

SIX HEURISTIC ALGORITHMS FOR SOLVING THE K -MINIMUM VOLUME ELLIPSOID (K -MVE) PROBLEM

Pavel Shcherbakov

Institute of Control Science,
Russian Academy of Sciences
Moscow, Russia
cavour118@mail.ru

Article history:

Received 13.12.2020, Accepted 25.12.2020

Abstract

We propose six heuristic methods for finding an approximate solution to the following combinatorial problem: Given N points in the n -dimensional space, find the minimum-size ellipsoid covering exactly $N - k$ of them, where k is much less than N . Various assumptions on the nature of the points and their amount are considered; the results of illustrative numerical experiments with the algorithms are discussed.

Key words

Ellipsoids, linear matrix inequalities, sampling, covariance matrix, principal component analysis, ℓ_1 -optimization.

1 Introduction

The presence of errors and inaccuracies in the data is typical to most of the real-life physical processes, since usually, there are no precise models of the phenomenon in question. Verification of experimental data is one of the key issues when applying physical theories to the practice; adequate decision making from available observations and model formation are the necessary inherent features of all approaches and algorithms used in cybernetics.

In many areas of system theory and cybernetics such as data mining and data transmission, decision making, sparse data representation, estimation and filtering, etc., it is often the case that the available data is not only erroneous and noise-corrupted, but is excessive too. At the same time, the capacity of the storage is limited, so that some of the observation data is to be discarded without an essential loss of useful information.

The amount of literature on the subject is enormous, and we list just several most influencing and classical texts [Ruan et al., 2005; Donoho, 2006; Becker

et al., 2009; Burke and Kendall, 2014]. In particular, see [Burke and Kendall, 2014] for a detailed discussion and pointers to the literature.

In this note we discuss a very simple model formulation that can be thought of as the core of problems of this sort. Namely, the following problem will be considered:

Discard k out of the $N \gg k$ given points $x_i \in \mathcal{S} \subset \mathbb{R}^n$ in such a way that the ellipsoid around the remaining ones be smallest in size.

In practical applications, the available multidimensional points are usually associated with sets of properties of a physical object or phenomenon of interest. The problem formulated above often arises in the processing of many samples of these data sets related to batches of the input raw material in order to detect abnormalities and improve the output of the manufacturing process; e.g., see [Van Aelst and Rousseeuw, 2009] for the examples of paper production and TV sets production lines and the experimental setup discussed there.

In cyber-physical systems of various nature, the formulation above may have numerous applications. For instance, the capacity of data transmission lines in such systems is limited; e.g., this is typically the case for multiagent systems (groups of robots, etc., as in [Tomashevich et al., 2018]), and some of the data is to be discarded. These considerations also relate to optimal task scheduling between the computing devices in distributed computer systems, [Amelina et al., 2018]. Some large-scale network-related problems, such as reducing the dimensionality and detecting critical in some sense nodes and paths in the network [Runge et al., 2015], can also be categorized as problems of this sort. Yet another important area where solution of the problem of our interest may be useful is control and optimization of power grids, with emphasis on robustness against various disturbances, [Mehrmann et al., 2013].

The ellipsoidal shape of the enclosing set in the problem above is adopted by many reasons. First, it is flexible enough (e.g., as compared to rectangles) yet simple (e.g., as compared to polytopes), being defined by a center and a shaping matrix. Even more importantly, this setup captures the typical case where noise-corrupted observations containing outliers from a certain distribution class are available, and are to be filtered out. For example, the points x_i are known to be generated from a Gaussian distribution with unknown mean and covariance and contain low-level noise and small number of outliers; the goal is to recover the underlying Gaussian distribution. Particularly, the Gaussian type of uncertainty is very common to standard representation of data obtained from physical experiments. This prospective formulation motivates the use of ellipsoids as comprising sets, thus linking the problem to finding high probability confidence ellipsoids explaining the available data.

In the specialized computer science literature; e.g., see [Ahipařaoglu, 2015], the problem formulated above and its variations is known as k -MVE (here, k denotes the amount of discarded points and MVE stands for Minimum Volume Ellipsoid); it has the two basic components: a) finding the optimal or near-optimal subset of points of the desired cardinality ($N - k$), and b) exact or approximate computation of the minimum-size ellipsoid around a given point set.

Problem b), often referred to as the MVE Problem or MVEP, can be solved exactly in many ways. The “standard” approach is via solving an appropriate convex optimization problem with linear matrix inequality (LMI) constraints using interior point methods as in [Boyd et al., 1994] or [Boyd and Vandenberghe, 2004] and numerous handy software such as `cvx` [Grant and Boyd, 2020]. However applicability of these tools is limited to relatively low dimensions of data and size of datasets. On the other hand, there exist alternative methods for higher dimensional problems, based on different ideas. Optimization methods of this sort are proposed, e.g., in [Sun and Freund, 2004; Ahipařaoglu et al., 2008]; the reported dimensions that can be handled with are up to $N = 30\,000$ and $n = 30$.

Problem a) is inherently combinatorial, and several types of approaches to solution can be found in the literature.

The first one attempts at finding the exact solution but with a reduced computational complexity. Examples of such methods are a specialized version of the Branch-and-Bound algorithm as in [Agullo, 1996], or use of the so-called *LP-type* property of the considered problem, as in [Dabbene and Shcherbakov, 2019] (see [Bai et al., 2002; Gärtner, 2015] for LP-type problems). Note that the proposed solutions are still combinatorial, but the computational cost is much lower.

The second, very popular approach is based on randomization in the spirit of [Tempo et al., 2013]; both the data and the outcome of a method are assumed to be random, and provable probabilistic estimates of the

accuracy and computational complexity can sometimes be provided. The related literature is large enough; we mention [Van Aelst and Rousseeuw, 2009], where an extensive bibliography on the topic is presented and resampling algorithms with modifications were analyzed together with some statistical properties of the solution obtained. A very deep paper [Gärtner, 2015] describes in detail the sampling-and-removal approach to solving similar problems, with the accent on LP-type problems mentioned above. In the large, this approach is closely related to the ideas of chance-constrained optimization in [Campi and Garatti, 2011]. We also mention the recent work [Dabbene et al., 2015], where a related problem was considered, namely, the construction of randomized approximations to nonconvex sets by regularly shaped bodies (boxes, polytopes, ellipsoids).

The third approach may be referred to as deterministic approximate; a solution is based on one or another “efficient heuristic,” the accuracy of the outcome can hardly be evaluated rigorously, but numerical experiments testify to reasonable or good performance. As an example of such methods, we mention [Ahipařaoglu, 2015], where the so-called 2-exchange iterative procedure is proposed; it is based on comparing two covering ellipsoids comprising the points sets which differ by one point. This procedure is rather efficient, but the outcome depends on the initial “guess,” so that the algorithm is to be run repeatedly from various initial conditions.

In this note we follow the third route and present several simple approximate deterministic methods. The proposed algorithms are based on either common sense reasonings or one or another efficient heuristics; they are then tested numerically over sets of randomly generated data. For small values of N and k , the exact solution of the k -MVE problem (optimal ellipsoid \mathcal{E}^*) can be found by checking all combinations of k discarded points out of the total N , computing the associated minimum-size ellipsoid around the rest of them and picking the best one. The outcomes of the algorithms are then compared to each other and to the optimal solution.

For large N, k , we simply run the algorithms over each of the data sets and compare the results using several performance indices.

This paper is of experimental and discussion flavor. Thus, no proofs of convergence and theoretical estimates of performance are given, and the methods were tested over a rather limited set of problems. These tasks are a matter of future work. The interested readers are welcome to propose their own methods to be tested and competed with the ones reported in this note; the MATLAB codes of the algorithms presented below are available upon request.

The rest of the paper is organized as follows. In the next section we recall some basic definitions of ellipsoids and provide an LMI-formulation of the problem of finding the minimum-volume ellipsoid around a given point set. The proposed approximate k -MVE algorithms,

together with possible modifications, are described in Section 3. The construction of the test datasets and the results of numerical experiments over this data are described in Section 4. In the last section we present concluding remarks and outline possible directions for future research.

This paper is an extension and revision of the preliminary text [Shcherbakov, 2020] presented at a conference. Numerous misprints in the text and the notation are corrected and the language is improved, many formulations are made more accurate, the overall motivation is modified in the extended introduction, the bibliography list is extended, and the numerical simulations are refined.

2 Minimum-Size Ellipsoid Around a Given Point Set

In this section, we introduce the necessary notation and recall some standard definitions; e.g., see [Boyd et al., 1994; Boyd and Vandenberghe, 2004].

An ellipsoid in \mathbb{R}^n is specified by its matrix and center, the typical definition being

$$\mathcal{E} = \{x \in \mathbb{R}^n : (x - x_c)^\top P^{-1}(x - x_c) \leq 1\}, \quad (1)$$

where $P \succ 0$ is a positive definite matrix defining the shape of the ellipsoid and $x_c \in \mathbb{R}^n$ is its center. There exist other descriptions which fit well different size-minimization problems. The most commonly used size criteria for ellipsoids are the following: (i) the radius of the smallest ball containing the ellipsoid; (ii) the sum of the squared semiaxes; (iii) the volume.

In this paper we consider the most natural *volumetric criterion* with $\text{Vol}(\mathcal{E}) = \nu_n(\det P)^{1/2}$, where ν_n is the volume of the n -dimensional unit Euclidean ball, and $\det(\cdot)$ stands for the determinant of a matrix. To facilitate optimization, we adopt the following equivalent definition of ellipsoid (1):

$$\mathcal{E} = \{x \in \mathbb{R}^n : \|Qx - a\| \leq 1\}, \quad (2)$$

where $Q = P^{-1/2}$, $a = P^{-1/2}x_c$, and $\|\cdot\|$ is the Euclidean vector norm. Using Schur complement, the inequality in (2) can be equivalently written as the LMI

$$\begin{pmatrix} 1 & (Qx - a)^\top \\ Qx - a & I \end{pmatrix} \succcurlyeq 0,$$

where I is the identity matrix. Next, since $P = Q^{-2}$, we have $\log \det P = -2 \log \det Q$. Hence, given the points x_1, \dots, x_N , and keeping in mind that the function $-\log \det X$ is convex for $X \succ 0$, we formulate the following *convex optimization problem* in the variables $Q = Q^\top \in \mathbb{R}^{n \times n}$ and $a \in \mathbb{R}^n$:

$$\text{MVE Problem :} \quad -\log \det Q \rightarrow \min \quad (3)$$

subject to the LMI constraints

$$\begin{pmatrix} 1 & (Qx_i - a)^\top \\ Qx_i - a & I \end{pmatrix} \succcurlyeq 0, \quad i = 1, \dots, N, \quad Q \succ 0.$$

The solution of this Minimum Volume Ellipsoid Problem, MVEP, defines the matrix $P = Q^{-2}$ and the center $x_c = Q^{-1}a$ of the desired minimum-volume ellipsoid (1) comprising the points x_1, \dots, x_N .

For simplicity, we assume that the point set $\{x_1, \dots, x_N\}$ is full-dimensional; i.e., it does not belong to a lower-dimensional subspace of \mathbb{R}^n .

The MVEP optimization problem is well-defined and can be efficiently solved by means of numerous available software; we will use the MATLAB-compatible optimization package `CVX` [Grant and Boyd, 2020]. From the structure of the problem, it is seen that the number of variables and constraints in this problem grow slowly as n, N grow, and memory requirements are also modest; hence, problems of reasonable dimensions can be easily handled within `CVX`. However even in low dimensions and small N , it still takes quite an amount of time; for instance, in a very simple situation $n = 2$, $N = 10$, $k = 3$, the computations require about 0.2sec on a standard modern laptop, while for $N = 100$ the solution requires about 1.5sec, and 11sec for $n = 10$, $N = 200$. In the experiments, we restrict ourselves to relatively small values of n and N ; see discussion in Sections 4 and 5.

The MVEP problem is at the nucleus of all methods discussed in this paper; it is to be solved repeatedly as an algorithm runs, so that the amount of calls to this procedure can be taken as one of the measures of numerical complexity of an algorithm.

3 The Six Tested Algorithms

The overall structure of all methods analyzed in this note is simple enough. The points are discarded one by one at every iteration until the amount of the remaining ones is exactly $N - k$. To discard a point, one or another heuristic is exploited, and attempt is made toward locally (sub)optimal solution. In this sense, the methods are *greedy*.

In this section we describe six methods of this sort and discuss their possible modifications.

3.1 “Naive” Methods

Algorithms in this group are based on common sense reasonings.

Method I: Spherical Peeling (SP). This is probably the simplest method that can be devised. At every iteration, having M remained points x_i , compute the mean value

$$\bar{x} = \frac{1}{M} \sum_{i=1}^M x_i \quad (4)$$

and compose the ball of minimum radius that contains all M points. Discard any of them which is on the surface of the ball (in other words, drop out the point that

maximizes the distance to the mean). After k iterations, solve the MVEP (3) for the remaining $N - k$ points.

This method is very simple and fast, it requires just one call to the MVEP routine; however, clearly, its performance is expected to be poor as confirmed by experiments.

Method II: Ellipsoidal Peeling (EP). This method is borrowed from [Tempo et al., 2013], see Algorithms 12.2, 12.3; it is a natural generalization of the previous one. At every iteration, we solve the MVEP problem (3) for the currently available M points and remove any one of them which belongs to the boundary. Repeat calculations until exactly $(N - k)$ points remain. This method requires $(k + 1)$ calls to the MVEP routine.

A simplified version of the EP-method consists in discarding in one shot the whole bunch of k points with the minimum distance to the surface of the initial ellipsoid (same goes to the SP). Alternatively, discard the points which are precisely on the boundary of the ellipsoid (in case their amount is less than or equal to k) and re-iterate until exactly k points are discarded. Experiments demonstrate similar performance of the versions with a slight advantage of the basic one-discarded-point version.

Method III: Convex Hull Peeling (CHP). The consideration behind this approach is the following. Having M points at a current iteration, construct their convex hull and consider all combinations of $M - 1$ points obtained by deleting one point from the convex hull. For each of the combinations, solve the MVEP problem (3) and choose the one with the smallest volume; re-iterate.

The cardinality of the convex hull of N points uniformly distributed in the cube in \mathbb{R}^n is $O(\log^{n-1} N)$ [Raynaud, 1970]. Hence, for this algorithm, the amount of calls to the MVEP solver is of the order of $k O(\log^{n-1} N)$. Similar estimates hold for Gaussian distributions, [Hueter, 1999] (hence, including balls, not cubes).

Note that the computation of the convex hull is a costly operation in higher dimensions. Experiments confirm reasonable performance of the method; however, it is rather sensitive to the presence of clusters.

We next pass on to “more advanced” algorithms that exploit heuristics based on known results and techniques from statistics, estimation and sparse data recovery.

3.2 Sample Covariance Matrix

Whereas the methods considered so far pay no attention to the origin of the data, the heuristics behind the approach below is dictated by the stochastic nature of the observations; namely, the points are generated by a random mechanism and have a certain probabilistic distribution.

Method IV: Covariance Matrix (Cov). At each step, having M points, compute their sample mean (4) and

the sample covariance matrix

$$H = \frac{1}{M} \sum_{i=1}^M (x_i - \bar{x})(x_i - \bar{x})^\top. \quad (5)$$

Discard the point x_{out} which maximizes the distance to \bar{x} in the metric defined by H :

$$x_{\text{out}} = \arg \max_i (x_i - \bar{x})^\top H^{-1} (x_i - \bar{x}).$$

Iterate until exactly k points are discarded.

Likewise SP, this method requires just one call to the MVEP routine, but its accuracy is drastically better. Overall execution time is slightly larger than that of SP due to extra operations needed for the computation of matrix (5). A simple recursive procedure can be used for the re-computation.

3.3 Principal Component Analysis

The second “advanced” approach to solving the problem uses the ideas of the principal component analysis (PCA), [Jolliffe, 2002; Pearson, 1901] targeted at filtering out outliers.

Method V: Principal Component Analysis (PCA). The simplest version of the method proceeds as follows. We find the minimum-volume ellipsoid around all of the current points, project them onto the smallest axis, and drop out the point which maximizes the distance to the center.

Let (Q, a) define the current ellipsoid (2)

$$\mathcal{E} = \{x \in \mathbb{R}^n : (x - x_c)^\top Q^{-1} (x - x_c) \leq 1\},$$

and let e be the eigenvector of Q associated with the smallest eigenvalue. Then the projection of x_i on the smallest axis of Q is equal to

$$\pi_e(x_i) = e^\top (x_i - x_c) e + x_c,$$

and the distance to x_c is equal to

$$\begin{aligned} \text{dist}(\pi_e(x_i), x_c) &= \|\pi_e(x_i) - x_c\| = \\ &= \|e^\top (x_i - x_c) e\| = |e^\top (x_i - x_c)|. \end{aligned}$$

Discard x_i that maximizes the distance, call it x_{small} .

In the experiments, we also considered projections on the *largest* axis, detected the corresponding point x_{large} and discarded the worst one in the pair $(x_{\text{small}}, x_{\text{large}})$. As per experiments, such a modification is more efficient; it requires $2k + 1$ calls to the MVEP routine.

Yet another modification of this algorithm is a mixture of methods *Cov* and *PCA*; namely, the points are projected on the small axis of the ellipsoid defined by the sample covariance matrix H , not Q . Then, the two points with maximum and minimum projections on the smallest and the largest axis, respectively, are determined, the two corresponding ellipsoids are compared and the best one is chosen.

3.4 ℓ_1 -optimization

The third general approach is based in the ideas of ℓ_1 -optimization and sparsity, [Donoho, 2006; Boyd and Vandenberghe, 2004].

Method VI: Ideas of ℓ_1 -optimization (ℓ_1). The scheme of the algorithm is as follows. Let Q and a denote the shaping matrix and the center of the current iteration ellipsoid. Instead of the condition $\|Qx - a\|^2 \leq 1$, consider the the constraints

$$\|Qx_i - a\|^2 \leq 1 + d_i, \quad d_i \geq 0, \quad i = 1, \dots, M,$$

where d_i penalizes the point x_i to leave the ellipsoid. Introduce the vector variable $d = (d_1, \dots, d_M)^\top$ and solve the following convex optimization problem in the variables Q, a, d :

$$-\log \det Q + \mu \|d\|_1 \rightarrow \min$$

subject to the constraints

$$\|Qx_i - a\|^2 \leq 1 + d_i, \quad d_i \geq 0; \quad i = 1, \dots, M, \quad Q \succ 0,$$

where $\|d\|_1 = \sum_i |d_i|$ is the vector ℓ_1 -norm and $\mu > 0$ is the scalar parameter (the constraints above can be re-written in the LMI-form similarly to those in the MVEP (3)). Let \mathcal{E}_μ denote the ellipsoid defined by the solution (Q, a) of this convex problem. Clearly, for large μ , all points are inside \mathcal{E}_μ , and as μ decreases, the resulting ellipsoid shrinks and contains no points.

Hence, we start with μ large enough and decrease it (by dichotomy) until exactly one point is outside the corresponding ellipsoid \mathcal{E}_μ . Drop it away and restart with the set of remaining points; repeat calculations until k points are outside.

In the experiments, we used the following modification of the method. We decreased μ until *half* of the points are outside the ellipsoid (more accurately, $\lfloor M/2 \rfloor$ of them), discard the point with the largest value of the penalty d_i and restart with the rest of them.

The heuristics behind this modification is not quite clear, but experiments testify to a much better performance. The amount of calls to the MVEP is hard to evaluate, but it is seen to be “large.”

3.5 Other Possible Modifications

Several obvious modifications are plausible.

1. Likewise method *EP*, all other methods admit a much faster yet essentially less accurate single-iteration simplification: Remove exactly k worst points in one shot.
2. At every iteration, the center of the ellipsoid is fixed as the mean of the currently remaining points.
3. Brute force at the last iteration: Having $N - k + 1$ points, test all combinations of $N - k$ points and choose the one that minimizes the volume (cf. method *CHP*).

4. In the special case $n = 2$, the ellipsoid is defined by a small number of parameters (two for the center and three for the matrix), so that straightforward optimization may be arranged.

4 Experiments

In the experiments, we restricted our analysis to low-dimensional small-sized datasets. The reason is that we used the `cvx` toolbox which implements interior point optimization methods; for high dimensions, memory requirements may be excessive and execution time may be large. As mentioned in the Introduction, more powerful optimization routines may be used to solve the MVEP problem in higher dimensions (e.g., as in [Sun and Freund, 2004; Ahipasaoglu et al., 2008]). However, in this work we were primarily interested in the volumetric performance of the presented algorithms that use different *heuristics for choosing a suboptimal subset of a given cardinality*, rather than in fast solution of the MVEP problem. Needless to say, the synthetic data that we operate with in the experiments below are just of illustrative nature; still, it is believed that, in a sense, they may approximately characterize simplest physical processes such as the behavior of small collections of particles, simple mechanical systems, etc.

4.1 Construction of the Datasets

In every experiment, we compare the performance of the algorithms over $N_{\text{set}} = 1,000$ randomly generated data sets, which are sampled from two distributions.

The first option is the Gaussian distribution $x_i \sim \mathcal{N}(0, \Sigma)$, which is the most natural one; it is generated as $\mathbf{x} = F * \text{randn}(n, 1)$, where $F \in \mathbb{R}^{n \times n}$, so that $\Sigma = FF^\top$. The matrix F is either fixed in every experiment, or also generated randomly as $F = 2 * \text{rand}(n) - 1$ for every dataset.

The second option is the uniform distribution over the unit box $\mathcal{B} = \{x \in \mathbb{R}^n : \|x\|_\infty \leq 1\}$.

Next, denote by $v_i(A)$ the volume of the ellipsoid obtained by algorithm *A* for the dataset i , and let \mathbf{v}_i be the best outcome for the dataset i . Alternatively, for small N, k , the quantity \mathbf{v}_i is associated with the optimal ellipsoid, which can be found by checking all $\text{nchoosek}(N, k)$ combinations of discarded points.

The performance of the algorithm *A* will be characterized by the following indices:

- the mean relative volume:

$$\text{volume}(A) = \frac{1}{N_{\text{set}}} \sum_{i=1}^{N_{\text{set}}} \frac{v_i(A)}{\mathbf{v}_i}$$

- the standard deviation of the random variable $\xi_i = \frac{v_i(A)}{\mathbf{v}_i}$:

$$\text{std}(A) = \left(\frac{1}{N_{\text{set}} - 1} \sum_{i=1}^{N_{\text{set}}} (\xi_i - \bar{\xi})^2 \right)^{1/2}, \quad \bar{\xi} = \frac{1}{N_{\text{set}}} \sum_{i=1}^{N_{\text{set}}} \xi_i$$

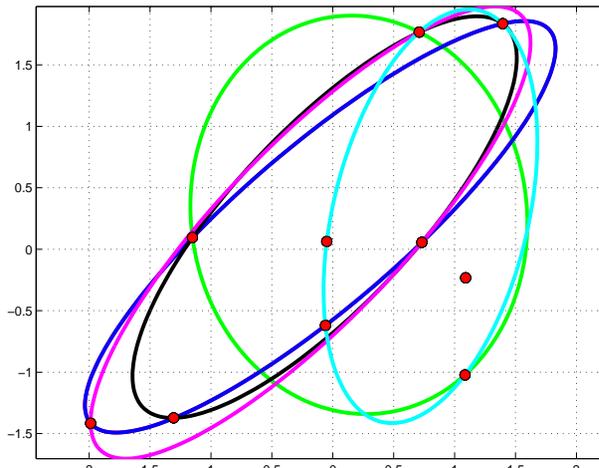


Figure 1. Outcomes of different algorithms for a (2,10,3) dataset. Cyan: *true*, Cov , ℓ_1 ; green: *SP*; blue: *EP*; black: *CHP*; magenta: *PCA*

- average amount of “wins” (algorithm A outperforms other algorithms in an experiment); alternatively, the averaged amount of true answers:

$$\text{wins}(A) = \text{card}\{v_i(A) = \mathbf{v}_i\}$$

- average amount of calls to the MVEP routine
- average execution time, $\text{time}(A)$.

4.2 Simulations

The first experiment: $n = 2, N = 10, k = 3$. This is an illustrative, very low-dimensional example where the exact solution \mathbf{v}_i can be found easily. This example is intended to show that (i) “often,” some of the methods produce the exact result, and (ii) the outcome may differ dramatically from method to method.

In all runs, the data was sampled from the Gaussian distribution $\mathcal{N}(0, \Sigma)$, where $\Sigma = FF^T$ and F is fixed:

$$F = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The results are given in the table below.

Table 1. Results of the first experiment

	volume	wins	calls	time	std
<i>SP</i>	1.3991	29%	1	0.3002	0.5701
<i>EP</i>	1.1301	62%	4	1.1785	0.2322
<i>CHP</i>	1.1077	63%	19.38	4.7701	0.2280
<i>Cov</i>	1.0310	76%	1	0.3102	0.0880
<i>PCA</i>	1.1200	56%	7	1.7777	0.2103
ℓ_1	1.0290	77%	20.12	8.1612	0.0861
<i>true</i>	1	100%	120	32.5182	0

The first observation is that, in terms of performance (volume), all the methods are similar (except for *SP* which is much worse). Perhaps this is explainable by low dimensions and sample-sizes.

The second observation is that, very often, the methods produce the true answer; especially, this is the case for *Cov* and ℓ_1 .

The third observation is that the two latter methods are very similar in performance, with 95% coincidences of the resulting ellipses; though ℓ_1 is much more time consuming.

Finally, the results demonstrate a potentially large variety of the outcomes of different methods; e.g., as shown in Fig. 1.

The second experiment: $n = 2, N = 10, k = 3$, but the points were sampled from the uniform distribution over the unit box. In this illustrative example, the methods exposed different behavior, indicating that the performance heavily depends on *a priori* assumptions on the nature of points.

The results are accumulated in Table 2, and they are seen to differ from those in the previous experiment.

Table 2. Results of the second experiment

	volume	wins	calls	time	std
<i>SP</i>	1.3298	15%	1	0.2801	0.3717
<i>EP</i>	1.2600	43%	4	1.1731	0.3412
<i>CHP</i>	1.2198	41%	18.73	4.9100	0.3501
<i>Cov</i>	1.1237	63%	1	0.2844	0.2399
<i>PCA</i>	1.2487	44%	7	1.7554	0.4371
ℓ_1	1.1262	55%	20.39	8.1113	0.2289
<i>true</i>	1	100%	120	30.5170	0

Indeed, whereas the execution time and the amount of calls to the MVEP routine remain the same (as expected), the number of exact answers is much lower, as well as the volumetric accuracy (the second column of Table 2). Another interesting observation is that the standard deviation (last column) increased considerably for all the methods except for *SP*. However, similarly to the first experiment, methods *Cov* and ℓ_1 outperform the other methods as seen from columns 2 and 3.

The third experiment: $n = 2, N = 100, k = 3$. In this slightly more “sizeable” experiment, we considered points x_i from Gaussian distributions with different covariance matrices for each of the $N_{\text{set}} = 1,000$ runs (see explanations at the beginning of Section 4.1). The rest of the quantities and indices are the same as above; clearly, computation of the exact answer is impossible due to the

size of the dataset (it would require 161,700 calls to the MVEP routine for every dataset). The results are summarized in Table 3.

Table 3. Results of the third experiment

	volume	wins	calls	time	std
<i>SP</i>	1.1412	6%	1	1.4391	0.1531
<i>EP</i>	1.0909	30%	4	6.1777	0.0531
<i>CHP</i>	1.0112	86%	29.35	42.0012	0.0161
<i>Cov</i>	1.0361	45%	1	1.4774	0.0509
<i>PCA</i>	1.0434	41%	7	9.7220	0.0602
ℓ_1	1.0399	42%	50.02	100.5192	0.0509

Several obvious observations are immediate.

First, of course, the execution time increased considerably because of the increased amount of constraints in the MVEP problem (3).

Second, the *CHP* method produced overwhelmingly better results; however, at the expense of a much higher computational effort (execution time increased by a factor of 8.5 with a ten-fold increase of the amount N of points). An explanation is that this method is somewhat close in nature to brute force methods, and it can hardly be applied to higher dimensional problems.

Finally, the three methods *Cov*, *PCA*, and ℓ_1 have shown a very similar accuracy, but the execution time is much lower for *Cov*.

Having the results of the experiments above and keeping all performance indices in mind, we conclude that method *Cov* based on use of sample covariance matrix is seen to be most attractive and promising.

5 Conclusions and Future Research

We proposed several simple and transparent methods for solving the k -MVE problem and presented preliminary simulation results. Theoretical estimates of the performance can hardly be obtained; hence, more experiments are desired and they will be performed in the future. Overall, this kind of research is thought of being useful in the development of new approaches to the construction of “economy” cybernetical models of various real-life physical systems.

Below, we list some of the important issues that should be taken into account in the experiments.

1. Origin of the points: From the experiments, the performance is seen to depend on the distribution of the data and the support set S .
2. Amount N of points and the relation between N and k . This knowledge perhaps can be taken into account to improve the methods.

3. Presence of outliers (use of intentionally contaminated Gaussian distributions) and clusters.

We conclude the paper by formulating in rough terms the four problems closely related to the k -MVE problem; it is believed that the analysis of these problems may be interesting and useful.

Problem 1: Given N points, find a nondegenerate ellipsoid \mathcal{E} that minimizes the cost

$$\frac{\text{Vol}(\mathcal{E})}{N_p},$$

where $N_p > 2$ is the amount of points in \mathcal{E} .

Problem 2: Given a distribution and a sample-size, evaluate the probability that the ellipsoid obtained with one or another method differs from the optimal one by no more than a prespecified quantity.

Problem 3: Given N points uniformly distributed in the cube, consider the random variable ω which is the volume of the minimal ellipsoid containing them. Evaluate $E\omega$, the mean of ω .

Problem 4: Given a fixed volume of an ellipsoid, maximize the amount of points that it covers by optimizing in Q, a . This can be thought of as an inverse to the k -MVE problem.

References

- Agullo, J. (1996). Exact iterative computation of the multivariate minimum volume ellipsoid estimator with a branch and bound algorithm. In Pratt, A., editor, *Proceedings in Computational Statistics*, pp. 175–180. Physica-Verlag, Heidelberg.
- Ahipařaođlu, S. (2015). Fast algorithms for the minimum volume estimator. *Journal of Global Optimization*, **62** (2), pp. 351–370.
- Ahipařaođlu, S., P, S., and Todd, M. (2008). Linear convergence of a modified Frank-Wolfe algorithm for computing minimum-volume enclosing ellipsoids. *Optimization Methods and Software*, **23**, pp. 5–19.
- Amelina, N., Granichin, O., Granichina, O., Ivanskiy, Y., and Jiang, Y. (2018). Adjustment of consensus protocol step-size in a network system with different task priorities via SPSA-like algorithm under the cost constraints. *IFAC-PapersOnLine*, **51** (23).
- Bai, E.-W., Chi, H., Tempo, R., and Ye, Y. (2002). Optimization with few violated constraints for linear bounded error parameter estimation. *IEEE Trans. Autom. Control*, **47** (7), pp. 1067–1077.
- Becker, S., Bobin, J., and Candes, E. (2009). NESTA: A fast and accurate first-order method for sparse recovery. *SIAM J. on Imaging Sciences*, **4** (1), pp. 1–39.
- Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia.
- Boyd, S. and Vandenberg, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge, MA.

- Burke, E. and Kendall, G., editors (2014). *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. Springer Science+Business Media, New York.
- Campi, M. and Garatti, S. (2011). A sampling-and-discarding approach to chance-constrained optimization: Feasibility and optimality. *Journal of Optimization Theory and Applications*, **148**, pp. 257–280.
- Dabbene, F., Henrion, D., Lagoa, C., and Shcherbakov, P. (2015). Randomized approximations of the image set of nonlinear discrete-time systems with applications to filtering. In *Proc. 8th IFAC Symposium on Robust Control Design (ROCOND 2015)*, Bratislava, Slovak Republic, pp. 37–42.
- Dabbene, F. and Shcherbakov, P. (2019). Minimum volume ellipsoid comprising a subset of points. In *Proc. 6th Int. Conf. Control, Decision Inform. Technologies (CoDIT 2019)*, paper WiP 20, Paris.
- Donoho, D. (2006). For most large underdetermined systems of linear equations the minimal 1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, **56**(6), pp. 797–829.
- Gärtner, B. (2015). Sampling with removal in LP-type problems. *Journal of Computational Geometry*, **6**(2), pp. 93–112.
- Grant, M. and Boyd, S. (2020). CVX: Matlab software for disciplined convex programming, ver. 2.2, <http://cvxr.com/cvx/>.
- Hueter, I. (1999). Limit theorems for the convex hull of random points in higher dimensions. *Trans. Amer. Math. Soc.*, **351**(11), pp. 4337–4363.
- Jolliffe, I. (2002). *Principal Component Analysis*. Springer Series in Statistics. Springer, New York.
- Mehrmann, V., Morandin, R., Olmiand, S., and Schöll, E. (2013). Qualitative stability and synchronicity analysis of power network models in port-Hamiltonian form. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, **28**(10), pp. 101102.
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, **2**, pp. 559–572.
- Raynaud, H. (1970). Sur l’enveloppe convexe des nuages de points aleatoires dans \mathbb{R}^n . I. *Journal of Applied Probability*, **7**(1), pp. 35–48.
- Ruan, D., Chen, G., Kerre, E., and Wets, G., editors (2005). *Intelligent Data Mining: Techniques and Applications*, vol. 5 of *Studies in Computational Intelligence*. Springer, New York.
- Runge, J., Petoukhov, V., Donges, J., Hlinka, J., Jajcay, N., Vejmelka, M., Hartman, D., Marwan, N., Palus, M., and Kurths, J. (2015). Identifying causal gateways and mediators in complex spatio-temporal systems. *Nature Communications*, **6**(8502).
- Shcherbakov, P. (2020). Finding minimum-size ellipsoids covering a subset of points: Several simple heuristic approaches. In *Proc. 7th Int. Conf. Control, Decision Inform. Technologies (CoDIT 2020)*, Pague, pp. 920–925.
- Sun, P. and Freund, R. (2004). Computation of minimum-volume covering ellipsoids. *Operations Research*, **52**(5), pp. 690–706.
- Tempo, R., Calafiore, G., and Dabbene, F. (2013). *Randomized Algorithms for Analysis and Control of Uncertain Systems: with Applications*. Springer.
- Tomashevich, S., Andrievsky, B., and Fradkov, A. (2018). Formation control of a group of unmanned aerial vehicles with data exchange over a packet erasure channel. In *Proc. 2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, St. Petersburg, Russia, pp. 38–43.
- Van Aelst, S. and Rousseeuw, P. (2009). Minimum volume ellipsoid. *WIREs Computational Statistics*, **1**, pp. 71–82.