Extended differential geometric LARS for high-dimensional GLMs with general dispersion parameter

Pazira, Hassan; Augugliaro, Luigi; Wit, Ernst

Published in:
Statistics and Computing

DOI:
10.1007/s11222-017-9761-7

Publication date:
2018

Extended differential geometric LARS for high-dimensional GLMs with general dispersion parameter

Hassan Pazira[1], Luigi Augugliaro[2] and Ernst Wit[1]
[1] *University of Groningen, The Netherlands*
[2] *University of Palermo, Italy*

Abstract

A large class of modelling and prediction problems involve outcomes that belong to an exponential family distribution. Generalized linear models (GLMs) are a standard way of dealing with such situations. Even in high-dimensional feature spaces GLMs can be extended to deal with such situations. Penalized inference approaches, such as the ℓ_1 or SCAD, or extensions of least angle regression, such as dgLARS, have been proposed to deal with GLMs with high-dimensional feature spaces. Although the theory underlying these methods is in principle generic, the implementation has remained restricted to dispersion free models, such as the Poisson and logistic regression models. The aim of this manuscript is to extend the differential geometric least angle regression method for high-dimensional GLMs to arbitrary exponential dispersion family distributions with arbitrary link functions. This entails, first, extending the predictor-corrector (PC) algorithm to arbitrary distributions and link functions, and second, proposing an efficient estimator of the dispersion parameter. Furthermore, improvements to the computational algorithm lead to an important speed-up of the PC algorithm. Simulations provide supportive evidence concerning the proposed efficient algorithms for estimating coefficients and dispersion parameter. The resulting method has been implemented in our R package (which will be merged with the original `dglars` package) and is shown to be an effective method for inference for arbitrary classes of GLMs.

Keywords: *High-dimensional inference, Generalized linear models, Least angle regression, Predictor-corrector algorithm, Dispersion parameter.*

1 Introduction

Nowadays, high-dimensional data problems, where the number of predictors is larger than the sample size, are becoming more common. In such scenarios, it is often sensible to assume that only a small number of predictors contributes to the response, i.e., that the underlying, generating model is sparse. With a sparse model we mean many elements equal to zero. Modern statistical methods for sparse regression models are usually based on using a penalty function to estimate a solution curve embedded in the parameter space and then to find the point that represents the best compromise between sparsity and predictive behaviour of the model. Some important examples are the Least Absolute Shrinkage and Selection Operator (LASSO) estimator (Tibshirani, 1996), the Smoothly Clipped Absolute Deviation (SCAD) method (Fan and Li, 2001), the Dantzig selector (Candes and Tao, 2007), which

[1] Corresponding: e.c.wit@rug.nl

was extended to generalized linear models (GLMs) in James and Radchenko (2009), and the MC+ penalty function introduced in Zhang (2010), among others.

Differently from the methods cited above, Efron *et al.* (2004) introduced a new method to select important variables in a linear regression model called least angle regression method (LARS) which was extended to Generalized Linear Models (GLM) in Augugliaro *et al.* (2013) by using the differential geometry. This method, which does not require an explicit penalty function, has been called differential geometric LARS (dgLARS) method because it is defined generalizing the geometrical ideas on which LARS is based. As underlined in Augugliaro *et al.* (2013), LARS is a proper likelihood method in its own right: it can be generalized to any model and its success does not depend on the arbitrary match of the constraint and the objective function, as is the case in penalized inference methods. In particular, using the differential geometric characterization of the classical signed Rao score test statistic, dgLARS gains important theoretical properties that are not shared by other methods. From a computational point of view, the dgLARS method essentially consists in the computation of the implicitly defined solution curve. In Augugliaro *et al.* (2013), this problem is solved by using a predictor-corrector (PC) algorithm.

Although the theory of the dgLARS method does not require restrictions on the dispersion parameter, the `dglars` package Augugliaro (2014b) is restricted to logistic and Poisson regression models, i.e., two specific GLMs with canonical link function and dispersion parameter is equal to one. Furthermore, the authors do not consider the problem of how to estimate the dispersion parameter in a high-dimensional setting. The aim of this paper is to overcome this restriction and to define dgLARS for any generalized linear model with arbitrary link function. First, we extend the PC algorithm to GLMs with generic link function and unknown dispersion parameter; we also improve the algorithm by proposing a new method to reduce the number of solution points needed to approximate the dgLARS solution curve. As we shall show in the simulation study, the proposed algorithm outperforms the old PC algorithm previously implemented in `dglars` package. Second, we explicitly consider the problem of how to do inference on the dispersion parameter and we propose an extension of the method developed in Fan *et al.* (2012) and then we present an iterative algorithm to improve the accuracy of the new proposed method for estimating the dispersion parameter.

The paper is organized as follows; In Section 2, firstly, we introduce the extended dgLARS method by giving some essential clues to the theory underlying a generalized linear model from a differential geometric point of view and present the general case of equations based on the class of the exponential family. Secondly, we propose our improved predictor-corrector algorithm, and thirdly we present an estimator for dispersion parameter which can be used during the solution path, and at the end of the section we consider some model selection strategies that are commonly used. In Section 3, we focus on the estimation of the dispersion parameter and propose a new method to do high-dimensional inference on the dispersion parameter of the exponential family, and after that, we propose an iterative algorithm to achieve a more stable and accurate estimation. In Section 4, the simulation studies is given divided into two subsections; in the first, a comparison in terms of performance between the improved PC algorithm and other methods is done; in the second, we investigate how well the new estimator of the dispersion parameter based on the proposed iterative algorithm behaves. The application and data analysis based on continuous outcome are described in Section 5.

2 Differential Geometric LARS for general GLM

The original LARS algorithm (Efron *et al.*, 2004) defines a coefficient solution path for a linear regression model by sequentially adding variables to the solution curve. To make this section self container, we briefly review the LARS method. Starting with only the intercept, the LARS algorithm finds the covariate that is most correlated with the response variable and proceeds in this direction by changing its associated linear parameter. The algorithm takes the largest step possible in the direction of this covariate until another covariate has as much correlation with the current residual as the current covariate. At that point the LARS algorithm proceeds in an equiangular direction between the two covariates until a new covariate earns its way into the equally most correlated set. Then it proceeds in the direction in which the residual makes an equal angle with the three covariates, and so on. Augugliaro *et al.* (2013) generalized these notions for GLMs by using differential geometry. The resulting defines a continuous solution path for GLM, with on the extreme of the path the maximum likelihood estimate of the coefficient vector and on the other side the intercept-only estimate. The aim of the method is to define a continuous model path with highest likelihood with the fewest number of variables. The reader interested in more of the differential geometric details of this method and its extensions is referred to Augugliaro *et al.* (2013, 2016). In this section, after a brief overview on GLMs, we derive the equations defining the dgLARS solution curve for a GLM with an arbitrary link function. Furthermore, we explicitly consider the role of the dispersion parameter and we shall show that it acts as a scale parameter of the tuning parameter γ. At the end of this section, we propose our improved algorithm and estimators of the dispersion parameter.

2.1 An overview on GLMs: terminology and notation

Let $\mathbf{Y} = (Y_1, Y_2, \cdots, Y_n)^\top$ be a n-dimensional random vector with independent components. In what follows we shall assume that Y_i is a random variable with probability density function belonging to an exponential dispersion family (Jorgensen, 1987, 1997), i.e.,

$$p_{Y_i}(y_i; \theta_i, \phi) = \exp\left\{\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i, \phi)\right\}, \qquad y_i \in \mathcal{Y}_i \subseteq \mathbb{R}, \tag{1}$$

where $\theta_i \in \Theta_i \subseteq \mathbb{R}$ is the canonical parameter, $\phi \in \Phi \subseteq \mathbb{R}^+$ is the dispersion parameter, and $a(.)$, $b(.)$ and $c(.,.)$ are given functions. In the following, we assume that each Θ_i is an open set and $a(\phi) = \phi$. We consider ϕ as an unknown parameter. The expected value of $\mathbf{Y}$ is related to the canonical parameter by $\boldsymbol{\mu} = \{\mu(\theta_1), \cdots, \mu(\theta_n)\}^\top$, where $\mu(\theta_i) = \frac{\partial b(\theta_i)}{\partial \theta_i}$ is called mean value mapping, and the variance of $\mathbf{Y}$ is related to its expected value by the identity $\mathrm{Var}(\mathbf{Y}) = \phi\mathbf{V}(\boldsymbol{\mu})$, where $\mathbf{V}(\boldsymbol{\mu}) = diagV(\mu_1), \ldots, V(\mu_n)$ is an $n \times n$ diagonal matrix with elements, called the variance functions, $V(\mu_i) = \frac{\partial^2 b(\theta_i)}{\partial \theta_i^2}$. Since μ_i is a reparameterization, model (1) can be also denoted as $p_{Y_i}(y_i; \mu_i, \phi)$.

Following McCullagh and Nelder (1989), a Generalized Linear Model (GLM) is defined by means of a known function $g(\cdot)$, called link function, relating the expected value of each Y_i to the vector of covariates $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^\top$ by the identity

$$g\{E(Y_i)\} = \eta_i = \mathbf{x}_i^\top \boldsymbol{\beta}$$

where η_i is called the i^{th} linear predictor and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^\top$ is the vector of regression coefficients. In order to simplify our notation we let $\boldsymbol{\mu}(\boldsymbol{\beta}) = \{\mu_1(\boldsymbol{\beta}), \ldots, \mu_n(\boldsymbol{\beta})\}^\top$ where $\mu_i(\boldsymbol{\beta}) = g^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta})$. Therefore, the joint probability density function can be written as $p_{\mathbf{Y}}(\mathbf{y}; \boldsymbol{\mu}(\boldsymbol{\beta}), \phi) = \prod_{i=1}^n p_{Y_i}(y_i; \mu_i(\boldsymbol{\beta}), \phi)$. In the following of this paper we shall use $\ell(\boldsymbol{\beta}, \phi; \mathbf{y}) = \log p_{\mathbf{Y}}(\mathbf{y}; \boldsymbol{\mu}(\boldsymbol{\beta}), \phi)$ as notation for the log-likelihood function. From (1), the m^{th} score function is given as

$$\begin{aligned} \partial_m \ell(\boldsymbol{\beta}, \phi; \mathbf{y}) &= \frac{\partial \ell(\boldsymbol{\beta}, \phi; \mathbf{y})}{\partial \beta_m} \\ &= \phi^{-1} \sum_{i=1}^n \frac{(y_i - \mu_i)}{V(\mu_i)} x_{im} \left(\frac{\partial \mu_i}{\partial \eta_i} \right) = \phi^{-1} \; \partial_m \ell(\boldsymbol{\beta}; \mathbf{y}), \end{aligned} \tag{2}$$

where $\mu_i = g^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta})$, and the Fisher Information matrix has terms

$$\begin{aligned} \mathcal{I}_{mn}(\boldsymbol{\beta}, \phi) &= E[\partial_m \ell(\boldsymbol{\beta}, \phi; \mathbf{y}) \cdot \partial_n \ell(\boldsymbol{\beta}, \phi; \mathbf{y})] \\ &= \phi^{-1} \sum_{i=1}^n \frac{x_{im} \; x_{in}}{V(\mu_i)} \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 = \phi^{-1} \; \mathcal{I}_{mn}(\boldsymbol{\beta}), \end{aligned} \tag{3}$$

Using (2) and (3), we obtain expressions $\partial_{mn} \ell(\boldsymbol{\beta}, \phi; \mathbf{y})$ and $r_m(\boldsymbol{\beta}, \phi)$ to be used in Section 3 and Section 2.2, respectively, as follows:

$$\begin{aligned} \partial_{mn} \ell(\boldsymbol{\beta}, \phi; \mathbf{y}) &= \frac{\partial^2 \ell(\boldsymbol{\beta}, \phi; \mathbf{y})}{\partial \beta_m \partial \beta_n} \\ &= \phi^{-1} \sum_{i=1}^n \left\{ x_{im} \; x_{in} \; (y_i - \mu_i) \left[\frac{\partial^2 \theta_i}{\partial \mu_i^2} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 + \frac{\partial \theta_i}{\partial \mu_i} \cdot \frac{\partial^2 \mu_i}{\partial \eta_i^2} \right] \right. \\ &\left. - \frac{\partial \theta_i}{\partial \mu_i} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right\} \\ &= \phi^{-1} \sum_{i=1}^n \left\{ x_{im} \; x_{in} \; (y_i - \mu_i) \left(\frac{\partial^2 \theta_i}{\partial \mu_i^2} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 + \frac{\partial \theta_i}{\partial \mu_i} \cdot \frac{\partial^2 \mu_i}{\partial \eta_i^2} \right) \right\} - \mathcal{I}_{mn}(\boldsymbol{\beta}, \phi) \end{aligned} \tag{4}$$

where $\frac{\partial \theta_i}{\partial \mu_i} = \frac{1}{V(\mu_i)}$ and $\frac{\partial^2 \theta_i}{\partial \mu_i^2} = -\frac{\partial V(\mu_i)/\partial \mu_i}{V(\mu_i)^2}$. The Rao's score test statistic is given as

$$\begin{aligned} r_m(\boldsymbol{\beta}, \phi) &= \frac{\partial_m \ell(\boldsymbol{\beta}, \phi; \mathbf{y})}{\sqrt{\mathcal{I}_m(\boldsymbol{\beta}, \phi)}} \\ &= \phi^{-1/2} \; \frac{\sum_{i=1}^n \left\{ \frac{(y_i - \mu_i) \; x_{im}}{V(\mu_i)} \cdot \frac{\partial \mu_i}{\partial \eta_i} \right\}}{\left(\sum_{i=1}^n \left\{ \frac{x_{im}^2}{V(\mu_i)} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right\} \right)^{1/2}} = \phi^{-1/2} \; r_m(\boldsymbol{\beta}) \end{aligned} \tag{5}$$

where $\mathcal{I}_m(\boldsymbol{\beta}, \phi) = \mathcal{I}_{mm}(\boldsymbol{\beta}, \phi)$. The Rao's score test statistic helps to define $\rho_m(\boldsymbol{\beta})$, the angle between the m^{th} basis function $\partial_m \ell(\boldsymbol{\beta}, \phi; \mathbf{Y})$ and the tangent residual vector $\boldsymbol{r}(\boldsymbol{\beta}, \phi, \mathbf{y}; \mathbf{Y}) = \sum_{i=1}^n (y_i - \mu_i) \; \frac{\partial \ell(\boldsymbol{\beta}, \phi; \mathbf{y})}{\partial \mu_i}$, defined as follows

$$\rho_m(\boldsymbol{\beta}, \phi) = \arccos \left[\frac{r_m(\boldsymbol{\beta}, \phi)}{\| \boldsymbol{r}(\boldsymbol{\beta}, \phi, \mathbf{y}; \mathbf{Y}) \|_{p(\boldsymbol{\mu}(\boldsymbol{\beta}))}} \right], \tag{6}$$

where $\|\cdot\|_{p(\boldsymbol{\mu}(\boldsymbol{\beta}))}$ is the norm defined on the tangent space $\mathcal{T}_{p(\boldsymbol{\mu}(\boldsymbol{\beta}))}\mathcal{M}$, where the set $\mathcal{M}$ is a p-dimensional submanifold of the differential manifold $\mathcal{S}$ (for details about the $\mathcal{M}$ and $\mathcal{S}$ sets, see Augugliaro *et al.*, 2013). The angle will be used in Section 2.2 to define an extension of the least angle regression (Efron *et al.*, 2004). From (6), the Rao's score test statistic contains the same information as the angle $\rho_m(\boldsymbol{\beta})$. Thereby we can define the dgLARS method with respect to the Rao's score test statistic rather than the angle as respects the smallest angle is equivalent to the largest Rao's score test statistic.

Gamma and Inverse Gaussian GLMs

The binomial, Poisson and Gaussian GLMs are by far the most commonly used, but there are a number of lesser known GLMs which are useful for particular types of data. The Gamma and Inverse Gaussian GLMs are intended for continuous and right-skewed responses. They are double-parameter GLMs and belong to the exponential dispersion family (EDF). The Gamma distribution is a member of the additive EDF and the Inverse Gaussian distribution is a member of the reproductive EDF (Panjer, 2006). We consider these two dispersion parameter models as follows; For Gamma family, we assume that $Y_i \sim G(\nu, \frac{\mu_i}{\nu})$ so that:

$$f_{Y_i}(y_i;\mu_i,\nu) = \exp\left\{\frac{-y_i\frac{1}{\mu_i} - \log(\mu_i)}{\frac{1}{\nu}} + \nu\log(y_i\nu) - \log(y_i\Gamma(\nu))\right\}, \quad y_i > 0,$$

then $E(Y_i) = -\frac{1}{\theta_i} = \mu_i$ and $\text{Var}(Y_i) = \phi\ V(\mu_i) = \frac{\mu_i^2}{\nu}$, where $\phi^{-1} = \nu$. We consider three of the most commonly used link functions: (i) the canonical link function, "*inverse*", $\eta_i = -\mu_i^{-1}$, (ii) "*log*", $\exp(\eta_i) = \mu_i$, and (iii) "*identity*", $\eta_i = \mu_i$. For Inverse Gaussian family, we assume that $Y_i \sim IG(\mu_i, \lambda)$ so that:

$$f_{Y_i}(y_i;\mu_i,\lambda) = \exp\left\{\frac{y_i(-\frac{1}{2\mu_i^2}) + 1/\mu_i}{1/\lambda} - \frac{\lambda}{2y_i} - \frac{1}{2}\log(\frac{2\pi y_i^3}{\lambda})\right\}, \quad y_i > 0,$$

then $E(Y_i) = \frac{1}{\sqrt{-2\theta_i}} = \mu_i$ and $\text{Var}(Y_i) = \phi\ V(\mu_i) = \frac{\mu_i^3}{\lambda}$, where $\phi^{-1} = \lambda$. We consider four of the most commonly used link functions: (i) the canonical link function, "*inverse-square*", $\eta_i = -0.5\mu_i^{-2}$, (ii) "*inverse*", $\eta_i = -\mu_i^{-1}$, (iii) "*log*", and (iv) "*identity*".

Table A1 in Appendix A shows all required equations for obtaining the dgLARS estimator based on the Gamma and Inverse Gaussian models with the most commonly used link functions.

2.2 The extended dgLARS method

Augugliaro *et al.* (2013) showed that the dgLARS estimator follows naturally from a differential geometric interpretation of a GLM, generalizing the LARS method (Efron *et al.*, 2004) using the angle between scores and tangent residual vector, as defined in (6). LARS and dgLARS algorithms define a coefficient solution curve by identifying the most important variables step by step and including them into the model at specific points of the path. The original algorithms took as starting point of the path the model with the intercept only. This is a sensible choice as it makes the model invariant under affine transformations of

the response or the covariates. However, the choice of the starting point of the least angle approach can be used to incorporate prior information about which variables are expected to be part of the final model and which ones one does not want to make subject to selection. The extended dgLARS method allows for a set of covariates, possible including the intercept, that are always part of the model. We define the set of the *protected variables* $\mathcal{P} = \{a_1^0, \ldots, a_b^0\}$, where $b = |\mathcal{P}| \leq \min(n, p+1)$ and a_j^0 is the index of the j^{th} protected variable. The idea is that variable a_j^0 is supposed to be of interest and should always be contained in the model during the path estimation procedure. The best example of a commonly protected variable is the intercept.

In the path estimation of the coefficients, we treat the protected variables in the set $\mathcal{P}$ differently from the other variables which are not protected, in the sense that the tangent residual vector is always orthogonal to the basis vector $\partial_j \ell(\hat{\boldsymbol{\beta}}(\gamma), \phi; \mathbf{Y})$ for $j \in \mathcal{P}$ at any stage (γ [1]) of the path algorithm $\hat{\boldsymbol{\beta}}(\gamma)$, and thereby by using (6) we have $r_{j\in\mathcal{P}}(\hat{\boldsymbol{\beta}}(\gamma), \phi) = \partial_{j\in\mathcal{P}} \ell(\hat{\boldsymbol{\beta}}(\gamma), \phi; \mathbf{Y}) = 0$. This means that at any stage of the path algorithm, the tangent residual vector contains only information on the non-protected variables denoted by $\mathcal{P}^c = \mathcal{A}(\gamma) \cup \mathcal{N}(\gamma)$, where $\mathcal{A}(\gamma) = \{a_1, \ldots, a_{k(\gamma)}\}$ is the *active set* and $\mathcal{N}(\gamma) = (\mathcal{P} \cup \mathcal{A}(\gamma))^c = \{a_1^c, \ldots, a_{h(\gamma)}^c\}$ is the *non-active set*. The numbers $k(\gamma) = |\mathcal{A}(\gamma)|$ and $h(\gamma) = |\mathcal{N}(\gamma)|$ are the number of included and non-included variables, respectively, in the model at location γ. Thus, we have $p + 1 = b + k(\gamma) + h(\gamma)$.

Let $\hat{\boldsymbol{\beta}}_{\mathbf{0}} = (\hat{\boldsymbol{\beta}}_{\mathcal{P}}, 0, \ldots, 0)^\top$ be the starting point, where $\hat{\boldsymbol{\beta}}_{\mathcal{P}} = (\hat{\beta}_{a_1^0}, \ldots, \hat{\beta}_{a_b^0})$ is the MLE of the protected variables and a zero for each $p + 1 - b$ non-protected variables $\{a_1, \ldots, a_{k(\gamma)}\} \cup \{a_1^c, \ldots, a_{h(\gamma)}^c\}$. Since at the beginning ($\gamma = \gamma_{max}$) the active set $\mathcal{A}(\gamma_{max})$ is empty ($k(\gamma_{max}) = 0$), we have $\mathcal{P}^c = \mathcal{N}(\gamma)$ and $h(\gamma_{max}) = p + 1 - b$. For a specified model (the model with the protected variables) with the starting point $\hat{\boldsymbol{\beta}}_{\mathbf{0}}$, we define γ_{max} to be the largest absolute value of the Rao's score statistic at $\hat{\boldsymbol{\beta}}_{\mathbf{0}}$, i.e.,

$$\gamma_{max} = \max_{m \in \mathcal{P}^c} \{|r_m(\hat{\boldsymbol{\beta}}_{\mathbf{0}})|\}.$$

Since the dispersion parameter in (2)-(6) is equal for any m, we can maximize $|r_{m\in\mathcal{P}^c}(\cdot)|$ (or minimize $\rho_{m\in\mathcal{P}^c}(\cdot)$) instead of $|r_{m\in\mathcal{P}^c}(\cdot, \phi)|$ (or $\rho_{m\in\mathcal{P}^c}(\cdot, \phi)$) in terms of m. The m^{th} variable which has the largest absolute value of $r_{m\in\mathcal{P}^c}(\hat{\boldsymbol{\beta}}_{\mathbf{0}})$ would make an excellent candidate for being included in the model. If we do not have any protected variables, $\hat{\boldsymbol{\beta}}_{\mathbf{0}} = (0, \ldots, 0)^\top$ can be used as the starting point, and in this case, $\boldsymbol{r}(\boldsymbol{\mu}(\mathbf{0}), \mathbf{y}; \mathbf{Y})$ is used to rank the covariates locally.

Before we define the dgLARS method, it can be described using Figure 1 in the following way. First the method selects the predictor, say X_{a_1}, whose basis vector $\partial_{a_1} \ell(\hat{\boldsymbol{\beta}}(\gamma_{max}); \mathbf{Y})$ has the smallest angle with the tangent residual vector, and includes it in the active set $\mathcal{A}(\gamma^{(1)}) = \{a_1\}$, where $\gamma^{(1)} = \gamma_{max}$. The solution curve, at this point $\gamma = \gamma^{(1)}$, $\hat{\boldsymbol{\beta}}(\gamma) = (\hat{\boldsymbol{\beta}}_{\mathcal{P}}(\gamma), \hat{\beta}_{a_1}(\gamma), 0, \ldots, 0)^\top$, where $\hat{\boldsymbol{\beta}}_{\mathcal{P}}(\gamma) = (\hat{\beta}_{a_1^0}(\gamma), \ldots, \hat{\beta}_{a_b^0}(\gamma))$, is chosen in such a way that the tangent residual vector is always orthogonal to the basis vectors $\partial_{j\in\mathcal{P}} \ell(\hat{\boldsymbol{\beta}}(\gamma); \mathbf{Y})$ of the tangent space $\mathcal{T}_{p(\boldsymbol{\mu}(\hat{\boldsymbol{\beta}}_{\mathcal{P}}(\gamma)))}\mathcal{M}$, while the direction of the curve $\hat{\boldsymbol{\beta}}(\gamma)$ is defined by the projection of the tangent residual vector onto the basis vector $\partial_{a_1} \ell(\hat{\boldsymbol{\beta}}(\gamma); \mathbf{Y})$. The curve $\hat{\boldsymbol{\beta}}(\gamma)$ continues as defined above until $\gamma = \gamma^{(2)}$, for which there exists a new predictor, say

[1] $\gamma \geq 0$ is a tuning parameter that controls the size of the coefficients. The increase of γ will shrink the coefficients closer to each other and to zero. In practice, it is usually determined by cross-validation.

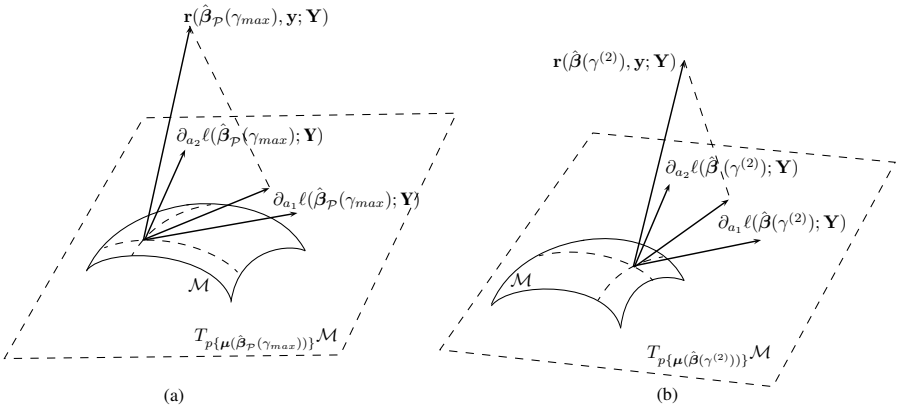


Figure 1: Differential geometrical description of the LARS algorithm with two covariates: (a) the first covariate X_{a_1} is found and included in the active set, where $\hat{\boldsymbol{\beta}}_{\mathcal{P}} = (\hat{\beta}_{a_1^0}, \ldots, \hat{\beta}_{a_b^0})$; (b) the generalized equiangularity condition (7) is satisfied for variables X_{a_1} and X_{a_2}.

X_{a_2}, that satisfies the equiangularity condition, namely

$$\rho_{a_1}(\hat{\boldsymbol{\beta}}(\gamma^{(2)})) = \rho_{a_2}(\hat{\boldsymbol{\beta}}(\gamma^{(2)})). \tag{7}$$

At this point, X_{a_2} is included in the active set $\mathcal{A}(\gamma^{(2)}) = \{a_1, a_2\}$ and the curve $\hat{\boldsymbol{\beta}}(\gamma) = (\hat{\beta}_{a_1^0}(\gamma), \ldots, \hat{\beta}_{a_b^0}(\gamma), \hat{\beta}_{a_1}(\gamma), \hat{\beta}_{a_2}(\gamma), 0, \ldots, 0)^\top$ continues, such that the tangent residual vector is always orthogonal to the basis vectors $\partial_{j\in\mathcal{P}}\ell(\hat{\boldsymbol{\beta}}(\gamma); \mathbf{Y})$ and with direction defined by the tangent residual vector that bisects the angle between $\partial_{a_1}\ell(\hat{\boldsymbol{\beta}}(\gamma); \mathbf{Y})$ and $\partial_{a_2}\ell(\hat{\boldsymbol{\beta}}(\gamma); \mathbf{Y})$, as shown on the right side of Figure 1.

The extended dgLARS solution curve, which is denoted by $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma) \subset \mathbb{R}^{b+k(\gamma)}$ where $\gamma \in [0, \gamma^{(1)}]$ and $0 \leqslant \gamma^{(p-b+1)} \leqslant \cdots \leqslant \gamma^{(2)} \leqslant \gamma^{(1)}$, is defined in the following way: for any $\gamma \in (\gamma^{(k+1)}, \gamma^{(k)}]$, the extended dgLARS estimator satisfies the following conditions:

$$\begin{aligned}
\mathcal{A}(\gamma) &= \{a_1, a_2, \cdots, a_{k(\gamma)}\}, \\
\mathcal{N}(\gamma) &= \{a_1^c, a_2^c, \cdots, a_{h(\gamma)}^c\}, \\
|r_{a_i}(\hat{\boldsymbol{\beta}}(\gamma))| &= |r_{a_j}(\hat{\boldsymbol{\beta}}(\gamma))| = \gamma\ , && \forall a_i, a_j \in \mathcal{A}(\gamma)\ , \\
r_{a_i}(\hat{\boldsymbol{\beta}}(\gamma)) &= s_{a_i} \cdot \gamma, && \forall a_i \in \mathcal{A}(\gamma)\ , \\
|r_{a_l^c}(\hat{\boldsymbol{\beta}}(\gamma))| &< |r_{a_i}(\hat{\boldsymbol{\beta}}(\gamma))| = \gamma, && \forall a_l^c \in \mathcal{N}(\gamma) \text{ and } \forall a_i \in \mathcal{A}(\gamma),
\end{aligned} \tag{8}$$

where $s_{a_i} = \text{sign}\{r_{a_i}(\hat{\boldsymbol{\beta}}(\gamma))\}$, $k(\gamma) = |\mathcal{A}(\gamma)| = \#\{m : \hat{\beta}_m(\gamma) \neq 0\}$ and $h(\gamma) = |\mathcal{N}(\gamma)| = \#\{m : \hat{\beta}_m(\gamma) = 0\}$ are the number of covariates in the active and non-active sets, respectively, at location γ. The new covariate is included in the active set at $\gamma = \gamma^{(k+1)}$ when the following condition is satisfied:

$$\exists a_l^c \in \mathcal{N}(\gamma^{(k+1)}) : \quad |r_{a_l^c}(\hat{\boldsymbol{\beta}}(\gamma^{(k+1)}))| = |r_{a_i}(\hat{\boldsymbol{\beta}}(\gamma^{(k+1)}))|\ , \qquad \forall a_i \in \mathcal{A}(\gamma^{(k+1)}). \tag{9}$$

It shows that the generalized equiangularity condition (8) does not depend on the value of the dispersion parameter. As mentioned before, the original `dglars` package (Augugliaro, 2014b) is developed only for Poisson and logistic regression models with canonical link

function and $\phi = 1$. Although, the value of the dispersion parameter ϕ does not change the order of the variables included in the active set and also the solution path $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$, it is important to take it into consideration that it causes the achieved Rao's score statistic to be shrunk or expanded, since it affects the value of the log-likelihood function $\ell(\boldsymbol{\beta}, \phi; \mathbf{y})$. Therefore, the important point to note here is that the value of the dispersion parameter affects the value of various information criteria such as AIC or BIC, and that is why the estimation of the dispersion parameter is critically important, and will be dealt with in Sections 2.4 and 2.5.

It is worth noting that in a high-dimensional setting, $n \leq p$, it is often assumed that the true model, $\mathcal{A}_0 = \{m : \beta_m \neq 0\}$, is sparse, i.e., the number of non-zero coefficients $|\mathcal{A}_0|$ is small (any number less than $\min(n-1, p)$). In fact, the maximum number of variables that the dgLARS method can include in the active set is $\min(n-1, p)$, namely $|\mathcal{A}| \leq \min(n-1, p)$. Hence, when $n \leq p$, the maximum number of non-zero coefficients selected by dgLARS method is $\min(n-1, p) = n-1$, namely $|\mathcal{A}| \leq n-1$. It means that, when $n \leq p$, the dgLARS method does not consider the cases in which $n \leq |\mathcal{A}_0|$, thus, we assume that $|\mathcal{A}_0| < n$.

2.3 Improved Predictor-Corrector algorithm

To compute the solution curve we can use the Predictor-Corrector (PC) algorithm (Allgower and Georg, 2003), which explicitly finds a series of solutions by using the initial conditions (solutions at one extreme value of the parameter) and continuing to find the adjacent solutions on the basis of the current solutions. From a computational point of view, using the standard PC algorithm lead to an increase in the run times needed for computing the solution curve. In this section we propose an improved version of the PC algorithm to decrease the effects stemming from this problem for computing the solution curve. Using the improved PC algorithm leads to potentially computational saving.

The PC method computes the exact coefficients at the values of γ at which the set of non-zero coefficients changes. This strategy yields a more accurate path in an efficient way than alternative methods and provides the exact order of the active set changes. Let us suppose that $k(\gamma)$ predictors are included in the active set $\mathcal{A}(\gamma) = \{a_1, \cdots, a_{k(\gamma)}\}$ at location γ, such that $\gamma \in (\gamma^{(k+1)}, \gamma^{(k)}]$ be a fixed value of the tuning parameter. The corresponding point of the solution curve will be denoted by $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma) = (\hat{\boldsymbol{\beta}}_{\mathcal{P}}(\gamma), \hat{\beta}_{a_1}(\gamma), \ldots, \hat{\beta}_{a_{k(\gamma)}}(\gamma))^{\top}$ where $\hat{\boldsymbol{\beta}}_{\mathcal{P}}(\gamma) = (\hat{\beta}_{a_1^0}(\gamma), \ldots, \hat{\beta}_{a_b^0}(\gamma))$ where b is the number of protected variables. Using (8), the extended dgLARS solution curve $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ satisfies the relationship

$$|r_{a_1}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))| = |r_{a_2}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))| = \cdots = |r_{a_{k(\gamma)}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))|, \tag{10}$$

and is implicitly defined by the following system of $k(\gamma) + b$ non-linear equations:

$$\begin{cases} \partial_{a_1^0} \ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma); \mathbf{y}) &= 0\ , \\ \vdots & \vdots \\ \partial_{a_b^0} \ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma); \mathbf{y}) &= 0\ , \\ r_{a_1}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) &= \upsilon_{a_1}\gamma\ , \\ \vdots & \vdots \\ r_{a_{k(\gamma)}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) &= \upsilon_{a_{k(\gamma)}}\gamma\ . \end{cases} \tag{11}$$

where $\upsilon_{a_i} = \text{sign}\{r_{a_i}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))\}$.

When $\gamma = 0$, we obtain the maximum likelihood estimates of the subset of the parameter vector $\boldsymbol{\beta}$, denoted by $\hat{\boldsymbol{\beta}}_{\mathcal{A}}$, of the covariates in the active set. The point $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma^{(k+1)})$ lies on the solution curve joining $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma^{(k)})$ with $\hat{\boldsymbol{\beta}}_{\mathcal{A}}$. We define $\tilde{\boldsymbol{\varphi}}_{\mathcal{A}}(\gamma) = \boldsymbol{\varphi}_{\mathcal{A}}(\gamma) - \mathbf{v}_{\mathcal{A}}\gamma$, where $\boldsymbol{\varphi}_{\mathcal{A}}(\gamma) = (\partial_{a_1^0}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y}), \ldots, \partial_{a_b^0}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y}), r_{a_1}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)), \cdots, r_{a_{k(\gamma)}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)))^\top$ and $\mathbf{v}_{\mathcal{A}} = (0, \ldots, 0, \upsilon_{a_1}, \ldots, \upsilon_{a_{k(\gamma)}})^\top$ starting with b zeros. By differentiating $\tilde{\boldsymbol{\varphi}}_{\mathcal{A}}(\gamma)$ with respect to γ, we can locally approximate the solution curve at $\gamma - \Delta\gamma$ by the following expression

$$\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma - \Delta\gamma) \approx \tilde{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma - \Delta\gamma) = \hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma) - \Delta\gamma \cdot \left(\frac{\partial \boldsymbol{\varphi}_{\mathcal{A}}(\gamma)}{\partial \hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)}\right)^{-1} \mathbf{v}_{\mathcal{A}}\ , \tag{12}$$

where $\Delta\gamma \in [0; \gamma - \gamma^{(k+1)}]$ and $\frac{\partial \boldsymbol{\varphi}_{\mathcal{A}}(\gamma)}{\partial \hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)}$ is the Jacobian matrix of the vector function $\boldsymbol{\varphi}_{\mathcal{A}}(\gamma)$ evaluated at the point $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$. Equation (12) with the step size given in (15) is used for the predictor step of the PC algorithm. In the corrector step, $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma - \Delta\gamma)$ is used as starting point for the Newton-Raphson algorithm that is used to solve (11). For obtaining the Jacobian matrix we need $\partial_m r_n(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma), \phi)$, which is as follows:

$$\begin{aligned}
\partial_m r_n(\boldsymbol{\beta}, \phi) &= \frac{\partial\ r_n(\boldsymbol{\beta}, \phi)}{\partial \beta_m} \\
&= \frac{\partial_{mn}\ell(\boldsymbol{\beta}, \phi; \mathbf{y})}{\sqrt{\mathcal{I}_n(\boldsymbol{\beta}, \phi)}} - \frac{1}{2}\ \frac{r_n(\boldsymbol{\beta}, \phi)\ \partial_m \mathcal{I}_n(\boldsymbol{\beta}, \phi)}{\mathcal{I}_n(\boldsymbol{\beta}, \phi)} = \phi^{-1}\ \partial_m r_n(\boldsymbol{\beta}),
\end{aligned}$$

where $m, n \in \mathcal{A}$ and

$$\begin{aligned}
\partial_m \mathcal{I}_n(\boldsymbol{\beta}, \phi) &= \frac{\partial\ \mathcal{I}_n(\boldsymbol{\beta}, \phi)}{\partial \beta_m} \\
&= \phi^{-1} \sum_{i=1}^{n} \left\{ \frac{x_{im}\ x_{in}^2}{V(\mu_i)} \left(2\ \frac{\partial \mu_i}{\partial \eta_i} \cdot \frac{\partial^2 \mu_i}{\partial \eta_i^2} - \frac{\partial V(\mu_i)/\partial \mu_i}{V(\mu_i)} \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^3 \right) \right\} = \phi^{-1}\ \partial_m \mathcal{I}_n(\boldsymbol{\beta}).
\end{aligned} \tag{13}$$

An efficient implementation of the PC method requires a suitable method to compute the smallest step size $\Delta\gamma$ that changes the active set of the non-zero coefficients. Using (9), we have a change in the active set when

$$\exists a_j^c \in \mathcal{N}(\gamma): \quad |r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma - \Delta\gamma))| = |r_{a_i}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma - \Delta\gamma))|, \qquad \forall a_i \in \mathcal{A}(\gamma). \tag{14}$$

By expanding $r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))$ in a Taylor series around γ, and observing that the solution curve satisfies (11), expression (14) can be rewritten in the following way:

$$\exists a_j^c \in \mathcal{N}(\gamma): \quad \left| r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) - \frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma} \Delta\gamma \right| \approx \gamma - \Delta\gamma, \quad \forall a_i \in \mathcal{A}(\gamma) \text{ and } \Delta\gamma \in [0; \gamma]$$

then

$$r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \approx \frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma} \Delta\gamma + (\gamma - \Delta\gamma) = -\Delta\gamma \left(1 - \frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma} \right) + \gamma,$$

or

$$r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \approx \frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma}\Delta\gamma - (\gamma - \Delta\gamma) = \Delta\gamma\left(1 + \frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma}\right) - \gamma,$$

so that, they give two values for $\Delta\gamma$, namely

$$\Delta\gamma_1 = \frac{\gamma - r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{1 - \dfrac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma}} \qquad \text{or} \qquad \Delta\gamma_2 = \frac{\gamma + r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{1 + \dfrac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma}},$$

where

$$\begin{aligned}
\frac{dr_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma} &= \frac{d}{d\gamma}\left(\frac{\partial_{a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y})}{\sqrt{\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}}\right) \\
&= \frac{\frac{d\ \partial_{a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y})}{d\gamma} \cdot \mathcal{I}^{1/2}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) - \partial_{a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y}) \cdot \frac{d\ \mathcal{I}^{1/2}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{d\gamma}}{\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))} \\
&= \mathcal{I}^{-1/2}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \langle \partial_{a_i a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y}), \frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}\rangle \\
&- \frac{1}{2}\ r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \mathcal{I}^{-1}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \langle \partial_{a_i}\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)), \frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}\rangle\ , \\
&= \frac{\sum_{a_i \in \mathcal{A}(\gamma)}\left\{\partial_{a_i a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y}) \cdot \frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}\right\}}{\mathcal{I}^{1/2}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))} \\
&- \frac{1}{2}\ \frac{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \sum_{a_i \in \mathcal{A}}\left\{\partial_{a_i}\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}\right\}}{\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))} \\
&= \sum_{a_i \in \mathcal{A}(\gamma)}\left\{\frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}\left[\frac{\partial_{a_i a_j^c}\ell(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma);\mathbf{y})}{\mathcal{I}^{1/2}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))} - \frac{1}{2}\ \frac{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) \cdot \partial_{a_i}\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{\mathcal{I}_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}\right]\right\},
\end{aligned}$$

where $\langle\cdot,\cdot\rangle$ is an inner product, $\partial_{a_i}\mathcal{I}_{a_j^c}(\boldsymbol{\beta})$ is given by (13), and $\frac{d\hat{\beta}_{a_i}(\gamma)}{d\gamma}$ is an element of the matrix of $\frac{d\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)}{d\gamma} = \left(\frac{\partial\boldsymbol{\varphi}_{\mathcal{A}}(\gamma)}{\partial\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)}\right)^{-1}\mathbf{v}_{\mathcal{A}}$. For each $a_j^c \in \mathcal{N}(\gamma)$ we have a value for $\Delta\gamma^{a_j^c}$ as follows

$$\Delta\gamma^{a_j^c} = \begin{cases} \Delta\gamma_1 & \text{if} \quad 0 \le \Delta\gamma_1 \le \gamma; \\ \Delta\gamma_2 & \text{if} \quad o.w. \end{cases}$$

and from the set of $\Delta\gamma^{a_j^c}$s, $\{\Delta\gamma^{a_j^c},\ a_j^c \in \mathcal{N}(\gamma)\}$, we consider the smallest value of this set as a optimal value for the step size. It can be shown by the following expression

$$\Delta\gamma^{opt} = \min\left\{\Delta\gamma^{a_j^c} \mid a_j^c \in \mathcal{N}(\gamma)\right\}. \tag{15}$$

The main problem of the original PC algorithm is related to the number of arithmetic operations needed to compute the Euler predictor, which requires the inversion of an adequate Jacobian matrix. From a computational point of view, using the PC algorithm leads to an increase in the run times needed to compute the solution curve. We propose a method to improve the PC algorithm to reduce the number of steps, thereby greatly reducing the computational burden because of reducing the number of points of the solution curve.

Since the optimal step size is based on a local approximation, we also include an exclusion step for removing incorrectly included variables in the model. When an incorrect variable is included in the model after the corrector step, we have that there is a non-active variable such that the absolute value of the corresponding Rao score test statistic is greater than γ. To adjust the step size in the case of incorrectly including certain variables in the active set, Augugliaro *et al.* (2013) reduced the optimal step size from the previous step, $\triangle\gamma^{opt}$, by using a small positive constant ε and then the inclusion step is redone until the correct variable is joined to the model. They proposed a half of $\Delta\gamma^{opt}$ for ε as a possible choice. Augugliaro *et al.* (2013, 2014a) and Augugliaro (2014b) used a contractor factor cf, which is a fixed value, (i.e., $\gamma_{cf} = \gamma_{old} - \Delta\gamma$, where $\gamma_{old} = \gamma_{new} + \triangle\gamma^{opt}$ and $\Delta\gamma = \Delta\gamma^{opt} \cdot cf$), where $cf = 0.5$ as a default. In this case, this method acts like a *Bisection* method. However, the predicted root, γ_{cf}, may be closer to γ_{new}, or γ_{old}, than the mid-point between them. The poor convergence of the Bisection method as well as its poor adaptability to higher dimensions (i.e., systems of two or more non-linear equations) motivate the use of better techniques. In this case, we apply the method of Regula-Falsi (or False-Position), which always converges, for more details see Press *et al.* (1992) and Whittaker and Robinson (1967). The regula-falsi method uses the information about the function, $h(.)$, to arrive at γ_{rf}, while in the case of the Bisection method finding γ is a *static* procedure since for a given γ_{new} and γ_{old}, it gives *identical* γ_{cf}, no matter what the function we wish to solve.

The regula-falsi method draws a secant from $h(\gamma_{new})$ to $h(\gamma_{old})$, and estimates the root as where it crosses the γ-axis, so that in our case $h(\gamma) = r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) - s_{a_j^c} \cdot \gamma$ where $s_{a_j^c} = \text{sign}\{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))\}$ and $a_j^c \in \mathcal{N}(\gamma)$. From (8), we have that $h(\gamma) = r_{a_i}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) - s_{a_i}\gamma = 0$ for all $a_i \in \mathcal{A}(\gamma)$. Indeed, after the corrector step, when there is a non-active variable such that the absolute value of the corresponding Rao score test statistic is greater than γ, we want to find a exact point, γ_{rf}, which is very close or even equal to the true point, called transition point, that changes the active set, so that at the end, it reduces the number of the points of the solution curve.

For applying the regula-falsi method to find the root of the equation $h(\gamma_{rf}) = 0$, let us suppose that k predictors are included in the active set, such that $\gamma_{new} < \gamma^{(k)}$. After the corrector step, when $\exists a_j^c \in \mathcal{N}(\gamma_{new})$ such that $|r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))| > \gamma_{new}$, we find an γ_{rf} in the interval $[\gamma_{new}, \gamma_{old}]$, where $\gamma_{old} = \gamma_{new} + \triangle\gamma^{opt}$, which is given by the intersection of the γ-axis and the straight line passing through $(\gamma_{new}, r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new})) - s_{a_j^c} \cdot \gamma_{new})$ and $(\gamma_{old}, r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{old})) - s_{a_j^c} \cdot \gamma_{old})$ where $s_{a_j^c} = \text{sign}\{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))\}$. It is easy to verify that the root γ_{rf} is given by

$$\gamma_{rf} = \frac{\gamma_{new}\ r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{old})) - \gamma_{old}\ r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))}{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{old})) - r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new})) + s_{a_j^c} \cdot (\gamma_{new} - \gamma_{old})}, \qquad \forall a_j^c \in \mathcal{N}(\gamma_{new}), \quad (16)$$

where $s_{a_j^c} = \text{sign}\{r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))\}$. Then, we first set $\triangle\gamma = \triangle\gamma^{opt} - (\gamma_{rf} - \gamma_{new})$ and then $\gamma = \gamma_{rf}$, to be able to go to the predictor step.

Table 1: Pseudo-code of the improved PC algorithm to compute the solution curve defined by the extended dgLARS method for a model with the protected variables.

Step	Algorithm
1	First compute $\hat{\boldsymbol{\beta}}_{\mathcal{P}} = (\hat{\beta}_{a_1^0}, \ldots, \hat{\beta}_{a_b^0})$
2	$\mathcal{A} \leftarrow \arg\max_{a_j^c \in \mathcal{N}(\gamma)} \{\lvert r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{P}})\rvert\}$ and $\gamma \leftarrow \lvert r_{a_1}(\hat{\boldsymbol{\beta}}_{\mathcal{P}})\rvert$
3	Repeat
4	Use (15) to compute $\triangle\gamma^{opt}$ and set $\triangle\gamma \leftarrow \triangle\gamma^{opt}$ and $\gamma \leftarrow \gamma - \triangle\gamma^{opt}$
5	Use (12) to compute $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ (*predictor step*)
6	Use $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ as starting point to solve system (11) (*corrector step*)
7	For all $a_j^c \in \mathcal{N}(\gamma)$ compute $r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))$
8	If $\exists N \subset \mathcal{N}(\gamma)$ such that $\left\lvert r_{a_l^{c*}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))\right\rvert > \gamma$ for all $a_l^{c*} \in N$, then
9	use (16) to compute $\gamma_{rf}^{(l)}$ and set $\gamma_{rf} \leftarrow \max_l\{\gamma_{rf}^{(l)}\}$
10	first set $\triangle\gamma \leftarrow \triangle\gamma^{opt} - (\gamma_{rf} - \gamma)$ and then $\gamma \leftarrow \gamma_{rf}$, and go to step 5
11	If $\exists a_j^c \in \mathcal{N}(\gamma)$ such that $\left\lvert r_{a_j^c}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))\right\rvert = \left\lvert r_{a_i}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))\right\rvert$ for all $a_i \in \mathcal{A}(\gamma)$, then
12	update $\mathcal{A}(\gamma)$ and $\mathcal{N}(\gamma)$
13	Until convergence criterion rule is met

If at γ_{new} there exists a set $N(\gamma_{new}) \subset \mathcal{N}(\gamma_{new})$ such that $|r_{a_l^{c*}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma_{new}))| > \gamma_{new}$ for all $a_l^{c*} \in N(\gamma_{new})$, the equation (16) gives a vector with an element of $\gamma_{rf}^{(l)}$, so that we consider $\gamma_{rf} = \max_l\{\gamma_{rf}^{(l)}\}$, and if $\max_l\{\gamma_{rf}^{(l)}\}$ is greater than γ_{old}, then we consider $\gamma_{rf} = \gamma_{old}$. When the Newton-Raphson algorithm does not converge, the step size is reduced by the contractor factor cf, and then the predictor and corrector steps are repeated.

In Table 1 we report the pseudo-code of the improved PC algorithm that was proposed in this section for a model with the protected variables $\{a_1^0, \ldots a_b^0\}$. In Section 4.1, we examine the performance of the proposed PC algorithm and compare it with the original PC algorithm.

2.4 Path estimation of dispersion parameter

Since in practice the dispersion parameter ϕ is often unknown, in this paper we consider ϕ as an unknown parameter which is the same for all Y_i. As we mentioned before, by estimating the dispersion parameter, the solution path $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ is not changed, although the value of the log-likelihood function $\ell(\boldsymbol{\beta}, \phi; \mathbf{y})$ is changed and so considerations about the selection of the optimal model are going to be importantly affected.

There are three classical methods to estimate ϕ: Deviance, Pearson and Maximum Likelihood (ML) estimators. The Deviance estimator is $\hat{\phi}_d = \mathcal{D}(\mathbf{y}, \hat{\boldsymbol{\mu}})/(n-p)$, where $\mathcal{D}(\mathbf{y}, \hat{\boldsymbol{\mu}}) = \phi\mathcal{D}(\mathbf{y}, \hat{\boldsymbol{\mu}}, \phi) = -2\phi(\ell(\hat{\boldsymbol{\mu}}, \phi; \mathbf{y}) - \ell(\mathbf{y}, \phi; \mathbf{y}))$ is the unscaled residual deviance. The ML esti-

mator of ϕ ($\hat{\phi}_{mle}$) is the solution of $\partial\ell(\hat{\boldsymbol{\mu}}, \phi; \mathbf{y})/\partial\phi = 0$; For instance, the ML estimators for the Gamma and Inverse Gaussian distributions are $\hat{\phi}_{mle,G} \approx 2\mathcal{D}_G/(n + \sqrt{(n^2 + 2n\mathcal{D}_G/3)})$ and $\hat{\phi}_{mle,IG} = \mathcal{D}_{IG}$, where $\mathcal{D}_G$ and $\mathcal{D}_{IG}$ are $\mathcal{D}(\mathbf{y}, \hat{\boldsymbol{\mu}})$ for the Gamma and Inverse Gaussian distributions, respectively (Cordeiro and McCullagh, 1991). McCullagh and Nelder (1989) note for the Gamma case that both the Deviance ($\hat{\phi}_{d,G}$) and MLE ($\hat{\phi}_{mle,G}$) are sensitive to rounding errors (the difference between the calculated approximation of a number and its exact mathematical value) and model error (deviance from the chosen model) in very small observations and in fact deviance is infinite if any component of $\mathbf{y}$ is zero. Commonly used estimates of the unknown dispersion parameter are the Pearson statistic or the modification of Farrington (1996), who proposed a first order linear correction term to Pearson's statistic. McCullagh and Nelder (1989) recommend the use of an approximately unbiased estimate, Pearson method, $\hat{\phi}_{P^*} = \frac{\mathcal{X}^2_P}{n-p} = \frac{1}{n-p}\sum_{i=1}^{n}\frac{(y_i-\hat{\mu}_i)^2}{V(\hat{\mu}_i)}$, where $\mathcal{X}^2_P$ is the Pearson's statistic, $V(.)$ is the variance function, and $\hat{\mu}_i = g^{-1}(\mathbf{x}_i^\top\hat{\boldsymbol{\beta}})$. Meng (2004) shows numerically that the choice of estimator can give quite different results in the Gamma case and that $\hat{\phi}_{P^*}$ is more robust against model error. Since we can use $\hat{\phi}_{P^*}$ only for $n > p$, in the high-dimensional setting ($p \geq n$) we define the dispersion estimator $\hat{\phi}_P(\gamma)$ at $\gamma \in [0, \gamma_{max}]$ by the Pearson-like dispersion estimator, as proposed by Wood (2006) and Ultricht and Tutz (2011);

$$\hat{\phi}_P(\gamma) = \frac{1}{n - k(\gamma)}\sum_{i=1}^{n}\frac{(y_i - g^{-1}(\mathbf{x}_i^\top\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)))^2}{V(g^{-1}(\mathbf{x}_i^\top\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)))}, \tag{17}$$

where $k(\gamma) = |\mathcal{A}(\gamma)| = \#\{j : \hat{\beta}_j(\gamma) \neq 0\}$ such that $\hat{\beta}_j(\gamma)$ is the element of the extended dgLARS estimator $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$. Note that, since the estimator $\hat{\phi}_P(\gamma)$ depends on γ, we can apply it into the improved PC algorithm in order to calculate the value of the information criteria such as AIC and BIC at each path point (γ), so that $AIC(\gamma)$ and $BIC(\gamma)$ are given in (19) and (20).

2.5 Model selection

Model selection is a process of seeking the model in a set of candidate models that gives the best balance between model fit and complexity (Burnham and Anderson, 2002). In the literature, selection criteria are usually classified into two categories: consistent (e.g., the Bayesian information criterion (BIC) (Schwarz, 1978)) and efficient (e.g., the Akaike information criterion (AIC) (Akaike, 1974), and the k-fold cross-validation (CV) (Hastie *et al.*, 2009)). A consistent criterion identifies the true model with a probability that approaches 1 in large samples when a set of candidate models contains the true model. An efficient criterion selects the model so that its average squared error is asymptotically equivalent to the minimum offered by the candidate models when the true model is approximated by a family of candidate models. Detailed discussions on efficiency and consistency can be found in Shibata (1981, 1984), Li (1987), Shao (1997), McQuarrie and Tsai (1998), and Arlot and Celisse (2010).

Stone (1977) shows that the AIC is asymptotically equivalent to Leave-One-Out CV. Both of these criteria are based on the Kullback-Leibler information criteria (Kullback and Leibler, 1951). While the BIC, which is based on the Bayesian posterior probability, is asymptotically equivalent to v-fold CV, where $v = n[1 - 1/(\log(n) - 1)]$. Actually, it is

well-known that CV on the original models behaves somewhere between AIC and BIC, depending on the data splitting ratio (Shao, 1997). In Section 5, we will compare the performance of these three criteria when the extended dgLARS method is involved as a variable selection method. The dgLARS approach involves the choice of a tuning parameter for variable selection. The selection of the tuning parameter γ is critically important because it determines the dimension of the selected model. A proper tuning parameter can improve the efficiency and accuracy for variable selection (Chen *et al.*, 2014). As an all-round option, the k-fold CV has always been a popular choice, especially in the early years. In the present paper, we use the k-fold CV deviance for the extended dgLARS, so that, data are randomly split into k arbitrary equal-sized subsets $L_1, L_2, \ldots, L_k$ and each subset L_v, $v = 1, \ldots, k$, is used as an validation data set $L_v = (\mathbf{y}_{n_v}^{(v)}, \mathbf{X}_{n_v \times p}^{(v)})$ consisting of n_v sample points (and its complement L_v^c is the v^{th} training data set consisting of the remaining n_t observations, where $n_v + n_t = n$) to evaluate the performance of each of the models fitted to the remaining $(k-1)/k$ of the data, L_v^c. The unscaled residual deviance $\mathcal{D}(.,.)$ of the predictions on the validation data set L_v is computed and averaged for the k validation subsets;

$$CV(\gamma) = \frac{1}{k} \sum_{v=1}^{k} \mathcal{D}(\mathbf{y}^{(v)}, \hat{\boldsymbol{\mu}}^{(v)}), \tag{18}$$

where $\hat{\boldsymbol{\mu}}^{(v)} = g^{-1}(\mathbf{X}^{(v)} \hat{\boldsymbol{\beta}}_{\mathcal{A}_v}(\gamma))$ and $\hat{\boldsymbol{\beta}}_{\mathcal{A}_v}(\gamma)$ is selected by L_v. The idea will be to select the model with the lowest average CV deviance.

Classical information criteria such as the AIC and BIC can also be used. We use the $AIC(\gamma)$ and $BIC(\gamma)$ for the extended dgLARS written as

$$AIC(\gamma) = -2\ell(\boldsymbol{\beta}_{\mathcal{A}}(\gamma), \phi; \mathbf{y}) + 2\ (k(\gamma) + 1)\,, \tag{19}$$

and

$$BIC(\gamma) = -2\ell(\boldsymbol{\beta}_{\mathcal{A}}(\gamma), \phi; \mathbf{y}) + \log(n)(k(\gamma) + 1)\,, \tag{20}$$

where $k(\gamma) = |\mathcal{A}(\gamma)|$ is an appropriate degree of freedom that measures complexity of the model with the tuning parameter γ. As it can be seen, the selection criteria 19 and 20 rely heavily on the dispersion parameter which has an important impact on them. Since the log-likelihood function $\ell(\boldsymbol{\beta}(\gamma), \phi; \mathbf{y})$ depends on the unknown dispersion parameter, an estimator (e.g., 17) is needed in order to evaluate these criteria, and as a result $k(\gamma)$ becomes $k(\gamma) + 1$ in the penalty term (Wood, 2006). In Sections 4 and 5, we will use $\hat{\gamma}_{AIC}$, $\hat{\gamma}_{BIC}$ and $\hat{\gamma}_{CV}$, where

$$\begin{aligned} \hat{\gamma}_{AIC} &= \underset{\gamma \in \mathbb{R}^+}{\arg\min}\ AIC(\gamma), \\ \hat{\gamma}_{BIC} &= \underset{\gamma \in \mathbb{R}^+}{\arg\min}\ BIC(\gamma), \\ \hat{\gamma}_{CV} &= \underset{\gamma \in \mathbb{R}^+}{\arg\min}\ CV(\gamma). \end{aligned}$$

The concept of degrees of freedom, which is often used for measurement of model complexity, plays an important role in the theory of linear regression models. This concept is involved in various model selection criteria such as the AIC and BIC. Within the classical

theory of linear regression models, it is well known that the degrees of freedom are equal to the number of covariates but for non-linear modelling procedures this equivalence is not satisfied. Generalized degrees of freedom (GDF) is a generic measure of model complexity for any modeling procedure. It accounts for the cost due to both model selection and parameter estimation. For the dgLARS estimator, Augugliaro *et al.* (2013) proposed the notion of generalized degrees of freedom (GDF) to define an adaptive model selection criterion. The authors showed that the cardinality of the active set, $k(\gamma) = |\mathcal{A}(\gamma)|$, is a biased estimator of the generalized degrees of freedom when the model is a logistic regression model, and also proposed a possible estimator of the GDF when it is possible to compute the MLE of the considered GLM. In general, $\text{gdf}(\gamma)$ is a function of the tuning parameter γ, so that $\text{gdf}(0) \approx p$. This estimator for a general GLM is given by

$$\widehat{\text{gdf}}(\gamma) = \text{tr}\{J_{\mathcal{A}}^{-1}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))\ I_{\mathcal{A}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma), \hat{\boldsymbol{\beta}}_{\mathcal{A}}(0))\}, \tag{21}$$

where $J_{\mathcal{A}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))$ is the unscaled observed Fisher Information matrix evaluated at the point $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ which has elements

$$J_{a_j a_k}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)) = \sum_{i=1}^{n} \frac{x_{ia_j}\ x_{ia_k}}{V(\mu_i)} \left\{ \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 + (y_i - \mu_i) \left(\frac{\partial V(\mu_i)/\partial \mu_i}{V(\mu_i)} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2 - \frac{\partial^2 \mu_i}{\partial \eta_i^2} \right) \right\},$$

and $I_{\mathcal{A}}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma), \hat{\boldsymbol{\beta}}_{\mathcal{A}}(0))$ is an unscaled matrix with elements

$$I_{a_j a_k}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma), \hat{\boldsymbol{\beta}}_{\mathcal{A}}(0)) = \sum_{i=1}^{n} x_{ia_j}\ x_{ia_k}\ \frac{V(\mu_i(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(0)))}{V(\mu_i(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)))^2} \left(\frac{\partial \mu_i(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))}{\partial \eta_i} \right)^2,$$

where $\mu_i(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(0))$ is the maximum likelihood estimate of $\mu_i(\boldsymbol{\beta})$, and $\eta_i = g(\mu_i(\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)))$. Note that, the proposed estimator (21) does not depend on ϕ. In general, $\widehat{\text{gdf}}(\gamma)$ is different from $k(\gamma)$. It can be used, instead of $k(\gamma)$, in the penalty term of (19) and (20) to have alternative criteria, namely, $AIC^*(\gamma) = -2\ell(\boldsymbol{\beta}_{\mathcal{A}}(\gamma), \phi; \mathbf{y}) + 2\ (\widehat{\text{gdf}}(\gamma) + 1)$ and $BIC^*(\gamma) = -2\ell(\boldsymbol{\beta}_{\mathcal{A}}(\gamma), \phi; \mathbf{y}) + \log(n)(\widehat{\text{gdf}}(\gamma) + 1)$.

Although $\hat{\phi}_P(\gamma)$ given in (17) can be used for estimating ϕ to obtain the criteria $AIC(\gamma)$, $BIC(\gamma)$ and k-fold $CV(\gamma)$, in the next section we provide another estimation of ϕ which is fixed on γ.

3 An stable estimation of dispersion parameter

In Section 2.4, we defined a Pearson-type path estimator of the dispersion parameter ϕ. Combined with model selection in Section 2.5 this could be used to estimate ϕ overall, but it is known that in shrinkage situations this under-estimates ϕ. In this section, we first propose an improved estimator of the dispersion parameter for high-dimensional generalized linear models, called General Refitted Cross-Validation (GRCV) estimator. Then, we present an algorithm to improve the proposed GRCV estimator to obtain a more stable and accurate estimator based on the GRCV estimator.

3.1 General Refitted Cross-Validation estimator of dispersion

Fan *et al.* (2012) introduced a two-stage refitted procedure for estimating the dispersion parameter in a linear regression model (variance in linear model) via a data splitting technique called refitted cross-validation (RCV), to attenuate the influence of irrelevant variables with high spurious correlations in the linear models. The RCV estimator is accurate and stable, and insensitive to model selection considerations and the size of the model selected.

For generalized linear models, we propose a general refitted procedure called general refitted cross-validation (GRCV) which is based on four stages. The idea of the GRCV method is as follows; We split the data $(\mathbf{y}_n, \mathbf{X}_{n\times p})$ randomly into two halves $(\mathbf{y}^{(1)}_{n_1}, \mathbf{X}^{(1)}_{n_1\times p})$ and $(\mathbf{y}^{(2)}_{n_2}, \mathbf{X}^{(2)}_{n_2\times p})$, where $n_1 + n_2 = n$. Without loss of generality, for notational simplicity, we assume that the sample size n is even [2], and $n_1 = n_2 = n/2$. In the first stage, our high dimensional variable selection method, extended dgLARS, is applied to these two data sets separately, to estimate whole solution path, which yields $\hat{\boldsymbol{\beta}}_{\mathcal{A}_i}(\gamma)$ selected by $(\mathbf{y}^{(i)}, \mathbf{X}^{(i)})$ where $|\mathcal{A}_i| \leq \min(\frac{n}{2} - 1, p)$, $\gamma \in [0, \gamma_{max}]$, and $i = 1, 2$. In the second stage, by using the Pearson-like dispersion estimate (17) on the two data sets separately, $\hat{\phi}^{(i)}_P(\gamma)$ where $i = 1, 2$, we determine two small subsets of selected variables $\hat{\mathcal{A}}_i$ where $\hat{\mathcal{A}}_i \subseteq \mathcal{A}_i$ and $i = 1, 2$, by model selection tools such as the AIC, on each data set. Although all three criteria mentioned in the present paper are available in our package, we recommend using the AIC criterion because the goal is to have a accurate prediction in the third stage (Aho *et al.*, 2014). In the third stage, the MLE method is applied to each subset of the data with the variables selected by another subset of the data, namely $(\mathbf{y}^{(2)}, \mathbf{X}^{(2)}_{\hat{\mathcal{A}}_1})$ and $(\mathbf{y}^{(1)}, \mathbf{X}^{(1)}_{\hat{\mathcal{A}}_2})$, to re-estimate the coefficient $\boldsymbol{\beta}$. Since the MLE may not always exist in GLMs, in this stage we propose to use the dgLARS method to estimate the coefficients based on the selected variables, $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_1}(\gamma_0)$ and $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_2}(\gamma_0)$, where γ_0 is close to zero, because the dgLARS estimate $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(0)$ is equal to the MLE of $\boldsymbol{\beta}_{\mathcal{A}}$. Therefore, we apply MLE to the first subset of the data with the variables selected by the second subset of the data $(\mathbf{y}^{(1)}, \mathbf{X}^{(1)}_{\hat{\mathcal{A}}_2})$ to obtain $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_2}(0)$, and similarly, we use MLE again for the second data set with the set of important variables selected by the first data set $(\mathbf{y}^{(2)}, \mathbf{X}^{(2)}_{\hat{\mathcal{A}}_1})$ to obtain $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_1}(0)$. The refitting in the third stage is fundamental to reduce the influence of the spurious variables in the second stage of variable selection. Finally, in the fourth stage, we estimate ϕ by averaging the two following estimators on the two data sets $(\mathbf{y}^{(2)}, \mathbf{X}^{(2)}_{\hat{\mathcal{A}}_1})$ and $(\mathbf{y}^{(1)}, \mathbf{X}^{(1)}_{\hat{\mathcal{A}}_2})$;

$$\hat{\phi}_1(\hat{\mathcal{A}}_2) = \frac{1}{\frac{n}{2} - |\hat{\mathcal{A}}_2|} \sum_{i=1}^{\frac{n}{2}} \frac{\left(y^{(1)}_i - g^{-1}\left(\left(\mathbf{x}^{(1)\top}_{i,\hat{\mathcal{A}}_2}\, \hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_2}(0)\right)\right)\right)^2}{V\left(g^{-1}\left(\mathbf{x}^{(1)\top}_{i,\hat{\mathcal{A}}_2}\, \hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_2}(0)\right)\right)},$$

and

$$\hat{\phi}_2(\hat{\mathcal{A}}_1) = \frac{1}{\frac{n}{2} - |\hat{\mathcal{A}}_1|} \sum_{i=1}^{\frac{n}{2}} \frac{\left(y^{(2)}_i - g^{-1}\left(\left(\mathbf{x}^{(2)\top}_{i,\hat{\mathcal{A}}_1}\, \hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_1}(0)\right)\right)\right)^2}{V\left(g^{-1}\left(\mathbf{x}^{(2)\top}_{i,\hat{\mathcal{A}}_1}\, \hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_1}(0)\right)\right)},$$

[2] If n is odd, we can consider $|n_1 - n_2| = 1$, and then we randomly apply one of the member of the larger data set to the smaller data set to both have the same dimension, $n_1 = n_2 = n/2$.

where $\mathbf{x}^{(l)}_{i,\hat{\mathcal{A}}_j}$ is the i^{th} row of the l^{th} subset of the data $\mathbf{X}^{(l)}_{\hat{\mathcal{A}}_j}$, $|\hat{\mathcal{A}}_j| = \#\{k : (\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_j}(\gamma))_k \neq 0\}$, $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_j}(\gamma)$ is the extended dgLARS estimator at γ, so that $\gamma \in [0, \gamma_{max}]$, and $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_j}(0)$ is the MLE estimator. The GRCV estimator is just the average of these two estimators:

$$\hat{\phi}_{GRCV}(\hat{\mathcal{A}}_1, \hat{\mathcal{A}}_2) = \frac{\hat{\phi}_1(\hat{\mathcal{A}}_2) + \hat{\phi}_2(\hat{\mathcal{A}}_1)}{2}. \tag{22}$$

In this procedure, although $\hat{\mathcal{A}}_1$ includes some extra unimportant variables besides the important variables, these extra variables will play minor roles when we estimate ϕ by using the second data set along with refitting since they are just some random unrelated variables over the second data set. Furthermore, even when some important variables are missed in the second stage of model selection, they have a good chance of being well approximated by the other variables selected in the second stage to reduce modeling biases. It should be mentioned that, by applying a variable selection tool, the GRCV estimator is sensitive to the model selection tool and the size of the model selected.

In the meantime, we can extend the GRCV technique to get a more accurate estimator. The first extension is to use a k-fold data splitting technique rather than twofold splitting. We can divide the data into k groups and select the model with all groups except one, which is used to estimate the dispersion with refitting. Although there are now more data in the second stage, there are only $n = k$ data points in the third stage for refitting. This means that the number of variables that are selected in the second stage should be much less than $n = k$. That is why we use $k = 2$. The second extension is using a repeated data splitting procedure; since there are many ways to split the data randomly, many GRCV estimators can be obtained. To reduce the influence of the randomness in the data splitting we may take the average of the resulting estimators. For an extensive review of the RCV method, for the linear models, the reader is referred to Fan and Lv (2008) and Fan *et al.* (2012).

3.2 An iterative GRCV algorithm

In Section 3.1, we proposed the GRCV estimator $\hat{\phi}_{GRCV}$ to estimate ϕ. In this section, we show how the GRCV estimator can be improved to have numerically more stable and accurate behavior. We propose an iterative algorithm which at convergence will also result in more stable and accurate model selection behavior. This algorithm yields a new estimate for ϕ which we call it the MGRCV estimate.

As mentioned in Section 3.1 to obtain the GRCV estimate, in the third stage we need to calculate the value of the AIC, BIC or some k-fold CV criteria which depend on the unknown dispersion parameter itself. Hence, the dispersion parameter has to be estimated and for this we used the Pearson-type estimator $\hat{\phi}_P(\gamma)$ given in (17) inside the extended dgLARS method during the calculation of the solution path. To improve the accuracy of the estimator $\hat{\phi}_{GRCV}$, we propose an algorithm which repeats the process of finding the GRCV estimate iteratively, such that for the $(k+1)^{\text{th}}$ iteration the k^{th} GRCV estimate ($\hat{\phi}^{(k)}_{GRCV}$) is used to compute the new $(k+1)^{\text{th}}$ GRCV estimate ($\hat{\phi}^{(k+1)}_{GRCV}$), and so on. Therefore, by using this algorithm, the GRCV estimator uses the Pearson-type estimate inside its process only for the first time, and after that the algorithm applies the obtained GRCV estimates instead of the Pearson-type estimate inside the extended dgLARS algorithm. Since the estimate contains some random variation due to the random CV splits, D_1 and D_2, the algorithm

Table 2: Pseudo code for the iterative algorithm to stabilize the GRCV estimator with T iterations.

Step	Algorithm
1	$pearson \leftarrow 1$
2	$grcv.vec \leftarrow 0$
3	$i \leftarrow 1$
4	**while** $i \leq T$
5	split the data into two random groups: D_1 and D_2
6	apply the extended dgLARS to D_1 and D_2 separately to obtain whole solution paths $\hat{\boldsymbol{\beta}}_{\mathcal{A}_1}(\gamma)$ and $\hat{\boldsymbol{\beta}}_{\mathcal{A}_2}(\gamma)$ (first stage)
7	**if** $pearson = 1$ **then**
8	use (17) to compute $\hat{\phi}_P^{(1)}(\gamma)$ and $\hat{\phi}_P^{(2)}(\gamma)$ for D_1 and D_2
9	use $\hat{\phi}_P^{(1)}(\gamma)$ and $\hat{\phi}_P^{(2)}(\gamma)$ to do model selection on D_1 and D_2, respectively, to obtain $\hat{\mathcal{A}}_1$ and $\hat{\mathcal{A}}_2$ (second stage)
10	$pearson \leftarrow 0$
11	**else**
12	use $\hat{\phi}_{GRCV}(\hat{\mathcal{A}}_1, \hat{\mathcal{A}}_2)$ for model selection on each D_1 and D_2 to obtain $\hat{\mathcal{A}}_1$ and $\hat{\mathcal{A}}_2$ (second stage)
13	**end if**
14	apply again extended dgLARS to D_1 and D_2 separately to obtain $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_1}(0)$ and $\hat{\boldsymbol{\beta}}_{\hat{\mathcal{A}}_2}(0)$ (third stage)
15	use (22) to compute $\hat{\phi}_{GRCV}(\hat{\mathcal{A}}_1, \hat{\mathcal{A}}_2)$ (fourth stage)
16	$grcv.vec[\ i\ ] \leftarrow \hat{\phi}_{GRCV}(\hat{\mathcal{A}}_1, \hat{\mathcal{A}}_2)$
17	$i \leftarrow i + 1$
18	**end while**
19	$\hat{\phi}_{MGRCV} \leftarrow$ median($grcv.vec$)
20	use $\hat{\phi}_{MGRCV}$ to do model selection

will not numerically converge, one in practice simply needs to define a maximal number of iterations T (which should not be too large). Therefore we propose as final GRCV estimate the median of the T GRCV estimates, for which we call it MGRCV estimate, $\hat{\phi}_{MGRCV} = \text{median}\{\hat{\phi}_{GRCV}^{(1)}, \ldots, \hat{\phi}_{GRCV}^{(T)}\}$. The MGRCV estimate $\hat{\phi}_{MGRCV}$ is more stable and accurate than the first estimate $\hat{\phi}_{GRCV}^{(1)}$. Finally, the overall model selection is performed using $\hat{\phi}_{MGRCV}$.

Table 2 shows how this algorithm works. It should be mentioned that, $\hat{\phi}_P^{(1)}(\gamma)$ and $\hat{\phi}_P^{(2)}(\gamma)$ are vectors of the estimates calculated during the solution path, while $\hat{\phi}_{GRCV}(\hat{\mathcal{A}}_1, \hat{\mathcal{A}}_2)$ is a fixed number. In order to investigate the performance of the algorithm we test it on simulated data in Section 4.2.

4 Simulation studies

The simulation studies are divided into two parts: the studies on the extended dgLARS method and the GRCV estimator. The first part is devoted to examining the performance of the extended dgLARS method, which uses the improved PC algorithm, and two other popular path-estimation methods. The second part is devoted to investigating the performance of the GRCV estimator based on the iterative GRCV algorithm.

4.1 Comparison of extended dgLARS with other methods

In this section, we compare the behavior of the extended dgLARS method obtained by using the improved PC algorithm (by a new package [3]) with two of the most popular sparse GLM packages; `dglars`: the dgLARS method obtained by using the PC algorithm (Augugliaro, 2014b), and `glmpath`: the L_1 Regularization Path method obtained by using the PC algorithm developed by Park and Hastie (2007b). The `dglars` package is available for the binomial and Poisson families with the canonical link function, and the `glmpath` package is available for the Gaussian, binomial and Poisson families with the canonical link function. To make the results comparable, the simulation study is based on a *Logistic* regression model (binomial family with *logit* link), with sample size $n = (50, 200)$ and three different values of p, namely $p = (10, 100, 500)$. The large values of p are useful to study the behavior of the methods in a high dimensional setting. The study is based on three different configurations of the covariance structure of the p predictors, such that $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{n^*}$ are sampled from an $N(\mathbf{0}, \Sigma)$ distribution, where the diagonal elements of Σ are 1 and the off-diagonal elements follow $corr(X_i; X_j) = \rho^{|i-j|}$, where X_i and X_j are the i^{th} and j^{th} covariates respectively, $i \neq j$ and $\rho = (0, 0.5, 0.75)$. Only the first five predictors are used to simulate the binary response variable. The intercept term is equal to one and the non-zero coefficients are equal to two. We simulate $n^* = 100$ data sets and let the algorithms compute the entire path of the coefficient estimates.

In Table 3 we report the mean number of the points of the whole solution curve (q) and the area under the receiver operating characteristic (ROC) curve (AUC, average AUC over 100 simulations), as the performance measure. A higher AUC indicates a better performance. The results show that, in the dgLARS method with both the original PC (PC) and improved PC (IPC) algorithms, when the number of predictors is sufficiently large, the mean number of the points of the solution curve (q) decreases as the correlation (ρ) increases. However, for the L_1 Regularization Path method, when $n < p$, q decreases as ρ increases, and when $n > p$ then q increases as ρ decreases. The dgLARS method obtained by using the IPC algorithm, in all scenarios, has the lowest q identified by the bold values, which leads to potentially computational saving.

Note that since the dgLARS method obtained by using the improved PC and original PC algorithms compute the same solution curve, their ROC curves and then the values of their AUC are equal, as it can be seen in the corresponding AUC columns of the dgLARS (IPC) and dgLARS (PC). The AUC value of the dgLARS (PC or IPC) method is always greater or equal than the L_1 Regularization Path method. In fact, without depending on p, when the sample size n is small, the dgLARS method has a greater AUC value, and when the sample size is large the AUC value of all methods are equal to one. In other word, when

[3] This package is being merged with the original package `dglars`.

Table 3: Results of the simulation study based on the Logistic regression model; For each p, n and ρ we report the mean number of the points of the entire solution curve (q) and the area under the ROC curve (AUC). Bold values identify the lowest q for each scenario.

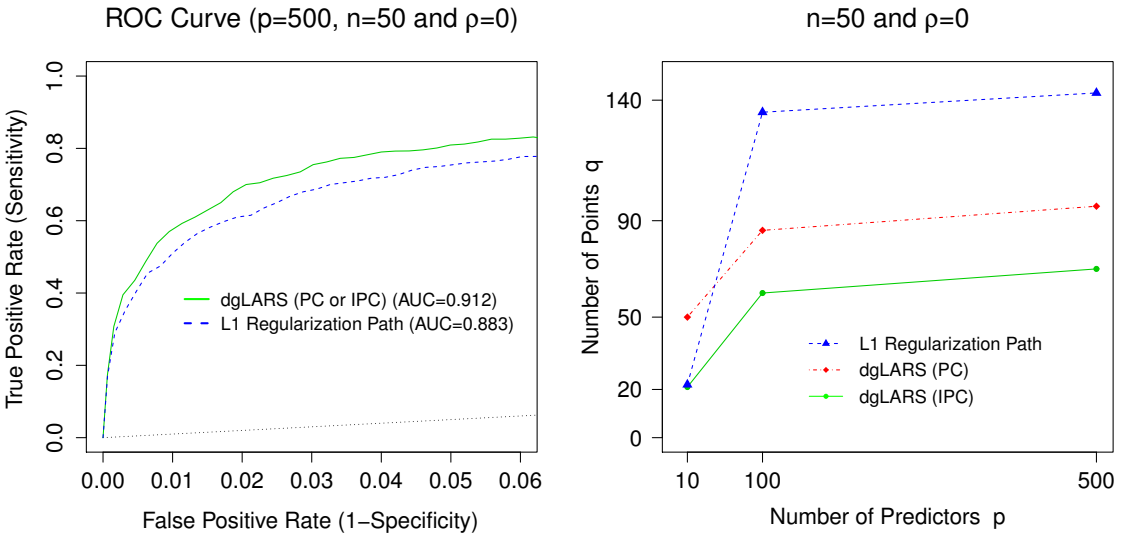
			dgLARS (IPC)*		dgLARS (PC)*		glmpath	
p	n	ρ	q	AUC	q	AUC	q	AUC
10	50	0	**21.06**	0.969	49.04	0.969	22.95	0.968
		0.5	**21.96**	0.970	44.59	0.970	27.78	0.968
		0.75	**22.39**	0.927	41.05	0.927	30.53	0.935
	200	0	**17.99**	1.000	46.65	1.000	18.53	1.000
		0.5	**18.61**	1.000	47.13	1.000	19.48	1.000
		0.75	**19.68**	0.999	45.67	0.999	19.68	0.999
100	50	0	**59.66**	0.955	84.87	0.955	106.3	0.944
		0.5	**51.00**	0.969	69.12	0.969	93.42	0.964
		0.75	**42.15**	0.930	56.24	0.930	83.32	0.930
	200	0	**125.5**	1.000	187.2	1.000	392.0	1.000
		0.5	**107.1**	1.000	155.9	1.000	527.1	1.000
		0.75	**96.33**	1.000	143.1	1.000	846.2	1.000
500	50	0	**70.23**	0.912	93.16	0.912	128.7	0.883
		0.5	**62.78**	0.952	77.78	0.952	119.0	0.941
		0.75	**53.12**	0.916	63.91	0.916	111.5	0.905
	200	0	**171.2**	1.000	212.1	1.000	322.7	1.000
		0.5	**139.7**	1.000	174.2	1.000	273.3	1.000
		0.75	**116.9**	1.000	145.9	1.000	248.7	1.000

* The dgLARS (PC) refers to the predictor-corector implementation of Augugliaro *et al.* (2013), whereas dgLARS (IPC) refers to the improved predictor-corector algorithm proposed in the present paper.

n is efficiently large without considering the number of predictors ($p > n$ or $p < n$) the value of AUC for the methods is 1.

In Figure 2(a) we show the ROC curves ($1 -$ specificity versus sensitivity, computed by averaging over the 100 simulations) corresponding to the dgLARS (by using any of the PC or IPC algorithms) and L_1 Regularization Path methods with $p = 500$, $n = 50$ and $\rho = 0$ based on the Logistic regression model. Also, in Figure 2(b), the mean number of the points of the solution curve (q), computed for these three algorithms, are showed as a function of $p = (10, 100, 500)$ with $n = 50$ and $\rho = 0$. What we mentioned above about q can be clearly seen in this figure.

However, the results related to the number of the covariates included in the final model is not reported for the sake of brevity, we point out that the dgLARS method selects sparser models than the L_1 Regularization Path method. At the end of this section, it should be mentioned that the dgLARS method does not use explicitly a penalized function, so that this method is based on a theory completely different from the L_1 Regularization Path method (L_1-penalized MLE) implemented in the `glmpath` package.



(a) (b)

Figure 2: (a) ROC curve and (b) the mean number of the points of the solution curve q computed by the dgLARS method with the PC and IPC algorithms, and the L_1 Regularization Path method from the simulation study based on the Logistic regression model with $n = 50$ and $\rho = 0$.

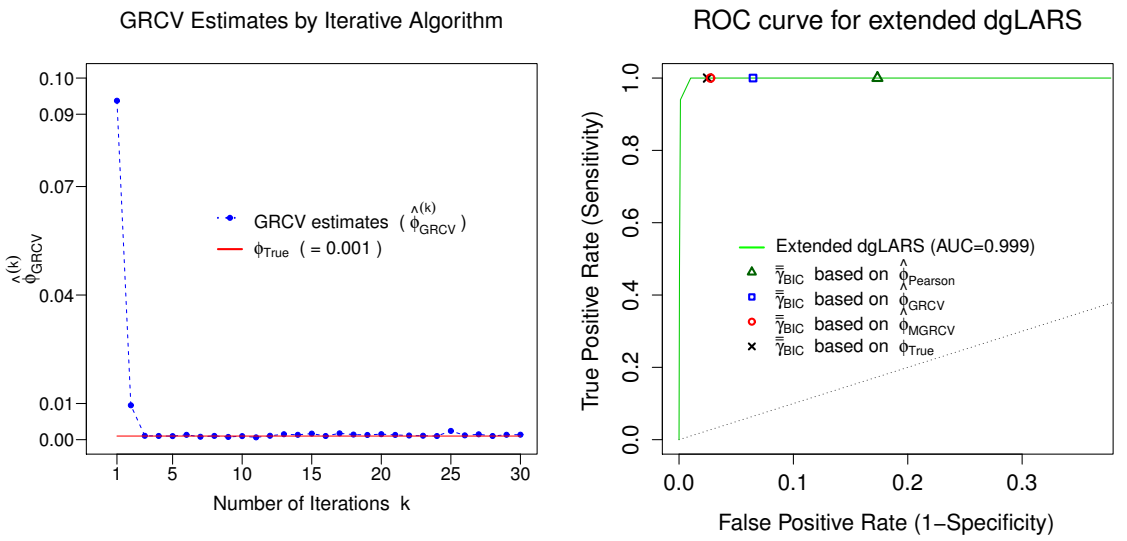
4.2 Comparison of dispersion estimators

This section is divided into two parts; first, in order to show how the GRCV estimator of ϕ and its proposed algorithm work, one simple, but illustrative, example which is a part of a simulation study is presented. Second, we compare the performance of the three dispersion estimators; Pearson ($\hat{\phi}_P$), GRCV ($\hat{\phi}_{GRCV}$) and MGRCV ($\hat{\phi}_{MGRCV}$, the median of the estimators obtained from the iterative GRCV algorithm).

In this simulation study, high-dimensional data are generated according to a Gamma regression model with a non-canonical *log* link, with the shape parameter equal to $\nu = \phi^{-1} = 10^3$ and the scale parameter $\frac{\mu_i}{\nu}$, where $\mu_i = \exp(\mathbf{x}_i^\top \boldsymbol{\beta})$ and $\mathbf{x}_i^\top = (1, x_{i1}, \ldots, x_{ip})$ is as i^{th} row of the design matrix $\mathbf{X}_{n\times(p+1)}$ in which the first column is a column of all ones and the sample size n is 40 and $p = 100$ ($p > n$). We simulate 50 data sets $(\mathbf{y}_1, \mathbf{X}_1), \ldots, (\mathbf{y}_{50}, \mathbf{X}_{50})$, such that $\mathbf{X}_i$ is sampled from an $N(\mathbf{0}, \Sigma)$ distribution, where the diagonal elements of Σ are 1 and the off-diagonal elements are 0, and only the first two predictors ($d = 2$) are used to simulate the response variable $\mathbf{y}_i$,

$$\boldsymbol{\beta} = (\underbrace{0}_{\text{Intercept}}, \underbrace{1, 2}_{2}, \underbrace{0, \ldots, 0}_{98}).$$

We show the result of the simulation study in two pictures (a) and (b) in Figure 3. Figure 3(a) displays the procedure of obtaining the GRCV estimates $\hat{\phi}^{(k)}_{GRCV}$, where $k = (1, 2, \ldots, 30)$, by using the iterative GRCV algorithm, described in Table 2, with only the first data set $(\mathbf{y}_1, \mathbf{X}_1)$. The values of the 30 GRCV estimates, $\{\hat{\phi}^{(1)}_{GRCV}, \ldots, \hat{\phi}^{(30)}_{GRCV}\}$, computed



(a) (b)

Figure 3: (a) GRCV estimates, $\hat{\phi}^{(k)}_{GRCV}$, produced by the iterative GRCV algorithm based on a simulated data set from Gamma model. (b) ROC curve of the extended dgLARS method computed by averaging over the 50 simulations along with some selected tuning parameters.

by the iterative GRCV algorithm, are showed as a function of the number of iterations k. What we mentioned in Section 3.2 can be clearly seen in this figure. It can be seen that, after two iterations, the estimate appears to have improved significantly and converges to the true value of the dispersion parameter $\phi_{True} = 0.001$, so that the median of the GRCV estimates, $\hat{\phi}_{MGRCV}$, is 0.0012. It shows that the proposed iterative algorithm can improve the accuracy of the GRCV estimator.

In Figure 3(b), we plot the ROC curve (computed by averaging over the 50 simulations) corresponding to the extended dgLARS method and present the area under the ROC curve (average AUC over 50 simulations). As seen in the figure, the average AUC is 0.999 which means that the accuracy of the model selected by the extended dgLARS method is quite high. We have reported this result for low- and high-dimensional datasets in the previous section (in Table 3).

Moreover, in the ROC curve in Figure 3(b), we also show the average values of the tuning parameter selected by the BIC criterion $\bar{\hat{\gamma}}_{BIC}$ (computed by averaging $\hat{\gamma}_{BIC}$ over 50 simulations) by means of the dispersion estimators $\hat{\phi}_P$, $\hat{\phi}_{GRCV}$ and $\hat{\phi}_{MGRCV}$, and also the true dispersion parameter ϕ_{True}. As Aho *et al.* (2014) noted, when $d \ll n$, where d (is 2 here) is the number of parameters in the true mode, then the BIC criterion is appropriate. That is why we prefer $\hat{\gamma}_{BIC}$ to $\hat{\gamma}_{AIC}$ and $\hat{\gamma}_{CV}$. We use (20) in which the number of non-zero estimated coefficients $k(\gamma)$ is used as the degree of freedom to calculate the values of the BIC criterion. The same results are obtained if we use the BIC based on the $\widehat{\text{gdf}}(\gamma)$, because the same final model is identified in both cases (this result is not reported for the sake of brevity).

The point on the ROC curve in the most upper left corner has the highest sensitivity

and specificity. A higher sensitivity and specificity indicates superior performance among the tuning parameters obtained by different dispersion estimators. Our results demonstrate that all three final models selected by the chosen tuning parameter $\bar{\hat{\gamma}}_{BIC}$, obtained by the three dispersion estimators $\hat{\phi}_P$, $\hat{\phi}_{GRCV}$ and $\hat{\phi}_{MGRCV}$, have the highest sensitivity (100%), while the specificities of them are 83%, 93% and 97%, respectively. Although these final models selected by means of the three dispersion estimators have a high sensitivity and specificity, the model selected by means of the MGRCV estimator $\hat{\phi}_{MGRCV}$ has the best performance. That means, the dispersion estimator $\hat{\phi}_{MGRCV}$ is a good compromise between specificity and sensitivity. The results also show that our proposed GRCV estimator has a better performance than the Pearson estimator. In addition, since the MGRCV estimate $\hat{\phi}_{MGRCV}$ has a better performance than the GRCV estimate $\hat{\phi}_{GRCV}$, the iterative GRCV algorithm can improve the GRCV estimate to have a more stable and accurate estimate, which proves our claim in Section 3.2.

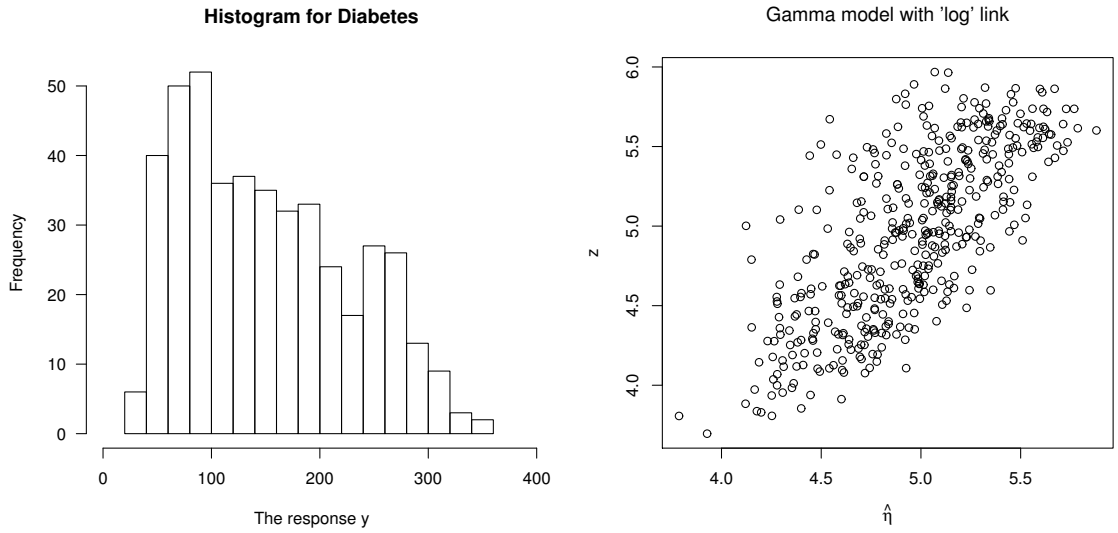
As a result, the results indicate that the extended dgLARS method with $\hat{\phi}_{MGRCV}$ provides a highly specific and sensitive model for high-dimensional GLMs.

5 Application to a diabetes dataset

In this section we consider the benchmark *diabetes* data used in Efron *et al.* (2004) and Ishwaran *et al.* (2010), among others. The response y is a quantitative measure of disease progression for patients with diabetes one year later. The data includes 10 baseline measurements for each patient, such as *age*, *sex* (gender, which is binary), *bmi* (body mass index), *map* (mean arterial blood pressure), and six blood serum measurements: *ldl* (high-density lipoprotein), *hdl* (low-density lipoprotein), *ltg* (lamotrigine), *glu* (glucose), *tc* (triglyceride) and *tch* (total cholesterol), in addition to 45 interactions and 9 quadratic terms, for a total of 64 variables for each patient, so that this data has $n = 442$ observations on $p = 64$ variables. The aim of the study is to identify which of the covariates are important factors in disease progression. Since the original diabetes data is a low-dimensional data ($p = 64$), we add a thousand noise variables to the original data to also have a high-dimensional dataset with $p = 1064$. These low- and high-dimensional diabetes data can be found in our package.

In the recent literature, variable selection techniques, such as LARS and Spike and Slab, were used in a linear regression model applied to this diabetes data. While we spot from Figure 4(a) that, surprisingly, the response y is markedly right-skewed which can arise from a non-normal distribution, for example, a Gamma (or Inverse Gaussian) distribution. Therefore, we fit a Gamma regression model for the (low- and high-dimensional) diabetes data and use the extended dgLARS method by means of the proposed algorithm (IPC). According to the results of the previous section (Section 4.2), the MGRCV estimate $\hat{\phi}_{MGRCV}$ is applied as the dispersion estimator to the data.

Since we do not have prior information on the link function, before starting analyzing we have to choose between three of the most commonly used link functions *inverse*, *log* and *identity*. Therefore, for each of the low- and high-dimensional diabetes data, we fit the Gamma model with these three link functions and then choose the most suitable link function in two ways. First, we plot the adjusted dependent variable $\mathbf{z} = \hat{\boldsymbol{\eta}} + (\mathbf{y} - \hat{\boldsymbol{\mu}})(\partial\boldsymbol{\eta}/\partial\boldsymbol{\mu})$ against the estimated linear predictor $\hat{\boldsymbol{\eta}} = \mathbf{X}\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$, suggested by McCullagh and Nelder (1989), where $\hat{\boldsymbol{\mu}} = g^{-1}(\mathbf{X}\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma))$ is the fitted value, $\hat{\boldsymbol{\beta}}_{\mathcal{A}}(\gamma)$ is the extended dgLARS estimator at γ,



(a) (b)

Figure 4: (a) Histogram of the response y for the diabetes data. (b) Plot of $\mathbf{z}$ versus $\hat{\boldsymbol{\eta}}$ with the *log* link function, computed for the low-dimensional diabetes data, $p = 64$.

and $\partial\boldsymbol{\eta}/\partial\boldsymbol{\mu}$ can be found in Table A1 in Appendix A. The plot should be linear, departure from linear suggests a poor choice of link function (Littell *et al.*, 2002). Second, after fitting these three models (the Gamma model with the three link functions), we choose the best model by comparing the BIC values to see which link function would be more suitable for the data.

The results based on the low- and high-dimensional diabetes data are reported in Sections 5.1 and 5.2, respectively.

5.1 Low-dimensional diabetes data

For the low-dimensional scenario, when $p < n$, we consider the diabetes data with $n = 442$ and $p = 64$ used in Efron *et al.* (2004). For this dataset, we plotted the adjusted dependent variable $\mathbf{z}$ versus the estimated linear predictor $\hat{\boldsymbol{\eta}}$ for the Gamma model with the *inverse*, *log* and *identity* link functions, but for the sake of brevity we only show the plot related to the *log* link (Figure 4(b)). The plots illustrate that while there are scatter in all three plots, there are no overt departure from linearity and hence no obvious evidence of the poor choice of these link functions. In addition, the results (not reported) show that, the model with the *log* link performs the best among these models with BIC of 4806, and the model with the *identity* link (with BIC 4814) fits better than the model with the *inverse* canonical link (with BIC 4829). Finally, we find out that the *log* link function is the most suitable link for the low-dimensional diabetes data and we choose it, in the following, as the selected link function.

We first apply a number of variable selection methods such as LARS (Efron *et al.*, 2004), LASSO (Tibshirani, 1996), Ridge (Hoerl and Kennard, 1970), Elastic Net (Zou and Hastie,

Table 4: The sequences of the top 20 predictors selected by the LARS, LASSO, Ridge, Elastic Net, Spike and Slab and dgLARS algorithms obtained for low-dimensional diabetes data.

Algorithm	Selected Variables																			
LARS	3	9	4	7	37	20	19	12	22	28	2	10	27	11	30	46	33	52	24	29
LASSO	3	9	4	7	37	20	19	12	22	28	2	10	27	11	30	46	33	52	24	29
Ridge	3	9	4	8	7	10	12	5	1	6	13	43	24	37	19	63	64	16	39	17
Elastic Net	3	9	4	7	37	12	20	19	10	22	28	2	27	30	11	52	46	33	24	29
Spike and Slab	3	9	4	7	2	20	37	19	12	27	52	11	10	22	63	30	24	58	43	5
dgLARS (*log*)	3	9	4	7	20	2	28	**60**	11	46	19	29	**18**	30	22	10	37	24	58	**25**
dgLARS (*inverse*)	3	9	4	7	20	**60**	2	46	**18**	10	**42**	28	11	19	30	**35**	29	**40**	24	63

2005a), and Spike and Slab (Ishwaran *et al.*, 2010) by using the `lars` (Hastie and Efron, 2013), `glmnet` (Friedman *et al.*, 2010b) and `spikeslab` (Ishwaran *et al.*, 2010b) packages, and then compare the results to the results obtained from the proposed dgLARS method implemented by our package. Note that, for the dgLARS method we use the *Gamma* family in our package, while this family is not available in other packages, so that we fit the *Gaussian* family to the data to be able to use these packages.

The top 20 selected variables obtained by these algorithms (without considering any model selection criterion) are reported on Table 4, where we used $type = 'lar'$ and $type = 'lasso'$ in the `lars` package for the LARS and LASSO methods, respectively, and for the Ridge and Elastic Net methods we used $\alpha = 0.001$ and $\alpha = 0.5$ in the `glmnet` package, respectively. For the Spike and Slab method we considered $set.seed(112358)$ in the `spikeslab` package, and for the dgLARS method we fitted the Gamma model with the *log* link and also the canonical *inverse* link, so that for this dataset we calculated the dispersion estimates based on each link function as $\hat{\phi}^{\,log}_{MGRCV} = 0.140$ and $\hat{\phi}^{\,inverse}_{MGRCV} = 0.145$.

When we compare the results of the dgLARS Gamma method to the results obtained from other algorithms, we find out the remarkable results. From Table 4 we can see that, the variables selected by the LARS, LASSO and Elastic Net methods are the same, and almost in all models the first 4 variables (3, 9, 4 and 7) are the same. Moreover, importantly, all models (except the dgLARS) have the same selected variables just in the different order. While all algorithms (except the dgLARS) select the covariates 12, 27, 33 and 52 in the first 20 variables, our proposed algorithm does not select them among the top 20 variables. Instead, the dgLARS algorithm by the Gamma model selects several new other variables (indicated in bold in Table 4) which none of the other algorithms do. For instance, the variables 60, 18 and 25 are selected into the first 20 selected variables by the dgLARS Gamma model with the *log* link function, and the variables 60, 18, 42, 35 and 40 are selected when the link function is the *inverse*. As a result, the extended dgLARS method based on a *Gamma* model, with the log link function, finds out that the variables "$hdl : ltg$", "ltg^2" and "$map : ltg$" (60, 18, and 25) are more important factor in disease progression than the variables "bmi^2", "$age : ltg$", "$sex : hdl$" and "$tc : tch$" (12, 27, 33 and 52).

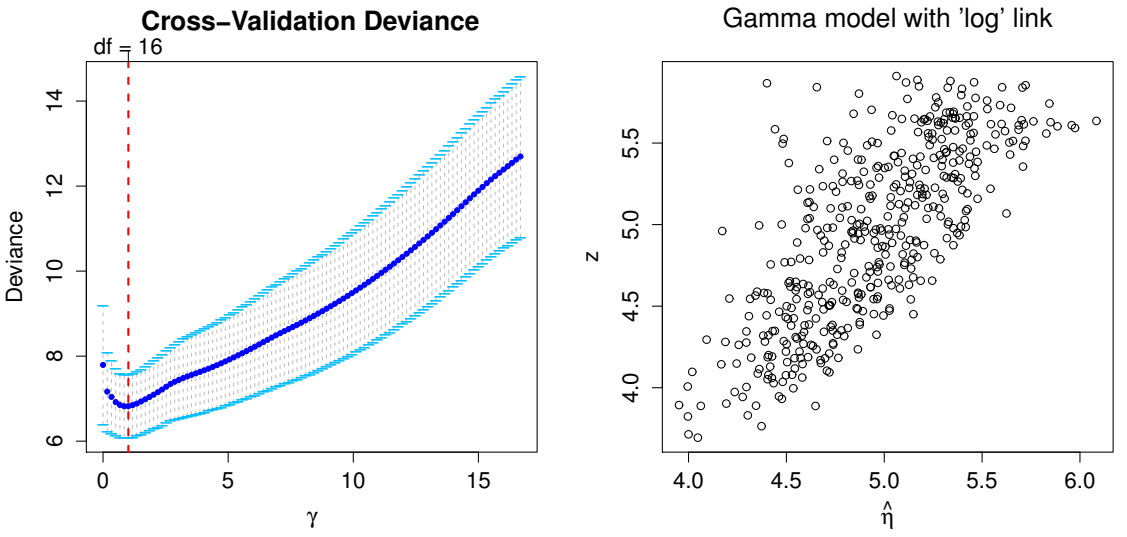
To identify and rank the most important variables, by the dgLARS Gamma regression model with the *log* link function, we use three model selection criteria; cross-validation deviance (CV), AIC and BIC, so that in Table 5, we report the sequence of the top 20 variables and their parameter estimates obtained based on all three model selection criteria. In interpreting the table, we note that the selected variables are those having non-zero

Table 5: A list of the top 20 selected variables and their parameter estimates obtained using the dgLARS Gamma method (with log link, $\eta_i = \log \mu_i$) for low-dimensional diabetes data. In each criterion, variables selected are those having non-zero coefficient estimates; $|\mathcal{A}_{CV}| = 16$, $|\mathcal{A}_{AIC}| = 16$ and $|\mathcal{A}_{BIC}| = 9$.

	Variable		Coefficient Estimate		
Step	Name	Number	CV	AIC	BIC
1	*bmi*	3	3.0757	3.0783	2.9998
2	*ltg*	9	3.4997	3.5071	3.2909
3	*map*	4	1.9033	1.9181	1.5009
4	*hdl*	7	-1.7297	-1.7416	-1.3879
5	*age* : *sex*	20	0.9493	0.9551	0.6846
6	*sex*	2	-1.2282	-1.2489	-0.6400
7	*age* : *glu*	28	0.2091	0.2109	0.1542
8	*hdl* : *ltg*	**60**	0.4284	0.4355	0.1377
9	*age*^2	11	0.2715	0.2815	0
10	*map* : *hdl*	46	0.2929	0.3077	0
11	*glu*^2	19	0.2497	0.2599	0
12	*sex* : *bmi*	29	0.1282	0.1380	0
13	*ltg*^2	**18**	0.0021	-0.1202	0
14	*sex* : *map*	30	0.1087	0.1206	0
15	*age* : *map*	22	0.0091	0.0116	0
16	*glu*	10	0	0	0
17	*bmi* : *map*	37	0	0	0
18	*age* : *ldl*	24	0	0	0
19	*ldl* : *glu*	58	0	0	0
20	*map* : *ltg*	**25**	0	0	0

coefficient estimates. First, we use a tenfold cross-validation to obtain the tuning parameter (γ) of the dgLARS Gamma model. Figure 5(a) shows the 10-fold cross-validation deviance curve as a function of the tuning parameter (γ), where the vertical red dashed line shows the optimal value of γ, which is $\hat{\gamma}_{CV} = 1.011$, with the number of non-zero estimated coefficients, which is $|\mathcal{A}_{CV}| = 16$, where $\mathcal{A}_{CV} = \mathcal{P} \cup \mathcal{A}(\hat{\gamma}_{CV}) = \{m \ : \ \hat{\beta}_m(\bar{\gamma}_{CV}) \neq 0 \ , m = 0, 1, \ldots, p\}$. Since we consider the protected variables set $\mathcal{P}$ contains only the intercept, $|\mathcal{P}| = b = 1$. Second, by means of the BIC criterion the dgLARS method estimates a Gamma regression model with a high level of sparsity, so that only $|\mathcal{A}_{BIC}| = |\mathcal{P} \cup \mathcal{A}(\hat{\gamma}_{BIC})| = 9$ covariates (i.e.,the intercept plus a subset of 8 parameters) are found to influence disease progression, where $\hat{\gamma}_{BIC} = 1.87$. While by using the AIC criterion the number of non-zero estimated coefficients is $|\mathcal{A}_{AIC}| = |\mathcal{P} \cup \mathcal{A}(\hat{\gamma}_{AIC})| = 16$, where $\hat{\gamma}_{AIC} = 0.98$ with AIC 4000.

One points should be mentioned here that, for this low-dimensional data set, the number of the points of the solution curve (q) by using the original PC and improved PC algorithms are 121 and 82, respectively, which shows that the improved algorithm works faster than the original one.



(a) (b)

Figure 5: (a) Plot of the 10-fold cross-validation deviance computed for the low-dimensional diabetes data with $p = 64$. (b) Plot of $\mathbf{z}$ against $\hat{\boldsymbol{\eta}}$ computed with the high-dimensional diabetes data, $p = 1064$, when the link functions is *log*.

5.2 High-dimensional diabetes data

For a p larger than n setup, we expanded the original diabetes data to become $n = 442$ and $p = 1064$, so that the 1000 additional variables are in reality just noise. We fit a Gamma regression model for this high-dimensional data and use the extended dgLARS method by means of the proposed algorithm (IPC). For the high-dimensional diabetes data, based on the plots of the adjusted dependent variable $\mathbf{z}$ versus the estimated linear predictor $\hat{\boldsymbol{\eta}}$ (not shown here except for the *log* link, Figure 5(b)), we obtained the same results for all three considered link functions, but based on the BIC values (not reported here) we chose the Gamma model with the *log* link function as the best model. Moreover, for this dataset we calculated the dispersion estimate based on this model by using the MGRCV estimator $\hat{\phi}_{MGRCV} = 0.147$.

Figure 6 consists of four images which are outputs of our package. The figure displays the dgLARS Gamma solution path, the Rao score path and the CV, AIC and BIC criteria obtained using the improved PC algorithm and the full data. Like Section 5.1, we consider three criteria; Firstly, Figure 6(a) shows the 10-fold cross-validation deviance curve in which the optimal value of the tuning parameter is $\hat{\gamma}_{CV} = 1.77$, with the number of non-zero estimated coefficients, which is $|\mathcal{A}_{CV}| = |\mathcal{P} \cup \mathcal{A}(\hat{\gamma}_{CV})| = 57$, where $\mathcal{P}$ contains only the intercept. Secondly, by the BIC model selection criterion the dgLARS method estimates a Gamma regression model with a high level of sparsity, so that $\hat{\gamma}_{BIC} = 2.76$ with BIC of 4817 and $|\mathcal{A}_{BIC}| = 11$ covariates (i.e., the intercept plus a subset ($\mathcal{A}(\hat{\gamma}_{BIC})$) of 10 parameters) are found to influence disease progression. While by the AIC model selection criterion, $\hat{\gamma}_{AIC} = 1.79$ (with AIC of 4760) and the number of non-zero estimated coefficients is $|\mathcal{A}_{AIC}| = 53$ (i.e., the subset $\mathcal{A}(\hat{\gamma}_{AIC})$ has 52 covariates).

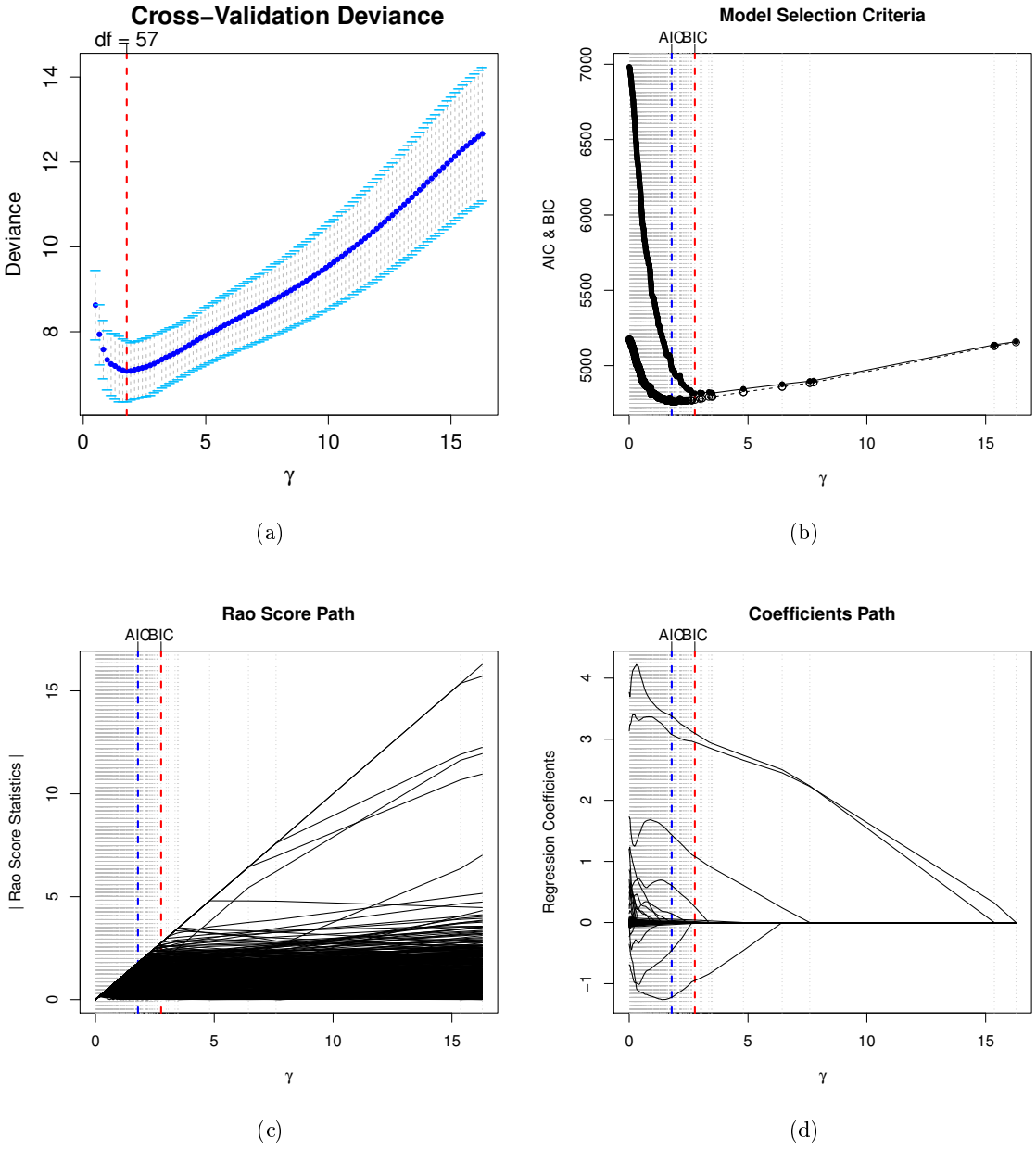


Figure 6: (a) Plot of the 10-fold cross-validation deviance, (b) Model selection criteria, (c) Rao score statistics path, (d) Regression coefficients path for the dgLARS Gamma regression model for the high-dimensional diabetes data with $p = 1000$ noise variables.

In addition, we report the sequence of the 25 selected variables and their parameter estimates based on all three criteria in Table 6. In interpreting the table, we note that variables starting with "n." are noise variables and the rest are the original variables.

Using Figure 6 and Table 6 we can see that, while only 4 variables ($3, 9, 4$ and 7) have path-profiles that clearly stand out in all three criteria, significantly these variables are the

Table 6: The top 25 variables and their parameter estimates obtained using the dgLARS Gamma method (with log link) for high-dimensional diabetes data. In each criterion, variables selected are those having non-zero coefficient estimates; $|\mathcal{A}_{CV}| = 57$, $|\mathcal{A}_{AIC}| = 53$ and $|\mathcal{A}_{BIC}| = 11$.

	Variable		Coefficient Estimate		
Step	Name	Number	CV	AIC	BIC
1	*bmi*	3	3.0794	3.0762	2.9489
2	*ltg*	9	3.3787	3.3746	3.0971
3	*map*	4	1.4391	1.4337	1.0788
4	*hdl*	7	-1.2253	-1.2228	-0.9491
5	*n*.312	376	0.0551	0.0548	0.0387
6	*n*.545	609	0.0320	0.0318	0.0155
7	*n*.543	607	-0.0341	-0.0338	-0.0142
8	*age* : *sex*	20	0.6080	0.6033	0.2545
9	*n*.423	487	0.0113	0.0112	0.0034
10	*n*.770	834	0.0177	0.0175	0.0036
11	*n*.657	721	0.0115	0.0113	0
12	*sex*	2	-0.4608	-0.4550	0
13	*n*.636	700	-0.0170	-0.0167	0
14	*n*.283	347	0.0124	0.0123	0
15	*n*.337	401	-0.0162	-0.0160	0
16	*n*.404	468	0.0090	0.0088	0
17	*n*.62	126	-0.0121	-0.0118	0
18	*n*.988	1052	0.0089	0.0086	0
19	*age* : *glu*	28	0.1465	0.1440	0
20	*n*.71	135	-0.0090	-0.0088	0
21	*n*.160	224	-0.0083	0.0083	0
22	*n*.635	699	-0.0080	-0.0087	0
23	*n*.466	530	-0.0085	-0.0083	0
24	*n*.612	676	-0.0084	-0.0082	0
25	*n*.969	1033	-0.0045	-0.0045	0

top 4 from our previous analysis obtained using the low-dimensional data (Section 5.1). It is interesting that 3 other non-noise variables, "*age* : *sex*", "*sex*" and "*age* : *glu*" (with variable numbers: $20, 2$ and 28) are in the top 25 variables, so that in Table 5, they have the variable number: $5, 6$ and 7, respectively, and along with "*bmi*","*ltg*","*map*" and "*hdl*" are the first 7 variables in Table 5. Regardless of the criteria used, when we inspected the first 100 variables selected by the improved PC algorithm, we found that 8 were from the original 64 variables, and 7 were from the top 25 variable from Table 6. This demonstrates stability of the improved PC algorithm even in ultra-high dimensional problems.

Moreover, for this data set the number of the points of the solution curve by using the original PC and improved PC algorithms are 482 and 465, respectively.

6 Conclusions

In this paper we extended the dgLARS method for a GLM to a larger class of the exponential family, namely the *exponential dispersion family* (when the dispersion parameter, ϕ, is unknown), and obtained the general framework of the dgLARS estimator for general GLM with general link function. We implemented explicitly the method for Gamma and Inverse Gaussian with a variety of link functions. To estimate the dispersion parameter we first presented an classical estimator which can be used during the solution path, and then proposed a new method to do high-dimensional inference on the dispersion parameter. We also proposed an iterative algorithm that produces a more stable and accurate estimation. Moreover, we proposed an improved version of the predictor-corrector (PC) algorithm to compute the solution curve. The improved PC algorithm allows the dgLARS method to be implemented using less steps, greatly reducing the computational burden because of reducing the number of points of the solution curve. The method was compared well with some well-known methods where can be used. The results show that the improved PC algorithm is better and quicker than the original PC algorithm, and now the dgLARS method can be used for a variety of distributions with different types of the canonical and non-canonical link functions.

Acknowledgement

We would like to thank the editor and anonymous reviewers for valuable comments which improved the presentation of the paper.

Appendices

A Required equations of extended dgLARS Gamma and Inverse Gaussian

Table A1 provides the list of equations required to obtain the general framework of the extended dgLARS estimator for the Gamma and Inverse Gaussian GLM with general link functions.

Table A1: Required Equations for obtaining extended dgLARS estimator based on Gamma (G) and Inverse Gaussian (IG) regressions, where $i = 1, \ldots, n$ and $m, n = 1, \ldots, p$.

Equations	$f_{Y_i}(y_i)$	$g(\mu_i) = \eta_i = \mathbf{x}_i^\top \boldsymbol{\beta}$			
		$-\frac{1}{2\mu_i^2}$ *	$-\frac{1}{\mu_i}$ **	$\log(\mu_i)$	μ_i
$\partial_m \ell(\boldsymbol{\beta}, \phi; \mathbf{y})$	G	-	$\nu \sum_{i=1}^{n} (y_i - \mu_i)\ x_{im}$	$\nu \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{\mu_i}\ x_{im}$	$\nu \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{\mu_i^2}\ x_{im}$
	IG	$\lambda \sum_{i=1}^{n} (y_i - \mu_i)\ x_{im}$	$\lambda \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{\mu_i}\ x_{im}$	$\lambda \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{\mu_i^2}\ x_{im}$	$\lambda \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{\mu_i^3}\ x_{im}$
$\partial_{mn} \ell(\boldsymbol{\beta}, \phi; \mathbf{y})$	G	-	$-\nu \sum_{i=1}^{n} x_{im}\ x_{in}\ \mu_i^2$	$-\nu \sum_{i=1}^{n} \frac{y_i}{\mu_i}\ x_{im}\ x_{in}$	$-\nu \sum_{i=1}^{n} \left(\frac{2y_i}{\mu_i^3} - \frac{1}{\mu_i^2}\right)\ x_{im}\ x_{in}$
	IG	$-\lambda \sum_{i=1}^{n} x_{im}\ x_{in}\ \mu_i^3$	$-\lambda \sum_{i=1}^{n} x_{im}\ x_{in}\ y_i$	$-\lambda \sum_{i=1}^{n} \left(\frac{2y_i}{\mu_i^2} - \frac{1}{\mu_i}\right) x_{im}\ x_{in}$	$-\lambda \sum_{i=1}^{n} \left(\frac{3y_i}{\mu_i^4} - \frac{2}{\mu_i^3}\right) x_{im}\ x_{in}$
$\mathcal{I}_{mn}(\boldsymbol{\beta}, \phi)$	G	-	$\nu \sum_{i=1}^{n} x_{im}\ x_{in}\ \mu_i^2$	$\nu \sum_{i=1}^{n} x_{im}\ x_{in}$	$\nu \sum_{i=1}^{n} \frac{x_{im}\ x_{in}}{\mu_i^2}$
	IG	$\lambda \sum_{i=1}^{n} x_{im}\ x_{in}\ \mu_i^3$	$\lambda \sum_{i=1}^{n} x_{im}\ x_{in}\ \mu_i$	$\lambda \sum_{i=1}^{n} \frac{x_{im}\ x_{in}}{\mu_i}$	$\lambda \sum_{i=1}^{n} \frac{x_{im}\ x_{in}}{\mu_i^3}$
$\partial_m \mathcal{I}_n(\boldsymbol{\beta}, \phi)$	G	-	$2\ \nu \sum_{i=1}^{n} x_{im}\ x_{in}^2\ \mu_i^3$	0	$-2\ \nu \sum_{i=1}^{n} \frac{x_{im}\ x_{in}^2}{\mu_i^3}$
	IG	$3\ \lambda \sum_{i=1}^{n} x_{im}\ x_{in}^2\ \mu_i^5$	$\lambda \sum_{i=1}^{n} x_{im}\ x_{in}^2\ \mu_i^2$	$-\lambda \sum_{i=1}^{n} \frac{x_{im}\ x_{in}^2}{\mu_i}$	$-3\ \lambda \sum_{i=1}^{n} \frac{x_{im}\ x_{in}^2}{\mu_i^4}$
$r_m(\boldsymbol{\beta}, \phi)$	G	-	$\sqrt{\nu}\ \frac{\sum_{i=1}^{n} (y_i - \mu_i)\ x_{im}}{\sqrt{\sum_{i=1}^{n} x_{im}^2\ \mu_i^2}}$	$\sqrt{\nu}\ \frac{\sum_{i=1}^{n} x_{im}(y_i - \mu_i)/\mu_i}{\sqrt{\sum_{i=1}^{n} x_{im}^2}}$	$\sqrt{\nu}\ \frac{\sum_{i=1}^{n} x_{im}(y_i - \mu_i)/\mu_i^2}{\sqrt{\sum_{i=1}^{n} x_{im}^2/\mu_i^2}}$
	IG	$\sqrt{\lambda}\ \frac{\sum_{i=1}^{n} (y_i - \mu_i)\ x_{im}}{\sqrt{\sum_{i=1}^{n} x_{im}^2\ \mu_i^3}}$	$\sqrt{\lambda}\ \frac{\sum_{i=1}^{n} x_{im}(y_i - \mu_i)/\mu_i}{\sqrt{\sum_{i=1}^{n} x_{im}^2\ \mu_i}}$	$\sqrt{\lambda}\ \frac{\sum_{i=1}^{n} x_{im}(y_i - \mu_i)/\mu_i^2}{\sqrt{\sum_{i=1}^{n} x_{im}^2/\mu_i}}$	$\sqrt{\lambda}\ \frac{\sum_{i=1}^{n} x_{im}(y_i - \mu_i)/\mu_i^3}{\sqrt{\sum_{i=1}^{n} x_{im}^2/\mu_i^3}}$
$\frac{\partial \mu_i}{\partial \eta_i}$		$(-2\eta_i)^{-1.5} = \mu_i^3$	$\eta_i^{-2} = \mu_i^2$	$\exp(\eta_i) = \mu_i$	1

* Canonical for IG
** Canonical for G

References

Aho K., Derryberry D., and Peterson T. Model selection for ecologists: the worldviews of aic and bic. *Ecology*, 95(3):631–636, 2014.

Akaike H. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.

Allgower E. and Georg K. *Introduction to Numerical Continuation Methods*. Society for Industrial and Applied Mathematics, New York, 2003.

Arlot S. and Celisse A. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010.

Augugliaro L. *dglars: Differential Geometric LARS (dgLARS) Method*. R package version 1.0.5, 2014b. http://CRAN.R-project.org/package=dglars.

Augugliaro L., Mineo A.M., and Wit E.C. Differential geometric least angle regression: A differential geometric approach to sparse generalized linear models. *Journal of the Royal Statistical Society: Series B*, 75(3):471–498, 2013.

Augugliaro L., Mineo A.M., and Wit E.C. dglars: An r package to estimate sparse generalized linear models. *Journal of Statistical Software*, 59(8):1–40, 2014a.

Augugliaro L., Mineo A.M., and Wit E.C. A differential geometric approach to generalized linear models with grouped predictors. *Biometrika*, 103:563–593, 2016.

Burnham K.P. and Anderson D.R. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer, New York, 2002, 2^{nd} edition.

Candes E.J. and Tao T. The dantzig selector: Statistical estimation when p is much larger than n. *Annals of Statistics*, 35:2313–2351, 2007.

Chen Y., Du P., and Wang Y. Variable selection in linear models. *Wiley Interdisciplinary Reviews: Computational Statistics*, 6:1–9, 2014.

Cordeiro G.M. and McCullagh P. Bias correction in generalized linear models. *Journal of the Royal Statistical Society: Series B*, 53(3):629–643, 1991.

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

Fan J., Guo S., and Hao N. Variance estimation using refitted cross-validation in ultrahigh dimensional regression. *Journal of the Royal Statistical Society: Series B*, 74(1):37–65, 2012.

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

Fan J. and Lv J. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B*, 70:849–911, 2008.

Farrington C.P. On assessing goodness of fit of generalized linear model to sparse data. *Journal of the Royal Statistical Society: Series B*, 58(2):349–360, 1996.

Friedman J., Hastie T., and R.Tibshirani. *glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models*. R package version 1.1-5, 2010b. http://CRAN.R-project.org/package=glmnet.

Hastie T. and Efron B. *lars: Least Angle Regression, Lasso and Forward Stagewise*. R package version 1.2, 2013. http://CRAN.R-project.org/package=lars.

Hastie T., Tibshirani R., and Friedman J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2009.

Hoerl A.E. and Kennard R. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

Ishwaran H., Kogalur U.B., and Rao J. spikeslab: Prediction and variable selection using spike and slab regression. *The R Journal*, 2(2):68–73, 2010.

Ishwaran H., Kogalur U.B., and Rao J. *spikeslab: Prediction and variable selection using spike and slab regression*. R package version 1.1.2, 2010b. http://CRAN.R-project.org/package=spikeslab.

James G. and Radchenko P. A generalized dantzig selector with shrinkage tuning. *Biometrika*, 96:323–337, 2009.

Jorgensen B. Exponential dispersion models. *Journal of the Royal Statistical Society, Series B*, 49:127–162, 1987.

Jorgensen B. *The Theory of Dispersion Models*. Chapman & Hall, London, 1997.

Kullback S. and Leibler R.A. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

Li K.C. Asymptotic optimality for c_p , c_l , cross-validation and generalized cross-validation: Discrete index set. *Annals of Statistics*, 15:958–975, 1987.

Littell R.C., Stroup W.W., and Feund R.J. *SAS for Linear Models*. Sas Institute Inc., Cary, North Carolina, 2002, 4^{th} edition.

McCullagh P. and Nelder J.A. *Generalized Liner Models*. Chapman & Hall, London, 1989.

McQuarrie A.D.R. and Tsai C.L. *Regression and Time Series Model Selection*. World Scientific Publishing Co. Pte. Ltd., Singapore, 1998, 1^{st} edition.

Meng R. *Estimation of Dispersion Parameters in GLMs with and without Random Effects*. Master's thesis, Stockholm University, 2004.

Panjer H.H. *Operational Risk: Modeling Analytics*. John Wiley & Sons, New York, 2006.

Park M.Y. and Hastie T. *glmpath: L_1 Regularization Path for Generalized Linear Models and Cox Proportional Hazards Model*. R package version 0.94, 2007b. http://CRAN.R-project.org/package=glmpath.

Press W.H., Flannery B.P., Teukolsky S.A., and Vetterling W.T. *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press, England, 1992, 2^{nd} edition.

Schwarz G. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

Shao J. An asymptotic theory for linear model selection. *Statistica Sinica*, 7:221–264, 1997.

Shibata R. An optimal selection of regression variables. *Biometrika*, 68:45–54, 1981.

Shibata R. Approximation efficiency of a selection procedure for the number of regression variables. *Biometrika*, 71:43–49, 1984.

Stone M. Asymptotics for and against cross-validation. *Biometrika*, 64:29–35, 1977.

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

Ultricht J. and Tutz G. Combining quadratic penalization and variable selection via forward boosting. Technical Report, Department of Statistics, Munich University, 2011. Technical Reports No. 99.

Whittaker E.T. and Robinson G. *The Calculus of Observations: An Introduction to Numerical Analysis*. Dover Publications, New York, 1967, 4^{th} edition.

Wood S.N. *Generalized Additive Models: An Introduction with R*. Chapman & Hall/CRC, Boca Raton, 2006.

Zhang C.H. Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, 38(2):894–942, 2010.

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