A MAP APPROACH FOR ℓ_q-NORM REGULARIZED SPARSE PARAMETER ESTIMATION USING THE EM ALGORITHM

Rodrigo Carvajal [1], *Juan C. Agüero*[1], *Boris I. Godoy* [2] *and Dimitrios Katselis* [3]

[1] Electronics Engineering Department, Universidad Técnica Federico Santa María, Chile.
email addresses: {rodrigo.carvajalg, juan.aguero}@usm.cl

[2] School of Electrical Engineering and Telecommunications, The University of New South Wales, Australia.
email address: b.godoy@unsw.edu.au

[3] Coordinated Science Laboratory,
University of Illinois at Urbana-Champaign, IL, USA.   email address: katselis@illinois.edu

ABSTRACT

In this paper, Bayesian parameter estimation through the consideration of the Maximum A Posteriori (MAP) criterion is revisited under the prism of the Expectation-Maximization (EM) algorithm. By incorporating a sparsity-promoting penalty term in the cost function of the estimation problem through the use of an appropriate prior distribution, we show how the EM algorithm can be used to efficiently solve the corresponding optimization problem. To this end, we rely on variance-mean Gaussian mixtures (VMGM) to describe the prior distribution, while we incorporate many nice features of these mixtures to our estimation problem. The corresponding MAP estimation problem is completely expressed in terms of the EM algorithm, which allows for handling nonlinearities and *hidden variables* that cannot be easily handled with traditional methods. For comparison purposes, we also develop a Coordinate Descent algorithm for the ℓ_q-norm penalized problem and present the performance results via simulations.

1. INTRODUCTION

In engineering, an overwhelming number of applications deal with relating a random response variable $\boldsymbol{y}$ with a set of explanatory variables or covariates $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)$ through a regression model. The most usual case is that of the linear regression, namely

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{y}$ is the $n \times 1$ response vector, $\boldsymbol{X}$ is the $n \times p$ regression matrix, $\boldsymbol{\theta}$ is the $p \times 1$ parameter vector and $\boldsymbol{\epsilon}$ is the $n \times 1$ noise vector. The traditional way to perform parameter estimation in this setup is by considering the Maximum Likelihood (ML) approach. The corresponding estimates can be iteratively computed based on the Expectation-Maximization (EM) algorithm [1], when the observations are seen as *incomplete* data, see, e.g., [2].

On the other hand, when the parameter vector $\boldsymbol{\theta}$ is sparse, i.e., many of its entries are equal to zero, it is common to include a sparsity promoting penalty to the ML problem, yield-

ing a regularized ML estimation problem. Those sparsity promoting penalties are usually the ℓ_1-norm, the ℓ_q-norm ($0 < q < 1$) and the ℓ_0 pseudo-norm [3, 4]. Depending on the penalty, different techniques are utilized to solve this regularized ML problem, such as *proximal methods* (PM), see, e.g., [5], or the *local linear approximation* (LLA) and the *local quadratic approximation* (LQA) methods [6, 7]. Another optimization approach for penalized problems corresponds to the implementation of the Coordinate Descent (CD) algorithm [8]. This technique has been successfully applied to the Lasso problem [9]. Its extension to non-convex penalties was introduced in [10]. Its particular application to sparsity problems with the ℓ_q-norm penalty and its convergence is analysed in [11]. In CD, the vector parameter estimates are obtained in a *per coordinate* base. Within the EM framework, the application of the CD approach yields the so-called Expectation-Conditional Maximum (ECM) algorithm [2]. In this paper, we show how to extend the ECM algorithm to ℓ_q-norm penalized ML problems with *hidden variables* using the CD method.

A different approach corresponds to turn the penalty function in the form of a probability density function (pdf), accounting for some prior knowledge of the vector parameter $\boldsymbol{\theta}$. Thus, we can turn the ML estimation problem into a Maximum a Posteriori (MAP) problem. If the observations correspond to *incomplete data*, the EM algorithm can be easily modified to produce the posterior mode of $\boldsymbol{\theta}$ instead of the ML estimate of $\boldsymbol{\theta}$. Denoting by $p(\boldsymbol{y}|\boldsymbol{\theta})$ the response sampling density, the corresponding log-likelihood function is $\mathcal{L}(\boldsymbol{\theta}) = \log p(\boldsymbol{y}|\boldsymbol{\theta})$. For ML estimation, the EM algorithm iteratively selects a value $\boldsymbol{\theta}^*$ that maximizes $\mathcal{L}(\boldsymbol{\theta})$. In the case of MAP estimation, the function that is maximized by the EM algorithm is $\mathcal{L}(\boldsymbol{\theta}) + G(\boldsymbol{\theta})$, where $G(\boldsymbol{\theta}) = \log p(\boldsymbol{\theta})$ is the log-prior of the parameter vector $\boldsymbol{\theta}$. From these observations, it becomes clear that the EM algorithm is a quite general tool for solving either classical or Bayesian estimation problems.

In this paper, we formalize an EM-based framework for regularized ML parameter estimation using a MAP perspective. Motivated by the recent advances in regularized parameter estimation and in compressed sensing, we propose a way to introduce the ℓ_q-norm sparsity-promoting penalty into our estimation problem by properly selecting the prior through the use of variance-mean Gaussian mixtures (VMGM) [12]. Different approaches have been considered in the literature using VMGM. In [13], the ℓ_1-norm penalty is considered for promoting sparsity, which is interpreted as a Laplace prior for a MAP interpretation of the ℓ_1 penalty using the EM algo-

rithm. For robust regression, VMGMs can be utilized to express a least ℓ_q regression as a ML estimation problem, which is then solved using the EM algorithm [14]. The convergence properties of the approach in [14] solved by using iterative re-weighted least squares were presented in [15]. A general approach for MAP estimation is considered in [16], where the main idea for utilizing the VMGMs and EM algorithm for the *prior* pdf was extended to different VMGMs. This idea was further extended in [17] for sparse parameter vectors with quantized data, which in turn is a generalization of the work in [18]. In this paper, we provide all the relevant theory for the formulation of the EM algorithm, when VMGMs are employed, for *hidden variables* in both the *likelihood function* and the *prior* pdf tailored to the ℓ_q-norm, $0 < q \leq 1$. To illustrate the effectiveness of our proposal, we present an example with both *hidden variables* and nonlinearities and compare the results with utilizing the ℓ_1- and the ℓ_q-norms with both the CD and the EM-based MAP approaches.

The remainder of the paper is organized as follows: Section 2 introduces the basic EM definitions for the ML, penalized ML and MAP estimation problems. The derivation of the ECM algorithm with the ℓ_q-norm penalty is included. Section 3 introduces the basic notion of mixtures for expressing a prior distribution and provides all the necessary expressions for the formulation of the EM algorithm, when the prior is expressed as a VMGM. Section 4 presents the VMGM expressions for the ℓ_q−norm penalty. Finally, simulation results are provided in Section 5, while Section 6 concludes the paper.

2. BASIC DESCRIPTION OF THE EM ALGORITHM

The EM algorithm is basically formulated to iteratively solve the ML estimation problem:

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}). \tag{2}$$

Using the notion of a *hidden* variable $\boldsymbol{z}$ [2], we can easily express the log-likelihood function as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) - \mathcal{H}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right), \tag{3}$$

where $\hat{\boldsymbol{\theta}}^{(i)}$ is the estimate of $\boldsymbol{\theta}$ at the ith iteration, and

$$\mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \log p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{\theta}) p\left(\boldsymbol{z}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{z},$$

$$\mathcal{H}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \log p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{\theta}) p\left(\boldsymbol{z}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{z}.$$

Using Jensen's inequality [19], one can readily see that [1]

$$\mathcal{H}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) \leq \mathcal{H}_{\mathrm{ML}}\left(\hat{\boldsymbol{\theta}}^{(i)}, \hat{\boldsymbol{\theta}}^{(i)}\right)$$

for any $\boldsymbol{\theta}$. Therefore, the EM algorithm for the ML estimation problem can be formulated as follows [2]:
E-step: Compute
$$\mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = E\left[\log p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{\theta})|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}\right].$$
M-step: Solve
$$\hat{\boldsymbol{\theta}}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right).$$

Under some general conditions, the above algorithm converges to the ML estimate of $\boldsymbol{\theta}$ [2].

2.1. Penalized ML estimation via the EM algorithm

Many regularized estimation problems involve the optimization of a cost function of the form

$$U(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) - g(\boldsymbol{\theta}), \tag{4}$$

where $g(\cdot)$ is a penalty function. If we consider that $\mathcal{L}(\boldsymbol{\theta})$ is the *log-likelihood* function, then this is a penalized ML estimation problem. For the attainment of the estimates, different approaches can be used, depending on the nature of the penalty function. In our case, in order to promote sparsity on $\boldsymbol{\theta}$, a popular choice for the penalty function is $g(\boldsymbol{\theta}) = \frac{1}{\tau}\|\boldsymbol{\theta}\|_q^q$ with $0 < q \leq 1$. When $q = 1$, we obtain the Lasso [20]. In order to solve the penalized ML estimation problem, the modifications to the EM algorithm are implemented on the M-step, while the E-step remains the same as before [2, Chap. 5.18]. Hence, the modified EM algorithm for penalized ML estimation is given as follows:
E-step: Compute

$$\mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = E\left[\log p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{\theta})|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}\right].$$

M-step: Solve

$$\hat{\boldsymbol{\theta}}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) - g(\boldsymbol{\theta}).$$

In order to understand why traditional methods cannot be applied straightforwardly, let us consider a linear regression as in (1), but where the matrix $\boldsymbol{X}$ is a function of a *hidden variable*, e.g., an unknown state in a state-space representation. In such a case, when performing the E-step, we need to compute $\boldsymbol{E}_1 = E[\boldsymbol{X}|\boldsymbol{y}]$ and $\boldsymbol{E}_2 = E[\boldsymbol{X}^T\boldsymbol{X}|\boldsymbol{y}]$. In general, methods that do not consider *hidden variables* rely on the relationship between the matrix $\boldsymbol{X}$ and $\boldsymbol{X}^T\boldsymbol{X}$. However, in our case, $\boldsymbol{E}_2 \neq \boldsymbol{E}_1^T\boldsymbol{E}_1$, and hence traditional methods should be modified accordingly.

2.1.1. The ECM algorithm for ℓ_q-norm penalized ML estimation

For non-convex penalties, a CD algorithm [8] (also called Cyclic Descent) was first proposed and analysed in [10] and specifically applied to the ℓ_q-norm penalty as a sparsity inducing norm in [11]. The basic idea behind [10] and [11] is to optimize the penalized ML problem in a coordinate-wise manner. This implies computing the gradient of the *log-likelihood* function with respect to one component of $\boldsymbol{\theta}$ and solving for this component, while the remaining components are considered constants. This is implemented for each component of the parameter vector $\boldsymbol{\theta}$ in a cyclic manner until convergence. In our analysis, the system model allows for obtaining an auxiliary function $\mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$ that is quadratic with respect to $\boldsymbol{\theta}$. To see this, let us first assume that the regressor matrix $\boldsymbol{X}$ is a function of a *hidden variable* $\boldsymbol{z}$, $\boldsymbol{X} = f(\boldsymbol{z})$, and that the measurement noise is zero mean Gaussian distributed with covariance matrix $\boldsymbol{\Sigma}_\epsilon$. With these assumptions, we have:

$$\mathcal{Q}_{\mathrm{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = -\frac{1}{2}\left(\boldsymbol{y}^T\boldsymbol{\Sigma}_\epsilon^{-1}\boldsymbol{y} - 2\boldsymbol{y}^T\boldsymbol{\Sigma}_\epsilon^{-1}E[\boldsymbol{X}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]\boldsymbol{\theta}\right.$$
$$\left. + \boldsymbol{\theta}^T E[\boldsymbol{X}^T\boldsymbol{\Sigma}_\epsilon^{-1}\boldsymbol{X}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]\boldsymbol{\theta}\right),$$

which is clearly a quadratic function of $\boldsymbol{\theta}$. Then, we compute the gradient with respect to only one component of $\boldsymbol{\theta}$. For simplicity, let us define $\boldsymbol{E}_1 = E[\boldsymbol{X}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]$ and $\boldsymbol{E}_2 = E[\boldsymbol{X}^T\boldsymbol{\Sigma}_\epsilon^{-1}\boldsymbol{X}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]$. With these we obtain

$$\frac{\partial}{\partial \theta_j} \mathcal{Q}_{\text{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \left([\boldsymbol{E}_1]_j^T \boldsymbol{\Sigma}_\epsilon \boldsymbol{y} - \mathbf{q}_j^T \boldsymbol{E}_2 \boldsymbol{\theta}\right), \quad (5)$$

where $[\boldsymbol{E}_1]_j$ is the jth column of $\boldsymbol{E}_1$ and $\mathbf{q}_j$ is the jth column of the identity matrix. If we solve for θ_j, we obtain

$$\theta_j^{(i+1)*} = \frac{[\boldsymbol{E}_1]_j^T \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{y} - [\boldsymbol{E}_2]_j^T [\theta_1^{(i+1)}, ..., \theta_{j-1}^{(i+1)}, 0, \theta_{j+1}^{(i)}, ..., \theta_p^{(i)}]^T}{[\boldsymbol{E}_2]_{j,j}}.$$

Since we are interested in the penalized ML problem, we include the penalty for the jth component of $\boldsymbol{\theta}$ and solve for

$$\theta_j^{(i+1)} = \arg\max_{\theta_j^{(i+1)}} -\frac{1}{2}\left(\theta_j^{(i+1)} - \theta_j^{(i+1)*}\right)^2 - \frac{\left|\theta_j^{(i+1)}\right|^q}{\tau[\boldsymbol{E}_2]_{j,j}},$$

using the scalar optimization algorithm described in [11].

2.2. MAP estimation via the EM algorithm

Let us consider now the case of the MAP estimation. Based on Bayes' theorem, the *log-posterior* pdf of $\boldsymbol{\theta}$ can be expressed as:

$$\log p(\boldsymbol{\theta}|\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\theta}) + G(\boldsymbol{\theta}) - \log p(\boldsymbol{y}). \quad (6)$$

Using (3), (6) and the fact that $\mathcal{H}_{\text{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$ is a decreasing function of $\boldsymbol{\theta}$, the EM algorithm for the MAP estimation problem is formulated as follows:
E-step: Compute

$$\mathcal{Q}_{\text{MAP}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = E\left[p(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{\theta})|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}\right] + G(\boldsymbol{\theta}).$$

M-step: Solve

$$\hat{\boldsymbol{\theta}}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}_{\text{MAP}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right).$$

In the next section, we will see how we can express the *log-prior* so that the EM algorithm can be used to tackle the ℓ_q−norm regularized ML estimation problem.

3. MAP: THE PRIOR PDF AS A MIXTURE OF DISTRIBUTIONS

Focusing on the prior distribution for MAP estimation, it is possible to express $p(\boldsymbol{\theta})$ as a marginal probability density function (pdf), occurring by integrating out the rest of the variables in a possible joint pdf. In this sense, one can express $p(\boldsymbol{\theta})$ as an infinite mixture, where there is an underlying process that generates the desired pdf for the parameters. More precisely:

$$p(\boldsymbol{\theta}) = \int p(\boldsymbol{\theta}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})d\boldsymbol{\lambda}, \quad (7)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^r$ is interpreted as a *hidden* (hyper)-parameter vector. If the conditional pdf $p(\boldsymbol{\theta}|\boldsymbol{\lambda})$ is Gaussian, then we obtain a Gaussian mixture, see, e.g., [12, 21, 22]. Hence, we can express the *log-prior* as

$$\log p(\boldsymbol{\theta}) = \mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) - \mathcal{H}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) \quad (8)$$

with

$$\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \log p(\boldsymbol{\theta}, \boldsymbol{\lambda})p\left(\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{\lambda},$$

$$\mathcal{H}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \log p(\boldsymbol{\lambda}|\boldsymbol{\theta})p\left(\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{\lambda}.$$

The auxiliary function $\mathcal{H}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$ is decreasing in $\boldsymbol{\theta}$[1]. Therefore, the EM algorithm can be formulated as follows:
E-step: Compute

$$\mathcal{Q}_{\text{MAP-EM}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \mathcal{Q}_{\text{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) + \mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right).$$

M-step: Solve

$$\hat{\boldsymbol{\theta}}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}_{\text{MAP-EM}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right).$$

For the M-step of the last EM algorithm, we need to evaluate the gradient of $\mathcal{Q}_{\text{MAP-EM}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$ with respect to $\boldsymbol{\theta}$. Assuming that $\nabla_{\boldsymbol{\theta}}\mathcal{Q}_{\text{ML}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$ is known from the ML estimation problem, we only need to evaluate $\nabla_{\boldsymbol{\theta}}\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$.

Lemma 1 *When expressing the prior pdf of $\boldsymbol{\theta}$, $p(\boldsymbol{\theta})$, as a VMGM and letting $\boldsymbol{\theta}|\boldsymbol{\lambda} \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{\lambda}), \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{\lambda}))$, the gradient of the auxiliary function $\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$, is given by*

$$\nabla_{\boldsymbol{\theta}} G(\boldsymbol{\theta})\big|_{\hat{\boldsymbol{\theta}}^{(i)}} = \quad E_{\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}}\left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})\right] \hat{\boldsymbol{\theta}}^{(i)}$$
$$+ E_{\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}}\left[\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{\lambda})\right]. \quad (9)$$

Proof: First, notice that the marginal of $\boldsymbol{\lambda}$ is independent of $\boldsymbol{\theta}$. Therefore,

$$\nabla_{\boldsymbol{\theta}}\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}|\boldsymbol{\lambda})p\left(\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{\lambda}.$$

Given that $\boldsymbol{\theta}|\boldsymbol{\lambda} \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\lambda}), \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{\lambda}))$, we obtain:

$$\nabla_{\boldsymbol{\theta}}\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right) = \int \left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})(\boldsymbol{\theta} - \boldsymbol{\mu}(\boldsymbol{\lambda}))\right] p\left(\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}\right) d\boldsymbol{\lambda}$$
$$= E_{\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}}\left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})\right] \boldsymbol{\theta}$$
$$+ E_{\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}}\left[\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})\boldsymbol{\mu}(\boldsymbol{\lambda}))\right]. \quad (10)$$

---

[1] In the same way as $\mathcal{H}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)$.

The expected values in the last equation can be computed based on the Gaussianity assumption for $\boldsymbol{\theta}|\boldsymbol{\lambda}$. Due to this assumption, $p(\boldsymbol{\theta})$ satisfies the following equation:

$$\nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}) = \int \left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})(\boldsymbol{\theta} - \boldsymbol{\mu}(\boldsymbol{\lambda})) \right] p(\boldsymbol{\theta}|\boldsymbol{\lambda}) \, p(\boldsymbol{\lambda}) d\boldsymbol{\lambda}. \tag{11}$$

On the other hand, Bayes rule yields:

$$p(\boldsymbol{\lambda}) = \frac{p(\boldsymbol{\lambda}|\boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{\lambda})}.$$

Plugging in the last expression into (11), we obtain:

$$\nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}) = \int \left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})(\boldsymbol{\theta} - \boldsymbol{\mu}(\boldsymbol{\lambda})) \right] p(\boldsymbol{\lambda}|\boldsymbol{\theta}) \, p(\boldsymbol{\theta}) d\boldsymbol{\lambda}.$$

Therefore,

$$\frac{1}{p(\boldsymbol{\theta})} \nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}) = E_{\boldsymbol{\lambda}|\boldsymbol{\theta}} \left[-\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{\lambda})(\boldsymbol{\theta} - \boldsymbol{\mu}(\boldsymbol{\lambda})) \right]. \tag{12}$$

We now note that (12) equals $\dot{G}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta})$. Nevertheless, $G(\boldsymbol{\theta})$ is known and so is $\dot{G}(\boldsymbol{\theta})$. Therefore, (9) follows. □

The expression in (9) corresponds to a linear system of equations that can be generally solved, because $p\left(\boldsymbol{\lambda}|\hat{\boldsymbol{\theta}}^{(i)}\right)$ is a known distribution. Thus, the expected values can be computed either in closed form when possible or using Monte Carlo techniques, such as the Metropolis-Hastings algorithm or the Gibbs sampler [23]. A particular case arises when $G(\boldsymbol{\theta})$ can be expressed as a function of the coordinate variables of $\boldsymbol{\theta}$ as follows:

$$G(\boldsymbol{\theta}) = \sum_{j=1}^{p} \kappa \left(\frac{\theta_j}{\tau} \right). \tag{13}$$

Here, τ is a factor controlling the strength of the regularization. Different selections of the kernel function $\kappa(\cdot)$ lead to different regularizations. Usual regularization choices can be obtained based on Gaussian mixtures, e.g., the Ridge regression for $\kappa(\theta_j/\tau) = (\theta_j/\tau)^2$ and the Lasso for $\kappa(\theta_j/\tau) = |\theta_j/\tau|$ [17]. The general form of a VMGM for a scalar parameter θ_j is given by

$$p(\theta_j) = \int_0^\infty \mathcal{N}\left(\mu_j + \lambda u_j, \tau^2 \lambda \right) p(\lambda) d\lambda. \tag{14}$$

Based on this formula, the Ridge regression is obtained for $u_j = \mu_j = 0$ and $\lambda = 1$ and the Lasso is obtained for $u_j = \mu_j = 0$ and λ exponentially distributed. Clearly, these two very usual regularizations are exactly expressed as VMGMs. For the cases where $\mu_j = 0$, the following lemma holds:

Lemma 2 *When* $\mu_j = 0, j = 1, 2, \ldots, p$, *then*

$$\frac{\partial \mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)}{\partial \theta_j} = \frac{1}{\theta_j^{(i)}} \kappa(\theta_j) \big|_{\theta_j = \theta_j^{(i)}} \theta_j. \tag{15}$$

Proof: See [17]. □

In the next section, we present in detail the VMGM descriptions of the ℓ_q-norm regularization term.

4. SPARSE PARAMETER ESTIMATION USING VMGMs

As it has already become apparent, the penalties that are usually considered for promoting sparsity can be interpreted as MAP problems. In this section, we explore the representation of the ℓ_q-norm as a *prior* distribution function for developing a sparse-promoting algorithm under the utilization of VMGMs. To this end, the density function that relates to the ℓ_q-norm is given by

$$p_{\ell_q}(\boldsymbol{\theta}) = \prod_{j=1}^{p} p_{\ell_q}(\theta_j), \tag{16}$$

$$p_{\ell_q}(\theta_j) = \frac{1}{2\Gamma\left(1 + \frac{1}{q}\right)\sqrt{\tau^2}} e^{-\frac{1}{\tau^q}|\theta_j|^q}, \tag{17}$$

which corresponds to a *leptokurtic* exponential power distribution [24]. The prior $p_{\ell_q}(\theta_j)$ in (17) can be expressed as a VMGM, with $p(\lambda_j)$ being a stable distribution in (14). Then, the MAP-EM algorithm can be used to solve the aforementioned sparsity-promoting estimation problems.

4.1. MAP-EM algorithm for sparse parameter estimation

In this section, we present the relevant expressions in the development of the MAP-EM algorithm for sparse-parameter estimation using the ℓ_q-norm.

Lemma 3 *In the case that the log-prior function is given by (13) and* $p(\theta_j)$ *is a VMGM as in (14),* $\frac{\partial \mathcal{Q}_{prior}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})}{\partial \theta_j}, \forall j = 1, 2, \ldots, p$, *is then given by*

$$\frac{\partial \mathcal{Q}_{prior}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})}{\partial \theta_j} = \int \left(-\frac{\theta_j}{\tau^2 \lambda_j} \right) p(\lambda_j | \hat{\theta}_j^{(i)}) d(\lambda_j)$$
$$= \left[-\frac{\theta_j}{\tau^2} E[\lambda_j^{-1} | \hat{\theta}_j^{(i)}] \right], \tag{18}$$

where $E[\lambda_j^{-1} | \hat{\theta}_j^{(i)}]$ *is the expectation obtained from*

$$-\frac{\hat{\theta}_j^{(i)}}{\tau^2} E[\lambda_j^{-1} | \hat{\theta}_j^{(i)}] = -\dot{\kappa}\left(\frac{\hat{\theta}_j^{(i)}}{\tau} \right). \tag{19}$$

Equivalently, we obtain

$$\frac{d\mathcal{Q}_{\text{prior}}\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}\right)}{d\boldsymbol{\theta}} = -\boldsymbol{K}^{(i)} \boldsymbol{\theta}, \tag{20}$$

where

$$\boldsymbol{K}^{(i)} = diag\left(\frac{1}{\hat{\theta}_1^{(i)}} \dot{\kappa}\left(\frac{\hat{\theta}_1^{(i)}}{\tau} \right), \ldots, \frac{1}{\hat{\theta}_p^{(i)}} \dot{\kappa}\left(\frac{\hat{\theta}_p^{(i)}}{\tau} \right) \right). \tag{21}$$

Proof: See [17]. □

Since the system is defined as a linear regression, it yields the derivative of $\mathcal{Q}_{\text{ML}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})$ with respect to $\boldsymbol{\theta}$ in the form:

$$\frac{d\mathcal{Q}_{\text{ML}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})}{d\boldsymbol{\theta}} = \frac{E[\boldsymbol{X}^T | \boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}] \boldsymbol{y} - E[\boldsymbol{X}^T \boldsymbol{X} | \boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}] \boldsymbol{\theta}}{\sigma^2}. \tag{22}$$

Then, combining (18) and (22), we obtain the derivative of $\mathcal{Q}_{\text{MAP-EM}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})$ with respect to $\boldsymbol{\theta}$ as:

$$\frac{d\mathcal{Q}_{\text{MAP-EM}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})}{d\boldsymbol{\theta}} = -(\boldsymbol{B}^{(i)} + \boldsymbol{K}^{(i)})\boldsymbol{\theta} + \boldsymbol{a}^{(i)}, \quad (23)$$

where $\boldsymbol{B}^{(i)} = E[\boldsymbol{X}^T\boldsymbol{X}|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]/\sigma^2$, $\boldsymbol{K}^{(i)}$ is defined in (21) and $\boldsymbol{a}^{(i)} = E[\boldsymbol{X}^T|\boldsymbol{y}, \hat{\boldsymbol{\theta}}^{(i)}]\boldsymbol{y}/\sigma^2$. In particular, the (component-wise) derivative in *Lemma* 3 for the ℓ_q−norm of $\boldsymbol{\theta}$ is given by

$$\frac{\partial\mathcal{Q}_{\text{prior}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})}{\partial\theta_j} = -\frac{q}{\tau}\left|\frac{\hat{\theta}_j^{(i)}}{\tau}\right|^{q-1}\frac{\text{sign}(\hat{\theta}_j^{(i)})}{\hat{\theta}_j^{(i)}}\theta_j. \quad (24)$$

Remark 1 *Notice that the MAP-EM algorithm yields an optimization problem whose solution is given by a system of linear equations, and hence, it is easy to solve[2]. This is due to the quadratic form of $\mathcal{Q}_{prior}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})$ with respect to $\boldsymbol{\theta}$.* ▽

Remark 2 ([17]) *In our approach, we have to deal with derivatives of functions of the form $d|a|/d\,a$, and the derivative at zero does not exist. However, it is possible to obtain the desired value at almost all points using*

$$\frac{d|a|}{d\,a} = sign(a), \quad \forall a \neq 0. \quad (25)$$

Most of the algorithms promoting sparsity have to deal with non-differentiable cost functions, see, e.g., [6,7], where a local approximation of the ℓ_1-norm is obtained. A common solution to overcome the issue of non-differentiability when $a \to 0$ in (25), is to start the algorithm from a value far from zero, and then remove the component a from the parameter vector when it enters in a neighbourhood (defined by the user) of $a = 0$ [13,16]. ▽

5. SIMULATIONS

In this section, we provide a numerical illustration for the applicability of our framework. For the simulations, we consider a limited number of measurement points, since, in general, MAP estimates converge to ML estimates as the number of data points increases. This is known as the *Bernstein-Von Mises theorem* [25]. In our experience, we have been able to observe this situation for the MAP-EM algorithm with the ℓ_q-norm, $0 < q \leq 1$.

For the example, we consider the following system model:
$$z_{k+1} = 0.9z_k + \xi_k,$$
$$y_k = \sin\left(\frac{\omega(k-1)}{N} + z_k\right)\boldsymbol{u}_{k:k+p-1}^T\boldsymbol{\theta} + \eta_k,$$

where $\boldsymbol{u}_{k:k+p-1} = [u_k, u_{k+1}, ..., u_{k+p-1}]^T$ is a known input, $\boldsymbol{\theta} = [-0.77, -1.55, 0, 0, 0, 0, 0.46]^T$ is the unknown sparse parameter vector, $N = 256$ is the number of measurement points, $\omega = 5$ is a known constant parameter, $\xi_k \sim \mathcal{N}(0, 0.1)$, and $\eta_k \sim \mathcal{N}(0, 0.1)$. For the estimation of $\boldsymbol{\theta}$, we consider the ℓ_q-norm as presented in this paper, with $q = 0.1$ and $\tau = 0.1$. For comparison purposes, we also consider the corresponding penalized ML problem and solve for it using the EM framework in Section 2.1 with $q = 0.1$ and $\tau = 0.1$ as well. Finally, we also consider the approach in [17] with $\tau = 1$, which corresponds to $q = 1$. Notice that the initial estimate was the same for the four estimation techniques.
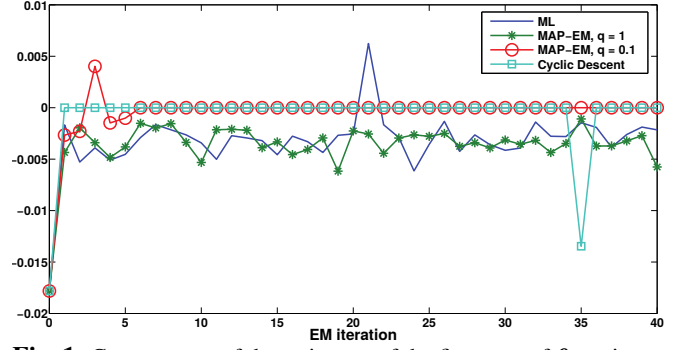

Fig. 1. Convergence of the estimates of the first zero of $\boldsymbol{\theta}$ per iteration of the EM algorithm.
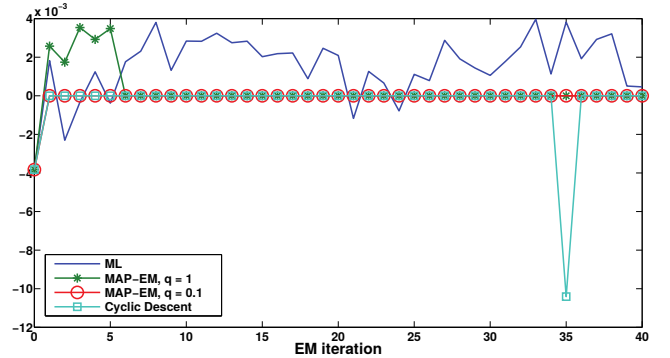

Fig. 2. Convergence of the estimates of the second zero of $\boldsymbol{\theta}$ per iteration of the EM algorithm.

The results of the estimation algorithms are shown in Figs. 1—4. From those figures, we can clearly see that the proposed approach can effectively estimate all the zeros of $\boldsymbol{\theta}$. However, the convergence is not achieved at the same time for each zero-entry of $\boldsymbol{\theta}$. Our approach clearly outperforms the ℓ_1-norm MAP-EM approach and ML. In contrast, the CD approach can accurately estimate the zeros of $\boldsymbol{\theta}$ at the first iteration of the EM algorithm. However, this estimation can vary from iteration to iteration, since the optimization procedure is carried out globally for the a single coordinate. As mentioned in [10], this kind of optimization should be carried out locally. From Figs. 1 and 2, we can see that at iteration no. 34, the zeros are correctly estimated. However, at iteration no. 35, those estimates are no longer zero. In contrast, our proposed method forces the zero value once the estimate converges to zero, following the ideas in *Remark* 2. In addition, the mean square error (MSE) of the estimates are: i) 3.56×10^{-5} for ML, ii) 1.124×10^{-4} for MAP-EM with $q = 1$, iii) 1.96×10^{-5} for MAP-EM with $q = 0.1$, and iv) 3.48×10^{-5} for CD. The difference in MSE is due to the slow convergence of the non-zero elements that CD exhibits. Thus, we believe that our estimation algorithm could be initialized with one iteration of CD for quicker convergence of the EM algorithm.

6. CONCLUSIONS

In this paper, we have described a general framework for the implementation of the EM algorithm based on mean-

---

[2]When compared to the equivalent nonlinear problems obtained by directly applying Lasso or the bridge regression [7] with $0 < q < 1$.
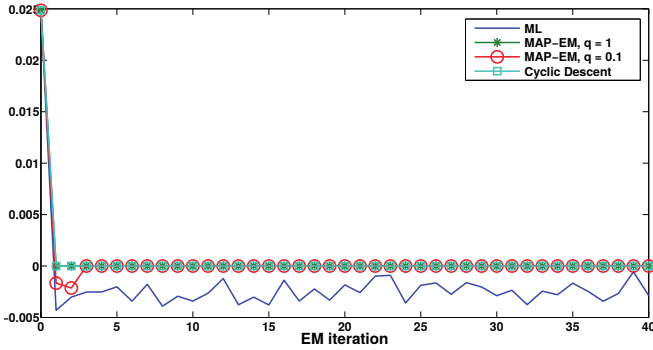
Fig. 3. Convergence of the estimates of the third zero of $\boldsymbol{\theta}$ per iteration of the EM algorithm.
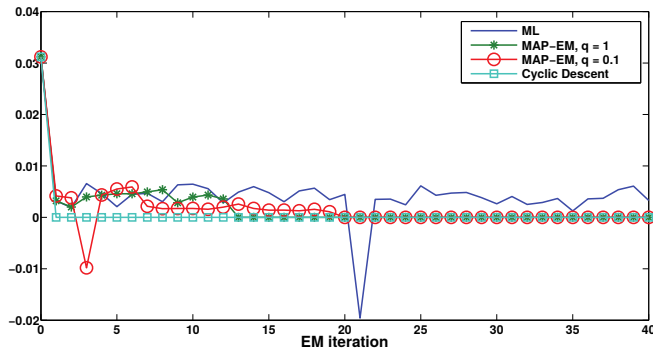

Fig. 4. Convergence of the estimates of the fourth zero of $\boldsymbol{\theta}$ per iteration of the EM algorithm.

variance Gaussian mixtures, for promoting sparsity via ℓ_q-norm regularization ($0 < q \leq 1$). Our proposal allows a (computationally-efficient) simple way of handling the sparsity estimation problem, yielding closed form expressions for the EM algorithm. A particular case of the proposed approach includes $q = 1$ (Lasso). The numerical illustration shows the effectiveness of our approach, outperforming both the ML and the Lasso estimators when choosing $q < 1$. In addition, the simulations show that our approach can obtain better estimates than Coordinate Descent within the EM framework. Since the Coordinate Descent algorithm is capable of finding the zeros quickly but may diverge, it can be used to initialize our approach for faster convergence.

7. REFERENCES

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. R. Stat. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.

[2] F. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, John Wiley & Sons, 2nd edition, 2008.

[3] D.L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[4] D.L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math*, vol. 59, no. 6, pp. 797–829, 2006.

[5] L. Baldassarre, J.M. Morales, A. Argyriou, and M. Pontil, "A general framework for structured sparsity via proximal optimization," in *15th Int. Conf. Artif. Intell. Stat. (AISTATS)*, La Palma, Canary Islands, Spain, April 2012.

[6] H. Zou and R. Li, "One-step sparse estimates in nonconcave penalized likelihood models," *The Annals of Statistics*, vol. 36, no. 4, pp. 1509–1533, 2008.

[7] J. Fan and J. Lv, "A selective overview of variable selection in high dimensional feature space," *Statistica Sinica*, vol. 20, no. 1, pp. 101–148, 2010.

[8] W. I. Zangwill, *Nonlinear programming: a unified approach*, Prentice-Hall, 1969.

[9] T. T. Wu and K. Lange, "Coordinate descent algorithms for lasso penalized regression," *Ann. Appl. Stat.*, vol. 2, no. 1, pp. 224–244, 2008.

[10] P. Breheny and J. Huang, "Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection," *Ann. Appl. Stat.*, vol. 5, no. 1, pp. 232–253, 2011.

[11] G. Marjanovic and V. Solo, "ℓ_q sparsity penalized linear regression with cyclic descent," *IEEE Trans. Signal Process.*, vol. 62, no. 6, pp. 1464–1475, 2014.

[12] O. Barndorff-Nielsen, J. Kent, and M. Sorensen, "Normal variance-mean mixtures and z distributions," *Int. Stat. Review*, vol. 50, pp. 145–159, 1982.

[13] M.A.T. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1150–1159, 2003.

[14] K. Lange and J. S. Sinsheimer, "Normal/independent distributions and their applications in robust regression," *J. Comput. Graph. Stat*, vol. 2, no. 2, pp. 175–198, 1993.

[15] D. E. Ba, B. Babadi, P. L. Purdon, and E. N. Brown, "Convergence and stability of iteratively re-weighted least squares algorithms.," *IEEE Trans. on Signal Process.*, vol. 62, no. 1, pp. 183–195, 2014.

[16] N.G. Polson and J.G. Scott, "Data augmentation for non-Gaussian regression models using variance-mean mixtures," *Biometrika*, vol. 100, no. 2, pp. 459–471, 2013.

[17] B. I. Godoy, J. C. Agero, R. Carvajal, G. C. Goodwin, and J. I. Yuz, "Identification of sparse FIR systems using a general quantisation scheme," *International Journal of Control*, vol. 87, no. 4, pp. 874–886, 2014.

[18] R. Carvajal, J.C. Agüero, B.I. Godoy, G.C. Goodwin, and J.I. Yuz, "EM-based identification of sparse FIR systems having quantized data," in *Proc. SYSID*, Brussels, Belgium, July 2012.

[19] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag, 2006.

[20] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal. Statist. Soc B*, vol. 58, no. 1, pp. 267–288, 1996.

[21] M. West, "On scale mixtures of normal distributions," *Biometrika*, vol. 74, no. 3, pp. 646–648, 1987.

[22] J. Keilson and F.W. Steutel, "Mixtures of distributions, moment inequalities and measures of exponentiality and normality," *The Annals of Probability*, vol. 2, no. 1, pp. 112–130, 1974.

[23] C.P. Robert and G. Casella, *Monte Carlo statistical methods*, Springer, New York, 1999.

[24] N. Solaro, "Random variate generation from multivariate exponential power distribution," *Statistica & Applicazioni*, vol. 2, no. 2, pp. 25–44, 2004.

[25] E. L. Lehmann, *Theory of point estimation*, Springer-Verlag, 2nd edition, 1998.