Statistics 312/612, fall 2010
Homework # 9
Due: Monday 29 November

The questions on this sheet all refer to (a slight modification of) the lasso modification of the LARS algorithm, based on the papers by Efron, Hastie, Johnstone, and Tibshirani (2004) and by Rosset and Zhu (2007).

I have made a few small changes to the algorithm described in the first paper. I hope I haven't broken anything.

The homework problems are embedded into an explanation of how and why the algorithm works. Look for the numbers [1], [2], ... in the left margin and the text enclosed in a $\boxed{\textit{box}}$.

1 The Lasso problem

The problem is: given an $n \times 1$ vector y and an $n \times p$ matrix X, find the $\widehat{b}(\lambda)$ that minimizes

$$L_\lambda(b) = \|y - Xb\|^2 + 2\lambda \sum |b_j|$$

for each $\lambda \geq 0$. [The extra factor of 2 eliminates many factors of 2 in what follows.] The columns of X will be assumed to be standardized to have zero means and $\|X_j\| = 1$.

> **Remark.** I thought the vector y was also supposed to have a zero mean. That is not the case for the *diabetes* data set in R, the data set used as an illustration in the LARS paper (Efron, Hastie, Johnstone, and Tibshirani 2004) where the LARS procedure was first analyzed.

The solution will turn out to be piecewise linear and continuous in λ.

To test my modification I also used the diabetes data set:

```
library(lars)
data(diabetes) # load the data set
# ?diabetes for description
dd = diabetes
LL = lars(diabetes$x,diabetes$y,type="lasso")
plot(LL,xvar="step") # the usual plot
```

My modified algorithm gets the same coefficients as lars() for this data set.

Remark. For the sake of exposition, the discussion below assumes that the predictors enter the active set in a way slightly different from what happens for *diabetes*.

I will consider only the "one at a time"case (page 417 of the LARS paper), for which the "active set" of predictors X_j changes only by either addition or deletion of a single predictor.

2 Directional derivative

The function L_λ has derivative at b in the direction u defined by

$$\begin{aligned} L_\lambda^\bullet(b,u) &= \lim_{t\downarrow 0} \frac{L_\lambda(b+tu) - L_\lambda(b)}{t} \\ &= 2\sum\nolimits_j \lambda f(b_j,u_j) - (y-Xb)'X_j u_j \qquad <1> \\ &\qquad \text{where } f(b_j,u_j) = u_j\{b_j>0\} - u_j\{b_j<0\} + |u_j|\{b_j=0\}. \end{aligned}$$

By convexity, a vector b minimizes L_λ if and only if $L_\lambda^\bullet(b,u) \geq 0$ for every u. Equivalently, for every j and every u_j the jth summand in <1> must be non-negative. [Consider u vectors with only one nonzero component to establish this equivalence.] That is, b minimizes L_λ if and only if

$$\lambda f(b_j,u_j) \geq (y-Xb)'X_j u_j \qquad \text{for every } j \text{ and every } u_j$$

When $b_j \neq 0$ the inequalities for $u_j = \pm 1$ imply an equality; for $b_j = 0$ we get only the inequality. Thus b minimizes L_λ if and only if

$$\begin{cases} \lambda = X_j'R & \text{if } b_j > 0 \\ \lambda = -X_j'R & \text{if } b_j < 0 \\ \lambda \geq |X_j'R| & \text{if } b_j = 0 \end{cases} \qquad \text{where } R := y - Xb$$

The LARS/lasso algorithm recursively calculates a sequence of breakpoints $\infty > \lambda_1 > \lambda_2 > \cdots \geq 0$ with $\widehat{b}(\lambda)$ linear for each interval $\lambda_{k+1} \leq \lambda \leq \lambda_k$. Define "residual" vector and "correlations"

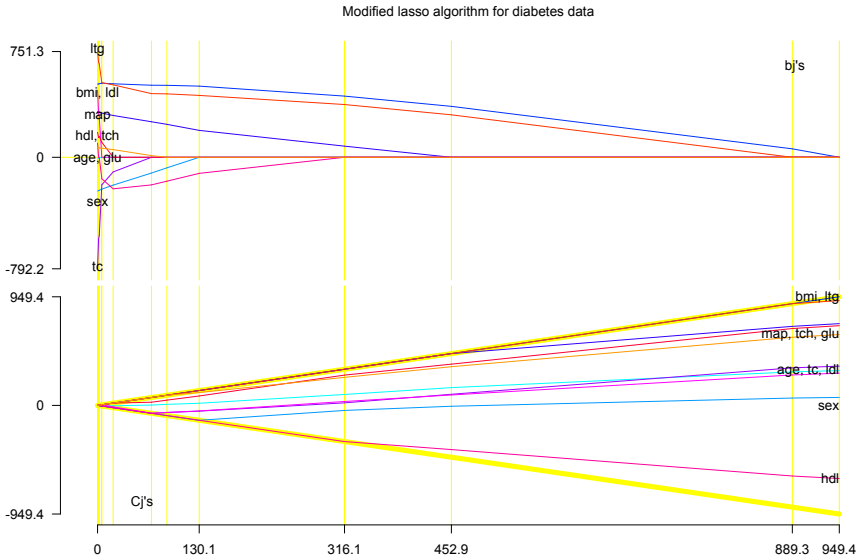
$$R(\lambda) := y - X\widehat{b}(\lambda) \qquad \text{and} \qquad C_j(\lambda) := X_j'R(\lambda).$$

Remark. To get a true correlation we would have to divide by $\|R(\lambda)\|$, which would complicate the constraints.

The algorithm will ensure that

<2>
$$\begin{cases} \lambda = C_j(\lambda) & \text{if } \widehat{b}_j(\lambda) > 0 \qquad \text{(constraint } \oplus) \\ \lambda = -C_j(\lambda) & \text{if } \widehat{b}_j(\lambda) < 0 \qquad \text{(constraint } \ominus) \\ \lambda \geq |C_j(\lambda)| & \text{if } \widehat{b}_j(\lambda) = 0 \qquad \text{(constraint } \odot) \end{cases}$$

That is, for the minimizing $\widehat{b}(\lambda)$ each $(\lambda, C_j(\lambda))$ needs to stay inside the region $\mathcal{R} = \{(\lambda, c) \in \mathbb{R}_+ \times \mathbb{R} : |c| \leq \lambda\}$, moving along the top bounday ($c = \lambda$) when $b_j(\lambda) > 0$ (constraint $\oplus$), along the lower boundary ($c = -\lambda$) when $\widehat{b}_j(\lambda) < 0$ (constraint $\ominus$), and being anywhere in $\mathcal{R}$ when $\widehat{b}_j(\lambda) = 0$ (constraint $\odot$).



[See the file lassoFull.pdf in the Handouts directory for higher resolution.]

3 The algorithm

The solution $\widehat{b}(\lambda)$ is to be constructed in a sequence of steps, starting with large λ and working towards $\lambda = 0$.

First step.

Define $\lambda_1 = \max |X_j' y|$. For $\lambda \geq \lambda_1$ take $\widehat{b}(\lambda) = 0$, so that $|C_j(\lambda)| \leq \lambda_1$. Constraint $\odot$ would be violated if we kept $\widehat{b}(\lambda)$ equal to zero for $\lambda < \lambda_1$; the $\widehat{b}(\lambda)$ must move away from zero as λ decreases below λ_1.

We must have $|C_j(\lambda_1)| = \lambda_1$ for at least one j. For convenience of exposition, suppose $C_1(\lambda_1) = \lambda_1 > |C_j(\lambda_1)|$ for all $j \geq 2$. The active set is now $A = \{1\}$.

For $\lambda_2 \le \lambda < \lambda_1$, with λ_2 to be specified soon, keep $b_j = 0$ for $j \ge 2$ but let

$$b_1(\lambda) = 0 + v_1(\lambda_1 - \lambda)$$

for some constant v_1.

We need $b_1(\lambda) = \lambda$ for a while, which forces $v_1 = 1$. That choice gives $R(\lambda) = y - X_1(\lambda_1 - \lambda)$ and

$$C_j(\lambda) = C_j(\lambda_1) - X_j' X_1(\lambda_1 - \lambda).$$

Notice that $b_1(\lambda) > 0$ so that $\oplus$ is the relevant constraint for b_1. Let λ_2 be the largest λ less than λ_1 for which $\max_{j\ge 2} |C_j(\lambda)| = \lambda$.

*[1] Find the value λ_2. That is, express λ_2 as a function of $C_j's$. Hint: Consider Efron, Hastie, Johnstone, and Tibshirani (2004, equation 2.13), with appropriate changes of notation.

Second step.

We have $C_1(\lambda_2) = \lambda_2$, by construction. For convenience of exposition, suppose $C_2(\lambda_2) = \lambda_2 > |C_j(\lambda_2)|$ for all $j \ge 3$. The active set is now $A = \{1, 2\}$. For $\lambda_3 \le \lambda < \lambda_2$ choose a new v_1 and a v_2 then define

$$\begin{aligned} b_1(\lambda) &= b_1(\lambda_2) + (\lambda_2 - \lambda)v_1 \\ b_2(\lambda) &= 0 + (\lambda_2 - \lambda)v_2 \end{aligned}$$

with all other b_j's still zero. Write Z for $[X_1, X_2]$. The new C_j's become

$$\begin{aligned} C_j(\lambda) &= X_j' \left(y - X_1 b_1(\lambda) - X_2 b_2(\lambda)\right) \\ &= C_j(\lambda_2) - (\lambda_2 - \lambda) X_j' Z v \qquad \text{where } v' = (v_1, v_2). \end{aligned}$$

*[2] Show that $C_1(\lambda) = C_2(\lambda) = \lambda$ if we choose v to make $Z'Zv = (1, 1)'$. That is, $v = (Z'Z)^{-1}\mathbf{1}$ with $\mathbf{1} = (1, 1)'$. Of course we must assume that X_1 and X_2 are linearly independent for $Z'Z$ to have an inverse.

*[3] Show that v_1 and v_2 are both positive, so that $\oplus$ is the relevant constraint for both b_1 and b_2.

Let λ_3 be the largest λ less than λ_2 for which $\max_{j\ge 3} |C_j(\lambda)| = \lambda$.

Third step.

We have $C_1(\lambda_3) = C_2(\lambda_3) = \lambda_3$ and $\max_{j\ge 3} |C_j(\lambda_3)| = \lambda_3$. Consider now what happens if the last equality holds because a C_j has hit the lower boundary of the

region $\mathcal{R}$: suppose $-C_3(\lambda_3) = \lambda_3 > \max_{j\ge 4} |C_j(\lambda_3)|$. The active set then becomes $A = \{1, 2, 3\}$. For $\lambda_4 \le \lambda < \lambda_3$ choose a new v_1, v_2 and a v_3 to make

$$b_1(\lambda) = b_1(\lambda_3) + (\lambda_3 - \lambda)v_1$$
$$b_2(\lambda) = b_2(\lambda_3) + (\lambda_3 - \lambda)v_2$$
$$b_3(\lambda) = 0 - (\lambda_3 - \lambda)v_3$$

with all other b_j's still zero. More concisely, with $s_j = \text{sign}(C_j(\lambda_3))$,

$$b_j(\lambda) = b_j(\lambda_3) + (\lambda_3 - \lambda)v_j s_j \qquad \text{for } j \in A.$$

For $\lambda_3 - \lambda$ small enough, we still have $b_1(\lambda) > 0$ and $b_2(\lambda) > 0$, keeping $\oplus$ as the relevant constraint for b_1 and b_2. To ensure that $b_3(\lambda) < 0$ we need $v_3 > 0$.

*[4] Define $Z = [X_1, X_2, -X_3]$. For $\lambda \le \lambda_3$ show that

$$R(\lambda) - R(\lambda_k) = -\sum\nolimits_{j\in A} Z_j v_j (\lambda_k - \lambda)$$

so that

$$C_j(\lambda) = C_j(\lambda_3) - (\lambda_3 - \lambda) X_j' Z v$$

Show that $v = (Z'Z)^{-1} = \mathbf{1}$ keeps b_1, b_2, and b_3 on the correct boundaries.

How should we choose λ_4? Is it possible for some active b_j to switch sign?

[5] Prove that $v_3 > 0$. I am not yet clear on why this must be true. Efron, Hastie, Johnstone, and Tibshirani (2004, Lemma 4) seems to contain the relevant argument.

The general step.

Just before $\lambda = \lambda_{k+1}$ suppose the active set is A, so that $|C_j(\lambda)| = \lambda$ for each j in A. For each j in A let $s_j = \text{sign}(C_j(\lambda_k))$. For $\lambda_{k+1} \le \lambda \le \lambda_k$ define

$$b_j(\lambda) = b_j(\lambda_k) + (\lambda_k - \lambda)v_j s_j \qquad \text{for } j \in A$$

*[6] Define Z as the matrix with jth column equal to $s_j X_j$ for each $j \in A$. For $\lambda_{k+1} \le \lambda \le \lambda_k$ show that

$$C_j(\lambda) = C_j(\lambda_k) - (\lambda_k - \lambda) X_j' Z v$$

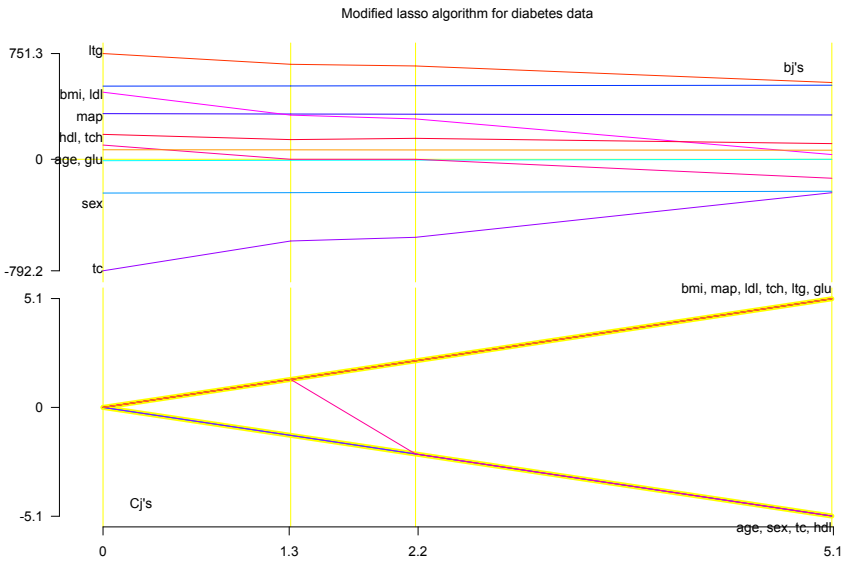
so that the choice $= (Z'Z)^{-1}\mathbf{1}$ keeps $C_j(\lambda) = s_j\lambda$ for $j \in A$.

Define λ_{k+1} to be the largest λ less than λ_k for which either of these conditions holds:

(i) $\max_{j \notin A} |C_j(\lambda)| = \lambda$. In that case add the new $j \in A^c$ for which $|C_j(\lambda_{k+1})| = \lambda_{k+1}$ to the active set, then carry out another general step.

(ii) $b_j(\lambda) = 0$ for some $j \in A$. In that case, remove j from the active set, then carry out another general step.

For *diabetes*, the second alternative caused the behavior shown below [see lassoEnding.pdf for higher resolution] for $1.3 \leq \lambda \leq 2.2$.



*[7] Find λ_{k+1}. That is, write out what an R program would have to calculate.

Note that this step is slightly tricky. If predictor j is removed from the active set because $b_j = 0$ then the corresponding C_j will lie on the boundary of $\mathcal{R}$. You will need to interpret (i) with care to avoid $\lambda_{k+1} = \lambda_k$.

[8] Show that the coefficient $b_j(\lambda)$, if j enters the active set, has the correct sign. Again Efron, Hastie, Johnstone, and Tibshirani (2004, Lemma 4) seems to contain the relevant argument.

If you think you understand the modified algorithm, as described above, try to submit your solution to the next problem to me by Friday 3 December.

[9] Write an R program (a script or a function) that implements the modified algorithm. Test it out on the *diabetes* data set.

References

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *The Annals of Statistics 32*(2), pp. 407–451.

Rosset, S. and J. Zhu (2007). Piecewise linear regularized solution paths. *Annals of Statistics 35*(3), 1012–1030.