# The lasso: some novel algorithms and applications

*Robert Tibshirani*

*Stanford University*

ASA Bay Area chapter meeting

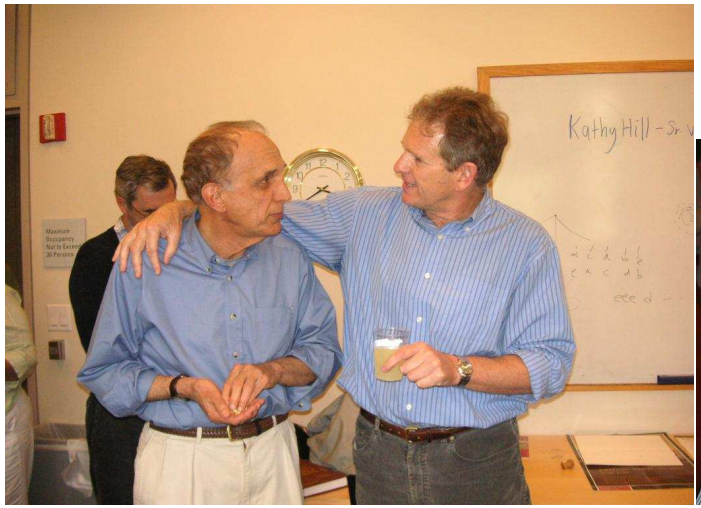
Collaborations with Trevor Hastie, Jerome Friedman, Ryan Tibshirani, Daniela Witten, Bradley Efron, Iain Johnstone, Jonathan Taylor

`Email:tibs@stat.stanford.edu`

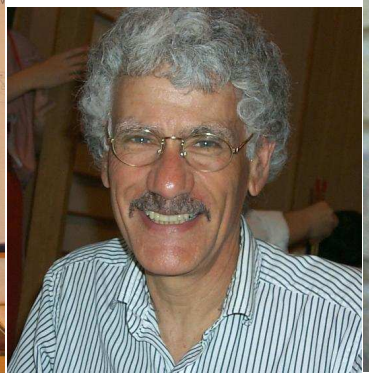
`http://www-stat.stanford.edu/~tibs`

# Plan for talk

- Richard's opening comments and introduction: **30 minutes**
- Talk: **5 minutes**
- Coffee break: **15-20 minutes**
- Questions from Richard: **15 minutes**
- Closing comments: **5 minutes**



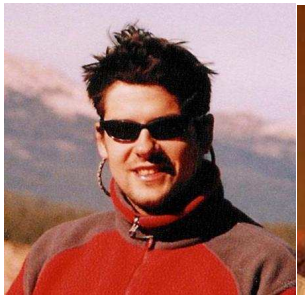

**Brad Efron** **Trevor Hastie**



**Jerome Friedman**

**Daniela Witten**

**Ryan Tibshirani**



**Jonathan Taylor**

**Iain Johnstone**

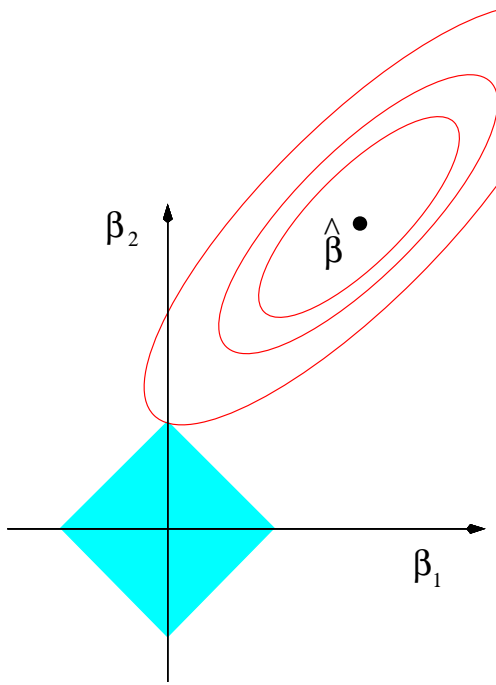
# Linear regression via the Lasso (Tibshirani, 1995)

- Outcome variable $y_i$, for cases $i = 1, 2, \ldots n$, features $x_{ij}$, $j = 1, 2, \ldots p$
- Minimize

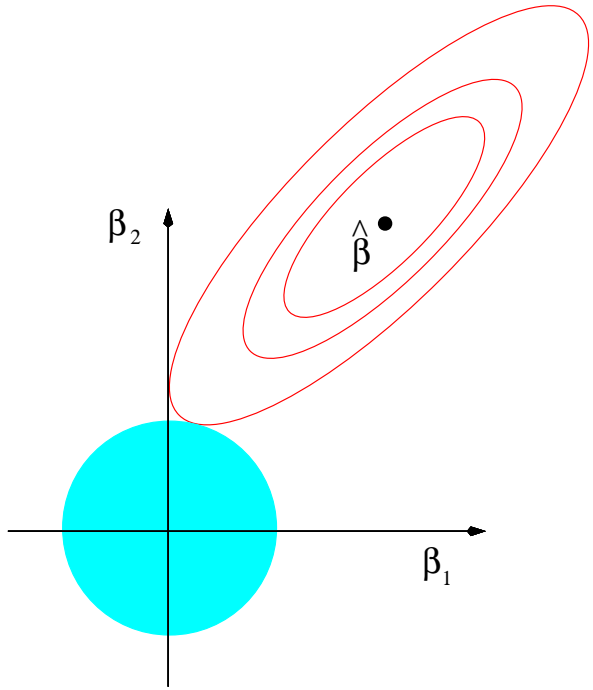
$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Equivalent to minimizing sum of squares with constraint $\sum |\beta_j| \leq s$.
- Similar to **ridge regression**, which has constraint $\sum_j \beta_j^2 \leq t$
- Lasso does variable selection and shrinkage; ridge only shrinks.
- See also "Basis Pursuit" (Chen, Donoho and Saunders, 1998).

# Picture of Lasso and Ridge regression



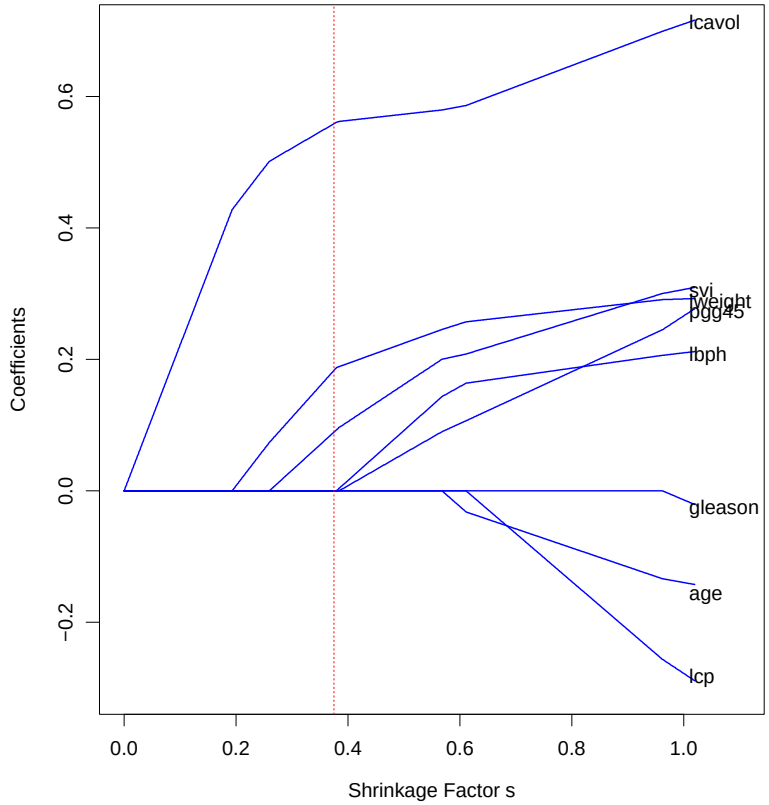

Lasso




Ridge Regression

# Example: Prostate Cancer Data

$y_i = \log$ (PSA), $x_{ij}$ measurements on a man and his prostate



# Estimated coefficients

| Term | Least Squares | Ridge | Lasso |
|---|---|---|---|
| Intercept | 2.465 | 2.452 | 2.468 |
| lcavol | 0.680 | 0.420 | 0.533 |
| lweight | 0.263 | 0.238 | 0.169 |
| age | −0.141 | −0.046 | |
| lbph | 0.210 | 0.162 | 0.002 |
| svi | 0.305 | 0.227 | 0.094 |
| lcp | −0.288 | 0.000 | |
| gleason | −0.021 | 0.040 | |
| pgg45 | 0.267 | 0.133 | |

# Emerging themes

- Lasso ($\ell_1$) penalties have powerful **statistical** and **computational** advantages
- $\ell_1$ penalties provide a natural to encourage/enforce sparsity and simplicity in the solution.
- **"Bet on sparsity principle"** (In the *Elements of Statistical learning*). Assume that the underlying truth is sparse and use an $\ell_1$ penalty to try to recover it. If you're right, you will do well. If you're wrong— the underlying truth is not sparse—, then no method can do well. [Bickel, Buhlmann, Candes, Donoho, Johnstone,Yu ...]
- $\ell_1$ penalties are convex and the assumed sparsity can lead to significant **computational** advantages

# Outline

- New fast algorithm for lasso- Pathwise coordinate descent
- Examples of applications/generalizations of the lasso:
  - **Logistic/multinomial for classification**. Example later of classification from microarray data
  - **Degrees of freedom**
  - **Near-isotonic regression** - a modern take on an old idea
  - **Sparse principal components analysis**
- More topics mentioned at end of talk

# Algorithms for the lasso

- Standard convex optimizer
- Least angle regression (LAR) - Efron et al 2004- computes entire path of solutions. State-of-the-Art until 2008
- Pathwise coordinate descent- new

# Pathwise coordinate descent for the lasso

- Coordinate descent: optimize one parameter (coordinate) at a time.
- How? suppose we had only one predictor. Problem is to minimize

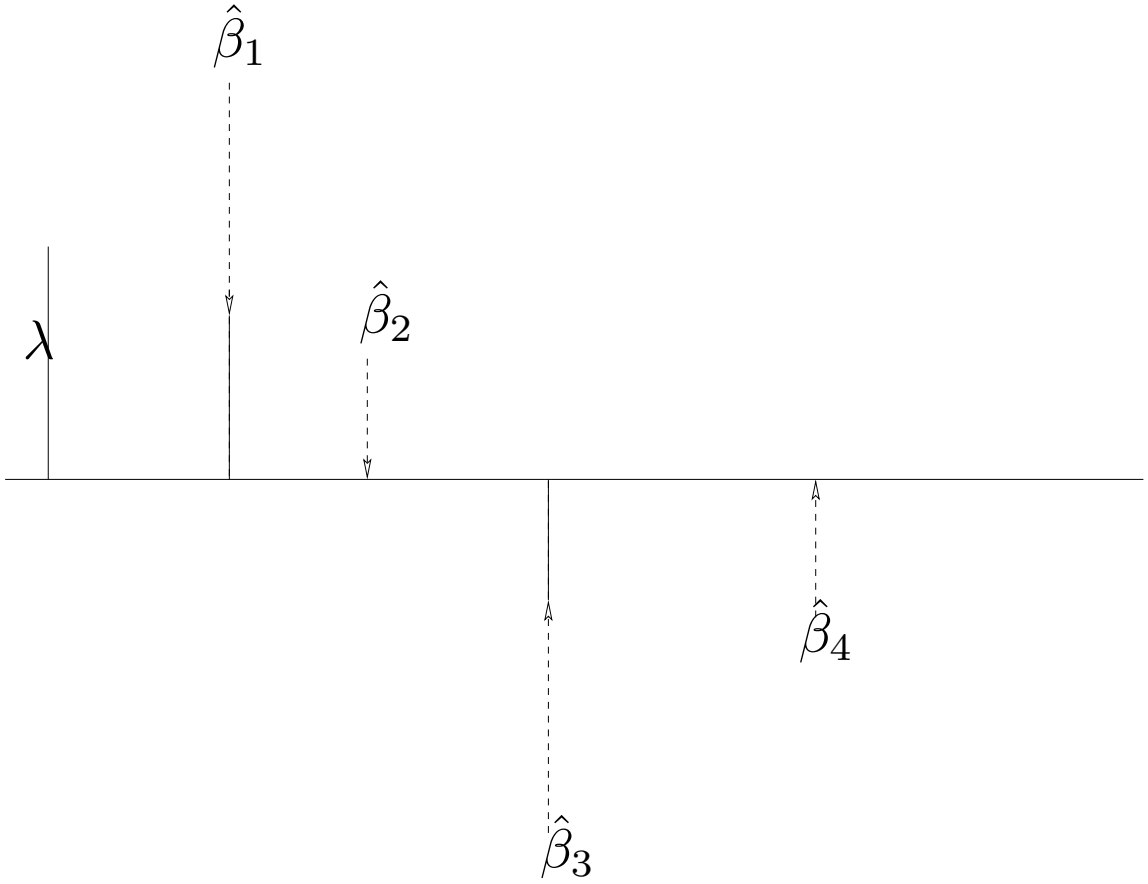
$$\sum_i (y_i - x_i\beta)^2 + \lambda|\beta|$$

- Solution is the soft-thresholded estimate

$$\text{sign}(\hat{\beta})(|\hat{\beta}| - \lambda)_+$$

  where $\hat{\beta}$ is usual least squares estimate.
- Idea: with multiple predictors, cycle through each predictor in turn. We compute residuals $r_i = y_i - \sum_{k \neq j} x_{ik}\hat{\beta}_k$ and applying univariate soft-thresholding, pretending that our data is $(x_{ij}, r_i)$.

# Soft-thresholding



- Turns out that this is coordinate descent for the lasso criterion

$$\sum_i (y_i - \sum_j x_{ij}\beta_j)^2 + \lambda \sum |\beta_j|$$

- like skiing to the bottom of a hill, going north-south, east-west, north-south, etc. [SHOW MOVIE]
- **Too simple?!**

# A brief history of coordinate descent for the lasso

- 1997: Tibshirani's student Wenjiang Fu at University of Toronto develops the "shooting algorithm" for the lasso. Tibshirani doesn't fully appreciate it
- 2002 Ingrid Daubechies gives a talk at Stanford, describes a one-at-a-time algorithm for the lasso. Hastie implements it, makes an error, and Hastie +Tibshirani conclude that the method doesn't work
- 2006: Friedman is the external examiner at the PhD oral of Anita van der Kooij (Leiden) who uses the coordinate descent idea for the Elastic net. Friedman wonders whether it works for the lasso. Friedman, Hastie + Tibshirani start working on this problem. See also Wu and Lange (2008)!

# Pathwise coordinate descent for the lasso

- Start with large value for $\lambda$ (very sparse model) and slowly decrease it
- most coordinates that are zero never become non-zero
- **coordinate descent code for Lasso is just 73 lines of Fortran!**

# Extensions

- Pathwise coordinate descent can be generalized to many other models: logistic/multinomial for classification, graphical lasso for undirected graphs, fused lasso for signals.
- Its speed and simplicity are quite remarkable.
- `glmnet` R package available on CRAN. Fits Gaussian, Binomial/Logistic, Multinomial, Poisson, and Cox models.

# Logistic regression

- Outcome $Y = 0$ or 1; Logistic regression model

$$\log(\frac{Pr(Y=1)}{1-Pr(Y=1)}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \ldots$$

- Criterion is binomial log-likelihood +absolute value penalty
- Example: sparse data. $N = 50,000$, $p = 700,000$.
- State-of-the-art interior point algorithm (Stephen Boyd, Stanford), exploiting sparsity of features : **3.5 hours** for 100 values along path

# Logistic regression

- Outcome $Y = 0$ or 1; Logistic regression model

$$\log(\frac{Pr(Y=1)}{1-Pr(Y=1)}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \ldots$$

- Criterion is binomial log-likelihood +absolute value penalty
- Example: sparse data. $N = 50,000$, $p = 700,000$.
- State-of-the-art interior point algorithm (Stephen Boyd, Stanford), exploiting sparsity of features : **3.5 hours** for 100 values along path
- Pathwise coordinate descent: **1 minute**

# Multiclass classification

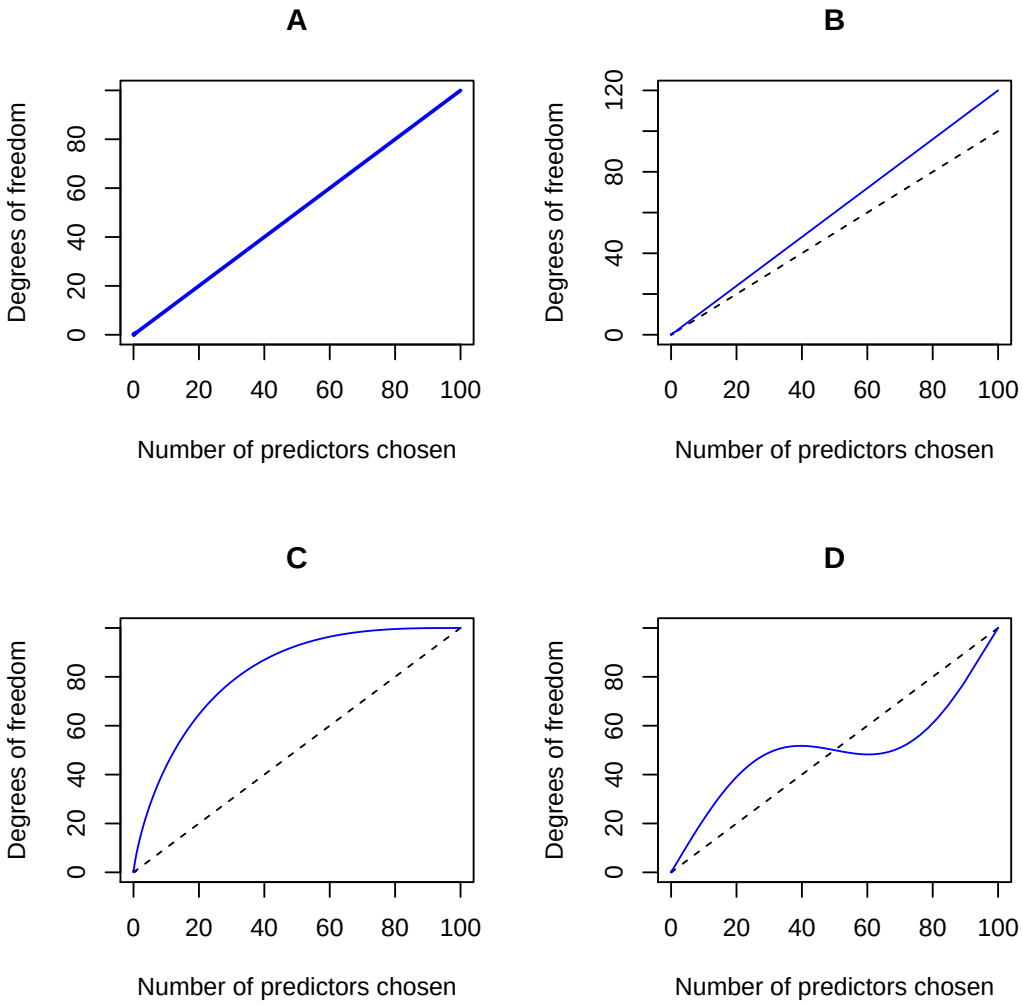
Microarray classification: 16,000 genes, 144 training samples 54 test samples, 14 cancer classes. Multinomial regression model.

| Methods | CV errors out of 144 | Test errors out of 54 | # of genes used |
|---|---|---|---|
| Support vector classifier | 26 (4.2) | 14 | 16063 |
| **Lasso-penalized multinomial** | 17 (2.8) | 13 | 269 |

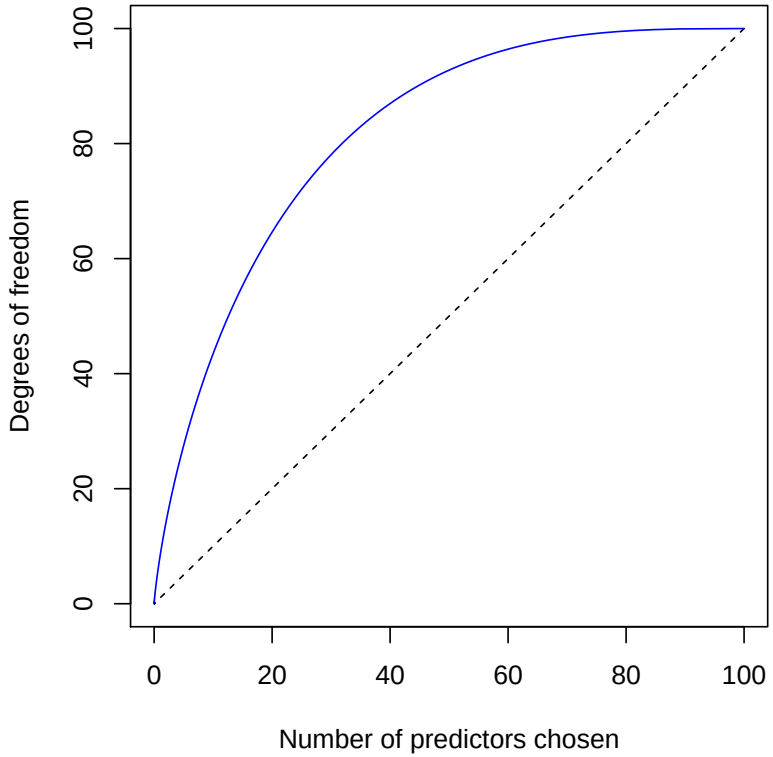
# Degrees of freedom

- Let's first consider all subset regression with total of $p = 100$ predictors.
- Suppose that we apply all-subset regression to obtain the best subset of size $k$.
- Define the degrees of freedom DF of the fit as the average drop in residual sum of squares divided by the error variance
- If we fit a fixed model with $k$ predictors, DF $= k$.
- In all-subset selection, how many DF do we use, as a function of $k$?

# Quiz


A
B
C
D
Degrees of freedom
Number of predictors chosen
0
20
40
60
80
100
120

# DF for all subset regression




Degrees of freedom is $> k$, since we selected the best $k$ out of $p$

# Degrees of freedom of the lasso

**Some magic occurs...**

- Define degrees of freedom using Efron's formula,

$$\mathrm{DF} = \sum_{i=1}^{n} \mathrm{cov}(\hat{y}_i, y_i)/\sigma^2$$

- Apply Stein's lemma: $\sum_i \mathrm{cov}(\hat{y}_i, y_i)/\sigma^2 = \mathrm{E} \sum_i \frac{d\hat{y}_i}{dy_i}$
- For the lasso fit $\hat{\mathbf{y}}(k)$ having $k$ non-zero predictors, this gives

$$\mathrm{DF}(\hat{\mathbf{y}}(k)) \approx k \quad \text{!!!!!!!!!!!!!!!!}$$

Although lasso chooses $k$ predictors adaptively, the coefficients are shrunken just the right amount to make DF equal to $k$!

[Efron, Hastie, **Johnstone**, Tibs]; [Zou, Hastie+Tibs]; [Ryan Tibs+Taylor]

# Near Isotonic regression

**Ryan Tibshirani, Holger Hoefling, Rob Tibshirani (2010)**

- generalization of isotonic regression: data sequence $y_1, y_2, \ldots y_n$.

$$\text{minimize} \sum (y_i - \hat{y}_i)^2 \text{ subject to } \hat{y}_1 \leq \hat{y}_2 \ldots$$

  Solved by Pool Adjacent Violators algorithm.

- Near-isotonic regression:

$$\beta_\lambda = \text{argmin}_{\beta \in \mathcal{R}^n} \frac{1}{2} \sum_{i=1}^{n} (y_i - \beta_i)^2 + \lambda \sum_{i=1}^{n-1} (\beta_i - \beta_{i+1})_+,$$

  with $x_+$ indicating the positive part, $x_+ = x \cdot 1(x > 0)$.

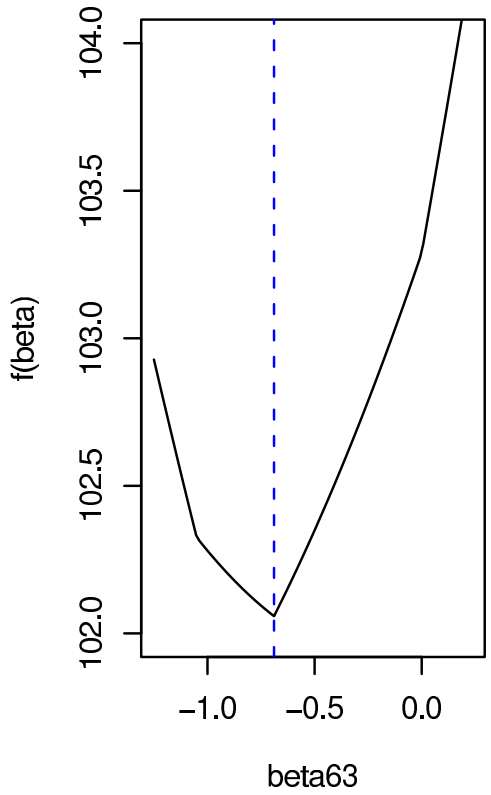
## Near-isotonic regression- continued

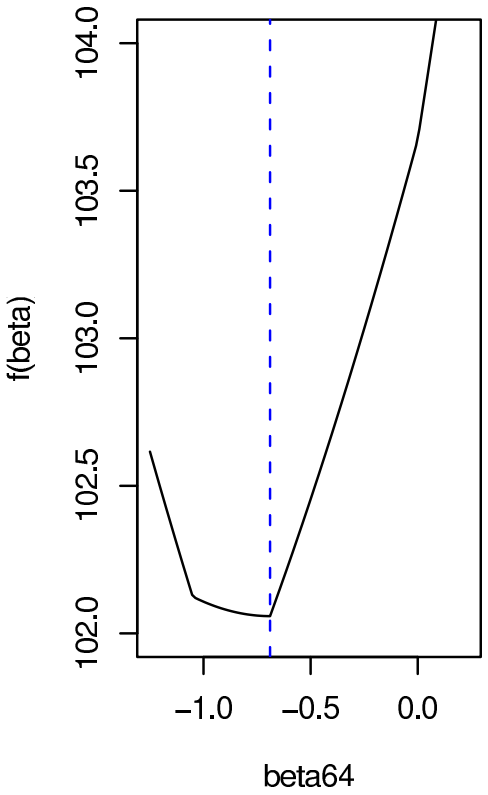
- Convex problem. Solution path $\hat{\beta}_i = y_i$ at $\lambda = 0$ and culminates in usual isotonic regression as $\lambda \to \infty$. Along the way gives **near monotone** approximations.

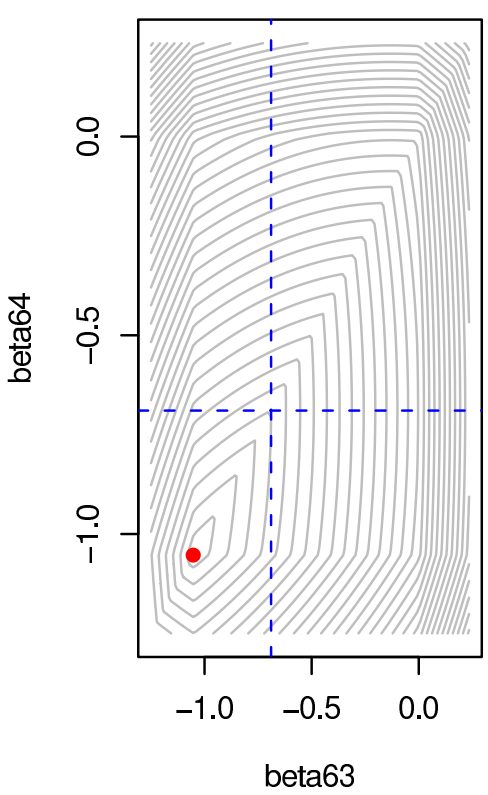
## Numerical approach

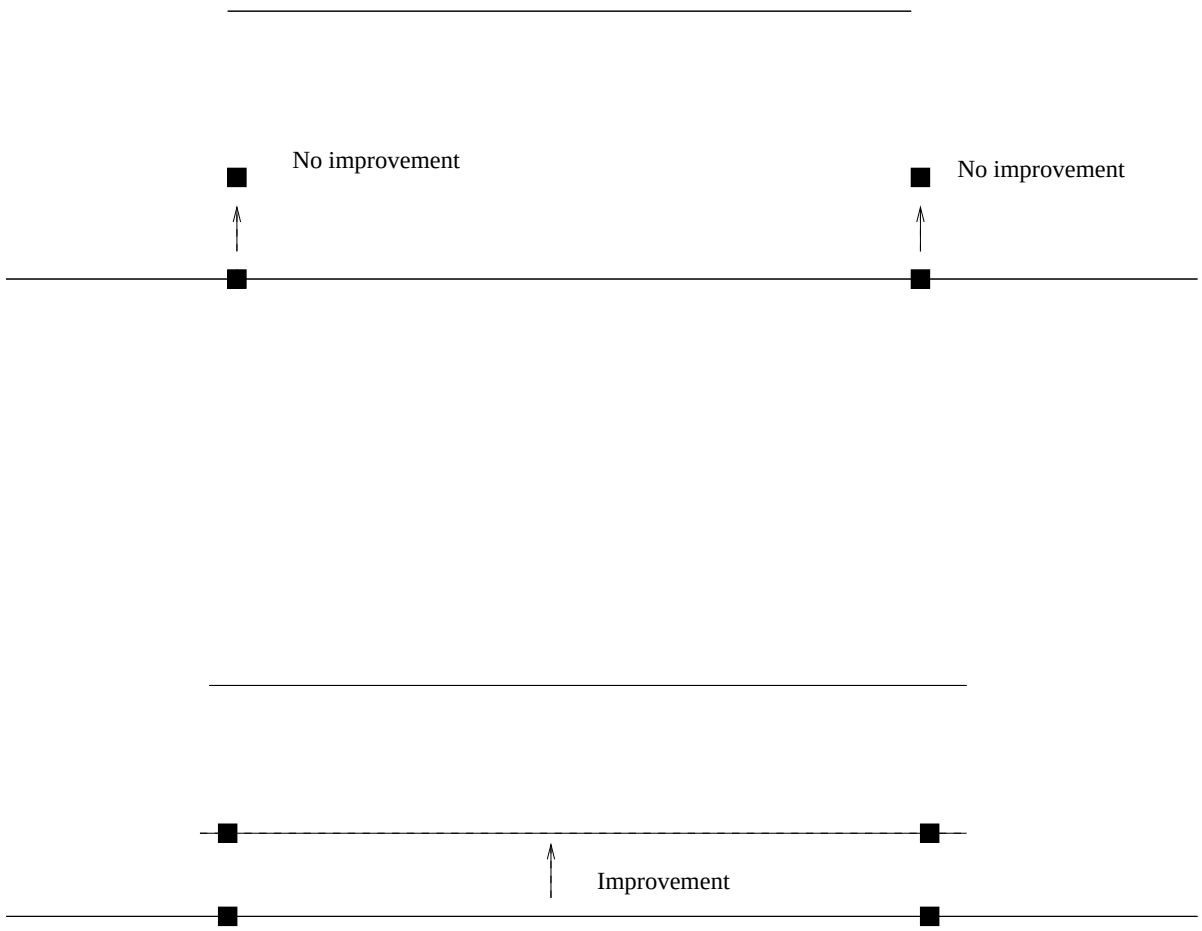
How about using coordinate descent?

- **Surprise!** Although criterion is convex, it is not differentiable, and coordinate descent can get stuck in the "cusps"


f(beta)
102.0
102.5
103.0
103.5
104.0
−1.0
−0.5
0.0
beta63


f(beta)
102.0
102.5
103.0
103.5
104.0
−1.0
−0.5
0.0
beta64


beta64
−1.0
−0.5
0.0
−1.0
−0.5
0.0
beta63


No improvement
No improvement
Improvement

## When does coordinate descent work?

Paul Tseng (1988), (2001)

If

$$f(\beta_1 \dots \beta_p) = g(\beta_1 \dots \beta_p) + \sum h_j(\beta_j)$$

where $g(\cdot)$ is convex and differentiable, and $h_j(\cdot)$ is convex, then coordinate descent converges to a minimizer of $f$.

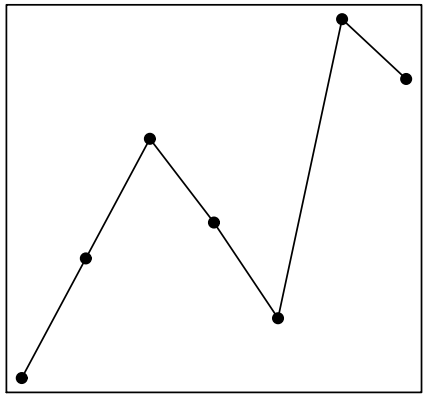
**Non-differential part of loss function must be separable**

## Solution: devise a path algorithm

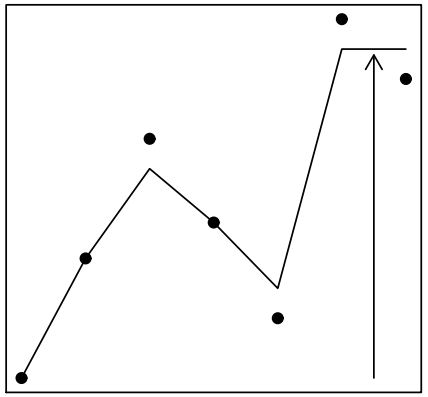
- Simple algorithm that computes the entire path of solutions, a modified version of the well-known **pool adjacent violators**
- Analogous to LARS algorithm for lasso in regression
- Bonus: we show that the degrees of freedom is the number of "plateaus" in the solution. Using results from **Ryan Tibshirani's** PhD work with **Jonathan Taylor**

# Toy example

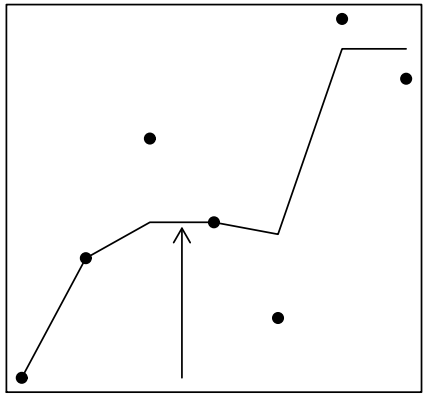
$\lambda = 0$



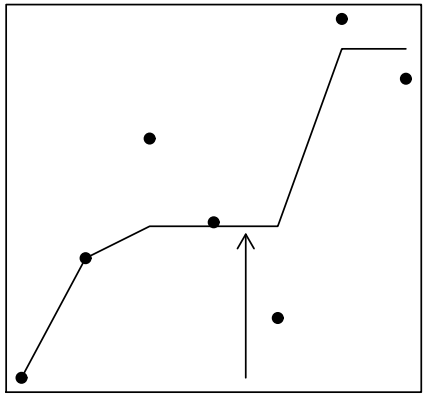
$\lambda = 0.25$



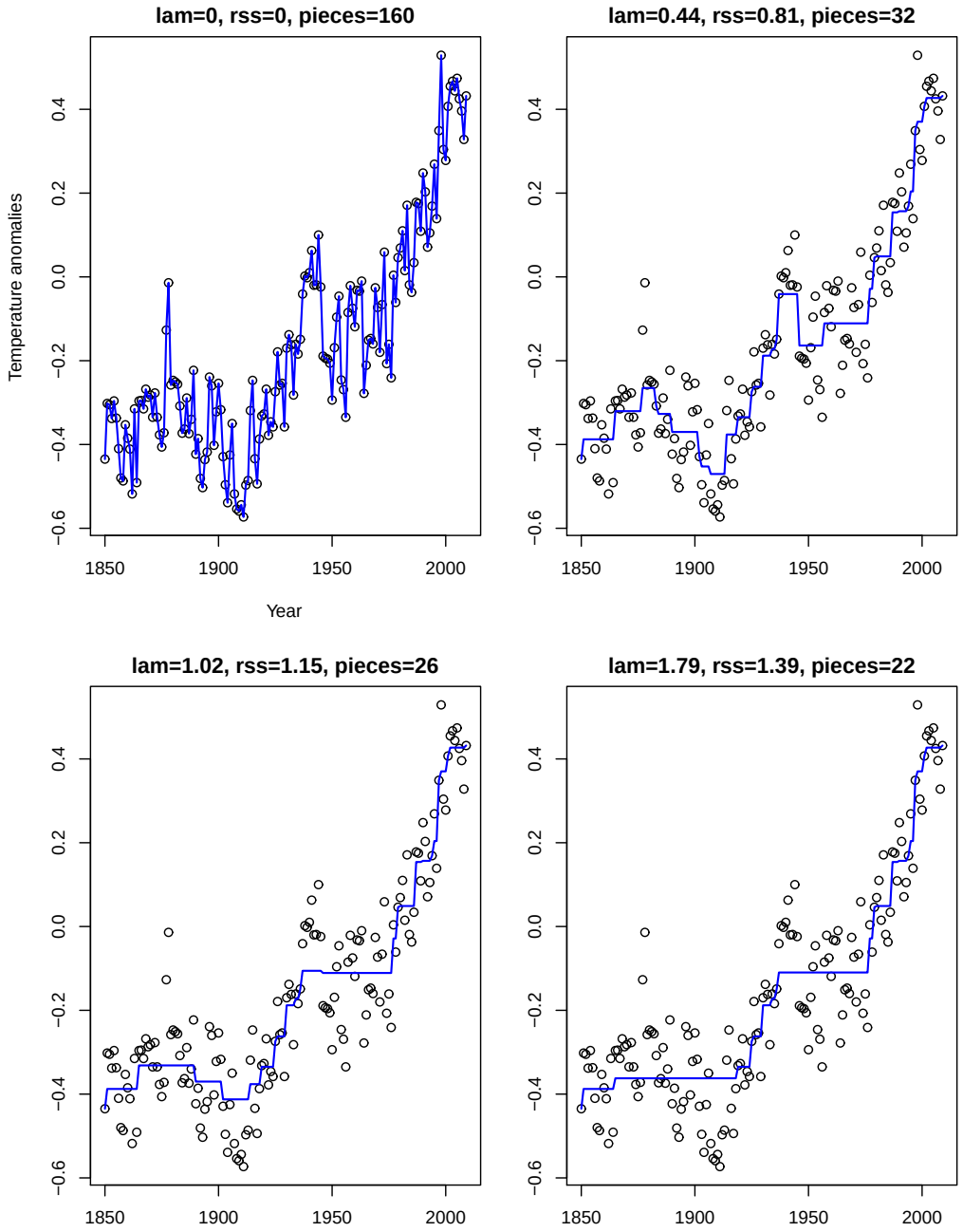
$\lambda = 0.7$



$\lambda = 0.77$



# Global warming data


lam=0, rss=0, pieces=160
lam=0.44, rss=0.81, pieces=32
lam=1.02, rss=1.15, pieces=26
lam=1.79, rss=1.39, pieces=22
Temperature anomalies
Year
−0.6
−0.4
−0.2
0.0
0.2
0.4
1850
1900
1950
2000

# Penalized Matrix Decomposition

Daniela Witten, PhD thesis

Start with $N \times p$ data matrix $\mathbf{X}$.

$$(\mathbf{u}, \mathbf{v}, d) = \arg\min ||\mathbf{X} - d\mathbf{u}\mathbf{v}^T||_F^2 \; s.t. \qquad ||\mathbf{u}||^2 = ||\mathbf{v}||^2 = 1, \; ||\mathbf{u}||_1 \leq c_1, ||\mathbf{v}||_1 \leq c_2,$$

Useful for "sparsifying" a wide variety of multivariate procedures (PCA, CCA, clustering)

Can write as $\arg\max[\mathbf{u}^T\mathbf{X}\mathbf{v}]$ with same constraints.

# Computation of Single-Factor PMD Model

- $\mathbf{u} \leftarrow \frac{S(\mathbf{X}\mathbf{v},\delta_1)}{||S(\mathbf{X}\mathbf{v},\delta_1)||_2}$,
- $\mathbf{v} \leftarrow \frac{S(\mathbf{X}^T\mathbf{u},\delta_2)}{||S(\mathbf{X}^T\mathbf{u},\delta_2)||_2}$.

Here $S(x,t) = \text{sign(x)}(|x| - t)_+$ (soft-threshold operator), $\delta_1$ is chosen so that $||\mathbf{u}||_1 = c_1$; similarly for $\delta_2$.

- Generalizes power method for obtaining first singular vector;
- With $c_1 = \infty$, yields first **sparse principal component** $\hat{v}$ of data.
- Criterion is not convex but is **bi-convex**
- Equivalent to **ScotLass** proposal of Jolliffe, Trendafilov and Uddin, (2003)

# Example: Corpus callosum shape study

569 elderly individuals: measurements of corpus collosum, walking speed and verbal fluency

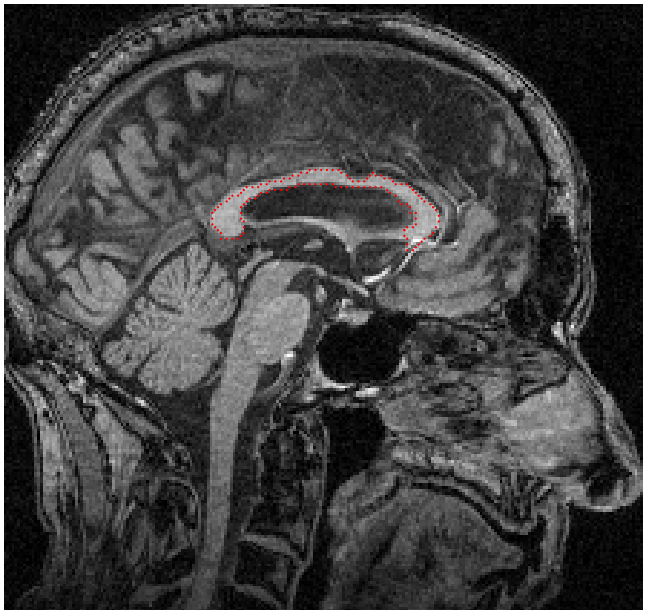
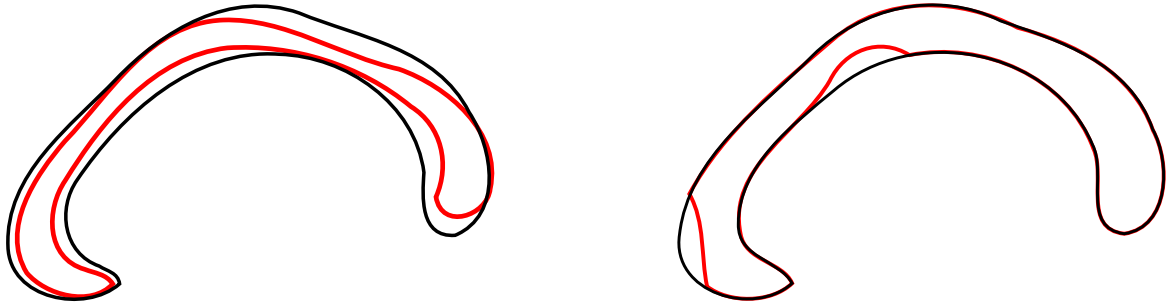
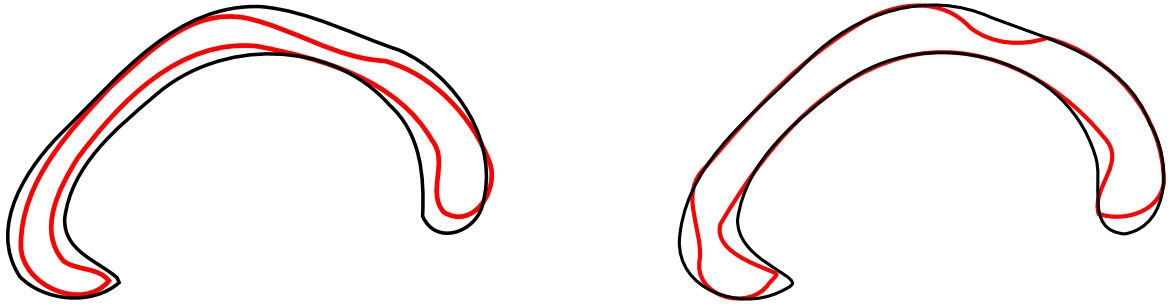


Figure 1: **An example of a mid-aggital brain slice, with the corpus collosum annotated with landmarks.**

**Walking Speed**



**Verbal Fluency**



**Principal Components** **Sparse Principal Components**

# Discussion

- lasso penalties are useful for fitting a wide variety of models with sparsity constraints; pathwise coordinate descent enables to fit these models to large datasets for the first time
- In CRAN: coordinate descent in R **glmnet**- linear regression, logistic, multinomial, Cox model, Poisson
- Also: LARS, nearIso, cghFLasso, glasso
- PMA (penalized multivariate analysis) R package
- Matlab software for glm.net and matrix completion
  http://www-stat.stanford.edu/∼ tibs/glmnet-matlab/
  http://www-stat.stanford.edu/∼rahulm/SoftShrink

# Ongoing work in lasso/sparsity

- grouped lasso (Yuan and Lin) and many variations
- multivariate- principal components, canonical correlation, clustering (Witten and others)
- matrix-variate normal (Genevera Allen)
- Matrix completion- Candes, Mazumder, Hastie+Tibs
- graphical models, graphical lasso (Yuan+Lin, Friedman, Hastie+Tibs, Mazumder, Witten, Simon; Peng, Wang et al)
- Compressed sensing (Candes and co-authors)
- "Strong rules" (Tibs et al 2010) provide a 5-80 fold speedup in computation, with no loss in accuracy
- Interactions (Bien, Simon, Lim/Hastie)

## Some challenges

- develop tools and theory that allow these methods to be used in statistical practice: standard errors, p-values and confidence intervals that account for the adaptive nature of the estimation.
- while it's fun to develop these methods, as statisticians, our ultimate goal is to provide better answers to scientific questions

# References