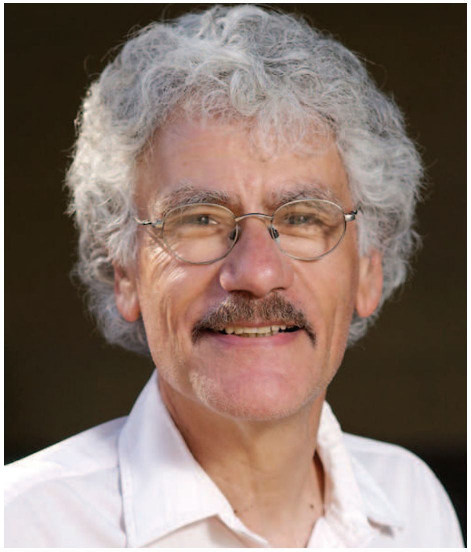
Sparse Linear Models

with demonstrations using GLMNET

Trevor Hastie

Stanford University

joint work with Jerome Friedman, Rob Tibshirani and Noah Simon



Linear Models for wide data

As datasets grow *wide*—i.e. many more features than samples—the linear model has regained favor as the tool of choice.

Document classification: bag-of-words can leads to $p = 20K$ features and $N = 5K$ document samples.

Genomics, microarray studies: $p = 40K$ genes are measured for each of $N = 100$ subjects.

Genome-wide association studies: $p = 500K$ SNPs measured for $N = 2000$ case-control subjects.

In examples like these we tend to use linear models — e.g. linear regression, logistic regression, Cox model. Since $p \gg N$, we cannot fit these models using standard approaches.

Forms of Regularization

We cannot fit linear models with $p > N$ without some constraints. Common approaches are

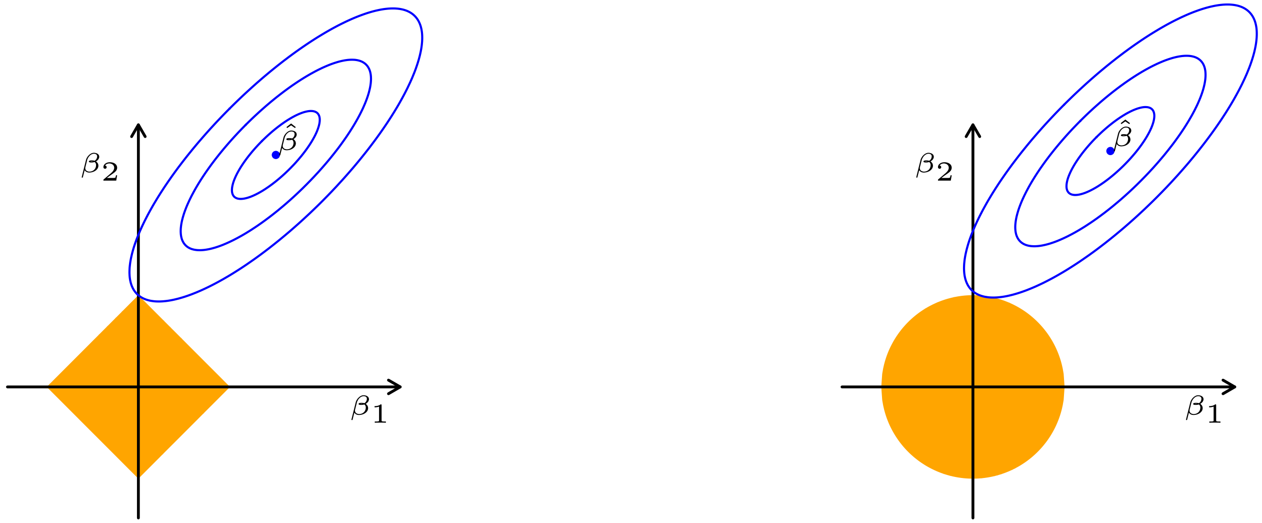
Forward stepwise adds variables one at a time and stops when overfitting is detected. This is a *greedy* algorithm, since the model with say 5 variables is not necessarily the best model of size 5.

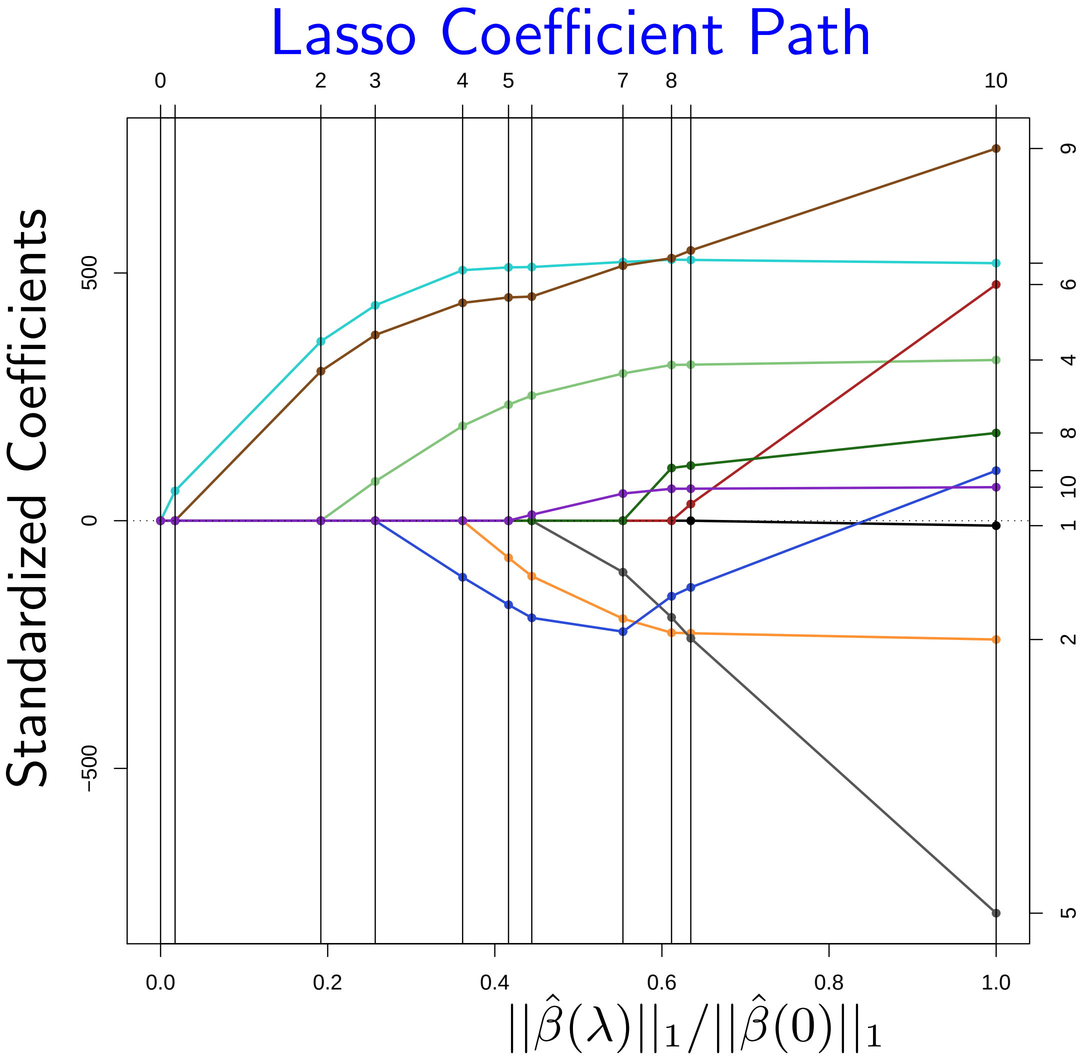
Best-subset regression finds the subset of each size k that fits the model the best. Only feasible for small p around 35.

Ridge regression fits the model subject to constraint $\sum_{j=1}^{p} \beta_j^2 \leq t$. Shrinks coefficients toward zero, and hence controls variance. Allows linear models with arbitrary size p to be fit, although coefficients always in row-space of X.

Lasso regression (Tibshirani, 1995) fits the model subject to constraint $\sum_{j=1}^{p} |\beta_j| \leq t$.

Lasso does variable selection and shrinkage, while ridge only shrinks.

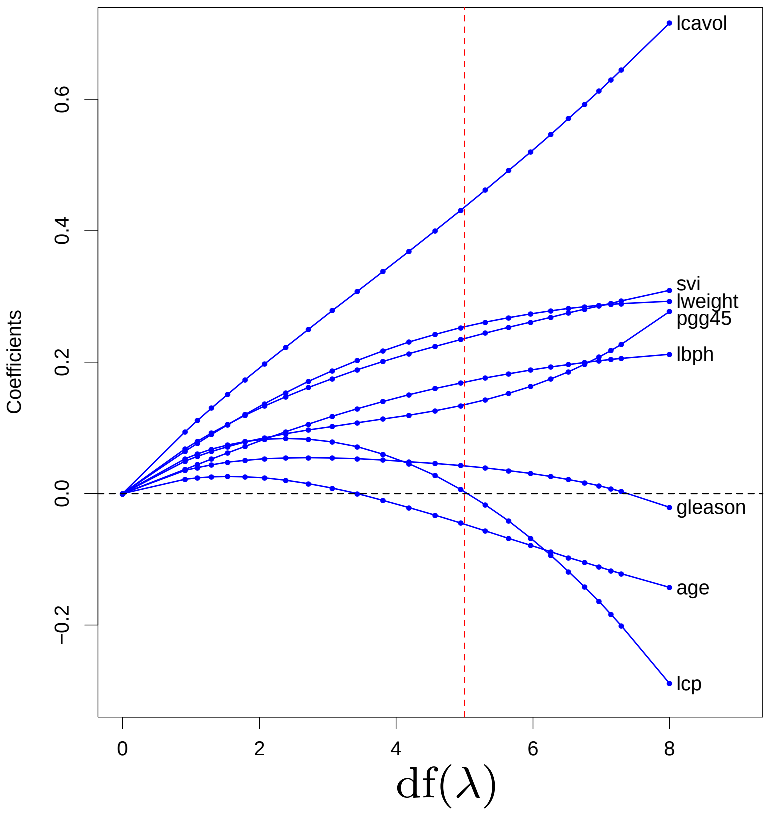


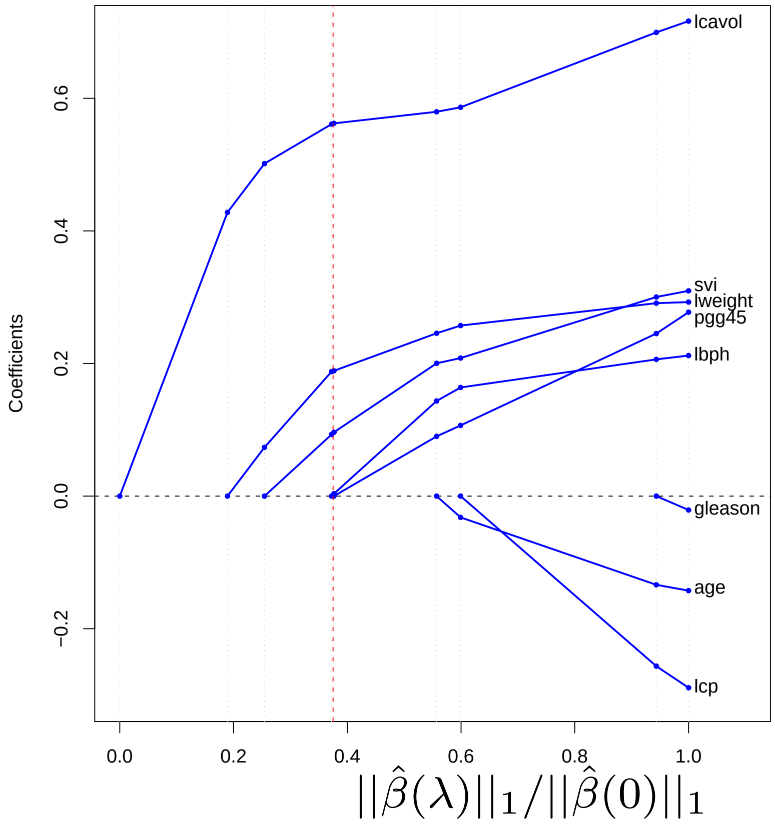


$$\text{Lasso: } \hat{\beta}(\lambda) = \text{argmin}_\beta \frac{1}{N}\sum_{i=1}^N (y_i - \beta_0 - x_i^T\beta)^2 + \lambda||\beta||_1$$

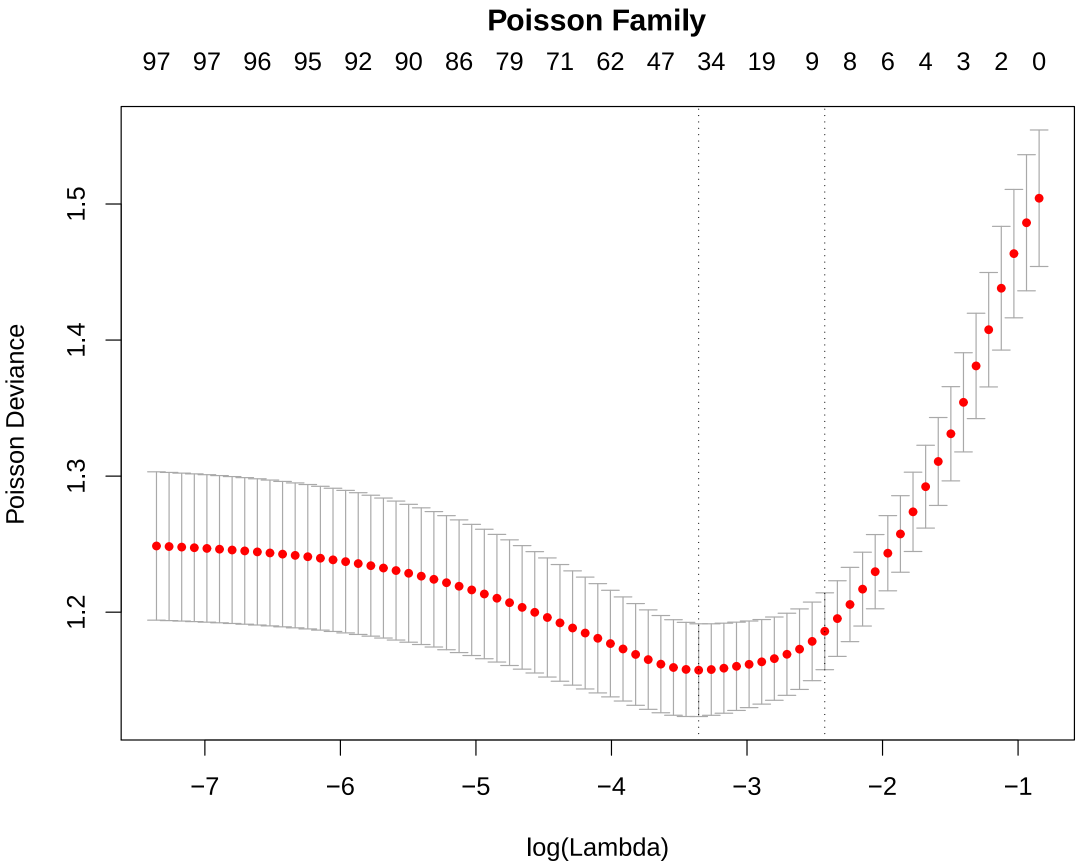
fit using LARS package in R (Efron, Hastie, Johnstone, Tibshirani 2002)

Ridge versus Lasso





Cross Validation to select λ



K-fold cross-validation is easy and fast. Here K=10, and the true model had 10 out of 100 nonzero coefficients.

History of Path Algorithms

Efficient path algorithms for $\hat{\beta}(\lambda)$ allow for easy and exact cross-validation and model selection.

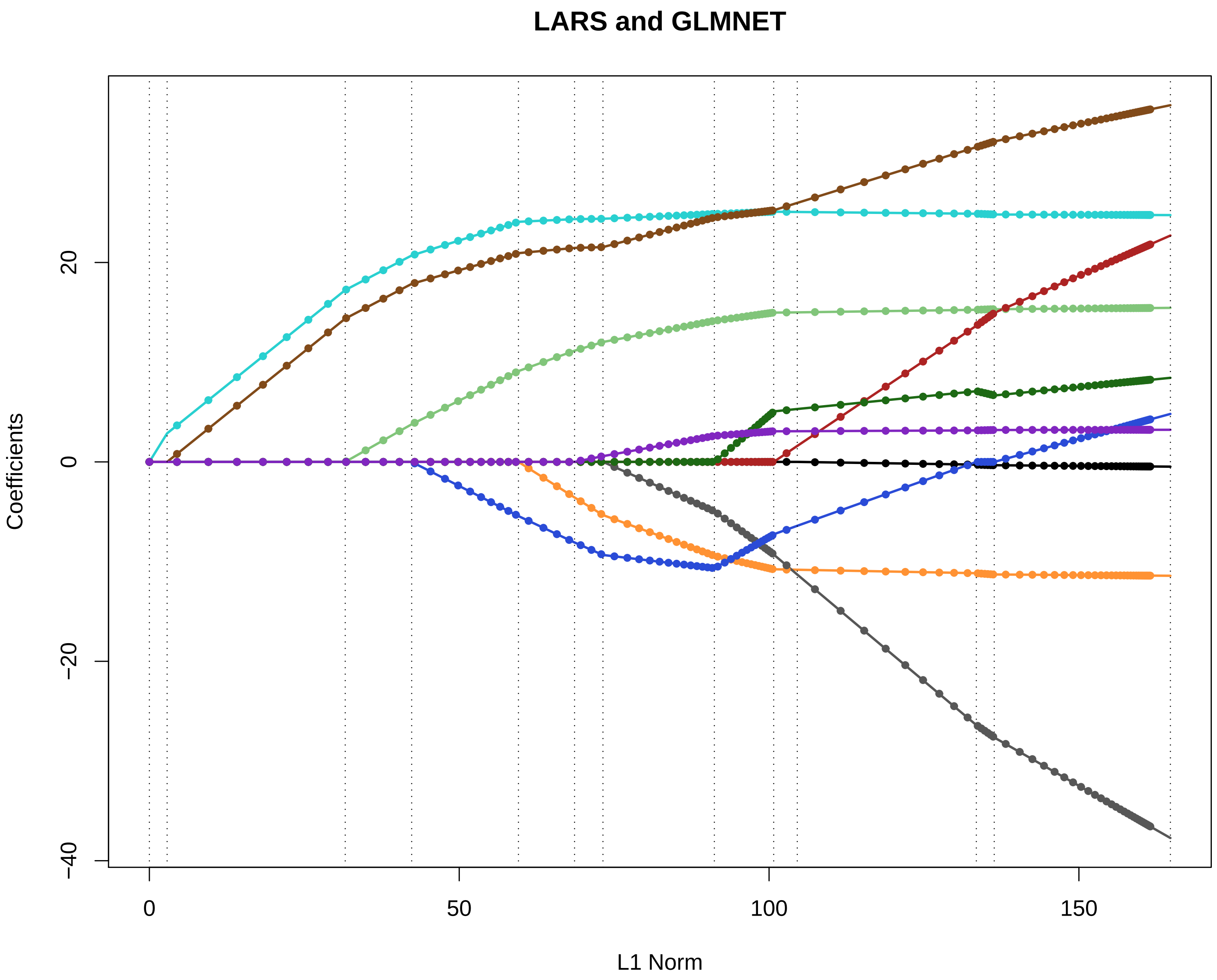
- In 2001 the LARS algorithm (Efron et al) provides a way to compute the entire lasso coefficient path efficiently at the cost of a full least-squares fit.
- 2001 – 2008: path algorithms pop up for a wide variety of related problems: Grouped lasso (Yuan & Lin 2006), support-vector machine (Hastie, Rosset, Tibshirani & Zhu 2004), elastic net (Zou & Hastie 2004), quantile regression (Li & Zhu, 2007), logistic regression and glms (Park & Hastie, 2007), Dantzig selector (James & Radchenko 2008), ...
- Many of these do not enjoy the piecewise-linearity of LARS, and seize up on very large problems.

GLMNET and coordinate descent

- Solve the lasso problem by coordinate descent: optimize each parameter separately, holding all the others fixed. Updates are trivial. Cycle around till coefficients stabilize.
- Do this on a grid of λ values, from λ_{max} down to λ_{min} (uniform on log scale), using warms starts.
- Can do this with a variety of loss functions and additive penalties.

Coordinate descent achieves dramatic speedups over all competitors, by factors of 10, 100 and more.

Friedman, Hastie and Tibshirani 2008 + long list of other who have also worked with coordinate descent.


LARS and GLMNET
Coefficients
L1 Norm
20
0
−20
−40
0
50
100
150

GLMNET **package in R**

Fits coefficient paths for a variety of different GLMs and the *elastic net* family of penalties.

Some features of `glmnet`:

- Models: linear, logistic, multinomial (grouped or not), Poisson, Cox model, and multiple-response grouped linear.
- Elastic net penalty includes *ridge* and *lasso*, and hybrids in between (more to come)
- *Speed!*
- Can handle large number of variables p. Along with exact screening rules we can fit GLMs on GWAS scale (more to come)
- Cross-validation functions for all models.
- Can allow for sparse matrix formats for $\mathbf{X}$, and hence massive

problems (eg $N = 11K$, $p = 750K$ logistic regression).

- Can provide lower and upper bounds for each coefficient; eg: positive lasso
- Useful bells and whistles:
  - Offsets — as in GLM, can have part of the linear predictor that is given and not fit. Often used in Poisson models (sampling frame).
  - Penalty strengths — can alter relative strength of penalty on different variables. Zero penalty means a variable is *always in* the model. Useful for adjusting for demographic variables.
  - Observation weights allowed.
  - Can fit no-intercept models
  - Session-wise parameters can be set with new `glmnet.options` command.

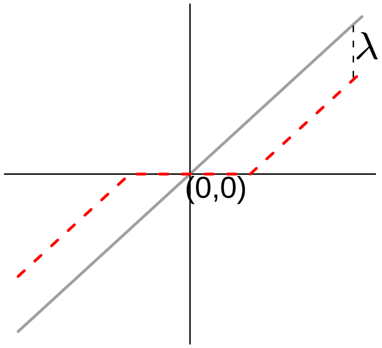
Coordinate descent for the lasso

$$\min_\beta \frac{1}{2N}\sum_{i=1}^N (y_i - \sum_{j=1}^p x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^p |\beta_j|$$

Suppose the p predictors and response are standardized to have mean zero and variance 1. Initialize all the $\beta_j = 0$.

Cycle over $j = 1, 2, \ldots, p, 1, 2, \ldots$ till convergence:

- Compute the partial residuals $r_{ij} = y_i - \sum_{k\neq j} x_{ik}\beta_k$.
- Compute the simple least squares coefficient of these residuals on jth predictor: $\beta_j^* = \frac{1}{N}\sum_{i=1}^N x_{ij}r_{ij}$
- Update β_j by *soft-thresholding*:



$$\begin{aligned} \beta_j &\leftarrow S(\beta_j^*, \lambda) \\ &= \text{sign}(\beta_j^*)(|\beta_j^*| - \lambda)_+ \end{aligned}$$

Elastic-net penalty family

Family of convex penalties proposed in Zou and Hastie (2005) for $p \gg N$ situations, where predictors are correlated in groups.

$$\min_\beta \frac{1}{2N}\sum_{i=1}^N (y_i - \sum_{j=1}^p x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^p P_\alpha(\beta_j)$$

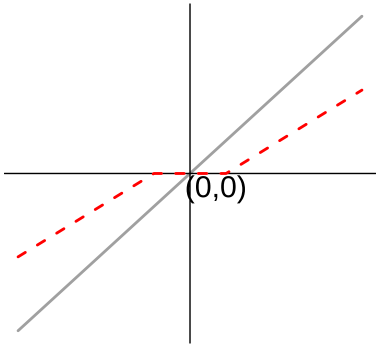
$$\text{with } P_\alpha(\beta_j) = \frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j|.$$

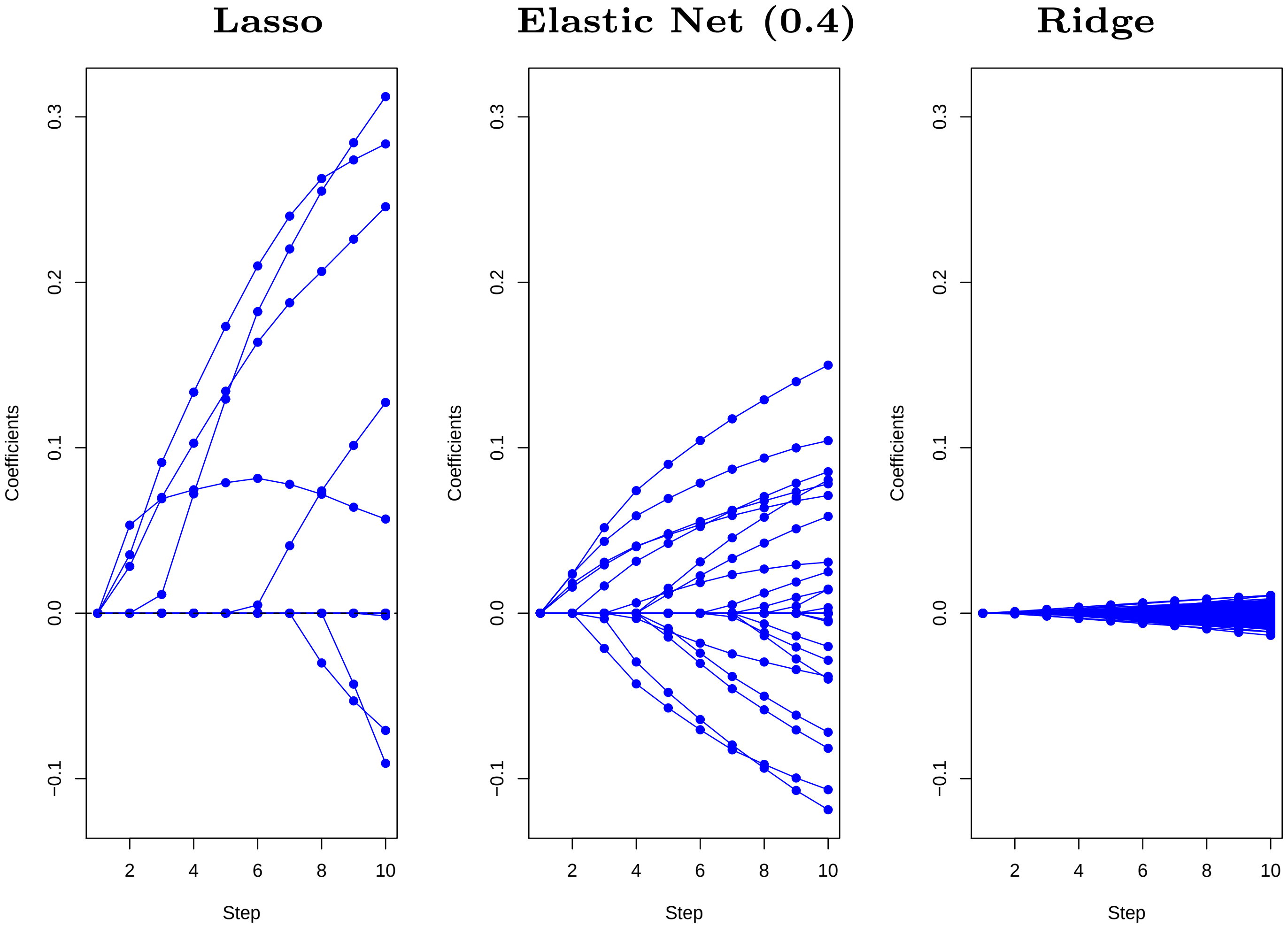
α creates a compromise between the *lasso* and *ridge*.

Coordinate update is now

$$\beta_j \leftarrow \frac{S(\beta_j^*,\ \lambda\alpha)}{1+\lambda(1-\alpha)}$$

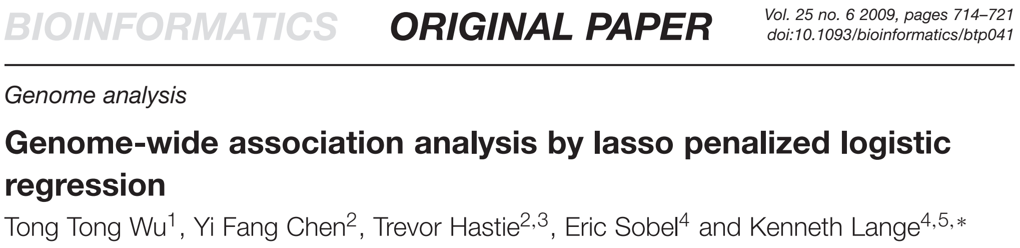
where $\beta_j^* = \frac{1}{N}\sum_{i=1}^N x_{ij}r_{ij}$ as before.





Leukemia Data, Logistic, N=72, p=3571, first 10 steps shown

Exact variable screening



BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 6 2009, pages 714–721
doi:10.1093/bioinformatics/btp041

Genome analysis

Genome-wide association analysis by lasso penalized logistic regression

Tong Tong Wu[1], Yi Fang Chen[2], Trevor Hastie[2,3], Eric Sobel[4] and Kenneth Lange[4,5,*]

Logistic regression for GWAS: $p \sim$ million, $N = 2000$.

- Compute $|\langle x_j, y - \bar{y}\rangle|$ for each Snp $j = 1, 2, \ldots, 10^6$.
- Fit lasso logistic regression path to largest 1000 (typically fit models of size around 20 or 30 in GWAS)
- Simple confirmations check that omitted Snps would not have entered the model.

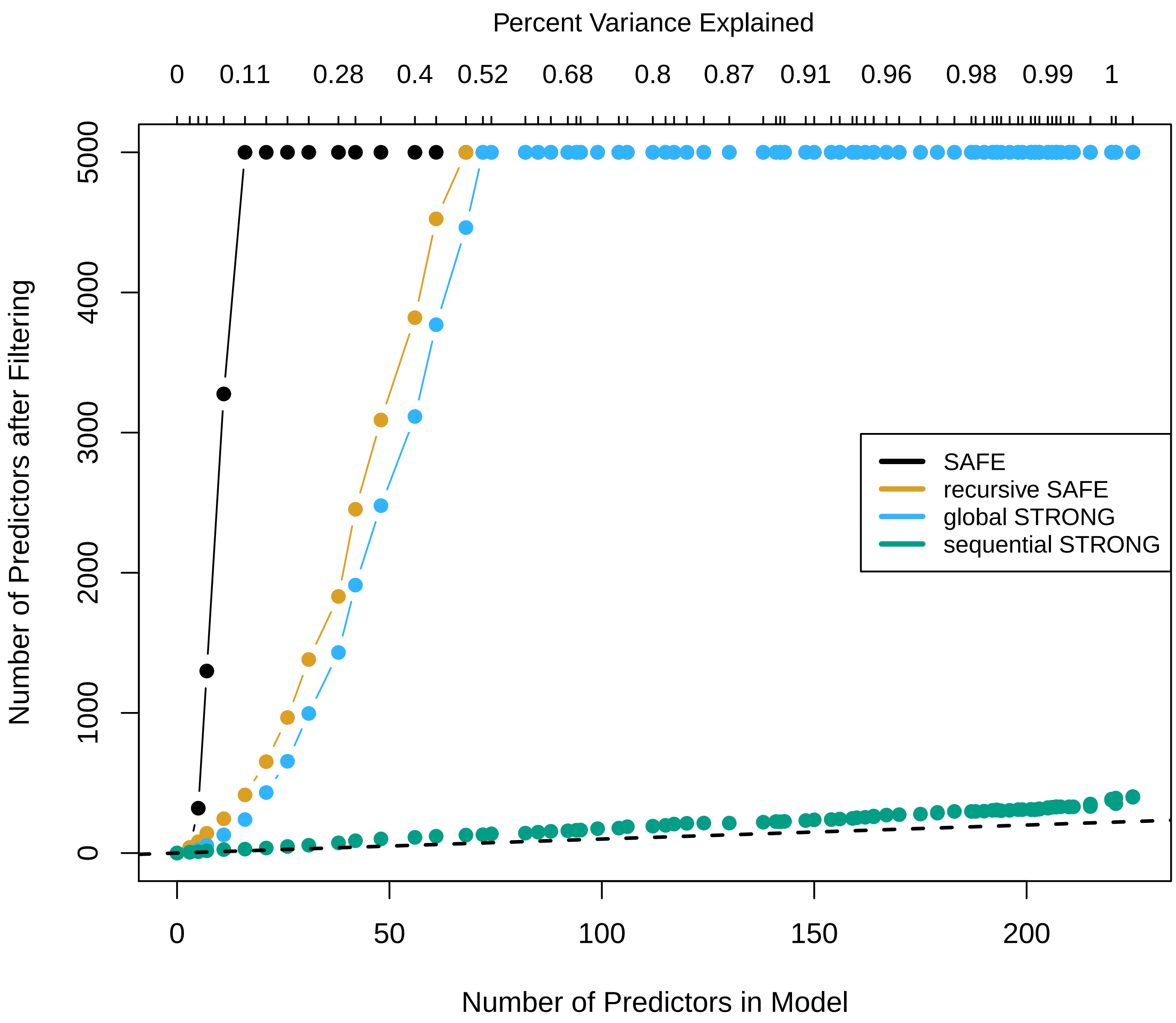
Safe and Strong Rules

- El Ghaoui et al (2010) propose SAFE rules for Lasso for screening predictors — quite conservative
- Tibshirani et al (JRSSB March 2012) improve these using STRONG screening rules.

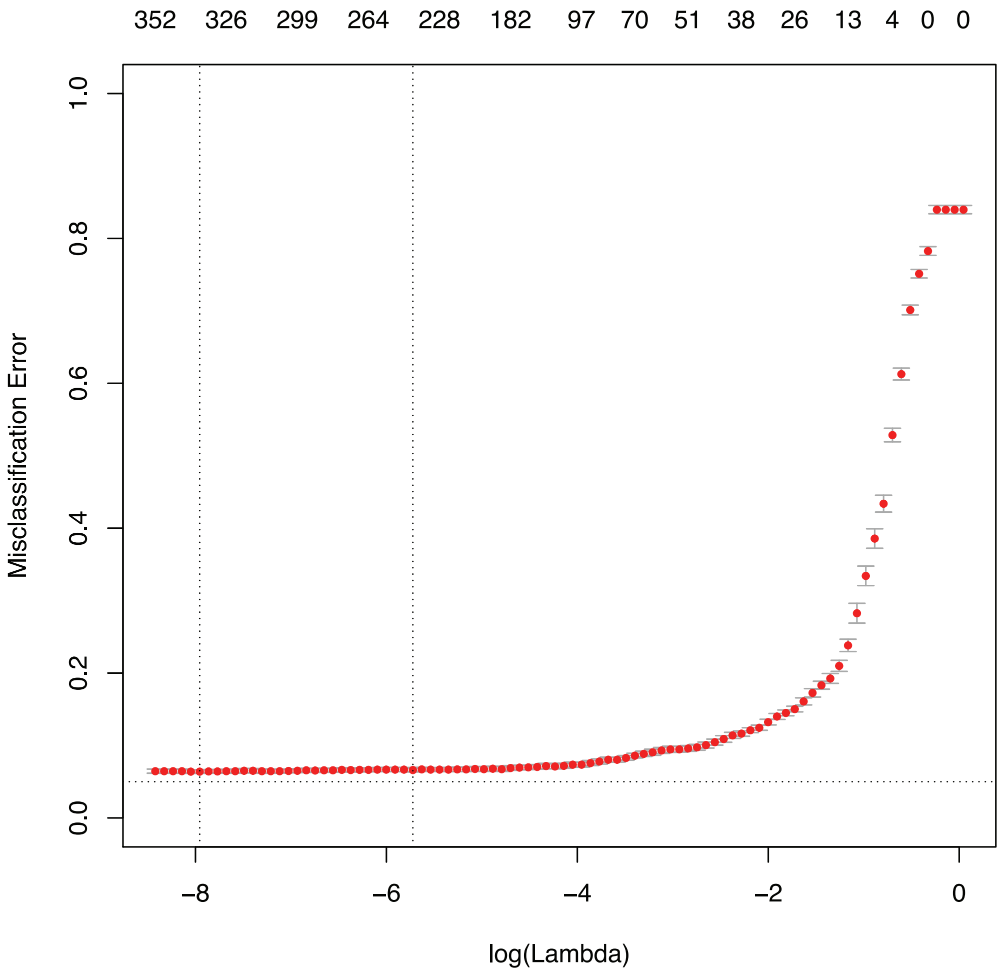
  Suppose fit at λ_ℓ is $\mathbf{X}\hat{\beta}(\lambda_\ell)$, and we want to compute the fit at $\lambda_{\ell+1} < \lambda_\ell$. Strong rules only consider set

  $$\left\{ j : |\langle \mathbf{x}_j, \mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_\ell)\rangle| > \lambda_{\ell+1} - (\lambda_\ell - \lambda_{\ell+1}) \right\}$$

  GLMNET screens at every λ step, and after convergence, checks if any violations.


Percent Variance Explained
0
0.11
0.28
0.4
0.52
0.68
0.8
0.87
0.91
0.96
0.98
0.99
1
Number of Predictors after Filtering
0
1000
2000
3000
4000
5000
SAFE
recursive SAFE
global STRONG
sequential STRONG
0
50
100
150
200
Number of Predictors in Model

Multiclass classification



Pathwork® Diagnostics

Microarray classification: tissue of origin

3220 samples

22K genes

17 classes (tissue type)

Multinomial regression model with

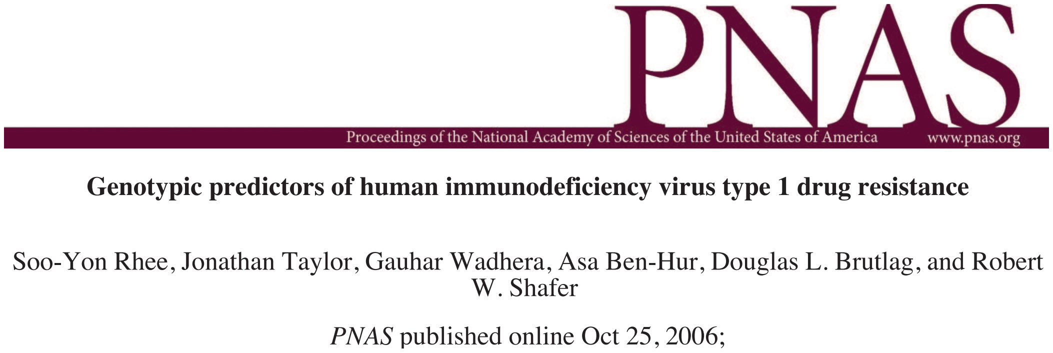
17×22K = 374K parameters

Elastic-net ($\alpha = 0.25$)

Example: HIV drug resistance

Paper looks at *in vitro* drug resistance of $N = 1057$ HIV-1 isolates to protease and reverse transcriptase mutations. Here we focus on Lamivudine (a Nucleoside RT inhibitor). There are $p = 217$ (binary) mutation variables.

Paper compares 5 different regression methods: decision trees, neural networks, SVM regression, OLS and LAR (lasso).



PNAS

Proceedings of the National Academy of Sciences of the United States of America www.pnas.org

Genotypic predictors of human immunodeficiency virus type 1 drug resistance

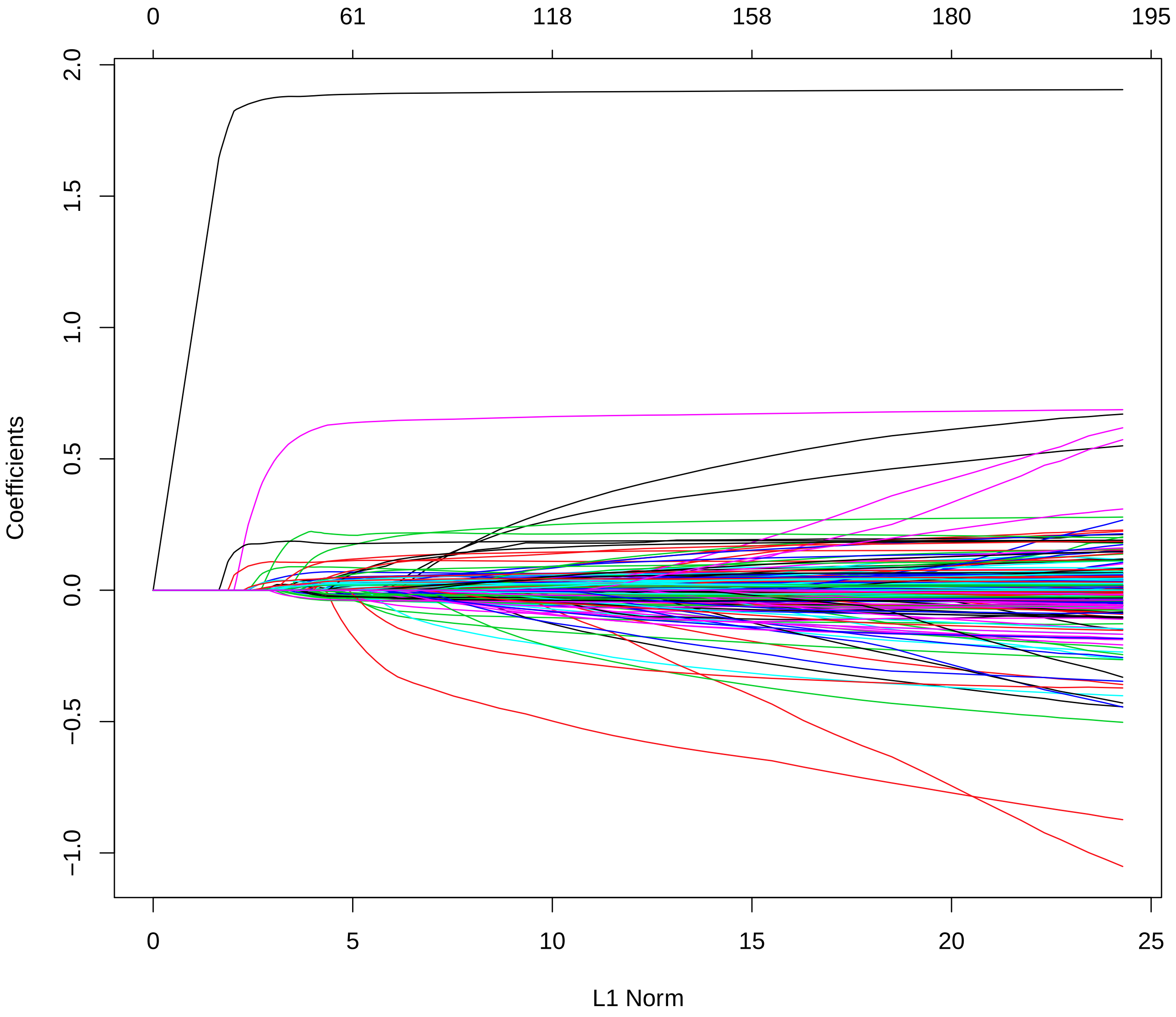
Soo-Yon Rhee, Jonathan Taylor, Gauhar Wadhera, Asa Ben-Hur, Douglas L. Brutlag, and Robert W. Shafer

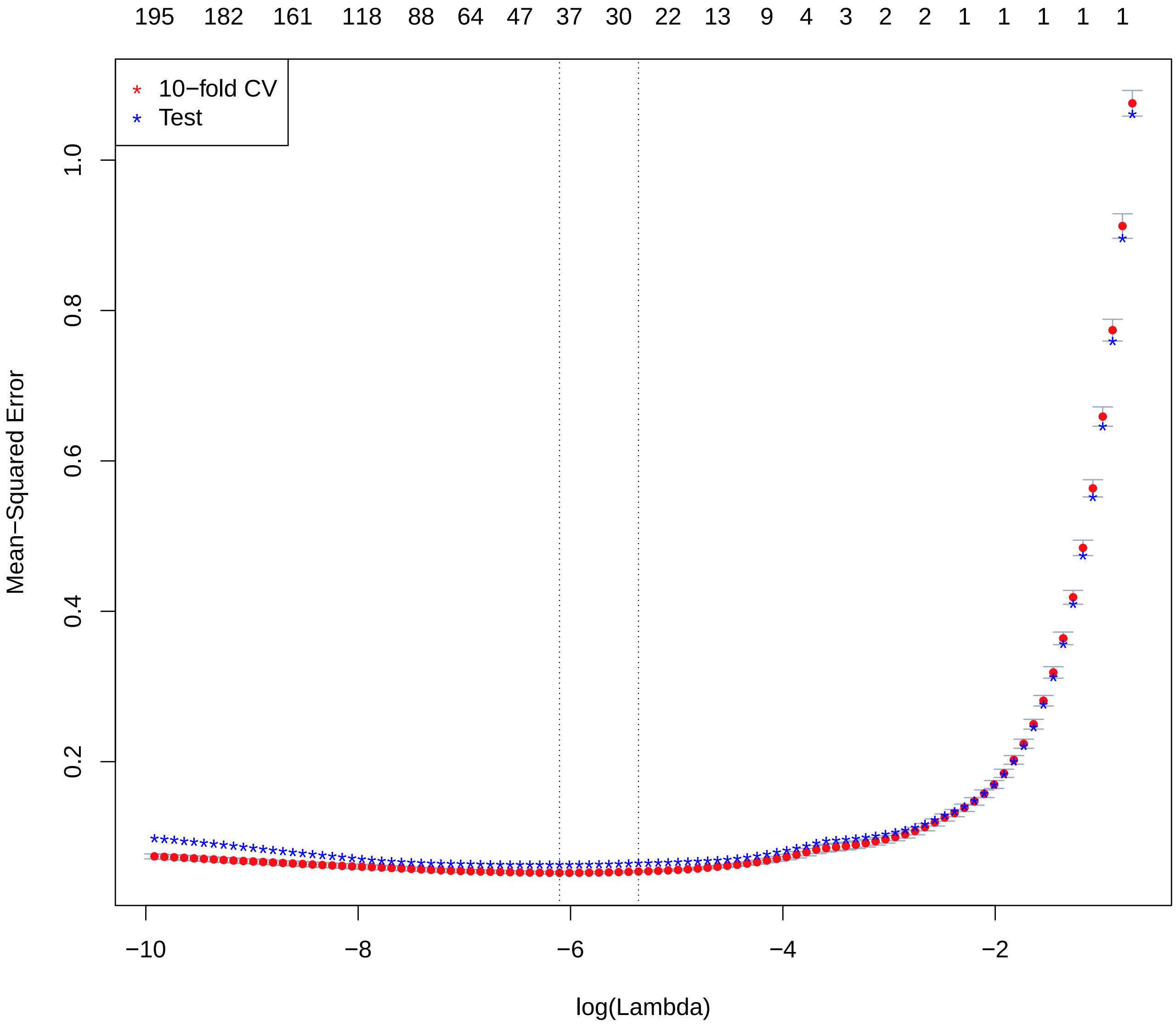
PNAS published online Oct 25, 2006;

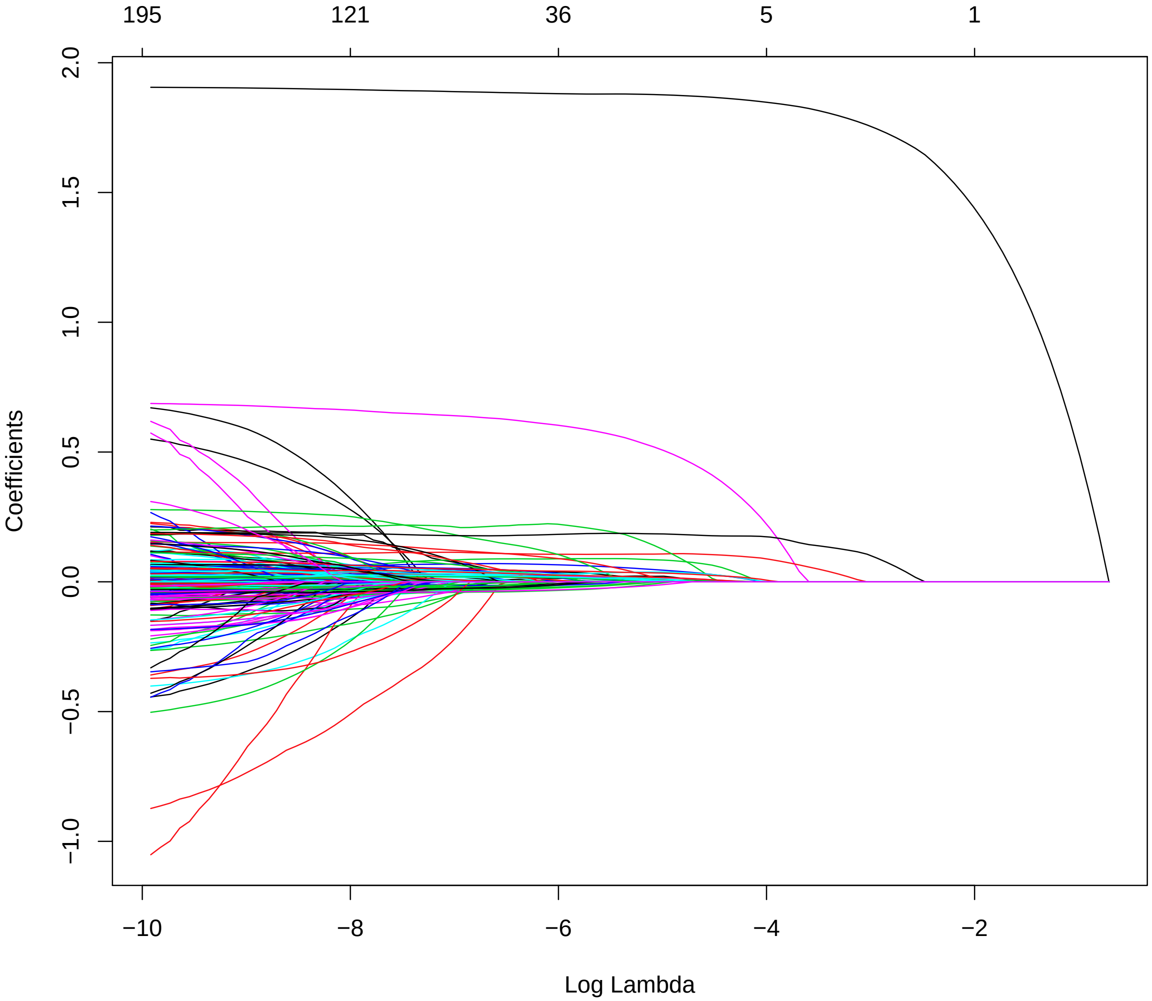
.

R code for fitting model

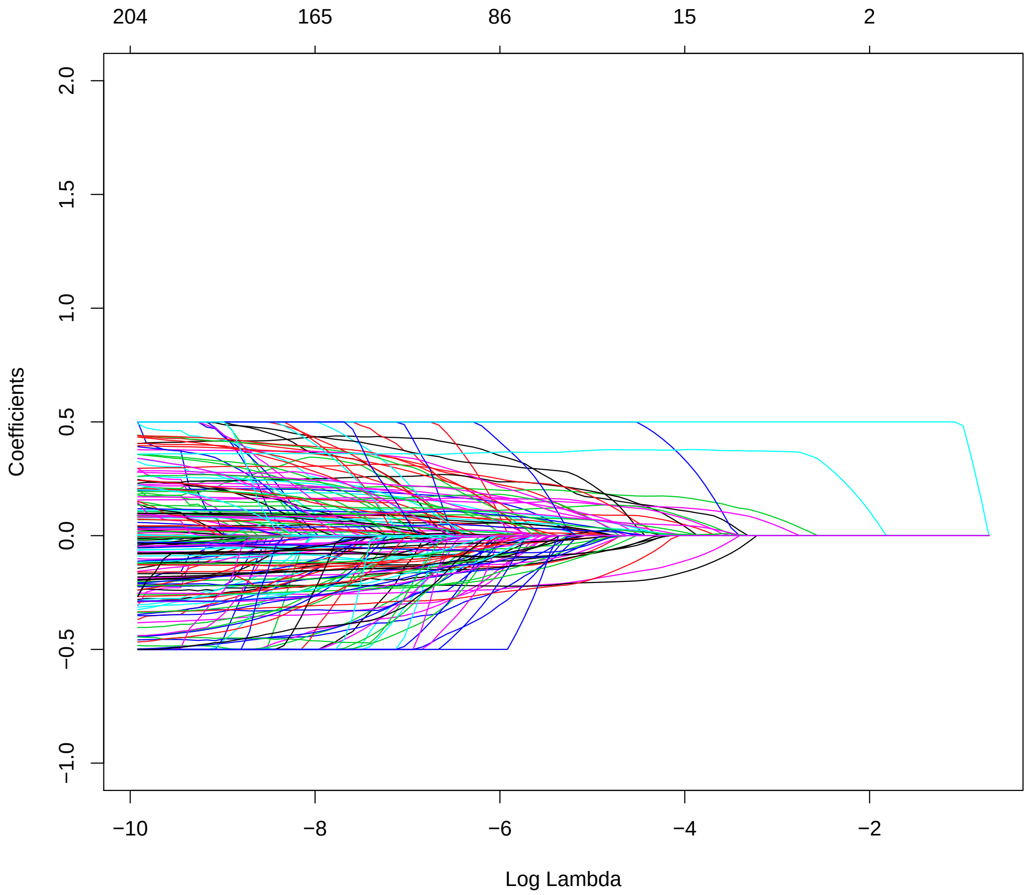
```
> require(glmnet)
> fit=glmnet(xtr,ytr,standardize=FALSE)
> plot(fit)

> cv.fit=cv.glmnet(xtr,ytr,standardize=FALSE)
> plot(cv.fit)
>
> mte=predict(fit,xte)
> mte= apply( (mte-yte)^2,2,mean)
> points(log(fit$lambda),mte,col="blue",pch="*")
> legend("topleft",legend=c("10-fold␣CV","Test"),
            pch="*",col=c("red","blue"))
```


0
61
118
158
180
195
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
Coefficients
0
5
10
15
20
25
L1 Norm

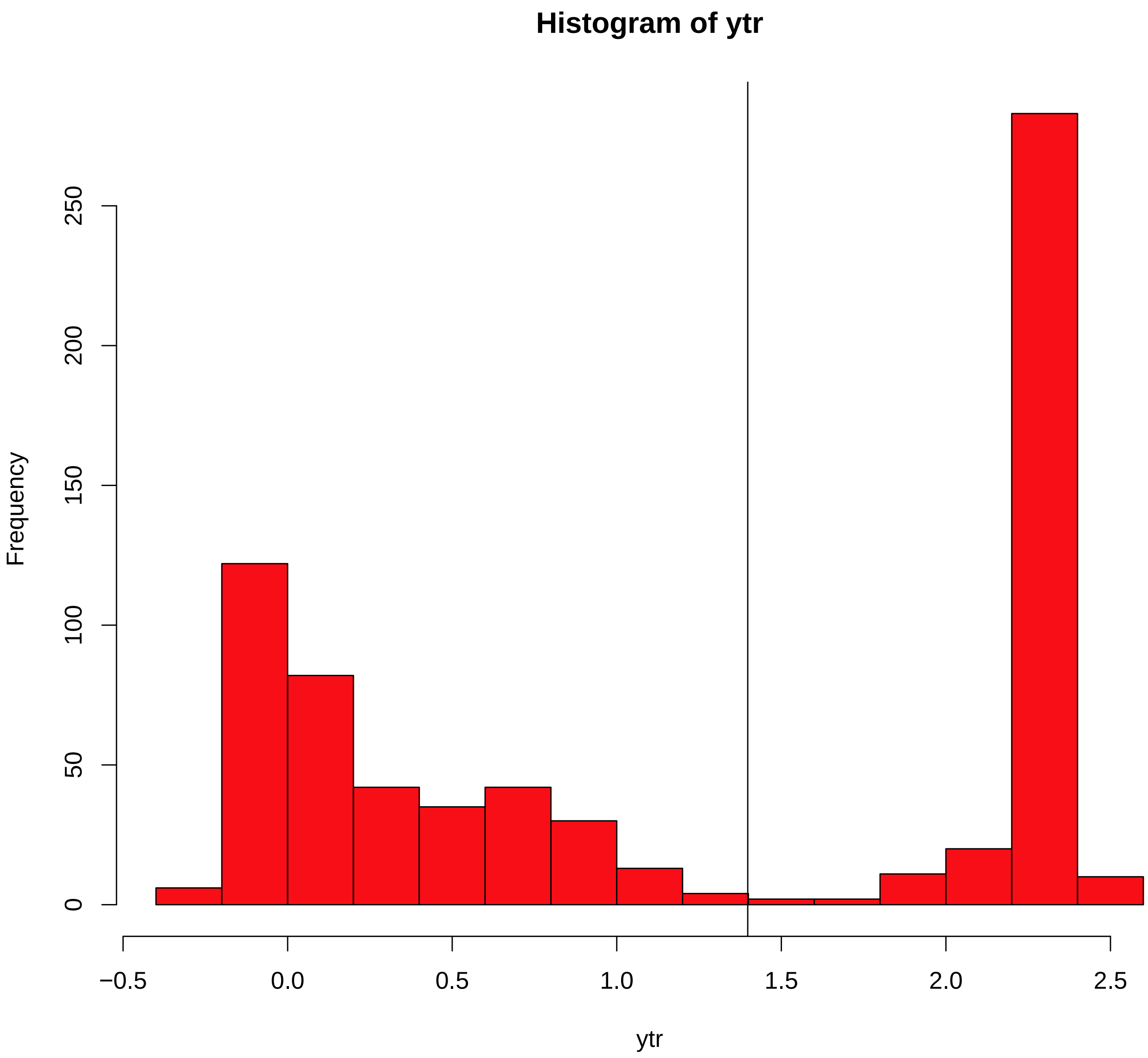

195
182
161
118
88
64
47
37
30
22
13
9
4
3
2
2
1
1
1
1
1
10−fold CV
Test
Mean−Squared Error
0.2
0.4
0.6
0.8
1.0
−10
−8
−6
−4
−2
log(Lambda)

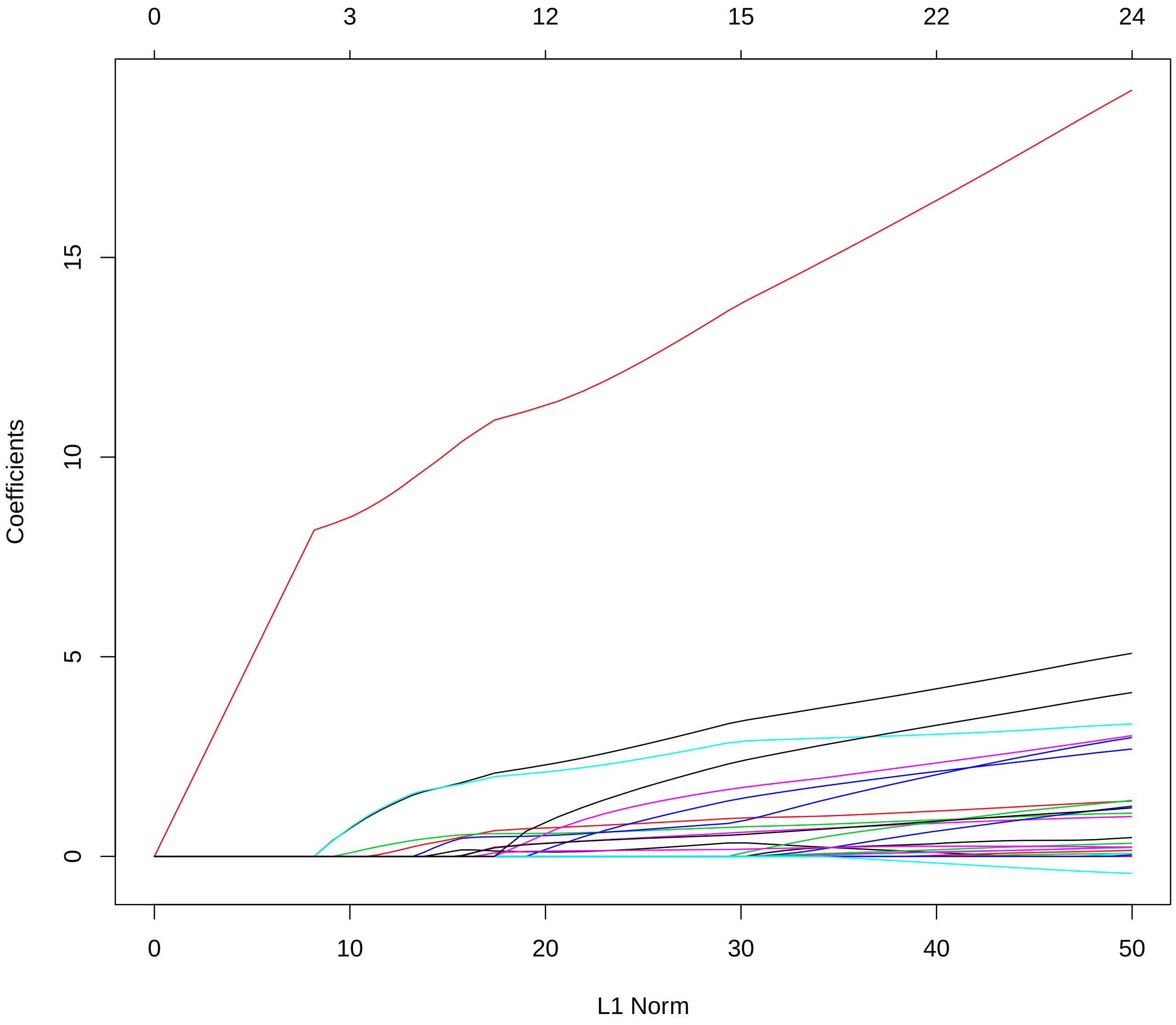


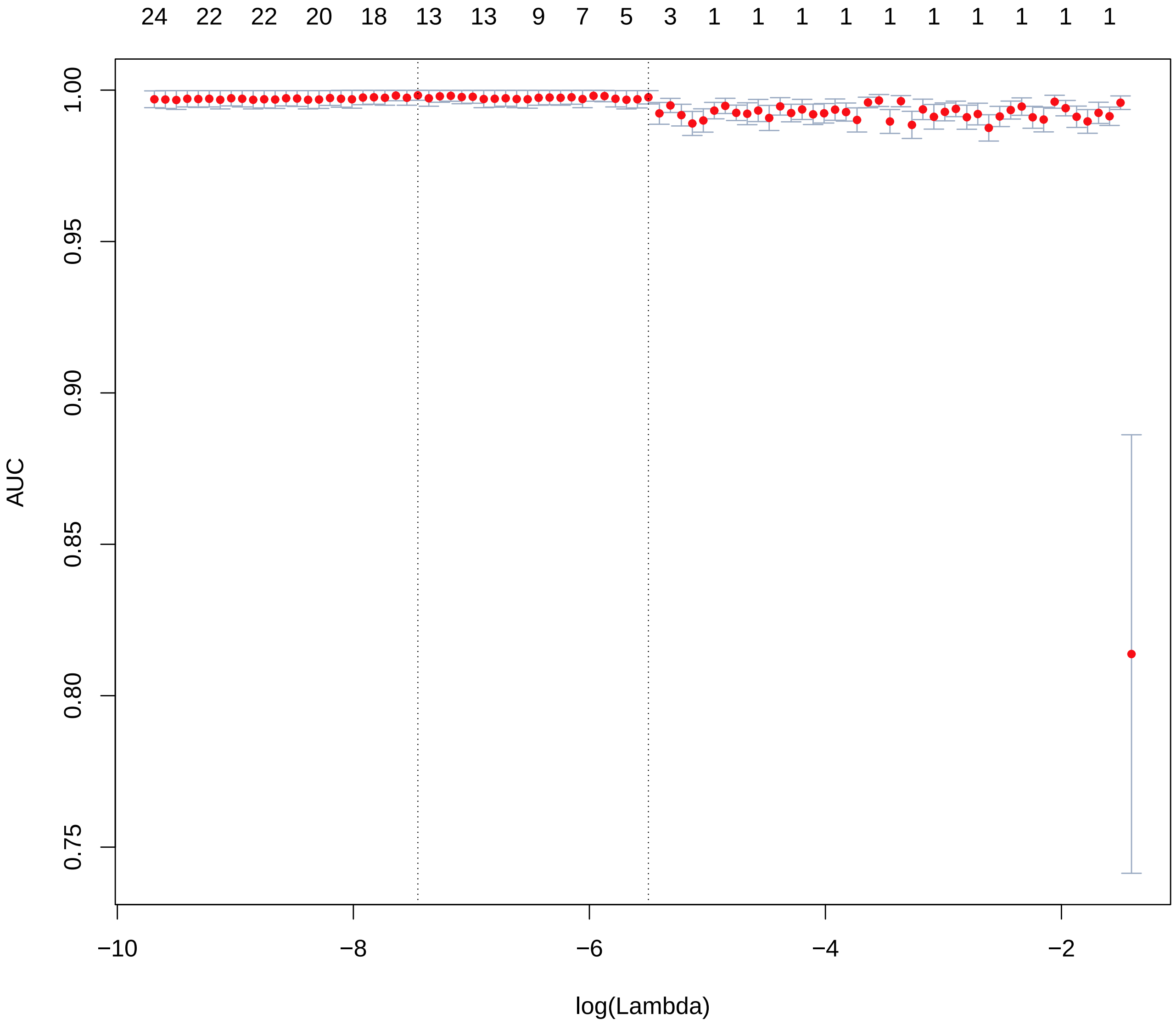
```
> plot(fit,xvar="lambda")
```



```
> glmnet.control(fdev=0)
> fit=glmnet(xtr,ytr,standardize=FALSE,lower=-0.5,upper=0.5)
> plot(fit,xvar="lambda",ylim=c(-1,2))
> hist(ytr,col="red")
> abline(v=log(25,10))
> ytrb=ytr>log(25,10)
> table(ytrb)
ytrb
FALSE  TRUE
  376   328
> fitb=glmnet(xtr,ytrb,family="binomial",standardize=FALSE)
> cv.fitb=cv.glmnet(xtr,ytrb,family="binomial",
               standardize=FALSE,type="auc")
> plot(fitb)
> plot(cv.fitb)
```

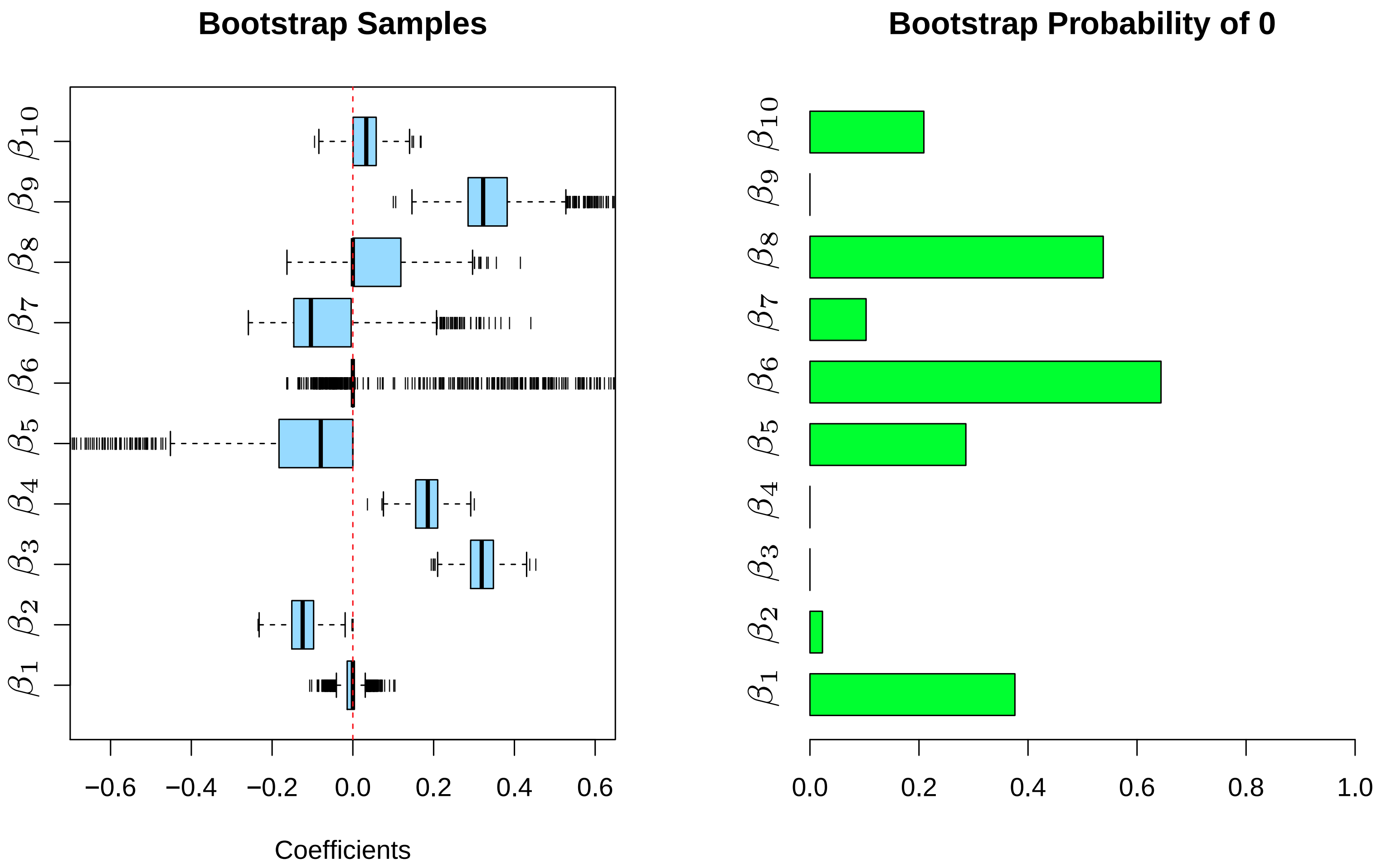

Histogram of ytr
Frequency
0
50
100
150
200
250
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
ytr


0
3
12
15
22
24
Coefficients
0
5
10
15
0
10
20
30
40
50
L1 Norm


24 22 22 20 18 13 13 9 7 5 3 1 1 1 1 1 1 1 1 1 1
AUC
1.00
0.95
0.90
0.85
0.80
0.75
−10
−8
−6
−4
−2
log(Lambda)

Inference?

- Can become Bayesian! Lasso penalty corresponds to Laplacian prior. However, need priors for everything, including λ (variance ratio). Easier to bootstrap, with similar results.
- Covariance Test. Very exciting new developments here: "A Significance Test for the Lasso" — Lockhart, Taylor, Ryan Tibshirani and Rob Tibshirani (2013)



Bootstrap Samples

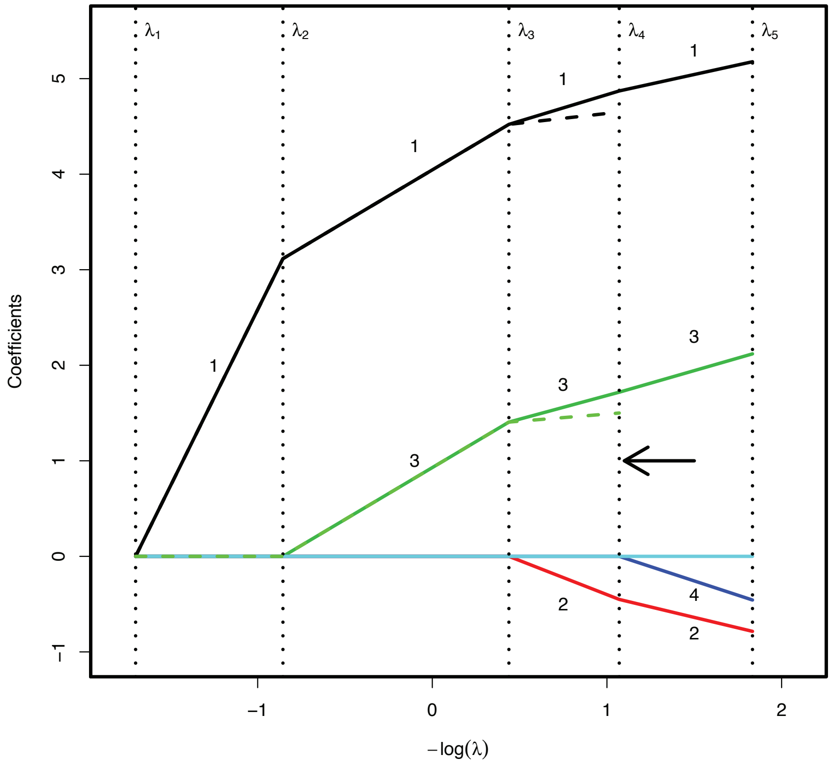
Bootstrap Probability of 0

Covariance Test

- We learned from the LARS project that at each step (knot) we spend one additional degree of freedom.
- This test delivers a test statistic that is Exp(1) under the null hypothesis that the included variable is noise, but all the earlier variables are signal.

"A Significance Test for the Lasso"— Lockhart, Taylor, Ryan Tibshirani and Rob Tibshirani (2013)

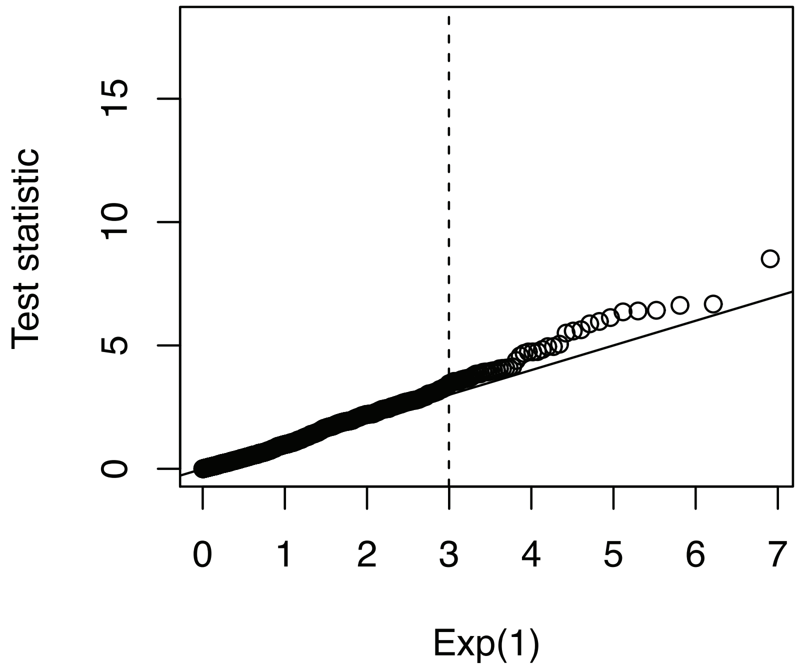
- Suppose we want a p-value for predictor 2, entering at step 3.
- Compute "covariance" at λ_4: $\langle \mathbf{y}, \mathbf{X}\hat{\beta}(\lambda_4)\rangle$
- Drop X_2, yielding active set A; refit at λ_4, and compute covariance at λ_4: $\langle \mathbf{y}, \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda_4)\rangle$



Covariance Statistic and Null Distribution

Under the null hypothesis that all signal variables are in the model:

$$\frac{1}{\sigma^2} \cdot \Big(\langle \mathbf{y}, \mathbf{X}\hat{\beta}(\lambda_{j+1}) \rangle - \langle \mathbf{y}, \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda_{j+1}) \rangle \Big) \to \mathrm{Exp}(1) \text{ as } p, n \to \infty$$



Summary and Generalizations

Many problems have the form

$$\min_{\{\beta_j\}_1^p} \left[R(y, \beta) + \lambda \sum_{j=1}^{p} P_j(\beta_j) \right].$$

- If R and P_j are convex, and R is differentiable, then coordinate descent converges to the solution (Tseng, 1988).
- Often each coordinate step is trivial. E.g. for lasso, it amounts to soft-thresholding, with many steps leaving $\hat{\beta}_j = 0$.
- Decreasing λ slowly means not much cycling is needed.
- Coordinate moves can exploit sparsity.

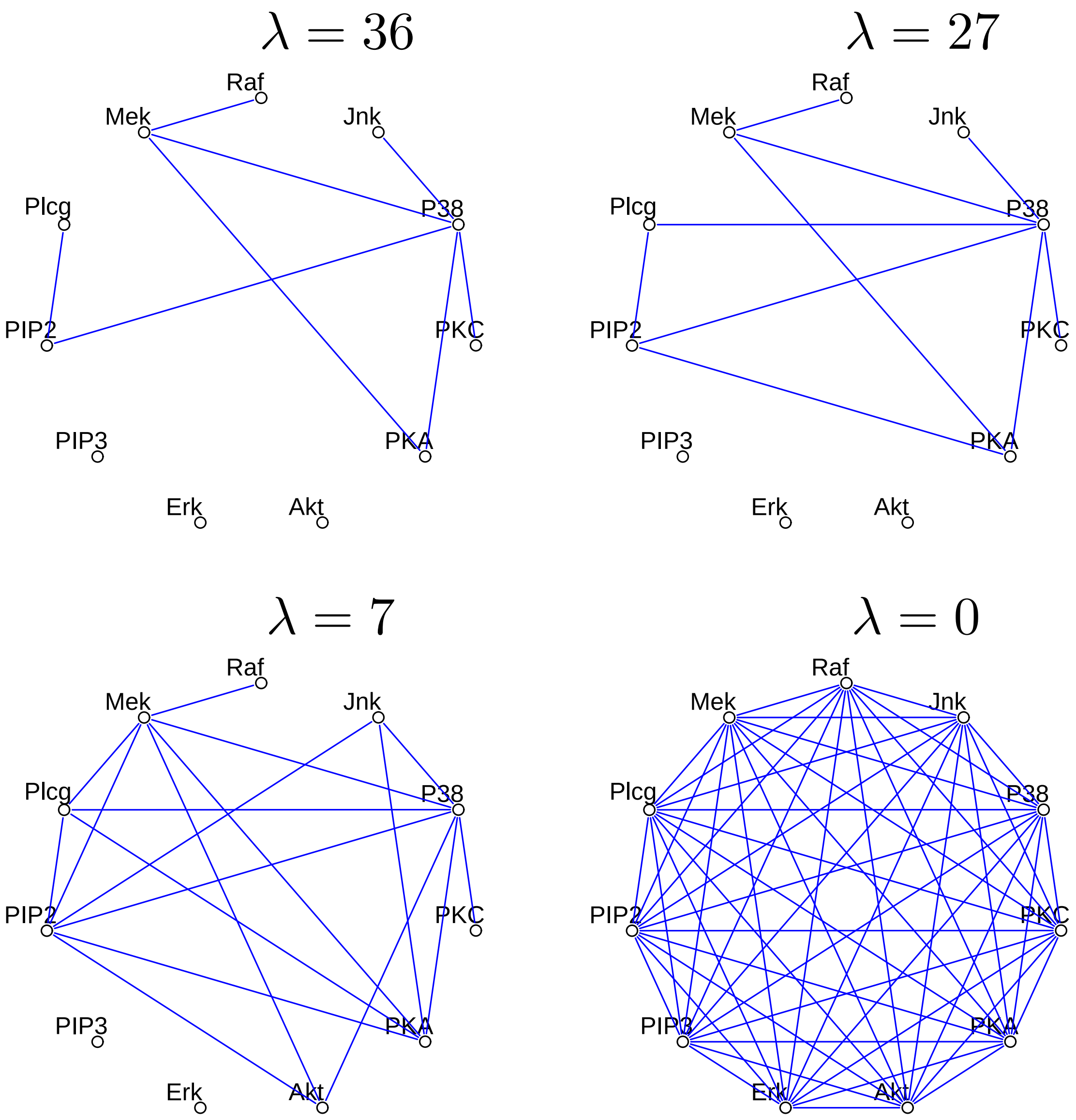
Other Applications

Undirected Graphical Models — learning dependence structure via the lasso. Model the inverse covariance $\mathbf{\Theta}$ in the Gaussian family with L_1 penalties applied to elements.

$$\max_{\mathbf{\Theta}} \log\det\mathbf{\Theta} - \mathrm{Tr}(\mathbf{S}\mathbf{\Theta}) - \lambda||\mathbf{\Theta}||_1$$

Modified block-wise lasso algorithm, which we solve by coordinate descent (FHT 2007). Algorithm is very fast, and solve moderately sparse graphs with 1000 nodes in under a minute.

Example: flow cytometry - $p = 11$ proteins measured in $N = 7466$ cells (Sachs et al 2003) (next page)



$\lambda = 36$

$\lambda = 27$

$\lambda = 7$

$\lambda = 0$

Grouped Lasso (Yuan and Lin, 2007, Meier, Van de Geer, Buehlmann, 2008) — each term $P_j(\beta_j)$ applies to *sets* of parameters:

$$\sum_{j=1}^{J} ||\beta_j||_2.$$

Example: each block represents the levels for a categorical predictor.

Leads to a block-updating form of coordinate descent.

Overlap Grouped Lasso (Jacob et al, 2009) Consider the model

$$\eta(X) = X_1\beta_1 + X_1\theta_1 + X_2\theta_2$$

with penalty

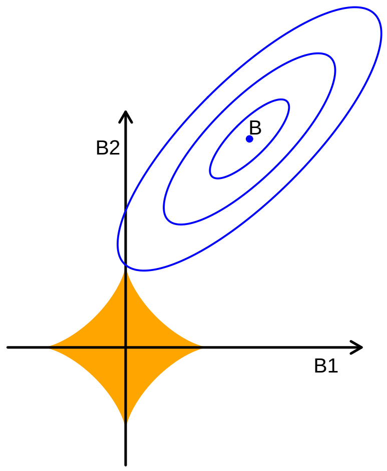
$$|\beta_1| + \sqrt{\theta_1^2 + \theta_2^2}$$

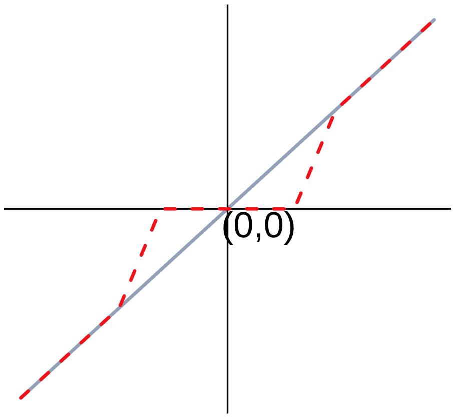
Note: Coefficient of X_1 is nonzero if either group is nonzero; allows one to enforce hierarchy.

- Interactions with weak or strong hierarchy — interaction present only when main-effect(s) are present (w.i.p. with student Michael Lim)
- Sparse additive models — overlap linear part of spline with non-linear part. Allows ”sticky” *null term*, *linear term*, or *smooth term* (with varying smoothness) for each variable. (w.i.p with student Alexandra Chouldechova)

Sparser than Lasso — Concave Penalties

Many approaches. Mazumder, Friedman and Hastie (2010) propose family that bridges ℓ_1 and ℓ_0 based on MC+ penalty (Zhang 2010), and a coordinate-descent scheme for fitting model paths, implemented in SPARSENET

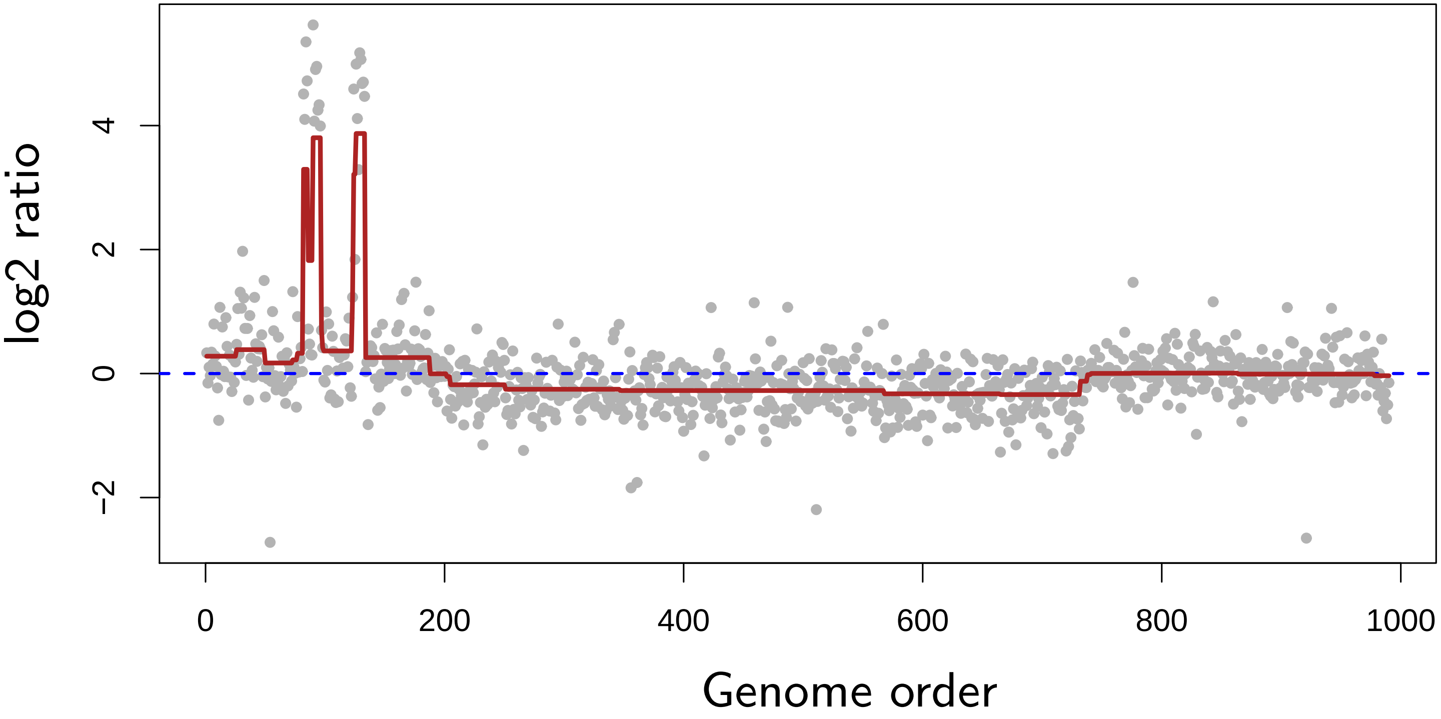




CGH modeling and the fused lasso. Here the penalty has the form

$$\sum_{j=1}^{p} |\beta_j| + \alpha \sum_{j=1}^{p-1} |\beta_{j+1} - \beta_j|.$$

This is not additive, so a modified coordinate descent algorithm is required (FHT + Hoeffling 2007).

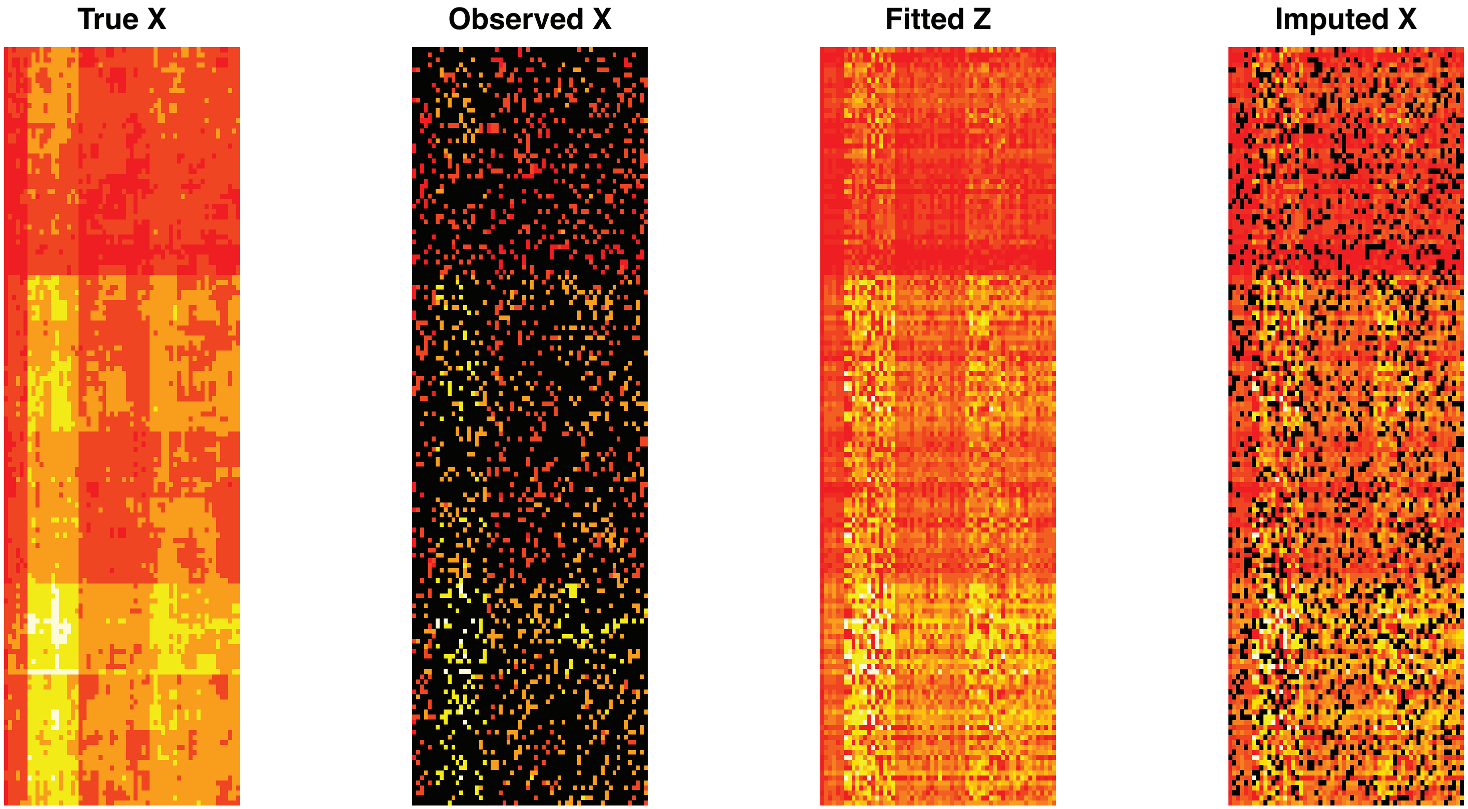


Matrix Completion

- Observe matrix X with (many) missing entries.
- Inspired by SVD, we would like to find $Z_{n\times m}$ of (small) rank r such that training error is small.

$$\min_Z \sum_{\text{Observed}(i,j)} (X_{ij} - Z_{ij})^2 \quad \text{subject to } \operatorname{rank}(Z) = r$$

- We would then impute the missing X_{ij} with Z_{ij}
- Only problem — this is a nonconvex optimization problem, and unlike SVD for complete X, no closed-form solution.



True X

Observed X

Fitted Z

Imputed X

Nuclear norm and SoftImpute

Use convex relaxation of rank (Candes and Recht, 2008, Mazumder, Hastie and Tibshirani, 2010)

$$\min_Z \sum_{\text{Observed}(i,j)} (X_{ij} - Z_{ij})^2 + \lambda ||Z||_*$$

where *nuclear norm* $||Z||_*$ is the sum of singular values of Z.

- Nuclear norm is like the lasso penalty for matrices.
- Solution involves iterative soft-thresholded SVDs of current completed matrix.

Thank You!