Chapter 1
Uncertainty quantification in lasso-type regularization problems

Tathagata Basu, Jochen Einbeck and Matthias C. M. Troffaes

Abstract Regularization techniques, which sit at the interface of statistical modeling and machine learning, are often used in the engineering or other applied sciences to tackle high dimensional regression (type) problems. While a number of regularization methods are commonly used, the 'Least Absolute Shrinkage and Selection Operator' or simply LASSO is popular because of its efficient variable selection property. This property of the LASSO helps to deal with problems where the number of predictors is larger than the total number of observations, as it shrinks the coefficients of non-important parameters to zero. In this chapter, both frequentist and Bayesian approaches for the LASSO are discussed, with particular attention to the problem of uncertainty quantification of regression parameters. For the frequentist approach, we discuss a refit technique as well as the classical bootstrap method, and for the Bayesian method, we make use of the equivalent LASSO formulation using a Laplace prior on the model parameters.

Tathagata Basu
Durham University, UK e-mail: tathagata.basu@durham.ac.uk
Jochen Einbeck
Durham University, UK e-mail: jochen.einbeck@durham.ac.uk
Matthias C. M. Troffaes
Durham University, UK e-mail: matthias.troffaes@durham.ac.uk

Contents

1.1 Introduction

Statistics is a collection of mathematical concepts to analyze and find the structure in data. Data can be either numeric or character-valued (representing a class) depending on the problem. There are several purposes of statistics; however one of the main purposes is description of the data and prediction of system behavior from the observed data. Elements of statistical reasoning have been traced back as early as 400 AD [14, p. 7] in India. However, the modern-day approach only started emerging in the 18th century, following advances in the theory of probability [14, p. 176].

In this chapter, we will discuss statistical regularization and uncertainty quantification problems using LASSO ('Least Absolute Shrinkage and Selection Operator') estimators [24, 25]. The LASSO estimator is a popular regularization method due to its variable selection property. After Tibshirani introduced LASSO in 1996 [24], numerous authors contributed further to the theory, including Osborne, Presnell, and Turlach [21] and Efron et al. [7]. Friedman et al. [10] discussed computational aspects of the LASSO. Park and Casella [22] introduced the Bayesian approach for LASSO estimators using a hierarchical mixture model for parameter estimation. Other notable works deal with the specification of shrinkage parameter by Lykou and Ntzoufras [18]; the Dirichlet LASSO by Das and Sobel [5]; and the spike and slab LASSO by Ročková [23].

First, we will introduce the basic notions behind statistical modeling and regularization. In Section 1.2, we will look at some important concepts of parameter estimation with and without regularization. Eventually, we will introduce the LASSO estimators in Section 1.3. In Section 1.4, we will discuss different uncertainty quantification methods for the LASSO followed by an extension to the logistic model in Section 1.5. Section 1.6 concludes the chapter.

1.1.1 Statistical Modeling

To make statistical inferences from data, first, we need variables, and a model describing the relations between those variables. We can categorize variables into *response variables* and *predictor variables*:

1. Predictor (or independent) variables are characteristics of the system which directly control the properties of the system.
2. Response (or dependent) variables are characteristics of the system which depend on the predictor variables. In other words, they respond to a change of values of the predictors in some systematic fashion.

Assume we have a dataset containing n independent and identically distributed (i.i.d.) observations of real-valued responses $y_1, \ldots, y_n \in \mathbb{R}$, along with

corresponding vector-valued predictors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^p$. We consider each $\boldsymbol{x}_i$ to be a column vector.

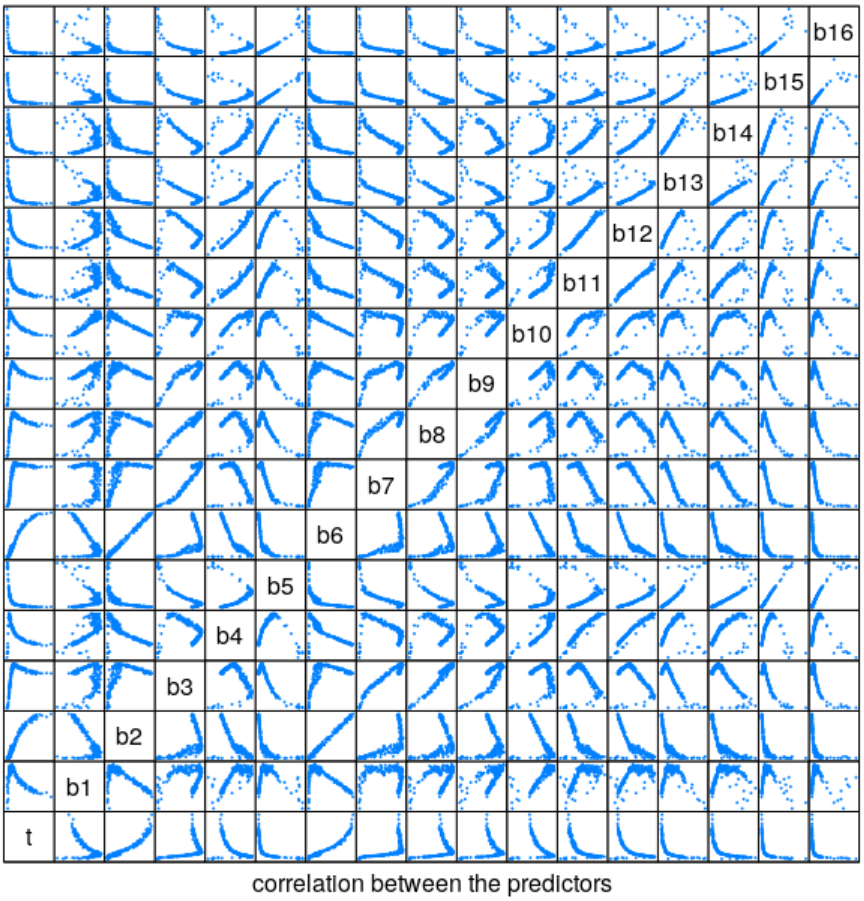

Fig. 1.1 Scatter plot matrix of the Gaia dataset. The variable denoted `t` (temperature) corresponds to the response; the variables denoted `b1` to `b16` (bands) correspond to the predictors. Note that the plot is symmetric w.r.t. the counterdiagonal.

Example 1 (Gaia Dataset). Gaia is a mission by the European Space Agency (ESA) to formulate a three dimensional map of our galaxy [1]. The data depicted in Fig. 1.1 are part of a dataset which was simulated prior to the launch of the mission from computer experiments [8, 2]. The data contain essentially spectral information divided into $p = 16$ wavelength bands (intervals), along with certain stellar parameters which are to be inferred from the spectral data. That is, each observation in the data set represents a stellar object, and

the measurement for each 'band' is the energy flux (photon counts) emitted from that object within that wavelength interval.

In this example, stellar-temperature (in Kelvin scale) is the response variable. In the dataset that we have available, a total of $n = 8286$ observations (stellar objects) are recorded. It can be seen from Fig. 1.1 that the 16 predictors variables are strongly correlated with each other, suggesting that they carry redundant information.

Often, one of the objectives of statistical modeling is to identify a functional relationship ('model') between the responses and the predictor variables:

$$E(y_i|\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i, \boldsymbol{\beta}) \tag{1.1}$$

where ϕ is a function that depends on a parameter vector $\boldsymbol{\beta}$. For instance, as will be described in Section 1.1.3 in more detail, in a linear regression context one typically has $\phi(\boldsymbol{x}_i, \boldsymbol{\beta}) = \boldsymbol{x}_i^T\boldsymbol{\beta}$. There also exist non-parametric approaches which do not assume an explicit parametric shape, but most of such approaches achieve this by simply introducing a large number of parameters, so that they still can be expressed as in Eq. (1.1).

1.1.2 Statistical Inference

Statistical inference is the process by which we use the available data to gain knowledge about the model parameters, such as $\boldsymbol{\beta}$ in Eq. (1.1), as well as their uncertainties. In a wider sense it will also include methods by which we quantify and validate our assumptions on the model. Statistical inference deals with the estimation of parameters that are used to specify the family of probability distributions which underlie the statistical model for $y_i|\boldsymbol{x}_i$. Inference has several applications in science and engineering. Generally, there are two conceptually different approaches to statistical inference: the *frequentist* approach and the *Bayesian* approach. There are some other concepts available which are beyond the scope of this chapter, but are addressed in other articles in this volume.

The frequentist approach is the most widely used estimation method. Sometimes it is referred to as the 'classical' approach. The estimation can be a point estimate where we simply try to find the best guess for the parameter of the parametric model. Alternatively, we seek an interval which covers the unknown parameter value with high probability (generally 0.95). We call this a 95% confidence interval.

While several point estimators are available, the *maximum likelihood estimator* (or, MLE) is among the most popular because of its simple and wide implementability and its consistency properties. It finds the parameter value which maximizes the probability density of the sample given the parameter,

i.e. the likelihood. For linear regression models under normal errors, MLE is equivalent to the ordinary least squares.

The Bayesian approach starts from Bayes's rule for conditional probability. Denote the data by $\boldsymbol{Y}$. For example, in our setting, $\boldsymbol{Y}$ is simply the vector of observed response values $(y_1, \ldots, y_n)^T$. The statistical model is specified through a likelihood function $p(\boldsymbol{Y} \mid \boldsymbol{\beta})$. In the context of the regression model in Eq. (1.1), this likelihood would be considered conditional on the observed values of the predictors, that is, the observed values of the predictors are considered as fixed. Finally, we need a prior distribution $p(\boldsymbol{\beta})$ for the model parameters $\boldsymbol{\beta}$ to incorporate our prior knowledge. Bayes's rule then tells us that the posterior distribution $p(\boldsymbol{\beta} \mid \boldsymbol{Y})$ is given by

$$p(\boldsymbol{\beta} \mid \boldsymbol{Y}) \propto p(\boldsymbol{\beta}) \times p(\boldsymbol{Y} \mid \boldsymbol{\beta}). \tag{1.2}$$

The normalization constant can be calculated from the law of total probability if necessary. However, this calculation may not be always trivial so that simulation methods like MCMC need to be employed. The posterior distribution is then used for further inference. For instance, we can look at its mean, mode, or other characteristics.

1.1.3 Linear Models

The linear model is one of the most popular forms for statistical modeling. Here, the functional relationship between the response and predictor is linear i.e. $y_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \epsilon_i$, where $\boldsymbol{\beta} \in \mathbb{R}^p$, and usually the assumption $\epsilon_i \overset{i.i.d.}{\sim} N(0, \sigma^2)$ is made for the random errors. The linear model can be written in a matrix form for all cases $i \in \{1, \ldots, n\}$ simultaneously as follows:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1.3}$$

where

$$\boldsymbol{Y} \coloneqq \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \boldsymbol{X} \coloneqq \begin{bmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_n^T \end{bmatrix} \qquad \boldsymbol{\beta} \coloneqq \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_p \end{bmatrix} \qquad \boldsymbol{\epsilon} \coloneqq \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}. \tag{1.4}$$

The matrix $\boldsymbol{X}$ is called the *design matrix*. Remember that each $\boldsymbol{x}_i \in \mathbb{R}^p$ is considered as a column-vector, so $\boldsymbol{X}$ is an $n \times p$ matrix.

1.1.4 Strong Duality and the Karush-Kuhn-Tucker Conditions

In this section, we briefly give the main duality result for non-linear optimization that we will apply further. Assume we aim to minimize a function $f(\boldsymbol{\beta})$, where $\boldsymbol{\beta} \in B \subseteq \mathbb{R}^p$ subject to a constraint $h(\boldsymbol{\beta}) \leq 0$. In the following sections, we will have either $B = \mathbb{R}^p$ or $B = \mathbb{R}^p_+$ (i.e. the set of non-negative vectors in $\mathbb{R}^p$), although in principle B can be an arbitrary convex set. So, we try to find

$$f^* \coloneqq \min_{\substack{\boldsymbol{\beta} \in B \\ h(\boldsymbol{\beta}) \leq 0}} f(\boldsymbol{\beta}). \tag{1.5}$$

One may think of the function $f(\cdot)$ as a least squares criterion or a negative (log-)likelihood. Define now the *Lagrangian*:

$$\ell(\boldsymbol{\beta}, \lambda) \coloneqq f(\boldsymbol{\beta}) + \lambda h(\boldsymbol{\beta}) \tag{1.6}$$

and the *Lagrange dual function*:

$$g(\lambda) \coloneqq \min_{\boldsymbol{\beta} \in B} \ell(\boldsymbol{\beta}, \lambda). \tag{1.7}$$

Note that

$$\max_{\lambda \geq 0} g(\lambda) = \max_{\lambda \geq 0} \min_{\boldsymbol{\beta} \in B} \ell(\boldsymbol{\beta}, \lambda) \leq \max_{\lambda \geq 0} \min_{\substack{\boldsymbol{\beta} \in B \\ h(\boldsymbol{\beta}) \leq 0}} \ell(\boldsymbol{\beta}, \lambda) \tag{1.8}$$

$$\leq \max_{\lambda \geq 0} \min_{\substack{\boldsymbol{\beta} \in B \\ h(\boldsymbol{\beta}) \leq 0}} f(\boldsymbol{\beta}) = f^*. \tag{1.9}$$

This inequality holds in general. Strong duality tells us that, under certain conditions, the inequality becomes an equality [3, §5.2.3].

Theorem 1 (Strong Duality). *If f and h are convex functions, and $h(\boldsymbol{\beta}) < 0$ for at least one $\boldsymbol{\beta} \in B$, then*

$$\max_{\lambda \geq 0} g(\lambda) = \min_{\substack{\boldsymbol{\beta} \in B \\ h(\boldsymbol{\beta}) \leq 0}} f(\boldsymbol{\beta}) = f^* \tag{1.10}$$

So, under strong duality, to minimize $f(\boldsymbol{\beta})$ over $\boldsymbol{\beta}$ subject to $h(\boldsymbol{\beta}) \leq 0$, we can also instead maximize the Lagrange dual function over $\lambda \geq 0$. In that case, the *Karush-Kuhn-Tucker conditions* provide necessary and sufficient conditions for optimality.

Definition 1 (Subgradient). For any function F on B, we say that $v \in \mathbb{R}^p$ is a *subgradient* of F at $\boldsymbol{\beta}$ whenever

$$F(\boldsymbol{\beta}') - F(\boldsymbol{\beta}) \geq \boldsymbol{v}^T(\boldsymbol{\beta}' - \boldsymbol{\beta}) \tag{1.11}$$

for all $\boldsymbol{\beta}' \in B$. The set of all subgradients of F at $\boldsymbol{\beta}$ is denoted by $\partial F(\boldsymbol{\beta})$.

Theorem 2 (Karush-Kuhn-Tucker). *If f and h are convex functions, and $h(\boldsymbol{\beta}) < 0$ for at least one $\boldsymbol{\beta} \in B$, then $f(\boldsymbol{\beta}) = f^*$ if and only if*

$$\mathbf{0} \in \partial f(\boldsymbol{\beta}) + \lambda \partial h(\boldsymbol{\beta}) \tag{1.12}$$

$$\lambda h(\boldsymbol{\beta}) = 0 \tag{1.13}$$

$$h(\boldsymbol{\beta}) \leq 0 \tag{1.14}$$

$$\lambda \geq 0 \tag{1.15}$$

So, Eq. (1.12) is just a fancy way of writing that $\boldsymbol{\beta}$ is a global minimum of $f + \lambda h$, for a fixed value of λ. Equation (1.12) is called the *stationarity condition*. Equation (1.13) is called the *complementary slackness condition*, and implies that either $\lambda = 0$ or $h(\boldsymbol{\beta}) = 0$. The inequality $h(\boldsymbol{\beta}) \leq 0$ is called *primal feasibility*, and the inequality $\lambda \geq 0$ is called *dual feasibility*.

To solve the Karush-Kuhn-Tucker conditions, we split the problem into two cases as per Eq. (1.13), $\lambda = 0$ and $h(\boldsymbol{\beta}) = 0$. We then solve Eq. (1.12) under each equality constraint. We throw away any solution that does not satisfy primal or dual feasibility, and then choose the solution that achieves the lowest value.

For the case $\lambda = 0$, we need to find the global unconstrained minimum of f. If the primal feasibility constraint $h(\boldsymbol{\beta}) \leq 0$ is satisfied at the global minimum of f, then we have found a solution. Obviously, this solution must be the optimal solution of the original constrained problem as well.

If $h(\boldsymbol{\beta}) > 0$ at the global minimum of f, then we need to find the minimum of f under the constraint that $h(\boldsymbol{\beta}) = 0$. We could do so by finding a joint solution to the system of equations formed by Eq. (1.12) and $h(\boldsymbol{\beta}) = 0$. Alternatively, we could gradually increase λ until the global unconstrained minimum $g(\lambda)$ of $f + \lambda h$ satisfies $h(\boldsymbol{\beta}) = 0$. Indeed, due to the form of the objective function, increasing λ will favor $\boldsymbol{\beta}$ that have lower values for $h(\boldsymbol{\beta})$, so eventually, $h(\boldsymbol{\beta}) = 0$. By strong duality, we also know that finding this λ is equivalent to maximizing the Lagrange dual function $g(\lambda)$ over $\lambda \geq 0$.

1.2 Parameter Estimation

In a statistical modeling problem our task is to estimate $\boldsymbol{\beta}$ from the data $\boldsymbol{Y}$ and $\boldsymbol{X}$. There are several methods to estimate these parameters in a linear model. We will discuss some of them and their properties.

1.2.1 Ordinary Least Squares

In ordinary least squares [6], we estimate the parameters by minimizing the sum of the squared errors:

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} \coloneqq \arg\min_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) \tag{1.16}$$

where

$$R(\boldsymbol{\beta}) \coloneqq \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i^T \boldsymbol{\beta})^2 = \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2. \tag{1.17}$$

We have used $\|\cdot\|_2$ to denote the standard Euclidean norm, that is $\|\boldsymbol{z}\|_2 \coloneqq \sqrt{\sum_{i=1}^{n} z_i^2}$. A necessary condition to have a minimum for Eq. (1.17) is

$$\frac{\partial}{\partial \boldsymbol{\beta}} R(\boldsymbol{\beta}) = -2\boldsymbol{X}^T\boldsymbol{Y} + 2(\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{\beta} = 0. \tag{1.18}$$

Therefore, if $\boldsymbol{X}^T\boldsymbol{X}$ is invertible (this requires that the number of observations, n, is larger or equal than the total number of predictors, p), then the ordinary least squares estimator is given by

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}, \tag{1.19}$$

where $(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$ is the Moore-Penrose inverse of $\boldsymbol{X}$.

The *Gauss-Markov theorem* states that when the errors are uncorrelated with expectation zero and constant variance, then the ordinary least squares estimate is the best linear unbiased estimator.

Two issues that often arise are:

1. If $p > n$ then $\boldsymbol{X}^T\boldsymbol{X}$ is singular, hence Eq. (1.18) has no unique solution.
2. Even if $p \leq n$, p may still be much larger than needed, and we may wish to identify sparse solutions where unnecessary parameters are set to zero. In other words, we may wish to perform variable selection as part of our statistical inference.

1.2.2 Non-Negative Garrote

The non-negative garrote was introduced by Breiman [4]. It is a two stage procedure that gives a sparse solution. It has a close relationship to the LASSO, however as a starting point of the problem the ordinary least square estimates are needed. Given the initial estimate $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} \in \mathbb{R}^p$, we solve the following optimization problem over $\boldsymbol{c} = (c_1, c_2, \cdots, c_p)^T$:

$$\hat{\boldsymbol{c}} = \arg \min_{\substack{\boldsymbol{c} \geq 0 \\ \|\boldsymbol{c}\|_1 \leq t}} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{C}\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}\|_2^2 \tag{1.20}$$

where $\boldsymbol{C} \coloneqq \mathrm{diag}(\boldsymbol{c}) \in \mathbb{R}^{p \times p}$, and where $\|.\|_1$ denotes the l_1-norm; that is $\|\boldsymbol{c}\|_1 = \sum_{i=1}^{p} |c_i|$. We get the final non-negative garrote parameter estimate $\hat{\boldsymbol{\beta}}$ by setting $\hat{\beta}_i = \hat{c}_i \hat{\beta}_i^{\mathrm{OLS}}$ for each $i \in \{1, 2, \ldots, p\}$.

Equivalently, we can solve the dual problem, by introducing a Lagrangian multiplier λ for the constraint $\|\boldsymbol{c}\|_1 - t \leq 0$ [15], similar to what we discussed in Section 1.1.4:

$$\max_{\lambda \geq 0} \min_{\boldsymbol{c} \geq 0} \left(\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{C}\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}\|_2^2 + \lambda(\|\boldsymbol{c}\|_1 - t) \right) \tag{1.21}$$

Effectively, we thus need to solve

$$\hat{\boldsymbol{c}}_\lambda = \arg \min_{\boldsymbol{c} \geq 0} \left(\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{C}\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}\|_2^2 + \lambda \|\boldsymbol{c}\|_1 \right) \tag{1.22}$$

where the Lagrange multiplier $\lambda \geq 0$ can be interpreted as a regularization weight. If $\|\hat{\boldsymbol{c}}_\lambda\|_1 \leq t$ for $\lambda = 0$, then we are done. Otherwise, λ is calibrated until $\|\hat{\boldsymbol{c}}_\lambda\|_1 = t$, as we discussed in Section 1.1.4. This value for λ is also the value that achieves the maximum in Eq. (1.21). If the columns of the design matrix $\boldsymbol{X}$ are orthogonal (i.e. $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}$), then the explicit solution of Eq. (1.22) is given by [26]:

$$\hat{c}_{\lambda i} = \max \left\{ 0, 1 - \frac{\lambda}{(\hat{\beta}_i^{\mathrm{OLS}})^2} \right\}. \tag{1.23}$$

Consequently, in this case, if the coefficient $\hat{\beta}_i^{\mathrm{OLS}}$ of a predictor is less than $\sqrt{\lambda}$, then $\hat{c}_{\lambda i} = 0$, and therefore also $\hat{\beta}_i = \hat{c}_{\lambda i} \hat{\beta}_i^{\mathrm{OLS}} = 0$. In this way, larger λ will produce sparser solutions.

The starting point of this method depends on the least square estimates $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}$. Therefore, if $p > n$, then no unique solution is available. However, alternative initial estimators such as the LASSO can be used in this case [26].

1.2.3 Regularization under l_q Penalty

Unfortunately, the non-negative garrote in Eq. (1.20) still fails to deliver when we have no least squares estimate to start from, which happens for instance when we have more predictors than observations. To solve this, we can use a different method, where no initial estimate is needed. The basic idea is to add a penalty term to the least squares problem, in order to penalize non-zero parameter values. This can be done in the following way:

$$\hat{\boldsymbol{\beta}}_\lambda = \arg\min_{\boldsymbol{\beta}} \left(\frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_q^q \right) \tag{1.24}$$

where $q \geq 0$ determines the shape of the penalty, and $\lambda \geq 0$ determines the strength of the penalty. Here,

$$\|\boldsymbol{z}\|_q^q := \begin{cases} \sum_{i=1}^n |z_i|^q & \text{if } q > 0 \\ \sum_{i=1}^n I_{z_i \neq 0} & \text{if } q = 0 \end{cases} \tag{1.25}$$

where $I_{z_i \neq 0} = 1$ if $z_i \neq 0$, and 0 otherwise. So, $\|\boldsymbol{z}\|_0^0$ simply counts the number of non-zero components of $\boldsymbol{z}$.

For different values of q we have different types of regularization. This leads to ridge regression for $q = 2$, LASSO for $q = 1$, and best subset selection method for $q = 0$ [15].

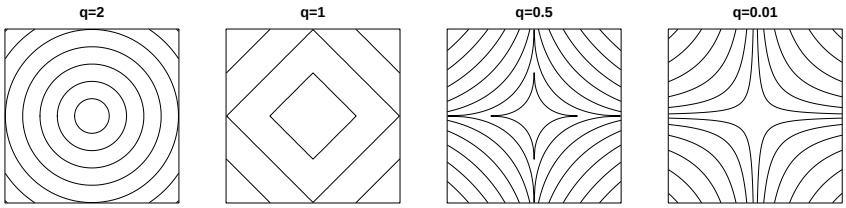

Fig. 1.2 Contour plots of different l_q penalty functions.

In Fig. 1.2, we illustrate some contour plots of the l_q penalty function, for different values of q. As will be illustrated in Section 1.3.1, it is the 'spiked' shape of the contours which leads to sparsity; in other words all penalties with $q \leq 1$ will lead to sparse estimators. However, for $q < 1$, the l_q penalty function is no longer convex, as can be seen from the contour plots. Therefore, $q = 1$ is the only value for which the problem is convex and allows sparse solutions.

1.3 The LASSO

The LASSO estimator was first proposed by Tibshirani [24]. The objective is to solve the ordinary least squares problem, but subject to an additional constraint on the 1-norm of the parameters, as follows:

$$\min_{\boldsymbol{\beta}:\ \|\boldsymbol{\beta}\|_1 \leq t} \left(\frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 \right). \tag{1.26}$$

It is usually assumed that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are standardized to mean 0. Otherwise, they can always be standardized without any loss of generality.

1.3.1 Solving The LASSO Optimization Problem

By strong duality (see Theorem 1 in Section 1.1.4), equivalently, we can solve the dual problem, by introducing a Lagrangian multiplier λ for the constraint $\|\boldsymbol{\beta}\|_1 - t \leq 0$:

$$\max_{\lambda \geq 0} \min_{\boldsymbol{\beta}} \left(\frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda(\|\boldsymbol{\beta}\|_1 - t) \right). \tag{1.27}$$

For the inner minimization problem, we need to find

$$\hat{\boldsymbol{\beta}}_\lambda \coloneqq \arg\min_{\boldsymbol{\beta}} \left(\frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \right). \tag{1.28}$$

From the discussion in Section 1.1.4, we know that if $\|\hat{\boldsymbol{\beta}}_0\|_1 \leq t$, then the solution is immediately given by $\hat{\boldsymbol{\beta}}_0$ (note that $\hat{\boldsymbol{\beta}}_0 = \hat{\boldsymbol{\beta}}^{\mathrm{OLS}}$). If $\|\hat{\boldsymbol{\beta}}_0\|_1 > t$, then we need find that value for $\lambda \geq 0$ for which $\|\hat{\boldsymbol{\beta}}_\lambda\|_1 = t$, and the solution is then given by the corresponding $\hat{\boldsymbol{\beta}}_\lambda$. In either case, this λ is also the λ which achieves the maximum in Eq. (1.27), and which solves the Karush-Kuhn-Tucker conditions in Theorem 2.

Let us derive the stationarity condition (Eq. (1.12) in Section 1.1.4) of the Karush-Kuhn-Tucker equations, specifically for the LASSO. As we saw, along with complementary slackness (either $\lambda = 0$ or $\|\boldsymbol{\beta}\|_1 = t$) and feasibility ($\lambda \geq 0$ and $\|\boldsymbol{\beta}\|_1 \leq t$), this condition fully characterizes the optimality of our solution.

For the LASSO, the Lagrangian is given by

$$\frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda(\|\boldsymbol{\beta}\|_1 - t).$$

The stationarity condition says that the subgradient with respect to $\boldsymbol{\beta}$ of this Lagrangian must contain the origin, that is, we need that:

$$\mathbf{0} \in -\boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \partial \|\boldsymbol{\beta}\|_1. \tag{1.29}$$

It can be shown that [20, §3.1.5]

$$\partial \|\boldsymbol{\beta}\|_1 = \mathrm{sign}(\beta_1) \times \cdots \times \mathrm{sign}(\beta_p) \tag{1.30}$$

where

$$\text{sign}(\beta_j) \coloneqq \begin{cases} \{-1\} & \text{if } \beta_j < 0 \\ [-1,1] & \text{if } \beta_j = 0 \\ \{1\} & \text{if } \beta_j > 0. \end{cases} \tag{1.31}$$

Therefore, we can write Eq. (1.29) in the following way

$$\boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) = \lambda \boldsymbol{s} \tag{1.32}$$

where $\boldsymbol{s} = (s_1, s_2, \ldots, s_p)$ are auxiliary variables subject to the constraint $s_j \in \text{sign}(\beta_j)$.

When the columns of $\boldsymbol{X}$ are orthogonal (this holds for instance when there is only one predictor) and are standardized such that $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}$, the solution to this system can be expressed as a thresholded version of the ordinary least squares [15]:

$$\hat{\beta}_{\lambda j} = S_\lambda(\hat{\beta}_j^{\text{OLS}}) \tag{1.33}$$

with *soft-thresholding operator* (see Fig. 1.3)

$$S_\lambda(\beta_j) \coloneqq \text{sign}(\beta_j)\max\{0, |\beta_j| - \lambda\}. \tag{1.34}$$

Otherwise, the solution can still be expressed through an iterative execution of soft-thresholding operations [15].

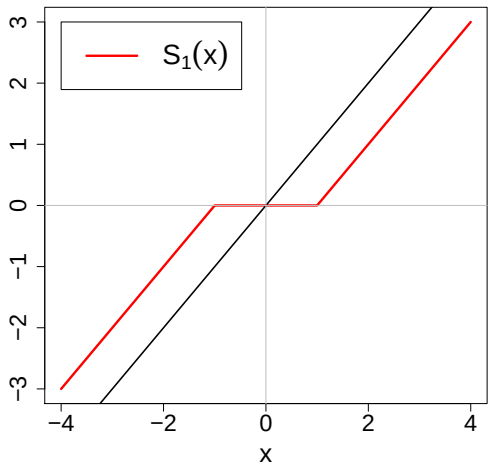

Fig. 1.3 Soft-thresholding function $S_\lambda(x)$ for $\lambda = 1$.

The contour lines in Fig. 1.4 illustrate why and how the LASSO works. The contours refer to the ordinary least squares problem, and the diamond

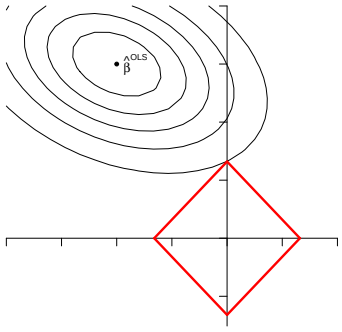

Fig. 1.4 Relationship between the OLS estimate and the l_1 constraint imposed by the LASSO (red); adapted from [16].

corresponds to the constraint $\|\boldsymbol{\beta}\|_1 = t$. Remember that $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = \hat{\boldsymbol{\beta}}_0$, so the figure depicts the case where $\|\hat{\boldsymbol{\beta}}_0\| > t$. We want the point on the diamond closest to the ordinary least squares. This is likely to lie on the axes, hence setting smaller parameters to 0.

1.3.2 Cross-Validation

Cross-validation is a commonly used method to identify the optimal value of a tuning parameter, which is in our case the penalty parameter λ. It is based on minimizing an estimate of the prediction error. In cross-validation, we use one part of the data to fit the LASSO model, and the other part of the data to validate it [16].

We fix initially a dense grid of values of λ, that is λ is discretized with small step-sizes over a suitable range which reflects the scope of the regularization trade-off that we are willing to consider. The dataset is then divided into K equally sized partitions. We assume for simplicity that K is a divisor of n so that each partition contains n/K elements. For each fixed value of λ of the grid, and the k'th partition, $k = 1, \ldots, K$, we fit the LASSO model using the remaining $K - 1$ parts and calculate the prediction error of the fitted model. Specifically, denote $\hat{\boldsymbol{\beta}}_\lambda^{-k}$ the parameter vector obtained under a penalty of λ when omitting the k'th partition, so that $\boldsymbol{x}_i^T \hat{\boldsymbol{\beta}}_\lambda^{-k}$ is the corresponding fitted model under predictor $\boldsymbol{x}_i$. Then the prediction error for the k'th partition is

$$P_k(\lambda) = \frac{K}{n} \sum_{i=1}^{n/K} L(y_i, \boldsymbol{x}_i^T \hat{\boldsymbol{\beta}}_\lambda^{-k}) \tag{1.35}$$

where, for the linear model (Eq. (1.3)), the loss function L is just the squared error. We repeat this step for every $k = 1, 2, \cdots, K$ and combine the values of $P_k(\lambda)$ to find the average prediction error, $P(\lambda) = K^{-1} \sum_{k=1}^{K} P_k(\lambda)$. This is then repeated for every value of λ in the grid, and we choose the value of λ which minimizes $P(\lambda)$ [15].

For smaller values of λ, the LASSO estimators contain more predictors which may lead to an over-fitted model. However, for larger values of λ, the model has fewer predictors leading to sparsity and producing a more easily interpretable model.

To avoid misunderstandings, it is noted that the problem of finding the optimal λ (in the sense of minimal prediction error), as discussed in this subsection, is very different from, and entirely unrelated to, the problem of maximizing over λ as in for instance in (Eq. (1.27)). The latter is a purely formal operation which ensures mathematical equivalence of the two dual versions of the LASSO optimization problem, and does not imply any statement on the best choice of λ.

Example: Gaia dataset

Figure 1.5 represents the cross-validation curve for the Gaia dataset. Here we have taken normalized data to get rid of scalability. The graph is consistent with the property of cross-validation i.e. we can see that for smaller values of λ the number of predictors is higher and for larger values of λ the number of predictors gets reduced. Here, $\log(\lambda)$ is used as the tuning parameter, increased values of which lead to reduced numbers of included variables (note that log denotes the natural logarithm throughout this chapter). From the cross-validation curve we get the value of $\log(\lambda)$ to be approximately 0.775 (shown by the solid vertical line) and hence the prediction error of the LASSO-fitted model is minimal at $\lambda \approx 2.17$. We use this value to estimate the coefficients of the parameters. Note that the plot for this dataset is somewhat unusual, as the minimum falls close to the boundary (solid vertical line); compare further with Fig. 1.11 for a more typical appearance.

Figure 1.6 shows the coefficient path of the parameters, i.e. the change in coefficients of the predictors as a function of λ. The black vertical line denotes the value of $\log(\lambda)$ for which the prediction error is minimal. For this particular value of λ we see that there are only 11 non-zero parameters and others are shrunk towards zero.

For the cross-validation method for LASSO, we have used the **glmnet** [11] package in R. It is noted at this occasion that this software by default also draws a second vertical line in the cross-validation plot (which is dotted in Fig. 1.5), which indicates the largest value of $\log(\lambda)$ which is less than one

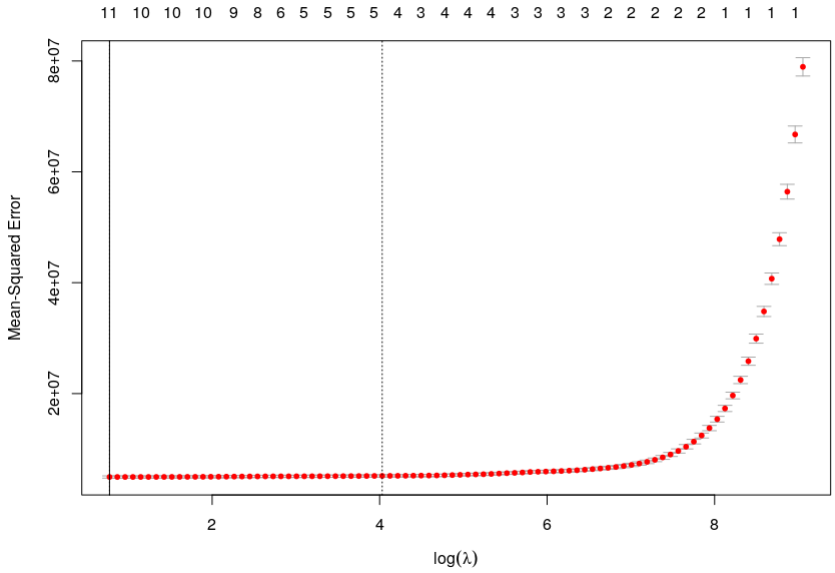

Fig. 1.5 Cross-validation curve for the Gaia dataset, with the number of selected predictor variables as a function of $\log(\lambda)$ given on top of the plot.

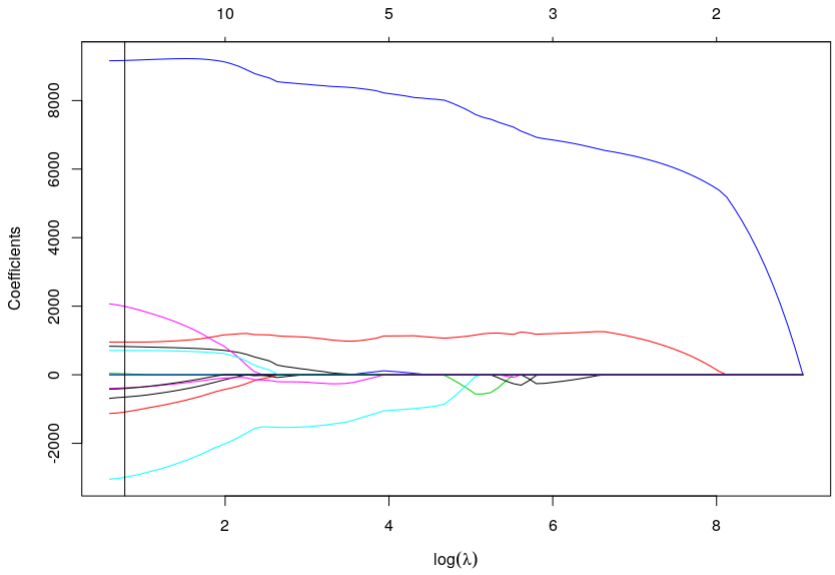

Fig. 1.6 Coefficient path of the parameters for the Gaia dataset.

standard error (calculated for each λ from the $P_k(\lambda)$, $k = 1, \ldots, K$) away from the minimum [15]. Arguably this gives an even sparser solution which is statistically not distinguishable from the one obtained under the minimum. We do not follow this line of reasoning in this exposition, and work with the estimator under the 'optimal' λ at all occasions.

1.4 Uncertainty Quantification

1.4.1 Refit-LASSO

The 'refit'-LASSO is one of the possible ways to quantify system uncertainty of a LASSO fitted model. The simple idea is to use the 'important' (non-zero) variables selected by the LASSO procedure in a subsequent ordinary least squares fit.

We implement a slight modification of this idea. We carry out the entire cross-validation procedure multiple times with random partitions, which gives us different optimized λ for each run, producing an ensemble of possible estimates of $\boldsymbol{\beta}$. We then let the ensemble vote on the inclusion of the variables into the model. We will consider variables as important if they have not been shrunk to 0 for a pre-defined proportion of the runs. Then we apply an ordinary least squares fit on the important variables to get the refit-LASSO estimates. Standard errors of the j'th parameter estimate, $\hat{\beta}_j$, are then obtained as $s\sqrt{(\boldsymbol{X}^T\boldsymbol{X})^{-1}_j}$, where the suffix j indicates the j'th diagonal element taken after application of the inverse, and s^2 denotes the unbiased estimator of σ^2.

Example: Gaia dataset

We applied the refit-LASSO on the Gaia dataset. We have taken 100 simulation runs for the selection of important variables. The result is displayed in Fig. 1.7. We set the desired proportion of inclusion at 50% as indicated by a horizontal line. Then we have applied OLS fit on the important variables; in Table 1.1 we show the standard error of our prediction with its 't-value' and corresponding probability. We also give a comparison between the refit-LASSO estimates and the original cross-validated LASSO estimates in the last two columns.

We notice from the Fig. 1.7 that the third variable appeared to be important in several runs. However, it is non-important in most of the runs.

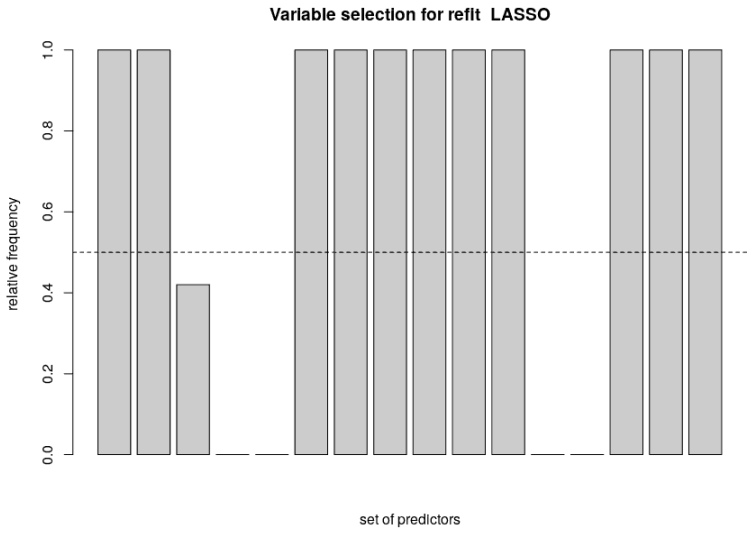

Fig. 1.7 Relative frequency of occurrence of variables, for refit-LASSO applied on the Gaia dataset.

Predictors	Estimate	Std. Error	t value	Pr(>|t|)	Original	Difference
band1	841.04	140.89	5.97	0.00	823.53	17.51
band2	1001.36	298.78	3.35	0.00	954.10	47.26
band6	8960.42	434.64	20.62	0.00	9169.52	-209.09
band7	-3664.57	257.19	-14.25	0.00	-2992.80	-671.77
band8	2842.23	260.48	10.91	0.00	1995.79	846.44
band9	-987.10	201.13	-4.91	0.00	-651.95	-335.15
band10	-1584.91	213.89	-7.41	0.00	-1088.03	-496.88
band11	150.19	175.58	0.86	0.39	28.85	121.33
band14	685.64	204.44	3.35	0.00	708.89	-23.25
band15	-588.20	234.04	-2.51	0.01	-381.77	-206.43
band16	-641.26	259.41	-2.47	0.01	-401.16	-240.10

Table 1.1 Summary of refit-LASSO for the Gaia dataset. The column 'Estimate' gives the parameter estimates from the refitted model using the selected variables. 'Original' estimates refer to a (single) initial cross-validated LASSO execution as discussed in Section 1.3.2, and 'Difference' refers to the difference between refit-LASSO and original estimates.

1.4.2 Bootstrap Method

Bootstrap is a general frequentist method to quantify statistical accuracy, where one randomly draws samples from a given training dataset with replacement, the sample size being equal to that of the original training dataset. This is done for B times (often multiples of 1000). Then one fits the model

to each of these B datasets and examines the empirical distributions of the estimated parameters.

Bootstrap for LASSO

For the LASSO estimation methodology as outlined in Section 1.3.1 and Section 1.3.2, the bootstrap technique is applied straightforwardly, but it has to be ensured that the selection of λ through cross-validation is part of the uncertainty being assessed. Specifically, for each sample dataset obtained through the aforementioned bootstrap routine, we perform cross-validation to obtain the minimal prediction error. This gives us a selected value of λ and hence a parameter estimate $\hat{\boldsymbol{\beta}}_\lambda$ for each bootstrap sample. Then, we use these to calculate the bootstrap standard deviations or empirical distributions of the parameters.

Example: Gaia dataset

At first, we get a one time LASSO estimate using the cross-validation method. Then we take 1000 bootstrap replicates of the original Gaia dataset to calculate the bootstrap statistics. In Table 1.2 we display the summary of our bootstrap result. In addition to the bootstrap mean, median and standard deviation, we also calculated the bootstrap bias using the formula

$$\text{Bias} = \text{Initial Estimate} - \text{Bootstrap Mean}$$

In Fig. 1.8, we visualize the bootstrapped distribution of the parameters through box-plots.

Clearly, it can be seen from Table 1.2 and Fig. 1.8 that band3, band4, band5, band12, band13 are the non-important parameters. While the mean for band3 and band13 is not very close to 0, they still act as non-important parameters with median being 0.

Predictors	mean	median	bias	sd	CI-lower	CI-upper
band1	827.98	835.75	9.33	132.04	483.41	1062.81
band2	991.57	986.84	37.78	333.28	350.15	1655.07
band3	10.21	0.00	10.21	64.39	-30.62	182.39
band4	-2.21	0.00	-2.21	27.26	-67.58	46.23
band5	0.25	0.00	0.25	2.50	0.00	0.87
band6	9127.26	9137.34	-49.87	366.25	8421.19	9797.16
band7	-2984.39	-3019.35	-37.65	338.35	-3441.90	-1557.63
band8	1998.24	2059.49	58.57	429.29	0.84	2486.86
band9	-678.97	-692.91	-46.31	188.08	-1013.58	-78.25
band10	-1092.59	-1123.82	-42.44	226.78	-1392.43	-127.41
band11	58.00	21.52	37.91	78.82	-3.37	254.11
band12	-6.95	0.00	-6.95	22.94	-80.57	0.24
band13	22.90	0.00	22.90	32.18	-0.39	102.34
band14	680.22	697.12	-27.19	147.25	215.06	920.35
band15	-400.80	-405.13	-30.81	127.43	-637.49	-139.13
band16	-391.75	-398.42	-8.78	136.66	-638.43	0.00

Table 1.2 Summary of bootstrap estimates for the Gaia dataset. The lower and upper bounds of the 95% confidence intervals for model parameters are obtained as the 2.5% and 97.5% quantiles of the empirical bootstrap distributions.

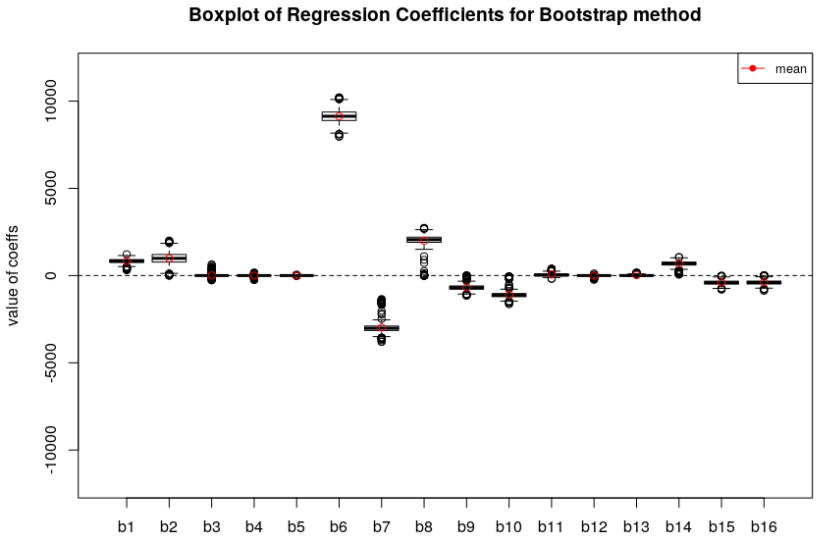

Fig. 1.8 Bootstrappped distribution of the parameters in the Gaia dataset.

1.4.3 Bayesian LASSO

The Bayesian methodology provides a natural way to quantify the model uncertainty in a LASSO-fitted model. To motivate this approach, recall firstly that, under the assumption $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \boldsymbol{I})$, we can write the likelihood of model (1.3) in the following way,

$$\begin{aligned} p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) &\propto e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\epsilon_i^2} \\ &\propto e^{-\frac{1}{2\sigma^2}\|\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2}. \end{aligned} \tag{1.36}$$

Tibshirani [24] suggested using a Laplace prior

$$p(\boldsymbol{\beta}) \propto e^{-\lambda\|\boldsymbol{\beta}\|_1} \tag{1.37}$$

for the model parameters, yielding the following posterior,

$$\begin{aligned} p(\boldsymbol{\beta} \mid \boldsymbol{X}, \boldsymbol{Y}) &\propto p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) \times p(\boldsymbol{\beta}) \\ &\propto e^{-\left(\frac{1}{2\sigma^2}\|\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2+\lambda\|\boldsymbol{\beta}\|_1\right)} \end{aligned} \tag{1.38}$$

It is a well-established result that the mode of (1.38), that is the posterior mode of $\boldsymbol{\beta}$ under Laplace priors, corresponds just to the frequentist LASSO estimate [18, 22, 24]. Draws from this posterior are not necessarily sparse, but still can be used to assess uncertainty of model parameters [15].

The Bayesian LASSO has been implemented in several different facets, which differ essentially in the way that sparsity is induced, and in the way that the regularization parameter is handled. In 2008, Park and Casella [22] proposed a hierarchical mixture model for parameter estimation:

$$\begin{aligned} \boldsymbol{Y}|\mu, \boldsymbol{X}\boldsymbol{\beta}, \sigma^2 &\sim N_n(\mu\mathbf{1}_n + \boldsymbol{X}\boldsymbol{\beta}, \sigma^2 I_n), \\ \boldsymbol{\beta}|\sigma^2, \tau_1^2, \cdots, \tau_p^2 &\sim N_p(\mathbf{0}_p, \sigma^2 \boldsymbol{D}_\tau) \\ \boldsymbol{D}_\tau &= \operatorname{diag}\left(\tau_1^2, \cdots, \tau_p^2\right), \\ \sigma^2, \tau_1^2, \cdots, \tau_p^2 &\sim \pi(\sigma^2)d\sigma^2 \prod_{j=1}^{p} \frac{\lambda^2}{2} e^{-\lambda^2\tau_j^2/2} d\tau_j^2, \\ \sigma^2, \tau_1^2, \cdots, \tau_p^2 &> 0. \end{aligned} \tag{1.39}$$

After marginalizing over $\tau_1^2, \cdots, \tau_p^2$ we get the conditional prior on $\boldsymbol{\beta}$ of the following form

$$\pi(\boldsymbol{\beta}|\sigma^2) = \prod_{j=1}^{p} \frac{\lambda}{2\sigma} e^{-\lambda|\beta_j|\sigma}. \tag{1.40}$$

For the choice of the LASSO penalty parameter, Park and Casella suggested two different techniques. Firstly, they suggested the possibility of using marginal maximum likelihood estimates for the choice of λ. They considered

a Monte Carlo EM algorithm which, in iteration k, updates the parameter λ using the iterative scheme

$$\lambda_k = \sqrt{\frac{2p}{\sum_{j=1}^{p} E_{\lambda_{k-1}}[\tau_j^2|\boldsymbol{Y}]}}, \tag{1.41}$$

where $\boldsymbol{Y}$ is assumed to be centered, and the conditional expectation is estimated via averages of a Gibbs sample. For $p < n$, the initial value λ_0 was suggested to be

$$\lambda_0 = \frac{p\sqrt{\hat{\sigma}^2_{\text{OLS}}}}{\sum_{j=1}^{p} |\hat{\beta}_j^{\text{OLS}}|},$$

where $\hat{\sigma}^2_{\text{OLS}}$ and $\hat{\beta}_j^{\text{OLS}}$ are ordinary least squares estimates. In another approach, they discussed the possibility of using gamma priors on λ^2:

$$\pi(\lambda^2) = \frac{\delta^r}{\Gamma(r)}(\lambda^2)^{r-1}e^{-\delta\lambda^2}; \quad \lambda^2 > 0\,(r > 0, \delta > 0), \tag{1.42}$$

where r is the shape parameter and δ the rate parameter. Lykou and Ntzoufras [18] used gamma priors for λ, and developed a concept for specification of the hyperparameters based on Bayes factors which evaluate the evidence for inclusion of the respective predictor variables.

Example: Gaia dataset

We obtained the posterior distribution of the parameters for the Gaia dataset using the `blasso` function from the **monomvn** [13] package in R. For the choice of the LASSO penalty parameter λ, we used marginal maximum likelihood estimates, as mentioned earlier. We drew 1000 posterior samples from this distribution, which are displayed in Fig. 1.9.

It can be seen that the output from the Bayesian method is similar to that of the Bootstrap method. For a better comparison between the methods we also show the standard errors for the coefficient estimates of each important variable in Fig. 1.10.

1.5 LASSO for Classification

Recall the linear model in row-wise notation, $y_i = \boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i$, or $E(y_i|\boldsymbol{x}_i) = \boldsymbol{x}_i^T\boldsymbol{\beta}$, which makes the implicit assumption on the distribution of the response variable:

$$y_i \sim N(\boldsymbol{x}_i^T\boldsymbol{\beta}, \sigma^2).$$

However, this assumption is too restrictive for many real data situations.

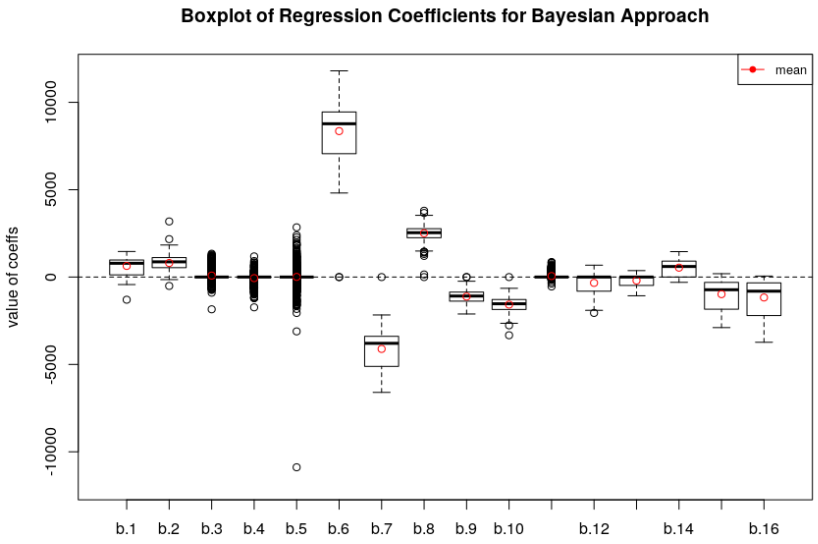

Fig. 1.9 Posterior distribution of the parameters in the Gaia dataset.

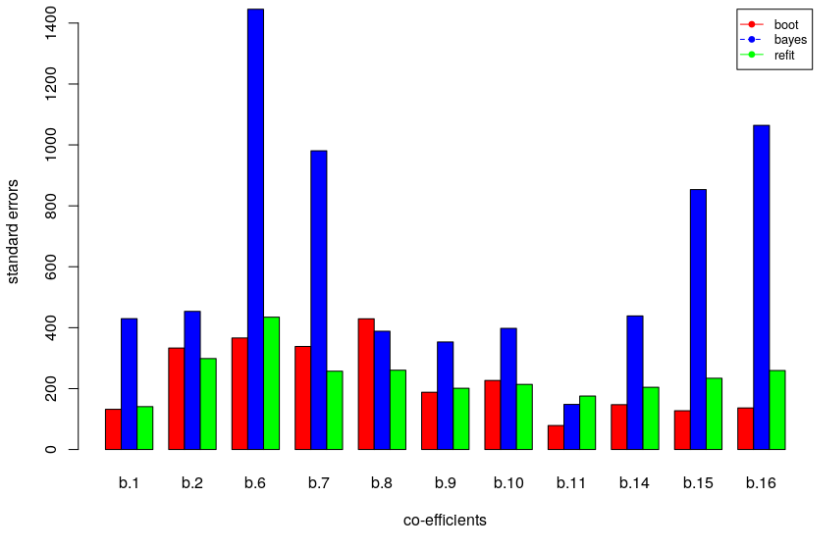

Fig. 1.10 Standard errors of LASSO based parameter estimates for the Gaia dataset, obtained from different methods.

One can use generalized linear models to relax the assumption of normality. We introduce a function g, which acts as a link function such that,

$$g(E(y_i|\boldsymbol{x}_i)) = \boldsymbol{x}_i^T\boldsymbol{\beta}; \tag{1.43}$$

here, y_i can possess any exponential family distribution, such as Poisson, Binomial, or Gamma. Note that if $y_i \in \{0, 1\}$ then,

$$\mu_i \equiv E(y_i|\boldsymbol{x}_i) = P(y_i = 1|\boldsymbol{x}_i) \tag{1.44}$$

hence we can (for our purposes) define:

Definition 2 (Classification). Classification is the process of carrying out a regression problem with 0/1-valued response, and allocating observations to one of the two classes according to the decision rule $\mu_i \geq 0.5$.

1.5.1 Logistic Regression

In logistic regression we start with the logistic model,

$$\log \frac{\mu_i}{1-\mu_i} = \boldsymbol{x}_i^T\boldsymbol{\beta} \tag{1.45}$$

with 'logit' link function $g(\mu_i) = \log \frac{\mu_i}{1-\mu_i}$. An alternative formulation of Eq. (1.45) is:

$$P(y_i = 1|\boldsymbol{x}_i) = h(\boldsymbol{x}_i^T\boldsymbol{\beta}) \tag{1.46}$$

where the *logistic function*

$$h(t) = \frac{\exp(t)}{1+\exp(t)} \tag{1.47}$$

maps the range $(-\infty, \infty)$ to $[-1, 1]$. The parameters in the logistic model are estimated through an iteratively weighted least squares technique known as 'Fisher Scoring', for details of which we refer to [9].

Example 2 (Sonar Dataset). Gorman and Sejnowski used this dataset in their study of the classification of sonar signals using a neural network [12]. The objective of the study was to discriminate between sonar signals bounced off a metal cylinder and a cylindrical rock. Each observation is a set of 60 numbers (serving as predictor variables) in the range 0.0 to 1.0. Each number represents the energy within a particular frequency band, integrated over a certain period of time. The label associated with each response contains the letter 'R' if the object is a rock and 'M' if it is a mine (metal cylinder). There are total of 208 observations in this dataset [17]. Here, due to computational limitations, we have taken the first 48 predictors of the Sonar dataset, and

used the standardized form to handle numerical scaling issues, throughout the examples.

Cross-validation

We apply cross-validation onto the Sonar dataset, and investigate the achieved sparsity as compared to the original model with 48 different predictors. The result of the cross-validation procedure is displayed in Fig. 1.11. The prediction error for this purpose is calculated as in Eq. (1.35) but now the loss function L is given by the deviance (that is, two times the difference of saturated and model log likelihood [9]). From Fig. 1.11 we find that the prediction error is minimal when $\log \lambda = -3.672$, so $\lambda = 0.0254$. Using this value of λ we calculate the coefficients of the parameters. For this particular dataset LASSO eliminates 29 predictors, and reduces the number of retained variables to 19. In Fig. 1.12, we illustrate the coefficient path of the parameters.

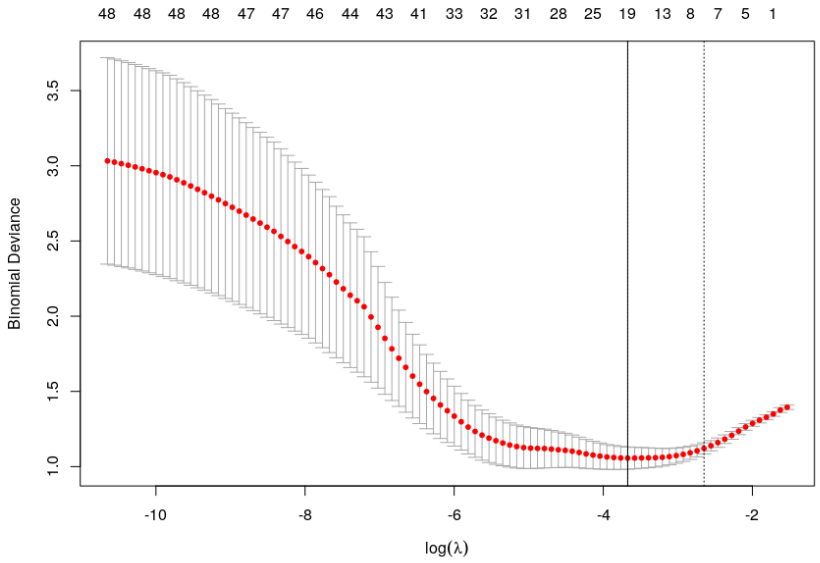

Fig. 1.11 Cross-validation curve for Sonar dataset.

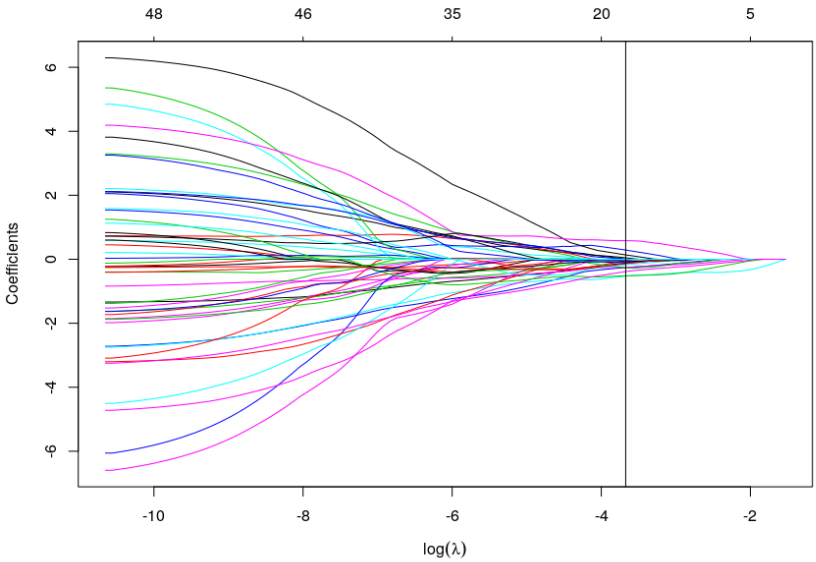

Fig. 1.12 Coefficient path of the parameters for the Sonar dataset.

1.5.2 Uncertainty Quantification

Here, we will discuss uncertainty quantification for the LASSO under the logistic model, by way of application on the Sonar dataset.

Refit-LASSO

We applied the refit-LASSO method on the Sonar dataset. We carried out 100 cross-validation runs with randomized partitions to check the behavior of variable selection. We considered variables as important if they appeared to be non-zero in 50 or more runs. We illustrate the selection of important variable in Fig. 1.13. Then we applied logistic regression on the important variables. We used the `glm` package in R for model-fitting. The corresponding refit-LASSO estimates are given in Table 1.3.

Bootstrap

We applied the bootstrap method on the Sonar dataset with 1000 bootstrap replicates. The procedure works identically as outlined in Section 1.4.2, except that for the Sonar dataset the response variable follows a Bernoulli distribution,

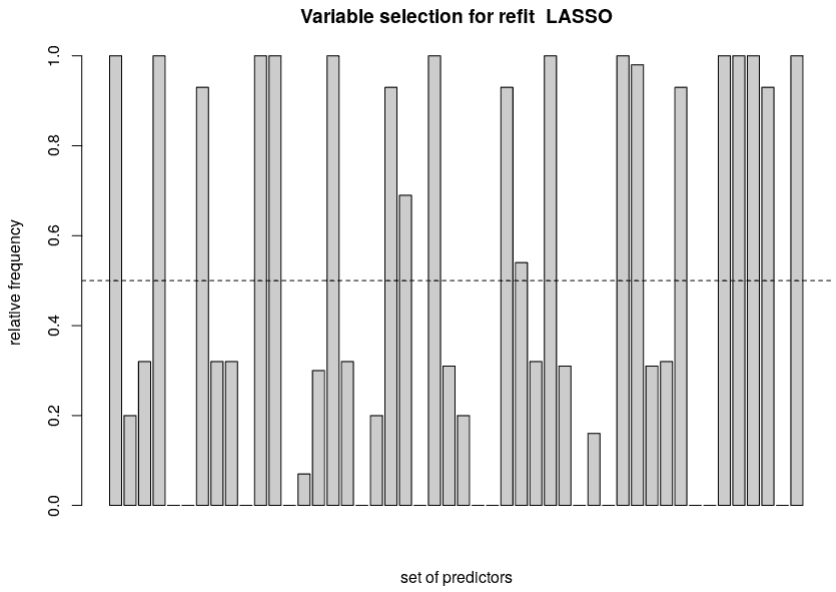

Fig. 1.13 Relative frequency of occurrence of variables, for refit-LASSO applied on the Sonar dataset.

so that for model fitting (and re fitting) we need to work with the binomial response family instead of the normal distribution. The graph in Fig. 1.14 shows the bootstrap distribution of the estimated parameters.

Bayesian approach

We obtained the posterior distribution of the parameters using the `MCMClogit` function from the **MCMCpack** [19] package in R. We took the Laplace priors for parameter estimation. We have taken 100,000 MCMC samples with a thinning interval length of 10 and a Metropolis tuning parameter set at 0.05, yielding 10,000 posterior samples for the assessment of the coefficient distribution. It can be seen that for the Bayesian approach the variability is almost same as that of bootstrap method.

For a better comparison between each parameter estimation method, we have shown the standard errors for the coefficient estimates of each important variable in Fig. 1.16 indexed according to the refit-LASSO method.

Predictors	Estimate	Std. Error	z value	Pr(>|z|)	Original	Difference
(Intercept)	-0.49	0.24	-2.01	0.04	-0.24	-0.24
V1	-0.72	0.33	-2.16	0.03	-0.12	-0.60
V4	-0.91	0.39	-2.32	0.02	-0.26	-0.65
V7	0.67	0.30	2.18	0.03	0.00	0.66
V11	-1.10	0.49	-2.24	0.02	-0.53	-0.57
V12	-0.34	0.41	-0.82	0.41	-0.25	-0.09
V16	1.05	0.34	3.14	0.00	0.29	0.76
V20	-0.88	0.57	-1.54	0.12	-0.03	-0.85
V21	0.25	0.57	0.44	0.66	-0.27	0.52
V23	-0.77	0.33	-2.35	0.02	-0.17	-0.60
V28	0.12	0.41	0.30	0.77	-0.10	0.22
V29	-0.63	0.48	-1.31	0.19	0.00	-0.63
V31	0.87	0.31	2.82	0.00	0.14	0.73
V36	1.04	0.58	1.80	0.07	0.58	0.46
V37	0.27	0.56	0.49	0.62	0.05	0.23
V40	0.34	0.33	1.04	0.30	0.01	0.34
V43	-0.03	0.46	-0.06	0.95	-0.07	0.04
V44	-0.80	0.58	-1.37	0.17	-0.14	-0.66
V45	-0.80	0.79	-1.01	0.31	-0.52	-0.28
V46	-0.06	0.64	-0.09	0.93	-0.02	-0.04
V48	-1.23	0.39	-3.16	0.00	-0.38	-0.85

Table 1.3 Summary of refit-LASSO for the Sonar dataset. The column 'Estimate' gives the parameter estimates obtained after refitting the model using the selected variables. 'Original' estimates refer to a (single) initial cross-validated LASSO execution as discussed in Section 1.5.1, and 'Difference' refers to the difference between refit-LASSO and the original estimates.

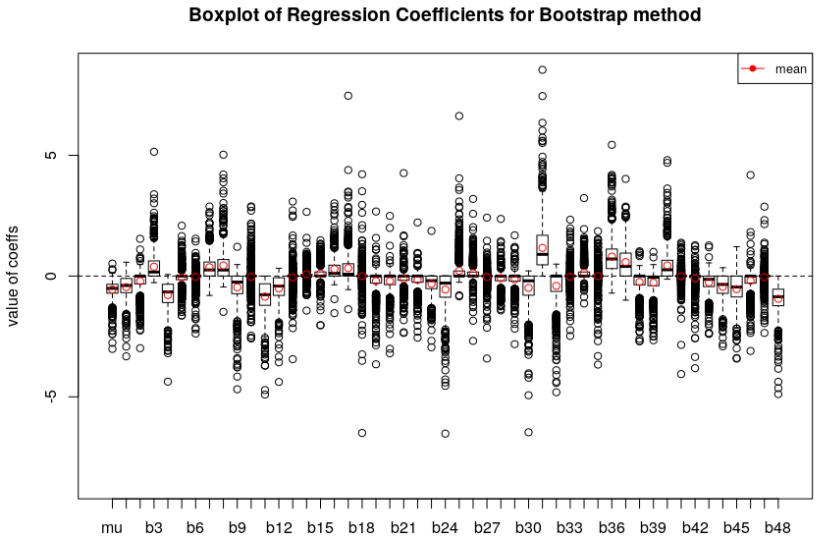

Fig. 1.14 Coefficient distribution of the bootstrap estimates for the Sonar dataset.

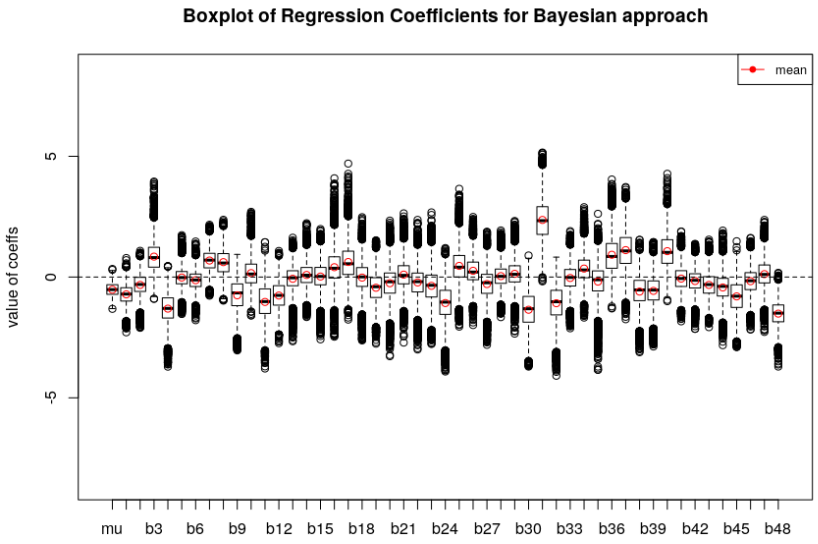

Fig. 1.15 Coefficient distribution of the Bayesian LASSO estimates for the Sonar dataset.

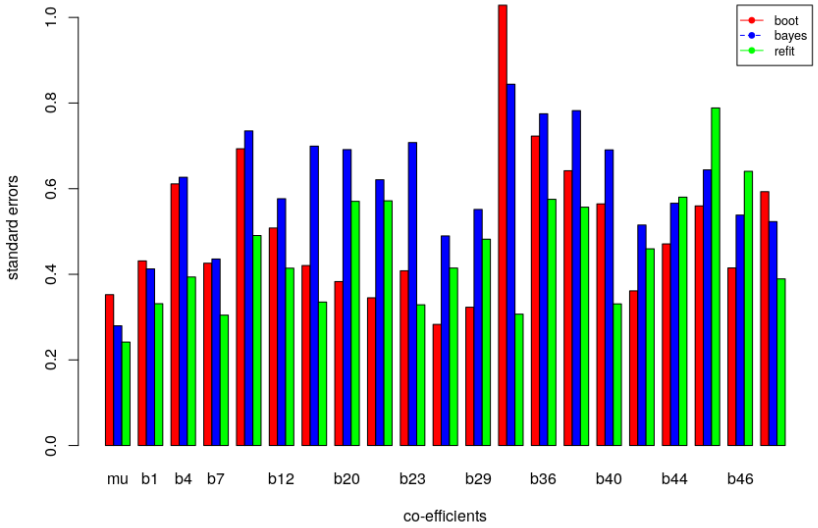

Fig. 1.16 Standard errors from different methods, for the logistic LASSO applied on the Sonar dataset.

1.6 Conclusion

We have presented an overview over commonly used methods for uncertainty quantification in the context of l_1-penalized linear or logistic regression, comprising refit, bootstrap and Bayesian approaches.

We have illustrated these methods in the context of two datasets, both of which have some relevance for aerospace engineering: one dataset relating to the current Gaia space mission, and another dataset involving the analysis of sonar signals.

For both modeling scenarios, we found good agreement of the parameter uncertainties obtained through the different methods. Standard errors of the bootstrap and refit methods agreed particularly closely, noting however the limitation of the latter to quantify uncertainty of inclusion as such. The Bayesian standard errors were of the same magnitude as their frequentist counterparts, however they tended to be larger, and also did show some differences for specific parameters. For the Sonar dataset the refit indicated sparser models than Bayes or bootstrap, which may appear unexpected at first glance, but can be explained by the cut-off threshold of 50% which happened to be just above the relative frequencies of occurrence for many of the variables.

While the discussed uncertainty quantification methods are well established and investigated for the linear model, this is less the case for the logistic model. This is not only reflected in the abundance of relevant literature, but also in the availability of statistical software. Since we had not been able to locate an implementation of the Bayesian logistic LASSO which could handle a model with 60 variables, we had to reduce this dataset from the start to 48 variables. We did so for all methods, to ensure comparability.

References

[1] ESA science & technology: Gaia. `http://sci.esa.int/gaia`. Accessed: 2018-02-06.

[2] C. A. L. Bailer-Jones. The ILIUM forward modelling algorithm for multivariate parameter estimation and its application to derive stellar parameters from Gaia spectrophotometry. *Monthly Notices of the Royal Astronomical Society*, 403(1):96–116, 2010.

[3] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[4] Leo Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384, 1995.

[5] Kiranmoy Das and Marc Sobel. Dirichlet lasso: A Bayesian approach to variable selection. *Statistical Modelling*, 15(3):215–232, 2015.

[6] Norman R. Draper and Harry Smith. *Fitting a Straight Line by Least Squares: Applied Regression Analysis*, pages 15–46. John Wiley & Sons, Inc., 1998.

[7] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *Ann. Statist.*, 32(2):407–499, April 2004.

[8] Jochen Einbeck, Ludger Evers, and Coryn Bailer-Jones. Representing complex data using localized principal components with application to astronomical data. In Alexander N. Gorban, Balázs Kégl, Donald C. Wunsch, and Andrei Y. Zinovyev, editors, *Principal Manifolds for Data Visualization and Dimension Reduction*, pages 178–201, Berlin, Heidelberg, 2008. Springer.

[9] L. Fahrmeir, G. Tutz, and W. Hennevogl. *Multivariate Statistical Modelling Based on Generalized Linear Models*. Springer Series in Statistics. Springer New York, 2001.

[10] Jerome Friedman, Trevor Hastie, Holger Hofling, and Robert Tibshirani. Pathwise coordinate optimization. *Ann. Appl. Stat.*, 1(2):302–332, December 2007.

[11] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[12] R. Paul Gorman and Terrence J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1(1):75–89, 1988.

[13] Robert B. Gramacy. *monomvn: Estimation for Multivariate Normal and Student-t Data with Monotone Missingness*, 2017. R package version 1.9-7.

[14] Ian Hacking. *The Emergence of Probability: A Philosophical Study of Early Ideas about Probability, Induction and Statistical Inference*. Cambridge University Press, 1975.

[15] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press, 2015.

[16] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[17] Friedrich Leisch and Evgenia Dimitriadou. *mlbench: Machine Learning Benchmark Problems*, 2010. R package version 2.1-1.

[18] Anastasia Lykou and Ioannis Ntzoufras. On Bayesian lasso variable selection and the specification of the shrinkage parameter. *Statistics and Computing*, 23(3):361–390, May 2013.

[19] Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park. MCMCpack: Markov chain Monte Carlo in R. *Journal of Statistical Software*, 42(9):22, 2011.

[20] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1st edition, 2014.

[21] Michael R. Osborne, Brett Presnell, and Berwin A. Turlach. On the lasso and its dual. *Journal of Computational and Graphical Statistics*, 9(2):319–337, 2000.

[22] Trevor Park and George Casella. The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

[23] Veronika Ročková and Edward I. George. The spike-and-slab lasso. *Journal of the American Statistical Association*, 113(521):431–444, 2018.

[24] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 58(1):267–288, 1996.

[25] Robert Tibshirani. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(3):273–282, 2011.

[26] Ming Yuan and Yi Lin. On the non-negative garrotte estimator. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2):143–161, 2007.