# Chapter 7: LASSO Regression[a]

As in the previous chapter we consider observations $\{(y_i^0, x_i^0)\}_{i=1}^n$ and assume the following linear regression model

$$Y_i^0 = \alpha + \beta^\top x_i^0 + \epsilon_i, \quad i = 1, \dots, n \tag{7.1}$$

where $\beta \in \mathbb{R}^p$, $\alpha \in \mathbb{R}$ and where, for all $i, l \in \{1, \dots, n\}$, $\mathbb{E}[\epsilon_i] =$ and $\mathbb{E}[\epsilon_i \epsilon_l] = \sigma^2 \delta_{il}$ for some $\sigma^2 > 0$.

For $r \geq 0$ let

$$(\hat{\alpha}_\lambda^{(r)}, \hat{\beta}_\lambda^{(r)}) \in \underset{\alpha \in \mathbb{R},\ \beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\| y^0 - \alpha - \boldsymbol{X}^0 \beta \right\|_2^2 + 2\lambda \sum_{j=1}^p |\beta_j|^r \tag{7.2}$$

with the convention $\sum_{j=1}^p |\beta_j|^r = \sum_{j=1}^p \mathbf{1}_{\mathbb{R} \setminus \{0\}}(\beta_j)$ when $r = 0$.

The ridge estimates $(\hat{\alpha}_{2\lambda}, \hat{\beta}_{2\lambda})$ of $(\alpha, \beta)$ corresponds to the case $r = 2$ and, as we saw in the previous chapter, shrinks the regression coefficient towards zero. However, none of the components of $\hat{\beta}_{2\lambda}$ is shrink exactly to zero.

In practice, it is sometimes convenient (notably to facilitate the interpretation of the fitted model) to have some regression coefficients exactly equal to zero, so that the corresponding effect is dropped from the model. In particular, for large $p$ we often would like to have a sparse estimate $\tilde{\beta}$ of $\beta$, where $\tilde{\beta}_j = 0$ for many $j \in \{1, \dots, p\}$. As we will see in this chapter, this is exactly what LASSO regression achieves.

**Remark:** In the terminology introduced at the very beginning of Chapter 1, LASSO regression is therefore a feature selection method.

[a] The main reference for this chapter is [4, Chapter 2].

## Preliminaries

A natural way to obtain a sparse estimate $\tilde{\beta}$ of $\beta$ is to penalize for the number of non-zero coefficients, that is to let $r = 0$ in (7.2).

However, computing $\hat{\beta}_\lambda^{(r)}$ for $r = 0$ is computationally hard, one reason being that for $r = 0$ the function

$$(\alpha, \beta) \mapsto \left\|y^0 - \alpha - \boldsymbol{X}^0\beta\right\|_2^2 + 2\lambda\sum_{j=1}^{p}|\beta_j|^r \tag{7.3}$$

is non-convex and may have several local minima.

The LASSO estimator $(\tilde{\alpha}_\lambda, \tilde{\beta}_\lambda)$ of $(\alpha, \beta)$ is obtained by letting $r = 1$ in (7.2), that is $(\tilde{\alpha}_\lambda, \tilde{\beta}_\lambda) = (\hat{\alpha}_\lambda^{(1)}, \hat{\beta}_\lambda^{(1)})$.

For $r = 1$ the function defined in (7.3) is the sum of two convex functions, and is therefore convex. Consequently, all the local minima of this function are also global minima. In addition, as we will see below, the estimate $\tilde{\beta}_\lambda$ is (typically) sparse.

In this chapter, for $A \subseteq \{1, \ldots, p\}$ we let $\boldsymbol{X}_A$ be the $n \times |A|$ matrix having the vectors $\{x_{(j)}\}_{j\in A}$ as columns and, for $\beta \in \mathbb{R}^p$, we let $\beta_A$ be the $|A|$-dimensional vector having elements $\{\beta_j\}_{j\in A}$.

Finally, we recall that for $z \in \mathbb{R}$ we have $\text{sign}(z) = 1$ if $z > 0$, $\text{sign}(z) = -1$ if $z < 0$ and $\text{sign}(z) = 0$ if $z = 0$. If $z \in \mathbb{R}^k$ for some integer $k > 1$ we abuse notation in what follows by using the shorthand $\text{sign}(z) = (\text{sign}(z_1), \ldots, \text{sign}(z_k))$.

## The LASSO estimator

The following theorem characterizes the solution of the optimization problem (7.3) when $r = 1$.

**Theorem 7.1** *Let $\lambda > 0$, $\tilde{\alpha} \in \mathbb{R}$ and $\tilde{\beta} \in \mathbb{R}^p$. Then, $(\tilde{\alpha}, \tilde{\beta})$ is a solution to (7.3) for $r = 1$ if and only if $\tilde{\alpha} = \bar{y}^0 - \tilde{\beta}^\top \bar{x}^0$ and if and only if, for all $j \in \{1, \dots, p\}$,*

$$\begin{aligned} x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}) &= \lambda \operatorname{sign}(\tilde{\beta}_j) \qquad && \textit{if } \tilde{\beta}_j \neq 0 \\ \left| x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}) \right| &\leq \lambda && \textit{if } \tilde{\beta}_j = 0. \end{aligned} \tag{7.4}$$

*Moreover, letting $A = \{j \in \{1, \dots, p\} : \tilde{\beta}_j \neq 0\}$, if the matrix $(\boldsymbol{X}_A^\top \boldsymbol{X}_A)$ is invertible we have*

$$\begin{aligned} \tilde{\beta}_A &= (\boldsymbol{X}_A \boldsymbol{X}_A^\top)^{-1} \Big( \boldsymbol{X}_A^\top y - \lambda \operatorname{sign}\big(\boldsymbol{X}_A^\top (y - \boldsymbol{X}\tilde{\beta})\big) \Big) \\ &= (\boldsymbol{X}_A \boldsymbol{X}_A^\top)^{-1} \big( \boldsymbol{X}_A^\top y - \lambda \operatorname{sign}(\tilde{\beta}_A) \big). \end{aligned}$$

**Remark:** By Theorem 7.1, $\tilde{\beta}_\lambda = 0$ if $\lambda \geq \max_{j \in \{1,\dots,p\}} |x_{(j)}^\top y|$.

*Proof of Theorem 7.1*[a]*:* Following the computations done in the proof of Proposition 6.1, we have $\tilde{\alpha}_\lambda = \bar{y}^0 - \tilde{\beta}_\lambda^\top \bar{x}^0$ and

$$\tilde{\beta}_\lambda \in \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \|y - \boldsymbol{X}\beta\|_2^2 + 2\lambda \sum_{j=1}^p |\beta_j|.$$

We let

$$F_\lambda(\beta) = \frac{1}{2} \|y - \boldsymbol{X}\beta\|_2^2 + \lambda \sum_{j=1}^p |\beta_j|$$

and first show that if $\tilde{\beta} \in \operatorname{argmin}_{\beta \in \mathbb{R}^p} F_\lambda(\beta)$ then (7.4) holds.

[a] A much shorter proof of this result, based on known necessary and sufficient subgradient conditions for finding the minimum of a non-differentiable convex function, can be found in [7].

## Proof of Theorem 7.1 (end)

Remark that for all $j \in \{1, \dots, p\}$ and $\beta \in \mathbb{R}^p$ such that $\beta_j \neq 0$ we have

$$\frac{\partial}{\partial \beta_j} F_\lambda(\beta) = x_{(j)}^\top (\boldsymbol{X}\beta - y) + \lambda \operatorname{sign}(\beta_j). \tag{7.5}$$

We now let $j \in \{1, \dots, p\}$ be such that $\tilde{\beta}_j \neq 0$. Then, we must have

$$\frac{\partial}{\partial \beta_j} F_\lambda(\beta)\Big|_{\beta=\tilde{\beta}} = 0 \tag{7.6}$$

since otherwise by slightly increasing or decreasing $\tilde{\beta}_j$ we can reduce the value of $F_\lambda(\tilde{\beta})$, which would contradict the fact that $\tilde{\beta} \in \operatorname{argmin}_{\beta \in \mathbb{R}^p} F_\lambda(\beta)$. Hence, using (7.5) it follows that if $\tilde{\beta}_j \neq 0$ then

$$x_{(j)}^\top (\boldsymbol{X}\tilde{\beta} - y) + \lambda \operatorname{sign}(\tilde{\beta}_j) = 0 \Leftrightarrow x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}) = \lambda \operatorname{sign}(\tilde{\beta}_j).$$

We now let $j$ be such that $\tilde{\beta}_j = 0$ and for every $\beta_j \neq 0$ let $\tilde{\beta}^{\beta_j}$ be the vector $\tilde{\beta}$ whose $j$th component has been replaced by $\beta_j$. Then,

- for all small enough $\beta_j > 0$ we must have $\frac{\partial}{\partial \beta_j} F_\lambda(\beta)\big|_{\beta=\tilde{\beta}^{\beta_j}} > 0$. By (7.5), this is equivalent to have $x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}^{\beta_j}) < \lambda$ for all small enough $\beta_j > 0$.

- for all large enough $\beta_j < 0$ we must have $\frac{\partial}{\partial \beta_j} F_\lambda(\beta)\big|_{\beta=\tilde{\beta}^{\beta_j}} > 0$. By (7.5), this is equivalent to have $x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}^{\beta_j}) > -\lambda$ for all large enough $\beta_j < 0$.

Therefore, for all $\beta_j$ such that $|\beta_j|$ is small enough we have $|x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}^{\beta_j})| < \lambda$ and therefore

$$|x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta})| = \lim_{\beta_j \to 0} |x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta}^{\beta_j})| \leq \lambda.$$

This shows that if $\tilde{\beta}_j = 0$ then $|x_{(j)}^\top (y - \boldsymbol{X}\tilde{\beta})| \leq \lambda$, which concludes to show that (7.4) holds if $\tilde{\beta} \in \operatorname{argmin}_{\beta \in \mathbb{R}^p} F_\lambda(\beta)$.

We now let $\tilde{\beta} \in \mathbb{R}^p$ be such that (7.4) holds and show that $\tilde{\beta} \in \operatorname{argmin}_{\beta \in \mathbb{R}^p} F_\lambda(\beta)$. Let

$$F_A(\beta) = \frac{1}{2} \sum_{i=1}^n \big(y_i - \sum_{j \in A} x_{ij}\beta_j\big)^2 + \lambda \sum_{j \in A} |\beta_j|, \quad \forall \beta \in \mathbb{R}^{|A|}.$$

Then, using (7.5) and under (7.4), we have $\frac{\partial}{\partial \beta} F_A(\beta)\big|_{\beta=\tilde{\beta}_A} = 0$ and since $F_A$ is convex it follows that $\tilde{\beta}_A \in \operatorname{argmin}_{\beta \in \mathbb{R}^{|A|}} F_A(\beta)$.

## Proof of Theorem 7.1 (end)

Consequently, for all $\beta \in \mathbb{R}^p$ we have

$$\begin{aligned} F_\lambda(\beta) - F_\lambda(\tilde\beta) &= F_A(\beta_A) - F_A(\tilde\beta_A) \\ &\quad + \frac{1}{2}\sum_{i=1}^n \Big(\sum_{j\notin A} \beta_j x_{ij}\Big)^2 - \sum_{j\notin A}\Big(\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j|\Big) \\ &\geq - \sum_{j\notin A}\Big(\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j|\Big) \end{aligned} \tag{7.7}$$

and we know show that the term after the inequality sign is non-negative under (7.4).

To this aim let $j \notin A$ and assume first that $\beta_j > 0$. In this case, we have

$$\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j| = \beta_j\big(x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda\big) \tag{7.8}$$

Under (7.4) we have $|x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta)| \leq \lambda$ and thus $x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) \leq \lambda$. Together with (7.8), this shows that if $\beta_j > 0$ then $\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j| \leq 0$.

Assume now that $\beta_j < 0$ so that

$$\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j| = -|\beta_j|\big(x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) + \lambda\big). \tag{7.9}$$

Under (7.4) we have $|x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta)| \leq \lambda$ and thus $x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) \geq -\lambda$. Together with (7.9), this shows that if $\beta_j < 0$ then $\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j| \leq 0$.

Therefore, using (7.7) it follows that

$$F_\lambda(\beta) - F_\lambda(\tilde\beta) \geq - \sum_{j\notin A}\Big(\beta_j x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) - \lambda|\beta_j|\Big) \geq 0, \quad \forall \beta \in \mathbb{R}^p$$

showing that $\tilde\beta \in \text{argmin}_{\beta\in\mathbb{R}^p} F_\lambda(\beta)$. This concludes to show that (7.4) holds if and only if $\tilde\beta \in \text{argmin}_{\beta\in\mathbb{R}^p} F_\lambda(\beta)$.

To prove the last part of the theorem note that for all $j \in A$ we have $x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta) = \lambda\,\text{sign}(\tilde\beta_j)$, and thus $\text{sign}\big(x_{(j)}^\top (y - \boldsymbol{X}\tilde\beta)\big) = \text{sign}(\tilde\beta_j)$ for all $j \in A$. Hence,

$$\lambda\text{sign}(\tilde\beta_A) = \boldsymbol{X}_A^\top (y - \boldsymbol{X}\tilde\beta) = \boldsymbol{X}_A^\top (y - \boldsymbol{X}_A\tilde\beta_A) = \lambda\text{sign}\Big(\boldsymbol{X}_A^\top (y - \boldsymbol{X}\tilde\beta)\Big)$$

and the proof is complete. □

## Computation of $\tilde\beta_\lambda$: A first preliminary result

For every $\lambda > 0$ let $A_\lambda \subseteq \{1, \dots, p\}$ be such that $\tilde\beta_{\lambda,j} \neq 0$ if and only if $j \in A_\lambda$.

**Proposition 7.1** *Let* $\lambda_0 > 0$, $\lambda_1 = \sup\{\lambda \in [0, \lambda_0] : A_\lambda \neq A_{\lambda_0}\}$ *and assume that the matrix* $\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}}$ *is invertible. Then, for all* $\lambda \in (\lambda_1, \lambda_0]$ *we have*

$$\tilde\beta_{A_\lambda} = (\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1}\Big(\boldsymbol{X}_{A_{\lambda_0}} y - \lambda\,\mathrm{sign}(\tilde\beta_{A_{\lambda_0}})\Big).$$

*Proof:* Remark that $\mathrm{sign}(\tilde\beta_{A_\lambda}) = \mathrm{sign}(\tilde\beta_{A_{\lambda_0}})$ for all $\lambda \in (\lambda_1, \lambda_0]$. Then, by Theorem 7.1, for all $\lambda \in (\lambda_1, \lambda_0]$ we have

$$\begin{aligned}\tilde\beta_{A_\lambda} &= (\boldsymbol{X}_{A_\lambda}^\top \boldsymbol{X}_{A_\lambda})^{-1}\Big(\boldsymbol{X}_{A_\lambda} y - \lambda\,\mathrm{sign}(\tilde\beta_{A_\lambda})\Big)\\ &= (\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1}\Big(\boldsymbol{X}_{A_{\lambda_0}} y - \lambda\,\mathrm{sign}(\tilde\beta_{A_{\lambda_0}})\Big)\end{aligned}$$

and the proof is complete. □

**Remark:** Proposition 7.1 shows that $\tilde\beta_{A_\lambda}$ (and thus $\tilde\beta_\lambda$) is a linear function of $\lambda$ as long as the set $A_\lambda$ remains unchanged.

## Computation of $\tilde{\beta}_\lambda$: A second preliminary result

**Proposition 7.2** *Let* $\lambda_0 > 0$, $\lambda_1 = \sup\{\lambda \in [0, \lambda_0] : A_\lambda \neq A_{\lambda_0}\}$ *and assume that the matrix* $\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}}$ *is invertible. Then,*

$$\lambda_1 = \max\left\{0, \lambda_1^{(1)}, \lambda_1^{(2)}\right\}$$

*where*

$$\lambda_1^{(1)} = \max_{j \notin A_{\lambda_0}} \left\{ \max\left\{ -\frac{w_j^0}{\tilde{w}_j^0 + 1}, -\frac{w_j^0}{\tilde{w}_j^0 - 1} \right\} \right\}, \qquad \lambda_1^{(2)} = \max_{j \in \{1,\dots,|A_{\lambda_0}|\}} \left\{ \frac{\tilde{v}_j^0}{v_j^0} \right\}$$

*with*

$$\begin{aligned}
\tilde{v}^0 &= (\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1} \boldsymbol{X}_{A_{\lambda_0}} y \\
v^0 &= (\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1} \operatorname{sign}(\tilde{\beta}_{A_{\lambda_0}}) \\
w_j^0 &= x_{(j)}^\top (y - \boldsymbol{X}_{A_{\lambda_0}} \tilde{v}^0), && \forall j \notin A_{\lambda_0} \\
\tilde{w}_j^0 &= x_{(j)}^\top \boldsymbol{X}_{A_{\lambda_0}} v^0, && \forall j \notin A_{\lambda_0}.
\end{aligned}$$

*Moreover,*

$$A_{\lambda_1} = \begin{cases} 0, & \lambda_1 = 0 \\ A_{\lambda_0} \cup \left\{ \operatorname{argmax}_{j \notin A_{\lambda_0}} \left\{ \max\left\{ -\frac{\tilde{w}_j^0}{w_j^0+1}, -\frac{\tilde{w}_j^0}{w_j^0-1} \right\} \right\} \right\}, & \lambda_1 = \lambda_1^{(1)} \\ A_{\lambda_0} \setminus \{ j \in A_{\lambda_0} : \frac{\tilde{v}_j^0}{v_j^0} = \max_{k \in \{1,\dots,|A_{\lambda_0}|\}} \left\{ \frac{\tilde{v}_k^0}{v_k^0} \right\}, & \lambda_1 = \lambda_1^{(2)}. \end{cases}$$

## Proof of Proposition 7.2

We first remark that we have either $A_{\lambda_0} \subsetneq A_{\lambda_1}$ (active set addition) or $A_{\lambda_1} \subsetneq A_{\lambda_0}$ (active set deletion).

We first compute $\lambda_1$ assuming that $A_{\lambda_0} \subsetneq A_{\lambda_1}$. In this case, by Theorem 7.1 and noting that, by Proposition 7.1, $\hat{\beta}_{A_\lambda} = \tilde{v}^0 - \lambda v^0$ for all $\lambda \in (\lambda_1, \lambda_0)$, we have

$$\begin{aligned}
\lambda_1 &= \sup\Big\{\lambda \in [0, \lambda_0] : |x_{(j)}^\top (y - \boldsymbol{X}_{A_{\lambda_0}} \tilde{\beta}_{A_\lambda})| \geq \lambda \text{ for a } j \notin A_{\lambda_0}\Big\} \\
&= \sup\Big\{\lambda \in [0, \lambda_0] : x_{(j)}^\top (y - \boldsymbol{X}_{A_{\lambda_0}} \tilde{\beta}_{A_\lambda}) \pm \lambda \text{ for a } j \notin A_{\lambda_0}\Big\} \\
&= \sup\Big\{\lambda \in [0, \lambda_0] : x_{(j)}^\top (y - \boldsymbol{X}_{A_{\lambda_0}} (\tilde{v}^0 - \lambda v^0)) \pm \lambda \text{ for a } j \notin A_{\lambda_0}\Big\} \\
&= \sup\Big\{\lambda \in [0, \lambda_0) : w_j^0 + \lambda(\tilde{w}_j^0 \pm 1) = 0 \text{ for a } j \notin A_{\lambda_0}\Big\} \\
&= \max\left\{0, \max_{j \notin A_{\lambda_0}} \left\{ -\frac{w_j^0}{\tilde{w}_j^0 + 1}\right\}, \max_{j \notin A_{\lambda_0}} \left\{ -\frac{w_j^0}{\tilde{w}_j^0 - 1}\right\}\right\}.
\end{aligned}$$

We now compute $\lambda_1$ assuming that $A_{\lambda_1} \subsetneq A_{\lambda_0}$. Since $\hat{\beta}_{A_\lambda} = \tilde{v}^0 - \lambda v^0$ as long as $A_\lambda = A_{\lambda_0}$, as $\lambda$ decreases from $\lambda_0$ an element will be removed from $A_{\lambda_0}$ as soon $\lambda$ is such that $\tilde{v}_j^0 - \lambda v_j^0 = 0$ for at least one $j \in A_{\lambda_0}$. Therefore, if as $\lambda$ decreases from $\lambda_0$ an active set deletion occurs before and active set addition we have

$$\lambda_1 = 0 \vee \max_{j \in \{1, \ldots, |A_{\lambda_0}|\}} \left\{\frac{\tilde{v}_j^0}{v_j^0}\right\}.$$

The proof is complete. □

## Computation of $\tilde{\beta}_\lambda$: An algorithm

Theorem 7.1 and Propositions 7.1-7.2 lead to the following algorithm for computing, for all $\lambda > 0$ such that the matrix $\boldsymbol{X}_{A_\lambda}^\top \boldsymbol{X}_{A_\lambda}$ is invertible, a solution $\tilde{\beta}_\lambda$ to the optimization problem (7.3) when $r = 1$.

### LASSO path algorithm

```
(i) Let $j_0$ be such that $|x_{(j_0)}^\top y| = \max_{j\in\{1,\dots,p\}} |x_{(j)}^\top y|$ and $\lambda = |x_{(j_0)}^\top y|$.

(ii) Let $\lambda_0 = \lambda$, $A_{\lambda_0} = \{j_0\}$ and $\tilde{\beta}_{\lambda_0}$ be such that

        $\tilde{\beta}_{A_{\lambda_0}} = (x_{(j_0)}^\top x_{(j_0)})^{-1}\big(x_{(j_0)}^\top y - \lambda_0 \operatorname{sign}\big(\tilde{\beta}_{A_{\lambda_0}}\big)\big).$

while $\lambda > 0$ do
    (iii) Let $\lambda_1$ and $A_{\lambda_1}$ be as in Proposition 7.2.

    (iv) For all $\lambda \in (\lambda_1, \lambda_0]$ let $\tilde{\beta}_\lambda$ be such that

        $\tilde{\beta}_{A_\lambda} = (\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1}\big(\boldsymbol{X}_{A_{\lambda_0}} y - \lambda \operatorname{sign}(\tilde{\beta}_{A_{\lambda_0}})\big)$

    if $\operatorname{rank}\big(\boldsymbol{X}_{A_{\lambda_1}}^\top \boldsymbol{X}_{A_{\lambda_1}}\big) < |A_{\lambda_1}|$ then
        (v) Break
    else
        (vi) Let $\tilde{\beta}_{\lambda_1}$ be such that

            $\tilde{\beta}_{A_{\lambda_1}} = (\boldsymbol{X}_{A_{\lambda_1}}^\top \boldsymbol{X}_{A_{\lambda_1}})^{-1}\big(\boldsymbol{X}_{A_{\lambda_1}} y - \lambda_1 \operatorname{sign}\big(\tilde{\beta}_{A_{\lambda_1}}\big)\big).$

        (vii) Let $\lambda_0 = \lambda_1$ and $\lambda = \lambda_1$.
    end if
end while

Return: $(\lambda, \tilde{\beta}_\lambda)_{\lambda\in(\lambda_1,\bar{\lambda}]}$
```

## Practical comments

1. The above algorithm requires to compute $(\boldsymbol{X}_{A_{\lambda_1}}^\top \boldsymbol{X}_{A_{\lambda_1}})^{-1}$ for all values of $\lambda_1$. Using the fact that the matrix is $\boldsymbol{X}_{A_{\lambda_1}}$ is obtained from the matrix $\boldsymbol{X}_{A_{\lambda_0}}$ either by adding or by removing one column, it is possible to compute $(\boldsymbol{X}_{A_{\lambda_1}}^\top \boldsymbol{X}_{A_{\lambda_1}})^{-1}$ from $(\boldsymbol{X}_{A_{\lambda_0}}^\top \boldsymbol{X}_{A_{\lambda_0}})^{-1}$ in $\mathcal{O}(|A_{\lambda_1}|^2)$ operations. By doing this, the complexity of computing the LASSO path is $\mathcal{O}(I^3)$, where $I$ is the number of different values for $\lambda_1$ computed in the above algorithm. In practice, it is often the case that $I = \min(n, p)$ but the worst case complexity of the LASSO path algorithm is exponential in $p$ [7].

   $\Longrightarrow$ The exact computation of $\tilde{\beta}_\lambda$ can be expensive when $p$ and $n$ is large and, in this case, $\tilde{\beta}_\lambda$ is often approximated using a coordinate descent algorithm (see below).

2. Due to the finite precision of computers, in practice the value $\lambda_1^{(2)}$ defined in Proposition 7.2, and the set $A_{\lambda_1}$ when $\lambda_1 = \lambda_1^{(2)}$, are often computed by excluding the last variable added in the path. (This is to avoid that, due to numerical errors, a variable is removed from the active set just after it has been added.)

3. As in ridge regression, and for the same reason, the LASSO estimator is usually computed after having standardized the variables $x_{(1)}, \dots, x_{(p)}$. The resulting estimate of the original model parameter is then obtained as explained in Chapter 6 (see page 112).

## The coordinate descent algorithm

Let $f : \mathbb{R}^p \to \mathbb{R}$ be a convex function and $x^* = \mathrm{argmin}_{x \in \mathbb{R}^p} f(x)$.

Then, assuming $p \geq 2$, the coordinate descent algorithm for computing $x^*$ is as follows:

```
Coordinate descent
Input: Starting value $x_0 \in \mathbb{R}^p$.
  for $k \geq 1$ do
     for $j = 1, \dots, p$ do
        (i) With obvious convention when $j \in \{1, p\}$, let
        $x_{k,j} = \underset{x_j \in \mathbb{R}}{\mathrm{argmin}} f(x_{k,1}, \dots, x_{k,j-1}, x_j, x_{k-1,j+1}, \dots, x_{k-1,p})$
     end for
     (ii): Set $x_k = (x_{k,1}, \dots, x_{k,p})$
  end for
```

Remark that each update of the approximation $x$ of $x^*$ performed at step (i) either leaves $f(x)$ unchanged or decreases $f(x)$.

Under suitable conditions on $f$[a] we have $x_k \to x^*$ as $k \to \infty$ (see [4], Section 5.4).

**Remark:** Instead of updating one component of $x$ at a time we can of course update several of them at the same time.

[a] e.g. $f$ is strictly convex in each of its components.

## LASSO with coordinate descent

We apply coordinate descent to the function $F_\lambda : \mathbb{R}^p \to \mathbb{R}$ defined by

$$F_\lambda(\beta) = \|y - \boldsymbol{X}\beta\|_2^2 + 2\lambda \sum_{j=1}^{p} |\beta_j|, \quad \beta \in \mathbb{R}^p$$

which, for $j \in \{1, \dots, p\}$ and $b \in \mathbb{R}^p$, requires to compute (with obvious convention when $j \in \{1, p\}$)

$$\beta_j^{(b)} \in \operatorname*{argmin}_{\beta_j \in \mathbb{R}} F_\lambda(b_1, \dots, b_{j-1}, \beta_j, b_{j+1}, \dots, b_p). \tag{7.10}$$

To this aim, for every $c \in \mathbb{R}$ we let $\mathcal{S}_c(z) = \operatorname{sign}(z)\big(|z| - c\big)_+$ be the soft-thresholding operator, recalling that for $z \in \mathbb{R}$ we have $z_+ = \max(0, z)$.

Using Theorem 7.1 we readily obtain the following result for the expression of $\beta_j^{(b)}$ defined in (7.10).

**Theorem 7.2** *Assume that the variables $x_{(1)}, \dots, x_{(p)}$ are normalized, so that $\sum_{i=1}^n x_{ij}^2 = 1$ for all $j \in \{1, \dots, p\}$. Let $j \in \{1, \dots, p\}$, $b \in \mathbb{R}^p$ and $\beta_j^{(b)}$ be as defined in* (7.10)*. Then,*

$$\beta_j^{(b)} = \mathcal{S}_\lambda\Big(\sum_{i=1}^{n} x_{ij}\big(y_i - \sum_{m \neq j} b_m x_{im}\big)\Big).$$

## Proof of Theorem 7.2

Let $y_i^{(b)} = y_i - \sum_{m\neq j} b_m x_{im}$ for all $i \in \{1,\dots,n\}$ and assume first that $\beta_j^{(b)} \neq 0$. Then, applying Theorem 7.1 with $p=1$, $y$ replaced by $y^{(b)}$ and $\boldsymbol{X}$ replaced by $x_{(j)}$, it follows that $\beta_j^{(b)}$ solves

$$\sum_{i=1}^n x_{ij}\Big(y_i - \sum_{m\neq j} b_m x_{im} - \beta_j^{(b)} x_{ij}\Big) - \lambda\,\mathrm{sign}(\beta_j^{(b)}) = 0$$

and thus, recalling that the variables $x_{(1)},\dots,x_{(p)}$ are assumed to be normalized

$$\begin{aligned}\beta_j^{(b)} &= \mathrm{sign}\Big(\sum_{i=1}^n x_{ij}\big(y_i - \sum_{m\neq j} b_m x_{im}\big)\Big)\Big(\Big|\sum_{i=1}^n x_{ij}\big(y_i - \sum_{m\neq j} b_m x_{im}\big)\Big| - \lambda\Big)\\ &= \mathcal{S}_\lambda\Big(\sum_{i=1}^n x_{ij}\big(y_i - \sum_{m\neq j} b_m x_{im}\big)\Big).\end{aligned}$$

Assume now that $\beta_j^{(b)} = 0$. Then, applying again Theorem 7.1 with $p=1$, $y$ replaced by $y^{(b)}$ and $\boldsymbol{X}$ replaced by $x_{(j)}$, we must have

$$\sum_{i=1}^n x_{ij}\Big(y_i - \sum_{m\neq j} b_m x_{im}\Big) - \lambda \leq 0. \tag{7.11}$$

We remark that (7.11) holds if and only if

$$\mathcal{S}_\lambda\Big(\sum_{i=1}^n x_{ij}\big(y_i - \sum_{m\neq j} b_m x_{im}\big)\Big) = 0$$

and thus when $\beta_j^{(b)} = 0$ we also have

$$\beta_j^{(b)} = \mathcal{S}_\lambda\Big(\sum_{i=1}^n x_{ij}\big(y_i - \sum_{m\neq j} b_m x_{im}\big)\Big).$$

The proof is complete. □

## Computing $\tilde{\beta}_\lambda$ using coordinate descent

Using Theorem 7.2 we obtain the following coordinate descent algorithm for computing, for a given $\lambda > 0$, a solution $\tilde{\beta}_\lambda$ of the optimization problem (7.3) when $r = 1$.

### Coordinate descent algorithm for computing $\tilde{\beta}_\lambda$

```
Input: Starting value $\beta_0 \in \mathbb{R}^p$.
  (i) Let $r_i = y_i - \sum_{j=2}^{p} \beta_{0,j} x_{ij}$ for $i = 1, \ldots, n$
  for $k \geq 1$ do
      (ii) Let $\beta_{k,1} = \mathcal{S}_\lambda\big( \sum_{i=1}^{n} x_{i1} r_i \big)$
      for $j = 2, \ldots, p$ do
          (iii) $r_i \leftarrow r_i - \beta_{k,j-1} x_{i(j-1)} + \beta_{k-1,j} x_{ij}$ for $i = 1, \ldots, n$
          (iv) Let $\beta_{k,j} = \mathcal{S}_\lambda\big( \sum_{i=1}^{n} x_{ij} r_i \big)$
      end for
      (v) $r_i \leftarrow r_i - \beta_{k,p} x_{ip} + \beta_{k,1} x_{i1}$ for $i = 1, \ldots, n$
      (vi) Set $\beta_k = (\beta_{k,1}, \ldots, \beta_{k,p})$.
      if Convergence=TRUE then
          (vii) return $\beta_k$.
          (viii) break
      end if
  end for
```

**Remark:** It can be shown that the above algorithm is indeed valid, in the sense that for a $\tilde{\beta}_\lambda \in \operatorname{argmin}_{\beta \in \mathbb{R}^p} \|y - \boldsymbol{X}\beta\|_2^2 + 2\lambda \sum_{j=1}^{p} |\beta_j|$ we have $\beta_k \to \tilde{\beta}_\lambda$ as $k \to \infty$ (see [4], Section 5.4).

**Remark:** Each iteration of the algorithm (i.e. computing $\beta_k$ from $\beta_{k+1}$) costs only $\mathcal{O}(p\,n)$ operations, making is suitable for large $p$ and/or large $n$ problems.

## Computing the LASSO path with coordinate descent

### LASSO path Coordinate descent

```
Input: Starting value $\beta \in \mathbb{R}^p$, integer $M \in \mathbb{N}$ and $\epsilon \in (0,1)$.
    (i) Let $\lambda_1 = \max_{j\in\{1,\dots,p\}} |x_{(j)}^\top y|$
    (ii) Compute an approximation $\beta'_{\lambda_1}$ of $\tilde{\beta}_{\lambda_1}$ using coordinate descent
    with starting value $\beta_0 = \beta$.
    for $m = 2, \dots M$ do
        (iii): Let $\lambda_m = \exp\Big(\log\lambda_{m-1} - \log(-\epsilon)/(M-1)\Big)$.
        (iv) Compute an approximation $\beta'_{\lambda_m}$ of $\tilde{\beta}_{\lambda_m}$ using coordinate descent with starting value $\beta_0 = \beta'_{\lambda_{m-1}}$.
    end for
 Return: $\{\lambda_m, \beta'_{\lambda_m}\}_{m=1}^M$
```

In the above algorithm $\lambda$ decreases from $\lambda_1$ to $\lambda_M = \epsilon\lambda_1$ linearly on a log scale (other choices for the sequence $\{\lambda_m\}_{m=1}^M$ are of course possible).

The tuning parameter $\epsilon$ is usually a small number (for instance $\epsilon = 0.0001$ if $n > p$ and $\epsilon = 0.01$ if $n < p$) while $M$ is a large number (e.g. $M = 100$).

**Remark:** The definition of $\{\lambda_m\}_{m=1}^M$ used in the above algorithm, as well as the aforementioned proposed default values for $\epsilon$ and $M$, are as in the R package `glmnet`.

## Choice of the parameter $\lambda$

$K$-fold cross-validation is often use in LASSO to choose $\lambda$.

In this case, we randomly divide the data sets into $K > 1$ groups of equal size, where typically $K = 5$ or $K = 10$.

Then, we successively treat each group as the test set and compute the (approximate) LASSO path using the observations from the remaining $K - 1$ groups. For each $\lambda$ we record the mean squared prediction error on the test set (i.e. on the group excluded from the estimation step).

At the end of the process we obtain $K$ estimates of the prediction error for a range of $\lambda$, which are averaged to produce the cross validation curve. We finally retain the value of $\lambda$ at which this curve reach its minimum.

**Remark:** $K$-fold cross-validation with $K = n$ reduces to the leave-one-out cross validation procedure discussed in Chapter 6.

**Remark:** We saw in Chapter 6 that, for ridge regression, cross-validation can be implemented in such a way that only one ridge regression needs to be performed for each value of the penalty parameter $\lambda > 0$[a]. Such a simplification of the cross-validation procedure does not exist for LASSO, and therefore for each value of $\lambda$ it is necessary to compute $K$ LASSO estimators.

[a] We saw this result for $K = n$ but it generalizes to any $K$.

## Bayesian perspective of LASSO

Consider the following Bayesian linear regression model

$$\beta_j \overset{\text{iid}}{\sim} \text{Laplace}(0, 2\sigma^2/\lambda), \quad Y_i \sim \mathcal{N}_1(x_i^\top \beta, \sigma^2), \quad i = 1, \dots, n$$

for some $\sigma^2 > 0$.

Recalling that the posterior distribution of $\beta$ given the observations $y$ is $\pi(\beta|y) \propto \pi(y|\beta)\pi(\beta)$, we have

$$\log \pi(\beta|y) = c - \frac{1}{2\sigma^2}\|y - \boldsymbol{X}\beta\|^2 - \frac{\lambda}{2\sigma^2}\sum_{j=1}^{p}|\beta_j|$$

for some constant $c \in \mathbb{R}$ (i.e. $c$ is independent of $\beta$).

Therefore, the posterior mode $\beta_{\text{mode}}$ of $\beta$ satisfies

$$\begin{aligned}\beta_{\text{mode}} &\in \operatorname*{argmax}_{\beta \in \mathbb{R}^p} \pi(\beta|y)\\ &= \operatorname*{argmax}_{\beta \in \mathbb{R}^p} \log \pi(\beta|y)\\ &= \operatorname*{argmax}_{\beta \in \mathbb{R}^p} -\|y - \boldsymbol{X}\beta\|^2 - \lambda\sum_{j=1}^{p}|\beta_j|\\ &= \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \|y - \boldsymbol{X}\beta\|^2 + \lambda\sum_{j=1}^{p}|\beta_j|.\end{aligned}$$

Consequently, while the posterior mode of $\beta$ in the Bayesian linear regression model with a Gaussian prior is the ridge regression estimator, it is the LASSO estimator when a Laplace prior is used.

**Remark:** Unlike for ridge regression, the posterior distribution for $\beta$ in the Bayesian model associated to LASSO regression is not tractable.

## Example: The prostate dataset[a]

The objective of this example is to examine the impact on the level of a prostate specific antigen of $p = 8$ clinical measures in $n = 67$ men who were about to receive a radical prostatectomy.

The variables $\{x_{(j)}\}_{j=1}^{p}$ are centred before estimating the regression parameter $\beta$ and $K$-fold cross-validation with $K = 10$ is used to choose $\lambda$.

From Figure 7.1 we observe the coordinate descent provides a very good estimate of the lasso path $\{\tilde{\beta}_\lambda\}_{\lambda>0}$. In addition, the value of $\lambda$ chosen by cross-validation is $\lambda \approx 0.89$, in which case one variable is not selected (i.e. one component of $\tilde{\beta}_\lambda$ is zero).

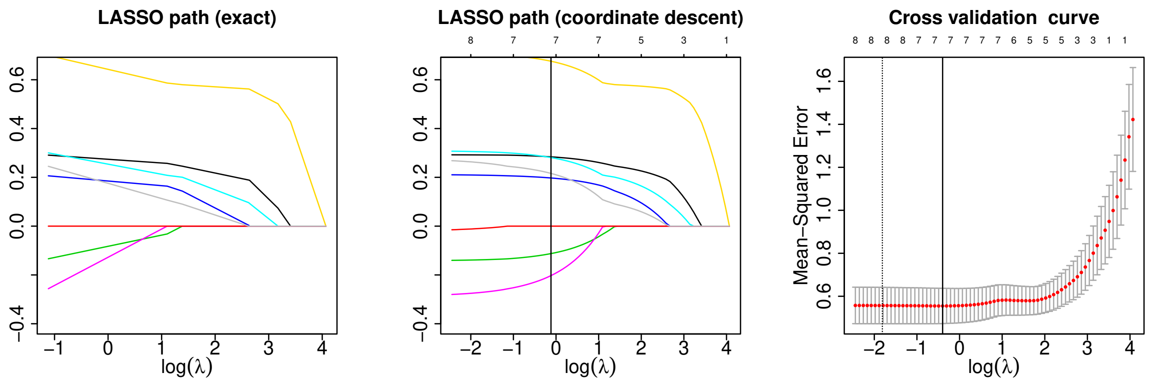

Figure 7.1: LASSO path for the prostate cancer dataset and cross-validation plot. In the middle and right plot the (black) vertical line represents the value of $\lambda$ which minimizes the cross-validation error.

[a] This example is taken from [4] and the dataset is available at `https://hastie.su.domains/ElemStatLearn/`

## Ridge v.s. LASSO regression: The prostate dataset

Figure 7.2 below compares the LASSO path and ridge path for the prostate cancer dataset.

As expected, and unlike in LASSO regression, we observe that none of the component of the ridge regression estimate $\hat{\beta}_\lambda$ is shrink to zero when $\lambda$ is large enough. It is also interesting to see that the ordering of the $\{\tilde{\beta}_{\lambda,j}\}_{j=1}^p$ is similar to that of $\{\hat{\beta}_{\lambda,j}\}_{j=1}^p$ and thus that, on this example, LASSO shrinks to zero the elements of $\{\hat{\beta}_{\lambda,j}\}_{j=1}^p$ which are close to zero.

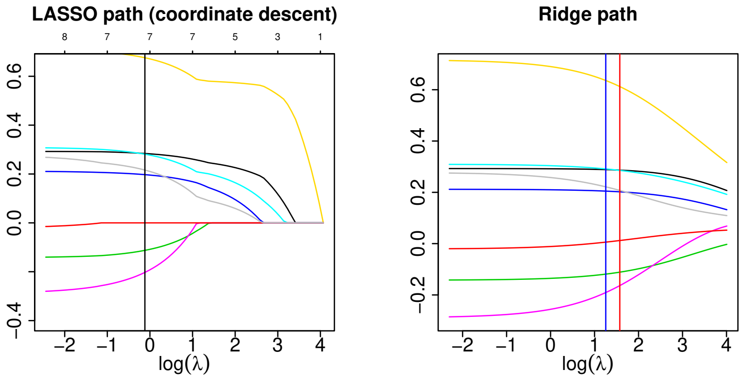

Figure 7.2: Ridge and LASSO paths for the prostate cancer dataset. For ridge regression, the horizontal blue line shows the optimal $\lambda$ according to the OCV criterion and the red line the optimal $\lambda$ according to the GCV criterion.

**Remark:** This example also illustrates the fact that, in ridge regression, ordinary cross-validation and generalized cross validation may lead to different choices of $\lambda$.