UNIVERSIDADE DE SANTIAGO DE COMPOSTELA
DEPARTAMENTO DE ESTATÍSTICA E INVESTIGACIÓN OPERATIVA

Lasso Logistic Regression, GSoft and the Cyclyc Coordinate Descent Algorithm. Application to Gene Expression Data

M. García-Magariños, A. Antoniadis, R. Cao, W. González-Manteiga

Report 09-01

Reports in Statistics and Operations Research

Lasso Logistic Regression, GSoft and the Cyclic Coordinate Descent Algorithm. Application to Gene Expression Data.

Manuel Garcia–Magariños Anestis Antoniadis* Ricardo Cao
Wenceslao Gonzalez–Manteiga

October 7, 2009

Abstract

Statistical methods generating sparse models are of great value in the gene expression field, where the number of covariates (genes) under study moves about the thousands, while the sample sizes seldom reach a hundred of individuals. For phenotype classification, we propose different lasso logistic regression approaches with specific penalizations for each gene. These methods are based on a generalized soft–threshold (GSoft) estimator. We also show that a recent algorithm for convex optimization, namely the cyclic coordinate descent (CCD) algorithm, provides with a fast way to solve the optimization problem posed in GSoft. Results are obtained for simulated and real data. The leukemia and colon datasets are commonly used to evaluate new statistical approaches, so they come in useful to establish comparisons with similar methods. Furthermore, biological meaning is extracted from the leukemia results, and compared with previous studies. In summary, the approaches presented here give rise to sparse, interpretable models, competitive with similar methods developed in the field.

1 Introduction

Advent of high–dimensional data in several fields (genetics, text categorization, combinatorial chemistry,...) is an outstanding challenge for statistics. Gene expression data is the paradigm of high–dimensionality, usually comprising thousands (p) of covariates (genes) for only a few dozens (n) of samples (individuals). Feature selection in regression and classification is then fundamental to get interpretable, understandable models, which might be of use to the field. First approaches to this problem [19, 20, 32, 46] were based

*Manuel Garcia–Magariños and Anestis Antoniadis contributed equally to this work.

on filtering to select a subset of covariates related with the outcome, usually a binary response. Nevertheless, common methods developed nowadays search for variable selection and classification carried out in the same step. Sparse models are needed to account for high–dimensionality (the $p >> n$ problem) and strong correlations between covariates.

Penalized regression methods have received much attention over the past few years, as a proper way to get sparse models in those fields with large datasets. Lasso [43] was originally proposed for linear regression models, and subsequently adapted to the logistic case [39, 41]. Lasso applies a l_1 penalization that, as opposed to ridge regression [21], gives rise to sparse models, ruling out the influence of most of the covariates on the response. Consistency properties of lasso for the linear regression case have been full well studied [29, 34, 36, 51, 52]. An evolution of lasso that allows for specific penalizations in the l_1 penalty (adaptive lasso) is developed in [53]. Lasso has been also adapted to work with categorical variables [2, 4, 35, 49] and multinomial responses [30]. Other penalized regression methods include bridge estimators [13], which replace the l_1 penalization with l_q penalization, being $0 < q < 1$, and the elastic net [54], that penalizes by means of a linear combination of l_1 and l_2 penalties. Consistency studies about bridge and elastic net can be found in [22] and [6], respectively. Application of both approaches to high–dimensional genetic data is carried out in [33]. Optimization of the lasso log–likelihood function is also an important subject of study [31, 40], as a result of the non–differentiability problems of the l_1 penalty around zero.

In this study, we adopt an adaptive lasso logistic regression approach based on the generalized soft–threshold estimator (GSoft) [28]. A theoretical connection between existence of solution in GSoft and convergence of the cyclic coordinate descent (CCD) algorithm [50] is established, allowing the solutions obtained with the latter to take advantage of the asymptotic properties of the former. We try different vectors Γ for the specific penalization of each covariate (gene) and some consistency results [24] are shown for each one. Extensive comparisons with similar approaches are carried out using simulated and real microarray data.

The rest of this paper is organized as follows: a short introduction about the CCD algorithm, GSoft and some of its asymptotic properties is given in Section 2, together with the theoretical connection between both and the three different Γ choices for the specific penalizations. Some consistency results for each one are added. Results of simulated and real data are shown in Section 3. Simulations include approximations of the variance–covariance matrix for the estimated coefficients. Real data includes leukemia [17] and colon [1] datasets. Finally Section 4 is devoted to conclusions, and the Appendix contains the proof of Theorem 2.

2 Methods

Our aim is to learn a binary gene expression classifier $y_i = f(\mathbf{x}_i)$ from a set $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ of independent and identically distributed observations. In each sample i, the vector

$$\mathbf{x}_i = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{ip} \end{pmatrix} \in \mathbb{R}^p \tag{1}$$

comprises gene expression measurements. The $n \times p$ design matrix is then $X = (\mathbf{x}_j, j \in \{1, \ldots, p\})$ where the $\mathbf{x}_j$'s represent the expression measurements of gene j along the entire set of samples. The vector of binary responses

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \tag{2}$$

informs about membership (+1) or nonmembership (-1) of the sample to the category. The logistic regression model with vector of regression coefficients $\boldsymbol{\beta} \in \mathbb{R}^p$ assumes that

$$P(y_i = 1|\mathbf{x}_i) = \frac{1}{1 + \exp(-\mathbf{x}_i^{'}\boldsymbol{\beta})}. \tag{3}$$

Adopting a generalized linear model framework, the associated linear predictor $\boldsymbol{\eta}$ is defined as

$$\boldsymbol{\eta} = X\boldsymbol{\beta} = \begin{pmatrix} \mathbf{x}_1^{'}\boldsymbol{\beta} \\ \vdots \\ \mathbf{x}_n^{'}\boldsymbol{\beta}, \end{pmatrix} \text{ where } X = \begin{pmatrix} \mathbf{x}_1^{'} \\ \vdots \\ \mathbf{x}_n^{'} \end{pmatrix} \text{ and } \boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}. \tag{4}$$

The decision of whether to assign the i sample to the category or not is usually accomplished by comparing the probability estimate with a threshold (e.g. 0.5). Consequently, minus the log–likelihood function is

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{n} \ln \left[1 + \exp(-y_i \mathbf{x}_i^{'}\boldsymbol{\beta})\right] \tag{5}$$

The lasso like logistic estimator $\hat{\boldsymbol{\beta}}$ with specific penalizations for each covariate is then given by the minimizer of the function

$$L_1(\boldsymbol{\beta}) = L(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} \gamma_j |\beta_j| \tag{6}$$

where λ is a common nonnegative penalty parameter and the vector $\boldsymbol{\Gamma} = (\gamma_1, \ldots, \gamma_p)$ with nonnegative entries penalizes each coefficient. The standard lasso regularization [43] takes $\gamma_j = 1 \ \forall j$. Minimization of these objective functions makes use of their derivatives. We refer to the gradient of $L(\boldsymbol{\beta})$ as the score vector whose components are defined by:

$$s_j(\boldsymbol{\beta}) = \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_j} \tag{7}$$

The negative Hessian with respect to the linear predictor η is defined as

$$H(\eta) = -\frac{\partial^2 L(\eta)}{\partial \eta \partial \eta'} \tag{8}$$

The basic requirement for the weights γ_j is that their value should be large enough to get $\hat{\beta}_j = 0$ if the true value β_j is zero, and small otherwise. Obtaining of a sparse, interpretable model is of paramount importance in those areas where the number of variables usually outperforms the sample size ($p >> n$ problem). The choice of the $\boldsymbol{\Gamma}$ vector is therefore essential to get an accurate estimator $\hat{\boldsymbol{\beta}}$.

2.1 Cyclic coordinate descent (CCD) algorithm

The choice of a proper algorithm to solve the minimization of (6) is a main issue, as it needs to be capable of dealing with the problem of non–differentiability of the absolute value function around zero. Furthermore, efficiency of the algorithm is fundamental, given the high–dimensionality of the problems at hand.

A number of different algorithms have been developed to obtain the optimum for the objective function. In [16] a "Split-Bregman" method is applied to solve L_1-regularized problems, while in [47] an algorithmic framework for minimizing the sum of a smooth convex function with a nonsmooth nonconvex one is proposed. A similar algorithm is used in [27] to obtain the solution for the SCAD estimator in high–dimensions. Two new approaches are developed in [40], together with a comparative study. An efficient algorithm is carried out in [31], using LARS [11] in each iteration. A local linear approximation (LLA) algorithm was recently proposed by [55], while [45] developed a method of least squares approximation (LSA) for lasso estimation, making use of the LARS algorithm.

Finding the estimate of $\boldsymbol{\beta}$ is a convex optimization problem. The cyclic coordinate descent algorithm is based on the CLG algorithm of Zhang and Oles [50]. An exhaustive description of the algorithm is beyond the scope of this paper, and interested readers are referred to the detailed description in [15]. The basis of all cyclic coordinate descent algorithms is to optimize with respect to only one variable at the time while all others are held constant. When this one–dimensional optimization problem has been solved, optimization is performed with respect to the next variable, and so on. When the procedure has gone through all variables it starts all over with the first one again, and the iterations proceed in this manner until some pre–defined convergence criterion is met. The one–dimensional optimization problem is to find β_j^{new}, the value for the j–th parameter that maximizes the penalized log–likelihood assuming that all other β_j's are held constant. In the end, the update equation for β_j becomes

$$\beta_j^{new} = \begin{cases} \beta_j - \Delta_j & \text{if } \Delta v_j < -\Delta_j \\ \beta_j + \Delta v_j & \text{if } -\Delta_j \leq \Delta v_j < \Delta_j \\ \beta_j + \Delta_j & \text{if } \Delta_j < \Delta v_j \end{cases}$$

where the interval $(\beta_j - \Delta_j, \beta_j + \Delta_j)$ is an iteratively adapted trust region for the suggested update Δv_j. The width of this interval is determined based on its previous value and the previous update made to β_j. The suggested update is given by

$$\Delta v_j = -\frac{s_j(\boldsymbol{\beta}) - \lambda\gamma_j \text{sign}(\beta_j)}{Q(\beta_j, \Delta_j)} \tag{9}$$

The essential idea in CCD is $Q(\beta_j, \Delta_j)$ to be an upper bound on the second derivative of $L_1(\boldsymbol{\beta})$ in the interval around β_j:

$$\frac{\partial^2 L_1(\boldsymbol{\beta})}{\partial \beta_j^2} = \sum_{i=1}^{n} \frac{x_{ij}^2 \exp(-y_i \mathbf{x}_i' \boldsymbol{\beta})}{\left[1 + \exp(-y_i \mathbf{x}_i' \boldsymbol{\beta})\right]^2} \tag{10}$$

The function $Q(\beta_j, \Delta_j)$ is given by the expression:

$$Q(\beta_j, \Delta_j) = \sum_{i=1}^{n} x_{ij}^2 F(y_i \mathbf{x}_i' \boldsymbol{\beta}, \Delta_j x_{ij}) \tag{11}$$

with the function F being defined by

$$F(B, \delta) = \begin{cases} 0.25 & \text{if } |B| \leq |\delta| \\ \left[2 + \exp(|B| - |\delta|) + \exp(|\delta| - |B|)\right]^{-1} & \text{otherwise.} \end{cases} \tag{12}$$

A proof of Q being an upper bound in the aforementioned interval is straightforward. Advantages of CCD can be summarized in efficiency of the algorithm, stability and ease of implementation. Efficiency is due to several factors: CCD works following a cycling procedure along the coefficients. From a certain iteration, CCD only visits the active set, reducing considerably its computational demands. Implementation has been carried out by means of the R package *glmnet*. This approach is explained in [14], where it is proved to be faster than its competitors.

2.2 GSoft

The generalized soft–threshold estimator or GSoft [28] is claimed to be a compromise between approximately linear estimators and variable selection strategies for high dimensional problems. Our interest in GSoft lies in the fact that once a solution $\boldsymbol{\beta}$ exists, a bunch of asymptotic properties can be derived. The next theorem from [28] establishes necessary and sufficient conditions for the existence of such solution.

Theorem 1. *The following set of conditions is necessary and sufficient for the existence of an optimum* $\hat{\boldsymbol{\beta}}$ *of* $L_1(\boldsymbol{\beta})$

(a)

$$\left\{ \begin{array}{ll} |s_j(\beta)| \leq \lambda\gamma_j & \textit{if } \beta_j = 0 \\ s_j(\beta) = \lambda\gamma_j & \textit{if } \beta_j > 0 \\ s_j(\beta) = -\lambda\gamma_j & \textit{if } \beta_j < 0 \end{array} \right\} \tag{13}$$

(b)

$$X_\lambda^{'} H(\eta) X_\lambda \textit{ is positive definite,} \tag{14}$$

where X_λ *retains only those columns (covariates)* $\mathbf{x}_j$ *of* X *fulfilling* $|s_j(\boldsymbol{\beta})| = \lambda\gamma_j$*, that is,* $X_\lambda = (\mathbf{x}_j, |s_j(\boldsymbol{\beta})| = \lambda\gamma_j)$.

2.2.1 Approximation of the covariance matrix for the estimated coefficients.

Approximations to the variance–covariance matrix of $\hat{\boldsymbol{\beta}}$ have to deal with the non–differentiability problem of the penalization term around $|\beta_j| = 0$. This fact is solved by taking a differentiable approximation $a(\beta_j, \delta)$ to the absolute value function, obtained by smoothing it around zero

$$a(\beta_j, \delta) = \left\{ \begin{array}{ll} |\beta_j| & \text{if } |\beta_j| > \delta \\ \frac{(\beta_j^2 + \delta^2)}{2\delta} & \text{if } |\beta_j| \leq \delta \end{array} \right. , \tag{15}$$

with $\delta > 0$ and satisfying $\lim_{\delta \to 0} a(\beta_j, \delta) = |\beta_j|$.

So an approximation can be constructed from the well–known sandwich form developed in [25]

$$V_\delta(\hat{\boldsymbol{\beta}}) = \left\{H(\hat{\boldsymbol{\beta}}) + \lambda\boldsymbol{\Gamma}G\left(\hat{\boldsymbol{\beta}},\delta\right)\right\}^{-1} \mathrm{Var}\left\{s(\hat{\boldsymbol{\beta}})\right\} \left\{H(\hat{\boldsymbol{\beta}}) + \lambda\boldsymbol{\Gamma}G\left(\hat{\boldsymbol{\beta}},\delta\right))\right\}^{-1} \tag{16}$$

where $H(\hat{\boldsymbol{\beta}})$ is the negative Hessian of L but now as a function of $\hat{\boldsymbol{\beta}}$ and G is the diagonal matrix made up of the second derivatives of the approximations $a(\beta_j,\delta)$:

$$G(\boldsymbol{\beta},\delta) = \mathrm{diag}\left(\frac{I\left\{|\beta_1| \le \delta\right\}}{\delta},\ldots,\frac{I\left\{|\beta_p| \le \delta\right\}}{\delta}\right) \tag{17}$$

In these conditions it is clear that, when $\delta \to 0$, the diagonal elements of the matrix $G(\hat{\boldsymbol{\beta}},\delta)$ corresponding to $\beta_j = 0$ tend to ∞, making the covariance matrix $V_\delta(\hat{\boldsymbol{\beta}})$ become singular in the limit. So regularity conditions of the asymptotic theory are not fulfilled with GSoft when any of the coefficients take the value zero. This is a major concern, since it is just one of the desirable characteristics in a proper variable selection method.

GSoft solves this problem developing an estimator of the covariance matrix that smooths the discontinuity in $G(\hat{\boldsymbol{\beta}},\delta)$ when $\delta \to 0$ by means of approximating using the expectation of G and a continuous variable (e.g. normal) with mean in $\hat{\boldsymbol{\beta}}$. The estimator is

$$\hat{V}(\hat{\beta}_j) = \left\{H(\hat{\boldsymbol{\beta}}) + \lambda\boldsymbol{\Gamma}G^*\left(\hat{\boldsymbol{\beta}},\hat{\sigma}\right)\right\}^{-1} \hat{F}(\hat{\boldsymbol{\beta}}) \left\{H(\hat{\boldsymbol{\beta}}) + \lambda\boldsymbol{\Gamma}G^*\left(\hat{\boldsymbol{\beta}},\hat{\sigma}\right)\right\}^{-1} \tag{18}$$

where

$$\begin{array}{l} G^*\left(\hat{\boldsymbol{\beta}},\sigma\right) = \mathrm{diag}\left\{\frac{2}{\sigma_1}\varphi(\hat{\beta}_1/\sigma_1),\ldots,\frac{2}{\sigma_p}\varphi(\hat{\beta}_p/\sigma_p)\right\} \\ \left(\hat{\sigma}_1^2,\ldots,\hat{\sigma}_p^2\right) = \mathrm{diag}\left[H(\hat{\boldsymbol{\beta}})^{-1}\hat{F}(\hat{\boldsymbol{\beta}})H(\hat{\boldsymbol{\beta}})^{-1}\right] \\ \varphi \text{ density function of the normal distribution} \end{array} \tag{19}$$

Anyhow, the main point to get a well established approach to the real variance–covariance matrix is to use an accurate estimator $\hat{F}$ of the Fisher matrix given by

$$F(\eta) = -E\left\{\frac{\partial L(\eta)}{\partial\eta\partial\eta'}\right\} \tag{20}$$

Firstly, we made use of the approach carried out in [3]. Nevertheless, after some tests we realized that such a choice really underestimates the true

variance–covariance values. Our solution consists of rescaling this matrix multiplying it by a factor equal to the number p of variables in the model. So

$$\hat{F}(\hat{\boldsymbol{\beta}}) = I(\hat{\boldsymbol{\beta}}) = \frac{p[\partial^2 L(\hat{\boldsymbol{\beta}})/\partial\beta_i\beta_j]}{n} \tag{21}$$

Goodness–of–fit for this estimator is discussed in the results section.

2.3 Connection GSoft - CCD algorithm

The main aim of this article is to establish a theoretical connection between the convergence of the CCD algorithm and the existence of an optimum for the objective function with GSoft. This theoretical connection is established by the next theorem (proof in the Appendix).

Theorem 2. *The following two statements are equivalent:*
(1) The CCD algorithm for the lasso case converges.
(2) An optimum for the objective function under the terms of the theorem in [28] exists.

In this way, positive results of convergence obtained with the CCD algorithm can take advantage of the asymptotic properties of GSoft. Similarly, solutions obtained with GSoft are consistent in the way proved in [35].

2.3.1 Choice of $\boldsymbol{\Gamma}$

As we mentioned above, we use a global threshold λ together with a vector of specific thresholds $\boldsymbol{\Gamma} = (\gamma_1, \ldots, \gamma_p)$ corresponding to the coefficients $\beta_1, \ldots, \beta_p$ of each variable in the model. In this study, we will evaluate the performance of three different choices for the Γ vector:

1. $\gamma_j = \sqrt{\text{var}(\mathbf{x}_j)}$. This is one of the choices carried out in [28]. As a consequence, we will refer to it as γ–Klinger. Adjusting the thresholds like this is equivalent to standardization.

2. $\gamma_j = \frac{1}{|\beta_j^{\text{ridge}}|}$. Ridge logistic regression was performed on data with a small global threshold λ_0, obtaining coefficients $\beta_j^{\text{ridge}} \neq 0$, $\forall j = 1, \ldots, p$. This choice is related to penalize according to the importance of the variable in ridge, and it is based on a special case of the adaptive lasso [53]. This choice will be designated as γ–ridge.

3. $\gamma_j = \frac{1}{|\beta_j^{\text{lasso}}|}$. Lasso logistic regression was performed on data with a small global threshold λ_0 and without using specific thresholds γ.

Obviously, some coefficients β_j^{lasso} will take zero values. In this case, these variables are excluded from the final model, which is equivalent to take $\gamma_j = \infty$. It will be named as γ–lasso.

2.3.2 Consistency results.

Variable selection consistency results in lasso can be found in the recent related literature. Oracle property [12] for the adaptive lasso in linear regression models is proved in [23]. Consistency results shown here are based on the subsequent adaptation of these results to the logistic case, carried out in [24], for the γ–lasso, there called iterated lasso.

The number of covariates p will be taken as a function of sample size, so the notation p_n will be used. For a set of indices $B \subseteq \{1, \ldots, p_n\}$ we consider $X_B = (\mathbf{x}_j, j \in B)$ and $C_B = X_B^{'} X_B / n$. From them we define:

$$\underline{c}(m) \quad = \min_{|B|=m} \min_{\|v\|=1} v^{'} C_B v \tag{22}$$

$$\overline{c}(m) \quad = \max_{|B|=m} \max_{\|v\|=1} v^{'} C_B v \tag{23}$$

The Sparse–Riesz Condition (SRC) [51] is satisfied by the covariance matrix X with rank q and spectrum bounds $0 < c_* < c^* < \infty$ if

$$c_* < \underline{c}(q) < \overline{c}(q) < c^* \tag{24}$$

Let us take the subset of indices with true nonzero coefficients $B_0 = \{j, \beta_j \neq 0\}$. Let $k_n = |B_0|$ and $m_n = p_n - k_n$ be the number of nonzero and zero coefficients, respectively, and $b_{n1} = \min_{j \in B_0} |\beta_j|$, $b_{n2} = \max_{j \in B_0} |\beta_j|$ the minimum and the maximum of the true nonzero coefficients. Let us assume the following conditions:

(i) Bounds for the true coefficients and the covariates:

  (i1) For some constant $0 < b < \infty$, it is fulfilled that $b_{n2} < b$.

  (i2) For some constant $M > 0$, it is fulfilled that $|x_{ij}| < M$ for all $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, p_n\}$.

(ii) The design matrix X satisfies the SRC with bounds $\{c_*, c^*\}$ and rank $q_n = M_1 n^2 / \lambda_0^2$ being M_1 a positive constant.

(iii) When $n \to \infty$, the following convergence is satisfied

$$\frac{\sqrt{\ln\, k_n}}{b_{n1}\sqrt{n}} + \frac{\sqrt{n \ln\, m_n}}{\lambda r_n} + \frac{\lambda \sqrt{k_n}}{n b_{n1}} \to 0 \tag{25}$$

where r_n is the order of consistency at zero [24] of the primary lasso estimator.

Under (i)–(iii) it has been proved that

$$P(\text{sign}(\hat{\boldsymbol{\beta}}) = \text{sign}(\boldsymbol{\beta})) \to 1 \tag{26}$$

where the sign function is now taken in a slightly different way than in (9): $\text{sign}(\theta_1, \ldots, \theta_p) = (\text{sign}(\theta_1), \ldots, \text{sign}(\theta_p))$ and

$$\text{sign}(t) = \begin{cases} -1 & \text{if } t < 0 \\ 0 & \text{if } t = 0 \\ 1 & \text{if } t > 0 \end{cases}$$

so nonzero coefficients are correctly selected with γ–lasso with probability converging to one. From the same assumptions a result for the asymptotic distribution of the estimated nonzero coefficients of $\hat{\boldsymbol{\beta}}$ wih respect to the true ones $\boldsymbol{\beta}$ can be constructed. The following definitions are needed:

$$\boldsymbol{\beta}_{B_0} = (\beta_j, j \in B_0)^{'} \tag{27}$$

$$\hat{\boldsymbol{\beta}}_{B_0} = (\hat{\beta}_j, j \in B_0)^{'} \tag{28}$$

$$\mathbf{x}_{iB_0} = (x_{ij}, j \in B_0)^{'} \tag{29}$$

$$\epsilon = (\epsilon_1, \ldots, \epsilon_n)^{'} \tag{30}$$

$$\Sigma_{B_0} = \frac{1}{n} X_{B_0}^{'} D X_{B_0} \tag{31}$$

where $\epsilon_i = y_i - (2P(y_i = 1|\mathbf{x}_i) - 1)$ and D is the diagonal matrix composed by the products of the logistic probabilities of case and control in each individual sample. Then, for $s_n^2 = \sigma^2 \alpha^{'} \Sigma_{B_0}^{-1} \alpha$ with α any vector of length k_n fulfilling $\|\alpha\| \leq 1$, the following asymptotic property is satisfied by logistic lasso estimators $\hat{\boldsymbol{\beta}}$ with the γ–lasso choice:

$$\frac{\sqrt{n}}{s_n} \alpha^{'} (\hat{\boldsymbol{\beta}}_{B_0} - \boldsymbol{\beta}_{B_0}) = \frac{\sum_{i=1}^{n} \epsilon_i \alpha^{'} \Sigma_{B_0}^{-1} \mathbf{x}_{iB_0}}{\sqrt{n} s_n} + o_p(1) \to_D N(0, 1) \tag{32}$$

whenever $\frac{\lambda \sqrt{k_n}}{\sqrt{n}} \to 0$.

These two results, (26) and (32), together mean the γ–lasso choice has the asymptotic oracle property. The proof can be found in [24], which also refers to the proof for the linear case in [23]. A careful study of both proofs is enough to realize that only minor changes in the assumptions have to be applied to transfer the oracle property to the γ–ridge choice of specific penalizations.

When γ–Klinger penalizations are selected, this is equivalent to standardization, as proved in [28]. Therefore, only usual consistency lasso results [24, 35] can be proved in this case, and oracle property does not hold. An upper bound for the number of estimated nonzero coefficients in lasso is given in [24]. There, it is proved that the dimension of the model selected by lasso is directly proportional to n^2 and inversely proportional to λ.

3 Results.

3.1 Simulated data.

We have simulated two scenarios with binary response according to one of the examples in [26]. In both of them, the response follows:

$$P(y=1|\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{x}'\boldsymbol{\beta})}$$

and the complementary probability for $y=-1$. This example has been adapted to two specifical scenarios carried out in [45] (Simulation 1) and [55] (Simulation 2), with the aim of comparing our results with those obtained there. Furthermore, a third bunch of simulations have been developed following [26]. We have also used the scenario in [55] to obtain the results of approximation of variance as explained in the last section.

3.1.1 Simulation 1.

Our aim is to compare our results with those obtained with the least squares approximation (LSA) estimator. Comparisons with the results of the Park and Hastie (PH) algorithm of [38] shown in [45] are also established. The model is 9–dimensional with coefficients $\boldsymbol{\beta} = (3,0,0,1.5,0,0,2,0,0)'$. The components of $\mathbf{x}_i$ are standard normal and the correlation between each pair of variables $\mathbf{x}_{j_1}$ and $\mathbf{x}_{j_2}$ is fixed to $0.5^{|j_1-j_2|}$. The sizes of the training samples are $n=200$ and $n=400$, and 500 simulation replications have been obtained each time. The BIC criterion is used to obtain the best solution for LSA and PH, while for the choice of λ in our models, we follow a slightly different approach. As choosing the λ giving rise to the smallest error rate (ER) does not necessarily produce a sparse model, we take the largest λ having an error rate smaller than $\min_\lambda ER + 2*\mathrm{sd}(ER)$. Results are shown in Table 1. From now on, lasso logistic regression will be referred with the abbreviation LLR.

The different estimators are compared in terms of model size (MS) and percentage of correct models identified (CM). Unlike [45], here we will not use the relative model error as a comparative measure, since it puts too much weight to the model error without penalty. Besides, in problems involving large amounts of noise, detection of the variables associated with the response is much more important than precise estimation of the true coefficients. Results obtained with our models are slightly better than those in [45], despite improvement of the results of LSA and PH seemed to be highly difficult. Comparisons between the different choices for the Γ vector are favorable to γ–ridge and γ–lasso, as the γ–Klinger seems to be more imprecise than those two regarding detection of the correct model. This

Sample size	Estimation Method	MS Mean	(SE)	CM Mean	(SE)
200	LLR γ–Klinger	3.266	(0.025)	0.762	(0.019)
	LLR γ–ridge	2.896	(0.025)	0.812	(0.017)
	LLR γ–lasso	2.96	(0.028)	0.798	(0.018)
	LSA	3.178	(0.026)	0.798	(0.018)
	PH	3.272	(0.033)	0.716	(0.020)
400	LLR γ–Klinger	3.046	(0.011)	0.956	(0.009)
	LLR γ–ridge	2.964	(0.021)	0.860	(0.016)
	LLR γ–lasso	2.982	(0.022)	0.902	(0.013)
	LSA	3.130	(0.018)	0.888	(0.014)
	PH	3.092	(0.023)	0.846	(0.016)

Table 1: *True model detection results. Comparison between our models and those in [45] is established in the same terms as there.*

imprecision grows when sample size decreases, until reaching the standard of LSA and PH.

3.1.2 Simulation 2.

Comparisons with the one–step sparse estimates developed in [55] are carried out, along with the SCAD and the other variable selection models used there. The second model is 12–dimensional with $\boldsymbol{\beta} = (3, 1.5, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0)^{'}$, while $\mathbf{x}$ is obtained as in Simulation 1, but with one important difference: variables with even index are translated to binary according to their sign. Size of the training sample is $n = 200$ and 1000 replicated datasets were obtained. Choice of the optimal λ for our models is carried out in a similar way to Simulation 1, but taking the largest λ having an error rate smaller than $\min_{\lambda} ER + 0.2 * \mathrm{sd}(ER)$. Results are shown in Table 2.

Same terms as in [55] are used: columns 'C' and 'IC' measure the average number of nonzero coefficients correctly estimated to be nonzero and

Method	C	IC	Proportion of Under-fit	Correct-fit	Over-fit
LLR γ–Klinger	2.84	1.68	0.16	0.14	0.70
LLR γ–ridge	2.77	0.82	0.22	0.40	0.37
LLR γ–lasso	2.71	0.71	0.29	0.40	0.31
one-step SCAD	2.95	0.82	0.051	0.565	0.384
one-step LOG	2.97	0.61	0.029	0.518	0.453
one-step $L_{0.01}$	2.97	0.61	0.028	0.516	0.456
SCAD	2.92	0.51	0.076	0.706	0.218
P-SCAD	2.92	0.5	0.079	0.707	0.214
AIC	2.98	1.56	0.021	0.216	0.763
BIC	2.95	0.22	0.053	0.800	0.147

Table 2: *True model detection results. Comparison between our models and those in [55] is established in the same terms as there.*

the average number of zero coefficients incorrectly estimated to be nonzero, respectively; "Under–fit" and "Over–fit" show the proportion of models excluding any nonzero coefficients and including any zero coefficients throughout the 1000 replications, respectively. "Correct–fit" shows the proportion of correct models obtained.

Our methods show a worse behaviour than those in [55]. After some tests (results not shown) we realized that the reason was that they suffer a lot from the presence of binary variables. This is not a major concern, since our aim was to apply these methods to gene expression data, where all the variables move in a continuous way. Therefore, with the intention of testing them in a continuous environment, conditions in [26] were replicated. These conditions are the same as in Simulation 1 but the correlation between variables is now fixed to $\rho = 0.25$ and $\rho = 0.75$. Sample size was also fixed to $n = 200$. Results are shown in Table 3.

	$\rho = 0.25$		$\rho = 0.75$	
Method	C	I	C	I
LLR γ–Klinger	5.96	0.034	5.562	0.326
LLR γ–ridge	5.9	0.166	5.912	0.778
LLR γ–lasso	5.9	0.176	5.916	0.76
New	5.922	0	5.534	0.222
LQA	5.728	0	4.97	0.090
BIC	5.86	0	5.796	0.304
AIC	4.93	0	4.86	0.092

Table 3: *True model detection results. Comparison between our models and those in [26] is established in the same terms as there.*

Optimal λ is chosen as in Simulation 1. "C" and "I" measure the average number of coefficients correctly and incorrectly set to zero, respectively. Comparisons are made with a new proposed algorithm in [26], a local quadratic approximation (LQA) algorithm developed in [12] and best subset variable selection using BIC and AIC scores. Competitive results are obtained with respect to the procedure in [26]. The best variable selection is obtained using BIC. The results obtained with the γ–Klinger are similar to the ones with γ–ridge and γ–lasso.

3.1.3 Approximation of variance.

Covariance matrix estimation for the estimated coefficients have been obtained according to the approach previously explained. The same model as in Simulation 2 has been used, without the translation to binary (for simplicity). In Figure 1 the behaviour of variance estimation for $\beta_1 = 3$, $\beta_2 = 1.5$ and $\beta_3 = 0$, respectively, is shown in comparison with the true variance, as a function of λ. The estimation, obtained as the median on 1000 replications, fits almost perfectly to the variance except for small devi-

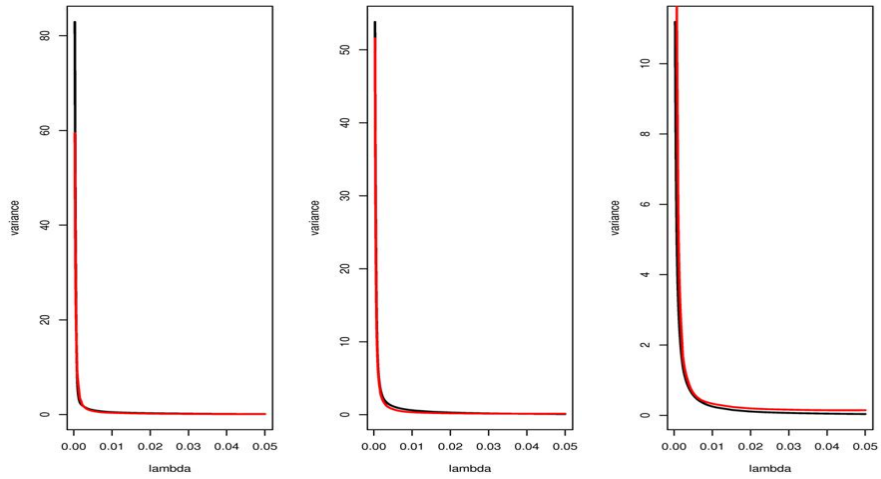

Figure 1: *Variance estimation (in red) for the estimated values of* β_1, β_2 *and* β_3 *in Simulation 2 according to the estimator* (18) *with* $\hat{F}$ *taken as in* (21). *True variance (in black) was approximated by means of recursive simulation–estimation. Variance is displayed as a function of the penalty parameter* λ.

ations when λ approachs zero (maximum likelihood estimator), as the true variance increases enormously.

3.2 Real data.

The leukemia dataset [17] has been used on countless occasions through the gene expression literature. It comprises gene expression data for 72 bone marrow and peripheral blood samples (47 cases of acute lymphoblastic leukemia (ALL) and 25 cases of acute myeloid leukemia (AML)) in 7129 genes. Initially [17] the total sample was divided into a training sample (38 bone marrow samples) and a test sample (34 bone marrow and peripheral blood samples).

The colon dataset was analyzed initially by [1]. As leukemia, it is another commonly used dataset in genomic studies. A number of 62 samples (40 tumors and 22 controls) were measured in 2000 human genes. Absolute measurements from Affymetrix high–density oligonucleotide arrays were taken for each sample in each gene in both datasets. Here, we have worked with data in two different ways. On one side, we have carried out preprocessing steps (P) following [10], (i) thresholding of the measurements, (ii) filtering of genes, (iii) base 10 logarithmic transformation. On the other, we have also tried our models over the raw data (RD). With preprocessing, leukemia and colon datasets reduce their dimensionality to 3571 and 1225 genes, respectively.

Leukemia	Test error	SD	Genes
RD-γ Klinger	0.062	(0.044)	67 (of 7129)
RD-γ Zou	0.064	(0.039)	11 (of 7129)
RD-γ Lasso	0.102	(0.055)	6 (of 7129)
P-γ Klinger	0.079	(0.032)	16 (of 3571)
P-γ Zou	0.067	(0.030)	5 (of 3571)
P-γ Lasso	0.064	(0.028)	5 (of 3571)

Table 4: *Test error and sparsity results for the leukemia dataset.*

As a result of combining these two ways to deal with data with the three different choices for γ, we have six different models. Table 4 shows the results for the leukemia dataset. To obtain accurate and precise measures for the error and its standard deviation, we split 50 times the set of 72 samples into a training set of 38 samples and a test set of 34 samples. We also record the number of genes with non-zero coefficient for the optimal lambda, in terms of cross–validation (CV) error.

Table 5 shows the results for the colon dataset. The 62–sample has been randomly splitted 50 times into a training subsample of 50 observations and a test subsample of 12 observations.

When looking for other error test results obtained with different methods, it is common and correct to think that leukemia and colon datasets have been often used in the scientific literature since its appearance years ago. Nevertheless, it is difficult to find a fair comparison between methods, since each author uses a different way to obtain an error measure. Some of them only focus on a leave–one–out cross–validation rate (too optimistic); others center on the same data subdivision carried out by [17]; finally, the fairest way to know the real performance of each method is to randomly split the total sample N times into two disjoint samples, training and test. Table 6 compare our best results with those from methods obtaining their error rate following the latter way.

Comparisons with the following methods have been established. In [5], a CART-based method is developed to discover the emerging patterns inside the set of variables. BagBoosting [8] is a combination of bagging and boosting, two ensemble learning algorithms, applied to stumps, decision trees with only one split and two terminal nodes. Different algorithms are presented

Colon	Test error	SD	Genes
RD-γ Klinger	0.195	(0.130)	10 (of 2000)
RD-γ Zou	0.147	(0.116)	17 (of 2000)
RD-γ Lasso	0.200	(0.128)	9 (of 2000)
P-γ Klinger	0.152	(0.096)	11 (of 1225)
P-γ Zou	0.182	(0.111)	15 (of 1225)
P-γ Lasso	0.215	(0.133)	10 (of 1225)

Table 5: *Test error and sparsity results for the colon dataset.*

Dataset	Method	Test error
Leukemia	Our best	0.062
	CART-Fisher [5] (*)	0.024–0.050
	BagBoosting [8] (**)	0.0408
	Pelora [9] (**)	0.0569
	Wilma [9] (**)	0.0262
	Forsela [9] (**)	0.0415
	PLS [37] (***)	0.033–0.047
	PCA [37] (***)	0.039–0.108
Colon	Our best	0.147
	CART-Fisher [5] (*)	0.128–0.234
	BagBoosting [8] (**)	0.161
	Pelora [9] (**)	0.1571
	Wilma [9] (**)	0.1648
	Forsela [9] (**)	0.1381

Table 6: *Test error rates obtained using different methods from the scientific literature for the leukemia and colon datasets. (*) In each random split, 10 observations in the test set. (**) In each random split, 2/3 of the data to the training set, 1/3 of the data to the test set. (***) In each random split, 1/2 of the data to the training set, 1/2 of the data to the test set.*

in [9]. *Pelora* is a penalized logistic regression method. *Forsela* is similar to *Pelora*, but making a search of single genes instead of groups, *Wilma* [7] shares some characteristics with *Pelora*, but suffers from a few limitations [9]. [37] uses dimension reduction through partial least squares (PLS) and principal component analysis (PCA), classifying with discriminant analysis. Our error results are only slightly worse than the others for the leukemia dataset, and among the best for colon. In any case, all the error rates are quite similar. Many of the methods we compare with stand out for grouping genes ([5], [37], *Pelora* and *Wilma* in [9]) in one way or another. Gene preselection is carried out by means of preexisting methods in [5] and [8]. Our logistic lasso methods neither makes use of grouping or gene preselection nor it is necessary to select a lot of different parameters, as in [5], appart from the penalty λ. Moreover, its sparsity (see Tables 4 and 5) and the interpretability associated with it are merits not fulfilled by these other methods.

Gene expression data is seen as the paradigm of the case $n << p$, as Affymetrix or oligonucleotide arrays map large parts of the human genome while only tens or hundreds of individuals are sampled. This situation makes most of traditional statistical methods inapplicable, so new variable selection approaches had to be developed to deal with this *curse of dimensionality* problem. Lasso selects a group of $p^{'} \leq n$ genes with high importance in the classification of samples, and assign a zero coefficient to the rest. Use of the CCD algorithm to solve the optimization problem is highly desirable, as it provides with the global solution of GSoft in the fastest way.

In a more biological way, we have also studied which genes are more related with the ALL/AML status in leukemia. Observations of the genes with nonzero coefficients for each model have been carried out. As expected, some recurrences have been found in the six different models. Table 7 shows those genes appearing more frequently.

The fact that some genes are discovered in some models and not in others can be explained from the correlations between them. These correlations arise as a result of co–inheritance of nearby genes throughout generations. For instance, gene M19507 takes a nonzero coefficient with all but two of the models, and gene M92287 takes nonzero coefficients only in these two models. If we take a careful look to the correlation between them, we detect it as abnormally high. A correlation study between all the genes with nonzero coefficient in any of the models has been carried out. With the aim of knowing the real significance of each correlation value, we have obtained a significance value as the proportion of values, in a set of 10000 random correlations between pairs of genes from the entire dataset, higher than the correlation. This way, significance of the correlation M19507–M92287 is 0.0558; the one between M84526–Y00787 is 0.048, which explains why they are partly complementary. Significances of correlations between gene Y00787 and the last eight genes in Table 7 are also very low, as they are detected specifically in those two models where Y00787 is not. In a similar way, pairwise correlations in this 8–gene group are often high. Complementarity in the detection by the different models emphasizes one of the biggest problems of lasso selection, also marked in [54]: when there is a group of significan variables with high pairwise correlation lasso selects only one, and does not care which one.

A bunch of articles can be found in the gene expression literature looking for the genes associated with the ALL/AML status. It is expected that exists some kind of intersection between the sets of genes given by the different studies. First five genes in the relation of Table 7 (M27891, M19507, M84526, Y00787 and M92287) are also discovered in [32], being M27891 the one showing the strongest association with disease, as happens here. Three of the four genes pointed out in [18] (U82759, HG1612 and X95735) are also discovered here. On the other hand, coincidences with the list given in [44] are more limited.

4 Conclusion.

We study lasso logistic regression by means of a generalized soft–threshold (GSoft) estimator. An equivalence between existence of a solution in GSoft and convergence of the CCD algorithm to the same solution is given. An approximation of the covariance matrix for the estimated coefficients $\widehat{\beta}$ based on the GSoft approach produces very accurate results. The CCD algorithm

Genes	RD-γ Klinger	RD-γ Zou	RD-γ Lasso	P-γ Klinger	P-γ Zou	P-γ Lasso
M27891	X	X	X	X	X	X
M19507	X	X	X		X	
M84526	X			X	X	X
Y00787		X	X		X	X
M92287				X		X
U05255		X	X			
M17733		X	X			
M63138	X		X			
M96326	X	X				
L07633	X			X		
U82759	X			X		
HG1612	X			X		
M13690	X			X		
M23197	X			X		
X95735	X			X		
Y07604	X			X		
X85116	X			X		

Table 7: *Genes with nonzero estimated coefficients in the different models for the leukemia dataset. Here we show the seventeen ones which are detected in more than one model.*

is fast, stable and efficient, and allows different kinds of implementations. Efficiency of the optimization algorithm is a main issue nowadays, as the datasets used in many fields (text categorization, image processing,...) have extraordinary high dimensions.

We tried different options for the vector $\mathbf{\Gamma}$ of specific penalizations in GSoft. Some of them are based in the variability shown by each covariate, while others depend on previous application of penalized regression approaches to data. Their consistency properties follow from appropriate developments in the recent literature.

Finally, we applied these methods to simulated and real gene expression data. The same simulations carried out in other studies were used here, in order to provide honest and fair comparisons. Common real gene expression datasets, like leukemia or colon, allow us to know the ability of these methods to detect genes related with the disease or trait under study. The penalized regression approaches performed in this work are expected to give rise to sparse models, where only a very small percentage of covariates (genes) have weight in classification/prediction.

Appendix. Proof of theorem 2

The log–likelihood functions in logistic regression and in lasso logistic regression with specific penalizations are given in (5) and (6), respectively. The first partial derivatives or score functions are:

$$s_j(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{-y_i x_{ij}}{1 + \exp(y_i \mathbf{x}_i^{'} \boldsymbol{\beta})} \tag{33}$$

The definition of the Δv_j for the lasso case in [15], applied on a penalized regression problem with specific penalizations for each variable, is given in (9). For ease of notation, we will use here S instead of $\text{sign}(\beta_j)$. We will base the entire proof in the steps and the notations used in Figures 4 and 5 in [15]. Many of the terms used there will be repeated here. To clarify the notation, we will use β_j for the true value of the coefficients and $\beta_j^{(I)}$ for the value of the jth coefficient in the iteration I of the CCD algorithm.

We will begin proving the equivalence for the case $\beta_j = 0$, and then we will move to the more general case of $\beta_j > 0$ (analogous proof for $\beta_j < 0$).

Case $\beta_j = 0$.

(1) $\Rightarrow$ (2)

We assume that the CCD algorithm, as explained in [15], converges. Therefore, from a certain iteration I we have $\beta_j^{(I)} = 0$ and $\Delta v_j^{(I)} = 0$. The CCD algorithm tries then to improve the objective function value searching in the positive and the negative direction, so:

$$\left\{ \begin{array}{l} S = 1 \text{ and } \Delta v_j^{(I+1)} \leq 0 \;\Leftrightarrow\; s_j(\boldsymbol{\beta}) - \lambda\gamma_j \leq 0 \\ S = -1 \text{ and } \Delta v_j^{(I+1)} \geq 0 \;\Leftrightarrow\; s_j(\boldsymbol{\beta}) + \lambda\gamma_j \geq 0 \end{array} \right\} \Leftrightarrow \tag{34}$$

$$\left\{ \begin{array}{l} s_j(\boldsymbol{\beta}) \leq \lambda\gamma_j \\ -s_j(\boldsymbol{\beta}) \leq \lambda\gamma_j \end{array} \right\} \Leftrightarrow \tag{35}$$

$$\Leftrightarrow\; |s_j(\boldsymbol{\beta})| \leq \lambda\gamma_j \tag{36}$$

(2) $\Rightarrow$ (1)

We assume now that the necessary and sufficient conditions for convergence in the GSoft theorem are fulfilled. That implies, for β_j

$$|s_j(\beta)| \leq \lambda\gamma_j$$

We need to bear in mind also that the initial value for β_j in the CCD algorithm is $\beta_j^{(0)} = 0$. In this situation and from the definitions of the CCD algorithm for the lasso case, we have that

- if we try $S = 1$ (positive direction) then $\Delta v_j^{(0)} \leq 0$ and positive direction failed.
- if we try $S = -1$ (negative direction) then $\Delta v_j^{(0)} \geq 0$ and negative direction failed.

Therefore, following the steps of the CCD algorithm for the lasso case, this means we take $\Delta v_j^{(0)} = 0$, as both directions failed, and then

$$\Delta\beta_j = \min\big(\max(0, -\Delta_j), \Delta_j\big) = \min\big(0, \Delta_j\big) = 0 \tag{37}$$

and the CCD algorithm converges.

Case $\beta_j > 0$ (the proof is analogous for $\beta_j < 0$).

(1) $\Rightarrow$ (2)
Let us suppose that $s_j(\boldsymbol{\beta}) \neq \lambda\gamma_j$ and we will try to show that this gives rise to a contradiction. As the true β_j is positive and the CCD algorithm converges, from any iteration I we will have $\beta_j^J > 0$ for all iteration $J > I$, so $S = 1$ and $\Delta v_j^J \neq 0$ following the definition in (9). This way, for any positive constant k,

$$\Delta\beta_j^{(J)} = \min\big(\max(\Delta v_j^{(J)}, -\Delta_j^{(J)}), \Delta_j^{(J)}\big) \neq 0 \quad \Rightarrow \tag{38}$$

$$\Rightarrow \qquad \Delta_j^{(J+1)} = \max\Big(2|\Delta\beta_j^{(J)}|, \tfrac{\Delta_j^{(J)}}{2}\Big) > k > 0 \tag{39}$$

and this happens for every iteration $J > I$, which enters in contradiction with the convergence of the CCD algorithm to β_j.
(2) $\Rightarrow$ (1)
We assume now that necessary and sufficient conditions for convergence in the GSoft theorem are fulfilled; let us suppose that the CCD algorithm converges to a different "solution" $\bar{\beta} \neq \boldsymbol{\beta}$ with $\bar{\beta}_j \neq \beta_j$.

In such case, as the conditions in (a) in the GSoft theorem determine an unique solution, it has to be $s_j(\bar{\beta}) \neq \lambda\gamma_j$; then $\Delta v_j^{(J)} \neq 0$, for all $J > I$ with $I \in \mathbb{N}$ and therefore $\Delta\bar{\beta}_j$ does not converge to 0, which means the CCD algorithm does not converge either, and we have reached a contradiction.

We have not mentioned or used anywhere in the proof the condition about the positive definite nature of the matrix $X_\lambda' H(\hat{\eta}) X_\lambda$. So we have to prove this condition is also fulfilled when the CCD algorithm converges. We will prove this by *reductio ad absurdum*.

Let us assume that $X_\lambda' H(\hat{\eta}) X_\lambda$ is not definite positive. As X_λ is a complete matrix, this implies that $H(\hat{\eta})$ is not definite positive, and therefore

$$\left.\begin{array}{l} -H(\hat{\eta}) \text{ (Hessian) is not definite negative} \\ \frac{\partial L_1(\hat{\beta})}{\partial \beta_j} = 0 \text{ for all } j \in \{1, \ldots, p\} \end{array}\right\} \tag{40}$$

and therefore the estimated linear predictor $\hat{\eta}$ cannot be a maximum of the objective function in [28], which means $\hat{\beta}$ is not a minimum of the objective function in [15] and the CCD algorithm does not converge (contradiction).

References

[1] Alon U., Barkai N., Notterman D., Gish K., Ybarra S., Mack D. and Levine A.J. *Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays.* Proceedings of the National Academy of Sciences USA, **96**(12), (1999), 6745–6750.

[2] Antoniadis A. and Fan J. *Regularization of wavelet approximations.* Journal of the American Statistical Association, **96**(455), (2001), 939–967.

[3] Antoniadis A., Gijbels I. and Nikolova M. *Penalized likelihood regression for generalized linear models with nonquadratic penalties.* Unpublished manuscript.

[4] Bakin S. *Adaptive regression and model selection in data mining problems.* PhD Thesis, Australian National University, Canberra, (1999).

[5] Boulesteix A.L., Tutz G. and Strimmer K. *A CART–based approach to discover emerging patterns in microarray data.* Bioinformatics, **19**(18), (2003), 2465–2472.

[6] De Mol C., De Vito E. and Rosasco L. *Elastic–net regularization in learning theory.* Journal of Complexity, **25**(2), (2009), 201–230.

[7] Dettling M. and Buhlmann P. *Supervised clustering of genes.* Genome Biology, **3**(12), (2002), 0069.1–0069.15.

[8] Dettling M. *BagBoosting for tumor classification with gene expression data.* Bioinformatics, **20**(18), (2004), 3583–3593.

[9] Dettling M. and Buhlmann P. *Finding predictive gene groups from microarray data.* Journal of Multivariate Analysis, **90**, (2004), 106–131.

[10] Dudoit S. Fridlyand J. and Speed T.P. *Comparison of discrimination methods for the classification of tumors using gene expression data.* Journal of the American Statistical Association, **97**(457), (2000), 77–87.

[11] Efron B., Hastie T., Johnstone I. and Tibshirani R. *Least angle regression.* Annals of Statistics, **32**(2) (2004), 407–499.

[12] Fan J. and Li R. *Variable selection via nonconcave penalized likelihood and its oracle properties.* Journal of the American Statistical Association, **96**(456), (2001), 1348–1360.

[13] Frank I.E. and Friedman J.H. *A statistical view of some chemometrics tools.* Technometrics, **35**(2), (1993), 109–135.

[14] Friedman J., Hastie T. and Tibshirani R. *Regularization paths for generalized linear models via coordinate descent.* Technical Report, Department of Statistics, Stanford University, (2008).

[15] Genkin A., Lewis D.D. and Madigan D. *Sparse logistic regression for text categorization.* DIMACS Working Group on Monitoring Message Streams, Project Report, (2005).

[16] Goldstein T. and Osher S. *The Split Bregman method for L1 regularized problems.* UCLA CAAM Report 08–29, (2008).

[17] Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeck M., Mesirov P., Coller H., Loh M.L., Downing J.R., Caligiuri M.A. et al. *Molecular classification of cancer: class discovery and class prediction by gene expression monitoring.* Science, **286**(5439), (1999), 531–537.

[18] Guyon I., Weston J., Barnhill S. and Vapnik V. *Gene selection for cancer classification using support vector machines.* Machine Learning, **46**(1–3), (2002), 389–422.

[19] Guyon I. and Elisseeff A. *An introduction to variable and feature selection.* Journal of Machine Learning Research, **3**(Mar), (2003), 1157–1182.

[20] Hall M. *Correlation–based feature selection for machine learning.* PhD Thesis, Department of Computer Science, Waikato University, New Zealand, (1999).

[21] Hoerl A.E. and Kennard R. *Ridge regression: biased estimation for nonorthogonal problems.* Technometrics, **12**(1), (1970), 55–67.

[22] Huang J., Horowitz J.L. and Ma S. *Asymptotic properties of bridge estimators in sparse high–dimensional regression models.* Technical Report, Department of Statistics and Actuarial Science, The University of Iowa, (2006).

[23] Huang J., Ma S. and Zhang C.H. *Adaptive lasso for sparse high–dimensional regression models.* Statistica Sinica, **18**(4), (2008), 1603–1618.

[24] Huang J., Ma S. and Zhang C.H. *The iterated lasso for high–dimensional logistic regression.* Technical report No. 392, The University of Iowa, (2008).

[25] Hubert P.J. *The behavior of maximum likelihood estimates under nonstandard conditions.* Proceedings of the Fifth Berkeley Simposium on Mathematical Statistics and Probability, (1967).

[26] Hunter D.R. and Li R. *Variable selection using MM algorithms.* Annals of Statistics, **33**(4), (2005), 1617–1642.

[27] Kim Y., Choi H. and Oh H.S. *Smoothly clipped absolute deviation on high dimensions.* Journal of the American Statistical Association, **103**(484), (2008), 1665–1673.

[28] Klinger A. *Inference in high dimensional generalized linear models based on soft thresholding.* Journal of the Royal Statistical Society Series B, **63**(2), (2002), 377–392.

[29] Knight K. and Fu W.J. *Asymptotics for lasso–type estimators.* Annals of Statistics, **28**(5), (2000), 1356–1378.

[30] Krishnapuram B., Carin L., Figueiredo M.A.T. and Hartemink A.J. *Sparse multinomial logistic regression: fast algorithms and generalization bounds.* IEEE Transactions on Pattern Analysis and Machine Intelligence, **27**(6), (2005), 957–968.

[31] Lee S.I., Lee H., Abbeel P. and Ng A.Y. *Efficient L1 regularized logistic regression.* Proceedings of the Twenty–first International Conference on Machine Learning (AAAI-06), (2006).

[32] Lee K.E., Sha N., Dougherty E.R., Vannucci M. and Mallick B.K. *Gene selection: a bayesian variable selection approach.* Bioinformatics, **19**(1), (2003), 90–97.

[33] Liu Z., Jiang F., Tian G., Wang S., Sato F., Meltzer S.J. and Tan M. *Sparse logistic regression with L_p penalty for biomarker identification.* Statistical Applications in Genetics and Molecular Biology, **6**(1), (2007), article 6.

[34] Lv J. and Fan Y. *A unified approach to model selection and sparse recovery using regularized least squares.* Annals of Statistics, **37**(6A), (2009), 3498–3528.

[35] Meier L., van de Geer S. and Bühlmann P. *The group lasso for logistic regression.* Journal of the Royal Statistical Society Series B, **70**(1), (2008), 53–71.

[36] Meinshausen N. and Buhlmann P. *High dimensional graphs and variable selection with the lasso.* Annals of Statistics, **34**(3), (2006), 1436–1462.

[37] Nguyen D.V. and Rocke D.M. *Tumor classification by partial least squares using microarray gene expression data.* Bioinformatics, **18**(1), (2002), 39–50.

[38] Park M.Y. and Hastie T. *An L_1 regularization–path algorithm for generalized linear models.* Manuscript, Department of Statistics, Stanford University, (2006).

[39] Roth V. *The generalized lasso.* IEEE Transactions on Neural Networks, **15**, (2004), 16–28.

[40] Schmidt M., Fung G. and Rosales R. *Fast optimization methods for L_1 regularization: a comparative study and two new approaches.* European Conference on Machine Learning (ECML), (2007).

[41] Shevade S. and Keerthi S. *A simple and efficient algorithm for gene selection using sparse logistic regression.* Bioinformatics, **19**(17), (2003), 2246–2253.

[42] Tarigan B. and van de Geer S. *Classifiers of support vector machine type with L_1 complexity regularization.* Bernoulli, **12**(6), (2006), 1045–1076.

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

[44] Thomas J.G., Olson J.M. and Tapscott S.J. *An efficient and robust statistical modeling approach to discover diferentially expressed genes using genomic expression profiles.* Genome Research, **11**, (2001), 1227–1236.

[45] Wang H. and Leng C. *Unified lasso estimation via least squares approximation.* Journal of the American Statistical Association, **102**(479), (2007), 1039–1048.

[46] Weston J., Elisseeff A., Scholkopf B. and Tipping M. *Use of the zero–norm with linear models and kernel methods.* Journal of Machine Learning Research, **3**(Mar), (2003), 1439–1461.

[47] Wright S.J., Nowak R.D. and Figueiredo M.A.T. *Sparse reconstruction by separable approximation.* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (2008).

[48] Wu T.T. and Lange K. *Coordinate descent algorithms for lasso penalized regression.* Annals of Applied Statistics, **2**(1), (2008), 224–244.

[49] Yuan M. and Lin Y. *Model selection and estimation in regression with grouped variables.* Journal of the Royal Statistical Society Series B, **68**(1), (2006), 49–67.

[50] Zhang T. and Oles F. *Text categorization based on regularized linear classifiers.* Information Retrieval, **4**(1), (2001), 5–31.

[51] Zhang C.H. and Huang J. *The sparsity and bias of the lasso selection in high–dimensional linear regression.* Annals of Statistics, **36**(4), (2008), 1567–1594.

[52] Zhang T. *Some sharp performance bounds for least squares regression with L1 regularization.* Annals of Statistics, **37**(5A), (2009), 2109–2144.

[53] Zou H. *The adaptive lasso and its oracle properties.* Journal of the American Statistical Association, **101**(476), (2006), 1418–1429.

[54] Zou H. and Hastie T. *Regularization and variable selection via the elastic net.* Journal of the Royal Statistical Society Series B, **67**(2), (2005), 301–320.

[55] Zou H. and Li R. *One–step sparse estimates in nonconcave penalized likelihood models.* Annals of Statistics, **36**(4), (2008), 1509–1533.

Reports in Statistics and Operations Research

2005

05-01 SiZer Map for Evaluating a Bootstrap Local Bandwidth Selector in Nonparametric Additive Models. *M. D. Martínez-Miranda, R. Raya-Miranda, W. González-Manteiga and A. González-Carmona.*

05-02 The Role of Commitment in Repeated Games. *I. García Jurado, Julio González Díaz.*

05-03 Project Games. *A. Estévez Fernández, P. Borm, H. Hamers*

05-04 Semiparametric Inference in Generalized Mixed Effects Models. *M. J. Lombardía, S. Sperlich*

2006

06-01 A unifying model for contests: effort-prize games. J. González Díaz

06-02 The Harsanyi paradox and the "right to talk" in bargaining among coalitions. J. J. Vidal Puga

06-03 A functional analysis of NOx levels: location and scale estimation and outlier detection. M. Febrero, P. Galeano, W. González-Manteiga

06-04 Comparing spatial dependence structures. R. M. Crujeiras, R. Fernández-Casal, W. González-Manteiga

06-05 On the spectral simulation of spatial dependence structures. R. M. Crujeiras, R. Fernández-Casal

06-06 An L_2-test for comparing spatial spectral densities. R. M. Crujeiras, R. Fernández-Casal, W. González-Manteiga.

2007

07-01 Goodness-of-fit tests for the spatial spectral density. R. M. Crujeiras, R. Fernández-Casal, W. González-Manteiga.

07-02 Presmothed estimation with left truncated and right censores data. M. A. Jácome, M. C. Iglesias-Pérez

07-03 Robust nonparametric estimation with missing data. G. Boente, W. González-Manteiga, A. Pérez-González

07-04 k-Sample test based on the common area of kernel density estimators, P. Martínez-Camblor, J. de Uña Álvarez, N. Corral-Blanco

07-05 A bootstrap based model checking for selection-biased data, J. L. Ojeda, W. González-Manteiga, J . A. Cristobal

07-06 The Gaussian mixture dynamic conditional correlation model: Bayesian estimation, value at risk calculation and portfolio selection, P. Galeano, M. C. Ausín

2008

08-01 ROC curves in nonparametric location-scale regression models, W. González-Manteiga, J. C. Pardo Fernández, I. Van Keilegom

08-02 On the estimation of α-convex sets, B. Pateiro-López, A. Rodríguez-Casal.

2009

09-01 Lasso Logistic Regression, GSoft and the Cyclyc Coordinate Descent Algorithm. Application to Gene Expression Data

Previous issues (2001 – 2004):
http://eio.usc.es/pub/reports.html