GRADO EN ECONOMÍA

CURSO ACADÉMICO 2016 / 2017

TRABAJO FIN DE GRADO

REGRESIÓN LOGÍSTICA PENALIZADA

PENALIZED LOGISTIC REGRESSION

AUTOR: VÍCTOR SANCIBRIÁN LANA

DIRECTOR: JUAN MANUEL RODRÍGUEZ POO

FECHA DE ENTREGA: 30 / JUNIO / 2017

Resumen

En este Trabajo de Fin de Grado presentamos un algoritmo para la estimación de modelos de regresión logística penalizados mediante la técnica de regresión en cresta. La disponibilidad de bases de datos masivas ha provocado que muchos modelos de regresión padezcan *sobreajuste*. Así, Tibshirani (1996) introdujo el 'Least Absolute Shrinkage and Selection Operator' (LASSO), un método de estimación que ayudaba a controlar el sobreajuste introduciendo restricciones al tamaño de los coeficientes estimados, de forma que son contraídos hacia cero en función de un parámetro de restricción. Desafortunadamente, es más complicado encontrar trabajos en la literatura donde este tipo de regularización se extienda a modelos econométricos donde la variable dependiente es limitada. Dado que dichos problemas son de interés en economía, este trabajo se centra en la aplicación de estas técnicas de regularización al caso particular de modelos de elección binaria. Para ello, revisamos la literatura existente sobre los Modelos Lineales Generalizados y funciones de verosimilitud penalizadas. Así, se pone de manifiesto que algunos de los resultados nos permiten desarrollar un algoritmo para estimar modelos de regresión logística penalizada. Adicionalmente, realizamos un ejercicio de simulación para comparar los estimadores obtenidos mediante las técnicas de regresión LASSO y regresión en cresta. Además, también estudiamos el rol que juega el parámetro de contracción en la estimación de estos modelos penalizados.

Palabras clave: LASSO, regresión en cresta, regularización, IRLS, regresión logística.

Abstract

In this dissertation we present a new algorithm to estimate penalized (ridge) logistic regression. With the availability of huge data sets, it is now frequent the curse of overfitting in regression models. For the standard linear regression model, in Tibshirani (1996) it was introduced the Least Absolute Shrinkage and Selection Operator (LASSO). This estimation technique guarded against overfitting by introducing a penalty term that somehow shrinkages some subset of parameter estimates towards some zero pre-specified values. Unfortunately, it is much more difficult to find papers where the LASSO approach is extended to regression models where the dependent variable is limited. As in economic analysis is rather frequent to find this type of problem, this dissertation is devoted to the study of how to apply the LASSO approach to the particular case of binary discrete choice models. In order to do so, we first revise the literature of Generalized Linear Models and penalized likelihood approaches. It turns out that some standard results of these fields provide us with tools to develop an algorithm to fit penalized logistic regression models. As an extension we compare through a simulation exercise the results obtained with our estimator against the corresponding LASSO estimators. We study also the crucial role that plays in the fitting of these models the so-called shrinkage parameter.

Keywords: LASSO, ridge regression, penalization, IRLS, logistic regression.

CONTENTS

1. INTRODUCTION

One of the most important results that we learn in any introductory econometrics course is the so-called Gauss-Markov Theorem. In fact, loosely speaking this result give us a nice efficiency outcome for the OLS estimator in the linear regression model against a broad class of estimators (the class of all linear and unbiased estimators) under fairly weak conditions. This important result is probably the reason why in standard econometrics it is rather common to prefer unbiased estimators against biased ones regardless of other important properties. In fact, if we consider as efficiency criteria the Mean Squared Error, in some cases, it can be desirable to incur in a little bias at the gain of considerably reducing the variance.

The idea of this dissertation then is to study estimation techniques that, taking advantage of the bias-variance trade off, enables us to trade a little bias for a substantial decrease in variance, so that the resulting mean squared error is lower. Among other techniques, we have available the so-called *regularization* or *penalization* methods. We focus in two of them, namely *ridge* regression (Hoerl and Kennard 1970) and *LASSO* regression (Tibshirani 1996). We will see how the latter is more intuitive and has some convenient features - such as allowing to perform variable selection -, however it is a non-linear and non-differentiable problem with no closed form.

These methods have been extensively covered in the literature when applied to the classical linear regression model with continuous dependent variable . See for example Tibshirani (2011) for a retrospective overview of these methods in the last years. Although we only focus on the LASSO and ridge alternatives, there have appeared a broad range of generalizations, such as the *grouped LASSO* (Yuan and Lin 2007), the *elastic-net* (Zou and Hastie 2005) or the *adaptive LASSO* (Zou 2006).

However, much of the problems posed in economics involve the estimation of econometric models where the dependent variable is limited. In particular, we turn our attention to the case where the endogenous variable is binary. The *logistic regression* model (Cox 1958) can be used to study economic problems where the outcomes represent success/failure, or the presence/absence of an attribute (smoker/nonsmoker, fail/pass, dead-/alive...). In Section 2 we outline Generalized Linear Models (henceforth GLMs) to lay the foundations for the study of logistic regression in Section 3, as it can be seen as a particular case of GLMs.

In this dissertation we aim at extending these regularization techniques to the study of discrete problems. Section 4 describes ridge regression and extends it to logistic regression. We also propose an algorithm written in *R* for fitting the model. Section 5 extends these results to the LASSO penalty, and tries to generalize them to logistic regression as well. As we will see, LASSO regularization has some computational disadvantages. The literature has proposed several algorithms to fit the model as in Efron et al. (2004), Friedman, Hastie, and Tibshirani (2010) or Lee et al. (2006), although there are still some unresolved issues. Finally, Section 6 contains several Monte Carlo experiments conducted to show how regularization works. In particular, we are concerned with showing how the coefficients in the estimated model behave under a particular constraint, and we also propose a statistic to decide whether regularization may outperform or not another estimation method. The code for these experiments can be found in Appendix A.

2. GENERALIZED LINEAR MODELS

We outline now the characteristics of GLMs, as a comprehensive understanding of these models is needed to master the techniques that we will later introduce. GLMs are a generalization of classical linear models that allow for dependent variables that are not normally distributed. This framework allows to model a potentially nonlinear relationship between covariates and a response variable with linear methods. GLMs were formulated by Nelder and Wedderburn (1972) as a way of unifying previous extensions for multiple errors in the literature. Throughout this section we use McCullagh and Nelder (1989) as the base reference.

To facilitate the transition to GLMs, we can use the classical linear model as the starting point. Suppose that a vector **y** of *n* observations is a realization of a random variable **Y** with independently distributed components with means $\boldsymbol{\mu}$. The latter is specified in terms of *p* unknown parameters and covariates. In the case of the usual linear models, we have

$$E(Y_i) = \sum_{j=1}^{p} x_{ij}\beta_j \qquad for\ i = 1, ..., n \tag{2.1}$$

where x_{ij} refers to the jth covariate for observation i. In matrix notation we may write

$$\boldsymbol{E}(\boldsymbol{Y}) = \boldsymbol{\mu} = \boldsymbol{X\beta} \tag{2.2}$$

where $\boldsymbol{X}$ is the $n \times p$ matrix of dependent variables and $\boldsymbol{\beta}$ is the $p \times 1$ vector of parameters. The random part of the model involves assuming that the errors follow a Normal distribution with constant variance. We may rearrange (2.2) to display a three-part specification:

1. The *random component*.
2. The *systematic component*: explanatory variables produce a linear predictor $\boldsymbol{\eta}$ given by
$$\boldsymbol{\eta} = \sum_{j=1}^{p} x_j\beta_j.$$
3. The *link* between then random and the systematic parts:
$$\boldsymbol{\eta} = g(\boldsymbol{\mu}).$$
where $g(.)$ is called the *link function*.

In this setting, the classical linear models have a Gaussian distribution and identity function link. GLMs allow for two departures: the distribution may come from an element of the exponential family other than the Normal, and the link may be any monotone, differentiable function.

Suppose that the observations y are realizations from a random variable Y whose distribution is a member of the exponential family characterised by the density function

$$f_Y(y; \theta, \phi) = exp\{(y\theta - b(\theta))/a(\phi) + c(y, \phi)\} \tag{2.3}$$

where θ is the *canonical* parameter. Thus, the log-likelihood function is

$$l(\theta, \phi; y) = log\ f_Y(y; \theta, \phi) = \{y\theta - b(\theta)\}/a(\phi) + c(y, \phi). \tag{2.4}$$

We can derive the expression of the mean and variance of Y using the results of Kendall and Stuart (1967, p. 9)

$$E\left(\frac{\partial l}{\partial \theta}\right) = 0 \tag{2.5}$$

$$E\left(\frac{\partial^2 l}{\partial\theta^2}\right)+E\left(\frac{\partial l}{\partial\theta}\right)^2=0. \tag{2.6}$$

From (2.4) we see that

$$\frac{\partial l}{\partial\theta}=\{y-b'(\theta)\}/a(\phi) \tag{2.7}$$

and

$$\frac{\partial^2 l}{\partial\theta^2}=\{-b''(\theta)\}/a(\phi). \tag{2.8}$$

Thus using (2.4) and (2.6) we obtain

$$E\left[\{y-b'(\theta)\}/a(\phi)\right]=\{\mu-b'(\theta)\}/a(\phi)=0, \tag{2.9}$$

so that

$$E(Y)=\mu=b'(\theta). \tag{2.10}$$

Equivalently, from (2.6), (2.7) and (2.8) we have

$$E\left[\{-b''(\theta)\}/a(\phi)\right]+E\left[\{y-b'(\theta)\}/a(\phi)\right]^2=0, \tag{2.11}$$

which becomes

$$\frac{\{-b''(\theta)\}}{a(\phi)}+\frac{var(Y)}{a(\phi)^2}=0 \tag{2.12}$$

so that

$$var(Y)=b''(\theta)a(\phi). \tag{2.13}$$

The function $b''(\theta)$ depends on θ and hence on the mean, and will be referred to as the *variance function* V,

$$V(\mu)=b''(\theta), \tag{2.14}$$

while $a(\phi)$ depends only on the *dispersion parameter* ϕ which is commonly constant.

To select the appropriate link function, one may compare different model fits. Each distribution has the so-called *canonical* link, which occurs when

$$\theta=\eta. \tag{2.15}$$

It has convenient statistical properties, although this does not mean that it is always the best choice (McCullagh and Nelder 1989, p. 32).

2.1. FITTING GENERALIZED LINEAR MODELS

Nelder and Wedderburn (1972) also proposed a Iteratively Reweighted Least Squares (IRLS or IWLS) method for estimation of the parameters β. Here we follow McCullagh and Nelder (1989) again to describe the fitting procedure. The specific numeric algorithm used to accomplish the parameter estimation is the Scoring method.

We need to obtain expressions for the first and second derivatives of the log-likelihood function for a unique observation given by (2.4). Hence by the chain rule

$$\frac{\partial l}{\partial\beta_j}=\frac{\partial l}{\partial\theta}\frac{d\theta}{d\mu}\frac{d\mu}{d\eta}\frac{\partial\eta}{\partial\beta_j}. \tag{2.16}$$

From (2.10) and (2.14) we derive

$$\frac{d\mu}{d\theta}=\frac{db'(\theta)}{d\theta}=b''(\theta)=V(\mu), \tag{2.17}$$

and from the systematic component $\eta = \sum \beta_j x_j$ we obtain

$$\frac{\partial \eta}{\partial \beta_j} = x_j. \tag{2.18}$$

Back to (2.16),

$$\frac{\partial l}{\partial \beta_j} = \frac{(y-\mu)}{a(\phi)} \frac{1}{V} \frac{d\mu}{d\eta} x_j. \tag{2.19}$$

The algorithm, as its very name indicates, depends on the weight W, which is defined by

$$W = V^{-1} \left(\frac{d\mu}{d\eta} \right)^2, \tag{2.20}$$

where V is the variance function defined in (2.14) evaluated at the fitted values $\hat{\mu}$. Taking this into account, and adding the term $\frac{d\mu}{d\eta} \frac{d\mu}{d\eta}^{-1}$ to equation (2.19),

$$\frac{\partial l}{\partial \beta_j} = \frac{(y-\mu)}{a(\phi)} \frac{1}{V} \frac{d\mu}{d\eta}^2 \frac{d\mu}{d\eta}^{-1} x_j = \frac{W}{a(\phi)} (y-\mu) \frac{d\eta}{d\mu} x_j. \tag{2.21}$$

As previously mentioned, the dispersion factor can be assumed to be constant and hence the factor $a(\phi)$ can be omitted. Considering the n observations, equation (2.21) becomes

$$\frac{\partial l}{\partial \beta_j} = \sum \frac{W}{a(\phi)} (y-\mu) \frac{d\eta}{d\mu} x_j. \tag{2.22}$$

The gradient equations in GLMs are non-linear, hence we cannot simply set them equal to zero and obtain a solution. Instead, we use an iterative algorithm, such as the well-known Fisher's Scoring method. Suppose that we want to calculate the Maximum Likelihood Estimator (MLE) θ^* of θ. Using the well-known fact that the gradient evaluated at θ^* equals zero $u(\theta^*) = 0$ and applying a first-order Taylor expansion around the true parameter θ_0 gives

$$u(\theta^*) = u(\theta_0) + u'(\theta_0)(\theta^* - \theta_0) + R_1(\theta^*), \tag{2.23}$$

where the remainder $R_1(\theta^*)$ is negligible and the first derivative of the gradient is the Hessian matrix $H(\theta)$. Rearranging,

$$\theta^* = \theta_0 - \frac{u(\theta_0)}{u'(\theta_0)}. \tag{2.24}$$

Hence the algorithm updates as follows:

$$\theta_{m+1} = \theta_m + \mathcal{J}^{-1}(\theta_m) u(\theta_m) \tag{2.25}$$

where $\mathcal{J}^{-1}(\theta_m)$ stands for the *observed* information matrix, that is the negative of the Hessian. In turn, $u(\theta_m)$ refers to the *score* function, that is the gradient of the log-likelihood function. In practice, the Fisher information (the expected value of the observed information) is used $I(\theta) = E\{\mathcal{J}(\theta)\}$, so the algorithm becomes

$$\theta_{m+1} = \theta_m + I^{-1}(\theta_m) u(\theta_m). \tag{2.26}$$

See Greene (2008) for a more detailed explanation.

Back to our problem, the method uses the vector of first derivatives or gradient,

$$\frac{\partial l}{\partial \boldsymbol{\beta}} = \boldsymbol{u}, \tag{2.27}$$

and the Fisher information

$$-E\left(\frac{\partial^2 l}{\partial\beta_r\partial\beta_s}\right)=\boldsymbol{A}. \tag{2.28}$$

Using the Scoring algorithm and letting $\boldsymbol{b}$ be the actual estimate of $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$ the new estimate, equation (2.26) can be seen as

$$\boldsymbol{b}^*=\boldsymbol{b}-\boldsymbol{H}^{-1}(\boldsymbol{b})u(\boldsymbol{b}), \tag{2.29}$$

where the second part of the right-hand side can be thought of as the adjustment $\boldsymbol{b}^*-\boldsymbol{b}=\delta\boldsymbol{b}$ that we define as the solution of

$$\boldsymbol{A}\delta\boldsymbol{b}=\boldsymbol{u} \tag{2.30}$$

making use of (2.28). If we omit the dispersion factor, the gradient $\boldsymbol{u}$ in (2.22) becomes

$$u_r=\sum W(y-\mu)\frac{d\eta}{d\mu}x_r \tag{2.31}$$

where we again use the notation proposed by McCullagh and Nelder (1989). In addition, (2.28) can be seen as

$$A_{rs}=-E\left(\frac{\partial u_r}{\partial\beta_s}\right)=-E\sum\left[(y-\mu)\frac{\partial}{\partial\beta_s}\left\{W\frac{d\eta}{d\mu}x_r\right\}+W\frac{d\eta}{d\mu}x_r\frac{\partial}{\partial\beta_s}(y-\mu)\right], \tag{2.32}$$

and the first term on the right-hand side cancels out as $E(y)=\mu$. Hence

$$\begin{aligned}A_{rs}&=-E\sum_i\left\{W\frac{d\eta}{d\mu}x_r\frac{\partial}{\partial\beta_s}(y-\mu)\right\}=-\sum_i\left[W\frac{d\eta}{d\mu}x_r\left\{-\frac{\partial\mu}{\partial\beta_s}\right\}\right]\\&=\sum_i Wx_r\frac{d\eta}{d\beta_s}=\sum_i Wx_rx_s.\end{aligned} \tag{2.33}$$

as $\partial y/\partial\beta_s=0$.

From equations (2.29) and (2.30) the new estimate may be written as

$$\begin{aligned}\boldsymbol{b}^*&=\boldsymbol{b}+\delta\boldsymbol{b}\\ \boldsymbol{A}\boldsymbol{b}^*&=\boldsymbol{A}\boldsymbol{b}+\boldsymbol{A}\delta\boldsymbol{b}=\boldsymbol{A}\boldsymbol{b}+\boldsymbol{A}(\boldsymbol{b}^*-\boldsymbol{b})=\boldsymbol{A}\boldsymbol{b}+\boldsymbol{u}.\end{aligned} \tag{2.34}$$

Using (2.33),

$$(\boldsymbol{A}\boldsymbol{b})_r=\sum_s A_{rs}b_s=\sum_s\left(\sum_i Wx_rx_s\right)b_s=\sum Wx_r\eta. \tag{2.35}$$

From (2.31) and (2.34),

$$\begin{aligned}(\boldsymbol{A}\boldsymbol{b}^*)_r=(\boldsymbol{A}\boldsymbol{b})_r+\boldsymbol{u}_r&=\sum Wx_r\eta+\sum W(y-\mu)\frac{\partial\eta}{\partial\mu}x_r\\&=\sum_i Wx_r\left\{\eta+(y-\mu)\partial\eta/\partial\mu\right\}.\end{aligned} \tag{2.36}$$

Given the form of A in (2.33), these equations resemble the solution of weighted least-squares (WLS) problem, with weights given by (2.20) and a *working* dependent variable z given by

$$z=\eta+(y-\mu)\frac{d\eta}{d\mu}. \tag{2.37}$$

Using matrix notation, it can be shown that the form of the estimate $\hat{\boldsymbol{b}}$ mimics that of a WLS problem:

$$\begin{aligned} \boldsymbol{A}\hat{\boldsymbol{b}} &= \boldsymbol{X^T W z} \\ \boldsymbol{X^T W X}\hat{\boldsymbol{b}} &= \boldsymbol{X^T W z} \\ \hat{\boldsymbol{b}} &= (\boldsymbol{X^T W X})^{-1}\boldsymbol{X^T W z}. \end{aligned} \tag{2.38}$$

Notice that the working or *adjusted* dependent variable is simply a linearized link function. Applying a first-order Taylor series about μ

$$g(y) \simeq g(\mu) + (y-\mu)g'(\mu), \tag{2.39}$$

and using the fact that $\eta = g(\mu)$, the right-hand side of the equation is just z. In addition, it can be shown that the variance of the working dependent variable is just W^{-1}, which is easily obtained using (2.20):

$$W^{-1} = \left(\frac{d\eta}{d\mu}\right)^2 V. \tag{2.40}$$

Proof.

$$\begin{aligned} &E(z) = \eta \\ &V(z) = E\left[(z-E(z))^2\right] = E\left[(y-\mu)\frac{d\eta}{d\mu}\right]^2 = E\left[(y-\mu)^2\right]\left(\frac{d\eta}{d\mu}\right)^2 = var(Y)\left(\frac{d\eta}{d\mu}\right)^2 \end{aligned} \tag{2.41}$$

Using (2.13) and (2.14) and ignoring again the constant $a(\phi)$,

$$V(z) = b''\left(\frac{d\eta}{d\mu}\right)^2 = \left(\frac{d\eta}{d\mu}\right)^2 V(\mu) = W^{-1}. \tag{2.42}$$

□

The IRLS process is described in Algorithm 1. At each iteration k we solve a WLS problem, and the procedure is referred to as *iterative* because both z and W depend on the fitted values, and therefore are updated with each new iteration.

Algorithm 1 IRLS

```
Set the initial estimate $\beta_0$;
while k<MaxIterations do
    Compute the linear predictor $\eta = \sum \beta_j x_j$
    Compute the fitted value $\mu = g(\eta)$
    Construct the working dependent variable
        $$z = \eta + (y - \mu)\frac{d\eta}{d\mu}$$
    Calculate the weight
        $$W = V(\mu)^{-1}\left(\frac{d\mu}{d\eta}\right)^2$$
    Regress $z$ on the covariates and weight $W$ to solve the WLS problem so as to
    obtain the new estimate $\beta_{k+1}$
        $$\beta = (X^T W X)^{-1} X^T W z$$
    if the stopping criterion is satisfied then
        Break;
    end if
end while
```

3. LOGISTIC REGRESSION FOR BINARY DATA

Instead of considering the generic model derived in Section 2, let us now focus on a particular case known as the *logistic regression* model. Logistic regression or *logit* is a discrete choice regression model where the response is categorical, that is it can take on one of a limited number of values. Although categorical variables can have more than two possible values (polytomous variables) we will consider here dichotomous variables, which can take on just two different values, say "0" and "1". The pioneer of the model was statistician Cox (1958).

3.1. INTRODUCTION TO LOGISTIC REGRESSION

We start by studying the structure of the data in terms of its probability distribution, and then we discuss the logit transformation and set-up the regression model.

3.1.1. The Binomial Distribution

Consider first the case where the dependent variable y_i is binary, which can only take two values, say one or zero. This variable is a realization of a random variable Y_i that takes the values

$$Y_i = \begin{cases} 1, & \text{with probability } \pi_i \\ 0, & \text{with probability } 1 - \pi_i. \end{cases} \tag{3.1}$$

Then, Y_i follows a *Bernouilli* distribution $Y_i \sim Ber(\pi_i)$ with probability function

$$Pr(Y_i = y_i) = \pi_i^{y_i}(1 - \pi_i)^{1-y_i}. \tag{3.2}$$

The expected value is

$$E(Y_i) = \mu_i = 1\ Pr(Y_i = 1) + 0\ Pr(Y_i = 0) = \pi_i, \tag{3.3}$$

and the variance, which is not constant and depends on the probability of success,

$$var(Y_i) = \sigma_i^2 = E(Y_i^2) - E(Y_i)^2 = \pi_i - \pi_i^2 = \pi_i(1 - \pi_i). \tag{3.4}$$

thus ruling out the possibility of fitting linear models with homoscedastic variance.

We consider now the possibility of extending this characterization. Sometimes, it is more convenient to work with *grouped* data, that is listing the data in terms of *covariate classes* rather than by the N individuals (McCullagh and Nelder 1989, p. 100). Each covariate class m_i is formed by a group of individuals that have identical values of all explanatory variables $x_{i1}, ..., x_{ip}$. This classification is more efficient mainly when the number of covariate vectors is significantly smaller than the number of individuals N. When binary data are grouped, the dependent variables have the form $y_1/m_1, ..., y_n/m_n$, where $0 \leq y_i \leq m_i$ is defined as the number of units having the attribute of interest out of the m_i individuals belonging to that group or class. In addition, the vector $\boldsymbol{m}$ of classes is called the *binomial index* vector.

Now, the responses $y_i = 0, 1, ..., m_i$ are viewed as a realization of a random variable Y_i that follows the binomial distribution with index m_i and parameter π_i, $Y_i \sim B(m_i, \pi_i)$. The probability distribution function is given by

$$Pr(Y_i = y_i) = f_i(y_i) = \binom{m_i}{y_i}\pi^{y_i}(1 - \pi)^{m_i - y_i} \qquad for\ y_i = 0, 1, ..., m_i \tag{3.5}$$

where y_i is the number of successes and $m_i - y_i$ the number of failures and the binomial coefficient represents the number of ways of obtaining them. The expected value and variance of Y_i are given by

$$E(Y_i) = \mu_i = m_i \pi_i \tag{3.6}$$

and

$$var(Y_i) = \sigma_i^2 = m_i \pi_i (1 - \pi_i). \tag{3.7}$$

Using grouped data is somehow more general as it includes individual data as a special case, that is when we have n groups with $m = 1$. In fact, the Bernouilli distribution can be viewed as a degenerated case of the binomial for which $m = 1$. When the outcomes are independent, the two specifications are equivalent and they lead to the same likelihood function. Hence we will be using the binomial distribution to outline the model and later limit ourselves to the $m = 1$ case.

From (3.5) and taking logs,

$$\begin{aligned} log\ f_i(y_i) &= y_i log(\pi_i) + (m_i - y_i) log(1 - \pi_i) + log\binom{m_i}{y_i} \\ &= y_i log\left(\frac{\pi_i}{1 - \pi_i}\right) + m_i log(1 - \pi_i) + log\binom{m_i}{y_i}. \end{aligned} \tag{3.8}$$

Thus the log-likelihood has the general form of (2.4), with canonical link

$$\theta_i = log\left(\frac{\pi_i}{1 - \pi_i}\right), \tag{3.9}$$

which is called the *logit*. Hence the binomial distribution, as noted by McCullagh and Nelder (1989, p. 30), belongs to the exponential family. As a consequence, logistic regression can be characterized following the framework outlined in Section 2.

3.1.2. The Logistic Regression Model

Given its probability distribution, we need to specify a function to model the probabilities π_i in terms of a set of explanatory variables . The *linear probability model*

$$\pi_i = \boldsymbol{x_i'}\beta \tag{3.10}$$

does not guarantee that the responses will fit within the response range $[0, 1]$. An alternative is to model the probabilities in a different way such that this *transformation* can finally be a linear function of $\boldsymbol{x_i}$. Following Rodríguez (2007), This involves computing the *odds*

$$odds_i = \frac{Pr(Y_i = 1)}{Pr(Y_i = 0)} = \frac{\pi_i}{1 - \pi_i}, \tag{3.11}$$

and taking logarithms to obtain the *logit* or log-odds

$$\eta_i = logit(\pi_i) = log\left(\frac{\pi_i}{1 - \pi_i}\right). \tag{3.12}$$

The transformation has removed the range restriction, as the logit maps the interval $[0, 1]$ onto the whole real line. The *antilogit* transformation allows us to go back to probabilities

$$\pi_i = logit^{-1}(\eta_i) = \frac{e^{\eta_i}}{1 + e^{\eta_i}}. \tag{3.13}$$

Suppose that the observed values $y_1, ..., y_n$ are realizations of independent random variables $Y_1, ..., Y_n$ where Y_i has the binomial distribution

$$Y_i \sim B(m_i, \pi_i) \tag{3.14}$$

and the systematic part of the model is specified as a linear function of the predictors and coefficients

$$\eta_i = g(\pi_i) = log\left(\frac{\pi_i}{1-\pi_i}\right) = \sum_j x_{ij}\beta_j \qquad for\ i = 1, ..., n. \tag{3.15}$$

Equation (3.15) defines a GLM with binomial errors and link logit, which is the canonical link as defined in (3.9). Using matrix notation, we can easily solve for the odds

$$log\left(\frac{\pi_i}{1-\pi_i}\right) = \boldsymbol{x_i'}\beta \rightarrow \frac{\pi_i}{1-\pi_i} = exp(\boldsymbol{x_i'}\beta) \tag{3.16}$$

and the probabilities

$$\pi_i = \frac{exp(\boldsymbol{x_i'}\beta)}{1+exp(\boldsymbol{x_i'}\beta)}. \tag{3.17}$$

Interpreting the parameters in (3.15) simply implies analyzing changes in the logit of the probability, while in (3.17) we have a nonlinear function of the β coefficients.

3.2. ESTIMATION

We follow here the procedure outlined in Section 2 to estimate the coefficients in (3.15) by IRLS, as the binomial distribution belongs to the family of exponential distributions.

We have $Y_1, ..., Y_n$ independent random variables with probability density function given by (3.5), thus the joint density is

$$\mathcal{L}(\boldsymbol{\pi}; \boldsymbol{y}) = \prod_{i=1}^{n} f_i(y_i; \pi_i) = \prod_{i=1}^{n}\left[\binom{m_i}{y_i}\pi^{y_i}(1-\pi)^{m_i-y_i}\right] \tag{3.18}$$

and the log-likelihood

$$log\ \mathcal{L}(\boldsymbol{\pi}; \boldsymbol{y}) = l(\boldsymbol{\pi}; \boldsymbol{y}) = \sum_{i=1}^{n} log\ f_i(y_i; \pi_i) = \sum_{i=1}^{n}\left[y_i log(\pi) + (m_i - y_i) log(1-\pi_i)\right], \tag{3.19}$$

where the combinatorial coefficient has been omitted because it is a constant function which does not depend on $\boldsymbol{\pi}$. Equivalently to (3.8), the log-likelihood may be written as

$$l(\boldsymbol{\pi}; \boldsymbol{y}) = \sum_{i=1}^{n}\left[y_i log\left(\frac{\pi_i}{1-\pi_i}\right) + m_i log(1-\pi_i)\right]. \tag{3.20}$$

McCullagh and Nelder (1989) also consider the log-likelihood as a function of the unknown parameters,

$$l(\boldsymbol{\beta}; \boldsymbol{y}) = \sum_i \sum_j y_i x_{ij}\beta_j - \sum_i m_i log\left(1 + exp\sum_j x_{ij}\beta_j\right) \tag{3.21}$$

which we show to be equivalent below.

Proof. From (3.19) and using (3.16) and (3.17)

$$\begin{aligned} l(\boldsymbol{\beta}; \boldsymbol{y}) &= \sum_i \sum_j y_i x_{ij}\beta_j + \sum_i m_i log\left(1 - \frac{exp\sum_j x_{ij}\beta_j}{1+exp\sum_j x_{ij}\beta_j}\right) \\ &= \sum_i \sum_j y_i x_{ij}\beta_j + \sum_i m_i log\left(\frac{1}{1+exp\sum_j x_{ij}\beta_j}\right) \end{aligned} \tag{3.22}$$

and using the property $log(a) = -log(a^{-1})$

$$l(\boldsymbol{\beta};\boldsymbol{y}) = \sum_i \sum_j y_i x_{ij}\beta_j - \sum_i m_i log\left(1 + exp\sum_j x_{ij}\beta_j\right). \tag{3.23}$$

□

The likelihood equations for the parameters β can be obtained using the chain rule:

$$\frac{\partial l}{\partial \beta_r} = \frac{\partial l}{\partial \pi_i}\frac{\partial \pi_i}{\partial \beta_r} = \frac{\partial l}{\partial \pi_i}\frac{d\pi_i}{d\eta_i}\frac{\partial \eta_i}{\partial \beta_r}. \tag{3.24}$$

Taking derivatives of the log-likelihood function, as given in (3.19), gives

$$\frac{\partial l}{\partial \pi_i} = y_i\,\frac{d\;log\left(\frac{\pi_i}{1-\pi_i}\right)}{d\pi_i} + m_i\,\frac{d\;log(1-\pi_i)}{d\pi_i} = \frac{y_i}{\pi_i(1-\pi_i)} - \frac{m_i}{(1-\pi_i)} = \frac{y_i - mi_i\pi_i}{\pi_i(1-\pi_i)}, \tag{3.25}$$

and using $\partial\eta_i/\partial\beta_r = x_{ir}$ we obtain

$$\frac{\partial l}{\partial \beta_r} = \sum_i \frac{y_i - mi_i\pi_i}{\pi_i(1-\pi_i)}\frac{d\pi_i}{d\eta_i}x_{ir}. \tag{3.26}$$

Now, by means of (3.17) and $E(Y_i) = \mu_i = m_i\pi_i$,

$$\frac{\partial l}{\partial \beta_r} = \sum_i \frac{y_i - m_i\,e^{\eta_i}/(1+e^{\eta_i})}{e^{\eta_i}/(1+e^{\eta_i})^2}\frac{e^{\eta_i}}{(1+e^{\eta_i})^2}x_{ir} = \sum_i (y_i - m_i\pi_i)x_{ir} = \sum_i (y_i - \mu_i)x_{ir}. \tag{3.27}$$

Hence, in matrix notation, the score reduces to

$$\partial l/\partial\boldsymbol{\beta} = \boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{\mu}). \tag{3.28}$$

The observed information matrix can be again be approximated by the Fisher information,

$$\begin{aligned} -E\left(\frac{\partial^2 l}{\partial\beta_r\partial\beta_s}\right) &= \sum_i \frac{m_i\left[\pi_i(1-\pi_i)\right]}{\left[\pi_i(1-\pi_i)\right]^2}\frac{\partial\pi_i}{\partial\beta_r}\frac{\partial\pi_i}{\partial\beta_s} = \sum_i \frac{m_i}{\pi_i(1-\pi_i)}\frac{\partial\pi_i}{\partial\beta_r}\frac{\partial\pi_i}{\partial\beta_s} \\ &= \sum_i m_i\,\frac{\left(d\pi_i/d\eta_i\right)^2}{\pi_i(1-\pi_i)}x_{ir}x_{is} = \left\{\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\right\}_{rs}, \end{aligned} \tag{3.29}$$

where we have made use of $d\pi_i/d\eta_i\;x_{ir} = \partial\pi_i/\partial\beta_r$ and $\boldsymbol{W}$ is a diagonal weight matrix

$$\boldsymbol{W} = diag\left\{m_i\,\frac{\left(d\pi_i/d\eta_i\right)^2}{\pi_i(1-\pi_i)}\right\}. \tag{3.30}$$

Using (3.17) and $d\pi_i/\eta_i = e^{\eta_i}/(1+e^{\eta_i})^2$, $\boldsymbol{W}$ reduces to

$$\boldsymbol{W} = diag\left\{m_i\,\frac{e^{\eta_i}}{(1+e^{\eta_i})^2}\right\} = diag\left\{m_i\pi_i(1-\pi_i)\right\}. \tag{3.31}$$

It can be shown that this diagonal matrix of weights is a particular case of the more general weight given by equation (2.20).

Proof. The generic weight W is defined as

$$W = V^{-1}\left(\frac{d\mu}{d\eta}\right)^2. \tag{3.32}$$

Using the expressions for the mean and the variance of Y_i given in equations (3.6) and (3.7),

$$\begin{aligned} W &= \frac{1}{m_i\pi_i(1-\pi_i)} m_i^2 \left(\frac{d\pi_i}{d\eta_i}\right)^2 = \frac{m_i^2}{m_i\, e^{\eta_i}/(1+e^{\eta_i})^2}\left[\frac{e^{\eta_i}}{(1+e^{\eta_i})^2}\right]^2 \\ &= m_i \frac{e^{\eta_i}}{(1+e^{\eta_i})^2} = m_i\pi_i(1-\pi_i), \end{aligned} \tag{3.33}$$

which again matches $V(Y_i)$. □

At this point we could estimate the parameters using a method such as the Scoring or the Newton-Raphson algorithm. The procedure is equivalent to IRLS. Given a current estimate $\boldsymbol{\beta}$, the working dependent variables has the form given in (2.37), that is

$$z_i = \hat{\eta}_i + \frac{y_i - m_i\hat{\pi}_i}{m_i}\frac{d\eta_i}{d\pi_i}, \tag{3.34}$$

which by the fact that $\hat{\mu}_i = m_i\hat{\pi}_i$ and $d\eta_i/d\pi_i = 1/\left[\pi_i(1-\pi_i)\right]$ reduces to

$$z_i = \hat{\eta}_i + \frac{y_i - \hat{\mu}_i}{m_i\pi_i(1-\pi_i)}, \tag{3.35}$$

where the *hats* highlight that the current estimate is used to compute the lineal predictor η and the fitted values μ. In addition, through an equivalent derivation to that of (2.42), it can be shown that the variance of z is just W^{-1}.

Proof.

$$\begin{aligned} E(z) &= \hat{\eta} + \frac{E(y)-\hat{\mu}}{m\pi(1-\pi)} = \hat{\eta} \\ V(z) &= E\left[(z-E(z))^2\right] = E\left[\hat{\eta} + \frac{y-\hat{\mu}}{m\pi(1-\pi)} - \hat{\eta}\right]^2 = \frac{1}{\left[m\pi(1-\pi)\right]^2}E\left[(y-\hat{\mu})^2\right] \\ &= \frac{var(Y)}{\left[var(Y)\right]^2} = \frac{1}{m\pi(1-\pi)} = W^{-1}, \end{aligned} \tag{3.36}$$

where the subscripts have been omitted for simplicity and we have used (3.6) and (3.7). □

The algorithm requires regressing the adjusted dependent variable on the covariates and solving a WLS problem at each iteration, thus the revised estimate for $\boldsymbol{\beta}$ has the form given in (2.38), which we prove below.

Proof. The Fisher's Scoring method, according to (2.26), updates as follows:

$$\boldsymbol{\beta_1} = \boldsymbol{\beta_0} + \boldsymbol{I}^{-1}(\boldsymbol{\beta_0})\boldsymbol{u}(\boldsymbol{\beta_0}) \tag{3.37}$$

where $\boldsymbol{\beta_1}$ and $\boldsymbol{\beta_0}$ represent the new and old estimates respectively, and the Fisher information matrix $\boldsymbol{I}$ and the gradient $\boldsymbol{u}$ are given by (3.28) and (3.29) respectively. Incorporating these expressions,

$$\boldsymbol{\beta_1} = \boldsymbol{\beta_0} + (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T(\boldsymbol{y}-\boldsymbol{\mu}), \tag{3.38}$$

and pre-multiplying $\boldsymbol{\beta_0}$ by $(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})$ and adding $\boldsymbol{W}\boldsymbol{W}^{-1}$,

$$\begin{aligned}\boldsymbol{\beta_1} &= (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})\boldsymbol{\beta_0} + (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{W}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}) \\ &= (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{W}\left\{\boldsymbol{X}\boldsymbol{\beta_0} + \boldsymbol{W}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})\right\} \\ &= (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{z}, \qquad (3.39)\end{aligned}$$

where $\boldsymbol{W} = diag\left\{m_i\pi_i(1-\pi_i)\right\}$ and $\boldsymbol{z} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{W}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})$. □

As noted in McCullagh and Nelder (1989, p. 43), for exponential families where we use the canonical link, the expected and actual values of the Hessian coincide, so the Fisher's Scoring method and the Newton-Raphson method coincide as well.

The iterative process is described in Algorithm 2. As it will be used later, we will make now the simplification $m = 1$, that is we work with individual data or, equivalently, the binomial distribution reduces to the Bernouilli distribution.

Algorithm 2 IRLS for Logistic Regression

```
Set the initial estimate for the parameters, say [1]
    $$\boldsymbol{\beta} = 0 \qquad and \qquad \beta_0 = log\left(\frac{\bar{y}}{1-\bar{y}}\right)$$
while k<MaxIterations do
    Compute the linear predictor
        $$\eta = \beta_o + \sum \beta_j x_j$$
    Compute the fitted value
        $$\mu \equiv \pi = \frac{exp(\eta)}{1+exp(\eta)}$$
    Construct the working dependent variable
        $$z = \eta + \frac{(y-\mu)}{w} \qquad where\ w \equiv \pi(1-\pi)$$
    Calculate the weight matrix
        $$W = diag\left\{w\right\}$$
    Regress $z$ on the covariates and weight $W$ to solve the WLS problem so as to obtain the new estimate $\beta_{k+1}$
        $$\beta = (X^TWX)^{-1}X^TWz$$
    if the stopping criterion is satisfied then
        Break;
    end if
end while
```

[1] When all the slopes are set to zero, the odds in the intercept-only model are given by $odds_i = exp(\beta_0) = \bar{y}/(1-\bar{y})$ thus an estimate for β_0 is obtained taking natural logarithms. Also, the subscripts i and the *hats* indicating estimates of the true parameters have been omitted for simplicity.

4. PENALIZED LOGISTIC REGRESSION: A RIDGE PARAMETER

In the fields of statistics and econometrics, penalization or regularization refers to a technique that introduces a penalty term to the objective function. It is used to tackle some situations where the usual OLS estimation may be unsatisfactory or to improve the estimates as judged by different quality measures.

Throughout this section, we first motivate the use of regularization techniques and outline one of the simplest ways to penalize the parameters, known as Ridge Regression. Then we extend this penalty to Logistic Regression, and propose an IRLS algorithm to fit the model parameters.

4.1. INTRODUCTION TO REGULARIZATION

4.1.1. Motivation and the bias-variance trade-off

The Gauss-Markov Theorem states that in the classical linear regression model the OLS estimator is the one giving the minimum variance among the class of all linear unbiased estimators. This important result is probably the reason why it is rather common for practitioners to prefer unbiased estimators against biased ones regardless of other important properties, such as low variance.

Even though we know that unbiasedness and low variance are two convenient features of an estimator, sometimes they can conflict with each other (Dougherty 2007). Consider as an example what happens in Figure 4.1. While estimator B is clearly biased, it has a smaller variance than A, which is unbiased.

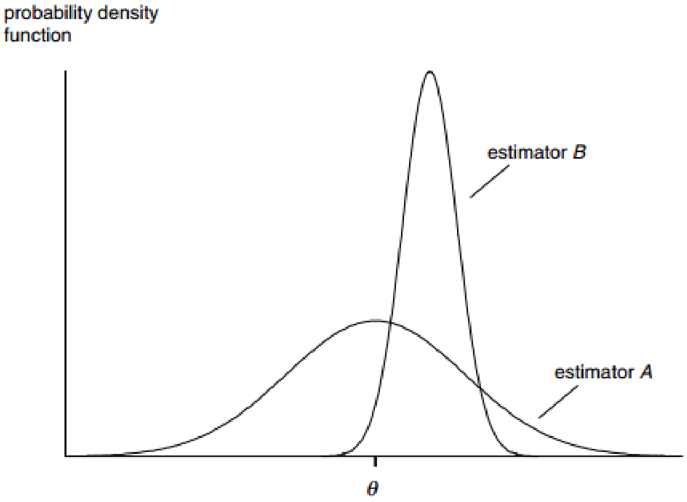

Figure 4.1: The bias-variance tradeoff
Source: Dougherty (2007, p. 26)

Which estimator is *better* can be determined using a function that weights the deviations of the estimated parameter from its true value, such as *risk functions*, and choosing the estimator that yields the smallest expected loss. One common *loss function* is the Mean Squared Error (henceforth MSE), that is the expected value of the quadratic loss. It can be shown that the MSE of an estimator $\hat{\theta}$ can be decomposed as the sum of the variance and the squared bias of the estimator,

$$MSE(\hat{\theta}) = E\left[(\hat{\theta} - \theta)^2\right] = var(\hat{\theta}) + bias(\hat{\theta}, \theta)^2. \tag{4.1}$$

Proof. When $\hat{\theta}$ is a scalar and an estimator of θ, and adding the term $E(\hat{\theta}) - E(\hat{\theta})$,

$$\begin{aligned} MSE(\hat{\theta}) &= E\left[(\hat{\theta} - E(\hat{\theta}) + E(\hat{\theta}) - \theta)^2\right] \\ &= E\left[(\hat{\theta} - E(\hat{\theta}))^2\right] + E\left[(E(\hat{\theta}) - \theta)^2\right] + E\left[2(\hat{\theta} - E(\hat{\theta}))(E(\hat{\theta}) - \theta)\right] \\ &= E\left[(\hat{\theta} - E(\hat{\theta}))^2\right] + (E(\hat{\theta}) - \theta)^2 + 2\left[E(\hat{\theta}) - E(\hat{\theta})\right](E(\hat{\theta}) - \theta) \\ &= E\left[(\hat{\theta} - E(\hat{\theta}))^2\right] + (E(\hat{\theta}) - \theta)^2 \\ &= var(\hat{\theta}) + bias(\hat{\theta}, \theta)^2. \end{aligned} \tag{4.2}$$

□

It follows that the OLS estimator is the minimum MSE estimator among the class of linear unbiased estimators. However, it is possible to find a biased estimator with smaller MSE, which would allow for a little bias in exchange for a larger reduction in variance (Hastie, Tibshirani, and Friedman 2008, p. 52).

This is precisely the basic intuition behind the idea of regularization: under some circumstances, the OLS estimator is susceptible to having very high variance, thus introducing a little bias may allow us to substantially reduce the variance, and possibly lead to a smaller MSE. Regularization is achieved by introducing a penalty term in the objective function which penalizes how large the coefficients can grow, therefore introducing bias but controlling the variance. The OLS estimator may have large variability when, for example, there is multicolinearity (a high correlation among the explanatory variables). The variance of the OLS estimator has the well-known form (Wooldridge 2009)

$$V(\hat{\beta}_j) = \frac{\sigma^2}{SST_j(1 - R_j^2)}, \tag{4.3}$$

where SST_j refers to the Total Sum of Squares and represents the total sample variation in the predictors, and R_j^2 is the R-squared from regressing the covariate x_j on all other predictors included in the model. Therefore, if there is perfect multicollinearity, the variance is infinite. When the explanatory variables are highly correlated, several problems arise (Greene 2008):

1. Unstable estimators, very sensitive to small changes in the data.
2. High standard errors, leading to confusing inference conclusions.
3. Not acceptable magnitudes or wrong signs.

Another reason why the variance may be large is due to a large number of predictors relative to the sample size. The variability in the explanatory variables is greater for large samples, which in turn helps to control the variance. To sum up, occasionally a penalized estimate that helps to control a large variance may be a more convenient choice.

The usual motivation to conduct regularization is to reduce overfitting, which happens for example if a model has an excessive number of parameters compared to the sample size. Overfitting implies that the model captures part of the random noise of the training set, thus generalizing poorly to the testing set. Regularization improves prediction performance, which is intimately related to MSE (Hastie, Tibshirani, and Friedman 2008, p. 52). Consider the model

$$y_i = f(x_i) + u_i, \tag{4.4}$$

then the expected prediction error is given by

$$E\left[y_i - \hat{f(x_i)}\right]^2 = \sigma^2 + E\left[\hat{f(x_i)} - f(x_i)\right]^2 = \sigma^2 + MSE\left[\hat{f(x_i)}\right], \tag{4.5}$$

thus prediction accuracy can be increased by reducing the MSE.

4.1.2. Ridge Regression

One of the simplest and most common techniques to place constraints on the coefficients is ridge regression. Ridge regression was first introduced by Hoerl and Kennard (1970), and works by shrinking the coefficients by introducing a penalty on how large they can be.

Let us first try to explain how ridge regression works in an intuitive way. In Figure 4.2, $\hat{\beta}$ represents the OLS estimates in the bi-dimensional plane. The ellipses around $\hat{\beta}$ represent different values of the parameters for which the Residual Sum of Squares (RSS) is the same, thus the unconstrained coefficients are chosen so as to minimize the RSS, and as we move away from the OLS estimates the RSS increases. Regularization works by minimizing the RSS subject to a constraint, which in the ridge regression case is represented by the shaded circle around zero.

As $\hat{\beta}$ lie outside the circle, it can be seen that the ridge estimates will differ from the OLS estimates. The optimal ridge estimates can be found where the constraint region intersects the closest ellipse to the OLS solution, that is the one that minimizes the RSS. It can easily be seen that this will happen at a frontier point of the constraint region.

The introduction of a penalty to the size of the coefficients limits the space where the coefficients can be found to the shaded circle, thereby reducing the variance. At the same time, if the true parameter vector β lies outside that region, the expected value of the estimates will not coincide with the true parameters and the ridge regression estimates will be biased. In addition, we can see that the variance will be reduced and the bias will increase as the circle becomes smaller - the penalty becomes stronger -, which sheds some light on the intuitive workings of the bias-variance trade-off implied by regularization.

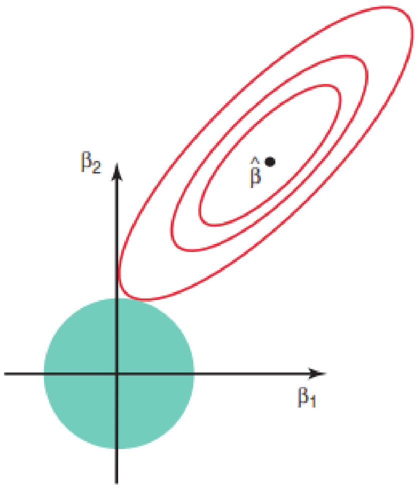

Figure 4.2: Ridge regression
Source: James et al. (2013, p. 222)

The ridge constraint is represented as a circle in the bi-dimensional space because it imposes an ℓ_2-norm penalty on the coefficients. In the specific case of the ℓ_2-norm or Euclidean norm, the length of the vector

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} \tag{4.6}$$

can be measured by the formula

$$\|\boldsymbol{\beta}\|_2 = \sqrt{\beta_1^2 + \beta_2^2}, \tag{4.7}$$

which by the Pythagorean theorem gives the distance from the origin to $\boldsymbol{\beta}$ and draws a circle.

Let us now dive into the details of ridge regression. The ridge regression estimates β_j^λ minimize a penalized RSS expression (James et al. 2013) given by

$$PRSS = \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = RSS + \lambda\sum_{j=1}^{p}\beta_j^2, \tag{4.8}$$

where $\lambda \geq 0$ is a *tuning* parameter. The shrinkage parameter $\lambda\sum_{j=1}^{p}\beta_j^2$ is small when the coefficients are close to zero, and the tuning parameter controls the amount of shrinkage applied to the parameters. When $\lambda = 0$, the ridge regression problem becomes the usual OLS problem, while as $\lambda \to \infty$ the regression coefficients will approach zero. The penalty term is sometimes written as $\frac{1}{2}\lambda\sum_{j=1}^{p}\beta_j^2$, which sometimes simplifies the mathematics. We will use both penalties interchangeably as a constant term does not play a relevant role in the problem.

Notice that the intercept is not subject to regularization, as it is just a measure of the mean value of the dependent variable when all the explanatory variables are set to zero. It has to be noted that the explanatory variables are usually *standardized* to have zero mean and unit standard deviation,

$$\tilde{x_{ij}} = \frac{x_{ij} - \bar{x}_j}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}}. \tag{4.9}$$

Centering the explanatory variables implies that the intercept now represents the expected value of the dependent variable when the predictors are set to their means, which may be a more realistic situation. *Scaling* the covariates is needed because penalized regression is not *scale invariant* (James et al. 2013). Multiplying a predictor by a constant does not lead to a simply rescaling of the associated coefficient due to the ℓ_2-norm penalty. In addition, as the shrinkage parameter considers the whole vector of coefficients, if the units of measurement of one explanatory variable is modified, this will in turn affect its coefficient and hence the *relative* amount of shrinkage applied to the other coefficients. We usually center the dependent variable as well, so that we can omit the intercept without loss of generality.

We can reformulate (4.8) to obtain a similar expression to the one that Tibshirani (1996) uses when describing another regularization technique known as *LASSO* regression,

$$(\hat{\alpha}, \hat{\beta}) = arg\ min\left\{\sum_{i=1}^{n}\left(y_i - \alpha - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\} \qquad subject\ to\ \sum_{j=1}^{p}\beta_j^2 \leq s, \tag{4.10}$$

where $s \geq 0$ is a *tuning* parameter. For every value of λ there will be a value of s which will give the same ridge estimates. This formulation is more intuitive to understand the workings of the constraint represented in Figure 4.2. When the number of explanatory variables is equal to two, $p = 2$, the ridge regression estimates are chosen so as to minimize the RSS given that they must lie within the unit circle given by $\beta_1^2 + \beta_2^2 \leq s$.

We can solve for the ridge regression solution by writing the problem in matrix form, as in Hastie, Tibshirani, and Friedman (2008),

$$PRSS(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X\beta})^T(\boldsymbol{y} - \boldsymbol{X\beta}) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta}, \tag{4.11}$$

where the ridge constraint can also be written as an ℓ_2-norm penalty of the form $\|\beta\|_2^2 = \sqrt{\beta_1^2 + ... + \beta_p^2}$. Taking a first derivative and setting the resulting equation to zero gives

$$\frac{\partial PRSS(\hat{\boldsymbol{\beta}})}{\partial\hat{\boldsymbol{\beta}}} = -2\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}) + 2\lambda\hat{\boldsymbol{\beta}} = 0, \tag{4.12}$$

and rearranging gives

$$\boldsymbol{X}^T\boldsymbol{y} = (\boldsymbol{X^T X} + \lambda\boldsymbol{I})^{-1}\hat{\boldsymbol{\beta}}, \tag{4.13}$$

where $\boldsymbol{I}$ is the $p \times p$ identity matrix. The ridge solution follows [2]

$$\hat{\boldsymbol{\beta}}^\lambda = (\boldsymbol{X^T X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}. \tag{4.14}$$

Again, the OLS estimate is obtained by simply setting $\lambda = 0$. The introduction of the penalty adds the constant λ to the diagonal entries of $\boldsymbol{X^T X}$.

Selecting the correct tuning parameter is critical. This is usually done through *cross-validation*. However, we do not really focus on this issue, and treat λ as given.

4.2. L_2-PENALIZED LOGISTIC REGRESSION

Regularization is becoming a recurrent solution to the problems posed by OLS estimation within the framework of the classical linear regression model, however there is much less literature regarding its extension to GLMs. We therefore aim at departing from the generic framework outlined in Sections 2 and 3 and generalizing regularization to Logistic Regression.

An ℓ_2 penalty can be imposed upon the unregularized logistic regression problem to reduce the size of the coefficients. We refer to this model as Ridge Logistic Regression or ℓ_2-Penalized Logistic Regression. We start from the unregularized log-likelihood $\ell(\beta)$ given by (3.20), and define the ridge log-likelihood as [3]

$$\ell_{ridge}(\beta) = \ell(\beta) - \frac{1}{2}\lambda\|\beta\|_2^2. \tag{4.15}$$

The maximizer of this equation is denoted as β^λ. Following the procedure in Section 3 gives the gradient

$$\frac{\partial\ell_{ridge}(\beta)}{\partial\beta} = X^T(y-\mu) - \lambda\beta, \tag{4.16}$$

where the first part in the right-hand side of the equation is the unregularized gradient, as given by in equation (3.28). The unregularized Fisher information given by (3.29) becomes

$$-E\left(\frac{\partial^2\ell_{ridge}(\beta)}{\partial\beta\beta'}\right) = X^T W X + \lambda I. \tag{4.17}$$

As noted in Section 3, in this particular case taking the expected value of the Hessian matrix is not of importance, however we simply note it to maintain consistency with previous derivations. To obtain the revised β^λ estimates we use the IRLS framework, and the Fisher's Scoring algorithm allows us to obtain a closed-form of the revised estimate dependent on an alternative, working response z. The proof follows below.

Proof. From (2.26) the Fisher's Scoring method updates as follows:

$$\beta_1 = \beta_0 + I^{-1}(\beta_0)u(\beta_0), \tag{4.18}$$

where β_1 and β_0 represent the new and old estimates respectively, and the Fisher information matrix I and the gradient u are given by (4.16) and (4.17) respectively. Incorporating these expressions,

$$\beta_1 = \beta_0 + (X^T W X + \lambda I)^{-1}\left[X^T(y-\mu) - \lambda\beta_0\right], \tag{4.19}$$

[2] From now on we can omit the use of *hats* to denote the estimated parameter for the sake of simplicity.

[3] We limit ourselves to the use of matrix notation along the derivation, hence the use of bold letters to denote matrix and vectors can be ignored for once.

and pre-multiplying the first parte of the right-hand side of the equation by $(X^TWX + \lambda I_n)^{-1}(X^TWX + \lambda I)$ gives

$$\beta_1 = (X^TWX+\lambda I_n)^{-1}(X^TWX+\lambda I)\beta_0+(X^TWX+\lambda I_n)^{-1}\left[X^T(y-\mu)-\lambda\beta_0\right], \quad (4.20)$$

and by expanding the equation and adding WW^{-1} just before $(y - \mu)$ we obtain

$$\begin{aligned}\beta_1 &= (X^TWX + \lambda I)^{-1}X^TWX\beta_0 + (X^TWX + \lambda I)^{-1}\lambda\beta_0 \\ &+ (X^TWX + \lambda I)^{-1}X^TWW^{-1}(y-\mu) - (X^TWX + \lambda I)^{-1}\lambda\beta_0 \\ &= (X^TWX + \lambda I)^{-1}X^TWX\beta_0 + (X^TWX + \lambda I_n)^{-1}X^TWW^{-1}(y-\mu) \\ &= (X^TWX + \lambda I)^{-1}X^TW\left\{X\beta_0 + W^{-1}(y-\mu)\right\}. \quad (4.21)\end{aligned}$$

Thus the revised penalized estimate has the form

$$\beta^\lambda = (X'WX + \lambda I)^{-1}X^TWz, \quad (4.22)$$

where the working dependent variable z has the usual form $z = X\beta + W^{-1}(y - \mu)$. □

The penalized parameter has the same form as in the unregularized case (see equation 3.39), but a constant value λ to the X^TWX matrix is added before taking its inverse. As we can see from (4.22), the ridge parameter λ controls the amount of shrinkage applied to the parameters. When $\lambda = 0$, the estimate β^λ has the known form derived in Section 3, so that this specification contains the usual logistic regression as a particular case. Conversely, as λ tends to infinity β^λ is shrunk towards zero.

As an extension to the fact that the IRLS algorithm solves a WLS problem at each iteration, resulting in the estimate given by equation (3.39), it can be shown that the penalized IRLS algorithm solves a penalized WLS problem at each iteration.

Proof. The ℓ_2-penalized residual sum of squares of a WLS problem could be written as:

$$PRSS(\beta) = (z - X\beta)^TW(z - X\beta) + \lambda\|\beta\|_2^2, \quad (4.23)$$

where the penalization term adds up as the residual sum of squares is to be minimized. Taking first derivatives and setting the equations to zero gives

$$\frac{\partial PRSS(\beta)}{\partial\beta} = -2X^TW(z - X\beta) + 2\lambda\beta = 0, \quad (4.24)$$

and rearranging gives

$$X^TWz = (X^TWX + \lambda I)\beta. \quad (4.25)$$

The penalized estimate follows,

$$\beta^\lambda = (X^TWX + \lambda I)^{-1}X^TWz, \quad (4.26)$$

which has the exact form as in (4.22). □

4.2.1. Bias and variance of the penalized estimator

Throughout this section, we try to obtain an expression for the bias and variance of the penalized estimator, following Le Cessie and Houwelingen (1992). In addition, we discuss the issues posed by regularization in inference and standard errors.

Consider again the ridge log-likelihood as defined in equation (4.15),

$$\ell^\lambda(\beta) = \ell(\beta) - \lambda\|\beta\|_2^2, \quad (4.27)$$

where we have trivially omitted the constant $1/2$ from the penalty term to keep consistency with the results of Le Cessie and Houwelingen (1992). Going through the maximization procedure again, the gradient, which is equivalent to the one derived in equation (4.16), may now be written as

$$u^{\lambda}(\beta) = X^T(y-\mu) - 2\lambda\beta = u(\beta) - 2\lambda\beta, \tag{4.28}$$

where $u(\beta)$ is the unregularized gradient. The negative of the Hessian matrix can be written as follows with simpler notation

$$\Omega^{\lambda}(\beta) = \Omega(\beta) + 2\lambda I, \tag{4.29}$$

where $\Omega = X^TWX$ (see equation 4.17). Large sample properties of the regularized maximum likelihood estimators can be obtained by carrying out a first-order Taylor series expansion of the gradient about the real population parameter β_o (Le Cessie and Houwelingen 1992, p. 194). Thus

$$u^{\lambda}(\hat{\beta}^{\lambda}) = u^{\lambda}(\beta_0) - (\hat{\beta}^{\lambda} - \beta_0)'\Omega^{\lambda}(\beta_0) + R_1(\hat{\beta}^{\lambda}) \tag{4.30}$$

where the remainder $R_1(\hat{\beta}^{\lambda})$ is negligible. Using the well-known fact that the gradient evaluated at $\hat{\beta}^{\lambda}$ equals zero $u^{\lambda}(\hat{\beta}^{\lambda}) = 0$, the expression translates into a Newton-Raphson algorithm,

$$\begin{aligned} u^{\lambda}(\beta_0) - (\hat{\beta}^{\lambda} - \beta_0)'\Omega^{\lambda}(\beta_0) &= 0 \\ \hat{\beta}^{\lambda\prime}\Omega^{\lambda}(\beta_0) &= u^{\lambda}(\beta_0) + \beta_0'\Omega^{\lambda}(\beta_0) \end{aligned} \tag{4.31}$$

from which we can obtain a first approximation to $\hat{\beta}^{\lambda}$, using (4.29) and (4.30) for β_0 as follows

$$\begin{aligned} \hat{\beta}^{\lambda} &= \Omega^{\lambda}(\beta_0)^{-1}u^{\lambda}(\beta_0) + \beta_0 \\ &= \beta_0 + \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\left\{u(\beta_0) - 2\lambda\beta_0\right\}, \end{aligned} \tag{4.32}$$

thus setting $\lambda = 0$ follows that a first approximation to the unregularized MLE estimator could be

$$\hat{\beta} = \beta_0 + \Omega^{-1}(\beta_0)u(\beta_0). \tag{4.33}$$

It is well-known that under certain regularity conditions, the MLE is asymptotically unbiased and its covariance matrix is given by $\Omega(\beta_0)^{-1}$. We can now show that the ridge logistic estimator is biased, obtain an expression for this bias, and derive and expression for the variance.

Proof. From (4.32) and taking expectations,

$$\begin{aligned} \hat{\beta}^{\lambda} &= \beta_0 + \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\left\{u(\beta_0) - 2\lambda\beta_0\right\} \\ E(\hat{\beta}^{\lambda}) &= \beta_0 + \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\Big\{E\left[u(\beta_0)\right] - 2\lambda\beta_0\Big\}, \end{aligned} \tag{4.34}$$

and using the fact that the score evaluated at the true parameter value has mean zero, $E[u(\beta_0)] = 0$, we obtain

Bias

$$E(\hat{\beta}^{\lambda}) = \beta_0 - 2\lambda\left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\beta_0 \tag{4.35}$$

thus the ridge logistic estimate is asymptotically biased. When $\lambda = 0$, it follows that the MLE estimator $\hat{\beta}$ is asymptotically unbiased. Hence the asymptotic bias is

$$E\left[\hat{\beta}^{\lambda} - \beta_0\right] = -2\lambda\left\{\Omega(\beta_0) + 2\lambda I_n\right\}^{-1}\beta_0. \tag{4.36}$$

The asymptotic variance of the ridge logistic estimator can be derived as follows. Subtract equation (4.35) from (4.32),

$$\begin{aligned}\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda}) &= \beta_0 + \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\left\{u(\beta_0) - 2\lambda\beta_0\right\} \\ &\quad - \left[\beta_0 - 2\lambda\left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\beta_0\right] \\ &= \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1} u(\beta_0), \qquad (4.37)\end{aligned}$$

now square the expression,

$$\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]' = \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1} u(\beta_0)u(\beta_0)'\left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}, \quad (4.38)$$

and finally take the expected value

$$E\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]' = \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1} E\left[u(\beta_0)u(\beta_0)'\right]\left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}. \quad (4.39)$$

Another useful statistical property of the score function (Rodríguez 2007) is that its covariance matrix is given by the observed information matrix $var[u(\beta_0)] = E\left[u(\beta_0)u(\beta_0)'\right] = \mathcal{J}(\beta_0)$ which as we said is replaced by the Fisher information $I(\beta_0)$ in practice. From (4.29), we know that $I(\beta_0) = \Omega(\beta_0)$ using our notation, so that finally

$$E\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]\left[\hat{\beta}^{\lambda} - E(\hat{\beta}^{\lambda})\right]' = \left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}\Omega(\beta_0)\left\{\Omega(\beta_0) + 2\lambda I\right\}^{-1}. \quad (4.40)$$

Variance

It allows us to demonstrate that the asymptotic variance of the unrestricted MLE is in fact

$$E\left[\hat{\beta} - E(\hat{\beta})\right]\left[\hat{\beta} - E(\hat{\beta})\right]' = \Omega(\beta_0)^{-1} \quad (4.41)$$

when $\lambda = 0$. □

These expressions are derived as a function of the true population parameter β_0. Their usefulness lies in the ability to provide a formal explanation to the introduction of bias in the estimator and the fact that you can think of unregularized logistic regression as a particular case of these more complex expressions.

4.2.2. A note on standard errors

It is fairly common to report the standard errors, *t-ratios*, confidence intervals or *p-values* of the estimated parameters $\hat{\beta}$ in econometric problems. We may wonder whether these techniques have an extension to penalized regression.

Although as we have seen that it can be difficult to obtain closed mathematical expressions for such measures, some insights into the variability of the estimators and standard errors can be drawn. Le Cessie and Houwelingen (1992, p. 194) cites *jackknife* and *bootstrapping* as feasible statistical methods. However, the usual practice in penalized regression applications is not to report standard errors. The reason is that as regularization introduces significant bias and artificially reduces the variance, the standard errors themselves would be *biased*, not being as informative as in the classical model. In fact, some software packages that incorporate penalized regression explicitly address this issue by arguing that they do not provide standard errors as these are meaningless (Goeman 2010). This view is also shared by Le Cessie and Houwelingen (1992, p. 194), who notes that the approximation given in (4.40) cannot be used to construct confidence intervals, as we need to account for the bias of the estimate.

Standard errors in a regularized regression problem may give a wrong impression of precision. The fact that estimates are considerably biased should not be overlooked, as it is an important source of inaccuracy and unbiasedness still remains a desirable property.

4.2.3. A penalized IRLS algorithm

In Section 3 we outlined the IRLS procedure and in Algorithm 2 we described the iterative process used to estimate the coefficients. Now, we aim at generalizing Algorithm 2 to handle the introduction of a regularization constraint in the form of a ridge penalty, as given in equation (4.8).

Algorithm 3 describes this *generalized* IRLS algorithm. It allows for the introduction of a penalty term depending on the value of λ, so that unconstrained regression can still be obtained by setting $\lambda = 0$. [4]

One of the most common *R* packages to fit regularized Generalized Linear Models is the *glmnet* package, developed by Friedman, Hastie, and Tibshirani (2010). The algorithm used is a form of cyclical coordinate descent, however the resulting coefficients should be the same as those calculated by our algorithm. In fact, this is how we have checked that our algorithm was correct. In order to obtain the same results, care should be taken as the objective function is slightly different from that in (4.15). Here the negative log-likelihood is used, thus the following function is to be minimized

$$\ell_{glmnet}(\beta) = -\frac{1}{n}\ell(\beta) + \lambda\big[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1\big], \tag{4.42}$$

where the tuning parameter λ is now accompanied by the parameter α, which is used to select the regularization technique to be applied to the data. If $\alpha = 0$, then a ridge penalty is used. Conversely, if $\alpha = 1$ a method known as $LASSO$ regression is applied.

Check Figure A.1 in Appendix A.1 for the *R* code of our algorithm, a more detailed description and the correct specification needed to as to obtain exactly the same results using the *glmnet* package.

[4] Note that we take advantage of the fact that standardization of the covariates is convenient to center the response variable y and hence omit the intercept.

Algorithm 3 IRLS for Ridge Logistic Regression

```
Set the initial estimate for the parameters to zero
    $\boldsymbol{\beta} = 0$
Set the value of the tuning parameter $lambda\ \lambda$
while k<MaxIterations do
    Compute the linear predictor
        $\eta = \sum_{j=1}^{p} \beta_j x_j$
    Compute the fitted value
        $\mu \equiv \pi = \frac{1}{1 + exp(-\eta)}$
    Calculate the weights and define the weight matrix
        $w = \pi(1-\pi); \qquad W = diag\{w\}$
    Construct the working dependent variable
        $z = \eta + \frac{(y-\mu)}{w}$
    Regress $z$ on the covariates and weight $W$ to solve the penalized WLS problem
    so as to obtain the new estimate $\beta_{k+1}$
        $\beta^{\lambda} = (X'WX + \lambda I)^{-1} X^T W z \qquad where\ I\ is\ the\ p \times p\ identity\ matrix$
    if the stopping criterion is satisfied then
        Break;
    end if
end while
```

5. EXTENSIONS: A LASSO SHRINKAGE PARAMETER

The availability of huge data sets has led to the apparition of many new regularization techniques. We will focus here on one of these alternatives, known as *LASSO* regression.

5.1. LASSO REGRESSION

The *LASSO* is an estimation method that stands for 'least absolute shrinkage and selection operator'. It was proposed by Tibshirani (1996).

It retains the good features of ridge regression, namely it performs coefficient shrinkage and may improve prediction accuracy and MSE compared to OLS estimates. In addition, it possesses a differential feature, which tackles another reason why the practitioner may not be satisfied with the OLS solution: it performs *variable selection*. Often we have a large number of explanatory variables p and we are interested in isolating a smaller subset which retains the variables exhibiting the strongest effects, for the sake of interpretation. Ridge regression is not able to do that, as it only penalizes the size of the coefficients but retains the p predictors in the final model. Only if $\lambda = \infty$ the coefficients will all be set equal to zero and we will obtain a sparse, simple model. As Tibshirani (1996) notes, the LASSO tries to outperform the standard shrinkage method - ridge regression - and the standard variable selection method - known as *subset selection*, which is an unstable procedure in the sense that selection is a discrete mechanism strongly influenced by small changes in the data -. To do so, the LASSO shrinks some coefficients toward zero and directly sets others to zero, resulting in a more interpretable model.

The intuitive way to understand how the LASSO penalty works is depicted in Figure 5.1, where again $\hat{\beta}$ represents the OLS estimates in the two-dimensional plane and this vector is surrounded by ellipses representing equal RSS.

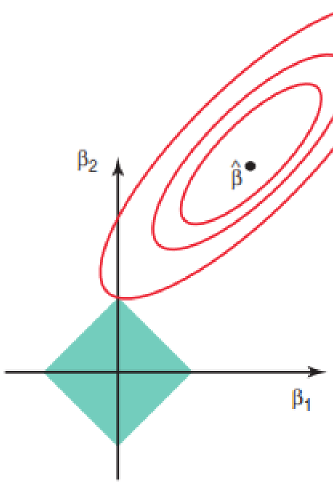

Figure 5.1: Ridge regression
Source: James et al. (2013, p. 222)

The LASSO constraint is now pictured as a diamond in the bi-dimensional space because it imposes an ℓ_1-norm penalty on the coefficients. The length of the vector $\boldsymbol{\beta}$ from (4.6) as measured by an ℓ_1-norm is given by

$$\|\boldsymbol{\beta}\|_1 = |\beta_1| + |\beta_2|. \tag{5.1}$$

Following the same intuition as for ridge estimates, it can as well be seen that LASSO estimates would be biased and show smaller variance than OLS estimates. However, this specific characterization allows for a very special departure from ridge regression. Since the ridge constraint was a circle, the intersection of the shaded region and the RSS curves will not generally occur over one of the axis in the bi-dimensional space, and the vector of estimates would not have zero entries. However, the LASSO constraint has sharp points at the axis, and the solution to the problem will often fall within one of them. Generalizing the example to higher dimensions, this implies that many coefficients will be set equal to exactly zero, thus performing feature selection (James et al. 2013) by identifying the most relevant subset of covariates. The diamond representation comes from the fact that the ℓ_1-norm penalty simply means that the sum of absolute magnitudes of the two coefficients must be equal or less than some fixed value.

Lasso regression is technically defined as follows. The LASSO coefficients $\tilde{\beta}_j$ minimize a penalized RSS (James et al. 2013) given by

$$PRSS = \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j|, \tag{5.2}$$

where $\lambda \geq 0$ is again the tuning parameter which serves to regulate the amount of shrinkage applied to the coefficients and the only difference from ridge regression is the substitution of an ℓ_2-norm penalty for an ℓ_1-norm penalty. Tibshirani (1996) does also mention that the LASSO assumes standardized explanatory variables.

Similarly to ridge regression, we can reformulate (5.2) to present the problem as Tibshirani (1996) does

$$(\hat{\alpha}, \hat{\beta}) = arg\ min\left\{\sum_{i=1}^{n}\left(y_i - \alpha - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\} \qquad subject\ to\ \sum_{j=1}^{p}|\beta_j| \leq t, \tag{5.3}$$

where $s \geq 0$ is a tuning parameter. This formulation allows us to intuitively understand how the LASSO solution is achieved and compare Figures 4.2 and 5.1. When $p = 2$, the LASSO regression estimates are chosen so as to minimize the RSS given that they must live within the constraint given by $|\beta_1| + |\beta_2| \leq t$.

The main problem posed by the LASSO penalty is that it is non-linear and non-differentiable, hence it is not possible to derive an analytical solution as in ridge regression (4.14).

5.2. L_1-PENALIZED LOGISTIC REGRESSION

The extension of the LASSO to logistic regression suffers from the same problem. As it stands, it is not possible to obtain a closed-form solution for the LASSO logistic estimator and investigate the behaviour of its bias and variance.

In an effort to better understand the workings of the LASSO parameter and to offer a mathematical approximation, we may follow the advice of Tibshirani (1996, p. 272) who considers that an approximate estimate may be derived by rewriting the LASSO penalty $\lambda\sum|\beta_j|$ as

$$\lambda\sum\beta_j^2/|\beta_j|. \tag{5.4}$$

This allows us in principle to rearrange a mathematical expression for the LASSO logistic estimator $\tilde{\beta}$. Tibshirani (1996) does so for multiple linear regression, hence proposing that the LASSO estimate may be approximated by a ridge regression of the form

$$\beta^* = (X^T + \lambda W^-)^{-1}X^T y, \tag{5.5}$$

where $W = diag\left\{|\tilde{\beta}_j|\right\}$ and W^- refers to its generalized inverse. The reason why the author uses W^- may be to avoid a singular matrix, as there may be some $\tilde{\beta}_j = 0$.[5] In

[5] See Rao (2002, p. 24) for more details about generalized inverses.

addition, the proposed approximation to the covariance matrix is

$$(X^T + \lambda W^-)^{-1} X^T X (X^T + \lambda W^-)^{-1} \hat{\sigma}^2, \tag{5.6}$$

where $\hat{\sigma}^2$ is the error variance estimate.

We now prove that an extension of this approximation is viable for LASSO logistic regression. The ridge logistic framework outlined in Section 4 will be of help throughout the derivation.

Proof. Using the ridge log-likelihood in (4.27) as a baseline, we should modify the ridge penalty as proposed in equation (5.4),

$$\ell^*(\beta) = \ell(\beta) - \lambda \beta' W^- \beta. \tag{5.7}$$

We now go through the maximization procedure, taking into account that a derivative with respect to β does not affect the components $|\tilde{\beta}|$ of the generalized inverse, so that the matrix is taken as a constant. Similarly to (4.28), the gradient becomes

$$u^*(\beta) = u(\beta) - 2\lambda W^- \beta, \tag{5.8}$$

and the Fisher information

$$\Omega^*(\beta) = \Omega(\beta) + 2\lambda W^-. \tag{5.9}$$

A first-order Taylor series expansion about the real population parameter β_0 gives

$$u^*(\hat{\beta}^*) = u^*(\beta_0) - (\hat{\beta}^* - \beta_0)' \Omega^*(\beta_0) + R_1(\hat{\beta}^*), \tag{5.10}$$

where the remainder is again negligible. Using $u^*(\hat{\beta}^*) = 0$, a first approximation to the estimate may be obtained as

$$\begin{aligned} \hat{\beta}^* &= \Omega^*(\beta_0)^{-1} u^*(\beta_0) + \beta_0 \\ &= \beta_0 + \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} \left\{ u(\beta_0) - 2\lambda . W^- \beta_0 \right\} \end{aligned} \tag{5.11}$$

Equivalently, taking expectations it can be shown that the approximated LASSO logistic estimate is asymptotically biased

$$\begin{aligned} E(\hat{\beta}^*) &= \beta_0 + \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} \left\{ E\left[u(\beta_0) \right] - 2\lambda W^- \beta_0 \right\} \\ &= \beta_0 - 2\lambda \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} W^- \beta_0, \end{aligned} \tag{5.12}$$

with bias given by

$$E(\hat{\beta}^* - \beta_0) = -2\lambda \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} W^- \beta_0. \tag{5.13}$$

Following the same steps as in the ridge logistic part, the asymptotic variance of this approximated LASSO estimator is obtained as follows from (5.11) and (5.12),

$$\begin{aligned} \hat{\beta}^* - E(\hat{\beta}^*) &= \beta_0 + \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} \left\{ u(\beta_0) - 2\lambda W^- \beta_0 \right\} \\ &\quad - \left[\beta_0 - 2\lambda \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} W^- \beta_0 \right] \\ &= \left\{ \Omega(\beta_0) + 2\lambda W^- \right\}^{-1} u(\beta_0), \end{aligned} \tag{5.14}$$

and taking the expectation of the squared difference,

$$\begin{aligned}\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]' &= \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1} u(\beta_0)u(\beta_0)' \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1} \\ E\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]' &= \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1} E\left[u(\beta_0)u(\beta_0)'\right] \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1}\end{aligned} \tag{5.15}$$

and finally replacing $E\left[u(\beta_0)u(\beta_0)'\right] = \mathcal{J}(\beta_0)$ by the Fisher information $I(\beta_0)$

$$E\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]\left[\hat{\beta}^* - E(\hat{\beta}^*)\right]' = \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1} \Omega(\beta_0) \left\{\Omega(\beta_0) + 2\lambda W^-\right\}^{-1}. \tag{5.16}$$

□

It turns out that the LASSO regression approximation proposed by Le Cessie and Houwelingen (1992) can be extended to LASSO logistic regression in this way. The usual asymptotically unbiased MLE estimator and its variance can still be obtained by setting $\lambda = 0.$, and the only difference with the ridge bias in (4.35) and variance in (4.40) is the generalized inverse matrix.

Finally, as discussed in Section 4.2.2, standard errors may be computed using bootstrap techniques, although as we noted they lack significance within the regularized regression framework. Kyung et al. (2010) discuss some alternatives for its computation, and suggests that there is not much consensus on a statistically accurate method.

5.2.1. Fitting the model

The approximation to the LASSO penalty given in (5.4) suggests that we may use an extension of the ridge regression algorithm to compute the LASSO estimate itself, however Tibshirani (1996) notes that it is quite inefficient.

Different estimation methods have been proposed. As an example, consider *Least Angle Regression* by Efron et al. (2004). In Section 6 we will use the *R glmnet* package to test the model. As noted in equation (4.41), the penalized log-likelihood in this package is given by

$$\ell_{glmnet}(\beta) = -\frac{1}{n}\ell(\beta) + \lambda\big[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1\big], \tag{5.17}$$

so we will set $\alpha = 1$ whenever we want to conduct ℓ_1 penalized logistic regression. To be consistent with the specification made in Algorithm 3 and the way *glmnet* fits ridge logistic regression, the adjustment proposed in equation (A.3) from Appendix A is needed.

6. MONTE CARLO SIMULATION RESULTS

Throughout this section, we explore how the logistic regression model reacts to regularization. The simulations are done using the software *R* and the *glmnet* package described in Section 4. Occasionally we use *glmnet* instead of our algorithm as the code looks cleaner, however remember that these are equivalent and substituting one for the another leads exactly the same results under the modifications proposed in Appendix A.1.

6.1. A LATENT-VARIABLE FORMULATION

In order to specify the model used in the experiments, we introduce a different mathematical formulation for logistic regression. The formulation given in Section 2 has an equivalent specification as a *latent-variable model*. A comprehensive derivation is included in Long (1997), which we follow here.

Suppose that there is a continuous latent variable which is unobserved and can be written as a function of a set of explanatory variables and a random error term that is distributed according to a standard logistic distribution with mean zero and variance $\pi^2/3$

$$y_i^* = \boldsymbol{X}_i\boldsymbol{\beta} + \epsilon_i \qquad where\ \epsilon_i \sim Logistic(0, \pi^2/3). \tag{6.1}$$

The latent variable could represent utility, for example in the labor force participation problem it could measure the underlying propensity to work that motivates the decision of the individual. This unobservable variable is linked to the observed binary response y_i by the equation

$$y_i = \begin{cases} 1, & \text{if } y_i^* > 0 \\ 0, & \text{if } y_i^* \leq 0, \end{cases} \tag{6.2}$$

where the *thresold* or *cutpoint* is assumed to be zero without loss of generality. This helps us avoid a source of unidentification of the intercept because we set $\tau = 0$. Note that we have previously discussed the convenience of setting the intercept equal to zero, which may also avoid this problem.

The assumption about the error term of the latent-variable specification may seem unusual, but is convenient because it leads to a particularly simple cumulative distribution function (Long 1997, p.42)

$$\Lambda(\epsilon) = \frac{exp(\epsilon)}{1 + exp(\epsilon)}, \tag{6.3}$$

which is the logistic function. Thus the probability of observing a positive outcome given $\boldsymbol{X}$ is

$$Pr(\boldsymbol{y} = 1|\boldsymbol{X}) = Pr(\boldsymbol{y}^* > 0|\boldsymbol{X}) = Pr(\boldsymbol{X}\boldsymbol{\beta} + \epsilon > 0|\boldsymbol{X}), \tag{6.4}$$

and rearranging gives

$$Pr(\boldsymbol{y} = 1|\boldsymbol{X}) = Pr(\epsilon > -\boldsymbol{X}\boldsymbol{\beta}|\boldsymbol{X}), \tag{6.5}$$

and since the cumulative distribution function is defined in terms of the probability of the random variable being less than some value, we change the direction of the inequality as

$$Pr(\boldsymbol{y} = 1|\boldsymbol{X}) = Pr(\epsilon > -\boldsymbol{X}\boldsymbol{\beta}|\boldsymbol{X}) = 1 - Pr(\epsilon \leq -\boldsymbol{X}\boldsymbol{\beta}|\boldsymbol{X}), \tag{6.6}$$

which gives the cumulative distribution function of the error term evaluated at $-\boldsymbol{X}\boldsymbol{\beta}$. Using (6.3), this can be expressed as

$$Pr(\boldsymbol{y} = 1|\boldsymbol{X}) = 1 - \Lambda(-\boldsymbol{X}\boldsymbol{\beta}). \tag{6.7}$$

Given that the logistic distribution is symmetric about zero, $F(\epsilon) = 1{-}F(-\epsilon)$, we can write

$$Pr(\boldsymbol{y} = 1|\boldsymbol{X}) = \Lambda(\boldsymbol{X\beta}) = \frac{exp(\boldsymbol{X\beta})}{1 + exp(\boldsymbol{X\beta})} = \pi, \tag{6.8}$$

which is the logistic regression model as specified in equation (3.17).

By assuming that the errors follow the *standard* logistic distribution function given in (6.3), we avoid another source of unidentification. As noted by Long (1997, p. 47), this is an arbitrary but *necessary* assumption to identify the model. Imagine that the error term has a variance with *scale* four $\epsilon_i \sim Logistic(0, 4 \times \pi^2/3)$. Then, to obtain the function given in (6.3), some standardization is needed

$$\begin{aligned} \epsilon \sim L(0, 4 \times \pi^2/3) \ &\rightarrow \ \frac{\epsilon}{2 \times \pi/\sqrt{3}} \sim L(0,1) \ \rightarrow \ \frac{\epsilon}{2 \times \pi/\sqrt{3}}\pi/\sqrt{3} \sim L(0, \pi^2/3) \\ &\rightarrow \ \frac{\epsilon}{2} \sim L(0, \pi^2/3). \end{aligned} \tag{6.9}$$

This has a similar effect to the introduction of a nonzero cutpoint, but in this case it rescales all the coefficients as $\hat{\beta}_j = \hat{\beta}^*_j/2$. If the variance of the error is known, then the assumption made about the error variance is trivial, because we can simply multiply the estimated coefficients by 2. However, if the variance is unknown, the $\hat{\beta}^*_j$ parameters remain unidentified.

The logit model is sometimes derived from a *two-way* latent-variable model, where the latent model is understood as a utility index model. A good review of these models is given by Train (2009).

6.2. COEFFICIENT SHRINKAGE

In this section we are concerned with showing how the coefficients of a penalized logistic regression model shrink toward zero as the value of the tuning parameter λ increases. The *R* code used for this experiment can be found in Appendix A.2 in Figure A.3.

In brief, we set-up a latent-variable model, where the explanatory variables are standardized and correlated and follow the uniform distribution, while the error term follows the standard logistic distribution. As described in the previous section in equation (6.2), the latent response assigns the values of the observed binary dependent variable depending on a zero cutpoint. Then, we obtain the coefficients of an unregularized logistic model, a ridge logistic model and a LASSO logistic model using the package *glmnet* - or, in the case of the ridge penalty, our algorithm -. This is done for a sequence of values of λ, which in this case ranges from 0 to 200 by 0.5 intervals. In this experiment we use a sample size equal to 1000 observations and introduce eight explanatory variables, where the true parameters are given by the vector [6]

$$\boldsymbol{\beta} = \begin{pmatrix} -3 & 5 & 2 & 0.7 & -0.3 & 0.5 & -0.8 & 1 \end{pmatrix}.$$

Figure 6.1 shows the behavior of the ℓ_2-penalized logistic regression model coefficients. Note that we are not interested in identifying a particular coefficient itself but in looking at the whole picture, thus we do not label the coefficients.

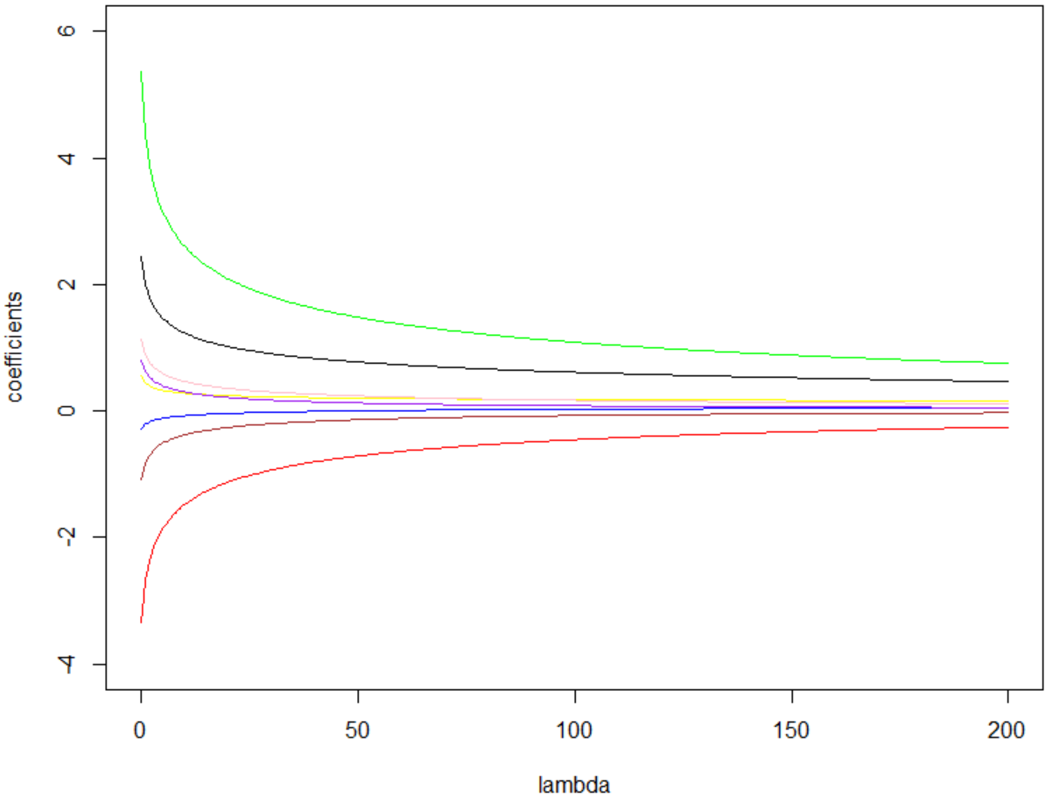

Figure 6.1: ℓ_2-penalized logistic regression
Source: Own elaboration

As expected, it can be seen that the coefficients approach zero as the value of the tuning parameter increases. Notice that the sequence of values of λ included has very

[6] In fact, we just need to define the vector of true parameters, and the code itself matches the number of predictors needed. For further details see Appendix A.2.

high values, which are probably not optimal, but interesting to study the behaviour of β. Although for small values of λ there are still notable differences among the coefficients, as it approaches $\lambda = 200$ all of them are shrunk toward zero and toward each other. It can also be checked that ridge regression does not perform variable selection, and all the covariates are still included in the model although all the coefficients get very close to zero. Note that according to Tibshirani (1996, p. 272), when the explanatory variables are correlated ridge regression does not necessarily perform proportional shrinkage, and estimates may be shrunken differently, so that a strange behaviour of some coefficients should not be surprising.

On the other hand, Figure 6.2 shows the behavior of the ℓ_2-penalized logistic regression model coefficients.

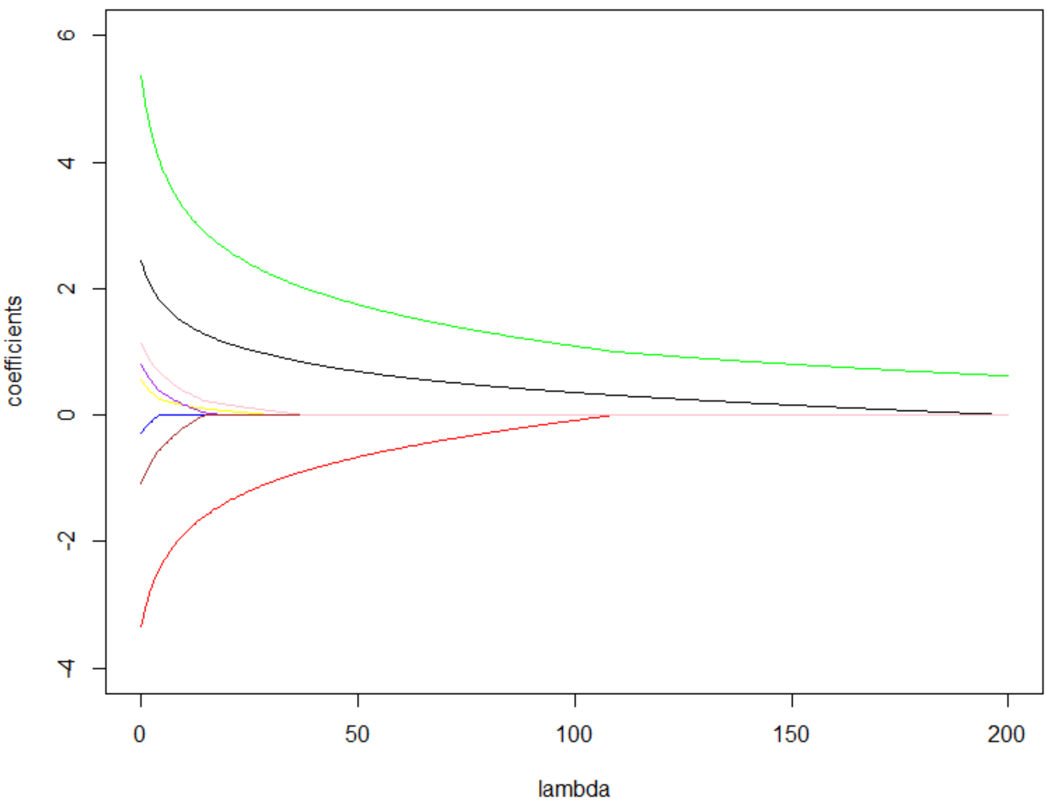

Figure 6.2: ℓ_1-penalized logistic regression
Source: Own elaboration

Although the shrinkage property of the LASSO is equally clear in this case, and for values of λ very close to zero the behavior of the parameters looks similar to the previous case, there are some striking differences, in line with the theory, namely the variable selection feature of the LASSO. The parameters that were close to zero in the unrestricted model are set to zero quite early in the sequence, in fact and for $\lambda = 50$ some explanatory variables would be dropped from the final model. As $\lambda = \infty$, the penalty gives the null model, with all coefficients set to zero. In fact, for $\lambda = 200$ only the parameter $\beta_j = 5$ is still nonzero.

One of the drawbacks of the LASSO is that if there is a group of highly correlated variables, as this may be the case, it tends to arbitrarily pick one and discard the others. Conversely, the ridge penalty shrinks the coefficients towards each other.

6.3. THE BIAS-VARIANCE TRADE-OFF

We explore here the bias-variance trade-off for the two penalized logistic regression models outlined before. In Section 4.1.1 we proposed the MSE of an estimator as a loss function which could be decomposed in bias and variance, as shown in (4.1). We will check here whether the regularization methods actually lead to lower MSE than the unregularized case.

As $\boldsymbol{\beta}$ is not a scalar, the MSE is written as

$$MSE(\hat{\boldsymbol{\beta}}) = E(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T, \tag{6.10}$$

and we add and subtract $E(\hat{\boldsymbol{\beta}})$ to both terms

$$\begin{aligned} MSE(\hat{\boldsymbol{\beta}}) &= E\left[\hat{\boldsymbol{\beta}} - \boldsymbol{\beta} + E(\hat{\boldsymbol{\beta}}) - E(\hat{\boldsymbol{\beta}})\right]\left[\hat{\boldsymbol{\beta}} - \boldsymbol{\beta} + E(\hat{\boldsymbol{\beta}}) - E(\hat{\boldsymbol{\beta}})\right]^T \\ &= E\left[\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}})\right]\left[\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}})\right]^T + \left[E(\hat{\boldsymbol{\beta}}) - \boldsymbol{\beta}\right]\left[E(\hat{\boldsymbol{\beta}}) - \boldsymbol{\beta}\right]^T, \end{aligned} \tag{6.11}$$

where the first term is the variance-covariance matrix and the second term the squared bias. In order to do comparisons, working with the variance-covariance matrix is not as practical as working with the variance as in the scalar case. Thus we redefine the MSE function as an alternative *risk function* given by

$$risk(\hat{\boldsymbol{\beta}}) = E(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}), \tag{6.12}$$

which is a scalar. Using the property of the trace $tr(c) = c$ for some scalar c this becomes

$$risk(\hat{\boldsymbol{\beta}}) = tr\left\{E(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})\right\}, \tag{6.13}$$

and by the fact that the trace is a linear operator so that it commutes with the mathematical expectation and using the property $tr(AB) = tr(BA)$ where A and B are matrices, the risk function becomes

$$risk(\hat{\boldsymbol{\beta}}) = E\left\{tr(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})\right\} = E\left\{tr(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T\right\}. \tag{6.14}$$

Finally, commuting the trace and the expectation once again, we obtain that the risk function defined in (6.15) is the sum of the diagonal elements of the MSE matrix

$$risk(\hat{\boldsymbol{\beta}}) = tr\left\{E(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T\right\} = tr\left[V(\hat{\boldsymbol{\beta}})\right] + ||bias(\hat{\boldsymbol{\beta}})||^2, \tag{6.15}$$

where $V(\hat{\boldsymbol{\beta}})$ is the covariance matrix and $||bias(\hat{\boldsymbol{\beta}})||^2 \equiv \sum_{j=1}^{p}(E(\hat{\beta}_j) - \beta_j)^2$. The former adds up the variances and the latter implies that $risk(\hat{\boldsymbol{\beta}})$ is $E\left[\sum_{j=1}^{p}(\hat{\beta}_j - \beta_j)^2\right]$. This is in fact the MSE for the multivariate case, where we use the ℓ_2 norm to measure the size of the vector (Flury 1997), but we will refer to it as our *risk* function to avoid confusion.

We check whether this loss function decreases or not when we introduce a regularization method. The *R* code for the experiment can be found in Figure A.4 of Appendix A.3. The set-up of the logistic regression model is the same outlined in Section 6.2. The only differences are that we consider now a larger vector of coefficients

$$\boldsymbol{\beta} = \begin{pmatrix} -3 & 4 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.16}$$

and that the value of λ is unique and chosen *ad-hoc*. The former difference aims at highlighting the variable selection feature of the LASSO, while the latter simplifies the problem, as we want to show that for *some* values of λ regularization performs well. We set a conservative value $\lambda = 0.5$. The sample size is 1000 observations and we conduct 1000 replications.

The code is designed so as to compute the $\boldsymbol{\beta}$ for the three different models - unregularized, ridge and LASSO logistic regression -, and then compute the *empirical* bias and variance for each coefficient as

$$\begin{aligned} bias &= E\left[\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right] \\ variance &= E\left[\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}})\right]\left[\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}})\right]^T, \end{aligned} \tag{6.17}$$

and then compute the MSE of *each* coefficient. Finally, as we sum over all the coefficients for each model, we are in fact computing the *risk* as defined in equation (6.15). The results of the experiment are summarized in the Table 6.1, which shows that regularization substantially decreases the *risk* or expected squared error, thus reinforcing the theoretical results outlined before.

Unregularized	Ridge	Lasso
1.233519	0.823114	0.8822589

Table 6.1: *Risk* results of the experiment
Source: Own elaboration

It looks that the ridge penalty performs slightly better than the LASSO penalty in this case, which may be due to the degree of correlation introduced among the predictors. In these cases, the arbitrary selection performed by the LASSO may lead ridge regression to perform better, as noted by Tibshirani (1996, p.283).

These results are mostly based on the fact that the choice of λ was *ad-hoc*. Therefore, it may be interesting to look at a sequence of λ values to see how the various components of the *risk* behave. We conduct a second experiment that builds on the code used for the first one but extends it so as to compute the *risk* and its bias and variance components for a user-provided sequence of λ values. The *R* code can be found in Figure A.5 of Appendix A.3. We now use a different vector of coefficients

$$\boldsymbol{\beta} = \begin{pmatrix} -3 & 4 & 2 & 1 & -1 & -0.5 & 0.5 & 2.5 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.18}$$

and re-sample the results 2000 times for an again 1000 observations sample size. Figure 6.3 shows the *risk* components for the ridge logistic model and Figure 6.4 plots the *risk* components for the LASSO logistic model.

The results show that *there exists*, in fact, a critical value λ^* which minimizes the bias and variance components of the *risk* function. Both figures show how the higher the shrinkage parameter, the lower the variance and the higher the bias incurred. This illustrates the bias-variance trade-off described in Section 4.1.1. Note that for high values of λ the aggregate *risk* explodes, mainly due to the fact that the bias is squared.

We can also see that the value $\lambda = 0.5$ chosen in the previous experiment would be a fairly good choice here as well, in the sense that it is close to the one yielding the lower *risk*. This is in fact the criterion that can be used to determine the optimal value of the tuning parameter, as we mentioned in Section 4.1.2. It can be seen that for moderately small penalties we can trade a substantial decrease in variance for a very small increase in bias, thus leading to a lower error.

In the particular case of the LASSO regularization, it looks like it may accept slightly larger penalties. This may be due to its variable selection feature, as some of the true parameters were exactly equal to zero and still included in the model. The inclusion of a penalty sets these coefficients to exactly zero, leading to a zero average deviation (bias). In fact, the path of the squared bias in Figure 6.4 is has a smoother slope than in Figure 6.3.

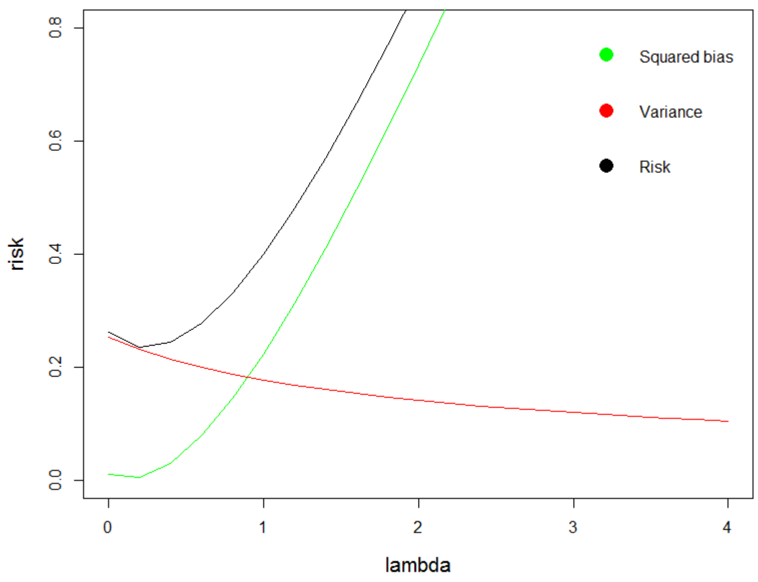

Figure 6.3: *Risk* components for ℓ_2-penalized logistic regression
Source: Own elaboration

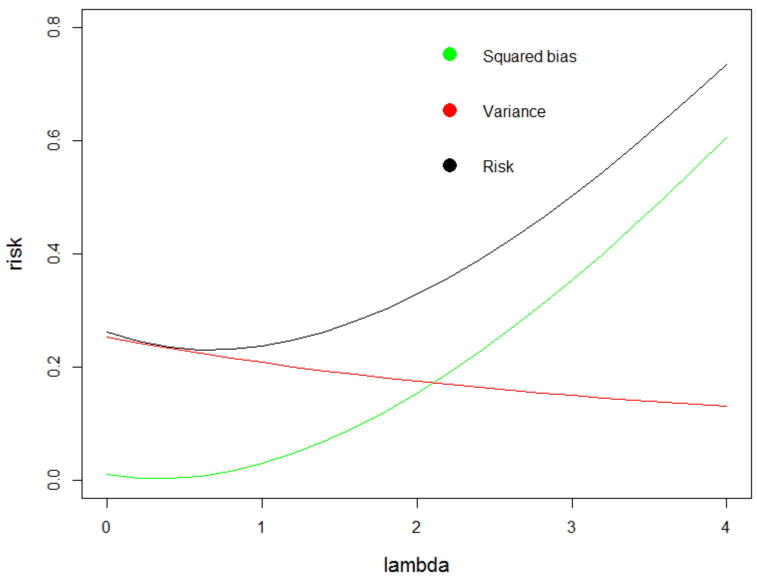

Figure 6.4: *Risk* components for ℓ_1-penalized logistic regressionn
Source: Own elaboration

7. CONCLUSIONS

In this dissertation we have presented a new algorithm to estimate penalized (ridge) logistic regression. In order to do so, the literature of Generalized Linear Models and penalized likelihood approaches is revised. It turns out that some standard results of these fields may be used used to develop an algorithm to fit penalized logistic regression models, departing from the standard Iteratively Reweighted Least Squares approach. As an extension, we have compared through a simulation exercise the results obtained with our estimator against the corresponding LASSO estimators. Particularly, we were concerned with showing how these techniques would trade a little increase in bias for a substantial decrease in variance, thus leading to a lower Mean Squared Error. It turns out that under certain situations, such as the presence of highly correlated variables and the inclusion of many explanatory variables, regularization techniques may outperform the unregularized logistic regression estimators. We have also studied the crucial role that plays in the fitting of these models the so-called shrinkage parameter.

A. R CODE

A.1. CODE FOR ALGORITHM 3

```
ridgeIRLS <- function(X, y, lambda, maxIter=10, tol=1E-6){

  b <- bLast <- rep(0, ncol(X))
  it <- 1
  while (it <= maxIter){

    eta <- X %*% b
    mu <- 1/(1 + exp(-eta))
    w <- as.vector(mu*(1 - mu))
    W <- diag(w)
    z <- eta + (y - mu)/w
    b <- solve(t(X)%*%W%*%X+lambda*diag(ncol(X)))%*%t(X)%*%W%*%z

    if (max(abs(b - bLast)/(abs(bLast) + 0.01*tol)) < tol)
      break

    bLast <- b
    it <- it + 1
  }

  if (it > maxIter) warning('maximum␣iterations␣exceeded')

  list(coefficients=b, iterations=it)
}
```

Figure A.1: A ridge IRLS algorithm
Source: Own elaboration

Figure A.1 contains the R code for Algorithm 3. The arguments included are:

1. The matrix of explanatory variables X.

2. The vector of observations of the dependent variable y.

3. The value of the tuning parameteR $lambda$.

4. The maximum number of iterations $maxIter$, which is set at 10 by default.

5. A tolerance criterion tol, which is set at 1e$-$6 by default.

The vector of estimates b is set at zero at the first iteration, and it is iteratively updated. At each iteration and until the convergence criterion is satisfied or the maximum number of iterations is reached, we compute the linear predictor eta η [7], the probabilities mu μ, and the weights w. Using this information, we construct the working dependent variable z and solve the penalized WLS problem, which results in an estimate of the form given in (4.22).

[7] The probabilities are specified here as $1/1 + exp(-\eta)$, which results from dividing $exp(\eta)/1 + exp(\eta)$ over $exp(\eta)$.

The convergence criterion is given by

$$max\left[\frac{|b - bLast|}{|bLast + 0.01 \times tol|}\right] < tol, \tag{A.1}$$

where $bLast$ refers to the vector of estimates obtained in the previous iteration. Finally, the algorithm is set so as to display the estimated coefficients and the number of iterations needed to reach convergence.

It should be noted that the algorithm is constructed to input a matrix of standardized explanatory variables $\boldsymbol{X}$ and a centered dependent variable $\boldsymbol{y}$. This removes the need for an intercept in the regression model.

Regarding the package *glmnet*, in order to replicate the results it is needed to take into account equation the penalized log-likelihood given by (4.41)

$$\ell_{glmnet}(\beta) = -\frac{1}{n}\ell(\beta) + \lambda\big[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1\big]. \tag{A.2}$$

Note that the unrestricted log-likelihood is divided over the number of observations n, thus the value of λ here needs to be redefined as

$$\lambda = \lambda^*/n, \tag{A.3}$$

where λ^* stands for the tuning parameter used as an argument in our algorithm. In addition, *glmnet* does not center the response variable and unstandardizes the resulting coefficients. To keep consistency with our algorithm, the best solution is to provide standardized explanatory variables and centered response as inputs.

Figure A.2 shows the exact arguments needed to replicate the results of our algorithm.

```
#Our algorithm
ridgeIRLS(X,y,lambda, maxIter = 50)

#Package "glmnet"
coef(glmnet(X,y,family = "binomial",standardize = FALSE,alpha =
   0, lambda = lambda/nrow(X), intercept = FALSE))
```

Figure A.2: Replicating the results of Algorithm 3
Source: Own elaboration

A.2. CODE FOR SECTION 6.2

```
coeffs <- function(seq_lambda, b, n){

  b_unreg <- matrix(nrow = length(b), ncol = length(seq_lambda))
  b_ridge <- matrix(nrow = length(b), ncol = length(seq_lambda))
  b_lasso <- matrix(nrow = length(b), ncol = length(seq_lambda))

  X <- matrix(nrow= n, ncol= length(b))
  u <- double(length(n))
  X[,1] <- runif(n, min = -3, max = 3)
  X[,1] <- scale(X[,1])*sqrt((length(X[,1])/(length(X[,1])-1)))
  for(j in 2:length(b)){
    X[,j] <- 0.5*runif(n, min = -3, max = 3) + 0.5*X[,(j-1)]
    X[,j] <- scale(X[,j])*sqrt((length(X[,j])/(length(X[,j])-1))
       )
  }
  u <- rlogis(n, location = 0, scale = 1)
  y <- X%*%b+u
  y <- scale(y)
  Y <- as.numeric(y>0)
  for(i in 1:length(seq_lambda)){
    b_unreg[,i] <- coef(glmnet(X,Y,family = "binomial",
       standardize = FALSE,alpha = 0, lambda = 0, intercept =
       FALSE))[1:length(b)+1]
    b_ridge[,i] <- coef(glmnet(X,Y,family = "binomial",
       standardize = FALSE,alpha = 0, lambda = seq_lambda[i]/
       nrow(X), intercept = FALSE))[1:length(b)+1]
    b_lasso[,i] <- coef(glmnet(X,Y,family = "binomial",
       standardize = FALSE,alpha = 1, lambda = seq_lambda[i]/
       nrow(X), intercept = FALSE))[1:length(b)+1]

  }
  list(b_unreg=b_unreg, b_ridge=b_ridge, b_lasso=b_lasso)
}
```

Figure A.3: A ridge IRLS algorithm
Source: Own elaboration

Figure A.3 contains the *R* code for Section 6.2. Note that the argument seq_lambda requires a sequence of λ values, while b refers to a vector of true parameters and $sampling$ introduces randomness in the model. It works for whatever experiment you might want to run. The logistic regression model is constructed following the formulation given in Section 6.1 and the variables are internally standardized in the way *glmnet* does. Some degree of correlation is introduced among the covariates as well.

Note that this code includes the *glmnet* package to solve the IRLS problem instead of our algorithm, but this is just to facilitate running the code without previously loading our algorithm.

A.3. CODE FOR SECTION 6.3

```
empRISK <- function(lambda, b, sampling, n){

  b_unreg <- matrix(nrow = length(b), ncol = length(sampling))
  b_ridge <- matrix(nrow = length(b), ncol = length(sampling))
  b_lasso <- matrix(nrow = length(b), ncol = length(sampling))
  bias <- matrix(nrow = length(b), ncol = 3)
  var <- matrix(nrow = length(b), ncol = 3)
  MSE <- matrix(nrow = length(b), ncol = 3)
  sumMSE <- matrix(nrow = 1, ncol = 3)
  for(i in sampling){
    set.seed(i)
    X <- matrix(nrow= n, ncol= length(b))
    u <- double(length(n))
    X[,1] <-runif(n, min = -3, max = 3)
    X[,1] <-scale(X[,1])*sqrt((length(X[,1])/(length(X[,1])-1)))
    for(j in 2:length(b)){
      X[,j] <- 0.5*runif(n, min = -3, max = 3) + 0.5*X[,(j-1)]
      X[,j] <- scale(X[,j])*sqrt((length(X[,j])/(length(X[,j])
         -1)))}
    u <- rlogis(n, location = 0, scale = 1)
    y <- scale(X%*%b+u)
    Y <- as.numeric(y>0)
    b_unreg[,which(sampling == i)] <- coef(glmnet(X,Y,family = "
       binomial",standardize = FALSE,alpha = 0, lambda = 0,
       intercept = FALSE))[1:length(b)+1]
    b_ridge[,which(sampling == i)] <- coef(glmnet(X,Y,family = "
       binomial",standardize = FALSE,alpha = 0, lambda = lambda/
       nrow(X), intercept = FALSE))[1:length(b)+1]
    b_lasso[,which(sampling == i)] <- coef(glmnet(X,Y,family = "
       binomial",standardize = FALSE,alpha = 1, lambda = lambda/
       nrow(X), intercept = FALSE))[1:length(b)+1]
  }
  for(k in 1:length(b)){
    bias[k,1] <- mean(b_unreg[k,])-b[k]
    bias[k,2] <- mean(b_ridge[k,])-b[k]
    bias[k,3] <- mean(b_lasso[k,])-b[k]
    var[k,1] <- mean((b_unreg[k,] - mean(b_unreg[k,]))^2)
    var[k,2] <- mean((b_ridge[k,] - mean(b_ridge[k,]))^2)
    var[k,3] <- mean((b_lasso[k,] - mean(b_lasso[k,]))^2)}
  for(t in 1:3){
    MSE[,t] <- bias[,t]^2+var[,t]
    RISK[t] <- sum(MSE[,t])}
  list(b_unreg=b_unreg, b_ridge=b_ridge, b_lasso=b_lasso, bias=
     bias, var=var, MSE=MSE, RISK=RISK)
}
```

Figure A.4: Empirical *risk* function
Source: Own elaboration

Figure A.4 contains the first part of the R code used in Section 6.3. The arguments of the function require to input a value of λ, a vector of true parameters and to set $sampling$, which is how we introduce randomness in the algorithm. Although we did not make use of it, note that the output includes the matrix of estimated coefficients and allows for the decomposition of MSE into bias and variance.

```
RISK_2 <- function(seq_lambda, b, sampling, n){
  sumbias <- matrix(nrow = length(seq_lambda), ncol = 3)
  sumbias2 <- matrix(nrow = length(seq_lambda), ncol = 3)
  sumvar <- matrix(nrow = length(seq_lambda), ncol = 3)
  sumMSE <- matrix(nrow = length(seq_lambda), ncol = 3)
  for(i in 1:length(seq_lambda)){
    trial <- empRISK(lambda = seq_lambda[i],b,sampling,n)
    for(j in 1:3){
      sumbias[i,j] <- sum(trial$bias[,j])}
    for(w in 1:3){
      sumbias2[i,w] <- sum(trial$bias[,w]^2)}
    for(k in 1:3){
      sumvar[i,k] <- sum(trial$var[,k])}
    for(t in 1:3){
      RISK[i,t] <- sum(trial$MSE[,t])}
  }

  list(sumbias=sumbias, sumbias2=sumbias2, sumvar=sumvar, RISK=
     RISK)
}
```

Figure A.5: *Risk* for a sequence of λ values
Source: Own elaboration

Figure A.5 contains the second part of the *R* code used in Section 6.3. It basically builds on the algorithm of Figure A.4 and outputs the aggregated components of the *risk* function in (6.15). Note that it you can decide whether you want to plot the squred bias or the bias itself.

REFERENCES

COX, D. R. 1958. The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, **20**(2), pp. 215–242. ISSN: 00359246. URL: `http://www.jstor.org/stable/2983890` (visited on 06/16/2017).

DOUGHERTY, C. 2007. *Introduction to Econometrics*. 3rd ed. New York: Oxford University Press. ISBN: 978-0-19-928096-4.

EFRON, B.; HASTIE, T.; JOHNSTONE, I.; TIBSHIRANI, R. 2004. Least angle regression. *The Annals of Statistics*, **32**(2), pp. 407–499. DOI: `10.1214/009053604000000067`.

FLURY, B. 1997. *A First Course in Multivariate Statistics*. 1st ed. New York: Springer-Verlag. ISBN: 978-0-387-98206-9. DOI: `10.1007/978-1-4757-2765-4`.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. 2010. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, **33**(1), pp. 1–22. URL: `http://www.jstatsoft.org/v33/i01/` (visited on 06/25/2017).

GOEMAN, Jelle J. 2010. L1 Penalized Estimation in the Cox Proportional Hazards Model. *Biometrical Journal*, **52**(1), pp. 70–84. ISSN: 1521-4036. DOI: `10.1002/bimj.200900028`. URL: `http://dx.doi.org/10.1002/bimj.200900028` (visited on 06/22/2017).

GREENE, W.H. 2008. *Econometric analysis*. 6th ed. New Jersey: Pearson Prentice Hall. ISBN: 978-0-13-513245-6.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. 2008. *The Elements of Statistical Learning*. 2nd ed. New York: Springer.

HOERL, A.E.; KENNARD, R.W. 1970. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, **12**(1), pp. 55–67. DOI: `10.1080/00401706.1970.10488634`.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. 2013. *An Introduction to Statistical Learning*. New York: Springer. ISBN: 978-1-4614-7137-0.

KENDALL, M.G.; STUART, A. 1967. *The Advanced Theory of Statistics*. 3rd ed. Vol. 2. London: Charles Griffin & Company Limited.

KYUNG, M.; GILL, J.; GHOSH, M.; CASELLA, G. 2010. Penalized regression, standard errors, and Bayesian lassos. *Bayesian Anal.* **5**(2), pp. 369–411. DOI: `10.1214/10-BA607`. URL: `http://dx.doi.org/10.1214/10-BA607` (visited on 06/22/2017).

LE CESSIE, S.; HOUWELINGEN, J.C. Van. 1992. Ridge Estimators in Logistic Regression. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **41**(1), pp. 191–201. ISSN: 00359254, 14679876. URL: `http://www.jstor.org/stable/2347628` (visited on 06/21/2017).

LEE, S.; LEE, H.; ABBEEL, P.; ANDREW, N.Y. 2006. Efficient L1 Regularized Logistic Regression. In: *AAAI*. AAAI Press.

LONG, J.S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: SAGE Publications. ISBN: 0-8039-7374-8.

MCCULLAGH, P.; NELDER, J.A. 1989. *Generalized Linear Models*. 2nd ed. London – New York: Chapman and Hall. ISBN: 9780412317606.

NELDER, J. A.; WEDDERBURN, R. W. M. 1972. Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General)*, **135**(3), pp. 370–384. ISSN: 00359238. URL: `http://www.jstor.org/stable/2344614` (visited on 06/14/2017).

RAO, C.R. 2002. *Linear Statistical Inference and its Applications*. New York: Wiley. ISBN: 978-0-471-21875-3.

RODRÍGUEZ, G. 2007. Lecture Notes on Generalized Linear Models. Unpublished. URL: `http://data.princeton.edu/wws509/notes/` (visited on 06/18/2017).

TIBSHIRANI, R. 1996. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**(1), pp. 267–288. ISSN: 0035-9246. URL: `http://www.jstor.org/stable/2346178` (visited on 06/13/2017).

TIBSHIRANI, R. 2011. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **73**(3), pp. 273–282. ISSN: 1467-9868. DOI: `10.1111/j.1467-9868.2011.00771.x`.

TRAIN, K.E. 2009. *Discrete Choice Models with Simulation*. 2nd ed. New York: Cambridge University Press. ISBN: 978-0-521-76655-5.

WOOLDRIDGE, J.M. 2009. *Introductory Econometrics: A Modern Approach*. 5th ed. Mason, OH: South-Western. ISBN: 978-1-111-53104-1.

YUAN, M.; LIN, Y. 2007. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, **68**(1), pp. 49–67. ISSN: 1467-9868. DOI: `10.1111/j.1467-9868.2005.00532.x`.

ZOU, H.; HASTIE, T. 2005. Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B*, **67**(2), pp. 301–320. ISSN: 13697412, 14679868. URL: `http://www.jstor.org/stable/3647580` (visited on 06/28/2017).

ZOU, Hui. 2006. The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, **101**(476), pp. 1418–1429. DOI: `10.1198/016214506000000735`.