DISTRIBUTED MULTIVARIATE REGRESSION WITH UNKNOWN NOISE COVARIANCE IN THE PRESENCE OF OUTLIERS: AN MDL APPROACH

Roberto López-Valcarce[⋆] *Daniel Romero*[†] *Josep Sala*[‡] *Alba Pagès-Zamora*[‡]

[⋆] Universidade de Vigo–AtlantTIC, Spain
[†] University of Minnesota, USA
[‡] Universitat Politècnica de Catalunya–Barcelona Tech, Spain

ABSTRACT

We consider the problem of estimating the coefficients in a multivariable linear model by means of a wireless sensor network which may be affected by anomalous measurements. The noise covariance matrices at the different sensors are assumed unknown. Treating outlying samples, and their support, as additional nuisance parameters, the Maximum Likelihood estimate is investigated, with the number of outliers being estimated according to the Minimum Description Length principle. A distributed implementation based on iterative consensus techniques is then proposed, and it is shown effective for managing outliers in the data.

Index Terms— Multivariate regression, outliers, distributed estimation, wireless sensor networks.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) can be deployed to perform inference from measurement samples. Often in practice, a fraction of the measurements, termed *outliers,* do not adhere to the modeling assumptions [1–4]. Managing outliers is often critical for successful operation of the inference engine. In particular, multivariate analysis techniques are known to be very sensitive to anomalous data, which become difficult to detect [2, 5]. Due to computation and communication constraints, outlier detection in WSNs should ideally be carried out in a decentralized way. Different techniques have been proposed along these lines over the years, including nearest neighbor-based, clustering-based, classification-based, and statistical-based approaches [6, 7].

We consider the estimation of the coefficients of a multivariable linear model by means of a WSN. Each sensor obtains a set of multivariate observations, affected by measurement noise and possibly by outliers. The noise at different sensors is assumed uncorrelated, but we allow for correlation across the entries of the noise vector at a given sensor, with unknown covariance matrix which may differ from one sensor to another. This fact, together with the detrimental effect of outliers, makes the estimation problem quite challenging. We do not adopt any particular outlier generation model, and instead treat outliers as nuisance parameters to be jointly estimated with the wanted regression coefficients; however, rather than pursuing sparsity-promoting regularization of a Least Squares (LS) cost as in [8–10], we adopt a Minimum Description Length (MDL) approach [11, 12], by which the Maximum Likelihood estimates (MLEs) are first obtained for different candidate number of outliers in the data, and then the one minimizing the MDL cost is retained. Our method is based on cyclic optimization of the outliers' support and the parameter values. In addition, we present a decentralized variant of this MDL-based scheme which, being based on distributed iterative consensus techniques [13, 14], is well suited for WSN implementation, and whose effectiveness is illustrated via simulation examples.

Supported by the Spanish Government and the European Regional Development Fund (ERDF) (TEC2013-47020-C2-1/2-R COMPASS, TEC2013-41315-R DISNET, TEC2015-69648-REDC Red COMONSENS), the Galician Government and ERDF (GRC2013/009, R2014/037 REdTEIC and AtlantTIC), and the Catalan Government (2014 SGR 60 AGAUR).

2. PROBLEM STATEMENT

Consider a network of K sensors deployed to estimate a deterministic $d\times 1$ vector parameter $\boldsymbol{\theta}$. The k-th sensor observes N $p\times 1$ vectors of noisy observations, with $N\geq p$:

$$\boldsymbol{y}_{kn}=\boldsymbol{H}_{kn}\boldsymbol{\theta}+\boldsymbol{w}_{kn}+\boldsymbol{c}_{kn}, \qquad \begin{array}{rcl} k & = & 1,\ldots,K, \\ n & = & 1,\ldots,N, \end{array} \quad (1)$$

where $\boldsymbol{H}_{kn}$ are known $p\times d$ regressor matrices, and $\boldsymbol{w}_{kn}$ denotes zero-mean Gaussian noise. It is assumed that $\boldsymbol{w}_{k_1n_1}$ is statistically independent of $\boldsymbol{w}_{k_2n_2}$ for $(k_1,n_1)\neq(k_2,n_2)$, and that $\mathbb{E}\{\boldsymbol{w}_{kn}\boldsymbol{w}_{kn}^T\}=\boldsymbol{R}_k$, $k=1,\ldots,K$. The noise covariance matrices $\boldsymbol{R}_k$ are assumed unknown.

The vectors $\boldsymbol{c}_{kn}$ represent a bias which can model a temporal sensor malfunction, an abnormal measurement, or even malicious data injected by an adversary; we assume that only a small number h of them are nonzero. The parameter h is unknown, as well as the indices (k,n) for which $\boldsymbol{c}_{kn}\neq\boldsymbol{0}$. Lacking any *a priori* knowledge about the nonzero $\boldsymbol{c}_{kn}$, we

choose to regard them as deterministic unknown, rather than adopting a particular model. This allows us to adopt the MDL criterion in order to jointly estimate all unknown parameters.

For an index pair (k,n), let $q \triangleq (k-1)N+n$, i.e., $k = \lceil \frac{q}{N} \rceil$, $n = 1 + (q \mod N)$, and relabel the sets $\{\boldsymbol{y}_{kn}\}$, $\{\boldsymbol{H}_{kn}\}$, $\{\boldsymbol{c}_{kn}\}$ as $\{\boldsymbol{y}_q\}$, $\{\boldsymbol{H}_q\}$, $\{\boldsymbol{c}_q\}$ respectively, with q ranging from 1 to $Q = KN$. Under hypothesis $\mathcal{H}_h$, there are h nonzero $\boldsymbol{c}_{q_1}, \ldots, \boldsymbol{c}_{q_h}$, and the set of unknown parameters is

$$\Theta_h = \{\boldsymbol{\theta}, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_K, \boldsymbol{c}_{q_1}, \ldots, \boldsymbol{c}_{q_h}, q_1, \ldots, q_h\}. \quad (2)$$

Thus, the number of unknown parameters is $(p+1)h + C$, with C a constant independent of h. Only $\boldsymbol{\theta}$ is of interest, and the remaining elements of Θ_h constitute nuisance parameters.

According to the MDL principle [11, 12], one should choose the hypothesis that minimizes

$$\mathrm{MDL}(h) = -\log f(\boldsymbol{Y}; \hat{\Theta}_h \,|\, \mathcal{H}_h) + (p+1)\frac{h}{2}\log Q, \quad (3)$$

with $\boldsymbol{Y} = [\, \boldsymbol{y}_1 \; \boldsymbol{y}_2 \; \cdots \; \boldsymbol{y}_Q \,]$ the matrix of observations, f the corresponding pdf, and $\hat{\Theta}_h$ the MLE of Θ_h under $\mathcal{H}_h$. For the Gaussian model (1), the pdf under $\mathcal{H}_h$, and given Θ_h, is

$$f(\boldsymbol{Y}; \Theta_h \,|\, \mathcal{H}_h) = \left[\prod_{k=1}^{K} \frac{\exp(-\operatorname{Tr}\{\boldsymbol{R}_k^{-1}\hat{\boldsymbol{S}}_k\})}{(2\pi)^p \det \boldsymbol{R}_k} \right]^{\frac{N}{2}} \quad (4)$$

where, for $k = 1, \ldots, K$,

$$\hat{\boldsymbol{S}}_k \triangleq \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\boldsymbol{\theta} - \boldsymbol{c}_{kn})(\boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\boldsymbol{\theta} - \boldsymbol{c}_{kn})^T, \quad (5)$$

and where we are implicitly assuming that $\boldsymbol{c}_{kn} = \boldsymbol{0}$ for $(k,n) \notin \{(k_1,n_1), \ldots, (k_h,n_h)\}$.

3. CENTRALIZED ESTIMATION

Suppose that all the data $\{\boldsymbol{y}_{kn}\}$ is available at a processing center. In order to obtain the MLE under $\mathcal{H}_h$, consider the partition $\Theta_h = \mathcal{P}_h \cup \mathcal{C}_h$, where

$$\mathcal{P}_h = \{\boldsymbol{\theta}, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_K, \boldsymbol{c}_{q_1}, \ldots, \boldsymbol{c}_{q_h}\}, \quad (6)$$

$$\mathcal{C}_h = \{q_1, \ldots, q_h\}. \quad (7)$$

(Alternatively, and using $q_i = (k_i - 1)N + n_i$ for $i = 1, \ldots, h$, we write $\mathcal{C}_h = \{(k_1,n_1), \ldots, (k_h,n_h)\}$). Our approach is based on cyclic optimization: first, the MLE of $\mathcal{P}_h$ is found for a given value of $\mathcal{C}_h$. Then, for the obtained value of $\mathcal{P}_h$, the MLE of $\mathcal{C}_h$ is sought, and the procedure is iterated.

3.1. MLE of $\mathcal{P}_h$

Given $\mathcal{C}_h$, (4) can be readily maximized w.r.t. $\{\boldsymbol{c}_{kn}\}$:

$$\hat{\boldsymbol{c}}_{kn} = \boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\boldsymbol{\theta}, \qquad (k,n) \in \mathcal{C}_h. \quad (8)$$

For each $k = 1, \ldots, K$, define now the set

$$\mathcal{S}_k = \{n \mid 1 \le n \le N \text{ and } (k,n) \notin \mathcal{C}_h\}. \quad (9)$$

Then, substituting (8) in (5), the matrices $\hat{\boldsymbol{S}}_k$ become

$$\hat{\boldsymbol{S}}_k \triangleq \frac{1}{N}\sum_{n \in \mathcal{S}_k} (\boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\boldsymbol{\theta})(\boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\boldsymbol{\theta})^T. \quad (10)$$

From (4), the negative of the log-likelihood function is

$$-\log f \propto \frac{N}{2}\sum_{k=1}^{K} \left(\log\det \boldsymbol{R}_k + \operatorname{Tr}\{\boldsymbol{R}_k^{-1}\hat{\boldsymbol{S}}_k\} \right). \quad (11)$$

Since no structure is imposed on $\boldsymbol{R}_k$, the values minimizing (11) are given by $\hat{\boldsymbol{R}}_k = \hat{\boldsymbol{S}}_k$, $k = 1, \ldots, K$, yielding

$$-\log f \propto \frac{KNp}{2} + \frac{N}{2}\sum_{k=1}^{K} \log\det \hat{\boldsymbol{S}}_k. \quad (12)$$

Recall now that, for $\alpha \in \mathbb{R}$ and $\boldsymbol{X}(\alpha) \in \mathbb{R}^{p \times p}$,

$$\frac{\partial}{\partial\alpha} \log\det \boldsymbol{X}(\alpha) = \operatorname{Tr}\left\{ \boldsymbol{X}^{-1}(\alpha)\frac{\partial}{\partial\alpha}\boldsymbol{X}(\alpha) \right\}. \quad (13)$$

Therefore, differentiating (12) w.r.t. $\boldsymbol{\theta}$ and taking (10) into account, after straightforward manipulations it is found that

$$\nabla_{\boldsymbol{\theta}} \sum_{k=1}^{K} \log\det \hat{\boldsymbol{S}}_k = \boldsymbol{0} \quad \Rightarrow$$

$$\left[\sum_{k=1}^{K} \sum_{n \in \mathcal{S}_k} \boldsymbol{H}_{kn}^T \hat{\boldsymbol{S}}_k^{-1} \boldsymbol{H}_{kn} \right] \boldsymbol{\theta} = \sum_{k=1}^{K} \sum_{n \in \mathcal{S}_k} \boldsymbol{H}_{kn}^T \hat{\boldsymbol{S}}_k^{-1} \boldsymbol{y}_{kn}. \quad (14)$$

Since the $\hat{\boldsymbol{S}}_k$ depend on $\boldsymbol{\theta}$ via (10), (14) is a nonlinear equation in $\boldsymbol{\theta}$, and no closed-form solution is known. Nevertheless, the structure of (14) suggests an iterative approach. Given an estimate $\hat{\boldsymbol{\theta}}^{(t)}$, the matrices $\hat{\boldsymbol{S}}_k$ are computed as in (10) with $\boldsymbol{\theta} \leftarrow \hat{\boldsymbol{\theta}}^{(t)}$. These are then used in (14), which becomes linear in $\boldsymbol{\theta}$. The corresponding solution is taken as $\hat{\boldsymbol{\theta}}^{(t+1)}$, and the process is repeated until convergence. The iteration may be initialized by setting $\hat{\boldsymbol{S}}_k^{-1} = \boldsymbol{I}$ for all k in (14), which yields the standard LS estimate for $\hat{\boldsymbol{\theta}}^{(1)}$ computed after leaving out those samples with indices in $\mathcal{C}_h$.

3.2. MLE of $\mathcal{C}_h$

Assume $\mathcal{P}_h$ given, and let $\hat{\boldsymbol{\theta}}$ be the corresponding estimate of $\boldsymbol{\theta}$. Note that finding $\mathcal{C}_h$ is equivalent to finding the sets $\mathcal{S}_k$, $k = 1, \ldots, K$ in (9). In view of (12), one must solve

$$\min_{\mathcal{S}_1, \ldots, \mathcal{S}_K} \sum_{k=1}^{K} \log\det \hat{\boldsymbol{S}}_k \quad \text{s. to} \quad \sum_{k=1}^{K} |\mathcal{S}_k| = KN - h. \quad (15)$$

This is a combinatorial problem, and thus very hard to solve by exhaustive search. The following result is inspired by [15] and suggests a computationally efficient heuristic approach to approximately solve (15).

Theorem 1. *Consider vectors* $\{\boldsymbol{v}_{kn},\ 1 \le k \le K,\ 1 \le n \le N\} \subset \mathbb{R}^p$*, and sets* $\mathcal{S}_k \subseteq \{1,\dots,N\}$ *such that* $\sum_{k=1}^K |\mathcal{S}_k| = H$*. Let* $\boldsymbol{A}_k = \sum_{n\in\mathcal{S}_k} \boldsymbol{v}_{kn}\boldsymbol{v}_{kn}^T$*. If* $\det \boldsymbol{A}_k \neq 0$*, define*

$$d_{kn}^2 = \boldsymbol{v}_{kn}^T \boldsymbol{A}_k^{-1} \boldsymbol{v}_{kn}. \tag{16}$$

Let $\mathcal{S}'_k$*,* $k = 1,\dots,K$ *be the sets of indices corresponding to the* H *smallest values of* d_{kn}^2*, and let* $\boldsymbol{B}_k = \sum_{n\in\mathcal{S}'_k} \boldsymbol{v}_{kn}\boldsymbol{v}_{kn}^T$*. Then* $\sum_{k=1}^K |\mathcal{S}'_k| = H$*, and it holds that*

$$\sum_{k=1}^K \log\det \boldsymbol{B}_k \le \sum_{k=1}^K \log\det \boldsymbol{A}_k, \tag{17}$$

with equality iff $\boldsymbol{B}_k = \boldsymbol{A}_k$*,* $k = 1,\dots,K$*.*

The proof can be developed along the lines of that of [15, Th. 1], and is omitted for brevity. In view of Theorem 1, it is possible to construct a non-negative, non-increasing (hence convergent) sequence of objective values for (15) as follows. Set $\boldsymbol{v}_{kn} = \boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\hat{\boldsymbol{\theta}}$ (residuals). Then, given a candidate collection of $H = KN - h$ uncontaminated samples $\mathcal{S}_1^{(t)} \cup \cdots \cup \mathcal{S}_K^{(t)}$, construct $\boldsymbol{A}_k$ and compute d_{kn}^2 as per (16). Obtain the new sets $\mathcal{S}_1^{(t+1)}$,..., $\mathcal{S}_K^{(t+1)}$ by leaving out the h samples yielding the largest d_{kn}^2 values, and iterate. Typically, convergence is attained after a few steps. For initialization, one can set $\boldsymbol{A}_k^{-1} = \boldsymbol{I}$ for all k, which in the first iteration leaves out the h samples with largest norm of the residual.

3.3. Cyclic optimization

The above procedures can be applied cyclically as follows. Let $h_{\max}$ be an upper bound to the number of contaminated samples. Then for each $h = 0, 1, \dots, h_{\max}$, one performs:

- Set $\hat{\boldsymbol{\theta}}_h^{[0]}$ to the LS estimate using all KN observations.
- Set $i = 0$ and repeat until convergence:
  1. Given $\hat{\boldsymbol{\theta}}_h^{[i]}$, estimate $\mathcal{C}_h^{[i+1]}$ as in Sec. 3.2.
  2. Given $\mathcal{C}_h^{[i+1]}$, estimate $\hat{\boldsymbol{\theta}}_h^{[i+1]}$ as in Sec. 3.1.
  3. Set $i \leftarrow i + 1$.
- Let $\hat{\boldsymbol{\theta}}_h$ and $\mathcal{C}_h$ be the values after convergence. Compute $\hat{\boldsymbol{S}}_k$ via (10) with $\boldsymbol{\theta} \leftarrow \hat{\boldsymbol{\theta}}_h$ and with $\mathcal{S}_k$ as in (9).
- Let $\mathrm{MDL}(h) = \frac{N}{2}\sum_{k=1}^K \log\det \hat{\boldsymbol{S}}_k + (p+1)\frac{h}{2}\log KN$.

The estimates are $\hat{\boldsymbol{\theta}}_{h_\star}$ and $\mathcal{C}_{h_\star}$, with $h_\star = \arg\min_h \mathrm{MDL}(h)$.

4. DISTRIBUTED ESTIMATION

The methods in Sec. 3 are centralized, in the sense that they make use of the whole dataset.We now focus on decentralized schemes in which there is no central processing center and the data is distributed over the network, i.e., node k only has access to its samples $\{\boldsymbol{y}_{kn},\ \forall\, n\}$. Nodes can only communicate with their neighbors, and the network is assumed connected (there exists a path connecting any two nodes) and undirected (any pair of neighboring nodes can both talk and listen to each other). The method proposed below involves a symmetric *weight matrix* $\boldsymbol{W} \in \mathbb{R}^{K\times K}$ with elements $W_{ij} \neq 0$ only if nodes i and j are neighbors[1], and satisfying (see [13])

$$\mathbf{1}^T\boldsymbol{W} = \mathbf{1}^T, \quad \boldsymbol{W}\mathbf{1} = \mathbf{1}, \quad \rho\left(\boldsymbol{W} - \tfrac{1}{K}\mathbf{1}\mathbf{1}^T\right) < 1, \tag{18}$$

with $\mathbf{1} \in \mathbb{R}^K$ the all-ones vector, and $\rho(\cdot)$ the spectral radius.

The distributed method is iterative in nature, and can be summarized as follows. Let t be the iteration number. Each node k keeps a local estimate $\hat{\boldsymbol{\theta}}_k^{(t)}$, as well as auxiliary variables $\boldsymbol{\Phi}_k^{(t)}$ and $\boldsymbol{\psi}_k^{(t)}$. Using this local estimate $\hat{\boldsymbol{\theta}}_k^{(t)}$, it runs the procedure of Sec. 3.2, particularized to the case $K = 1$, over its data $\{\boldsymbol{y}_{kn},\ \forall\, n\}$; i.e., with the residuals $\boldsymbol{v}_{kn}^{(t)} - \boldsymbol{H}_{kn}\hat{\boldsymbol{\theta}}_k^{(t)}$, $n = 1,\dots,N$, the MDL cost is computed for different values of h using the method from Sec. 3.2. Then, node k obtains a candidate set of uncontaminated samples $\mathcal{S}_k^{(t)}$, by taking the value of h minimizing this local MDL cost.

The next step is based on the observation that the matrix and vector featuring respectively in the left- and right-hand side of (14), and which constitute global quantities, are the summation of local quantities. This suggests the use of consensus-based techniques in order for the network to iteratively compute the global quantities by means of local exchanges. First, at each node k the following are computed:

$$\boldsymbol{v}_{kn}^{(t)} = \boldsymbol{y}_{kn} - \boldsymbol{H}_{kn}\hat{\boldsymbol{\theta}}_k^{(t)}, \tag{19}$$

$$\hat{\boldsymbol{S}}_k^{(t)} = \frac{1}{N}\sum_{n\in\mathcal{S}_k^{(t)}} \boldsymbol{v}_{kn}^{(t)}\left(\boldsymbol{v}_{kn}^{(t)}\right)^T, \tag{20}$$

$$\boldsymbol{M}_k^{(t)} = \sum_{n\in\mathcal{S}_k^{(t)}} \boldsymbol{H}_{kn}^T\left(\hat{\boldsymbol{S}}_k^{(t)}\right)^{-1}\boldsymbol{H}_{kn}, \tag{21}$$

$$\boldsymbol{g}_k^{(t)} = \sum_{n\in\mathcal{S}_k^{(t)}} \boldsymbol{H}_{kn}^T\left(\hat{\boldsymbol{S}}_k^{(t)}\right)^{-1}\boldsymbol{y}_{kn}. \tag{22}$$

After the information exchange with their neighbors, the nodes update $\boldsymbol{\Phi}_k^{(t-1)}$, $\boldsymbol{\psi}_k^{(t-1)}$ as follows:

$$\boldsymbol{\Phi}_k^{(t)} = \sum_j W_{kj}\left[\beta^{(t)}\boldsymbol{\Phi}_j^{(t-1)} + \alpha^{(t)}\boldsymbol{M}_j^{(t)}\right], \tag{23}$$

$$\boldsymbol{\psi}_k^{(t)} = \sum_j W_{kj}\left[\beta^{(t)}\boldsymbol{\psi}_j^{(t-1)} + \alpha^{(t)}\boldsymbol{g}_j^{(t)}\right]. \tag{24}$$

Finally, the local estimate of $\boldsymbol{\theta}$ is updated by solving

$$\boldsymbol{\Phi}_k^{(t)}\hat{\boldsymbol{\theta}}_k^{(t+1)} = \boldsymbol{\psi}_k^{(t)}. \tag{25}$$

The quantities to be exchanged are those in brackets in (23)-(24). These updates are inspired by "consensus+innovations" techniques [14, 16] for distributed estimation. By *innovation* we mean the new information supplied

[1] By convention, every node is neighbor to itself.

by the update of $\boldsymbol{M}_j^{(t)}$ and $\boldsymbol{g}_j^{(t)}$ (rather than that brought in by new samples, as in [14]). The sequences $\beta^{(t)}$, $\alpha^{(t)}$ are suitable time-varying weights for the consensus and innovation terms, respectively. A possible choice is

$$\alpha^{(t)} = \frac{1}{t}, \qquad \beta^{(t)} = 1 - \frac{1}{t^{\delta}}, \quad 0 < \delta < 1, \quad t \geq 1. \quad (26)$$

Thus $\alpha^{(t)}$ monotonically decreases to 0 from $\alpha^{(1)} = 1$, whereas $\beta^{(t)}$ monotonically increases to 1 from $\beta^{(1)} = 0$. As a consequence, in the beginning of the iterative process the innovations term is dominant, and a gradual switch takes place in order to drive the network toward consensus. A simple way to initialize the iteration is to have each node k take $\hat{\boldsymbol{\theta}}_k^{(1)}$ as the standard LS estimate of $\boldsymbol{\theta}$ based on its local data[2].

5. SIMULATION RESULTS

We considered a network of $K = 20$ nodes, randomly deployed on a 1×1 square, and collecting $N = 25$ samples each. Nodes are linked if their distance is less than 0.7, the weight matrix $\boldsymbol{W}$ was built by the Metropolis rule [17], and $\delta = 0.5$ was taken in (26). The matrices $\boldsymbol{H}_{kn}$ and $\boldsymbol{R}_k^{1/2}$ were randomly generated with i.i.d. Gaussian entries in each of 100 Monte Carlo runs. For given $\{\boldsymbol{H}_{kn}\}$ and $\{\boldsymbol{R}_k\}$, the SNR is

$$\text{SNR} = \frac{\sum_{k,n} \|\boldsymbol{H}_{kn}\boldsymbol{\theta}\|^2}{N \sum_k \text{Tr}\{\boldsymbol{R}_k\}} \leq \frac{\sum_{k,n} \|\boldsymbol{H}_{kn}\|_F^2}{N \sum_k \text{Tr}\{\boldsymbol{R}_k\}} \|\boldsymbol{\theta}\|^2. \quad (27)$$

For simplicity, the upper bound in (27) is taken as the SNR in the simulations, and the matrices $\boldsymbol{R}_k^{1/2}$ are properly scaled at each run to yield the desired value. As performance metric we consider the Normalized Mean Square Error:

$$\text{NMSE} = \frac{1}{K} \sum_{k=1}^{K} \frac{\mathbb{E}\{\|\hat{\boldsymbol{\theta}}_k^{(t)} - \boldsymbol{\theta}\|^2\}}{\|\boldsymbol{\theta}\|^2}. \quad (28)$$

Contamination vectors $\boldsymbol{c}_{kn}$, with i.i.d. entries drawn from a uniform distribution in $[-500,\ 500]$, were randomly added to the samples with a given probability of occurrence p_c. We set $d = 4$ and $\boldsymbol{\theta} = [\,-0.7172\ \ -0.0878\ \ 0.6414\ \ 0.2578\,]^T$.

Three distributed schemes were tested. First, the decentralized LS algorithm from [17], adapted to the case of multiple samples per sensor. Since this scheme requires knowledge of the noise covariance matrices, which in our setting is not available, it was run with the assumption that $\boldsymbol{R}_k = \boldsymbol{I}$ for all k. The second scheme is a variant of the distributed algorithm from Sec. 4 in which the nodes are oblivious to the potential presence of outliers, so they do not run the outlier detection procedure of Sec. 3.2 on their data. Finally, the complete distributed scheme of Sec. 4 including outlier detection at each individual node (taking $h_{\max} = 10$) is also tested.

[2]Since $\beta^{(1)} = 0$, the initial values $\boldsymbol{\Phi}_j^{(0)}$, $\boldsymbol{\psi}_j^{(0)}$ in (23)-(24) are irrelevant.

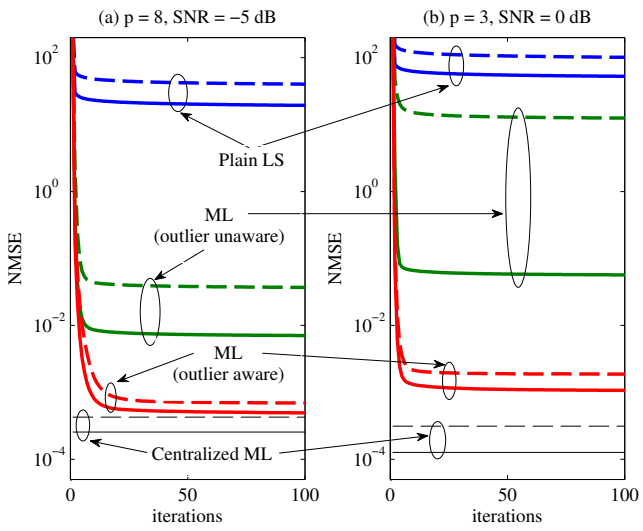

Fig. 1. NMSE trajectories for the distributed algorithms. 10% (solid) and 20% (dashed) of outliers in the data.

Results are shown in Fig. 1, together with the NMSE of the centralized method of Sec. 3 as benchmark. Two scenarios were considered: $(p = 8, \text{SNR} = -5\,\text{dB})$ and $(p = 3, \text{SNR} = 0\,\text{dB})$. For each setting, two cases were simulated, corresponding to $p_c = 0.1$ and 0.2. The LS algorithm from [17] performs poorly, due to outliers and noise correlation. Even without outlier detection, the proposed scheme improves over plain LS since it incorporates estimation of the unknown noise covariance matrices; nevertheless, the influence of outliers is seen to be detrimental, more so for smaller p. By including the outlier detection step at the nodes, performance is significantly improved in both scenarios, getting close to the benchmark of the centralized method.

6. CONCLUSIONS

We have proposed a consensus-based distributed algorithm for estimating a parameter vector in a multivariate Gaussian linear model with unknown noise covariances and contaminated by outliers. By regarding outliers as deterministic nuisance parameters and advocating the MDL principle to estimate their number, only an upper bound to the fraction of outliers in the data is to be set; in particular, it becomes unnecessary neither to specify an outlier generation model, nor to tune a detection threshold or a sparsity-controling regularization parameter, as in other anomaly detection schemes.

One difference between the centralized and distributed implementations is that the former performs the outlier estimation step over the complete dataset, whereas in the latter each node runs such step on its own data. Although this allows decentralized operation, it entails some performance loss with respect to the centralized version. Investigating means to overcome this loss is the object of ongoing research.

7. REFERENCES

[1] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley, 1986.

[2] V. Barnett and T. Lewis, *Outliers in Statistical Data*, Wiley, 3rd edition, 1994.

[3] P. J. Huber and E. M. Ronchetti, *Robust Statistics*, Wiley, 2nd edition, 2009.

[4] A.M. Zoubir, V. Koivunen, Y. Chakhchoukh, and M. Muma, "Robust estimation in signal processing," *IEEE Signal Process. Mag.*, vol. 29, no. 4, pp. 61–80, Jul. 2012.

[5] M. Hubert, P. J. Rousseeuw, and S. Van Aelst, "High-breakdown robust multivariate methods," *Statistical Science*, vol. 23, no. 1, pp. 92–119, 2008.

[6] Yang Zhang, N. Meratnia, and P. Havinga, "Outlier detection techniques for wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 2, pp. 159–170, 2010.

[7] A. Mahapatro and P. M. Khilar, "Fault diagnosis in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 2000–2026, 2013.

[8] E. Candes and P. Randall, "Highly robust error correction by convex programming," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 2829–2840, 2008.

[9] G. B. Giannakis, G. Mateos, S. Farahmand, V. Kekatos, and H. Zhu, "USPACOR: Universal sparsity-controlling outlier rejection," in *Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2011, pp. 1952–1955.

[10] Y. Jin and B. D. Rao, "Algorithms for robust linear regression by exploiting the connection to sparse signal recovery," in *Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2010, pp. 3830–3833.

[11] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–658, 1978.

[12] S. M. Kay, *Fundamentals of Statistical Signal Processing, vol. 2: Detection Theory*, Prentice-Hall, 1998.

[13] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[14] S. Kar and J. M. F. Moura, "Consensus + innovations distributed inference over networks," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 99–109, May 2013.

[15] P. J. Rousseeuw and K. Van Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, Aug 1999.

[16] R. López-Valcarce, S. Silva Pereira, and A. Pagès-Zamora, "Distributed total least squares estimation over networks," in *Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2014, pp. 7580–7584.

[17] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Int. Symp. Inf. Process. Sensor Netw. (IPSN)*, 2005, pp. 63–70.