The Elements of Statistical Learning

Ch3: Liner Methods for Regression - Forward type selection methods

Philip Lin

Data Science in Hsinchu

2015.08.19

DSHC is a non-profit studying group.

This slide is created to help us to discuss the content of "Hastie, T., Tibshirani, R. & Friedman, J., 2009. The Elements of Statistical Learning" and is not used in any profit-oriented activity.

Framework

Today, we talk about foreward type selection methods

- Boosting (L_2Boost)
  - Gradient Boosting Machine (Friedman, 1999)
  - L_2boost in high dimension (Peter Buhlmann, 2006)
- Forward type selections
  - Incremental Forward Stagewise (FS_ϵ)
  - FS_0 versus L_2Boost
  - Lasso versus FS_0 (Hastie, 2007)
- An unified algorithm: Least Angle Regression (Efron, 2003)

keywords: steepest descent, monotone Lasso, path-based algorithm

In today's discussion, I take the following thesis as main reference

"Ehrlinger, J. (2011). Regularization: Stagewise regression and bagging. Ph.D. thesis, Case Western Reserve Univ., Cleveland, OH. MR2873516"

To avoid massive footenotes for citation, all uncited figures and equations come from this thesis.

Boosting

Steepest Descent

- Suppose a real-valued, differentiable function $\psi : \mathbb{R}^n \to \mathbb{R}$ decreases fastest from a point z if one goes in the direction $-\nabla\psi(z)$ of the negative gradient of ψ at z

Gradient Descent *Steepest Descent*

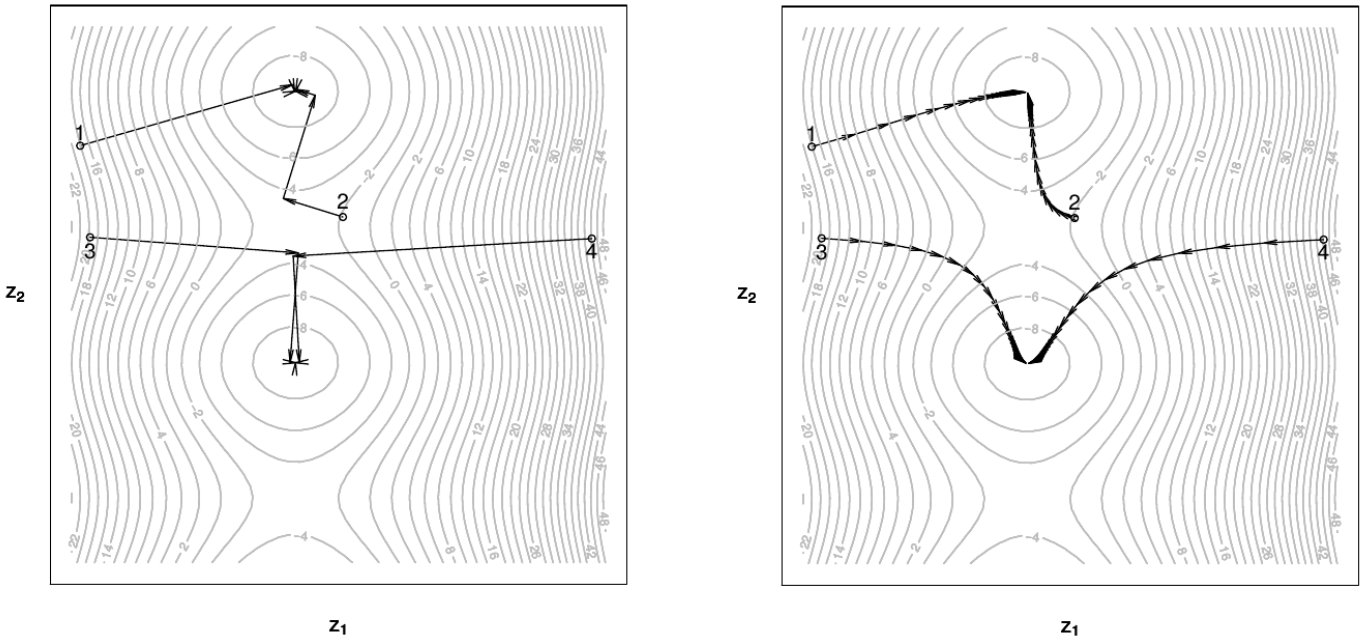
Algorithm 2.1 *Steepest Descent*

1: Initialize $\mathbf{z}_0$; set $m = 0$
2: **while** $\nabla\psi(\mathbf{z}_{m-1}) \neq \mathbf{0}$ **do**
3: Update $m \leftarrow m + 1$
4: Set $\mathbf{g}_m = -\nabla\psi(\mathbf{z}_{m-1})$
5: Find ρ_m by a line search: $\rho_m = \arg\min_{\rho \geq 0} \psi(\mathbf{z}_{m-1} + \rho\mathbf{g}_m)$
6: Let $\mathbf{z}_m = \mathbf{z}_{m-1} + \rho_m\mathbf{g}_m$
7: **end while**

- Exact line search in line 5 is not necessary to ensure convergence. Instead, one can replace line 5 with $z_m = z_{m-1} + \rho^*_m g_m$ for any $0 < \rho^*_m \leq \rho_m$, where ρ^*_m is determined by setting the directional directive equal to zero: $\psi'(z_{m-1} + \rho^*_m g_m) = 0$. That is, at step m,

$$\frac{\partial}{\partial\rho}\psi(z_{m-1} + \rho g_m) = \nabla\psi(z_m)^T \frac{\partial}{\partial\rho}\{z_{m-1} + \rho g_m\} = \nabla\psi(z_m)^T g_m = 0$$

we see that $\rho = \rho^*_m$ should be chosen such that g_m is orthogonal to $\nabla\psi(z_m)$



- The directional derivative is often computationally expensive to evaluate, especially in high dimensions.
- We can set the descent parameter to a fixed value ρ, small enough to ensure that $\rho < \rho_m^*$ for all m. (Why ?)

Functional gradient descent (Friedman, 1999)

- Recall statistical decision theory in Ch1, the goal is to approximate the unknown function $F(x)$ (i.e., regression assumption)

$$\arg\min_{F} \; E_x E_{y|\boldsymbol{x}} \Big(L(y, F(\boldsymbol{x}))|\boldsymbol{x} \Big)$$

where $L(y, F(\boldsymbol{x}))$ is a prespecified loss function.

Technically Given training data set, the optimization problem becomes

$$F^*(\boldsymbol{x}) = \arg\min_{F} \; \psi(F(\boldsymbol{x})) = \arg\min_{F} \; E_{y|\boldsymbol{x}} \Big(L(y, F(\boldsymbol{x}))|\boldsymbol{x} \Big)$$

we solve this by **functional gradient descent** (cf: steepest descent)

$$F^*(\boldsymbol{x}) = \sum_{m=0}^{M} \rho_m g_m(\boldsymbol{x})$$

with

$$\begin{aligned} g_m(\boldsymbol{x}) &= - \left[\frac{\partial \psi(F(\boldsymbol{x}))}{\partial F(\boldsymbol{x})} \right]_{F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x})} \\ &= - \left[\frac{\partial E_{y|\boldsymbol{x}} \big[L(y, F(\boldsymbol{x}))|\boldsymbol{x} \big]}{\partial F(\boldsymbol{x})} \right]_{F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x})} \\ &= - E_{y|\boldsymbol{x}} \left[\frac{\partial L(y, F(\boldsymbol{x}))}{\partial F(\boldsymbol{x})} \middle| \boldsymbol{x} \right]_{F(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x})} \\ \rho_m &= \arg\min_{\rho} E_{y|\boldsymbol{x}} L(y, F_{m-1}(\boldsymbol{x}) - \rho g_m(\boldsymbol{x})) \end{aligned}$$

Friedman, J.H., 2001. Greedy function approximation: A gradient boosting machine. The Annals of Statistics, 29(5), pp.1189 - 1232.

In statistical view

- Usually, the loss function L is assumed to be smooth and convex in the second argument, to ensure that the gradient method works well. e.g.
  1. $L(y, F) = (y - F)^2/2$ with $y \in \mathbb{R}$: L_2Boost
  2. $L(y, F) = \exp(yF)$ with $y \in \{-1, 1\}$: AdaBoost
  3. $L(y, F) = \log_2(1 + \exp(-2yF))$ with $y \in \{-1, 1\}$: LogitBoost
- Assume an additive expansion for $F(\boldsymbol{x})$ of the form

$$F(\boldsymbol{x}; \{\beta_m, \boldsymbol{a}_m\}_1^M) = \sum_{m=1}^{M} \beta_m h(\boldsymbol{x}; \boldsymbol{a}_m)$$

  such expansions are at heart of many function approximation method such as **neural networks**, **radial basis function**, **MARS**, **wavelets**, **support vector machines** and **CART**.
- By the additive expansion assumption, we can parameterize a function $F(\boldsymbol{x})$ by **finite** parameters $\{\beta_m, \boldsymbol{a}_m\}_1^M$.

Thus

$$\{\beta_m, \boldsymbol{a}_m\}_1^M = \underset{\{\beta'_m, \boldsymbol{a}'_m\}_1^M}{\arg\min} \sum_{i=1}^{n} L\Big(y_i, \sum_{m=1}^{M} \beta'_m h(\boldsymbol{x}; \boldsymbol{a}'_m)\Big)$$

By the greedy stagewise approach. For $m = 1, 2, \ldots, M$

$$(\beta_m, \boldsymbol{a}_m) = \arg\min_{\beta, \boldsymbol{a}} \sum_{i=1}^{n} L\Big(y_i, F_{m-1}(\boldsymbol{x}_i) + \beta h(\boldsymbol{x}_i; \boldsymbol{a})\Big)$$

and then

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \beta_m h(\boldsymbol{x}; \boldsymbol{a}_m) = \sum_{i=1}^{m} \beta_i h(\boldsymbol{x}; \boldsymbol{a}_i)$$

- In greedy stagewise approach, we further restrict our parameter space (sequential relationship and **local optimal** in each step) in order to make a connection with the gradient descent algorithm.
- note that the base learner $h(\boldsymbol{x}; \boldsymbol{a}_m)$ can be seen as steepest descent direction and is parameterized by $\boldsymbol{a}_m$.
- When (the second) optimization is difficult to obtain, we can choose $\boldsymbol{a}_m$ that is closet to the negative gradient.

$$\boldsymbol{a}_m = \arg\min_{\boldsymbol{a}} \sum_{i=1}^{n} \Big[g_m(\boldsymbol{x}_i) - h(\boldsymbol{x}_i; \boldsymbol{a})\Big]^2 \quad , g_m(\boldsymbol{x}_i) = -L'(y_i, F_{m-1}(\boldsymbol{x}_i))$$

Algorithm 2.2 *Gradient_Boost*

1: Initialize $F_0(\mathbf{x}) = \hat{\rho}$, where $\hat{\rho} = \arg\min_{\rho \geq 0} \sum_{i=1}^{n} L(y_i, \rho)$
2: **for** $m = 1, \ldots, M$ **do**
3: $\quad g_m(\mathbf{x}_i) = -L'(y_i, F_{m-1}(\mathbf{x}_i))$
4: $\quad \mathbf{a}_m = \arg\min_{\mathbf{a} \in \mathbf{A}} \sum_{i=1}^{n} [g_m(\mathbf{x}_i) - h(\mathbf{x}_i; \mathbf{a})]^2$
5: $\quad \rho_m = \arg\min_{\rho \in \mathbb{R}} \sum_{i=1}^{n} L(y_i, F_{m-1}(\mathbf{x}_i) + \rho h(\mathbf{x}_i; \mathbf{a}_m))$
6: $\quad$ Update $F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h(\mathbf{x}; \mathbf{a}_m)$
7: **end for**

Remarks:

- Using the same strategy as modified steepest descent, it is possible to find a local minimum along the basis function vector by setting directional derivative to zero

$$\frac{\partial}{\partial \rho} \sum_{i=1}^{n} L\Big(y_i, F_{m-1}(\boldsymbol{x}) + \rho h(\boldsymbol{x}_i; \boldsymbol{a}_m)\Big)$$

- regularized strategy is of course possible

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \nu \rho_m h(\boldsymbol{x}; \boldsymbol{a}_m) \quad , 0 < \nu \leq 1$$

- $\hat{F}_m(\boldsymbol{x})$ is a direct estimate of $E(y|\boldsymbol{x})$

Algorithm 3.1 *LS_Boost*

1: $F_0(\mathbf{x}) = \sum_i y_i/n$
2: **for** $m = 1$ to M **do**
3: $\quad g_m(\mathbf{x}_i) = y_i - F_{m-1}(\mathbf{x}_i)$
4: $\quad \{\rho_m, \mathbf{a}_m\} = \arg\min_{\{\rho,\mathbf{a}\}} \sum_{i=1}^{n} [g_m(\mathbf{x}_i) - \rho h(\mathbf{x}_i; \mathbf{a})]^2$
5: $\quad F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \nu\rho_m h(\mathbf{x}; \mathbf{a}_m)$
6: **end for**

L_2 Gradient Boost

$$y_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \epsilon_i \quad , i = 1, \ldots, n$$

- $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$, $F(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{\beta}$, $h(\boldsymbol{x}; \boldsymbol{a}) = \boldsymbol{x}$
- assume squared loss, $L(y, z) = (y - z)^2/2$

$$\begin{aligned} g_m(\boldsymbol{x}_i) =& - L'(y_i, F_{m-1}(\boldsymbol{x}_i)) \\ =& y_i - F_{m-1}(\boldsymbol{x}_i) \end{aligned}$$

- note that $g_m(\boldsymbol{x})$ is the partial residual in $m-1$ step and line 4 is an least square optimization of $g_m(\boldsymbol{x})$ on base learner $h(\boldsymbol{x}; \boldsymbol{a})$.

It is also possible to use **coordinate directions** as the base-learner.

Suppose $\sum_{i=1}^{n} y_i = 0, \sum_{i=1}^{n} x_{ik} = 0, k = 1, \ldots, p$, we can modify line 4 in Algm 3.1:

$$\{\rho_m, k_m\} = \underset{(\rho,k)\in\mathbb{R}\times\{1,\ldots,p\}}{\arg\min} \|\boldsymbol{g}_m - \rho \boldsymbol{X}_k\|^2$$

where

$$k_m = \underset{k\in\{1,\ldots,p\}}{\arg\min} \|\boldsymbol{g}_m - P_k \boldsymbol{g}_m\|^2$$
$$\rho_m = (\boldsymbol{X}_{k_m}^T \boldsymbol{X}_{k_m})^{-1} \boldsymbol{X}_{k_m}^T \boldsymbol{g}_m$$

Least Squares Gradient_Boost *LS_Boost_Linear and LARS*

Algorithm 3.2 *LS_Boost_Linear*

1: $\mathbf{F}_0(\mathbf{x}) = 0$
2: **for** $m = 1$ to M **do**
3: $\quad \mathbf{g}_m = \mathbf{y} - \mathbf{F}_{m-1}$
4: $\quad k_m = \arg\min_{k\in\{1,\ldots,p\}} \|\mathbf{g}_m - P_k \mathbf{g}_m\|^2$
5: $\quad \mathbf{F}_m = \mathbf{F}_{m-1} + \nu\rho_m \mathbf{X}_{k_m}$
6: **end for**

L_2Boost in high dimension

for $n > p$ (Peter Buhlmann, 2003)

$$y_i = f(x_i) + \epsilon_i, \quad i = 1, \ldots, n$$

- $\epsilon_1, \ldots, \epsilon_n$ iid with $E[\epsilon_i] = 0$, $\text{var}(\epsilon_i) = \sigma^2$
- $f(\cdot)$ is a real-valued, typical nonlinear function.
- Define

$$\begin{aligned}
\text{bias}^2(m) =& n^{-1} \sum_{i=1}^{n} \Big(E[\hat{F}_m(x_i)] - f(x_i) \Big)^2 \\
\text{var}(m) =& n^{-1} \sum_{i=1}^{n} \text{var}\Big(\hat{F}_m(x_i) \Big) \\
\text{MSE}(m) =& n^{-1} \sum_{i=1}^{n} E[(\hat{F}_m(x_i) - f(x_i))^2] \\
=& \text{bias}^2(m) + \text{var}(m)
\end{aligned}$$

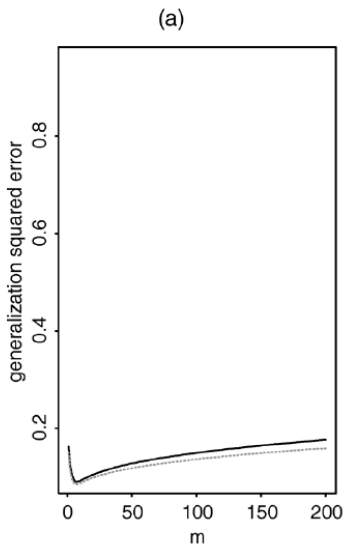
Buhlmann, P. & Yu, B., 2003. Boosting With the L2 Loss. Journal of the American Statistical Association, 98(462), pp.324-339.

then, we have the following properties

- if X is a new test observation from the design-generating distribution but independent from the training set, then

$$\text{MSE} \xrightarrow{p} E\left[(\hat{F}_m(X) - f(X))^2\right]$$

- By some regular conditions on eigenvalue of boosting operator,
  1. $\text{bias}^2(m)$ decays exponentially fast with increasing m
  2. $\text{var}(m)$ exhibits exponentially small increase with increasing m



that is, we meet samllest MSE with few steps.

for $p > n$ (Peter Buhlmann, 2006)

$$y_i = f_n(X_i) + \epsilon_i, i = 1, \ldots, n$$

$$f_n(x_i) = \sum_{j=1}^{p_n} \beta_{jn} x_j$$

regular conditions

- A1 $p_n = O(\exp(Cn^{1-\xi})), n \to \infty$, for some $0 < \xi < 1, 0 < C < \infty$
- A2 $\sum_{n \in \mathbb{N}} \sum_{j=1}^{p_n} |\beta_{jn}| < \infty$
- A3 $\sup_{1 \le j \le p_n} \|X_j\|_\infty < \infty$
- A4 $E|\epsilon|^s < \infty$ for some $s > 4/\xi$ with ξ from (A1)

Note: no assumptions are needed on the correlation structure of the predictor variables.

Buhlmann, P., 2006. Boosting for high-dimensional linear models. Annals of Statistics, 34(2), pp.559-583.

Theorem consider the linear model with condition (A1)-(A4). the boosting estimate $\hat{F}^{(m)}(\cdot)$ with the componentwise linear base procedure satisfies: for some sequence $(m_n)_{n\in\mathbb{N}}$ with $m_n \to \infty (n \to \infty)$ sufficient slowly,

$$E_X|\hat{F}_n^{(m_n)}(X) - f_n(X)|^2 = o_p(1), n \to \infty$$

where X denotes a new predictor variable, independent of and with the same distribution as the $X-$component of the data $(X_i, Y_i), i = 1, \ldots, n$

Incremental Foreward Stagewise

In Algm 3.2 (line 5), we can do some modification to obtain $FS(\nu)$

- consider a limiting case for $\rho_m \to 0$
- assume $\sum_{i=1}^{n} x_{ik}^2 = 1, \quad k = 1, \ldots, p$, it can be shown that

$$\arg\min_{k \in \{1,\ldots,p\}} \|\boldsymbol{g}_m - P_k \boldsymbol{g}_m\|^2 = \arg\min_{k \in \{1,\ldots,p\}} |\boldsymbol{X}_k^T \boldsymbol{g}_m|$$

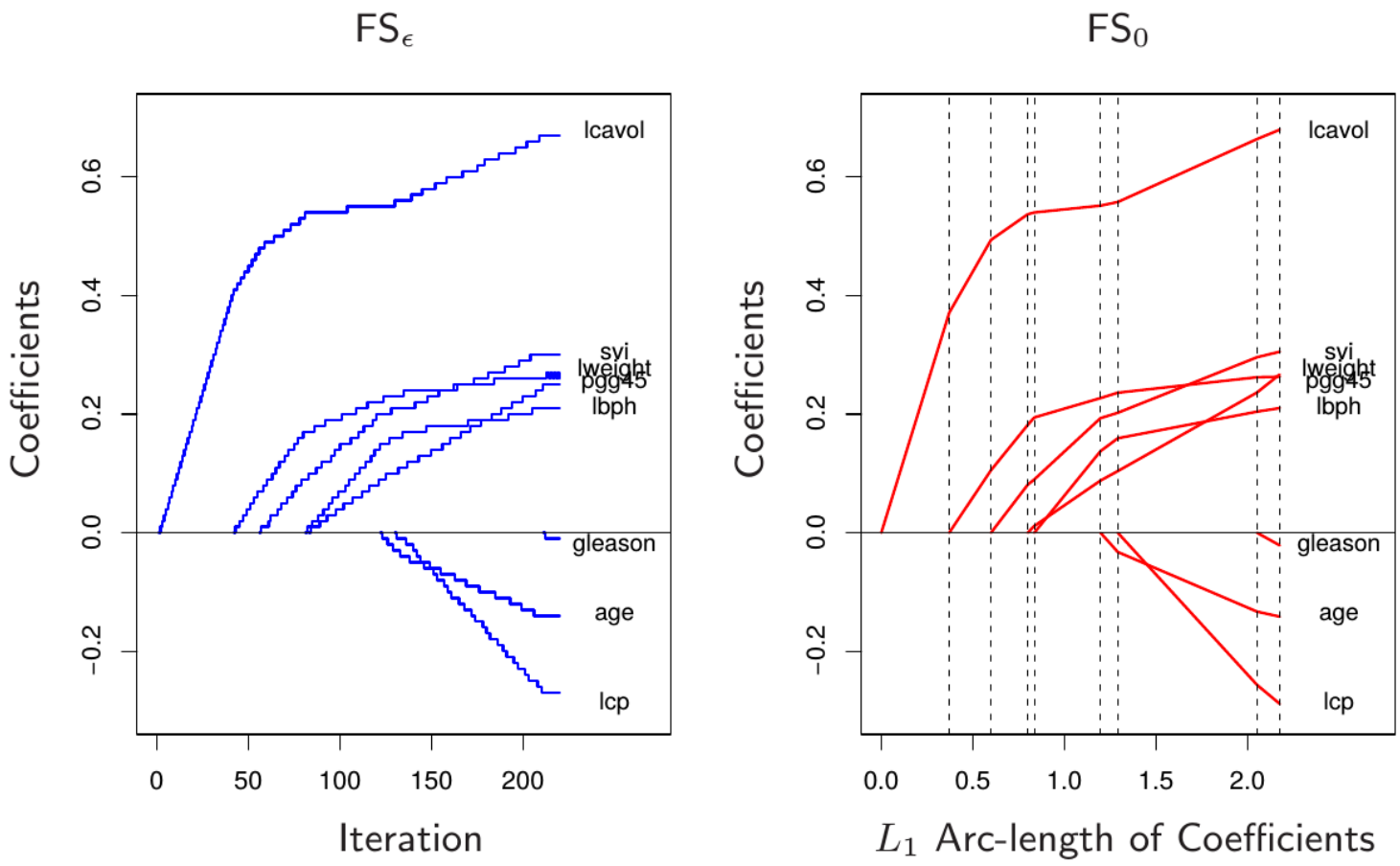
Least Squares Gradient_Boost *LS_Boost_Linear and LARS*

Algorithm 3.2 *LS_Boost_Linear*

1: $\mathbf{F}_0(\mathbf{x}) = 0$
2: **for** $m = 1$ to M **do**
3: $\quad \mathbf{g}_m = \mathbf{y} - \mathbf{F}_{m-1}$
4: $\quad k_m = \arg\min_{k \in \{1,\ldots,p\}} \|\mathbf{g}_m - P_k \mathbf{g}_m\|^2$
5: $\quad \mathbf{F}_m = \mathbf{F}_{m-1} + \nu \rho_m \mathbf{X}_{k_m}$
6: **end for**

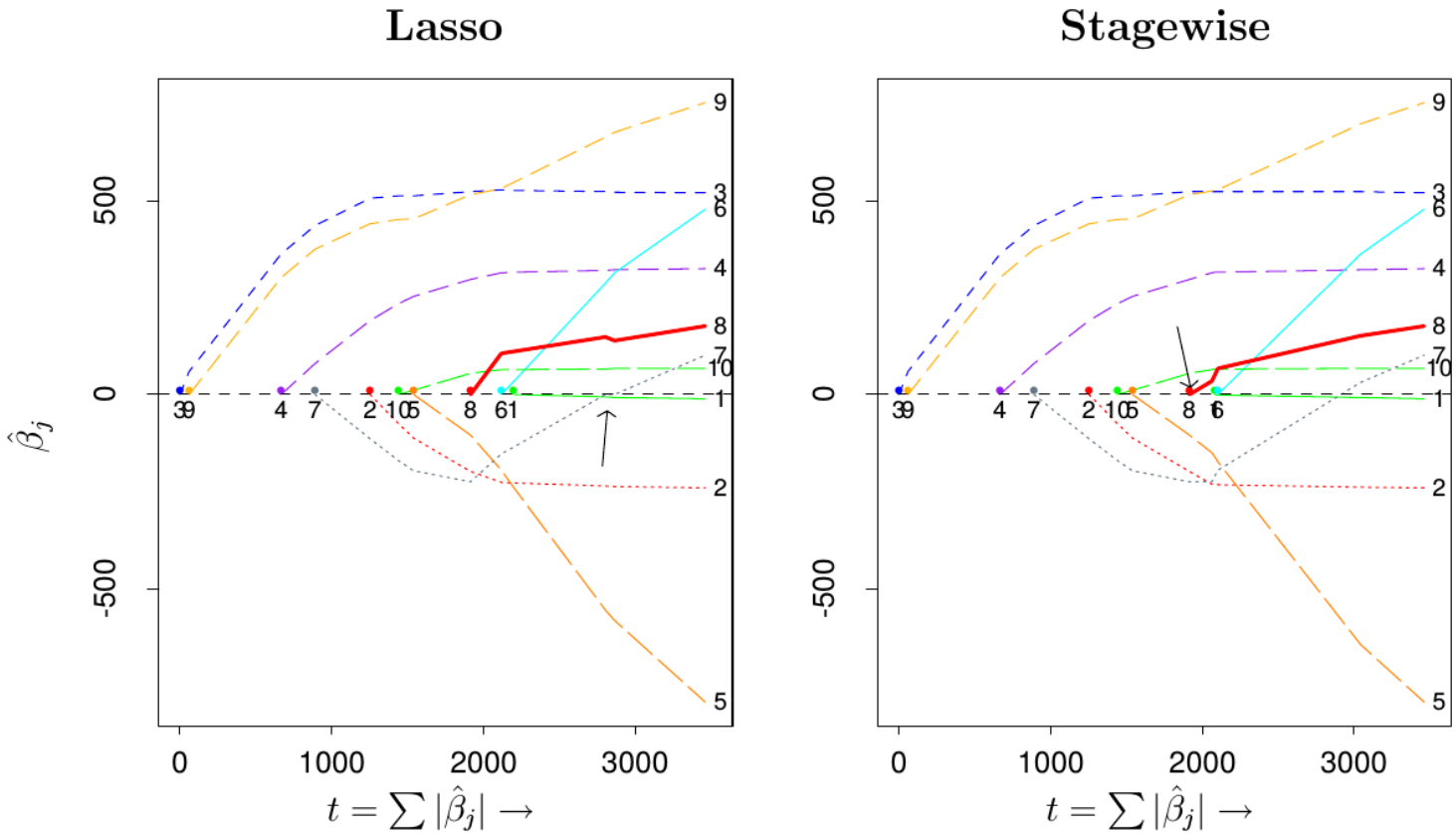
Algorithm 3.3 *FS(ν) (Incremental Forward Stagewise)*

1: $\mathbf{F}_0(\mathbf{x}) = 0$
2: **for** $m = 1$ to M **do**
3: $\quad \mathbf{g}_m = \mathbf{y} - \mathbf{F}_{m-1}$
4: $\quad k_m = \arg\max_{k \in \{1,\ldots,p\}} |\operatorname{corr}(\mathbf{g}_m, \mathbf{X}_k)|$
5: $\quad \mathbf{F}_m = \mathbf{F}_{m-1} + \nu \delta_m \mathbf{X}_{k_m}$, where $\delta_m = \operatorname{sgn}[\operatorname{corr}(\mathbf{g}_m, \mathbf{X}_{k_m})]$
6: **end for**



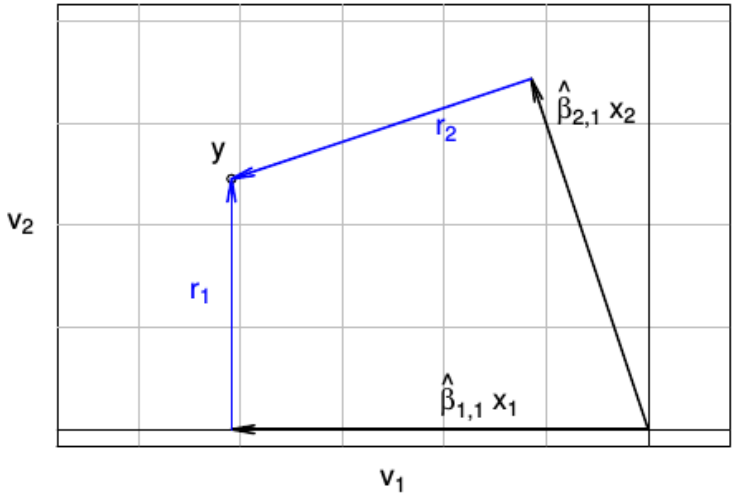
Efron realized that the path of β is **piecewise constant** when taking $\nu \to 0$

Hastie, T., Tibshirani, R. & Friedman, J., 2009. The Elements of Statistical Learning.

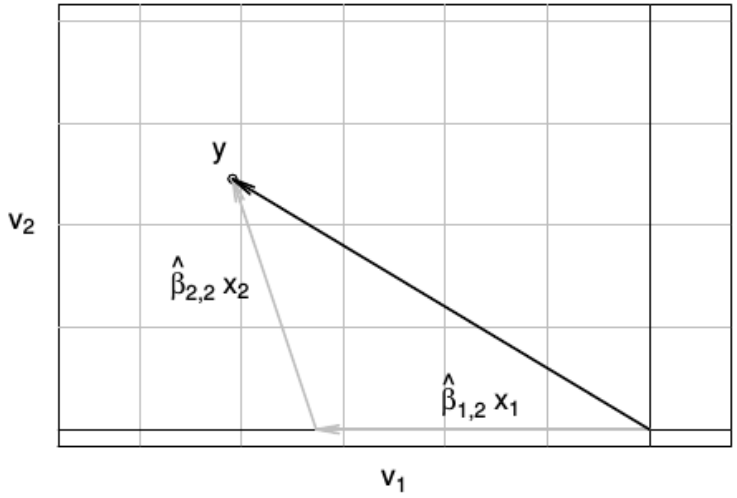


Moreover, Efron also realized that there're some connections between FS_0 and Lasso.

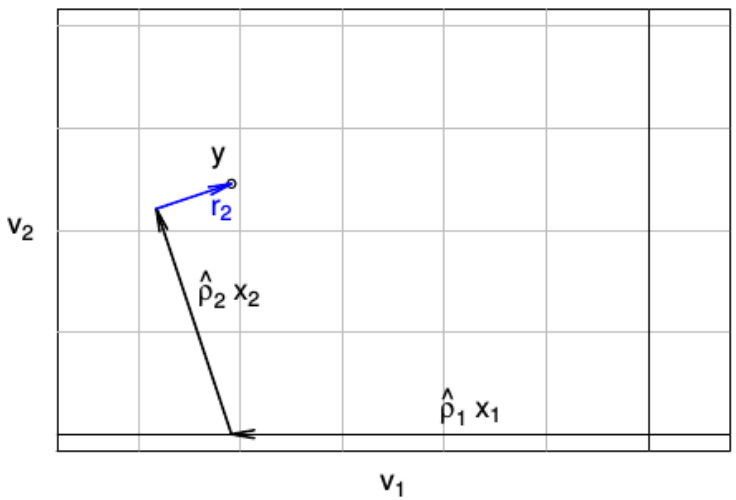
Efron, B. et al., 2004. Least angle regression. Annals of Statistics, 32, pp.407-499.



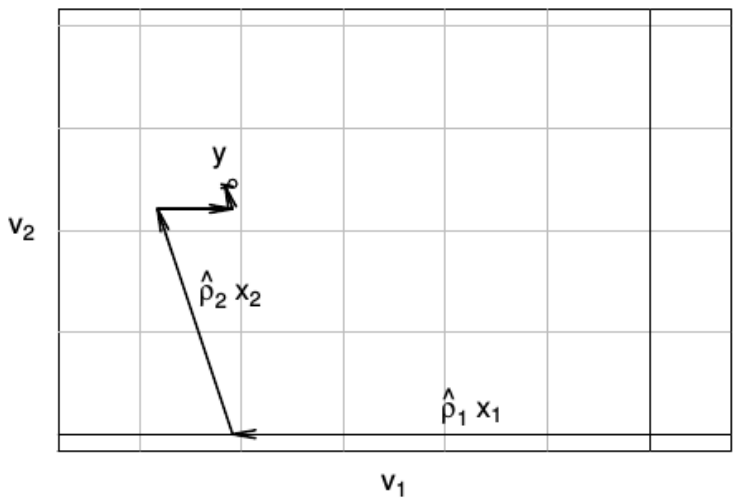
(a) Step $m = 1$: forward stepwise.



(b) Step $m = 2$: forward stepwise.

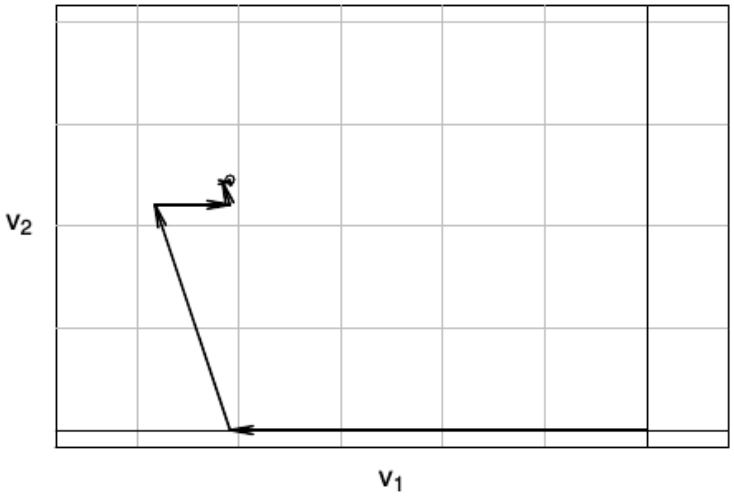


(c) Step $m = 2$: forward stagewise.

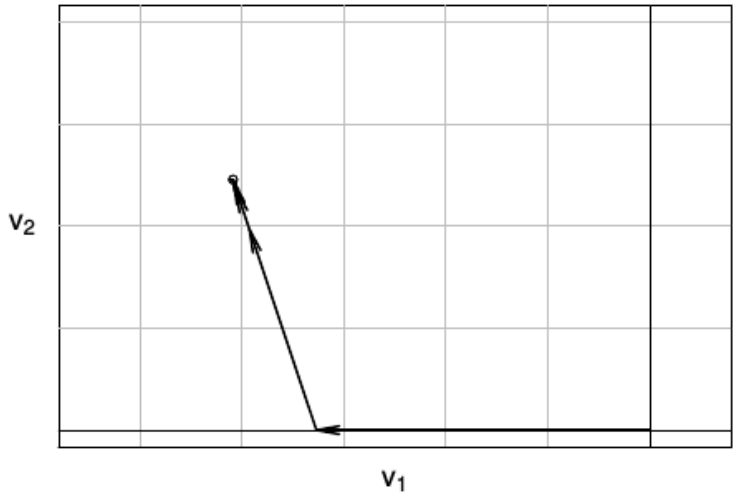


(d) Iterative stagewise.

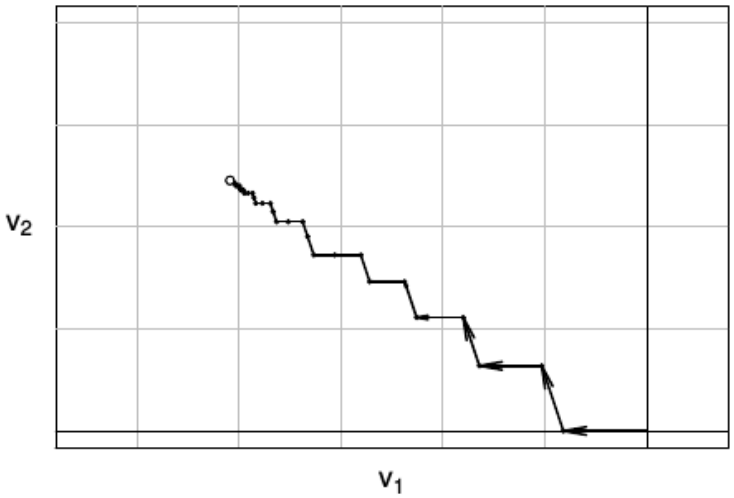
Regularized forward stagewise



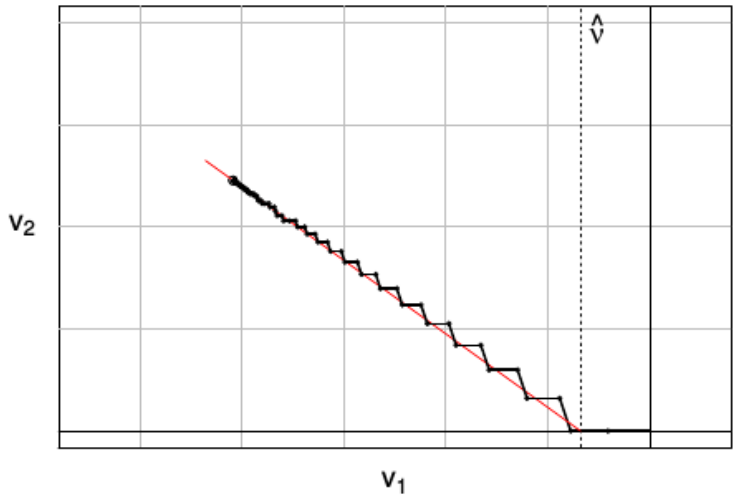
(a) Iterative stagewise ($\nu = 1.0$).



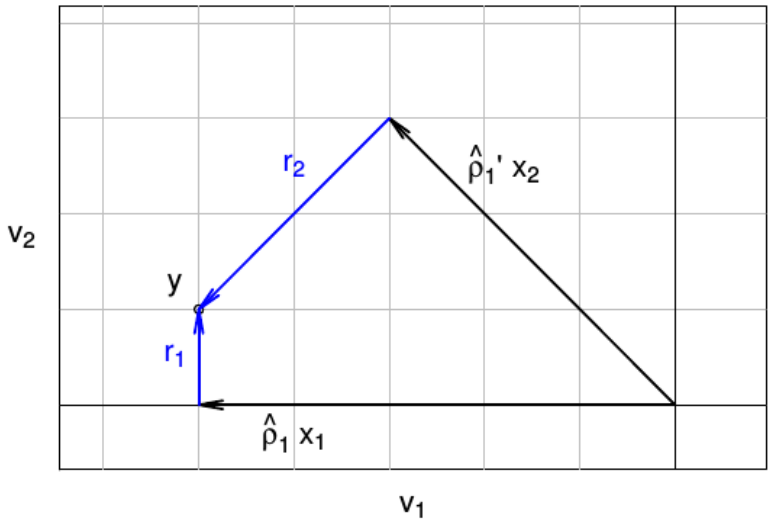
(b) Regularized stagewise ($\nu = 0.8$).



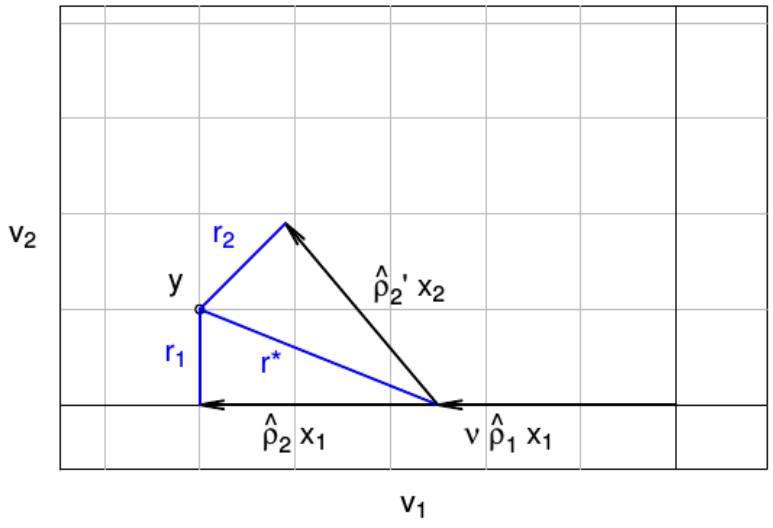
(c) Regularized stagewise ($\nu = 0.2$).



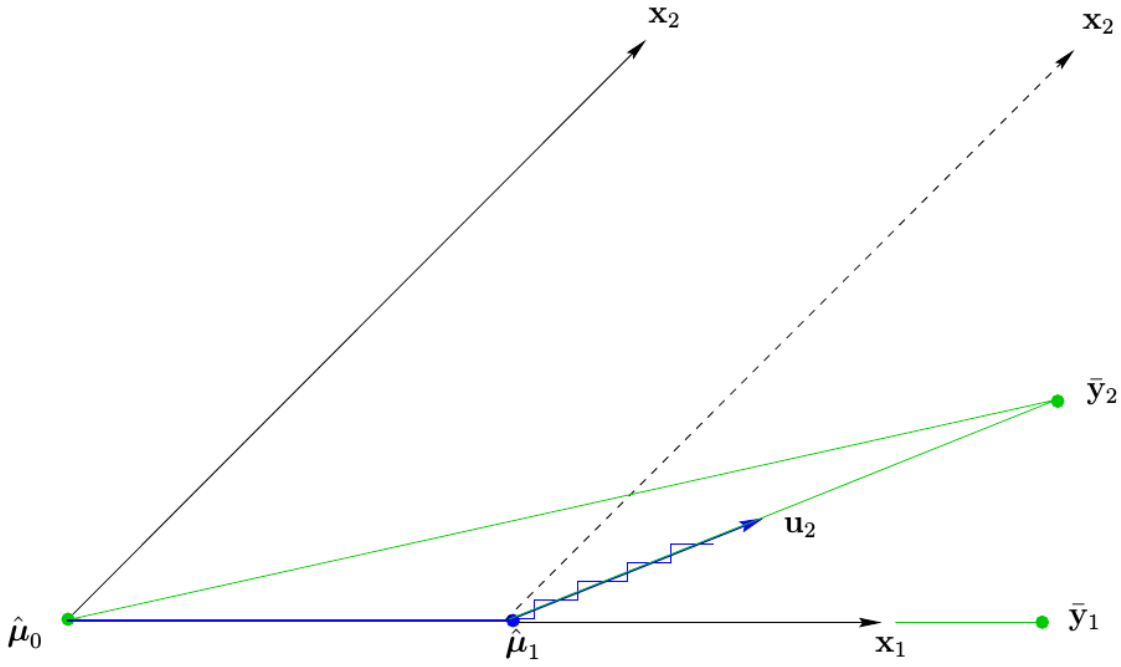
(d) Regularized stagewise ($\nu = 0.1$) with LARS equiangular direction in red



(a) $m = 1$.



(b) $m = 2$.



- By observing FS_0, Efron found that the path move along the **equiangular direction** among the predictos in the active set.

Efron, B. et al., 2004. Least angle regression. Annals of Statistics, 32, pp.407-499.

Algorithm 3.2 *Least Angle Regression.*

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.

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.

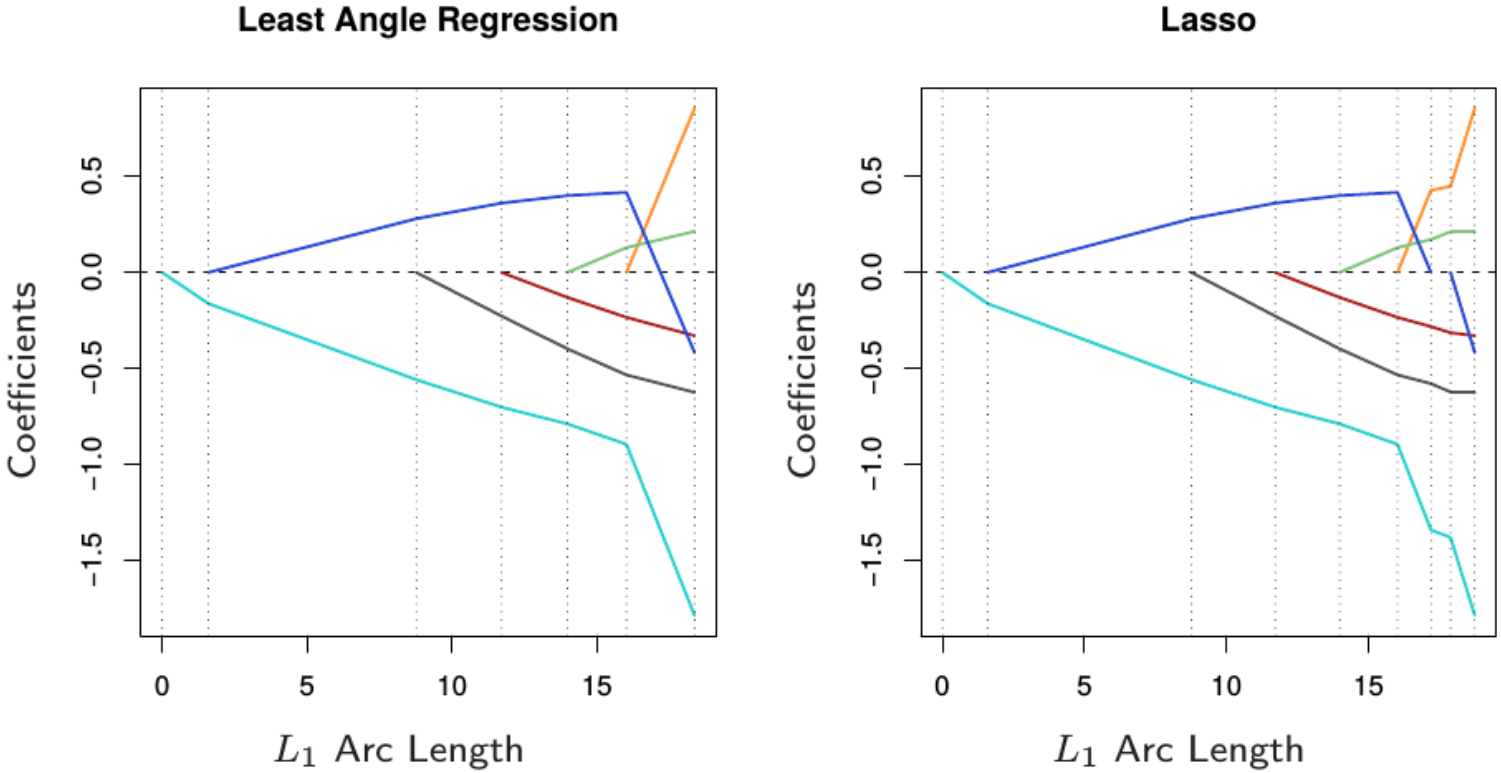
5. Continue in this way until all p predictors have been entered. After $\min(N - 1, p)$ steps, we arrive at the full least-squares solution.

Algorithm 3.2a *Least Angle Regression: 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.

Algorithm 3.4 *Incremental Forward Stagewise Regression—FS_ϵ.*

1. Start with the residual $\mathbf{r}$ equal to $\mathbf{y}$ and $\beta_1, \beta_2, \ldots, \beta_p = 0$. All the predictors are standardized to have mean zero and unit norm.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$
3. Update $\beta_j \leftarrow \beta_j + \delta_j$, where $\delta_j = \epsilon \cdot \text{sign}[\langle \mathbf{x}_j, \mathbf{r} \rangle]$ and $\epsilon > 0$ is a small step size, and set $\mathbf{r} \leftarrow \mathbf{r} - \delta_j \mathbf{x}_j$.
4. Repeat steps 2 and 3 many times, until the residuals are uncorrelated with all the predictors.



- note that in LARS, predictors cannot be removed out from active set once it is included, but Lasso can. (see blue line)
- In the path-based algorithm (LAR-Lasso modification), only p (or a bit more) step is required. However, based on coordinate descent, we have to compute in grids of a wide range of λ. (same benifit in LAR-FS_0 modification.)
- "$lars$" Package in R

Hastie, T., Tibshirani, R. & Friedman, J., 2009. The Elements of Statistical Learning.

Lasso v.s. FS_0

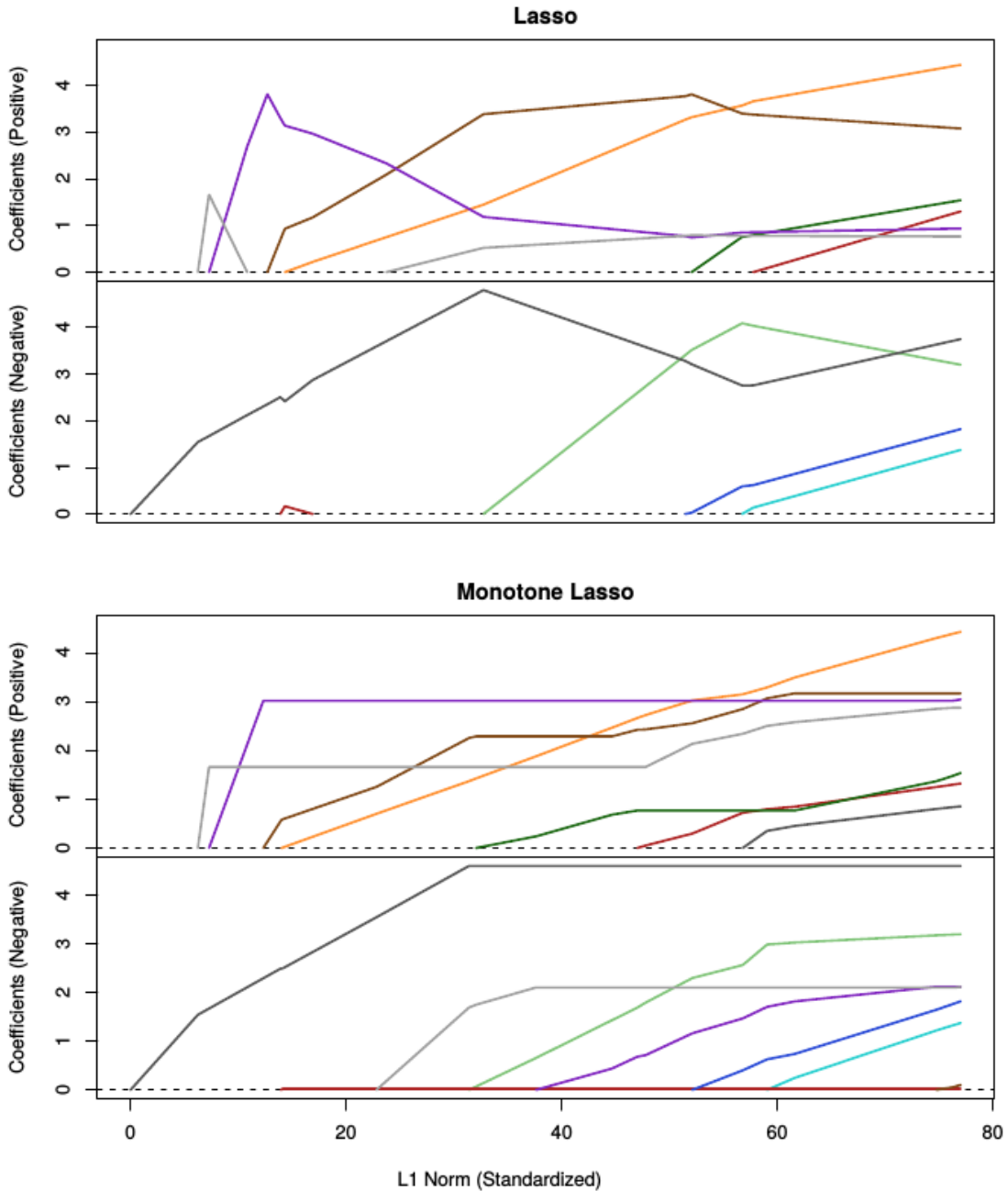
If we create an expanded data $\tilde{\boldsymbol{X}} = [\boldsymbol{X} : -\boldsymbol{X}]$, standard Lasso problem can be represented as

$$\min_{\beta_0, \beta_j^+, \beta_j^-} \sum_{i=1}^{n} \left(y_i - \beta_0 - \left[\sum_{j=1}^{p} x_{ij} \beta_j^+ - \sum_{j=1}^{p} x_{ij} \beta_j^- \right] \right)^2$$

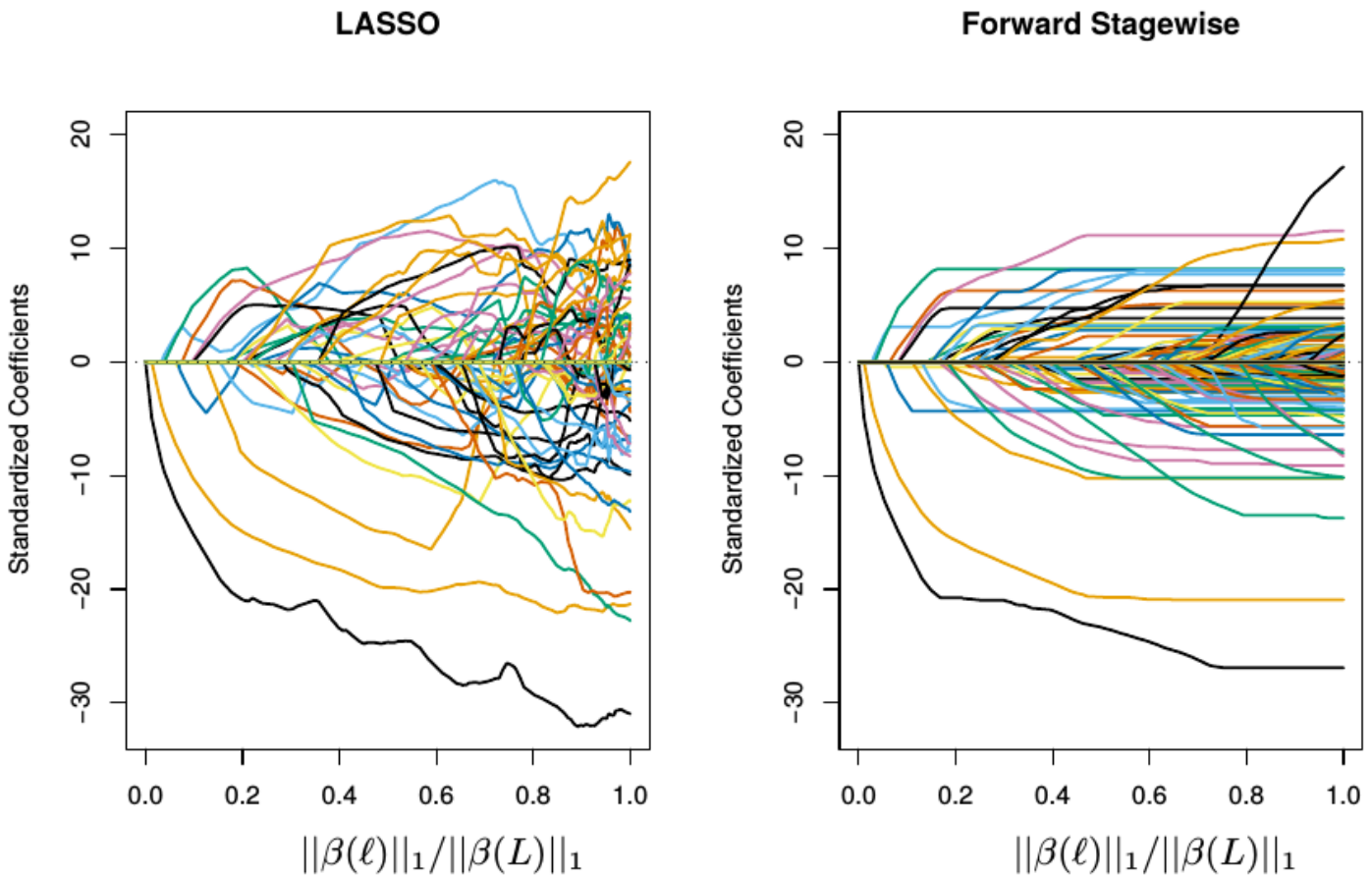
$$\text{subject to } \beta_j^+, \beta_j^- \geq 0 \ \forall j \quad \text{and} \quad \sum_{k=1}^{p} (\beta_j^+ + \beta_j^-) \leq s$$

- Hastie (2007) showed that if we seperate the Lasso path on positive side and negative side, the stagewise forward path can be obtained by constraining them to be **monotone nondecreasing**.

Hastie, T. et al., 2007. Forward stagewise regression and the monotone lasso. Electronic Journal of Statistics, 1, pp.1-29.

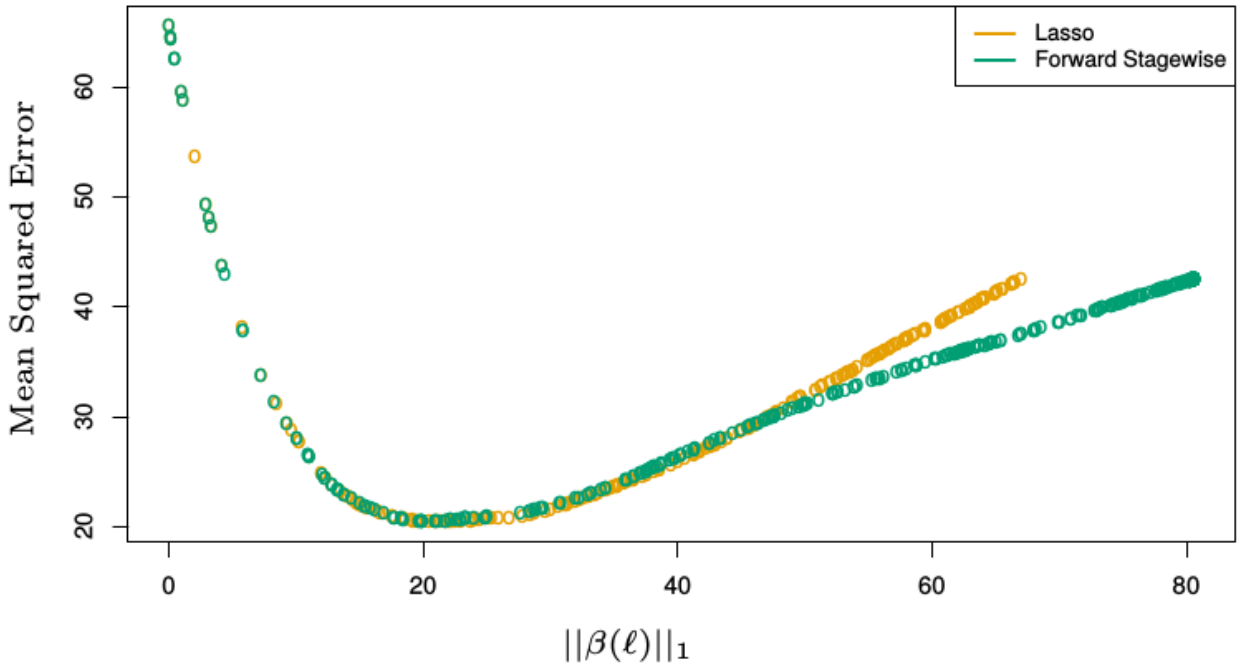

Lasso
Coefficients (Positive)
Coefficients (Negative)
Monotone Lasso
Coefficients (Positive)
Coefficients (Negative)
0
20
40
60
80
L1 Norm (Standardized)

A simulation with very high correlation in the structure of predictors
($p = 1000, N = 60, \rho = 0.95$)



Hastie, T. et al., 2007. Forward stagewise regression and the monotone lasso. Electronic Journal of Statistics, 1, pp.1-29.

the **testing error** performance are very similar



Hastie, T. et al., 2007. Forward stagewise regression and the monotone lasso. Electronic Journal of Statistics, 1, pp.1-29.

The End