Robust subgaussian estimation of a mean vector in nearly linear time

Jules Depersin and Guillaume Lecué

email: jules.depersin@ensae.fr, email: guillaume.lecue@ensae.fr
CREST, ENSAE, IPParis. 5, avenue Henry Le Chatelier, 91120 Palaiseau, France.

Abstract

We construct an algorithm, running in time $\tilde{\mathcal{O}}(Nd + uKd)$, which is robust to outliers and heavy-tailed data and which achieves the subgaussian rate from [31]

$$\sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\frac{\|\Sigma\|_{op} K}{N}} \tag{1}$$

with probability at least $1 - \exp(-c_0 K) - \exp(-c_1 u)$ where Σ is the covariance matrix of the *informative data*, $K \in \{1, \dots, K\}$ is some parameter (number of block means) and $u \in \mathbb{N}^*$ is another parameter of the algorithm. This rate is achieved when $K \geq c_2|\mathcal{O}|$ where $|\mathcal{O}|$ is the number of outliers in the database and under the only assumption that the informative data have a second moment. The algorithm is fully data-dependent and does not use in its construction the proportion of outliers nor the rate in (1). Its construction combines recently developed tools for Median-of-Means estimators and covering-Semi-definite Programming [8, 37]. We also show that this algorithm can automatically adapt to the number of outliers.

AMS subject classification: 62F35
Keywords: Robustness, algorithms, heavy-tailde data.

1 Introduction on the robust mean vector estimation problem

Estimating the mean of a random variable in a d-dimensional space when given some of its realizations is arguably the oldest and most fundamental problem of statistics. In the past few years, it has received important attention from two communities: the Statistics [5, 34, 7, 6, 31, 35, 32, 22, 9] and Computer Science [14, 13, 16, 15, 17, 18, 8] communities. Both communities consider the problem of *robust mean estimation*, focusing mainly on different definitions of robustness.

In recent years, many efforts have been made by the Statistics community on the construction of estimators performing in a *subgaussian way* for heavy-tailed data. Such estimators achieve the same statistical properties as the empirical mean of a N-sample of i.i.d. gaussian variables $\mathcal{N}(\mu, \Sigma)$ where $\mu \in \mathbb{R}^d$ and $\Sigma \succeq 0$ is the covariance matrix. In that case, for a given confidence $1 - \delta$, the subgaussian rate as defined in [31] is (up to an absolute multiplicative constant)

$$r_\delta = \sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\frac{||\Sigma||_{op} \log(1/\delta)}{N}} \tag{2}$$

where $\mathrm{Tr}(\Sigma)$ is the trace of Σ and $||\Sigma||_{op}$ is the operator norm of Σ. Indeed, it follows from Borell-TIS's inequality (see Theorem 7.1 in [26] or pages 56-57 in [27]) that with probability at least $1 - \delta$,

$$\left\|\bar{X}_N - \mu\right\|_2 = \sup_{\|v\|_2 \leq 1} \left\langle \bar{X}_N - \mu, v \right\rangle \leq \mathbb{E} \sup_{\|v\|_2 \leq 1} \left\langle \bar{X}_N - \mu, v \right\rangle + \sigma\sqrt{2\log(1/\delta)}$$

where $\sigma = \sup_{\|v\|_2 \leq 1} \sqrt{\mathbb{E}\langle \bar{X}_N - \mu, v\rangle^2}$. It is straightforward to check that $\mathbb{E}\sup_{\|v\|_2 \leq 1}\langle \bar{X}_N - \mu, v\rangle \leq \sqrt{\mathrm{Tr}(\Sigma)/N}$ and $\sigma = \sqrt{\|\Sigma\|_{op}/N}$, which leads to the rate in (2) (up to the constant $\sqrt{2}$ on the second term in (2)). In most of the recent works, the effort has been made to achieve the rate r_δ for i.i.d. heavy-tailed data even under the minimal requirement that the data only have a second moment. Under this second-moment assumption only, the empirical mean cannot achieve the rate (2) and one needs to consider other procedures[1]. Over the years, some

[1] Under only a second-moment assumption, the empirical mean achieves the rate $\sqrt{\mathrm{Tr}(\Sigma)/(\delta N)}$ which can not be improved in general.

procedures have been proposed to achieve such a goal: a Le Cam test estimator, called a tournament estimator in [31], a minmax Median-Of-Means estimator in [32] and a PAC-Bayesian estimator in [6]. The first two one are based on the median-of-means principle that we will also use.

On the other side, the Computer Science community mostly considers a different definition of robustness and targets a different goal. In many recent CS papers, algorithms (not only estimators) have been constructed and proved to be robust with respect to a *contamination* of the dataset that is when some of the data are replaced by other data which may have nothing to do with the original batch. This covers the Huber ϵ-contamination model but also adversarial data which receives an important attention recently in the deep learning community. Moreover, the Computer Science community looks at the problem of robust mean estimation from algorithmic perspectives such as the running time. A typical result in this line of research is Theorem 1.3 from [8] that we recall now.

Theorem 1 (Theorem 1.3, [8]). *Let $X_1, \dots, X_N$ be random vectors in $\mathbb{R}^d$. We assume that there is a partition $\{1, \dots, N\} = \mathcal{O} \cup \mathcal{I}$ such that nothing is assumed on $(X_i)_{i\in\mathcal{O}}$ and $(X_i)_{i\in\mathcal{I}}$ are independent with mean μ and covariance matrix $\Sigma \preceq \sigma^2 I_d$. We assume that $\epsilon = |\mathcal{O}|/N$ is such that $0 < \epsilon < 1/3$ and $N \gtrsim d\log(d)/\epsilon$. There exists an algorithm running in $\tilde{\mathcal{O}}(Nd)/\mathrm{poly}(\epsilon)$ which outputs $\hat{\mu}_\epsilon$ such that with probability at least $9/10$, $\|\hat{\mu}_\epsilon - \mu\|_2 \lesssim \sigma\sqrt{\epsilon}$.*

The first result proving the existence of a polynomial time algorithm robust to contamination may be found in [14]. Theorem 1 improves upon many existing results since it achieves the optimal information theoretic-lower bound with a (nearly) linear-time algorithm.

Finally, there are two recent papers for which both algorithmic and statistical considerations are important. In [22, 9], algorithms achieving the subgaussian rate in (2) have been constructed. They both run in polynomial time : $\mathcal{O}(N^{24} + Nd)$ for [22] and $\mathcal{O}(N^4 + N^2 d)$ for [9] (see [9] for more details on these running times). They do not consider a contamination of the dataset even though their results easily extend to this setup. Some other estimators which have been proposed in the Statistics literature are very fast to compute but they do not achieve the optimal subgaussian rate from (2). A typical example is Minsker's geometric median estimator [34] which achieves the rate $\sqrt{\mathrm{Tr}(\Sigma)\log(1/\delta)/N}$ in linear time $\tilde{\mathcal{O}}(Nd)$. All the later three papers use the Median-of-means principle. We will use this principle but only to construct a starting point (which will simply be the coordinate-wise median) and for the computation of the step size (where we will only use the one dimensional definition of the median along the descent line direction). What we mainly borrow from the literature on MOM estimators is the advantage to work with local block means instead of the data themselves. We will identify two such advantages by doing so: a stochastic one and a computational one (see Remark 4 below).

Robust mean estimation have been raised in pioneered works in robust statistics from Huber [23, 24], Tukey [39, 40] or Hampel [21, 20]. Their concerns was more about robustness to model misspecification and on the breakdown point property ("smallest amount of contamination necessary to upset an estimator entirely" taken from [19]). The computational problem connected to this issue was not of primary interest even though it was already raised, for instance, in Section 5.3 from [19] for the construction of Tukey contours (a d-dimensional definition of quantiles).

The aim of this work is to show that a single algorithm can answer the three problems: robustness to heavy-tailed data, to contamination and computational cost. In this article, we construct an algorithm running in time $\tilde{\mathcal{O}}(Nd + u\log(1/\delta)d)$ which outputs an estimator of the true mean achieving the subgaussian rate (2) with confidence $1-\delta$ (for $\exp(-c_0 N) \le \delta \le \exp(-c_1|\mathcal{O}|)$) on a corrupted database and under a second moment assumption only. It is therefore robust to heavy-tailed data and to contamination. Our approach takes ideas from both communities: the median-of-means principle which has been recently used in the Statistics community and a SDP relaxation from [8] which can be computed fast. The baseline idea is to construct K equal size groups of data from the N given ones and to compute their empirical means $\bar{X}_k, k = 1, \dots, K$. These K empirical means are used successively to find a robust descent direction thanks to a SDP relaxation from [8]. We prove the robust subgaussian statistical property of the resulting descent algorithm under the only following assumption.

Assumption 1. *There exists a partition $\mathcal{I} \cup \mathcal{O} = \{1, \dots, N\}$ of the dataset $(X_i)_{i\le N}$ such that 1) nothing is assumed on $(X_i)_{i\in\mathcal{I}}$ 2) $(X_i)_{i\in\mathcal{I}}$ are independent with mean μ and covariance $\mathbb{E}(X_i - \mu)(X_i - \mu)^\top) \preceq \Sigma$ where Σ is a given (unknown) covariance matrix.*

Assumption 1 covers the two concepts of robustness considered in the Statistics and Computer Science communities since the *informative data* (data indexed by $\mathcal{I}$) are only assumed to have a second moment and there are $|\mathcal{O}|$ outliers onto which we do not make any assumption. Our aim is to show that the rate of convergence (2) which is the rate achieved by the empirical mean in the ideal i.i.d. Gaussian case can be achieved in the corrupted and heavy-tailed setup from Assumption 1 with a fast algorithm.

The paper is organized as follows. In the next section, we give a high-level description of the algorithm and its statistical and computation performances. In section 3, we prove its statistical properties and give a precise definition

of the algorithm. In Section 4, we study the statistical performance of the SDP relaxation at the heart of the descent direction. In Section 5, we fully characterize its computational cost. In Section 6, we construct a procedure achieving the same statistical properties and can automatically adapt to the number of outliers.

2 Construction of the algorithms and main result

The construction of our robust subgaussian descent procedure is using two ideas. The first one comes from the median-of-means (MOM) approach which has recently received a lot of attention in the statistical and machine learning communities [4, 30, 12, 33, 34]. The MOM approach [36, 1, 25, 2] often yields robust estimation strategies (but usually at a high computational cost). Let us give the general idea behind that approach: we first randomly split the data into K equal-size blocks $B_1, \dots, B_K$ (if K does not divide N, we just remove some data). We then compute the empirical mean within each block: for $k = 1, \dots, K$,

$$\bar{X}_k = \frac{1}{|B_k|} \sum_{i \in B_k} X_i$$

where we set $|B_k| = \mathrm{Card}(B_k) = N/K$. In the one-dimensional case, we then take the median of the latter K empirical means to construct a robust *and subgaussian* estimator of the mean [12]. It is more complicated in the multi-dimensional case, where there is no *definitive* equivalent of the one dimensional median but several candidates: coordinate-wise median, the geometric median (also known as Fermat point), the Tukey Median, among many others (see [38]). The strength of this approach is the robustness of the median operator, which leads to good statistical properties even on corrupted databases. For the construction of our algorithm, we actually only use the idea of grouping the data and computing their K means $\bar{X}_k, k = 1, \dots, K$.

Finding good descent directions in the heavy-tailed and corrupted scenario considered in Assumption 1 in reasonnable time is a main issue. A construction has been proposed by [9] which also uses a SDP relaxation, which costs $\mathcal{O}(N^4 + Nd)$ to be computed. Our approach also uses a SDP relaxation, with an other SDP. It is based on the observation that μ is solution of the minimization problem $\min_{\nu \in \mathbb{R}^d} f(\nu)$ where $f : \nu \in \mathbb{R}^d \to \|\mathbb{E}X - \nu\|_2^2$ and X is any random vector with mean μ. One way to approach μ is therefore to run a gradient descent algorithm using f as an objective function: from $x_c \in \mathbb{R}^d$ we go to the next iteration with $x_c - \theta \nabla f(x_c)$ where $\theta \geq 0$ is a step size. Since $\nabla f(x_c) = x_c - \mathbb{E}X$, for $\theta = 1$, the latter algorithm achieves the target mean μ in one step, which is not surprising given that $x_c - \mathbb{E}X$ is the best descent direction towards $\mathbb{E}X$ starting from x_c. We can also re-write that as a matrix problem : the top eigenvector of

$$\underset{M \succeq 0, \mathrm{Tr}(M)=1}{\mathrm{argmax}} \left\langle M, (\mathbb{E}X - x_c)(\mathbb{E}X - x_c)^\top \right\rangle \tag{3}$$

is given by $\frac{x_c - \mathbb{E}X}{\|x_c - \mathbb{E}X\|_2}$, which is the best descent direction we are looking for.

Of course, we don't know $(\mathbb{E}X - x_c)(\mathbb{E}X - x_c)^\top$ in (3) but we are given a database of N data $X_1, \dots, X_N$ (among which $|\mathcal{I}|$ of them have mean μ). We use these data to estimate in a robust way the unknown quantity $(\mathbb{E}X - x_c)(\mathbb{E}X - x_c)^\top$ in (3). Ideally, we would like to identify the *informative data* and then use $(1/|\mathcal{I}|) \sum_{i \in \mathcal{I}} (X_i - x_c)(X_i - x_c)^\top$ or its block means version $(1/|\mathcal{K}|) \sum_{k \in \mathcal{K}} (\bar{X}_k - x_c)(\bar{X}_k - x_c)^\top$, where $\mathcal{K} = \{k : B_k \cap \mathcal{O} = \emptyset\}$, to estimate this quantity but this information is not available either.

To address this problem we use a tool introduced in [8] adapted to the block means. The idea is to endow each block mean $\bar{X}_k$ with a weight ω_k taken in Δ_K defined as

$$\Delta_K = \left\{ (\omega_k)_{k=1}^K : 0 \leq \omega_k \leq \frac{1}{9K/10}, \sum_{k=1}^K \omega_k = 1 \right\}.$$

Ideally we would like to put 0 weights to all block means $\bar{X}_k$ corrupted by an outliers. But, we cannot do it since $\mathcal{K}$ is unknown. To overcome this issue, we learn the optimal weights and consider the following minmax optimization problem

$$\max_{M \succeq 0, \mathrm{Tr}(M)=1} \min_{w \in \Delta_K} \left\langle M, \sum_{k=1}^K \omega_k (\bar{X}_k - x_c)(\bar{X}_k - x_c)^\top \right\rangle. \tag{E_{x_c}}$$

This is the dual problem from [8] adapted to the block means. The key insight from [8] is that an approximating solution M_c of the maximization problem in (E_{x_c}) can be obtained in reasonable time using a covering SDP approach [8, 37] (see Section 4). We expect a solution (in M) to (E_{x_c}) to be close to a solution of the minimization problem in (3) – which is $M^* = (\mu - \nu)(\mu - \nu)^\top / \|\mu - \nu\|_2^2$ – and the same for their top eigenvectors (up to the sign).

At a high level description, the robust descent algorithm we perform outputs $\hat{\mu}_K$ after at most $\log d$ iterations of the form $x_c - \theta_c v_1$ where v_1 is a top eigenvector of an approximating solution M_c to the problem (E_{x_c}) and θ_c is a step size. It starts at the coordinate-wise median of the means $\bar{X}_1, \dots, \bar{X}_K$. In Algorithm 4, we define precisely the step size and the stopping criteria we use to define the algorithm (it requires too many notation to be defined at this stage). This algorithm outputs the vector $\hat{\mu}_K$: its running time and statistical performances are gathered in the following result.

Theorem 2. *Grant Assumption 1. Let $K \in \{1, \dots, N\}$ be the number of equal-size blocks and assume that $K \geq 300|\mathcal{O}|$. Let $u \in \mathbb{N}^*$ be a parameter of the covering SDP used at each descent step. With probability at least $1 - \exp(-K/180000) - (1/10)^u$, the descent algorithm finishes in $\tilde{\mathcal{O}}(Nd + Kud)$ and outputs $\hat{\mu}_K$ such that*

$$\|\hat{\mu}_K - \mu\|_2 \leq 808 \left(1200 \sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\frac{1200 \|\Sigma\|_{op} K}{N}} \right).$$

To make the presentation of the proof of Theorem 2 as simple as possible we did not optimize the constants. Theorem 2 generalizes and improves Theorem 1 in several ways. We first improve the confidence from a constant "9/10" to an exponentially large confidence $1 - \exp(-c_0 K)$. We obtain the result for any covariance structure Σ and $\hat{\mu}_K$ does not require the knowledge of Σ for its construction. We obtain a result which holds for any N (even under the sample complexity). The construction of $\hat{\mu}_K$ does not require the knowledge of the exact proportion of outliers ϵ in the dataset unlike $\hat{\mu}_\epsilon$ in Theorem 1. We only need to know that $K \gtrsim |\mathcal{O}|$. Moreover, using a Lepskii adaptation method it is also possible to automatically choose K and therefore to adapt to the proportion of outliers if we have some extra knowledge on $\mathrm{Tr}(\Sigma)$ and $\|\Sigma\|_{op}$ (see Section 6 for more details). Moreover, if we only care about constant 9/10 confidence, our runtime does not depend on ϵ and is nearly-linear $\tilde{\mathcal{O}}(Nd)$. We also refer the reader to Corollary 2 for more comparison with Theorem 1.

Remark 1 (Nearly-linear time)**.** *We identify two important situations where the algorithm from Theorem 2 runs in nearly-linear time that is in $\tilde{\mathcal{O}}(Nd)$. First, when the number of outliers is known to be less than $\sqrt{N}$, we can choose $K \leq \sqrt{N}$ and $u = K$. In that case, the algorithm runs in $\tilde{\mathcal{O}}(Nd)$ and the subgaussian rate is achieved with probability at least $1 - 2\exp(-c_0 K)$ for some constant c_0 (see also Corollary 3 for an adaptive to K version of this result). Another widely investigated situation is when we only want to have a constant confidence like 9/10. In that case, one may chose $u = 1$ and any values of $K \in [N]$ can be chosen (so we can have any number of outliers) to achieve the subgaussian rate with constant probability and in nearly-linear time $\tilde{\mathcal{O}}(Nd)$ (see also Corollary 2 for an adaptive to K version of this result).*

Theorem 2 improves the result from [22, 9] since $\hat{\mu}_K$ runs faster than the polynomial times $\mathcal{O}(N^{24} + Nd)$ and $\mathcal{O}(N^4 + Nd)$ in [22] and [9]. The algorithm $\hat{\mu}_K$ also does not require the knowledge of $\mathrm{Tr}(\Sigma)$ and $\|\Sigma\|_{op}$. Finally, Theorem 2 provides running time guarantees on the algorithm unlike in [31, 32, 6] and it improves upon the statistical performances from [34].

3 Proof of the statistical performance in Theorem 2

In this section, we prove the statistical performance of $\hat{\mu}_K$ as stated in Theorem 2. We first identify an event $\mathcal{E}$ onto which we will derive the rate of convergence of the order of (2). This event is also used to compute the running time of $\hat{\mu}_K$ in the next section as announced in Theorem 2.

Proposition 1. *Denote by $\mathcal{E}$ the event onto which for all matrix $M \succeq 0$ such that $\mathrm{Tr}(M) = 1$, there are at least $9K/10$ of the blocks for which $\left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 \leq 8r$ where*

$$r = 1200 \sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\frac{1200 \|\Sigma\|_{op} K}{N}}. \tag{4}$$

If Assumptions 1 holds and $K \geq 300|\mathcal{O}|$ then $\mathbb{P}[\mathcal{E}] \geq 1 - \exp(-K/180000)$.

Proposition 1 contains all the stochastic arguments we will use in this paper (constants have not been optimized). In other words, after identifying $\mathcal{E}$ all the remaining arguments do not involve any other stochastic tools. Before proving Proposition 1, let us first state a result that is of particular interest beyond our problem.

Corollary 1. *On the event $\mathcal{E}$, for all $M \in \mathbb{R}^{d\times d}$ such that $M \succeq 0$ and $\mathrm{Tr}(\Sigma) = 1$ there are at least $9K/10$ blocks such that for all $x_c \in \mathbb{R}^d$,*

$$\left\|M^{1/2}(\mu - x_c)\right\|_2 - 8r \leq \left\|M^{1/2}(\bar{X}_k - x_c)\right\|_2 \leq \left\|M^{1/2}(\mu - x_c)\right\|_2 + 8r. \tag{5}$$

Let us now turn to a proof of Proposition 1. We first remark that if we were to only consider matrices M of rank 1, Proposition 1 would boil down to show that for all $v \in \mathcal{S}_2^{d-1}$ (the unit sphere in ℓ_2^d) on more than 9/10 blocks $|\langle v, \bar{X}_k - \mu\rangle| \leq 8r$. This is a "classical" result in the MOM literature which has been proved in [31] and [32]. We recall now this result and the short proof from [32] for completeness. We will use it to prove Proposition 1.

Lemma 1. *Grant Assumption 1 and assume that $K \geq 300|\mathcal{O}|$. With probability at least $1 - \exp(-K/180000)$, for all $v \in \mathcal{S}_2^{d-1}$, there are at least $99K/100$ of the blocks k such that $|\langle v, \bar{X}_k - \mu\rangle| \leq r$.*

Proof. We want to show that with probability at least $1 - \exp(-K/180000)$, for all $v \in \mathcal{S}_2^{d-1}$,

$$\sum_{k\in[K]} I(|\langle \bar{X}_k - \mu, v\rangle| > r) \leq K/100.$$

We take $\mathcal{K} = \{k \in [K] : B_k \cap \mathcal{O} = \emptyset\}$. We define $\phi(t) = 0$ if $t \leq 1/2$, $\phi(t) = 2(t - 1/2)$ if $1/2 \leq t \leq 1$ and $\phi(t) = 1$ if $t \geq 1$. We have $I(t \geq 1) \leq \phi(t) \leq I(t \geq 1/2)$ for all $t \in \mathbb{R}$ and so

$$\begin{aligned}
&\sum_{k\in\mathcal{K}} I(|\langle \bar{X}_k - \mu, v\rangle| > r) \leq \sum_{k\in\mathcal{K}} I(|\langle \bar{X}_k - \mu, v\rangle| > r) - \mathbb{P}[|\langle \bar{X}_k - \mu, v\rangle| > r/2] + \mathbb{P}[|\langle \bar{X}_k - \mu, v\rangle| > r/2] \\
&\leq \sum_{k\in\mathcal{K}} \phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right) - \mathbb{E}\phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right) + \mathbb{P}[|\langle \bar{X}_k - \mu, v\rangle| > r/2] \\
&\leq \sup_{v\in\mathcal{S}_2^{d-1}} \left(\sum_{k\in\mathcal{K}} \phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right) - \mathbb{E}\phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right)\right) + \sum_{k\in\mathcal{K}} \mathbb{P}[|\langle \bar{X}_k - \mu, v\rangle| > r/2].
\end{aligned}$$

For all $k \in \mathcal{K}$, we have

$$\mathbb{P}[|\langle \bar{X}_k - \mu, v\rangle| > r/2] \leq \frac{\mathbb{E}\langle \bar{X}_k - \mu, v\rangle^2}{(r/2)^2} \leq \frac{4Kv^\top\Sigma v}{Nr^2} \leq \frac{4K\sup_{v\in\mathcal{S}_2^{d-1}} v^\top\Sigma v}{Nr^2} = \frac{4K\|\Sigma\|_{op}}{Nr^2} \leq \frac{1}{300}$$

because $r^2 \geq 1200K\|\Sigma\|_{op}/N$. Next, using the bounded difference inequality (Theorem 6.2 in [3]), the symmetrization argument and the contraction principle (Chapter 4 in [27]), with probability at least $1 - \exp(-K/180000)$,

$$\begin{aligned}
&\sup_{v\in\mathcal{S}_2^{d-1}} \left(\sum_{k\in\mathcal{K}} \phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right) - \mathbb{E}\phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right)\right) \\
&\leq \mathbb{E}\sup_{v\in S} \left(\sum_{k\in\mathcal{K}} \phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right) - \mathbb{E}\phi\left(\frac{|\langle \bar{X}_k - \mu, v\rangle|}{r}\right)\right) + \sqrt{\frac{|\mathcal{K}|K}{360000}} \\
&\leq \frac{4K}{Nr}\mathbb{E}\sup_{v\in\mathcal{S}_2^{d-1}} \left\langle v, \sum_{i\in\cup_{k\in\mathcal{K}}B_k} \epsilon_i(X_i - \mu)\right\rangle + \sqrt{\frac{|\mathcal{K}|K}{360000}} \\
&= \frac{4K}{\sqrt{N}r}\mathbb{E}\left\|\frac{1}{\sqrt{N}}\sum_{i\in\cup_{k\in\mathcal{K}}B_k} \epsilon_i(X_i - \mu)\right\|_2 + \sqrt{|\mathcal{K}|K/360000} \leq \frac{K}{300}
\end{aligned}$$

because $r \geq 1200\mathbb{E}\left\|\sum_{i\in\cup_{k\in\mathcal{K}}B_k} \epsilon_i(X_i - \mu^*)\right\|_2/\sqrt{N}$ since

$$\mathbb{E}\left\|\frac{1}{\sqrt{N}}\sum_{i\in\cup_{k\in\mathcal{K}}B_k} \epsilon_i(X_i - \mu)\right\|_2 \leq \sqrt{\mathbb{E}\left\|\frac{1}{\sqrt{N}}\sum_{i\in\cup_{k\in\mathcal{K}}B_k} \epsilon_i(X_i - \mu)\right\|_2^2} = \sqrt{\frac{|\cup_{k\in\mathcal{K}}B_k|}{N}}\sqrt{\mathrm{Tr}(\Sigma)} \leq \sqrt{\mathrm{Tr}(\Sigma)}.$$

As a consequence, when $K \geq 300|\mathcal{O}|$, with probability at least $1 - \exp(-K/180000)$, for all $v \in \mathcal{S}_2^{d-1}$,

$$\sum_{k\in[K]} I(|\langle \bar{X}_k - \mu, v\rangle| > r) \leq |\mathcal{O}| + \frac{|\mathcal{K}|}{300} + \frac{K}{300} \leq \frac{K}{100}.$$

■

Proof of Proposition 1: Let $M \in \mathbb{R}^{d\times d}$ be such that $M \succeq 0$ and $\mathrm{Tr}(\Sigma) = 1$. Denote by $\mathcal{A}_M = \{k \in [K] : \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 \geq 8r\}$ and assume that $|\mathcal{A}_M| \geq 0.1K$. Let G be a Gaussian vector in $\mathbb{R}^d$ with mean 0 and covariance matrix M (and independent from $X_1, \ldots, X_N$). We consider the random variable $Z = \sum_{k\in[K]} I\left(|\langle \bar{X}_k - \mu, G\rangle| > 5r\right)$. We work conditionally to $X_1, \ldots, X_N$ in this paragraph. For all $k \in [K]$, $\langle \bar{X}_k - \mu, G\rangle$ is a centered Gaussian variable with variance $\sigma_k^2 := \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2^2$. In particular, for all $k \in \mathcal{A}_M$, if we denote by g a standard real-valued Gaussian variable, we have $\mathbb{P}_G\left[|\langle \bar{X}_k - \mu, G\rangle| > 5r\right] \geq \mathbb{P}_G\left[|\langle \bar{X}_k - \mu, G\rangle| > 5\sigma_k/8\right] = 2\mathbb{P}[g > 5/8] \geq 0.528$ (where $\mathbb{P}_G$ (resp. $\mathbb{E}_G$) denotes the probability (resp. expectation) w.r.t. G conditionally on $X_1, \ldots, X_N$). Hence, $\mathbb{E}_G Z \geq 0.528|\mathcal{A}_M| \geq 0.0528K$. Since $|Z| \leq K$ a.s., it follows from Paley-Zygmund inequality (see Proposition 3.3.1 in [10]) that

$$\mathbb{P}_G[Z > 0.01K] \geq \frac{(\mathbb{E}_G Z - 0.01K)^2}{\mathbb{E}_G Z^2} \geq (0.0428)^2 = 0.0018.$$

Moreover, it follows from the Borell-TIS inequality (see Theorem 7.1 in [26] or pages 56-57 in [27]) that with probability at least $1 - \exp(-8)$, $\|G\|_2 \leq \mathbb{E}\|G\|_2 + 4\sqrt{\|M\|_{op}}$. Moreover, $\mathbb{E}\|G\|_2 \leq \sqrt{\mathrm{Tr}(M)} \leq 1$ and $\|M\|_{op} \leq \mathrm{Tr}(M) \leq 1$, so $\|G\|_2 \leq 5$ with probability at least $1 - \exp(-8) \geq 0.9996$. Since $0.9996 + 0.0018 > 1$ there exists a vector $G_M \in \mathbb{R}^d$ such that $\|G_M\|_2 \leq 5$ and $\sum_{k\in[K]} I\left(|\langle \bar{X}_k - \mu, G_M\rangle| > 5r\right) > 0.01K$. We recall that this latter result holds when we assume that $|\mathcal{A}_M| \geq 0.1K$.

Next, we denote by Ω_0 the event onto which for all $v \in \mathcal{S}_2^{d-1}$, there are at least $99K/100$ blocks such that $|\langle \bar{X}_k - \mu, v\rangle| \leq r$. We know from Lemma 1 that $\mathbb{P}[\Omega_0] \geq 1 - \exp(-K/180000)$. Let us place ourselves on the event Ω_0 up to the end of the proof. Let $M \in \mathbb{R}^{d\times d}$ be such that $M \succeq 0$ and $\mathrm{Tr}(\Sigma) = 1$ and assume that $|\mathcal{A}_M| \geq 0.1K$. It follows from the first paragraph of the proof that there exists $G_M \in \mathbb{R}^d$ such that $\|G_M\|_2 \leq 5$ and $\sum_{k\in[K]} I\left(|\langle \bar{X}_k - \mu, G_M\rangle| > 5r\right) > 0.01K$. Given that we work on the event Ω_0, we have for $v_M = G_M/\|G_M\|_2$, that for more than $99K/100$ blocks $|\langle \bar{X}_k - \mu, v_M\rangle| \leq r$ and so $|\langle \bar{X}_k - \mu, G_M\rangle| \leq \|G_M\|_2 r \leq 5r$ which contradicts the fact that $\sum_{k\in[K]} I\left(|\langle \bar{X}_k - \mu, G_M\rangle| > 5r\right) > 0.01K$. Therefore, we necessarily have $|\mathcal{A}_M| \leq 0.1K$, which concludes the proof. ∎

Proof of Corollary 1: Let us assume that the event $\mathcal{E}$ holds up to the end of the proof. Let $M \in \mathbb{R}^{d\times d}$ be such that $M \succeq 0$ and $\mathrm{Tr}(\Sigma) = 1$. Let $\mathcal{K}_M = \{k \in [K] : \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 \leq 8r\}$. On the event $\mathcal{E}$, we have $|\mathcal{K}_M| \geq 9K/10$. Let $x_c \in \mathbb{R}^d$. For all $k \in \mathcal{K}_M$, we have $\left\|M^{1/2}(\mu - x_c)\right\|_2 \leq 8r$ and so

$$\begin{aligned}\left\|M^{1/2}(\bar{X}_k - x_c)\right\|_2 &\in \left[\left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 - \left\|M^{1/2}(\mu - x_c)\right\|_2, \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 + \left\|M^{1/2}(\mu - x_c)\right\|_2\right] \\ &\subset \left[\left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 - 8r, \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 + 8r\right].\end{aligned}$$

∎

Let us now turn to the study of the optimization problem (E_{x_c}) on the event $\mathcal{E}$. Like in [8], we denote by OPT_{x_c} the optimal value of (E_{x_c}) and by $h_{x_c} : M \to \min_{w\in\Delta_K} \langle M, \sum_k \omega_k (\bar{X}_k - x_c)(\bar{X}_k - x_c)^\top\rangle$ its objective function to be minimized over the constraint set $\{M \in \mathbb{R}^{d\times d} : M \succeq 0, \mathrm{Tr}(M) = 1\}$.

Remark 2. *For a given M, the optimal choice of $w \in \Delta_K$ in the definition of $h_{x_c}(M)$ is straightforward: one just have to put the maximum possible weight on the $9K/10$ smallest $\langle M, (\bar{X}_k - x_c)(\bar{X}_k - x_c)^\top\rangle, k \in [K]$. Formally, we set $\mathcal{S}_M = \sigma(\{1, 2, \cdots, 9K/10\})$, where σ is a permutation on $[K]$ that arranges the $(\bar{X}_k - x_c)^\top M(\bar{X}_k - x_c), k \in [K]$ in ascending order:*

$$(\bar{X}_{\sigma(1)} - x_c)^\top M(\bar{X}_{\sigma(1)} - x_c) \leq (\bar{X}_{\sigma(2)} - x_c)^\top M(\bar{X}_{\sigma(2)} - x_c) \leq \cdots \leq (\bar{X}_{\sigma(K)} - x_c)^\top M(\bar{X}_{\sigma(K)} - x_c).$$

Then we get $h_{x_c}(M) = (1/|\mathcal{S}_M|)\sum_{k\in\mathcal{S}_M} (\bar{X}_k - x_c)^\top M(\bar{X}_k - x_c)$.

The first lemma deals with the optimal value of (E_{x_c}) when the current point x_c is far from μ.

Lemma 2. *On the event $\mathcal{E}$, for all $x_c \in \mathbb{R}^d$, if $\|x_c - \mu\|_2 > 16r$ then*

$$(8/9)(\|x_c - \mu\|_2 - 8r)^2 \leq OPT_{x_c} \leq (\|x_c - \mu\|_2 + 8r)^2.$$

Proof. Let M be a matrix such that $M \succeq 0$ and $\mathrm{Tr}(M) = 1$. Set $\mathcal{K}_M = \{k \in [K] : \left\|M^{1/2}(\bar{X}_k - \mu)\right\|_2 \leq 8r\}$. On the event $\mathcal{E}$, we have $|\mathcal{K}_M| \geq 9K/10$ and it follows from the proof of Corollary 1 that for all $k \in \mathcal{K}_M$ and all $x_c \in \mathbb{R}^d$,

$$\left\|M^{1/2}(\mu - x_c)\right\|_2 - 8r \leq \left\|M^{1/2}(\bar{X}_k - x_c)\right\|_2 \leq \left\|M^{1/2}(\mu - x_c)\right\|_2 + 8r. \tag{6}$$

Then we define a weight vector $\tilde{\omega} \in \Delta_K$ by setting for all $k \in [K]$

$$\tilde{\omega}_k = \begin{cases} 1/|\mathcal{K}_M| & \text{if } k \in \mathcal{K}_M \\ 0 & \text{else.} \end{cases}$$

It follows from the definition of h_{x_c} and (6) that

$$h_{x_c}(M) \leq \sum_{k\in[K]} \tilde{\omega}_k (\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c) = \frac{1}{|\mathcal{K}_M|} \sum_{k\in\mathcal{K}_M} \left\| M^{1/2}(\bar{X}_k - x_c) \right\|_2^2 \leq \left(\left\| M^{1/2}(\mu - x_c) \right\|_2 + 8r \right)^2. \quad (7)$$

Taking the maximum over all $M \in \mathbb{R}^d$ such that $M \succeq 0$ and $\mathrm{Tr}(\Sigma) = 1$ on both side of the latter inequality yields the right-hand side inequality of Lemma 2.

For the left-hand side inequality of Lemma 2, we let $x_c \in \mathbb{R}^d$ be such that $\|x_c - \mu\|_2 > 16r$. Let M be such that $M \succeq 0$ and $\mathrm{Tr}(M) = 1$. We use the notation and observation from Remark 2: we note that $|\mathcal{K}_M \cap \mathcal{S}_M| \geq 8K/10$ so that it follows from Corollary 1 that

$$\begin{aligned} h_{x_c}(M) &= \frac{1}{9K/10} \sum_{k\in\mathcal{S}_M} \left\| M^{1/2}(\bar{X}_k - x_c) \right\|_2^2 \geq \frac{1}{9K/10} \sum_{k\in\mathcal{A}_M\cap\mathcal{S}_M} \left\| M^{1/2}(\bar{X}_k - x_c) \right\|_2^2 \\ &\geq \frac{8K/10}{9K/10} \left(\left\| M^{1/2}(\mu - x_c) \right\|_2 - 8r \right)^2. \end{aligned}$$

Then, taking the maximum over all $M \succeq 0$ such that $\mathrm{Tr}(M) = 1$ on both sides, finishes the proof. ■

Next lemma shows that the top eigenvector of an approximating solution to (E_{x_c}) is aligned with the best possible descent direction $(\mu - x_c)/\|\mu - x_c\|_2$. It is taken from the proof of Lemma 3.3 in [8]. We reproduce here a short proof for completeness.

Proposition 2. *On the event $\mathcal{E}$, if M is a matrix such that $M \succeq 0$, $\mathrm{Tr}(M) = 1$ and $h_{x_c}(M) \geq (\beta \|x_c - \mu\|_2 + 8r)^2$ for some $1/\sqrt{2} \leq \beta \leq 1$, then any top eigenvector v_1 of M satisfies*

$$\left| \left\langle v_1, \frac{x_c - \mu}{\|x_c - \mu\|_2} \right\rangle \right| > \sqrt{2\beta^2 - 1}.$$

Proof. Let M be a matrix such that $M \succeq 0$, $\mathrm{Tr}(M) = 1$ and $h_{x_c}(M) \geq (\beta \|x_c - \mu\|_2 + 8r)^2$ for some $1/\sqrt{2} \leq \beta \leq 1$. We know from the proof of Lemma 2 (see Equation (7)) that $h_{x_c}(M) \leq \left(\left\| M^{1/2}(\mu - x_c) \right\|_2 + 8r \right)^2$. This implies that $\left\| M^{1/2}(\mu - x_c) \right\|_2^2 \geq \beta^2 \|\mu - x_c\|_2^2$.

Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d \geq 0$ denote the eigenvalues of M and let $v_1, \ldots, v_d$ denote corresponding eigenvectors. The conditions on M implies that $\sum_j \lambda_j = 1$ and $\mathcal{B}_M = (v_1, \ldots, v_d)$ is an orthonormal basis of $\mathbb{R}^d$. We denote $v = (\mu - x_c)/\|\mu - x_c\|_2$. We decompose v in $\mathcal{B}_M$ as $v = \sum_j \alpha_j v_j$ with $\sum_j \alpha_j^2 = 1$. Using this decomposition, we have $v^\top M v = \sum_j \lambda_j \alpha_j^2$. We have $\lambda_1 = \lambda_1 \sum_j \alpha_j^2 \geq \sum_j \lambda_j \alpha_j^2 \geq \beta^2$, so $\lambda_1 \geq \beta^2$. Moreover, since $\sum_j \lambda_j = 1$, we have $\beta^2 \sum_j \alpha_j^2 \leq \sum_j \lambda_j \alpha_j^2 \leq \lambda_1 \alpha_1^2 + (1 - \lambda_1)(1 - \alpha_1^2) \leq \alpha_1^2 + (1 - \beta^2) \sum_j \alpha_j^2$, so we have $\alpha_1^2 \geq (2\beta^2 - 1)$. As we know that $\alpha_1 = \langle v_1, v \rangle$, we get the result. ■

Proposition 2 is the first tool we need to construct a descent algorithm since it provides a descent/ascent direction (depending on the sign of the top eigenvector of an approximate solution to (E_{x_c})). It remains to specify three other quantities to fully characterize our algorithm: a starting point, a step size and a stopping criteria. We start with the starting point. Here we simply use the coordinate-wise median-of-means. The following statistical guarantee on the coordinate-wise median-of-means is known or folklore but we want to put forward that in our case it holds on the event $\mathcal{E}$. This again shows that $\mathcal{E}$ is the only event we need to fully analyze all the building blocks of our algorithm. We recall that the coordinate-wise median-of-means is the estimator $\hat{\mu}^{(0)} \in \mathbb{R}^d$ whose coordinates are for all $j \in [d]$, $\hat{\mu}_j^{(0)} = \mathrm{med}(\bar{X}_{k,j} : k \in [K])$ where $\bar{X}_{k,j}$ is the j-th coordinate of the block mean $\bar{X}_k$ for all $k \in [K]$.

Proposition 3. *On the even $\mathcal{E}$, we have $\left\| \hat{\mu}^{(0)} - \mu \right\|_2 \leq 8\sqrt{d}r$.*

Proof. Let us place ourselves on the event $\mathcal{E}$ during all the proof. For all direction, $v \in \mathcal{S}_2^{d-1}$, there are at least $9K/10$ blocks k such that $|\langle \bar{X}_k - \mu, v \rangle| \leq 8r$. In particular, for all $j \in [d]$, $|\langle \bar{X}_k - \mu, e_j \rangle| \leq 8r$ where $(e_1, \ldots, e_d)$ is the canonical basis of $\mathbb{R}^d$. That is for at least $9K/10$ blocks $|\bar{X}_{k,j} - \mu_j| \leq 8r$. In particular, the latter result is true for the median of $\{\bar{X}_{k,j} : k \in [K]\}$ that is for $\hat{\mu}_j^{(0)}$. We therefore have $\left\| \hat{\mu}^{(0)} - \mu \right\|_\infty \leq 8r$ and so $\left\| \hat{\mu}^{(0)} - \mu \right\|_2 \leq 8r\sqrt{d}$. ■

Proposition 3 guarantees that starting from the coordinate-wise Median-of-Means we are off by a $\sqrt{d}$ proportional factor from the optimal rate r. This will play a key role to analyze the number of steps we need to reach μ within the optimal rate r. Indeed, if we prove a geometric decay of the distance to μ along the descent step then only $\log d$ steps (up to a mutliplicative constants) would be enough to reach μ by a distance at most of the order of r.

Let us now specify the step size we use at each iteration. At the current point x_c we compute a top eigenvector v_1 of an approximating solution M to (E_{x_c}) (i.e. M such that $h_{x_c}(M) \geq (\beta \|x_c - \mu\|_2 + 8r)^2$ for some $1/\sqrt{2} \leq \beta \leq 1$). Next iteration is $x_{c+1} = x_c - \theta_c v_1$ where the step size is

$$\theta_c = -\text{Med}\left(\left\langle \bar{X}_k - x_c, v_1 \right\rangle : k \in [K]\right). \tag{8}$$

In particular, since $\theta_c v_1$ does not depend on the sign of v_1 (the product $\theta_c v_1$ is the same if we replace v_1 by $-v_1$), we do not care which top eigenvector of M we choose.

Let us now prove a geometric decay of the algorithm while x_c is far from μ. Again, this result is proved on the event $\mathcal{E}$.

Proposition 4. *On the event $\mathcal{E}$, the following holds. Let $x_c \in \mathbb{R}^d$ (be the current point of the algorithm). Assume that M is an approximating solution of (E_{x_c}): M is such that $h_{x_c}(M) \geq (\beta \|x_c - \mu\|_2 + 8r)^2$ for some $0.78 \leq \beta \leq 1$ and let v_1 be one of its top eigenvector. Then, we have*

$$\|x_{c+1} - \mu\|_2^2 \leq 0.8 \|x_c - \mu\|_2^2 + 64r^2$$

when $x_{c+1} = x_c - \theta_c v_1$ for θ_c defined in (8).

Proof. Let us assume that the event $\mathcal{E}$ holds up to the end of the proof. Let M be an approximating solution to (E_{x_c}) such that $h_{x_c}(M) \geq (\beta \|x_c - \mu\|_2 + 8r)^2$ for some $0.78 \leq \beta \leq 1$ and let v_1 be a top eigenvector of M.

In direction v_1, there are at least $9K/10$ blocks such that $|\langle \bar{X}_k - \mu, v_1 \rangle| \leq 8r$ hence on these blocks we also have

$$|\theta_c - \langle x_c - \mu, v_1 \rangle| = |\text{Med}\left(\langle \mu - \bar{X}_k, v_1 \rangle : k \in [K]\right)| \leq \text{Med}\left(|\langle \mu - \bar{X}_k, v_1 \rangle| : k \in [K]\right) \leq 8r. \tag{9}$$

Let $v = (\mu - x_c)/\|\mu - x_c\|_2$ denote the optimal normalized descent direction. We write $v = \lambda_1 v_1 + \lambda_2 v_1^\perp$ where $v_1^\perp$ is a normalized orthogonal vector to v_1. We have $\lambda_1^2 + \lambda_2^2 = 1$ and it follows from Proposition 2 that $|\lambda_1| = |\langle v_1, v \rangle| > \sqrt{2\beta^2 - 1}$. We conclude that

$$\begin{aligned}\|x_{c+1} - \mu\|_2^2 &= \|x_c - \mu - \theta_c v_1\|_2^2 = \left\|(\langle x_c - \mu, v_1 \rangle - \theta_c) v_1 + \langle x_c - \mu, v_1^\perp \rangle v_1^\perp\right\|_2^2 \\ &= (\langle x_c - \mu, v_1 \rangle - \theta_c)^2 + \langle x_c - \mu, v_1^\perp \rangle^2 \leq (8r)^2 + \lambda_2^2 \|x_c - \mu\|_2^2\end{aligned}$$

As $\lambda_2^2 = 1 - \lambda_1^2 < 2 - 2\beta^2 < 0.8$ we get the result. ∎

We now have almost all the building blocks to fully characterize the algorithm. The last and final step is to find a stopping rule. The idea we use to design such a rule is based on Proposition 4: we know that when the current point x_c is not in a ℓ_2^d-neighborhood of μ with a radius of the order of r then the ℓ_2^d-distance between the next iteration x_{c+1} and μ should be less than $\sqrt{0.81}$ times the ℓ_2^d-distance between x_c and μ. We therefore have a geometric decay of the distance to μ along the iterations until we reach μ in a ℓ_2^d-neighborhood of radius proportional to r. Starting from the coordinate-wise median(-of-means) which is in a $8\sqrt{d}r$ neighborhood of μ, we only have to do $\log(8\sqrt{d})/\log(1/\sqrt{0.81})$ iterations to output a current point which is r-close to μ w.r.t. the ℓ_2^d-norm (see Proposition 3).

We are now in a position to write an "almost final" pseudo-code of our algorithm. In the next section, we will dive a bit deeper in this pseudo-code (and in particular on the covering SDP algorithm used to construct an approximating solution to (E_{x_c})) in order to provide a final pseudo-code together with its total running time.

input : $X_1, \dots, X_N$ and a number K of blocks
output: A robust subgaussian estimator of μ

Construct an equipartition $B_1 \sqcup \cdots \sqcup B_K = \{1, \cdots, N\}$
Construct the K empirical means $\bar{X}_k = (N/K) \sum_{i \in B_k} X_i, k \in [K]$
Compute $\hat{\mu}^{(0)}$ the coordinate-wise median-of-means and put $x_c \leftarrow \hat{\mu}^{(0)}$
for $T = 1, 2, \cdots, \log(8\sqrt{d})/\log(1/\sqrt{0.81})$ **do**
    Compute M_c an approximating solution to (E_{x_c}) such that

$$h_{x_c}(M_c) \geq \left(0.78 \left\|x_c - \mu\right\|_2 + 8r\right)^2$$

    Compute v_1 a top eigenvector of M_c
    Compute a step size $\theta_c = -\operatorname{Med}\left(\left\langle \bar{X}_k - x_c, v_1 \right\rangle : k \in [K]\right)$
    Update $x_c \leftarrow x_c - \theta_c v_1$
end
Return x_c

Algorithm 1: "Almost final" pseudo-code of the robust sub-gaussian estimator of μ

Algorithm 1 is "almost" our final algorithm. There is one last step we need to check carefully: given a current point x_c we need to find a way to construct M_c satisfying "$h_{x_c}(M_c) \geq \left(0.78 \left\|x_c - \mu\right\|_2 + 8r\right)^2$" without knowing r or μ. This is the last issue we need to address in order to explain how step **5** from Algorithm 1 can be realized in a fully data-dependent way in a good time. This issue is answered in the next section together with the computation of its running time.

4 Solving (approximatively) the SDP (E_{x_c})

The aim of this section is to show that, on the event $\mathcal{E}$, it is possible to construct in reasonnable time a matrix M_c such that "$h_{x_c}(M_c) \geq \left(0.78 \left\|x_c - \mu\right\|_2 + 8r\right)^2$" without any extra information than the data. To that end we construct in an efficient way an approximation solution to the optimization problem (E_{x_c}) using covering SDP as in [8]. The main result of this section is the following.

Theorem 3. *Let $u \in \mathbb{N}^*$. On $\mathcal{E}$, for every $x_c \in \mathbb{R}^d$ such that $\left\|x_c - \mu\right\|_2 \geq 800r$, we can either compute, in time $\tilde{\mathcal{O}}(Kud)$, with probability $> 1 - (1/10)^{u+5}/\sqrt{d}$:*

- *A matrix M_c such that*

$$h_{x_c}(M_c) \geq \left(0.78 \left\|x_c - \mu\right\|_2 + 8r\right)^2$$

- *Or directly a subgaussian estimate of μ, using only the block means $\bar{X}_1, \dots, \bar{X}_K$ as inputs.*

Theorem 3 answers the last issue raised at the end of Section 3 and provides the running time for step **5** of Algorithm 1. It therefore concludes the statement that there exists a fully data-driven robust subgaussian algorithm for the estimation of a mean vector under the only Assumption 1 (the total running time of Algorithm 1 is studied in Section 5).

Remark 3. *Theorem 3 states that we either find an approximating solution M_c to (E_{x_c}) or a good estimate of μ (at the current point x_c). As we will see in this section, this second case is degenerate as it is not the typical situation.*

We now turn to the proof of Theorem 3. It is decomposed into several lemmas adapted from techniques developed by [8] to approximately solve the semi-definite positive problem (E_{x_c}) in polynomial time. To that end, we first introduce the following covering SDP

$$\begin{array}{ll} \text{minimize} & \operatorname{Tr}(M') + \left\|y'\right\|_1 \\ \text{subject to} & M' \succeq 0,\ y' \geq 0, \\ & \forall k \in [K],\ \rho(\bar{X}_k - x_c)^\top M'(\bar{X}_k - x_c) + 9K/10\ y'_k \geq 1 \end{array} \tag{C_ρ}$$

where $\rho > 0$ is some parameter that we will show how to fine-tune later. Then, we show that, for a good choice of ρ, we can turn a good approximation solution for (C_ρ) into a good approximation solution for (E_{x_c}).

We note $g(\rho)$ the optimal objective value of (C_ρ). We begin with a first lemma that shows how to link the two optimization problems (E_{x_c}) and (C_ρ). The proof can be found in Lemma 4.2 from [8]. We adapt it here for our purpose.

Lemma 3. *Let $\rho > 0$. From a feasible solution (M', y') for (C_ρ) that achieves $\mathrm{Tr}(M') + ||y'||_1 \leq 1$, we can construct a feasible solution for (E_{x_c}) with objective value $\geq 1/\rho$ (and conversely).*

Proof. We first note that the optimization problem (E_{x_c}) is equivalent to the following one:

$$\begin{array}{ll} \text{maximize} & z - \dfrac{\|y\|_1}{9K/10} \\ \text{subject to} & M \succeq 0,\ \mathrm{Tr}(M) = 1,\ y \geq 0,\ z \geq 0 \\ & \forall k \in [K],\ (\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c) + \ y_k \geq z \end{array} \qquad (\tilde{E}_{x_c})$$

Indeed, for a given $M \succeq 0$ such that $\mathrm{Tr}(M) = 1$, one can notice that the optimal value is achieved in $(\tilde{E}_{x_c})$ for $y_k = \max(0, z - (\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c)), k \in [K]$ and $z = \mathcal{Q}_{9/10}\left((\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c)\right)$ the 9/10-th quantile of $\{(\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c) : k \in [K]\}$, so that $z - \|y\|_1 / (9K/10) = h_{x_c}(M)$ which gives the equivalence between (E_{x_c}) and $(\tilde{E}_{x_c})$.

Then, once a feasible solution (M', y') for (C_ρ) that achieves $\mathrm{Tr}(M') + \|y'\|_1 \leq 1$ is obtained, by taking $M = M'/\mathrm{Tr}(M')$, $z = 1/(\rho\,\mathrm{Tr}(M'))$ and $y = (9K/10)/(\rho\,\mathrm{Tr}(M'))y'$, we get the desired result (and the converse follows from inverting those relations). ∎

From Lemma 3, it is enough to solve (C_ρ) – for a good choice of ρ – to find a good approximating solution for (E_{x_c}). It therefore remains to find such a good ρ. To do so, we rely on the next two lemmas. The first one is adapted from Lemma 4.3 in [8].

Lemma 4. *For every $\rho > 0$ and every $\alpha \in (0, 1)$, $g((1-\alpha)\rho) \geq g(\rho) \geq (1-\alpha) g((1-\alpha)\rho)$.*

Proof. A feasible pair (M', y') for $(C_{(1-\alpha)\rho})$ is feasible for (C_ρ), which gives the first inequality. If (M', y') is a feasible pair for (C_ρ), then $(M'/(1-\alpha), y'/(1-\alpha))$ is a feasible pair for $(C_{(1-\alpha)\rho})$, which gives the second inequality. ∎

It follows from Lemma 4, that g is continuous, non increasing, and (from Lemma 3, using both sides of the implication, we have that $g(\rho) \leq 1$ iff $1/\rho \geq OPT_{x_c}$) that $g(1/OPT_{x_c}) = 1$. So in order to find a good solution, we must find a ρ such that $g(\rho)$ is as close to 1 as possible. Unfortunately, we do not know how to solve (C_ρ) exactly for a given $\rho > 0$, but we can compute efficiently a good approximation (M', y') and a top eigenvector of M' thanks to the following result which can be found in [37] and is detailed in [8] (see Section 4 and Remark 3.4).

Lemma 5. *[[37]] Let $u \geq 1$ be an integer. For every $\rho > 0$ and every fixed $\eta > 0$, we can find with probability $> 1 - (1/10)^{u+10}/d$ a feasible solution to (C_ρ) that is η-close to the optimal, that is to say a feasible pair (M', y') so that $\mathrm{Tr}(M') + \|y'\|_1 \leq (1+\eta) g(\rho)$ in time $\tilde{\mathcal{O}}(uKd)$. Moreover, it is possible to find a top eigenvector of M' in $\tilde{\mathcal{O}}(Kd)$.*

We compute $(u + 3\log(d) + 10)$ times independently the (randomized) algorithm from [37] that has a runtime of $\tilde{\mathcal{O}}(Kd)$ and that outputs an η-close feasible solution with probability 9/10. By taking the largest of the output's objective value, we have an η-close feasible solution with probability $1 - (1/10)^{u+3\log(d)+10}$, in time $\tilde{\mathcal{O}}(uKd)$, proving Lemma 5. Let us call $\mathtt{ALG}_\rho$ the algorithm from Lemma 5, that takes as input $((\bar{X}_k)_{k=1}^K, x_c, \rho, \eta, u)$ and returns a feasible pair (M', y') for (C_ρ) satisfying $\mathrm{Tr}(M') + \|y'\|_1 \leq (1+\eta) g(\rho)$ in $\tilde{\mathcal{O}}(uKd)$, with probability $> 1 - (1/10)^{u+10}/d$. Next, in order to find a good ρ, we have to get some additional information on the function g. We will get it on the event $\mathcal{E}$.

Lemma 6. *On the event $\mathcal{E}$, for all $x_c \in \mathbb{R}^d$, if $\|x_c - \mu\|_2 > 8r$ then*

$$g(\rho) \leq \frac{1}{\rho\ OPT_{x_c}} \left(1 + \rho OPT_{x_c} \left(\frac{9(\|x_c - \mu\|_2 + 8r)^2}{8(\|x_c - \mu\|_2 - 8r)^2} - 1\right)\right).$$

Proof. We use the same notation as in the proof of Lemma 3. For any $\nu > 0$, we can choose a triplet (z, y, M) feasible for $(\tilde{E}_{x_c})$ such that $z - \|y\|_1 / (9K/10) > OPT_{x_c} - \nu$. On the event $\mathcal{E}$, Lemma 2 yields $OPT_{x_c} > (8/9)(\|x_c - \mu\|_2 - 8r)^2$ and we have from Corollary 1 that

$$z = \mathcal{Q}_{9/10}\left((\bar{X}_k - x_c)^\top M (\bar{X}_k - x_c)\right) = \mathcal{Q}_{9/10}\left(\left\|M^{1/2}(\bar{X}_k - x_c)\right\|_2\right) \leq \left(\left\|M^{1/2}(x_c - \mu)\right\|_2 + 8r\right)^2 \leq (\|x_c - \mu\|_2 + 8r)^2$$

because $M \succeq 0$ and $\mathrm{Tr}(M) = 1$. Let $M' = M/(\rho z), y' = y/[z(9K/10)]$. We have

$$g(\rho) \leq \mathrm{Tr}(M') + \|y'\|_1 \leq \frac{1 + \rho \|y\|_1 / (9K/10)}{\rho z}$$
$$< \frac{1 + \rho(z - OPT_{x_c} + \nu)}{\rho z} \leq \frac{1 + \rho\nu + \rho OPT_{x_c} \left(\frac{9(\|x_c - \mu\|_2 + 8r)^2}{8(\|x_c - \mu\|_2 - 8r)^2} - 1 \right)}{\rho(OPT_{x_c} - \nu)}.$$

By taking $\nu \to 0$, we get the result. ■

Proof of Theorem 3. Let us place ourselves on the event $\mathcal{E}$ so that we can apply Lemma 6. Let $x_d \in \mathbb{R}^d$ and assume that $\|x_c - \mu\|_2 > 800r$. It follows from Lemma 6 that $g(\rho) \leq 1/(\rho\, OPT_{x_c}) + 0.171$. Therefore, if we can find a ρ such that $g(\rho) \geq 1 - \epsilon + 0.171$ for some $0 < \epsilon < 1$, then necessarily $1/\rho \geq OPT_{x_c}(1 - \epsilon)$. Let us take $\epsilon = 0.173$, and $\eta = 0.0001$. Then if $\texttt{ALG}_\rho$ returns, a feasible pair (M', y') for (C_ρ) so that $0.9981 \leq \mathrm{Tr}(M') + \|y'\|_1 \leq 1$, then, since $0.9981 > 1.0001 \times 0.998 = (1 + \eta)(1 - \epsilon + 0.171)$ we will know that, with probability $> 1 - (1/10)^{u+10}/d$,

$$(1 + \eta) g(\rho) \geq \mathrm{Tr}(M') + \|y'\|_1 \geq (1 + \eta)(1 - \epsilon + 0.171)$$

hence $1/\rho \geq OPT_{x_c}(1 - \epsilon)$, and by Lemma 3, we can construct a feasible solution M_c for (E_{x_c}) with objective value satisfying $h_{x_c}(M_c) \geq OPT_{x_c}(1 - \epsilon)$. Next, using Lemma 2, we obtain that when $\|x_c - \mu\|_2 \geq 800r$

$$h_{x_c}(M_c) \geq OPT_{x_c}(1 - \epsilon) \geq (1 - \epsilon)(8/9) \left(\|x_c - \mu\|_2 - 8r \right)^2 \geq \left(0.78 \|x_c - \mu\|_2 + 8r \right)^2$$

for $\epsilon = 0.173$, solving step **5** from Algorithm 1.

Therefore, it only remains to show how to find a ρ such that $\texttt{ALG}_\rho$ returns a pair (M', y') (feasible for (C_ρ)) satisfying $0.9981 \leq \mathrm{Tr}(M') + \|y'\|_1 \leq 1$. We do it first by assuming that we have access to an initial ρ_0 such that $\texttt{ALG}_{\rho_0}$ returns a feasible pair (M', y') for (C_ρ) (for $\rho = \rho_0$) so that $\mathrm{Tr}(M') + \|y'\|_1 \leq 1$ and to a maximal number T of iterations (we will also see later how to choose such ρ_0 and T). The following algorithm (which is a binary search) taking as input $(\bar{X}_1, \ldots, \bar{X}_K, x_c, \rho_0, u, T)$ returns a feasible pair (M', y') for (C_ρ) so that $0.9981 \leq \mathrm{Tr}(M') + \|y'\|_1 \leq 1$ (when T is large enough). This is simply due to the fact that g is continuous, non increasing, $g(0) = 10/9 > 1$ and $g(\rho) \leq 2/8$ when $\rho \to +\infty$ and $\|x_c - \mu\|_2 > 800r$ (because of Lemma 6). For this to work, we need that for each iteration, $\texttt{ALG}_\rho$ returns a feasible pair (M', y') for (C_ρ) (for $\rho = \rho_0$) so that $\mathrm{Tr}(M') + \|y'\|_1 \leq (1 + 0.0001) g(\rho)$. We will suppose that it is the case for the rest of the proof. By union bound, this happens with probability at least $> 1 - T(1/10)^{u+10}/d$

input : $\bar{X}_1, \ldots, \bar{X}_K$, x_c, ρ_0,u, T
output: A feasible pair (M', y') for (C_ρ) satisfying $0.9981 \leq \mathrm{Tr}(M') + \|y'\|_1 \leq 1$

$\rho_m \leftarrow 0$, $\rho_M \leftarrow \rho_0$, $V \leftarrow \texttt{ALG}_{\rho_0}(u)$, $i \leftarrow 0$
while $V \notin [0, 9981, 1]$ *and* $i < T$ **do**
  if $V < 0,9981$ **then**
    $\rho_M \leftarrow (\rho_M + \rho_m)/2$
  end
  else
    $\rho_m \leftarrow (\rho_M + \rho_m)/2$
  end
  $V \leftarrow objective(\texttt{ALG}_{\frac{\rho_m + \rho_M}{2}}(u))$, $i \leftarrow i + 1$
end
Return $\texttt{ALG}_{\frac{\rho_m + \rho_M}{2}}(u)$

Algorithm 2: The `BinarySearch` algorithm to find a ρ so that $\texttt{ALG}_\rho$ returns a pair (M', y') (feasible for (C_ρ)) satisfying $0.9981 \leq \mathrm{Tr}(M') + \|y'\|_1 \leq 1$.

If we can find a ρ_0 (such that $\texttt{ALG}_{\rho_0}$ returns a feasible pair (M', y') for (C_ρ) so that $\mathrm{Tr}(M') + \|y'\|_1 \leq 1$) and a large enough number of iterations T in `BinarySerach`, Algorithm 2 returns a feasible pair (M', y') for (C_ρ) from which we can construct an approximating solution M_c for (E_{x_c}) with objective value $h_{x_c}(M_c)$ larger than $\left(0.78 \|x_c - \mu\|_2 + 8r \right)^2$ whenever $\|x_c - \mu\|_2 \geq 800r$. This is exactly what we expect in step **5** of Algorithm 1. Next, the last and final step that remains to be explained is to show how one can get such a ρ_0 and T using only the block means $(\bar{X}_k)_{k=1}^K$ in $\tilde{\mathcal{O}}(Nd + uKd)$.

Let us consider $\hat{\mu}^{(0)}$ the coordinate-wise median(-of-means) and let us define $\delta = \text{Med}(\left\|\bar{X}_k - \hat{\mu}^{(0)}\right\|_2 : k \in [K])$ – both quantities can be computed in $\tilde{\mathcal{O}}(Kd)$. On the event $\mathcal{E}$, it follows from Corollary 1 (for $M = I_d/d$) and Proposition 3 that $\delta \leq 16\sqrt{d} \times r$. So if one takes $\rho_0 = d/\delta^2 \geq 1/[(16)^2 r^2]$, and if $\|x_c - \mu\|_2 > 800r$, Lemma 2 and Lemma 6 guarantee that $OPT_{x_c} \geq (8/9)\left(\|x_c - \mu\|_2 - 8r\right)^2 \geq (8/9)(792)^2 r^2$ and so

$$g(\rho_0) \leq \frac{1}{\rho\, OPT_{x_c}} + 0.171 \leq \frac{16^2}{(8/9)(792)^2} + 0.171 < 0.18$$

so $\texttt{ALG}_{\rho_0} \leq (1+\eta)g(\rho) < 1.0001 \times 0.18 < 1$ (for the same choice of $\eta = 0.0001$).

Now we tackle the question of the number T of iterations, which is crucial for the runtime. We know from Lemma 4 and Lemma 6 that the interval I of all ρ's such that $0.9981 \leq \textit{objective}(\texttt{ALG}_\rho) \leq 1$ is at least of size $0.001/OPT_{x_c}$ when $\|x_c - \mu\|_2 > 800r$. Indeed, since $g(\rho) \leq \textit{objective}(\texttt{ALG}_\rho) \leq (1+\eta)g(\rho)$, if ρ is such that $0.9981 \leq g(\rho) \leq 1/(1+\eta)$ then $0.9981 \leq \textit{objective}(\texttt{ALG}_\rho) \leq 1$. Now, if we let $\rho_1 > 0$ and $0 < \alpha < 1$ be such that $g(\rho_1) = 0.9981$ and $g((1-\alpha)\rho_1) = 1/(1+\eta)$ the interval I is at least of size $\alpha\rho_1$. Moreover, from Lemma 4 we have $1/(1+\eta) \leq g((1-\alpha)\rho_1) \leq g(\rho_1)/(1-\alpha)$ and so $0.9981 = g(\rho_1) \geq (1-\alpha)/(1+\eta)$, i.e. $\alpha \geq 1 - 0.9981(1+\eta) > 0.001$. Finally, since $g(\rho_1) \leq 1$, $g(1/OPT_{x_c}) = 1$ and g is non-increasing, we conclude that $\rho_1 \geq 1/OPT_{x_c}$ and so the length of I is at least $\alpha\rho_1 \geq 0.001/OPT_{x_c}$.

So, in the case where $\|x_c - \mu\|_2 > 800r$, $\log_2(\rho_0 \times OPT_{x_c}/0.001)$ iterations are enough to insure that `BinarySearch` outputs (M', y') (from $\texttt{ALG}_\rho$ for a well-chosen ρ) feasible for (C_ρ) and such that $0.9981 \leq \text{Tr}(M') + \|y'\|_1 \leq 1$. Moreover, on the event $\mathcal{E}$ it is possible to show that for all iterations x_c along the algorithm we have $\|x_c - \mu\|_2 < C\sqrt{d}r$ for a constant $C \leq 800$ (we may take that as an induction hypothesis for the firsts iterates x_c, and the proof of Theorem 2 below in Section 5 shows that it will still holds for x_{c+1}). So if $\delta > r/d$ then $\rho_0 < d^3/r^2$, and since $OPT_{x_c} < (C^2 d + 8)r^2$ (this follows from Lemma 2), the binary search ends in time $T = \log_2(\tilde{C}d^4)$ with $\tilde{C} < 10^6$.

Thus, if the binary search has not ended in that time, we have either $\delta < r/d$ (which is a degenerate case) or $\|x_c - \mu\|_2 < 800r$ (or both). If $\|x_c - \mu\|_2 > 800r$ and $\delta < r/d$, then, taking $\rho_1 = 1/(d\delta)^2$, we have, by Lemma 6, $\texttt{ALG}_{\rho_1} < 1/2$. So, if we can not end our binary search in time $\log_2(\tilde{C}d^4)$, we compute $\texttt{ALG}_{1/(d\delta)^2}$: if this gives something smaller than 1, that means that $1/(d\delta)^2 > 1/OPT_{x_c} \Rightarrow \delta < \sqrt{(C^2 d + 8)}r/d < (C+1)r/\sqrt{d}$. We notice that on $\mathcal{E}$, $\left\|\hat{\mu}^{(0)} - \mu\right\|_2 < \delta + 8r$, so if $\texttt{ALG}_{1/(d\delta)^2} < 1$, then $\hat{\mu}^{(0)}$ is a good estimate for μ. If on the contrary we have $\texttt{ALG}_{\rho_1} > 1$, it means that $\|x_c - \mu\|_2 < 800r$, so we stop the algorithm and return x_c.

Let us write now in pseudo-code the procedure we just described. This is an algorithm, named `SolveSDP`, running in $\tilde{\mathcal{O}}(Kud)$ which takes as inputs $\bar{X}_1, \ldots, \bar{X}_K$, x_c, u and which outputs, on the event $\mathcal{E}$, with probability $> 1 - \log(\tilde{C}d^4)(1/10)^{u+10}/d$, for every $x_c \in \mathbb{R}^d$ such that $\|x_c - \mu\|_2 \geq 800r$ either a matrix M_c such that

$$h_{x_c}(M_c) \geq \left(0.78\,\|x_c - \mu\|_2 + 8r\right)^2$$

or a subgaussian estimate of μ. It therefore describes step **5** from Algorithm 1.

input : $\bar{X}_1, \ldots, \bar{X}_K$, x_c and u
output: A feasible solution for (E_{x_c})

Compute $\hat{\mu}^{(0)}$, compute δ
$T \leftarrow \log(\tilde{C}d^4)$, $\rho_0 \leftarrow d/\delta^2$
$(M', y') \leftarrow \text{BinarySearch}(T, \rho_0)$
if $\text{Tr}(M') + ||y||_1 \in [0, 9981, 1]$ **then**
  $M \leftarrow M'/\text{Tr}(M')$
  Return (True, M)
end
else
  if $\textit{ALG}_{1/(d\delta)^2} < 1$ **then**
    Return (False, $\hat{\mu}^{(0)}$)
  end
  else
    Return (False, x_c)
  end
end

Algorithm 3: SolveSDP

Remark 4. *[Two advantages of block means] During the whole algorithm, we solve the program (C_ρ) up to a factor $(1+\eta)$ where η is* fixed *(here we take it equal to 0.0001). This differs crucially from the work of [8] where η depends on the fraction of outliers, which decreases the performance of the algorithm in Lemma 5, the true runnnig time being $\tilde{\mathcal{O}}(Kd/Poly(\eta))$. This is another advantages of using the mean blocks instead of the data themselves. Indeed, using blocks of data, we work with a constant fraction of corrupted blocks (we took it equal to 1/10), therefore the approximation parameter used to approximately solved (C_ρ) can be taken equal to a constant (we took it equal to $\eta = 0.0001$) unlike [8] where η depends on $\epsilon = |\mathcal{O}|/N$. Taking the block means has therefore two advantages: a stochastic one, which is to exhibit a subgaussian behavior for $9K/10$ blocks even under a L_2-moment assumption and a computational one, which is to make the proportion of corrupted blocks constant.*

5 The final algorithm and its computational cost: proof of Theorem 2.

We are now in a position to fully describe our robust subgaussian descent algorithm running in $\tilde{\mathcal{O}}(Nd + uKd)$. One may check that its construction is fully data-dependent, in particular, we do not need to know the value of r or the proportion of outliers.

input : $X_1, \dots, X_N$ and $K \in [N]$ and $u \in \mathbb{N}^*$
output: A robust subgaussian estimator of μ

Construct an equipartition $B_1 \sqcup \cdots \sqcup B_K = \{1, \cdots, N\}$
Construct the K empirical means $\bar{X}_k = (N/K)\sum_{i\in B_k} X_i, k \in [K]$
Compute $\hat{\mu}^{(0)}$ the coordinate-wise median
$x_c \leftarrow \hat{\mu}^{(0)}$, Bool $\leftarrow$ True, $T \leftarrow 0$
while *Bool and* $T < \log(8\sqrt{d})/\log(1/0.81)$ **do**
    Bool, $A \leftarrow$SolveSDP$(\bar{X}_1, \dots, \bar{X}_K, x_c)$
    if *Bool* **then**
        $M \leftarrow A$
        Compute v_1 a top eigenvector of M_c
        Compute a step size $\theta_c = -\operatorname{Med}\left(\langle \bar{X}_k - x_c, v_1\rangle : k \in [K]\right)$
        Update $x_c \leftarrow x_c - \theta_c v_1$
        $T \leftarrow T + 1$
    end
    else
        $x_c \leftarrow A$
    end
end
Return x_c

Algorithm 4: Final Algorithm: covSDPofMeans

Proof of Theorem 2. From Theorem 3, we know that on $\mathcal{E}$, when, $\|x_c - \mu\|_2 > 800r$, we get, with probability $> 1 - (1/10)^{u+5}/\sqrt{d}$, an M_c so that $h_{x_c}(M_c) \geq (0.8\|x_c - \mu\|_2 + 8r)^2$ (or directly a subgaussian estimate, in which case our work is done). Proposition 4, states that in that case $\|x_{c+1} - \mu\|_2^2 \leq 0.8\|x_c - \mu\|_2^2 + 64r^2 \leq 0.81\|x_c - \mu\|_2^2$. So we have a geometric decays and Proposition 3 guarantees that our starting point is at most $8\sqrt{d}r$ far away from the mean so that in at most $\log(8\sqrt{d})/\log(1/0.81))$ steps the algorithm outputs its current point which is r-close to μ, with probability $> 1 - (1/10)^{u+5}\log(8\sqrt{d})/(\log(1/0.81))\sqrt{d}) > 1 - (1/10)^u$ (by union bound).

The last thing to do is to control what happens when $\|x_c - \mu\|_2 < 800r$. Then, we have no guarantees on v_1, but using the similar argument as in the proof of Proposition 4 we know that

$$|\theta_c - \langle x_c - \mu, v_1\rangle| = |\operatorname{Med}\left(\langle \mu - \bar{X}_k, v_1\rangle : k \in [K]\right)| \leq \operatorname{Med}\left(|\langle \mu - \bar{X}_k, v_1\rangle| : k \in [K]\right) \leq 8r \tag{10}$$

and (for some $v_1^\perp$ a normalized orthogonal vector to v_1)

$$\begin{aligned}\|x_{c+1} - \mu\|_2^2 &= \|x_c - \mu - \theta_c v_1\|_2^2 = \left\|(\langle x_c - \mu, v_1\rangle - \theta_c)v_1 + \langle x_c - \mu, v_1^\perp\rangle v_1^\perp\right\|_2^2 \\ &= (\langle x_c - \mu, v_1\rangle - \theta_c)^2 + \langle x_c - \mu, v_1^\perp\rangle^2 \leq (8r)^2 + \|x_c - \mu\|_2^2 .\end{aligned}$$

Hence, $\|x_{c+1}-\mu\|_2 \leq (8r)+\|x_c-\mu\|_2$. Therefore, in the worst case scenario where $\|x_c-\mu\|_2 > 800r$ at the last iteration, the algorithm outputs the next iteration $\hat{\mu}_K = x_{c+1}$ so that $\|\hat{\mu}_K-\mu\|_2 \leq 808r$.

We end this proof with the computation of the running time of Algorithm 4. We detail the computation cost for each line of Algorithm 4: line **1** cost N, line **2** costs Nd, line **3** costs $\mathcal{O}(dK\log(K))$. The while loop in line **5** is running at least $\log d$ times (up to constant) so that the computational cost of all remaining lines of Algorithm 4 are at worst to be multiplied by $\log d$. Line **6** costs $\log(\tilde{C}d^4)$ steps, each of cost $\tilde{\mathcal{O}}(Kud)$ (that comes from Lemma 5). Line **9** can be computed in $\tilde{\mathcal{O}}(Nd)$ thanks to Lemma 5. Finally, line **10** costs $\mathcal{O}(Kd)$. Other lines take time at most d. We thus recover the running time announced in Theorem 2. ■

6 Adaptive choice of K

Given a number of blocks $K \in \{1,\dots,N\}$, a parameter $u \geq 1$ (so that the covering SDPs from [37] (used in Lemma 5) is ran $u+3\log d+10$ times) and the dataset $\{X_1,\dots,X_N\}$, Algorithm 4 returns a vector $\hat{\mu}_K$ in $\mathbb{R}^d$ and Theorem 2 insures that $\hat{\mu}_K$ estimates the true mean μ at the subgaussian rate (1) with large probability as long as $K \geq 300|\mathcal{O}|$. As a consequence, we have certified statistical guarantees for μ_K only when some a priori knowledge on the number $|\mathcal{O}|$ of outliers is provided (such as "the corruption of this database is less than 5%") or if we choose K like N- but, in this later case the rate (1) may be too pessimistic. The aim of this section is to overcome this issue by constructing a procedure which can automatically adapt to the number of outliers. The resulting procedure satisfies the same statistical bounds as μ_K for all $K \geq 300|\mathcal{O}|$ without knowing $|\mathcal{O}|$ (up to constants).

The adaptation method we use is based on the Lepski method [28, 29] which is another tool used by the "MOM community" since [31]. The price we pay for this adaptation is the a priori knowledge of the rate (1) for all K which means that we know in advance $\mathrm{Tr}(\Sigma)$ and $\|\Sigma\|_{op}$ – this is for instance the case when it is known that Σ is the identity matrix I_d. Of course, one can design robust estimators for $\mathrm{Tr}(\Sigma)$ (see [11]) and $\|\Sigma\|_{op}$ but this requires stronger assumptions that we want to avoid at this stage.

Lepski's method proceeds as follows. We set for all $K \in \{1,\dots,N\}$ and all $j \in \{0,1,\dots,\log_2 N\}$

$$r_K^* = 808\left(1200\sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}}+\sqrt{\frac{1200\,\|\Sigma\|_{op}K}{N}}\right) \text{ and } r^{(j)} = r^*_{\lceil N/2^j\rceil}$$

the rate of convergence from Theorem 2. For a given parameter $u_j \in \mathbb{N}^*$, we construct from Algorithm 4

$$\hat{\mu}^{(j)} \leftarrow covSDPofMeans(X_1,\dots,X_N, K=\lceil N/2^j\rceil, u=u_j). \tag{11}$$

Classical Lepski's method considers the largest J such that $\cap_{j=0}^J B_2(\hat{\mu}^{(j)}, r^{(j)})$ is none empty and then take any point $\hat{\mu}$ in this none empty intersection. Standard analysis of Lepski's method shows that $\hat{\mu}$ estimates μ at the rate r_K^* (up to an absolute constant) simultaneously for all $K \in \{300|\mathcal{O}|,\dots,N\}$ without knowing $|\mathcal{O}|$. Given that checking that the intersection of several ℓ_2^d-balls may not be straigtforward, we use a slightly modified version of Lepski's method as described in the following algorithm.

input : $X_1,\dots,X_N$ and $\{u_j : j=0,1,2,\dots,\log_2 N\} \subset \mathbb{N}^*$
output: A robust subgaussian estimator of μ with adaptive choice of K
init : $J=0$ and $\hat{\mu}^{(0)} = covSDPofMeans(X_1,\dots,X_N, K=N, u=u_0)$

1 **while** $\left\|\hat{\mu}^{(J)}-\hat{\mu}^{(j)}\right\|_2 \leq r^{(J)}+r^{(j)}, j=J-1,J-2,\dots,0$ **do**
2 $\quad J \leftarrow J+1$
3 $\quad \hat{\mu}^{(J)} \leftarrow covSDPofMeans(X_1,\dots,X_N, K=\lceil N/2^J\rceil, u=u_J)$
4 **end**
5 **Return** $\hat{\mu}^{(J)}$

Algorithm 5: Adaptive choice of K in covSDPofMeans

Unlike for the traditional Lepski's method we check that $\hat{\mu}^{(J)}$ is in $\cap_{j=0}^{J-1} B_2(\hat{\mu}^{(j)}, r^{(J)}+r^{(j)})$ instead of checking that $\cap_{j=0}^J B_2(\hat{\mu}^{(j)}, r^{(j)})$ is none empty – this simplifies the adaptation step. It is also possible to speed up the whole procedure by constructing iteratively the block means. Indeed, given that we consider a dyadic grid for K, i.e. $K \in \{N, \lceil N/2\rceil, \lceil N/4\rceil, \dots\}$, for all $j \in \mathbb{N}$, we can construct the block means $\{\bar{X}_k^{(j+1)}, k=1,\dots,\lceil N/2^{j+1}\rceil\}$ at step

$K = \lceil N/2^{j+1} \rceil$ using the block means from the previous step $K = \lceil N/2^j \rceil$ by simply averaging two successive block means: $\bar{X}_k^{(j+1)} \leftarrow (\bar{X}_{2k}^{(j)} + \bar{X}_{2k+1}^{(j)})/2$.

Let us now turn to the statistical analysis of the output $\hat{\mu}^{(\hat{J})}$ from Algorithm 5 where

$$\hat{J} = \max\left(J \in \{0, 1, \dots, \log_2 N\} : \hat{\mu}^{(J)} \in \cap_{j=0}^{J-1} B_2(\hat{\mu}^{(j)}, r^{(J)} + r^{(j)})\right).$$

Theorem 4. *Let $\{u_j : j = 0, 1, 2, \dots, \log_2 N\} \subset \mathbb{N}^*$ be the family of parameters used to construct the family of estimators $\{\hat{\mu}^{(j)}, j = 0, 1, \dots\}$ in Algorithm 5 (see also (11)). For all $K \in \{600|\mathcal{O}|, \dots, N\}$, with probability at least*

$$1 - 2\exp(-K/360000) - \sum_{j=0}^{\log_2(N/(K-1))} (1/10)^{u_j} \tag{12}$$

the output $\hat{\mu}^{(\hat{J})}$ of Algorithm 5 is such that $\left\|\hat{\mu}^{(\hat{J})} - \mu\right\|_2 \leq 3r_K^*$.

Proof. For all $j \in \{0, 1, \dots, \log_2 N\}$ denote by $\mathcal{E}_j$ the event onto which Theorem 2 is valid for $K = \lceil N/2^j \rceil$ and for $u = u_j$: that is on $\mathcal{E}_j$, if $\lceil N/2^j \rceil \geq 300|\mathcal{O}|$, $\left\|\hat{\mu}^{(j)} - \mu\right\|_2 \leq r^{(j)}$ and $\mathbb{P}[\mathcal{E}_j] \geq 1 - \exp(-\lceil N/2^j \rceil/180000) - (1/10)^{u_j}$. Let $K \in \{600|\mathcal{O}|, \dots, N\}$ and $J \in \{0, 1, \dots, \log_2 N\}$ be such that $\lceil N/2^J \rceil \leq K < \lceil N/2^{J-1} \rceil$. On the event $\cap_{j=0}^J \mathcal{E}_j$, we have $\left\|\hat{\mu}^{(j)} - \mu\right\|_2 \leq r^{(j)}$ for all $j = 0, 1, \dots, J$, in particular, for all $j = 0, 1, \dots, J-1$, $\left\|\hat{\mu}^{(J)} - \hat{\mu}^{(j)}\right\|_2 \leq r^{(J)} + r^{(j)}$ and so $\hat{\mu}^{(J)} \in \cap_{j=0}^{J-1} B_2(\hat{\mu}^{(j)}, r^{(J)} + r^{(j)})$. As a consequence $\hat{J} \geq J$ therefore $\left\|\hat{\mu}^{(\hat{J})} - \hat{\mu}^{(J)}\right\|_2 \leq r^{(\hat{J})} + r^{(J)} \leq 2r^{(J)} \leq 2r_K^*$. Finally, we have

$$\mathbb{P}[\cap_{j=0}^J \mathcal{E}_j] \geq 1 - \sum_{j=0}^J \exp(-\lceil N/2^j \rceil/180000) - (1/10)^{u_j} \geq 1 - 2\exp(-K/360000) - \sum_{j=0}^{\log_2(N/(K-1))} (1/10)^{u_j}.$$

■

We can see in Algorithm 5 that $\hat{\mu}^{(\hat{J})}$ does not use any information on the number of outliers $|\mathcal{O}|$ for its construction but it can still estimate μ at the optimal rate r_K^* for all deviation parameters K in $\{600|\mathcal{O}|, \dots, N\}$. The maximum total running time of Algorithm 5 is achieved when $\hat{J} = \log_2 N$; in that case, it is at most $\tilde{\mathcal{O}}(Nd + \sum_{j=0}^{\log_2 N} \lceil N/2^j \rceil u_j d)$. In particular, if one chooses $u_j = 2^j$ for all $j = 0, 1, \dots, \log_2 N$ then the total running time for the construction of $\hat{\mu}^{(\hat{J})}$ is nearly-linear $\tilde{\mathcal{O}}(Nd)$. For this choice of u_j, the probability deviation in (12) is constant and so one should choose the smallest possible K allowed in Theorem 4, that is $K = 600|\mathcal{O}|$. Let us write formally this result.

Corollary 2. *If one takes $u_j = 2^j$ for all $j = 0, 1, \dots, \log_2 N$ in Algorithm 5 then, in nearly-linear time $\tilde{\mathcal{O}}(Nd)$, with probability at least $1 - 2\exp(-600|\mathcal{O}|/360000) - 1/11$, the output $\hat{\mu}^{(\hat{J})}$ from Algorithm 5 satisfies*

$$\left\|\hat{\mu}^{(\hat{J})} - \mu\right\|_2 \leq 2r_{600|\mathcal{O}|}^* = 1616\left(1200\sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + 850\sqrt{\frac{\|\Sigma\|_{op}|\mathcal{O}|}{N}}\right). \tag{13}$$

In particular, considering the setup from Theorem 1, if $|\mathcal{O}| = \epsilon N$ for some $\epsilon \leq 1/600$ then the rate achieved by $\hat{\mu}^{(\hat{J})}$ in Corollary 2 is of the order of

$$\sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\|\Sigma\|_{op}\epsilon}$$

which is like $\sqrt{\|\Sigma\|_{op}\epsilon}$ when $N \geq (\mathrm{Tr}(\Sigma)/\|\Sigma\|_{op})/\epsilon$. As a consequence, the result from Corollary 2 improves the one from Theorem 1 by removing an extra $\log d$ factor in the sample complexity in the case considered in Theorem 1 that is when $\Sigma \preceq \sigma^2 I_d$. Moreover, Corollary 2 also shows that the sample complexity depends on the *effective rank* $\mathrm{Tr}(\Sigma)/\|\Sigma\|_{op}$ of Σ. This ratio can be much smaller than d if the spectrum of Σ decays sufficiently fast. Finally, Corollary 2 also covers the case where the sample size N is less than the sample complexity – that is when $N \leq (\mathrm{Tr}(\Sigma)/\|\Sigma\|_{op})/\epsilon$. In that case, the estimation rate is given by $\sqrt{\mathrm{Tr}(\Sigma)/N}$ which is the complexity coming from the estimation of μ in the none corrupted case. As a consequence, Corollary 2 exhibits a phase transition happening at $N \sim (\mathrm{Tr}(\Sigma)/\|\Sigma\|_{op})/\epsilon$ above which corruption is the main source of estimation mistakes and below which corruption does not play any role.

Corollary 2 covers the case where $\hat{\mu}^{(\hat{J})}$ is computed in nearly-linear time and with statistical guarantees happening with constant probability. In the following final result, we show that $\hat{\mu}^{(\hat{J})}$ can estimate μ at the optimal rate r_K^* for

all $K \geq 600|\mathcal{O}|$ with a subgaussian deviation $1 - 2\exp(-K/360000)$ if we perform more iterations u_j of the covering SDP from Lemma 5. The price we pay for this subgaussian behavior of $\hat{\mu}^{(\hat{J})}$ is on the total running time which goes from nearly-linear time $\tilde{\mathcal{O}}(Nd)$ to $\tilde{\mathcal{O}}(N^2 d)$ by taking $u_j = \lceil N/2^j \rceil$ for $j = 0, 1, \ldots, \log_2 N$ ($u_j = N$ would do as well). We write formally this statement in the next corollary which follows directly from Theorem 4.

Corollary 3. *If one takes $u_j = \lceil N/2^j \rceil$ for all $j = 0, 1, \ldots, \log_2 N$ in Algorithm 5 then, in time $\tilde{\mathcal{O}}(N^2 d)$, for all $K \geq 600|\mathcal{O}|$, with probability at least $1 - 4\exp(-K/360000)$, the output $\hat{\mu}^{(\hat{J})}$ from Algorithm 5 satisfies*

$$\left\| \hat{\mu}^{(\hat{J})} - \mu \right\|_2 \leq 2r_K^* = 1616 \left(1200 \sqrt{\frac{\mathrm{Tr}(\Sigma)}{N}} + \sqrt{\frac{1200 \left\| \Sigma \right\|_{op} K}{N}} \right). \tag{14}$$

As a consequence $\hat{\mu}^{(\hat{J})}$ is a subgaussian estimator of μ for all range of K from $600|\mathcal{O}|$ to N which can handle up to $|\mathcal{O}|$ outliers in the database (even when $|\mathcal{O}| \sim N$) and that can be constructed in time $\tilde{\mathcal{O}}(N^2 d)$. It does not require any knowledge on $|\mathcal{O}|$ for its construction.

Acknowlegements: We would like to thank Yeshwanth Cherapanamjeri, Ilias Diakonikolas, Yihe Dong, Nicolas Flammarion, Sam Hopkins and Jerry Li for helpful comments on our work.

References

[1] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. System Sci.*, 58(1, part 2):137–147, 1999. Twenty-eighth Annual ACM Symposium on the Theory of Computing (Philadelphia, PA, 1996).

[2] Lucien Birgé. Stabilité et instabilité du risque minimax pour des variables indépendantes équidistribuées. *Ann. Inst. H. Poincaré Probab. Statist.*, 20(3):201–223, 1984.

[3] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities*. Oxford University Press, Oxford, 2013. A nonasymptotic theory of independence, With a foreword by Michel Ledoux.

[4] Sébastien Bubeck, Nicolò Cesa-Bianchi, and Gábor Lugosi. Bandits with heavy tail. *IEEE Trans. Inform. Theory*, 59(11):7711–7717, 2013.

[5] Olivier Catoni. Challenging the empirical mean and empirical variance: A deviation study. *Annales de l'I.H.P. Probabilités et statistiques*, 48(4):1148–1185, 2012.

[6] Olivier Catoni and Ilaria Giulini. Dimension-free pac-bayesian bounds for matrices, vectors, and linear least squares regression. Technical report, CNRS and LSPM, 2017.

[7] Mengjie Chen, Chao Gao, and Zhao Ren. Robust covariance and scatter matrix estimation under Huber's contamination model. *Ann. Statist.*, 46(5):1932–1960, 2018.

[8] Yu Cheng, Ilias Diakonikolas, and Rong Ge. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2755–2771. SIAM, Philadelphia, PA, 2019.

[9] Yeshwanth Cherapanamjeri, Nicolas Flammarion, and Peter L. Bartlett. Fast mean estimation with sub-gaussian rates, 2019.

[10] Víctor H. de la Peña and Evarist Giné. *Decoupling*. Probability and its Applications (New York). Springer-Verlag, New York, 1999. From dependence to independence, Randomly stopped processes. U-statistics and processes. Martingales and beyond.

[11] Jules Depersin and Guillaume Lecué. Fast algorithms for robust estimation of a mean vector. 2019.

[12] Luc Devroye, Matthieu Lerasle, Gabor Lugosi, and Roberto I. Oliveira. Sub-Gaussian mean estimators. *Ann. Statist.*, 44(6):2695–2725, 2016.

[13] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust Estimators in High-Dimensions Without the Computational Intractability. *SIAM J. Comput.*, 48(2):742–864, 2019.

[14] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *57th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2016*, pages 655–664. IEEE Computer Soc., Los Alamitos, CA, 2016.

[15] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 655–664. IEEE, 2016.

[16] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robustly learning a gaussian: Getting optimal error, efficiently. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2683–2702. Society for Industrial and Applied Mathematics, 2018.

[17] Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. List-decodable robust mean estimation and learning mixtures of spherical Gaussians. In *STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1047–1060. ACM, New York, 2018.

[18] Ilias Diakonikolas, Weihao Kong, and Alistair Stewart. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2745–2754. SIAM, Philadelphia, PA, 2019.

[19] David L. Donoho and Miriam Gasko. Breakdown properties of location estimates based on halfspace depth and projected outlyingness. *Ann. Statist.*, 20(4):1803–1827, 1992.

[20] Frank R. Hampel. A general qualitative definition of robustness. *Ann. Math. Statist.*, 42:1887–1896, 1971.

[21] Frank R. Hampel. Robust estimation: a condensed partial survey. *Z. Wahrscheinlichkeitstheorie und Verw. Gebiete*, 27:87–104, 1973.

[22] Samuel B Hopkins. Sub-gaussian mean estimation in polynomial time. *arXiv preprint arXiv:1809.07425*, 2018.

[23] Peter J. Huber. Robust estimation of a location parameter. *Ann. Math. Statist.*, 35:73–101, 1964.

[24] Peter J. Huber and Elvezio M. Ronchetti. *Robust statistics*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, NJ, second edition, 2009.

[25] Mark R. Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoret. Comput. Sci.*, 43(2-3):169–188, 1986.

[26] Michel Ledoux. *The concentration of measure phenomenon*, volume 89 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 2001.

[27] Michel Ledoux and Michel Talagrand. *Probability in Banach spaces*. Classics in Mathematics. Springer-Verlag, Berlin, 2011. Isoperimetry and processes, Reprint of the 1991 edition.

[28] O. V. Lepskiĭ. A problem of adaptive estimation in Gaussian white noise. *Teor. Veroyatnost. i Primenen.*, 35(3):459–470, 1990.

[29] O. V. Lepskiĭ. Asymptotically minimax adaptive estimation. I. Upper bounds. Optimally adaptive estimates. *Teor. Veroyatnost. i Primenen.*, 36(4):645–659, 1991.

[30] M. Lerasle and R. Oliveira. Robust empirical mean estimators. Technical report, IMPA and CNRS, 2011.

[31] Gábor Lugosi, Shahar Mendelson, et al. Sub-gaussian estimators of the mean of a random vector. *The Annals of Statistics*, 47(2):783–794, 2019.

[32] Z. Szabo M. Lerasle, T. Matthieu and G. Lecué. Monk – outliers-robust mean embedding estimation by median-of-means. Technical report, CNRS, University of Paris 11, Ecole Polytechnique and CREST, 2017.

[33] S Minsker and N. Strawn. Distributed statistical estimation and rates of convergence in normal approximation. Technical report, arXiv: 1704.02658, 2017.

[34] Stanislav Minsker. Geometric median and robust estimation in banach spaces. *Bernoulli*, 21(4):2308–2335, 2015.

[35] Stanislav Minsker. Sub-Gaussian estimators of the mean of a random matrix with heavy-tailed entries. *Ann. Statist.*, 46(6A):2871–2903, 2018.

[36] A. S. Nemirovsky and D. B. and Yudin. *Problem complexity and method efficiency in optimization*. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York, 1983. Translated from the Russian and with a preface by E. R. Dawson, Wiley-Interscience Series in Discrete Mathematics.

[37] Richard Peng, Kanat Tangwongsan, and Peng Zhang. Faster and simpler width-independent parallel algorithms for positive semidefinite programming, 2012.

[38] Christopher G Small. A survey of multidimensional medians. *International Statistical Review/Revue Internationale de Statistique*, pages 263–277, 1990.

[39] John W. Tukey. A survey of sampling from contaminated distributions. In *Contributions to probability and statistics*, pages 448–485. Stanford Univ. Press, Stanford, Calif., 1960.

[40] John W. Tukey. The future of data analysis. *Ann. Math. Statist.*, 33:1–67, 1962.