10000, \quad N = 200$$

$$\mathbf{x}_i \sim N(\mathbf{0}, \mathbf{C}); \quad C_{jj} = 1, \quad C_{jk} = 0.4$$

$$y_i = \sum_{j=1}^n a_j^* x_{ij} + \varepsilon_i$$

$$\varepsilon_i \sim N(0, \sigma^2); \quad \sigma \sim 3/1 \text{ signal/noise}$$

$$|a_j^*| = [31 - j]_+, \quad \text{sign}(a_{j+1}^*) = -\text{sign}(a_j^*)$$



$\beta \in \{1.9, 1.7, 1.5, 1.0 \text{ (lasso, blue)}, 0.5, 0.3, 0.2, 0.1,$

THEREFORE

$P_\beta(\mathbf{a}) =$ generalized elastic net

$\beta \downarrow \Rightarrow S(\hat{\mathbf{a}}) \uparrow$ monotonically

at all path points

Penalty Selection (β)

Regression: under-determined example

$$n = 10000, \quad N = 200$$

50 data sets $\sim p(\mathbf{x}, y)$

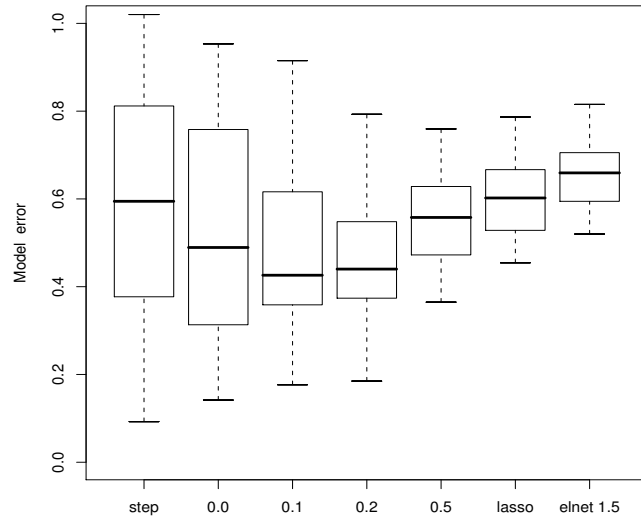
Distribution of closest “distance” to truth \mathbf{a}^* (risk)

Methods:

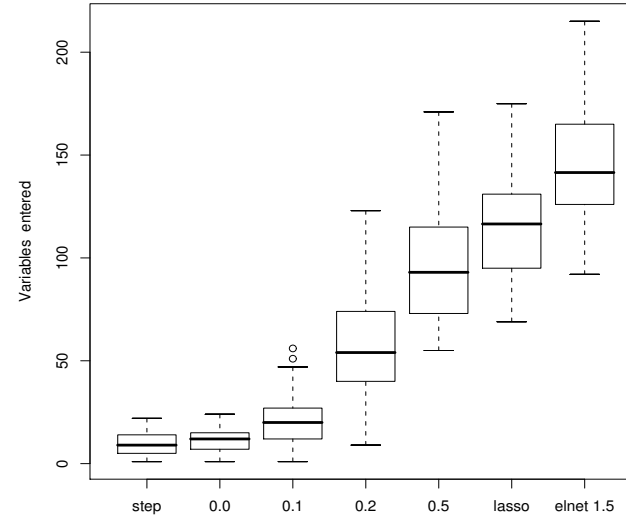
GPS: $\beta \in \{0.0, 0.1, 0.2, 0.5\}$

forward stepwise, lasso, elastic net (1.5)

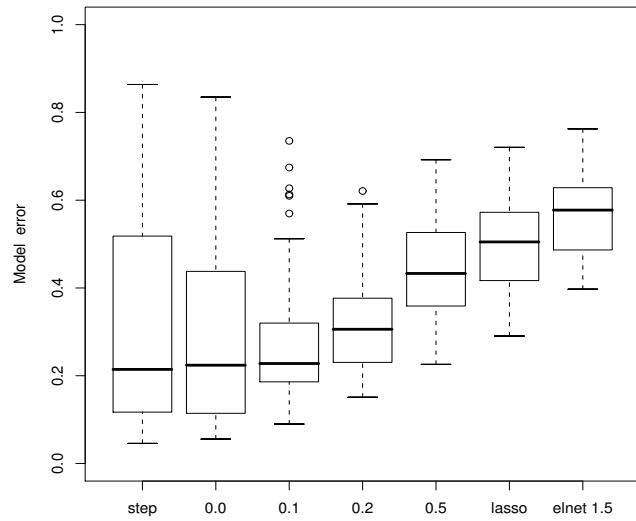
Corr = 0.4



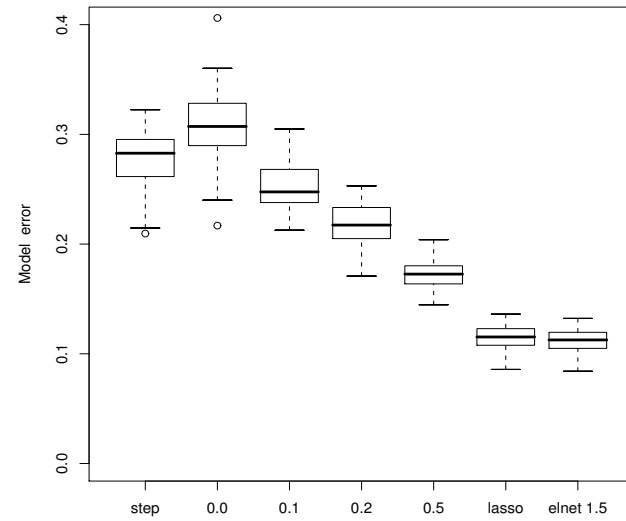
Corr = 0.4



Corr = 0.0



Corr = 0.4, a > 0



Post-processing Selectors

$$(1) \quad \tilde{\mathbf{a}}(\lambda) = \arg \min_{\mathbf{a}} \hat{R}(\mathbf{a}) + \lambda P(\mathbf{a})$$

$$P(\mathbf{a}) = \text{convex (lasso)}$$

$$(2) \quad A(\lambda) = \{j\}_{\tilde{a}_j(\lambda) \neq 0} \quad (\text{active variables at } \lambda)$$

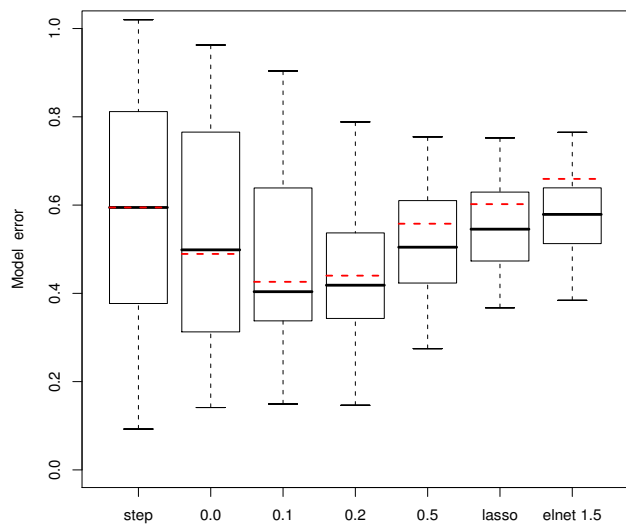
$$(3) \quad \hat{\mathbf{a}}(\lambda) = \arg \min_{\mathbf{a}} \hat{R}(\mathbf{a}) \quad \text{s.t.} \quad \{a_j = 0\}_{j \notin A(\lambda)}$$

Intuition (sparse problems):

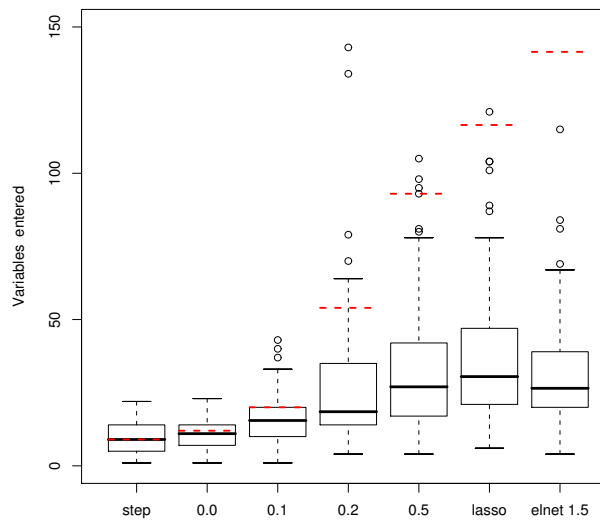
$\tilde{\mathbf{a}}(\lambda) \simeq$ selects correct variables

but over shrinks their values

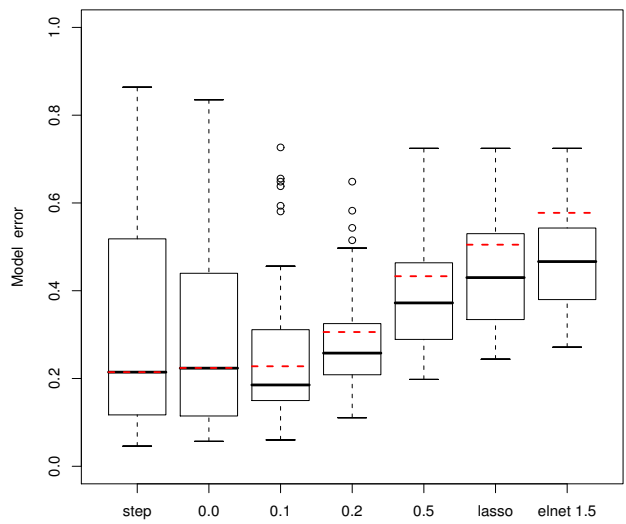
Corr = 0.4, Post



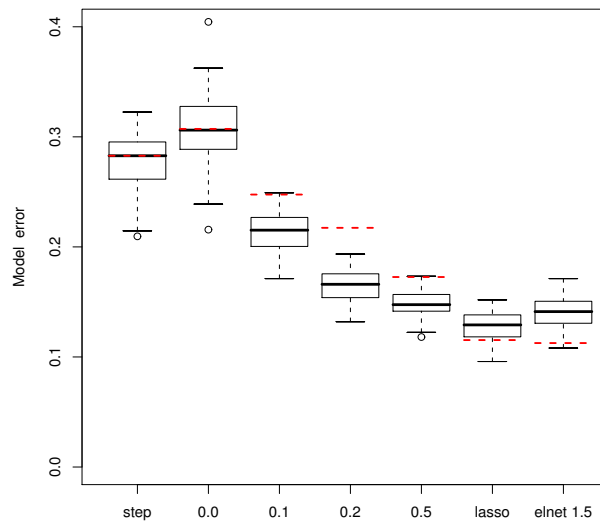
Corr = 0.4, Post



Corr = 0, Post



Corr = 0.4, a > 0, Post



CONCLUSIONS

(1) best penalty (prior) for $\{| a_j |\}$ depends on

$\{| a_j^* |, \text{sign}(a_j^*)\}$ and \mathbf{x} – distribution

(2) need bridge regression to chose $(\hat{\beta}, \hat{\lambda})$

(3) when sparse non convex $P(\mathbf{a})$ is best:

better variable *selection* & shrinkage

(4) best direct methods \rightarrow best selectors

(5) results same for logistic regression

Generalized Path Seeking

For same $L(y, F)$ & $P(\mathbf{a})$:

paths close to exact solutions

same sparseness properties

Can be applied with:

non convex $P(\mathbf{a}) \Rightarrow$ sparser than lasso

any convex $L(y, F)$, some non convex

Used as selector \rightarrow further improvement

Multinomial regression

TALK

http://www-stat.stanford.edu/~jhf/talks/GPS_R.pdf

PAPER

<http://www-stat.stanford.edu/~jhf/ftp/GPSpaper.pdf>

R PROCEDURE

<http://www-stat.stanford.edu/~jhf/R-GPS.html>