

# Robust Estimation of the Covariance Matrix From Data With Outliers

PETRE STOICA <sup>1</sup>, PRABHU BABU <sup>2</sup>,  
AND PIYUSH VARSHNEY <sup>2</sup> (Graduate Student Member, IEEE)

<sup>1</sup>Division of Systems and Control, Department of Information Technology, Uppsala University, 75237 Uppsala, Sweden

<sup>2</sup>Centre for Applied Research in Electronics (CARE), Indian Institute of Technology Delhi, New Delhi 110016, India

CORRESPONDING AUTHOR: PRABHU BABU (e-mail: prabhubabu@care.iitd.ac.in).

The work of Petre Stoica was supported by the Swedish Research Council under VR Grant 2017-04610, Grant 2016-06079, and Grant 2021-05022.

**ABSTRACT** The robust estimation of the covariance matrix is a frequent task in practical applications in which, more often than not, some data samples are outliers. There are several methods that can be used to robustly estimate a covariance matrix from corrupted data, a representative example of which is the **minimum covariance determinant (MCD)** method. In this paper we present a maximum conditional likelihood interpretation of MCD that provides a new motivation of as well as further insights into this method. To perform at its best MCD requires information on the number of outliers in the data, which usually is not available. We propose two new methods for covariance matrix estimation from data with outliers that do not suffer from this problem: **TEST** (multiple-hypothesis **testing** method) which uses the **FDR** (false discovery rate) to test a set of model hypotheses and hence estimate the number of outliers and their locations, and **LIKE** (penalized **likelihood** method) that solves the outlier estimation problem using a **GIC** (generalized information criterion) to penalize the complexity of a high-dimensional data model. We show by means of numerical simulations that the performances of **TEST** and **LIKE** are relatively similar to one another as well as to the performance of the oracle MCD (which uses the true number of outliers) and significantly better than the performance of MCD that uses an upper bound on the outlier number.

**INDEX TERMS** Robust covariance matrix estimation, outlier detection, minimum covariance determinant, false discovery rate.

## I. INTRODUCTION

The estimation of a covariance matrix from data containing outliers is an omnipresent problem in applications and consequently it has received significant attention in the literature. The statistical literature contains a multitude of works on this subject, see for example [1], [2], [3], [4], [5], [6], [7]. Comparisons of performance reported in these papers have found that the minimum covariance determinant (MCD) method proposed in [1], [3] outperformed most of the other state-of-the-art procedures for robust covariance matrix estimation. The problem of covariance matrix estimation from data with outliers has been investigated also in the engineering literature, see for instance [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. The MCD has been found to be one of the most competitive methods for robust covariance matrix estimation in some of these works as well, see for example [13],

[14], [17], [18]. In the present paper we will consider and use the MCD as a reference method (a.k.a. benchmark) due to its simplicity and frequent use in a host of diverse applications. We begin this paper by showing that MCD is a maximum conditional-likelihood method, which provides not only a new motivation of but also further insights into MCD. A difficulty with using MCD in practice is that it requires information on the number of outliers in order to achieve its full potential. Because such information is rarely available in applications we make use of the false discovery rate (FDR) (see, e.g., [19], [20]) and the generalized information criterion (GIC) ([21], [22]) to propose a multi-hypothesis **testing** method (**TEST**) and, respectively, a penalized **likelihood** procedure (**LIKE**) for estimating the number of outliers and their positions in the data string. We use numerical simulations to show that the performances of **TEST** and **LIKE** are relatively similar to

one another as well as to the performance of the oracle MCD (which uses the true number of outliers) and significantly better than the performance of MCD that uses an upper bound on the outlier number.

## II. PROBLEM FORMULATION

To formally state the problem, let  $\mathbf{y}_t \in \mathcal{R}^n$  (for  $t = 1, \dots, N$ ) denote the collected data samples, out of which  $N_c$  samples are corrupted (i.e. outliers):  $\{\mathbf{y}_t\}$  for  $t \in C \subset \{1, \dots, N\}$  with  $|C| = N_c$ . The rest of the samples are uncorrupted:  $\{\mathbf{y}_t\}$  for  $t \in U$ , with  $|U| = N_u$  and  $N_u + N_c = N$ . The latter samples are assumed to be independently drawn from a normal distribution with zero mean and covariance matrix  $\mathbf{R}$ :

$$p(\mathbf{y}_t; \mathbf{R}) = \frac{1}{(2\pi)^{n/2} |\mathbf{R}|^{1/2}} e^{-\frac{1}{2} \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t} \quad \text{for } t \in U. \quad (1)$$

Note that we use the same symbol  $|\cdot|$  to denote either the cardinality of a discrete set or the determinant of a matrix, depending on the context. Also note that, in order to keep the notation and the following discussion as simple as possible we have assumed that the mean in (1) is equal to zero. The case of nonzero (unknown) mean is slightly more complicated but mostly only at the algorithmic level (see, e.g., [1], [14]).

The problem considered in the following sections is the estimation of  $\mathbf{R}$  as well as  $U$  (and implicitly  $N_u$ ) from  $\{\mathbf{y}_t\}_{t=1}^N$ .

*Notation:* When dealing with quadratic expressions of the form  $\mathbf{y}_t^T \mathbf{A}^{-1} \mathbf{y}_t$ , where  $\mathbf{A}$  is a positive definite matrix, we always assume that the samples are ordered such that:

$$\mathbf{y}_1^T \mathbf{A}^{-1} \mathbf{y}_1 \geq \mathbf{y}_2^T \mathbf{A}^{-1} \mathbf{y}_2 \geq \dots \geq \mathbf{y}_N^T \mathbf{A}^{-1} \mathbf{y}_N. \quad (2)$$

The indexes of the ordered samples obviously depend on the matrix  $\mathbf{A}$ . This convention avoids the use of a different symbol for the indexes of the permuted samples (such as  $[t]$  or  $\pi(t)$  that is sometimes used in such a case) and hence simplifies the notation.

## III. MINIMUM COVARIANCE DETERMINANT METHOD: MCD

The original derivation of the MCD algorithm in [1] relied on the following simple result. Let  $\mathcal{W}$  be any subset of  $\{1, \dots, N\}$  with  $|\mathcal{W}| = N_u$  and let (for  $N_u \geq n$ )

$$\tilde{\mathbf{R}} = \frac{1}{N_u} \sum_{t \in \mathcal{W}} \mathbf{y}_t \mathbf{y}_t^T. \quad (3)$$

Then:

$$|\tilde{\mathbf{R}}| \geq |\hat{\mathbf{R}}|, \quad (4)$$

where

$$\hat{\mathbf{R}} = \frac{1}{N_u} \sum_{t=1}^{N_u} \mathbf{y}_{N+1-t} \mathbf{y}_{N+1-t}^T \quad (5)$$

and where the samples are ordered as in (2) with  $\mathbf{A} = \tilde{\mathbf{R}}$ . Furthermore the equality in (4) holds if and only if  $\hat{\mathbf{R}} = \tilde{\mathbf{R}}$ .

---

### Algorithm 1: MCD.

---

**Input:**  $\{\mathbf{y}_t\}_{t=1}^N, N_u$

**Output:**  $\hat{\mathbf{R}}$

- 1 Initialize  $\hat{\mathbf{R}}$
  - 2 **repeat**
  - 3     Order the samples as in (2) with  $\mathbf{A} = \hat{\mathbf{R}}$
  - 4     Compute  $\hat{\mathbf{R}}$  using (5)
  - 5 **until** convergence
- 

Next we present a simple proof of this result (which is more direct than other proofs in the literature, see e.g. [1], [14]):

$$\begin{aligned} (|\hat{\mathbf{R}}|/|\tilde{\mathbf{R}}|)^{1/n} &= |\hat{\mathbf{R}}\tilde{\mathbf{R}}^{-1}|^{1/n} \\ &= \left| \frac{1}{N_u} \sum_{t=1}^{N_u} \mathbf{y}_{N+1-t} \mathbf{y}_{N+1-t}^T \tilde{\mathbf{R}}^{-1} \right|^{1/n} \\ &\leq \frac{1}{n} \frac{1}{N_u} \sum_{t=1}^{N_u} \mathbf{y}_{N+1-t}^T \tilde{\mathbf{R}}^{-1} \mathbf{y}_{N+1-t} \\ &\leq \frac{1}{n} \text{Tr}(\tilde{\mathbf{R}}\tilde{\mathbf{R}}^{-1}) = 1 \end{aligned} \quad (6)$$

The first inequality in (6) follows from the AM (arithmetic mean)-GM (geometric mean) inequality (see e.g., [23]), and the second from the ordering of the samples as in (2) (with  $\mathbf{A} = \tilde{\mathbf{R}}$ ). The AM-GM inequality also implies that equality in (4) holds if and only if  $\hat{\mathbf{R}} = \tilde{\mathbf{R}}$ .

The MCD algorithm uses the above result (assuming that  $N_u$ , or  $N_c = N - N_u$ , is given) to minimize the determinant of the covariance matrix (and hence the volume of the uncertainty hyperellipsoid estimated from  $N_u$  samples). Its main steps can be summarized as follows:

- 0) Initialization: choose an initial estimate  $\hat{\mathbf{R}}$  of  $\mathbf{R}$  see, e.g., [1], [14] and also Step 0 of TEST in Section IV.
- 1) Using the latest estimate of  $\mathbf{R}$ , sort the samples as in (2) with  $\mathbf{A} = \hat{\mathbf{R}}$  and obtain a new estimate of  $\mathbf{R}$  from (5).
- 2) Iteration: iterate step 1 until  $\hat{\mathbf{R}}$  remains the same in two consecutive iterations.

The pseudocode of the MCD is summarized in Algorithm 1.

The MCD algorithm decreases  $|\hat{\mathbf{R}}|$  at each iteration and converges in a finite (usually quite small) number of iterations [1]. Because in general the MCD iterative algorithm has multiple stationary points it can be advisable to run it with several initializations, obtain several estimated covariance matrices and then choose the one that has the smallest determinant. For details on multiple random initializations or deterministic initializations of a special type see, e.g., [1], [2].

## IV. CONDITIONAL LIKELIHOOD INTERPRETATION OF MCD

Let  $\{b_t\}_{t=1}^N$  be binary latent variables defined as shown below:

$$b_t = \begin{cases} 1 & \text{if } t \in U \\ 0 & \text{if } t \in C \end{cases} \quad (7)$$

For given  $\{b_t\}$  the conditional likelihood function of the uncorrupted samples has the following expression:

$$p(\{\mathbf{y}_t\}_{t \in U} | \mathbf{b}; \mathbf{R}) = \prod_{t=1}^N [p(\mathbf{y}_t; \mathbf{R})]^{b_t} \quad (8)$$

Maximization of (8) with respect to (w.r.t.)  $\mathbf{R}$  and  $\mathbf{b} = \{b_t\}_{t=1}^N$  is equivalent to minimizing the negative log-likelihood function:

$$-2 \ln p(\{\mathbf{y}_t\}_{t \in U} | \mathbf{b}; \mathbf{R}) = \sum_{t=1}^N b_t [n \ln(2\pi) + \ln |\mathbf{R}| + \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t] \quad (9)$$

where

$$\sum_{t=1}^N b_t = N_u = N - N_c.$$

### A. CYCLIC ALGORITHM

Assuming that  $N_u$  is given, we can use the following cyclic algorithm (see, e.g., [24] for a concise discussion on this type of algorithm) to minimize (9):

- 0) Initialization: choose an initial estimate  $\hat{\mathbf{R}}$  of  $\mathbf{R}$ .
- 1) With the samples ordered as in (2) with  $\mathbf{A} = \hat{\mathbf{R}}$ , observe that the minimization of (9) w.r.t.  $\mathbf{b}$  (for given  $\mathbf{R} = \hat{\mathbf{R}}$  and fixed  $N_c$ ) yields:

$$\hat{b}_t = \begin{cases} 0 & \text{for } t = 1, \dots, N_c \\ 1 & \text{for } t = N_c + 1, \dots, N \end{cases} \quad (10)$$

Furthermore, the problem of minimizing (9) w.r.t.  $\mathbf{R}$  (for given  $\mathbf{b} = \hat{\mathbf{b}}$ ) has the following solution:

$$\hat{\mathbf{R}} = \frac{1}{N_u} \sum_{t=N_c+1}^N \mathbf{y}_t \mathbf{y}_t^T = \frac{1}{N_u} \sum_{t=1}^{N_u} \mathbf{y}_{N+1-t} \mathbf{y}_{N+1-t}^T \quad (11)$$

(assuming that the above matrix is nonsingular).

- 2) Iteration: iterate step 1 until  $\hat{\mathbf{R}}$  does not change anymore.

Comparing (11) and (5) shows that the above cyclic algorithm is identical to MCD. The maximum conditional likelihood interpretation of MCD, which follows from this observation, strengthens the statistical basis of the MCD method.

### B. MAJORIZATION-MINIMIZATION ALGORITHM

Interestingly we can also arrive at MCD in another way. Using the fact that  $N_u$  is given, we can rewrite (9) as:

$$\text{const.} + N_u \ln |\mathbf{R}| + \text{Tr} \left( \mathbf{R}^{-1} \sum_{t=1}^N b_t \mathbf{y}_t \mathbf{y}_t^T \right) \quad (12)$$

The minimization of (12) w.r.t.  $\mathbf{R}$ , for fixed  $\mathbf{b}$ , yields the following expression for the minimizer (as a function of  $\mathbf{b}$ ):

$$\mathbf{R}(\mathbf{b}) = \frac{1}{N_u} \sum_{t=1}^N b_t \mathbf{y}_t \mathbf{y}_t^T \quad (13)$$

Inserting (13) in (12) yields the concentrated conditional likelihood that is to be minimized w.r.t. to  $\mathbf{b}$  (omitting the constants):

$$\min_{\mathbf{b}} \ln |\mathbf{R}(\mathbf{b})| \quad (14)$$

This is exactly the type of problem (i.e. covariance determinant minimization) solved by MCD using the result in (4). However here we will use a different approach to solve (14). First we note that the function in (14) can be upper bounded as shown below:

$$\ln |\mathbf{R}| \leq \ln |\hat{\mathbf{R}}| + \text{Tr} [\hat{\mathbf{R}}^{-1} (\mathbf{R} - \hat{\mathbf{R}})] \quad (15)$$

where  $\hat{\mathbf{R}}$  is any positive definite matrix (this result is known, see e.g. [25]; however to make the paper as self-contained as possible we include a simple proof of (15) in Appendix A). Because equality holds in (15) for  $\mathbf{R} = \hat{\mathbf{R}}$ , the right hand side of (15) is what is called a majorizer of  $\ln |\mathbf{R}|$ . It follows from this observation and the properties of the majorization-minimization approach (see [24], [25]) that we can monotonically decrease  $\ln |\mathbf{R}(\mathbf{b})|$  (or, equivalently,  $|\mathbf{R}(\mathbf{b})|$ , as the  $\ln(\cdot)$  is a monotone function) by minimizing the majorizing function in (15), that is (after omitting the additive constants):

$$\min_{\mathbf{b}} \left\{ \text{Tr} [\hat{\mathbf{R}}^{-1} \mathbf{R}(\mathbf{b})] = \frac{1}{N_u} \sum_{t=1}^N b_t \mathbf{y}_t^T \hat{\mathbf{R}}^{-1} \mathbf{y}_t \right\} \quad (16)$$

Clearly the minimizer  $\hat{\mathbf{b}}$  of (16) is the same as (10). Therefore the majorization-minimization approach also leads to the MCD algorithm.

The fact that MCD can be obtained in different ways, as shown above, provides interesting interpretations of this method as a maximum likelihood procedure. However MCD has a problem in that in order to perform at its maximum potential it requires the knowledge of  $N_u$ . In the next section we will present a solution to this type of problem.

### V. MULTIPLE HYPOTHESIS TESTING METHOD : TEST

The quadratic forms  $\{\mathbf{y}_t^T \hat{\mathbf{R}}^{-1} \mathbf{y}_t\}$  (for given  $\hat{\mathbf{R}}$ ) are the sufficient statistics in the MCD decision/estimation process. Under the null hypothesis  $H_{ot}$  that  $\mathbf{y}_t \in U$ , and assuming that  $\hat{\mathbf{R}}$  is an accurate estimate of the true covariance matrix, the (approximate) distribution of  $\mathbf{y}_t^T \hat{\mathbf{R}}^{-1} \mathbf{y}_t$  is chi-square with  $n$  degrees of freedom (see, e.g., [1], [26]):

$$T_t \triangleq \mathbf{y}_t^T \hat{\mathbf{R}}^{-1} \mathbf{y}_t | H_{ot} \sim \chi^2(n) \quad (17)$$

The consequence of this observation is that we can use hypothesis testing to estimate  $U$  (and implicitly  $N_u$ ). Individual testing of each  $T_t$  may not be advisable for the values of  $N$  encountered in applications: indeed this type of testing can achieve a small  $P_{fa}$  = probability of false alarm (i.e. the probability of classifying uncorrupted samples as outliers) only at the expense of a large  $P_{miss}$  = probability of miss (i.e. the probability of classifying outliers as uncorrupted data). Multiple hypothesis testing appears to be preferable in such cases. One of the most prominent methods for multiple hypothesis

**Algorithm 2:** TEST.

---

**Input:**  $\{\mathbf{y}_t\}_{t=1}^N, \alpha$ .  
**Output:**  $\hat{\mathbf{R}}$ .

- 1 Initialize  $\hat{\mathbf{R}}$ .
- 2 **repeat**
- 3     Compute the test statistics  $\{T_t\}$  in (17)
- 4     Obtain  $\hat{U}$  using (20)
- 5     Compute  $\hat{\mathbf{R}}$  as in (21)
- 6 **until** convergence

---

testing is the FDR, see e.g. [19], [20]. To briefly describe it let:

$$p_t = \frac{\alpha t}{N} \quad t = 1, \dots, N \quad (18)$$

where  $\alpha$  is a user parameter that (implicitly) controls the  $P_{fa}$  (see [19], [20] for details on this aspect). Also, let  $\eta_t$  denote the quantile corresponding to  $p_t$ :

$$\text{prob}(T_t \geq \eta_t | H_{ot}) = p_t \quad (19)$$

Following these preliminaries we can now summarize the main steps of TEST:

- 0) Initialization: a) Let  $\bar{N}_c$  be a (loose) upper bound on  $N_c$ . For instance, in many applications  $N_c \leq 0.5N$  and in such a case we can conservatively use  $\bar{N}_c = 0.75N$  provided that  $N - \bar{N}_c \gg n$ . Clearly the selection of  $\bar{N}_c$  should be made with the following tradeoff in mind:  $\bar{N}_c$  should be large enough to eliminate as many outliers as possible but not too large such that sufficiently many samples are left for the initial estimation of  $\mathbf{R}$  (see b) below).  
 b) Compute the sample covariance matrix  $\hat{\mathbf{R}}$  of  $\{\mathbf{y}_t\}_{t=1}^N$  (e.g. use (5) with  $N_u = N$ ), sort the data samples as in (2) with  $\mathbf{A} = \hat{\mathbf{R}}$ , and obtain an enhanced initial estimate of  $\mathbf{R}$  from (5) with  $N_u = N - \bar{N}_c$ .
- 1) Using the most recent estimate  $\hat{\mathbf{R}}$  obtain an estimate  $\hat{N}_c$  of  $N_c$  as the largest value of  $t$  for which the ordered test statistics are consecutively larger than the thresholds:

$$\begin{cases} T_t \geq \eta_t & (t = 1, \dots, \hat{N}_c) \\ T_{\hat{N}_c+1} < \eta_{\hat{N}_c+1} \end{cases} \quad (20)$$

Reject the corresponding null hypotheses  $H_{ot}$  for  $t = 1, \dots, \hat{N}_c$ . Besides an estimate of  $N_u$  viz.  $\hat{N}_u = N - \hat{N}_c$ , the above test also provides an estimate  $\hat{U}$  of  $U$ , which we use to re-estimate  $\mathbf{R}$ :

$$\hat{\mathbf{R}} = \frac{1}{\hat{N}_u} \sum_{t \in \hat{U}} \mathbf{y}_t \mathbf{y}_t^T \quad (21)$$

- 2) Iteration: Iterate step 1 until convergence.

The pseudocode of TEST is summarized in Algorithm 2.

It is our empirical experience that the above algorithm, like MCD, converges in a small number of iterations (usually less than 10) and, once again like MCD, it can be run with multiple initializations. On the other hand, unlike the case of MCD

whose convergence follows from any of the three derivations presented in Sections II and III, the convergence of TEST does not follow from the previous discussion and remains to be analyzed, which we will do in the rest of this section.

To prove that TEST converges we will consider the following function:

$$\ln |\mathbf{R}| + \frac{1}{N_u} \sum_{t=1}^N b_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t + \sum_{t=1}^{N_c} \frac{1}{N-t} \eta_t \quad (22)$$

We will show that TEST (approximately) minimizes (22) cyclically w.r.t.  $\mathbf{R}$  and  $(\mathbf{b}, N_c)$ :

- \* For given  $\hat{\mathbf{b}}$  and  $\hat{N}_c$  the minimization of (22) w.r.t.  $\mathbf{R}$  yields

$$\hat{\mathbf{R}} = \frac{1}{\hat{N}_u} \sum_{t=1}^N \hat{b}_t \mathbf{y}_t \mathbf{y}_t^T = \frac{1}{\hat{N}_u} \sum_{t \in \hat{U}} \mathbf{y}_t \mathbf{y}_t^T \quad (23)$$

where  $\hat{N}_u = N - \hat{N}_c$ . Observe that (23) coincides with (21).

- \* For given  $\mathbf{R} = \hat{\mathbf{R}}$  we have to minimize the following function w.r.t.  $(\mathbf{b}, N_c)$ :

$$\frac{1}{N - N_c} \sum_{t=1}^N b_t T_t + \sum_{t=1}^{N_c} \frac{1}{N-t} \eta_t \quad (24)$$

where  $\{T_t\}$  are the ordered test statistics in (17). For fixed  $N_c$  the minimizer of (24) w.r.t.  $\{b_t\}$  is given (as a function of  $N_c$ ) by:

$$\hat{b}_t = \begin{cases} 0 & \text{for } t = 1, \dots, N_c \\ 1 & \text{for } t = N_c + 1, \dots, N \end{cases} \quad (25)$$

Inserting (25) in (24) we get the following concentrated function that is to be minimized w.r.t.  $N_c$ :

$$C_{N_c} \triangleq \frac{1}{N - N_c} \sum_{t=N_c+1}^N T_t + \sum_{t=1}^{N_c} \frac{1}{N-t} \eta_t \quad (26)$$

A straightforward calculation shows that:

$$\begin{aligned} C_{N_c-1} - C_{N_c} &= \frac{1}{N - N_c + 1} \sum_{t=N_c}^N T_t \\ &\quad - \frac{1}{N - N_c} \sum_{t=N_c+1}^N T_t - \frac{1}{N - N_c} \eta_{N_c} \\ &= \frac{1}{N - N_c} \left[ \left(1 - \frac{1}{N_u + 1}\right) \sum_{t=N_c}^N T_t \right. \\ &\quad \left. - \sum_{t=N_c+1}^N T_t - \eta_{N_c} \right] \\ &\approx \frac{1}{N_u} (T_{N_c} - \eta_{N_c}) \end{aligned} \quad (27)$$

where the first equality follows from the definition in (26), the second equality from a simple calculation, and the approximate equality from the assumption that  $N_u \gg N_c$ . Consequently, the two inequalities below are equivalent (for  $N_u \gg N_c$ ):

$$C_{N_c-1} \geq C_{N_c} \Leftrightarrow T_{N_c} \geq \eta_{N_c} \quad (28)$$

Therefore the TEST estimate of  $N_c$  coincides with the (smallest) minimizer of (26), and the TEST estimates of  $\mathbf{R}$  and  $\mathbf{b}$  are identical to (23) and, respectively, (25). This implies that TEST is an (approximate) cyclic minimization algorithm for (22) and therefore it monotonically decreases this function at each iteration. This observation, combined with the fact that the function in (22) is bounded from below, proves that TEST is a convergent algorithm (for  $N_u \gg N_c$ , which is usually the case in practical applications). For another interpretation of TEST as an (approximate) cyclic minimization algorithm (for  $N_u \gg N_c$ ), which leads to the same conclusion as above, see Appendix B.

Interestingly the above result, which shows that TEST is a cyclic minimizer of (22), is also relevant to MCD. To see this, observe that minimization of (22) w.r.t.  $\mathbf{R}$ , for fixed  $\mathbf{b}$  and  $N_c$ , yields

$$\hat{\mathbf{R}}(\mathbf{b}, N_c) = \frac{1}{N_u} \sum_{t=1}^N b_t \mathbf{y}_t \mathbf{y}_t^T \quad (29)$$

and the following concentrated function

$$\ln |\hat{\mathbf{R}}(\mathbf{b}, N_c)| + \sum_{t=1}^{N_c} \frac{1}{N-t} \eta_t \quad (30)$$

which is to be minimized w.r.t.  $\mathbf{b}$  and  $N_c$ . We can use the MCD algorithm to minimize (30) w.r.t.  $\mathbf{b}$  for each value of  $N_c$  in a pre-specified interval (such as  $[1, \bar{N}_c]$ ) and select the pair  $(\mathbf{b}, N_c)$  that gives the minimum value of (30). The minimization of the penalized MCD criterion in (30) should theoretically lead to the same estimates of  $\mathbf{b}$  and  $N_c$  as TEST's. While the above idea provides MCD with a possible approach for estimation of  $N_c$ , we should note that from a computational viewpoint minimizing (30) requires running MCD for many values of  $N_c$  and therefore can be significantly more costly than cyclically minimizing (22) using TEST.

## VI. PENALIZED LIKELIHOOD METHOD: LIKE

In order to write the likelihood of the full data string  $\{\mathbf{y}_t\}_{t=1}^N$  we need to make some assumption on the distributions of the outliers. We consider shift outliers, the distributions of which are:

$$\mathbf{y}_t \sim \mathcal{N}(\mathbf{c}_t, \mathbf{R}) \quad \text{for } t \in C \quad (31)$$

where  $\{\mathbf{c}_t\}$  denote unknown arbitrary means. Because these means are allowed to vary with the sample index, the outlier model in (31) is reasonably general. Using (31) the negative

log-likelihood of  $\{\mathbf{y}_t\}_{t=1}^N$  can be written as follows:

$$\begin{aligned} & \frac{nN}{2} \ln(2\pi) + \frac{N}{2} \ln |\mathbf{R}| + \frac{1}{2} \sum_{t \in U} \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t \\ & + \frac{1}{2} \sum_{t \in C} [\mathbf{y}_t - \mathbf{c}_t]^T \mathbf{R}^{-1} [\mathbf{y}_t - \mathbf{c}_t] \end{aligned} \quad (32)$$

Minimization of (32) w.r.t.  $\{\mathbf{c}_t\}$  yields the following concentrated function (after multiplication by 2 and omitting an additive constant):

$$N \ln |\mathbf{R}| + \sum_{t \in U} \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t \quad (33)$$

Notice the difference between (33) and the conditional likelihood in (12) where the factor in front of  $\ln |\mathbf{R}|$  is  $N_u$  instead of  $N$  as in (33). This seemingly minor difference is quite important when we also want to estimate  $N_u$  in addition to  $U$  and  $\mathbf{R}$ , as we want to do here. Indeed a scaling of the data in (33) adds a constant term that is independent of  $N_u$ , whereas this is not true for (12).

We will use GIC to estimate  $N_c$ . This model selection criterion adds a penalty, which increases with the number of estimated parameters, to the concentrated negative-likelihood function and estimates  $\mathbf{R}$ ,  $U$  and  $N_c$  as the solutions to the following penalized negative-likelihood minimization problem:

$$\min_{\mathbf{R}, U, N_c} N \ln |\mathbf{R}| + \sum_{t \in U} \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t + \eta \sum_{t=1}^{N_c} \frac{N}{N-t} \quad (34)$$

where the penalty factor  $\eta$  may depend on  $n$  and possibly on  $N$  too (note that  $\sum_{t=1}^{N_c} \frac{N}{N-t} \approx N_c$  for  $N \gg N_c$ ). Before discussing the choice of the penalty factor  $\eta$  in (34) we comment on a number of aspects regarding the above minimization problem.

First observe that (34) is quite similar to (22): the main difference is that the penalties in (34) and (22) may be different but, for the purpose of solving (34), that is an insignificant difference indeed. Therefore the same cyclic minimization algorithm used for (22) can be employed for (34). The algorithm outputs estimates of  $\mathbf{R}$ ,  $U$  and  $N_c$ . The latter two need no correction but the estimate  $\hat{\mathbf{R}}$  of  $\mathbf{R}$  is biased (in much the same way as the maximum-likelihood estimate of a covariance matrix is biased when one also estimates the mean). Consequently we will use the unbiased estimate  $\hat{\mathbf{R}}_{N_u}^{\frac{N}{N_u}}$  in lieu of  $\hat{\mathbf{R}}$ .

Next we remark on the fact that, similarly to what we said at the end of Section IV, the function in (34) could be minimized w.r.t.  $\mathbf{R}$  (for fixed  $N_c$  and  $U$ ) which would yield a penalized MCD criterion. However the minimization of the said criterion would require a repeated use of the MCD algorithm for many values of  $N_c$ , which would be significantly less efficient than using the cyclic algorithm mentioned above.

The method based on (34) has been introduced as a penalized negative-likelihood minimization method but, as we

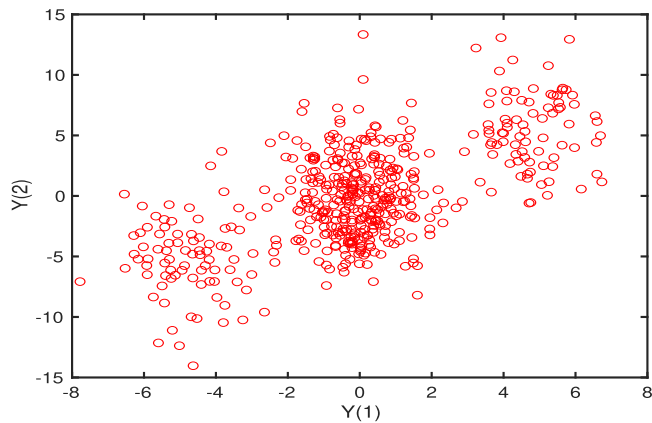


FIGURE 1. Scatter plot of two dimensional data with outlier clouds at  $(-5, -5)$  and  $(5, 5)$ .  $n = 2$ ,  $N = 500$ , and  $N_c = 0.5 N$ .

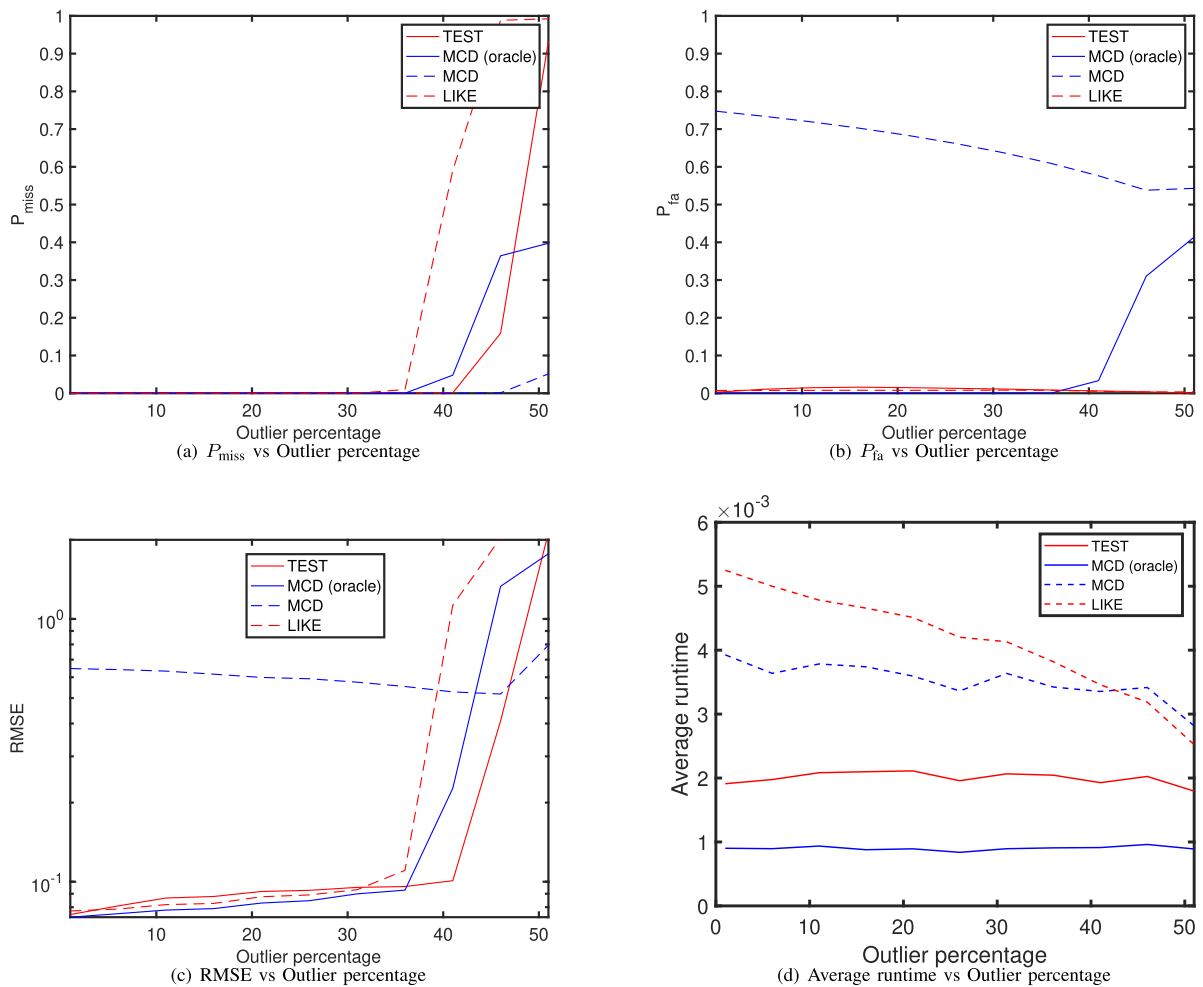


FIGURE 2.  $N = 500$ ,  $n = 5$ , and two outlier clouds with means  $\pm 10$ .

show below, it can be reformulated as a multi-hypothesis testing procedure. This is the opposite of what we have done in Section IV where TEST was introduced as a multi-hypothesis testing method and was later shown to be also a penalized-function minimization procedure.

To interpret (34) as a multi-hypothesis testing approach, observe that the minimization w.r.t.  $U$ , for given  $\mathbf{R} = \hat{\mathbf{R}}$  and

fixed  $N_c$ , yields the following concentrated function:

$$C_{N_c} \triangleq \sum_{t=N_c+1}^N T_t + \eta \sum_{t=1}^{N_c} \frac{N}{N-t} \tag{35}$$

(we keep the same notation as in Section IV for the above criterion due to its similarity to (26)). Next we note that,

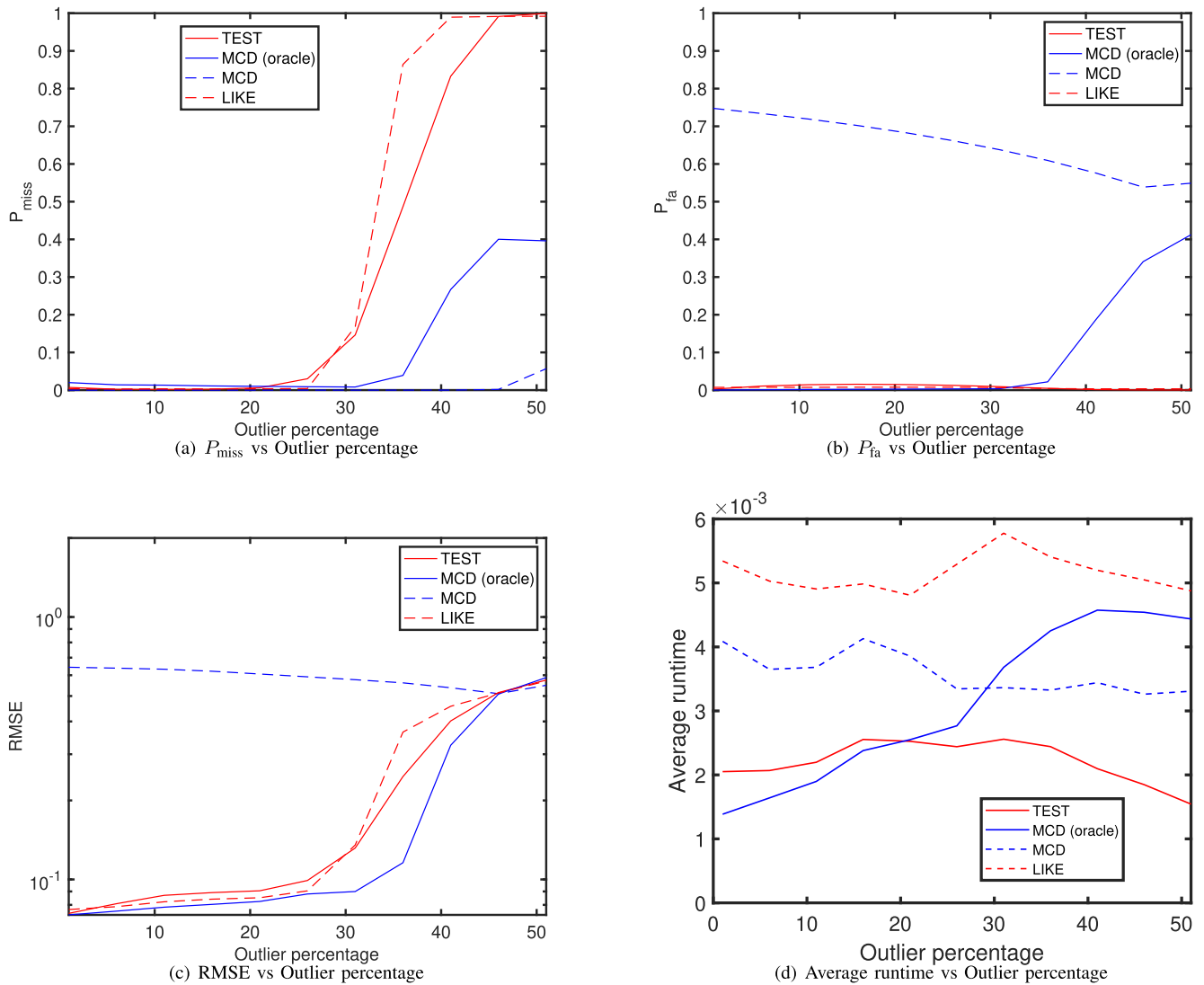


FIGURE 3.  $N = 500$ ,  $n = 5$  and two outlier clouds with means  $\pm 5$ .

similarly to (28), it follows from (35) that:

$$C_{N_c-1} - C_{N_c} = T_{N_c} - \frac{N}{N_u} \eta \quad (36)$$

and therefore

$$C_{N_c-1} \geq C_{N_c} \Leftrightarrow \frac{N_u}{N} T_{N_c} \geq \eta \quad (37)$$

The conclusion is that similar to TEST (34) can also be interpreted as a multi-hypothesis testing method (as expected, using the corrected estimate  $\frac{N}{N_u} \hat{\mathbf{R}}$  of  $\mathbf{R}$ ).

Finally we consider the important problem of choosing  $\eta$  in (34). Because we are dealing with a high-dimensional scenario, the first GIC that comes to mind is the extended Bayesian information criterion (EBIC) whose penalty has the following general expression [22]:

$$n_p \ln n_d + 2 \ln \binom{n_m}{n_c} \quad (38)$$

where  $\binom{n_m}{n_c}$  denotes the binomial coefficient (i.e.  $n_m$  choose  $n_c$ ),  $n_p$  is the number of free parameters in the current model,  $n_d$  the number of data points,  $n_m$  the maximum possible model size, and  $n_c$  the size of the current model (the size of a model is the number of its free parameters, or parameter vectors whenever an entire parameter vector is either zero or free as in (32)). In our case:

$$n_p = nN_c, \quad n_d = nN, \quad n_m = N, \quad n_c = N_c. \quad (39)$$

Note that there are  $n(n+1)/2$  free parameters in  $\mathbf{R}$ , but their number is not included in  $n_p$  because it does not depend on  $N_c$ . Inserting (39) in (38) and using the fact that  $\ln \binom{n_m}{n_c} \approx n_c \ln(n_m)$  (for  $n_m \gg n_c$ ) and  $n \ln(nN) = n \ln(n) + n \ln(N) \approx n \ln(N)$  (for  $N \gg n$ ), the EBIC penalty becomes:  $N_c(n+2) \ln(N)$ .

In our experience EBIC tends to underestimate  $N_c$  (especially for  $n \gg 1$ ). To understand this issue of EBIC consider the hypothesis testing interpretation of (34) (see (37)). Under  $H_{\text{ot}}$  it follows from the properties of  $\chi^2$  distribution that the

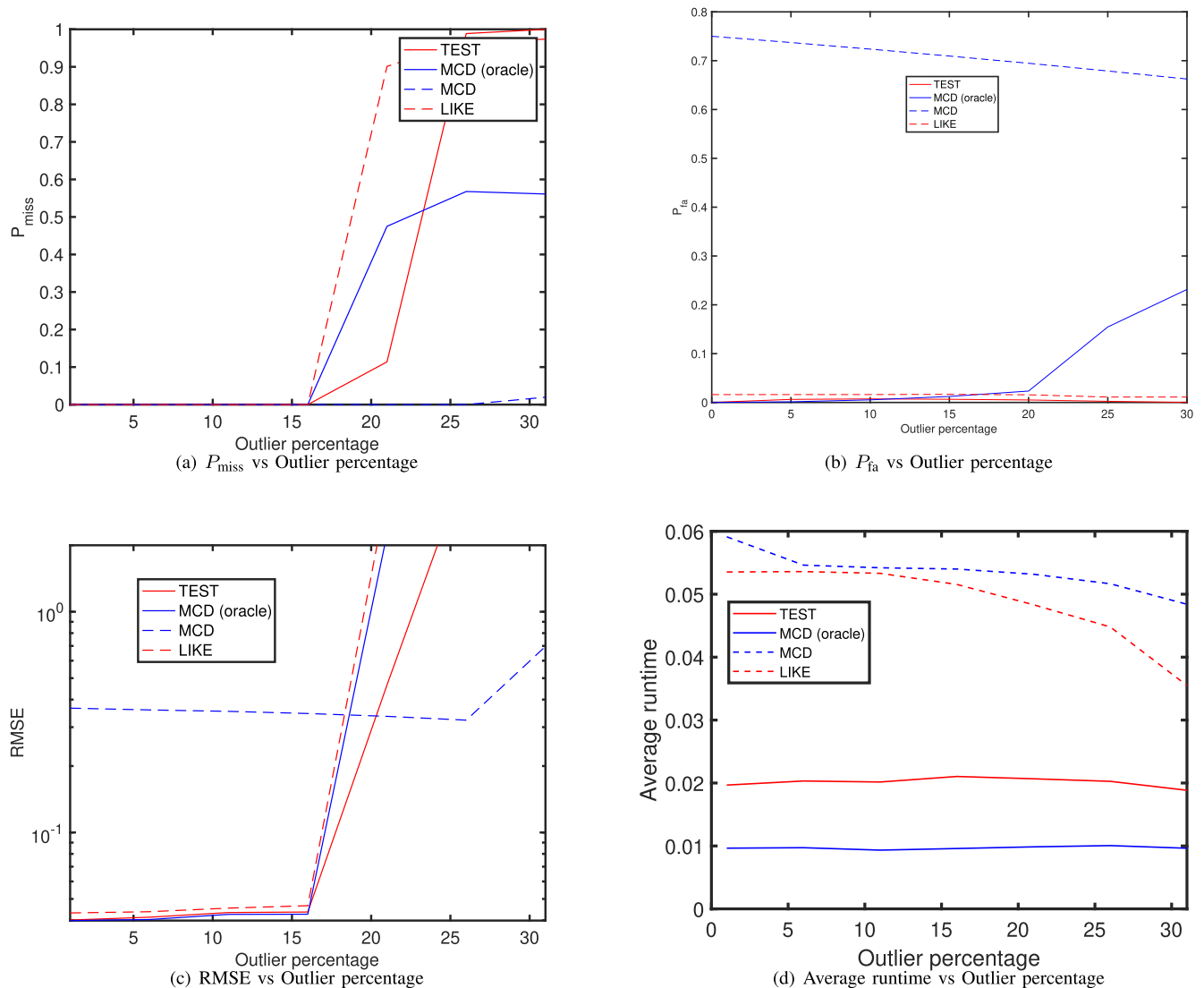


FIGURE 4.  $N = 2000$ ,  $n = 20$ , and two outlier clouds with means  $\pm 10$ .

standard deviation of  $(\frac{N_u}{N}T_t - n)$  is proportional to  $\sqrt{n}$ . However the threshold associated with EBIC is such that  $\eta - n = n(\ln(N) - 1) + \text{const.}$  grows faster than  $\sqrt{n}$  as  $n$  increases. This observation implies that, for large values of  $n$ ,  $\eta$  can take values that are much larger than those recommended by standard statistical practice. Consequently the corresponding  $P_{\text{fa}}$  of (34) with EBIC can be quite small but the  $P_{\text{miss}}$  can be much larger than what one would want. Note that methods with small  $P_{\text{fa}}$  ( $P_{\text{miss}}$ ) but large  $P_{\text{miss}}$  ( $P_{\text{fa}}$ ) do not perform well. Striking a balance between  $P_{\text{fa}}$  and  $P_{\text{miss}}$  and keeping both small appears to be a requirement for a method to achieve a good performance.

To fix the above issue of EBIC we have to modify the penalty factor  $\eta$ . To do so we make use of a property of the  $\chi^2$  distribution [27]:

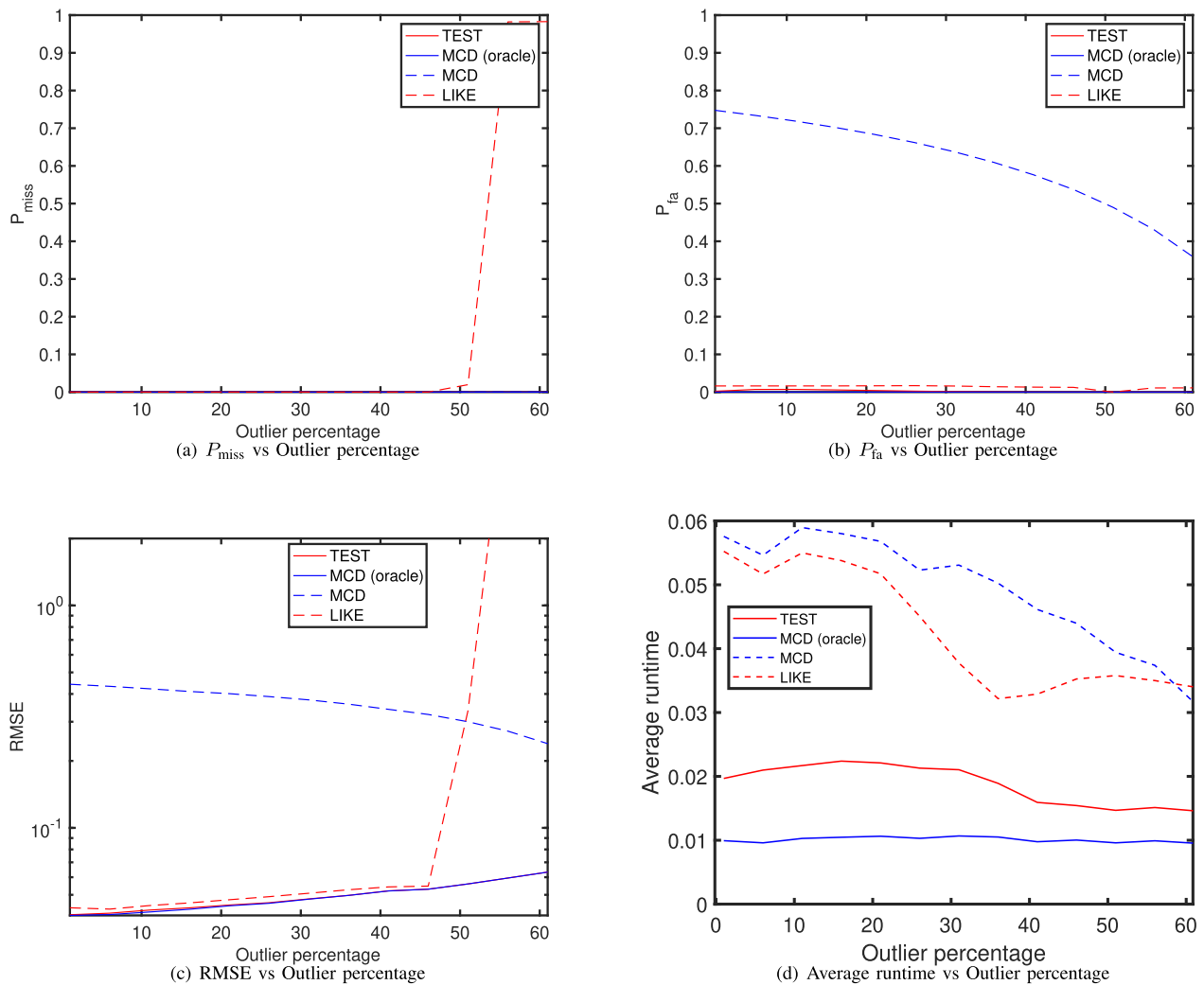
$$\text{Prob} \left( \frac{N_u}{N}T_t \geq n + \sqrt{2n\rho} + 2\rho | H_{\text{ot}} \right) \leq e^{-\rho} \quad (40)$$

Therefore to achieve a  $P_{\text{fa}} \leq e^{-\rho}$  we can choose

$$\eta = n + \sqrt{2n\rho} + 2\rho \quad (41)$$

(observe that  $\eta - n$  is proportional to  $\sqrt{n}$ , as expected). The penalized likelihood method (34) with the penalty factor (or threshold) in (41) will be called LIKE (penalized likelihood method). Note that, unlike TEST's thresholds, the threshold used by LIKE is constant. The performance of LIKE can therefore be expected to be inferior to that of TEST.<sup>1</sup> Nevertheless our experience with these two methods is that they have relatively similar performances in a reasonable number of cases (see the next section for examples). The pseudocode of LIKES is summarized in Algorithm 3.

<sup>1</sup>However note that in Appendix B we briefly describe an enhanced version of LIKE that uses the same FDR thresholds as TEST.



**FIGURE 5.**  $N = 2000$ ,  $n = 20$ , and two outlier clouds with means  $\pm 10$ . The four methods are initialized using step 0 of TEST but with  $\mathbf{I}$  instead of the sample covariance matrix of  $\{\mathbf{y}_t\}_{t=1}^N$  (which was used in Figs. 2–4).

---

**Algorithm 3: LIKE.**

---

**Input:**  $\{\mathbf{y}_t\}_{t=1}^N$ ,  $\rho$ .

**Output:**  $\hat{\mathbf{R}}$ .

- 1 Initialize  $\hat{\mathbf{R}}$ .
  - 2 **repeat**
  - 3     Compute the test statistics  $\{T_t\}$  in (17)
  - 4     Obtain  $\hat{U}$  by finding the global minimum of (35)
  - 5     Compute  $\hat{\mathbf{R}} = \frac{1}{N} \sum_{t \in \hat{U}} \mathbf{y}_t \mathbf{y}_t^T$
  - 6 **until** convergence
  - 7  $\hat{\mathbf{R}} \leftarrow \frac{N}{N_u} \hat{\mathbf{R}}$
- 

Finally we comment on the selection of the user parameters  $\rho$  of LIKE and  $\alpha$  of TEST. While a precise guideline for choosing  $\alpha$  and  $\rho$  is difficult to give in general, our experience is that the performance of TEST or LIKE is relatively insensitive to the variation of these parameters within certain reasonable limits. For example, while we will use TEST with  $\alpha = 0.2$  in the next section we have also tried other values of  $\alpha$

and noticed that the performance did not change significantly when  $\alpha$  was decreased to 0.05 or even 0.01.

**VII. NUMERICAL PERFORMANCE STUDY**

We will compare the performance of the following methods

- \* Oracle MCD (with  $N_c = \text{true value}$ )
- \* MCD (with  $N_c = 0.75N$ )
- \* TEST (with  $\alpha = 0.2$ ) and LIKE (with  $\rho = 3$ ) both with  $\tilde{N}_c = 0.75N$  in the initialization step

The following metrics will be used in this comparison exercise:

- \*  $P_{fa}$
- \*  $P_{miss}$
- \*  $\text{NRMSE} = \text{average} \left( \frac{\|\hat{\mathbf{R}} - \mathbf{R}\|}{\|\mathbf{R}\|} \right)$

where  $\|\cdot\|$  denotes the Frobenius matrix norm. The above metrics are computed using  $10^3$  Monte-Carlo simulations. Note that, unless otherwise indicated, all four algorithms are initialized using the estimated covariance matrix provided by step 0 of TEST. Also note that, we will not use the corrective

step suggested in [1] for modifying the MCD estimate. The reason for not considering the said post-processing step is that we want to study the accuracy of the covariance matrix estimate provided to it by the main MCD step whose performance is important in itself. The specifications of the data generation are as follows:

- \*  $N = 500$  or  $2000$
- \*  $n = 5$  or  $20$
- \*  $\mathbf{R}$  = a randomly generated diagonal covariance matrix with condition number = 100.
- \*  $N_c$  = varied from  $0.01N$  up to  $0.3N$  or  $0.5N$  or even  $0.6N$ .
- \* The outliers are generated adding  $10\mathbf{u}$  or  $5\mathbf{u}$  to  $N_c/2$  data samples and subtracting  $10\mathbf{u}$  or  $5\mathbf{u}$  from  $N_c/2$  samples, where  $\mathbf{u} = [1, \dots, 1]^T$ .

For a visual illustration of the data scatter as well as the separation (or lack thereof) between uncorrupted and corrupted samples, Fig. 1 shows the uncorrupted data cloud (centered at  $(0,0)$ ) and the two outlier clouds (at  $(-5, -5)$  and  $(5,5)$ ) for  $N = 500$ ,  $N_c = 0.5N$  and  $n = 2$ .

Figs. 2 and 3 show the metrics for  $N = 500$  and  $n = 5$ . As one can see from these figures the performances of TEST and LIKE are relatively similar to one another as well as to the performance of the oracle MCD and much better than the performance of MCD with a loose upper bound on  $N_c$ .

Fig. 4 considers the same scenario as in Fig. 2, but now with  $n = 20$  and  $N = 2000$ . Once again TEST and oracle MCD have similar and quite good performances and the performance of LIKE is relatively close to that of TEST.

In the last example of this paper we will illustrate the fact that, as expected, the initialization matters. To do so we consider the scenario in Fig. 4 for which the performance of the four methods was only mildly satisfactory, but now we initialize the methods using  $\mathbf{I}$  (in step 0 of TEST) in lieu of the sample covariance matrix of  $\{\mathbf{y}_t\}_{t=1}^N$  (using the latter matrix can lead to the masking of some outliers, see Appendix C for an illustration of this effect). The results, shown in Fig. 5, are much better than those displayed in Fig. 4. While using  $\mathbf{I}$  to initialize the four methods might not work well in all cases (such as in cases in which the distribution of the uncorrupted samples is very skewed) the point of this example is that the initialization is important (for details on this aspect as well as suggestions for both deterministic and random initializations we refer the reader to [1], [2]).

Finally note that Figs. 2–5 also contain a fourth box that shows the corresponding average run times for one data realization. All four methods are reasonably fast (their run times are on the order of  $10^{-3}$  sec. for  $\{N = 500, n = 5\}$  and  $10^{-2}$  sec. for  $\{N = 2000, n = 20\}$ ) and the differences between them are relatively small.

## VIII. CONCLUSION

We have considered a maximum conditional likelihood problem and showed that two algorithms (viz. a cyclic procedure and a majorization-minimization technique) for solving this

problem coincide with the MCD algorithm. The derivation of MCD in a maximum likelihood context provides a new statistical interpretation of this method.

We used the FDR to formulate a multiple hypothesis testing procedure called TEST for estimating the number and locations of the outliers as well as the data covariance matrix. To establish the convergence of the TEST algorithm we showed that it can be viewed as a cyclic minimization procedure of a penalized objective function (for  $N_u \gg N_c$ ). Using the GIC we derived another algorithm called LIKE that cyclically minimizes a penalized negative-likelihood function, and which can also be interpreted as a hypothesis testing method.

As indicated above both TEST and LIKE can be viewed as penalized function minimization algorithms as well as multi-hypothesis testing procedures. The latter interpretation of these methods appears to be preferable when their user parameters that determine the  $P_{fa}$  (viz.  $\alpha$  in (18) and, respectively,  $\rho$  in (40)) have to be selected. The numerical examples included in the paper confirmed the good performance of TEST and LIKE : these methods performed as well as the oracle MCD and in general much better than MCD with a loose upperbound on  $N_c$ .

## APPENDIX A PROOF OF (15)

Using the AM-GM inequality once again (see (6)) we have that:

$$|\hat{\mathbf{R}}^{-1}\mathbf{R}|^{1/n} \leq \frac{1}{n}\text{Tr}(\hat{\mathbf{R}}^{-1}\mathbf{R}) \quad (42)$$

or, equivalently,

$$\frac{1}{n}[\ln|\mathbf{R}| - \ln|\hat{\mathbf{R}}|] \leq \ln\left[\frac{1}{n}\text{Tr}(\hat{\mathbf{R}}^{-1}\mathbf{R})\right] \triangleq \ln(x) \quad (43)$$

Because  $\ln(x)$  is a concave function it is majorized by its tangent at any point  $x_0$ :

$$\ln(x) \leq \ln(x_0) + \frac{1}{x_0}(x - x_0) \quad (44)$$

which, for  $x_0 = 1$ , becomes:

$$\ln(x) \leq x - 1 \quad (45)$$

From (43) and (45) it follows that

$$\ln|\mathbf{R}| - \ln|\hat{\mathbf{R}}| \leq \text{Tr}(\hat{\mathbf{R}}^{-1}\mathbf{R}) - n \quad (46)$$

which proves (15).

## APPENDIX B FURTHER INTERPRETATIONS OF TEST AND LIKE IN THE PENALIZED LIKELIHOOD CONTEXT

### B1 TEST

Consider the following penalized form of the negative log-likelihood in (9):

$$nN_u \ln(2\pi) + N_u \ln |\mathbf{R}| + \sum_{t=1}^N b_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t + \underbrace{\left[ \sum_{t=1}^{N_c} \eta_t - nN_u \ln(2\pi) - N_u \ln |\mathbf{R}| \right]}_{\text{Penalty}} \quad (47)$$

where  $\{\eta_t\}$  are the FDR thresholds defined in (19).

1) For given  $\mathbf{R} = \hat{\mathbf{R}}$  and fixed  $N_c$  the minimization of (47) w.r.t.  $\{b_t\}$  reduces to the problem:

$$\min_{\{b_t\}} \sum_{t=1}^N b_t T_t \quad (48)$$

the solution of which is given by (25). The remaining problem of minimizing w.r.t.  $N_c$  is:

$$\min_{N_c} C_{N_c} \triangleq \sum_{t=N_c+1}^N T_t + \sum_{t=1}^{N_c} \eta_t \quad (49)$$

(see (25) for the definition of  $T_t$ ). It can be easily verified that:

$$C_{N_c-1} - C_{N_c} = T_{N_c} - \eta_{N_c} \quad (50)$$

and therefore the (smallest) minimizer of  $C_{N_c}$  coincides with the TEST estimate of  $N_c$  and  $\{b_t\}$ .

2) For given  $\{b_t = \hat{b}_t\}$  minimizing (47) w.r.t.  $\mathbf{R}$ , on the other hand, is not a well defined problem. To see why this is so note that the minimization of (47) w.r.t.  $\mathbf{R}$  reduces to the problem:

$$\min_{\mathbf{R}} \sum_{t=1}^N \hat{b}_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t \quad (51)$$

which does not have a finite solution. It follows that we cannot use an exact cyclic algorithm to minimize (47). However we can use an approximate algorithm in which the update of  $\hat{\mathbf{R}}$  is obtained by minimizing only the first three terms in (47) (therefore considering that the penalty term, evaluated at the current estimate  $\hat{\mathbf{R}}$  of  $\mathbf{R}$ , is constant). The so-obtained update of  $\hat{\mathbf{R}}$  coincides with the TEST estimate in (21).

In the signal processing literature the above type of algorithm is called iterative approximate ML (IAML) or iterative quadratic ML (IQML) when the problem in Step 2 is quadratic (see, e.g., [28] and references therein). Therefore *TEST can be interpreted as an IAML algorithm* for the penalized likelihood in (47). Such an algorithm does not necessarily converge to a minimum of (47) but usually it converges to a point that is close to the global minimum of this function (see [28]).

Next we consider the function in (47) but with a different penalty term:

$$nN_u \ln(2\pi) + N_u \ln |\mathbf{R}| + \sum_{t=1}^N b_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t + \underbrace{\left[ \sum_{t=1}^{N_c} \eta_t + nN_c \ln(2\pi) + N_c \ln |\mathbf{R}| \right]}_{\text{Penalty}} \quad (52)$$

Clearly TEST is an IAML algorithm also for (52). More interestingly, *TEST can be shown to be an approximate cyclic algorithm* for (52) (for  $N \gg N_c$ , or equivalently  $N_u \gg N_c$ ). To see this, first observe that Step 1 for (52) is the same as for (47). However, Step 2 is different. Indeed now for given  $\{b_t = \hat{b}_t\}$  the minimization of (52) w.r.t.  $\mathbf{R}$  reduces to:

$$\min_{\mathbf{R}} \left[ N \ln |\mathbf{R}| + \sum_{t=1}^N \hat{b}_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t \right] \quad (53)$$

The solution to the above problem, which is

$$\hat{\mathbf{R}} = \frac{1}{N} \sum_{t \in \hat{U}} \mathbf{y}_t \mathbf{y}_t^T \quad (54)$$

is not exactly equal to the TEST estimate of  $\mathbf{R}$  (see (21)) but it is close to it for  $N_u \gg N_c$ .

## B2 LIKE

Observe that the function in (52) becomes the penalized log-likelihood used to derive LIKE (see (34)) if we slightly change the first term of the penalty in (52):

$$\text{const.} + N \ln |\mathbf{R}| + \sum_{t=1}^N b_t \mathbf{y}_t^T \mathbf{R}^{-1} \mathbf{y}_t + \sum_{t=1}^{N_c} \frac{N}{N-t} \eta_t \quad (55)$$

The main implication of this observation is that we can use the FDR thresholds  $\{\eta_t\}$  (see (19)) in LIKE, instead of the constant threshold  $\eta$ , which is expected to yield an *enhanced version* of this method.

## APPENDIX C

### ILLUSTRATION OF THE MASKING EFFECT

In this appendix we will show that using the sample covariance matrix  $\hat{\mathbf{R}}$  of  $\{\mathbf{y}_t\}_{t=1}^N$  in the quadratic forms  $\{\mathbf{y}_t^T \hat{\mathbf{R}}^{-1} \mathbf{y}_t\}$  can lead to a masking effect in the sense that a group of outliers can mask another outlier, which would have been detectable if the group did not exist. To do so we consider the scenario in Fig. 2 with two sub-groups of outliers with means  $10\mathbf{u}$  and, respectively,  $-10\mathbf{u}$  where  $\mathbf{u} = [1, \dots, 1]^T$ . To keep the following algebraic calculations as simple as possible we assume that  $\mathbf{R} = \mathbf{I}$ . We also assume that there is an additional outlier  $\mathbf{y}_c = 5\mathbf{u}$ . The sample covariance matrix can then be approximately written as (for  $N \gg 1$ ):

$$\begin{aligned} \hat{\mathbf{R}} &\approx \frac{25}{N} \mathbf{u}\mathbf{u}^T + \frac{N_c}{N} (100\mathbf{u}\mathbf{u}^T + \mathbf{I}) + \frac{N_u}{N} \mathbf{I} \\ &= \frac{25}{N} \mathbf{u}\mathbf{u}^T + \frac{100N_c}{N} \mathbf{u}\mathbf{u}^T + \mathbf{I} \end{aligned} \quad (56)$$

To simplify let us say that  $N/N_c = 4$ . Then (again for  $N \gg 1$ )

$$\hat{\mathbf{R}} \approx 25\mathbf{u}\mathbf{u}^T + \mathbf{I} \quad (57)$$

On the other hand, if the group of outliers was absent then:

$$\hat{\mathbf{R}} \approx \mathbf{I} \quad (58)$$

Let  $\mathbf{y}_u$  denote a generic uncorrupted sample. Then, if the only outlier was  $\mathbf{y}_c$ , we have from (58) that

$$\mathbf{y}_c^T \hat{\mathbf{R}}^{-1} \mathbf{y}_c = 25n \tag{59}$$

and

$$\mathbf{y}_u^T \hat{\mathbf{R}}^{-1} \mathbf{y}_u = \|\mathbf{y}_u\|^2 \sim \chi^2(n) \tag{60}$$

It follows that the inequality

$$\mathbf{y}_c^T \hat{\mathbf{R}}^{-1} \mathbf{y}_c > \mathbf{y}_u^T \hat{\mathbf{R}}^{-1} \mathbf{y}_u \tag{61}$$

holds with a probability close to one (see (40)), and therefore the outlier  $\mathbf{y}_c$  will be detected in this case.

However, in the presence of the  $N_c$  outliers we have:

$$\begin{aligned} \mathbf{y}_c^T (25\mathbf{u}\mathbf{u}^T + \mathbf{I})^{-1} \mathbf{y}_c &= 25\mathbf{u}^T \left[ \mathbf{I} - \frac{25\mathbf{u}\mathbf{u}^T}{1 + 25n} \right] \mathbf{u} \\ &= 25n - \frac{(25)^2 n^2}{1 + 25n} \\ &= \frac{25n}{1 + 25n} \approx 1 \end{aligned} \tag{62}$$

and

$$\begin{aligned} \mathbf{y}_u^T \hat{\mathbf{R}}^{-1} \mathbf{y}_u &= \|\mathbf{y}_u\|^2 - \frac{25(\mathbf{u}^T \mathbf{y}_u)^2}{1 + 25n} \\ &\approx \|\mathbf{y}_u\|^2 - \frac{(\mathbf{u}^T \mathbf{y}_u)^2}{n} \approx \|\mathbf{y}_u\|^2 \end{aligned} \tag{63}$$

where the approximation holds for reasonably large  $n$  (for which, in particular,  $\mathbf{u}^T \mathbf{y}_u \approx 0$ ). Because a  $\chi^2$ -distributed random variable with  $n$  degrees of freedom is larger than 1 with a probability that approaches one (as  $n$  increases) it follows from the above calculation that for nearly all uncorrupted samples,

$$\mathbf{y}_u^T \hat{\mathbf{R}}^{-1} \mathbf{y}_u > \mathbf{y}_c^T \hat{\mathbf{R}}^{-1} \mathbf{y}_c \tag{64}$$

and consequently the outlier  $\mathbf{y}_c$  will not be detected (it has been masked, and disguised as an uncorrupted sample).

**REFERENCES**

[1] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.  
 [2] M. Hubert, M. Debruyne, and P. J. Rousseeuw, "Minimum covariance determinant and extensions," *Wiley Interdiscipl. Rev., Comput. Statist.*, vol. 10, no. 3, 2018, Art. no. 1421.  
 [3] P. J. Rousseeuw, "Least median of squares regression," *J. Amer. Stat. Assoc.*, vol. 79, no. 388, pp. 871–880, 1984.  
 [4] M. Maechler et al., "Robustbase: Basic robust statistics. R package, CRAN," 2022.  
 [5] E. A. Cator and H. P. Lophuaä, "Central limit theorem and influence function for the MCD estimators at general multivariate distributions," *Bernoulli*, vol. 18, no. 2, pp. 520–551, 2012.

[6] R. Butler, P. Davies, and M. Jhun, "Asymptotics for the minimum covariance determinant estimator," *Ann. Statist.*, vol. 21, no. 3, pp. 1385–1400, 1993.  
 [7] P. Filzmoser, R. G. Garrett, and C. Reimann, "Multivariate outlier detection in exploration geochemistry," *Comput. Geosci.*, vol. 31, no. 5, pp. 579–587, 2005.  
 [8] H. Wang, H. Li, J. Fang, and H. Wang, "Robust Gaussian Kalman filter with outlier detection," *IEEE Signal Process. Lett.*, vol. 25, no. 8, pp. 1236–1240, Aug. 2018.  
 [9] Y. Liu, Y. H. Hu, and Q. Pan, "Distributed, robust acoustic source localization in a wireless sensor network," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4350–4359, Aug. 2012.  
 [10] E. Bassak and S. M. Karbasi, "Outlier censoring via block sparse learning," *IEEE Trans. Signal Process.*, vol. 71, pp. 1307–1318, 2023.  
 [11] M. Thottan and C. Ji, "Anomaly detection in IP networks," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2191–2204, Aug. 2003.  
 [12] K. Gopalakrishnan, M. Z. Li, and H. Balakrishnan, "Identification of outliers in graph signals," in *Proc. IEEE 58th Conf. Decis. Control*, 2019, pp. 4769–4776.  
 [13] S. M. Karbasi, "Joint likelihood estimation and model order selection for outlier censoring," *IET Radar, Sonar Navigation*, vol. 15, no. 6, pp. 561–573, 2021.  
 [14] S. Han, A. De Maio, V. Carotenuto, L. Pallotta, and X. Huang, "Censoring outliers in radar data: An approximate ML approach and its analysis," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 2, pp. 534–546, Apr. 2019.  
 [15] B. Tang, J. Tang, and Y. Peng, "Detection of heterogeneous samples based on loaded generalized inner product method," *Digit. Signal Process.*, vol. 22, no. 4, pp. 605–613, 2012.  
 [16] K. Gerlach, "Outlier resistant adaptive matched filtering," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 3, pp. 885–901, Jul. 2002.  
 [17] S. Han, L. Pallotta, V. Carotenuto, A. De Maio, and X. Huang, "An approximate regularized ML approach to censor outliers in Gaussian radar data," *IEEE Access*, vol. 7, pp. 66263–66274, 2019.  
 [18] C. Vogler, S. Goldenstein, J. Stolfi, V. Pavlovic, and D. Metaxas, "Outlier rejection in high-dimensional deformable models," *Image Vis. Comput.*, vol. 25, no. 3, pp. 274–284, 2007.  
 [19] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *J. Roy. Stat. Soc., Ser. B.*, vol. 57, no. 1, pp. 289–300, 1995.  
 [20] P. Stoica and P. Babu, "False discovery rate (FDR) and familywise error rate (FER) rules for model selection in signal processing applications," *IEEE Open J. Signal Process.*, vol. 3, pp. 403–416, 2022.  
 [21] P. Stoica and Y. Selen, "Model-order selection: A review of information criterion rules," *IEEE Signal Process. Mag.*, vol. 21, no. 4, pp. 36–47, Jul. 2004.  
 [22] J. Chen and Z. Chen, "Extended Bayesian information criteria for model selection with large model spaces," *Biometrika*, vol. 95, no. 3, pp. 759–771, 2008.  
 [23] D. J. Garling, *Inequalities: A Journey Into Linear Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2007.  
 [24] P. Stoica and Y. Selen, "Cyclic minimizers, majorization techniques, and the expectation-maximization algorithm: A refresher," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 112–114, Jan. 2004.  
 [25] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2017.  
 [26] T. Söderström and P. Stoica, *System Identification*. New York, NY, USA: Prentice-Hall, 1989.  
 [27] B. Laurent and P. Massart, "Adaptive estimation of a quadratic functional by model selection," *Ann. Statist.*, vol. 28, no. 5, pp. 1302–1338, 2000.  
 [28] J. Li, P. Stoica, and Z.-S. Liu, "Comparative study of IQML and MODE direction-of-arrival estimators," *IEEE Trans. Signal Process.*, vol. 46, no. 1, pp. 149–160, Jan. 1998.