Machine Learning (2020) 109:1635–1665
https://doi.org/10.1007/s10994-019-05863-6

Robust classification via MOM minimization

Guillaume Lecué[1] · **Matthieu Lerasle**[1] · **Timlothée Mathieu**[2]

Received: 6 November 2018 / Revised: 19 September 2019 / Accepted: 28 November 2019 /
Published online: 27 April 2020

Abstract
We present an extension of Chervonenkis and Vapnik's classical empirical risk minimization (ERM) where the empirical risk is replaced by a median-of-means (MOM) estimator of the risk. The resulting new estimators are called MOM minimizers. While ERM is sensitive to corruption of the dataset for many classical loss functions used in classification, we show that MOM minimizers behave well in theory, in the sense that it achieves Vapnik's (slow) rates of convergence under weak assumptions: the functions in the hypothesis class are only required to have a finite second moment and some outliers may also have corrupted the dataset. We propose algorithms, inspired by MOM minimizers, which may be interpreted as MOM version of block stochastic gradient descent (BSGD). The key point of these algorithms is that the block of data onto which a descent step is performed is chosen according to its " centrality" among the other blocks. This choice of " descent block" makes these algorithms robust to outliers; also, this is the only extra step added to classical BSGD algorithms. As a consequence, classical BSGD algorithms can be easily turn into robust MOM versions. Moreover, MOM algorithms perform a smart subsampling which may help to reduce substantially time computations and memory resources when applied to non linear algorithms. These empirical performances are illustrated on both simulated and real datasets.

Keywords Robust machine learning · Empirical process · Vapnik · Gradient descent

1 Introduction

The article presents a class of robust (to outliers and heavy-tailed data) estimators and algorithms for the classification problem. Consider the classical binary classification problem, let

Editor: Gabor Lugosi.

✉ Guillaume Lecué
Guillaume.lecue@ensae.fr

Matthieu Lerasle
matthieu.lerasle@ensae.fr

Timlothée Mathieu
timothee.mathieu@u-psud.fr

[1] CREST-ENSAE, IPParis, Palaiseau, France

[2] Université Paris Orsay, Orsay, France

$\mathcal{F}$ denote a class of functions from $\mathcal{X}$ to $\{\pm 1\}$, the empirical risk minimizer (ERM) is defined by

$$\widehat{f}_{\text{ERM}} \in \underset{f \in \mathcal{F}}{\text{argmin}} \frac{1}{N} \sum_{i=1}^{N} I\{Y_i \neq f(X_i)\} \tag{1}$$

where $I\{Y_i \neq f(X_i)\} = 1$ if $Y_i \neq f(X_i)$ and 0 otherwise. In this paper, we are interested in the case where the random variables $f(X_i)$ only satisfy a second moment assumption and where the dataset $\{(X_i, Y_i)_{i\in\{1,\dots,N\}}\}$ may contain outliers. The ERM behaves well under these assumptions (see Theorem 1 below). The reason is that the $0-1$ loss $\ell_f^{0-1}(x, y) = I_{\{y\neq f(x)\}}$ is bounded, which grants concentration no matter the distribution of X and a small number of data cannot really impact the empirical mean performance. However, it is well known that ERM is a theoretical estimators that can only be approximated in most situations by efficient algorithms. Indeed, the minimization problem (1) is NP-hard even for classes $\mathcal{F}$ of half-spaces indicators (Guruswami and Raghavendra 2009; Feldman et al. 2012). One of the most classical way to approximate ERM is to choose a convex relaxation of the problem (1) and design an algorithm solving the associated convex problem. The problem of these approaches in the setting of this paper is that the relaxed criteria are unbounded and therefore way more sensitive to outliers or heavy tailed inputs. This results into poor performance of the algorithms on corrupted and/or heavy-tailed data. Figure 1 illustrates this problem on a toy example where most data would be well separated by a linear classifier like Perceptron (Rosenblatt 1958) or logistic classifier, but some anomalies flaw these algorithms.

The example in Fig. 1 is representative of a general problem that this paper intends to study. Robust learning has received particular attention in recent years by practitioners working on large datasets which are particularly sensitive to data corruption. Challenges recently posted on "kaggle", the most popular data science competition platform, have put forward this topic (see, the 1.5 million dollars problem "Passenger Screening Algorithm Challenge" involves the discovery of terrorist activity from 3D images or the challenge named "NIPS 2017: Defense Against Adversarial Attack" consists in building algorithms robust to adversarial data). Robust algorithms have also been studied theoretically both in statistical and computer science communities. In statistics, robust results usually deal with issues arising when data have heavy-tailed distribution (Lugosi and Mendelson 2017; Minsker 2015; Chen et al. 2018; Fan and Kim 2018). In computer science, most works deal with corrupted datasets,

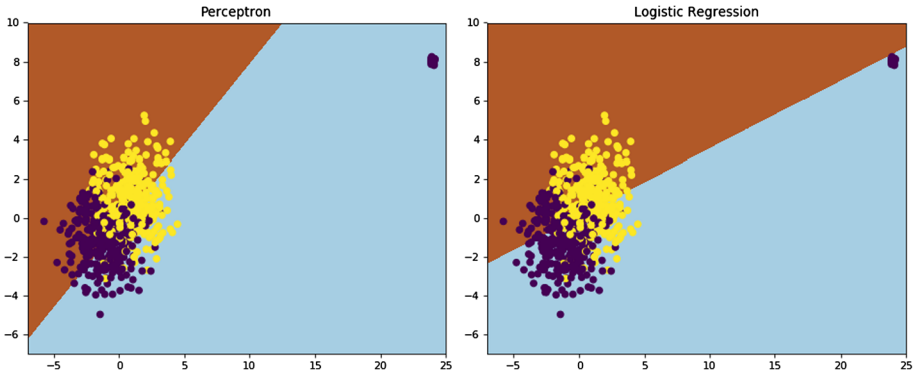


Fig. 1 Scatter plot of the toy dataset, the color of the points gives their class. The background color gives the linear separation provided by the perceptron (left) and the logistic regression (right) trained on this corrupted dataset (Color figure online)

in particular when this corruption arise from adversarial outliers (Diakonikolas et al. 2016, 2017; Cheng et al. 2019). Only few papers consider both problems simultaneously (Lecué and Lerasle 2017, 2019).

In learning theory, most alternatives to ERM manage the problem of outliers and heavy tail distributions for outputs only. These solutions are based on the pioneering work of Tukey (1960, 1962), Huber (1964, 1967) and Hampel (1971, 1974), replacing the square loss by a robust alternative like Huber loss or Tukey's biweight loss. These methods do not allow to treat the case where the inputs are with heavy tails or corrupted, which is a classical problem in robust statistics also known as the "*leverage point problem*", see Huber and Ronchetti (2009).

In this article, we address this question by considering an alternative to M-estimators, called median-of-means (MOM) minimizers. Several estimators based on MOM have recently been proposed in the literature Minsker (2015), Lecué and Lerasle (2017, 2019), Lugosi and Mendelson (2017, 2019a, b) and Mendelson (2017). To our knowledge, these articles use the small ball hypothesis (Koltchinskii and Mendelson 2015; Mendelson 2014) to treat problems of least squares regression or Lipschitzian loss regression. This assumption is restrictive in some classic functional regression frameworks (Saumard 2018; Han and Wellner 2017) or for problems such as the construction of recommendation system where inputs are sampled in the canonical basis and therefore do not satisfy a small ball condition.

We construct a natural estimator based on the MOM principle, which is called MOM minimizer. This estimator is studied here without the small ball hypothesis. Instead, we assume an a priori bound on the L^2-norm of learning functions. We can identify mainly two streams of hypothesis in Learning theory: 1) boundedness with respect to some norm of the class F of functions and the output Y, the typical example is the boundedness in L^∞ assumption or 2) norm equivalence assumption over the class F (or, more precisely, on the shifted class $F - f^* = \{f - f^* : f \in F\}$ where f^* is the oracle in F, i.e. the minimizer of the theoretical risk among the functions in F) and Y, the typical example being the subgaussian assumption, i.e. $\|f - f^*\|_{\psi_2} \leq L \|f - f^*\|_{L^2}, \forall f \in F$ where for $g \in F$ $\|g\|_{\psi_2} = \inf\{t > 0 : \mathbb{E}[\exp(X^2/t^2)] \leq 2\}$. The small ball assumption is a norm equivalence assumption between the L^1 and L^2 norms and is concerned with the second type of assumptions. Our approach here deals with the first type of assumption. As we only assume boundedness in L^2-norm, this can be seen as a significant relaxation upon the L^∞ boundedness assumption. It turns out that, in this relaxed setting, MOM minimizers achieve minimax rates of convergence (Devroye et al. 1997) in the absence of a margin condition (Mammen and Tsybakov 1999) even under a L^∞ assumption.

The estimation of the expectation of a univariate variable by median-of-means (MOM) (Alon et al. 1999; Jerrum et al. 1986; Nemirovsky and Yudin 1983) is done as follows: given a partition of the dataset into blocks of the same size, an empirical mean is constructed on each block and the MOM estimator is obtained by taking the median of these empirical means (see Sect. 2.2 for details). These estimators are naturally resistant to the presence of a few outliers in the dataset: if the number of these outliers does not exceed half the number of blocks, more than half of these blocks are made of "clean" data and the median is a reliable estimator.

On the practical side, we introduce algorithms inspired by the MOM minimizers. In these algorithms, the MOM principle is used within algorithms originally intended for the evaluation of ERM estimators associated to convex loss functions. In Sect. 4, we present a " MOM version" of gradient descent algorithms following this approach. The general principle of this iterative algorithm is as follows: at iteration t, a dataset equipartition $B_1, \ldots, B_K$ is

selected uniformly at random and the most central block B_{med} is determined according to the following formula

$$\sum_{i\in B_{\text{med}}} \ell_{f_t}(X_i, Y_i) = \text{median}\left(\sum_{i\in B_k} \ell_{f_t}(X_i, Y_i) : \quad k = 1, \ldots, K\right) = \text{MOM}_K\left(\ell_{f_t}\right) \qquad (2)$$

where $\ell_{f_t}(X_i, Y_i) = \ell(f_t(X_i), Y_i)$ is the loss of the prediction $f_t(X_i)$ of the label Y_i. Next iteration f_{t+1} is then produced by taking from f_t a step down in the direction opposite to the gradient of $f \to \sum_{i\in B_{\text{med}}} \ell_f(X_i, Y_i)$ at f_t, cf. Algorithm 1. The underlying heuristic is that the data in the selected block B_{med} are safe for estimating the risk of f_t, in the sense that empirical risk $|B_{\text{med}}|^{-1} \sum_{i\in B_{\text{med}}} \ell_{f_t}(X_i, Y_i)$ is a subgaussian estimator of $\mathbb{E}\ell_f(X_i, Y_i)$, cf. Devroye et al. (2016a) and that data indexed by B_{med} should not be outliers. The differentiation properties of $f \to \text{MOM}_K\left(\ell_f\right)$ are studied in Sect. 4.2. One additional advantage of our algorithm is that it is based on a simple idea: select a "good" block of data in such as way that it does not contain outliers and it is a subgaussian estimator of the risk. As a result, it requires only little modifications on existing Gradient descent based algorithms to make them robust to outliers and heavy-tailed data. As a proof of concept, in this article, we perform this "MOM modification" to the Logistic Regression, Perceptron and SVM-like algorithm.

In Sect. 5, the practical performances of thess algorithms are illustrated on several simulations, involving in particular different loss functions. These simulations illustrate not surprisingly the gain of robustness that there is to use these algorithms in their MOM version rather than in their traditional version, as can for example be appreciated on the toy-example of Fig. 1 (see also Fig. 4 below). MOM estimators are compared to different learning algorithms on real datasets that can be modeled by heavy tailed data, obtaining in each case performances comparable to the best of these benchmarks.

Another advantage of our procedure is that it works on blocks of data. This can improve speed of execution and reduce memory requirements, which can be decisive on massive datasets and/or when one wishes to use non-linear algorithms as in Sect. 4.3. This principle of dividing the dataset to calculate estimators more quickly and then aggregating them is a powerful tool in statistics and machine learning (Jordan 2013). Among others, one can mention bagging methods (Breiman 1996) or subagging—a variant of bagging where the bootstrap is replaced by subsampling—(Bühlmann and Bin 2002). These methods are considered difficult to study theoretically in general and their analysis is often limited to obtaining asymptotic guarantees. By contrast, the theoretical tools for non-asymptotic risk analysis of MOM minimizers have already essentially been developed. Finally, subsampling by the central block B_{med} ensures robustness properties that cannot be guaranteed by traditional alternatives.

Moreover, the algorithm provides an empirical notion of data depth: data providing good risk estimates of $f \to \mathbb{E}\ell_f(X, Y)$ are likely to be selected many times in the central block B_{med} along the descent, while outliers will be systematically ignored. This notion of depth, based on the risk function, is very natural for prediction problems. It is complemented by an outliers detection procedure: data that are selected a number of times below a predetermined threshold are classified outliers. This procedure is evaluated on our toy example of Fig. 1—for this example, data represented by the dots in the top right corner (the outliers) all end with a null score (see Fig. 7 below). The procedure is then tested on a real dataset on which the conclusions are more interesting. On this experiment, according to the theoretical upper bounds in Theorem 2, MOM minimizer's prediction qualities are deteriorated with large values of K, and this result is verified in some practical cases cf. Fig. 10. On the other hand, when there are enough data and when the data are not too heavy tailed (finite third moment

of the $f(X_i)$), the article Minsker (2019) decouples K and N in the risk bound and find an optimal scaling of $K \asymp \sqrt{N}$, and one might think that this decoupling ought to be possible also in our context. On the other hand, outlier detection is best when the number of blocks is large, cf. Fig. 8. Outlier resistance and anomaly detection tasks can therefore both be handled using the MOM principle, but the main hyper-parameter K—the number of blocks of data—for setting this method must be chosen carefully according to the objective. A number of blocks as small as possible (about twice the number of outliers) will give the best predictions, while large values of this number of blocks will accelerate the detection of anomalies. Note that it is essential for outliers detection to use different (for instance, random) partitions at each step of the descent to avoid giving the same score to an outlier and to all the data in the same block containing it.

Detecting outliers is usually performed in machine learning via some unsupervised preprocessing algorithm that detects outliers outside a bulk of data, see for example Hubert and Van Driessen (2004), He and Fung (2000), Christophe and Catherine (2001), Gunduz and Fokoué (2015) or other algorithms like DBSCAN (Birant and Kut 2007) or isolation forest (Liu et al. 2008). These algorithms assume elliptical symmetry of the data, a solution for skewed data can also be found in Hubert and Van Der Veeken (2010). These unsupervised preprocessing removes outliers in advance, i.e. before starting any learning task. As expected, these strategies work well in the toy example from Fig. 1. There are several cases where it will fail though. First, as explained in Huber and Ronchetti (2009), this strategy classifies data independently of the risk, it is likely to remove from the dataset outlier coming from heavy-tailed distribution, yielding biased estimators. Moreover, a small group of misclassified data inside a bulk won't be detected. Our notion of depth, based on the risk, seems more adapted to the learning task than any preprocessing procedure blind to the risk.

The paper is organized as follows. Section 2 presents the classification problem, the ERM and its MOM versions and gathers the assumptions granted for the main results. Section 3 presents theoretical risk bounds for the ERM estimator and MOM minimizers on corrupted datasets. Section 4 deals with theoretical results on the algorithm computing MOM minimizers. We present the algorithm, study the differentiation property of the objective function $f \to \mathrm{MOM}_K\left(\ell_f\right)$ and provide theoretical bounds on its complexity. Section 5 shows empirical performance of our estimators in both simulated and real datasets. Proofs of the main results are postponed to Sect. 6 where we also added heuristics on the practical choice of the hyper-parameters.

2 Setting

2.1 Empirical risk minimization for binary classification

Consider the supervised binary classification problem, where one observes a sample (X_1, Y_1), …, (X_N, Y_N) taking values in $\mathcal{X} \times \mathcal{Y}$. The set $\mathcal{X}$ is a measurable space and $\mathcal{Y} = \{-1, 1\}$. The goal is to build a classifier—that is, a measurable map $f : \mathcal{X} \to \mathcal{Y}$—such that, for any new observation (X, Y), $f(X)$ is a good prediction for Y. For any classifier f, let

$$\ell_f^{0-1}(x, y) = I\{y \neq f(x)\}, \qquad R^{0-1}(f) = P\ell_f^{0-1} = \mathbb{P}_{(X,Y)\sim P}\left(Y \neq f(X)\right) .$$

The $0-1$ risk $R^{0-1}(\cdot)$ is a standard measure of the quality of a classifier. Following Chervonenkis and Vapnik (2000), a popular way to build estimators is to replace the unknown measure P in the definition of the risk by the empirical measure P_N defined for any real valued

function g by $P_N g = N^{-1} \sum_{i=1}^{N} g(X_i, Y_i)$ and minimize the empirical risk. The *empirical risk minimizer* for the $0-1$ loss on a class $\mathcal{F}$ of classifiers is $\widehat{f}_{\mathrm{ERM}}^{0-1} \in \operatorname{argmin}_{f\in\mathcal{F}}\{P_N \ell_f^{0-1}\}$.

The main issue with $\widehat{f}_{\mathrm{ERM}}^{0-1}$ is that it cannot be computed efficiently in general. One source of computational complexity is that both $\mathcal{F}$ and the $0-1$ loss function are non-convex. This is why various convex relaxations of the $0-1$ loss have been introduced in statistical learning theory. These proceed in two steps. First, $\mathcal{F}$ should be replaced by a convex set F of functions taking values in $\mathbb{R}$. Then one builds an alternative loss function ℓ for ℓ^{0-1} defined for all $f \in F$. The new function ℓ should be convex and put less weight on those $f \in F$ such that $f(X_i)Y_i > 0$, these loss functions are commonly called "classification-calibrated losses" in the literature. Classical examples include the *hinge loss* $\ell_f^{\mathrm{hinge}}(x, y) = (1 - yf(x))_+$, or the *logistic loss* $\ell_f^{\mathrm{logistic}}(x, y) = \log(1 + e^{-yf(x)})$. A couple (F, ℓ) such that F is a convex set of real valued functions and ℓ is a convex function (i.e. for all $y \in \{-1, 1\}$ and $x \in \mathcal{X}$, $f \in F \to \ell_f(x, y)$ is convex) such that $\ell_f(x, y) < \ell_f(x, -y)$ whenever $yf(x) > 0$ will be called a convex relaxation of $(\mathcal{F}, \ell^{0-1})$. Given a convex relaxation (F, ℓ) of $(\mathcal{F}, \ell^{0-1})$, one can define the associated empirical risk minimizer by

$$\widehat{f}_{\mathrm{ERM}} \in \operatorname*{argmin}_{f\in F} P_N \ell_f \ . \tag{3}$$

Note that $\widehat{f}_{\mathrm{ERM}}$ does not build a classifier. To deduce, a classification rule from $\widehat{f}_{\mathrm{ERM}}$ one can simply consider its sign function defined for all $x \in \mathcal{X}$ by $\operatorname{sign}(\widehat{f}_{\mathrm{ERM}}(x)) = 2(I\{\widehat{f}_{\mathrm{ERM}}(x) \geq 0\} - 1/2)$. The procedure $\widehat{f}_{\mathrm{ERM}}$ is solution of a convex optimization problem that can therefore be approximated using a descent algorithm. We refer for example to Bubeck (2015) for a recent overview of this topic and Sect. 4 for more examples.

2.2 Corrupted datasets

In this paper, we consider a framework where the dataset may have been corrupted by *outliers* (or anomalies). There are several definitions of outliers in the literature, here, we assume that the dataset is divided into two parts. The first part is the set of inliers, indexed by $\mathcal{I}$, data $(X_i, Y_i)_{i\in\mathcal{I}}$ are hereafter always assumed to be independent and identically distributed (i.i.d.) with common distribution P. The second one is the set of outliers, indexed by $\mathcal{O} \subset [N]$ which has cardinality $|\mathcal{O}|$. Nothing is assumed on these data which may not be independent, have distributions P_i totally different from P, satisfying $P_i|f|^\alpha = \infty$ for any $\alpha > 0$, etc…Doing no hypothesis on the outliers is commonly done in Machine Learning with adversarial examples, see Gao et al. (2018) and Diakonikolas et al. (2017) for examples of such application. In particular, this framework is sufficiently general to cover the case where outliers are i.i.d. with distribution $Q \neq P$ as in the ϵ-contamination model (Huber and Ronchetti 2009; Chen et al. 2017; Gao 2017; Donoho and Montanari 2015).

Our first result shows that the rate of convergence of $\widehat{f}_{\mathrm{ERM}}^{0-1}$ is not affected by this corruption as long as $|\mathcal{O}|$ does not exceed $N \times$ (rate of convergence) see Theorem 1 and the remark afterward. However, it is easy to remark that, when the number N of data is finite as it is always the case in practice, even one aggressive outliers may yield disastrous breakdown of the empirical mean's statistical performance. Consequently, even if $\widehat{f}_{\mathrm{ERM}}^{0-1}$ behaves correctly, its proxy $\widehat{f}_{\mathrm{ERM}}$ defined in (3) for a convex relaxation (F, ℓ) can have disastrous statistical performances, particularly when F and ℓ are unbounded, cf. Fig. 4 for an illustration.

To bypass this problem, we consider in this paper an alternative to the empirical mean called *median-of-means* (Alon et al. 1999; Jerrum et al. 1986; Nemirovsky and Yudin 1983). Let $K \leq N$ denote an integer and let $B_1, \ldots, B_K$ denote a partition of $\{1, \ldots, N\}$ into bins

B_k of equal size $|B_k| = N/K$. If K doesn't divide N, one can always drop a few data. For any function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and any non-empty subset $B \subset \{1, \dots, N\}$, define the empirical mean on B by $P_B f = |B|^{-1} \sum_{i \in B} f(X_i, Y_i)$. The median-of-means (MOM) estimator of Pf is defined as the empirical median of the empirical means on the blocks B_k

$$\mathrm{MOM}_K\left(f\right) = \mathrm{median}\left\{P_{B_k} f \,:\, k = 1, \dots, K\right\} \ .$$

As the classical Huber's estimator (Huber 1964), MOM estimators interpolate between the unbiased but non robust empirical mean (obtained for $K = 1$) and the robust but biased median (obtained for $K = N$). In particular, when applied to loss functions, these new estimators of the risk $P\ell_f$, $f \in F$ suggest to define the following alternative to Chervonenkis and Vapnik's ERM estimator, called MOM minimizers

$$\widehat{f}_{\mathrm{MOM},K} \in \operatorname*{argmin}_{f \in F} \mathrm{MOM}_K\left(\ell_f\right) \ . \tag{4}$$

From a theoretical point of view, we will prove that, when the number $|\mathcal{O}|$ of outliers is smaller than $N \times$ (rate of convergence), $\widehat{f}_{\mathrm{MOM},K}$ performs well under a second moment assumptions on F and ℓ. To illustrate our main assumptions and theoretical results, we will regularly use the following classical example.

Example 1 (Linear classification) Let $\mathcal{X} = \mathbb{R}^p$ and let $\|\cdot\|_2$ denote the classical Euclidean norm on $\mathbb{R}^p$. Let F denote a set of linear functions

$$F = \{f_t : x \mapsto \langle x, t\rangle : \|t\|_2 \leq \Gamma\} \ .$$

Let ℓ denote either the hinge loss or the logistic loss defined respectively for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $f \in F$ by

$$\ell_f^{\mathrm{hinge}}(x, y) = (1 - yf(x))_+, \qquad \ell_f^{\mathrm{logistic}}(x, y) = \log(1 + e^{-yf(x)}) \ .$$

Remark that the case with an intercept is included in this linear case by adding an artificial $(p+1)$th dimension: we consider $x' = (x_1, \dots, x_p, 1)$ where $x_1, \dots, x_p$ are the coordinated of x, and then $\langle x', (t_1, \dots, t_p, t_{p+1}\rangle = \langle x, (t_1, \dots, t_p)\rangle + t_{p+1}$. In practice this correspond to adding a column of 1 at the end of the design matrix.

2.3 Main assumptions

As already mentioned, data are divided into two groups, a subset $\{(X_i, Y_i) : i \in \mathcal{O}\}$ made of outliers (on which we will make no assumption) and the remaining data $\{(X_i, Y_i) : i \in \mathcal{I}\}$ contains all data that bring information on the target/oracle

$$f^* \in \operatorname*{argmin}_{f \in F} P\ell_f \ .$$

Data indexed by $\mathcal{I}$ are therefore called *inliers* or *informative data*. To keep the presentation as simple as possible, inliers are assumed to be i.i.d. distributed according to P although this assumption could be relaxed as in Lecué and Lerasle (2017, 2019). Finally, note that the $\mathcal{O} \cup \mathcal{I} = \{1, \dots, N\}$ partition of the dataset is of course unknown from the statistician. Moreover, since no assumption is granted on the set of data indexed by $\mathcal{O}$, this setup covers the framework of adversarial attack where one may imagine that the data indexed by $\mathcal{O}$ have been changed in the worst possible way by some malicious adverser.

Let us now turn to the set of assumptions we will use to study MOM minimizers procedures. For any measure Q and any function f for which it makes sense, denote by $Qf = \int f dQ$.

Denote also, for all $q \geq 1$, by L^q the set of real valued functions f such that $\int |f|^q dP < \infty$ and, for any $f \in L^q$, by

$$\|f\|_{L^q} = \left(\int |f|^q dP \right)^{1/q} .$$

Our first assumption is an L^2-assumption on the functions in F.

Assumption 1 For all $f \in F$, we have $\|f\|_{L^2} \leq \theta_2$.

Of course, Assumption 1 is granted if F is a set of classifiers. It also holds for the linear class of functions from Example 1 as long as $P\|X\|_2^2 < \infty$ with $\theta_2 = \Gamma(P\|X\|_2^2)^{1/2}$. As announced in the introduction, it is a boundedness assumption (w.r.t. the L_2-norm) and not a norm equivalence assumption. For instance, it covers cases that cannot be handled via norm equivalence. A typical example is for matrix completion problems where X is uniformly distributed over the canonical basis $(E_{pq} : p \in [m], q \in [T])$ of the linear space $\mathbb{R}^{m\times T}$ of $m \times T$ matrices. One has for $f(\cdot) = \langle \cdot, E_{11} \rangle$ and any $r \geq 1$, $\|f\|_{L_r} = (\mathbb{E}|f(X)|^r)^{1/r} = (1/(mT))^{1/r}$. Hence, any norm equivalence assumption on the class $F = \{f_A = \langle \cdot, A \rangle : \|f_A\|_{L^2} \leq \theta_2\}$ will depend on the dimension mT of the problem resulting either in wrong rates of convergence or in assumption on the number of data. Our approach does not use any norm equivalence assumption so that our rates of convergence do not depend on dimension dependent ratio. Rates depend only on the L_2 radius θ_2 of F from Assumption 1.

The second assumption deals with the complexity of the class F. This complexity appears in the upper bound of the risk. It is defined using only informative data. Let

$$\mathcal{K} = \{k \in \{1, \ldots, K\} : B_k \cap \mathcal{O} = \emptyset\} \quad \text{and} \quad \mathcal{J} = \cup_{k\in\mathcal{K}} B_k .$$

Definition 1 Let $\mathcal{G}$ denote a class of functions $f : \mathcal{X} \to \mathbb{R}$ and let $(\epsilon_i)_{i\in\mathcal{I}}$ denote i.i.d. Rademacher random variables independent from $(X_i, Y_i)_{i\in\mathcal{I}}$. The *Rademacher complexity* of $\mathcal{G}$ is defined by

$$\mathcal{R}(\mathcal{G}) = \max_{A\in\{\mathcal{I},\mathcal{J}\}} \mathbb{E}\left[\sup_{f\in\mathcal{G}} \sum_{i\in A} \epsilon_i f(X_i) \right] .$$

The Rademacher complexity is a standard measure of complexity in classification problems (Bartlett and Mendelson 2002). It can be upper bounded by comp$/\sqrt{N}$ where comp is a measure of complexity such as the square root of the VC dimension or the Dudley's entropy integral or the Gaussian mean width of the class F see for example Boucheron et al. (2005, 2013), Koltchinskii (2008), Bartlett and Mendelson (2002) and Devroye et al. (1997) for a presentation of these classical bounds. Our second assumption is simply that the Rademacher complexity of the class F is finite.

Assumption 2 The Rademacher complexity of F is finite, $\mathcal{R}(F) < \infty$.

Assumption 2 holds in the linear classification example under Assumption 1 since it follows from Cauchy-Schwarz inequality that $\mathcal{R}(F) \leq \theta_2\sqrt{|\mathcal{I}|p}$. Finally, our last assumption is that the loss function ℓ considered is Lipschitz in the following sense.

Assumption 3 The loss function ℓ satisfies for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and all $f, f' \in F$,

$$|\ell_f(x, y) - \ell_{f'}(x, y)| \leq L|f(x) - f'(x)| .$$

Assumption 3 holds for classical convex relaxation of the $0-1$ loss such as hinge loss ℓ^{hinge} or logistic loss ℓ^{logistic} as in Example 1. In these examples, the constant L can be chosen equal to 1. It also covers non-convex loss functions such as the one in Baraud et al. (2017), Catoni (2012) and Audibert and Catoni (2011) or sigmoid loss functions such as the one used in Deep Learning. In particular, our results do not follow from other work on MOM estimators using convex loss functions such as in Chinot et al. (2019).

3 Theoretical guarantees

Our first result follows Vapnik–Chervonenkis's original risk bound for the ERM and shows that $\widehat{f}^{0-1}_{\text{ERM}}$ is insensitive to the presence of outliers in the dataset. Moreover, it quantifies this robustness property since Vapnik–Chervonenkis's rate of convergence is still achieved by $\widehat{f}^{0-1}_{\text{ERM}}$ when there are less than (number of observations) times (Vapnik's rate of convergence) outliers.

Theorem 1 *Let $\mathcal{F}$ denote a collection of classifiers. Let $\mathcal{L}^{0-1}_{\mathcal{F}} = \{\ell^{0-1}_f - \ell^{0-1}_{f^*} : f \in \mathcal{F}\}$ be the family of excess loss functions indexed by $\mathcal{F}$ where $f^* \in \operatorname{argmin}_{f\in\mathcal{F}} R^{0-1}(f)$. For all $K > 0$, with probability at least $1 - e^{-K}$, we have*

$$R^{0-1}(\widehat{f}^{0-1}_{ERM}) - \inf_{f\in\mathcal{F}} R^{0-1}(f) \leq \frac{2\mathcal{R}(\mathcal{L}^{0-1}_{\mathcal{F}})}{N} + \sqrt{\frac{K}{2|\mathcal{I}|}} + \frac{2|\mathcal{O}|}{N}.$$

Theorem 1 is proved in Sect. 6.1. It is an adaptation of Vapnik–Chervonenkis's proof of the excess risk bounds satisfied by $\widehat{f}^{0-1}_{\text{ERM}}$ in the presence of outliers.

Remark 1 In the last result, one can easily bound the excess risk using $\mathcal{R}(\mathcal{F})$ instead of $\mathcal{R}(\mathcal{L}^{0-1}_{\mathcal{F}})$ since

$$\mathcal{R}(\mathcal{L}^{0-1}_{\mathcal{F}}) = \max_{A\in\{\mathcal{I},\mathcal{J}\}} \mathbb{E}\left[\sup_{f\in\mathcal{F}} \sum_{i\in A} \epsilon_i (f(X_i) - f^*(X_i))\right] = \mathcal{R}(\mathcal{F}) \ .$$

The final bound is of similar flavor: for all $K > 0$, with probability at least $1 - e^{-K}$, we have

$$R^{0-1}(\hat{f}^{0-1}_{\text{ERM}}) - \inf_{f\in\mathcal{F}} R^{0-1}(f) \lesssim \max\left(\frac{\mathcal{R}(\mathcal{F})}{N}, \sqrt{\frac{K}{|\mathcal{I}|}}, \frac{|\mathcal{O}|}{N}\right) \ . \tag{5}$$

Remark 2 When $\mathcal{F}$ is the class of all linear classifiers, that is when $\mathcal{F} = \{\text{sgn}(\langle t, \cdot\rangle) : t \in \mathbb{R}^p\}$, one has $\mathcal{R}(\mathcal{F}) \leq \sqrt{|\mathcal{I}|p}$ [see Theorem 3.4 in Boucheron et al. 2005]. Therefore, when $|\mathcal{I}| \geq N/2$, Theorem 1 implies that for all $1 \leq K \leq p$, with probability at least $1-\exp(-K)$,

$$R^{0-1}(\hat{f}^{0-1}_{\text{ERM}}) - \inf_{f\in\mathcal{F}} R^{0-1}(f) \lesssim \max(\sqrt{p/N}, |\mathcal{O}|/N) \ .$$

As a consequence, when the number of outliers is such that $|\mathcal{O}| \lesssim N \times \sqrt{p/N}$, Vapnik–Chervonenkis's classical "slow" rate of convergence $\sqrt{p/N}$ is still achieved by the ERM estimator even if $|\mathcal{O}|$ outliers have polluted the dataset. The interested reader can also check that "fast rates" p/N could also be achieved by the ERM estimator in the presence of outliers if $|\mathcal{O}| \lesssim p$ and when the so-called strong margin assumption holds (see, Boucheron et al. 2005). Note also that the previous remark also holds if F is a class with VC dimension p beyond the case of indicators of half spaces.

The conclusion of Theorem 1 can be misleading in practice. Indeed, theoretical performance of the ERM estimator for the $0-1$ loss function are not downgraded by outliers, but its proxies based on convex relaxation (F, ℓ) of $(\mathcal{F}, \ell^{0-1})$ are. This can be seen on the toy example in Fig. 1 and in Fig. 4 from Sect. 5. In this work, we propose a robust surrogate, based on MOM estimators of the risk and defined in (2), to the natural empirical risk estimation of the risk which works for unbounded loss functions. In the next result, we prove that the MOM minimizer $\hat{f}_{\mathrm{MOM},K}$ defined as

$$\hat{f}_{\mathrm{MOM},K} \in \operatorname*{argmin}_{f\in F} \mathrm{MOM}_K\left(\ell_f\right) \tag{6}$$

satisfies an excess risk bound under weak assumptions introduced in Sect. 2.

Theorem 2 *Grant Assumptions 1, 2 and 3. Assume that* $N > K > 4|\mathcal{O}|$ *and let* $\Delta = 1/4 - |\mathcal{O}|/K$. *Then, with probability larger than* $1 - 2\exp\left(-2\Delta^2 K\right)$, *we have*

$$R(\hat{f}_{MOM,K}) \leq \inf_{f\in F} R(f) + 4L \max\left(\frac{4\mathcal{R}(F)}{N}, 2\theta_2\sqrt{\frac{K}{N}}\right) .$$

Theorem 2 is proved in Sect. 6.2. Compared to Theorem 1, $\hat{f}_{\mathrm{MOM},K}$ achieves the same rate $(\mathcal{R}(F)/N) \vee (\sqrt{K/N})$ under the same conditions on the number of outliers with the same exponential control of the probability as for the ERM estimator f^{0-1}_{ERM}. The main difference is that the loss function may be unbounded, which is often the case in practice. Moreover, unlike classical analysis of ERM obtained by minimizing an empirical risk associated with a convex surrogate loss function, we only need a second moment assumption on the class F.

These theoretical improvements have already been noticed in previous works (Minsker 2015; Devroye et al. 2016b; Lugosi and Mendelson 2017, 2019a, b; Lecué and Lerasle 2017, 2019; Mendelson 2017). Contrary to tournaments of Lugosi and Mendelson (2017), Le Cam MOM estimators of Lecué and Lerasle (2019) or minmax MOM estimators Lecué and Lerasle 2019, Theorem 2 does not require the small ball assumption on F but only shows "slow rates" of convergence. These slow rates are minimax optimal in the absence of a margin or Bernstein assumption (Bartlett and Mendelson 2006; Mammen and Tsybakov 1999). Removing the small ball assumption may be useful in some examples. As an illustration, consider the toy example where the design

$$X = \begin{bmatrix} \mathbf{1}_{W\in I_1} \\ \vdots \\ \mathbf{1}_{W\in I_d} \end{bmatrix}$$

where $I_1, \ldots, I_d$ is a partition of a measurable set $\mathbb{W}$ into subsets such that $\mathbb{P}(W \in I_i) = 1/d$ for each $i \in \{1, \ldots, d\}$. Then $\mathbb{X} = [0,1]^d$ and one can consider the set F of linear functions $f(X) = \langle t, X\rangle$, where the Euclidean norm of t satisfies $\|t\| \leq B\sqrt{d}$. Then, as $\|\langle t, \cdot\rangle\|^2_{L^2} = \sum_{i=1}^d t_i^2 \mathbb{P}(W \in I_i) = \|t\|^2/d$, Assumption 1 holds with $\theta_2 = B$. In this example, Assumption 2 holds with $\mathcal{R}(F) \leq \theta_2\sqrt{|\mathcal{I}|d} \leq B\sqrt{Nd}$. It follows from Theorem 2 that the remainder term in this example is bounded from above by

$$4LB \max\left(4\sqrt{\frac{d}{N}}, 2\theta_2\sqrt{\frac{K}{N}}\right) .$$

In particular, it converges to 0 if $d \vee K \ll N$. By comparison, in the same example, it is shown in Chinot et al. (2019) that the remainder term converges to 0 only if $d \lesssim \sqrt{N}$.

Proof of Theorem 2 does not enable fast rates to be obtained. Indeed, the non-linearity of the median excludes the possibility of using localization techniques leading to these fast rates. However, we show in the simulation study (cf. left side picture of Fig. 12) that fast rates seem to be reached by the MOM minimizer.

Remark 3 The MOM principle has been used together with Lipschitz loss functions recently in Chinot et al. (2019). In this paper, a minmax MOM estimator is constructed which can achieve fast rates of convergence under a margin condition. The argument from Chinot et al. (2019) relies heavily on the convexity of the loss—an assumption we do not have here. The reason why the convexity of the loss is so important in Chinot et al. (2019) is that it allows to exclude (as potential minmax MOM estimator) all the functions in F outside a L_2-ball centered in f^* with radius r if all the functions in F in the sphere $f^* + r\mathcal{S}_2$ are excluded. Therefore, thanks to convexity, the latter " homogeneity argument" reduces the problem to the study of the sub-model $F \cap (f^* + r\mathcal{S}_2)$ (which is bounded in L_2 with the right radius r). Here, no such homogeneity argument can be used because we did not assume the loss to be convex. Nevertheless, if we assume that the loss is convex then we may still apply Theorem 4 in Chinot et al. (2019) and replace all the localized sets by the entire set F and the variance term by the L_2 uniform bound θ_2 coming from Assumption 1 to obtain a similar result as Theorem 2 for a minmax MOM estimator. These stronger results require the convexity of the loss and a Bernstein assumption that may be satisfied only under strong assumptions as discussed in the toy example.

Finally, the main advantage of our approach is its simplicity, we just have to replace empirical means by their MOM alternative in the definition of the ERM estimator. Moreover, as expected, this simple alternative to ERM estimators yields a systematically way to modify algorithms designed for approximating the ERM estimator. The resulting "MOM versions" of these algorithms are both faster and more robust than their original "ERM version". Before illustrating these facts on simulations, let us describe algorithms approximating MOM minimizers.

4 Computation of MOM minimizers

In this section, we present a generic algorithm to provide a MOM version of descent algorithms. We study the differentiation property of the objective function $f \to \mathrm{MOM}_K(\ell_f)$. Then we check on simulated and real databases the robustness and outlier detection property of these MOM algorithms.

4.1 MOM algorithms

The general idea is that any descent algorithms such as gradient descent, Newton method, alternate gradient descent, etc. (cf. Moulines and Bach 2011; Bubeck 2015; Boyd and Vandenberghe 2004; Bach et al. 2012) can easily be turned into a robust MOM-version. To illustrate this idea, a basic gradient descent is analyzed in the sequel. We start with a block splitting policy of the database.

The choice of blocks greatly influences the practical performance of the algorithm. In particular, a recurring flaw is that iterations tend to get stuck in local minima, which greatly slows the convergence of the alogorithm. To overcome this default and improve the stability

of the procedure, a new partition is constructed at each iteration by drawing it uniformly at random, cf. step **2** of Algorithm 1.

Let $\mathcal{S}_N$ denote the set of permutations of $\{1,\dots,N\}$. For each $\sigma \in \mathcal{S}_N$, let $B_0(\sigma) \cup \cdots \cup B_{K-1}(\sigma) = \{1,\dots,N\}$ denote an equipartition of $\{1,\dots,N\}$ defined for all $j \in [\![0, K-1]\!]$ by

$$B_j(\sigma) = \{\sigma(Kj+1),\dots,\sigma(K(j+1))\} = \sigma\left(\{Kj+1,\dots,K(j+1)\}\right) \ .$$

To simplify the presentation, let us assume the class F to be parametrized $F = \{f_u : u \in \mathbb{R}^p\}$, for some $p \in \mathbb{N}^*$. Let's assume that the function $u \mapsto f_u$ is as regular as needed and convex (a typical example is $f_u(x) = \langle u, x\rangle$ for all $x \in \mathbb{R}^p$). Denote by $\nabla_u \ell_{f_u}$ the gradient or a subgradient of $u \mapsto \ell_{f_u}$ in $u \in \mathbb{R}^p$. The step-sizes sequence is denoted by $(\eta_t)_{t\geq 0}$ and satisfies the classical conditions: $\sum_{t=1}^{\infty} \eta_t = \infty$ and $\sum_{t=1}^{\infty} \eta_t^2 < \infty$. Iterations will go on until a stopping time $T \in \mathbb{N}^*$ has been achieved. With these notations, a generic MOM version of a gradient descent algorithm (with random choice of blocks) is detailed in Algorithm 1 below.

input : $u_0 \in \mathbb{R}^p$, $K \in [\![3, N/2]\!]$, $T \in \mathbb{N}^*$ and $(\eta_t)_{t\in\{0,\dots,T-1\}} \in \mathbb{R}_+^T$
output: a MOM version of BSGD

1 **for** $t = 0, \cdots, T-1$ **do**
2 choose a permutation at random: $\sigma_t \sim \mathrm{Unif}(\mathcal{S}_N)$,
3 build a partition of the dataset: $B_0(\sigma_t), \dots, B_{K-1}(\sigma_t)$,
4 find a median block: $k_{med}(t)$ s.t. $\mathrm{MOM}_K\left(\ell_{f_{u_t}}\right) = P_{B_{k_{med}(t)}(\sigma_t)}\left(\ell_{f_{u_t}}\right)$,
5 do a descent step on the median block

$$u_{t+1} = u_t - \eta_t \nabla_t \text{ where } \nabla_t = \sum_{i \in B_{k_{med}(t)}(\sigma_t)} \nabla_{u_t} \ell_{f_{u_t}}(X_i, Y_i).$$

6 **end**
7 **Return** u_T

Algorithm 1: MOM gradient descent algorithm.

Remark 4 (MOM gradient descent algorithm and stochastic block gradient descent) Algorithm 1 can be seen as a stochastic block gradient descent (SBGD) algorithm minimizing the function $t \to \mathbb{E}\ell_t(X,Y)$ using a given dataset. The main difference with the classical SBGD is that the choice of the block along which the gradient direction is performed is chosen according to a centrality measure computed thanks to the median operator in step **4** of Algorithm 1.

In Sect. 5, we use the MOM principle (as in the generic Algorithm 1) to construct MOM versions for various classical algorithms such as Perceptron, Logistic Regression, Kernel Logistic Regression, SGD Classifiers or Multi-layer Perceptron.

4.2 Differentiation properties of $f \to \mathrm{MOM}_K(\ell_f)$, random partition and local minima

Let us try to explain the choice of the descent direction ∇_t in step **5** of Algorithm 1. In the previous sections, we introduced and studied MOM minimization procedures which are

minimizers of $f \to \mathrm{MOM}_K\left(\ell_f\right)$ over F. The optimization problem that needs to be solved to construct a MOM minimizer is not convex, in general. It therefore raises difficulties since classical tools and algorithms from the convex optimization toolbox cannot be used a priori. Nevertheless, one may still try to do a gradient descent algorithm for this (non-convex) optimization problem with objective function given by $f \to \mathrm{MOM}_K\left(\ell_f\right)$. To do so, we first need to check the differentiation properties of $f \to \mathrm{MOM}_K\left(\ell_f\right)$ over F.

First observe that the descent direction ∇_t is the gradient of the empirical risk constructed on the median block of data $B_{k_{med}(t)}(\sigma_t)$ at f_{u_t} (we recall that F is parametrized like $\{f_u : u \in \mathbb{R}^p\}$). A classical Gradient Descent algorithm on $f \to \mathrm{MOM}_K\left(\ell_f\right)$ starting from f_{u_t} would use a gradient at f_{u_t} of the objective function. Let us first identify situations where this is indeed the case i.e. when ∇_t is the gradient of $f \to \mathrm{MOM}_K\left(\ell_f\right)$ in f_{u_t}.

Assumption 4 For almost all datasets $\mathcal{D}_N = \{(X_i, Y_i) : i = 1, \ldots, N\}$ and Lebesgue almost all $u \in \mathbb{R}^p$, there exists an open convex set B containing u such that for any equipartition of $\{1, \ldots, N\}$ into K blocks $B_1, \ldots, B_K$ there exists $k_{med} \in \{1, \cdots, K\}$ such that for all $v \in B$, $P_{B_{k_{med}}}(\ell_{f_v}) \in \mathrm{MOM}_K\left(\ell_{f_v}\right)$.

In other word, under Assumption 4, for almost all $u_0 \in \mathbb{R}^p$, the median block $B_{k_{med}}$ achieving $\mathrm{MOM}_K\left(\ell_{f_{u_0}}\right)$ is the same as the one achieving $\mathrm{MOM}_K\left(\ell_{f_u}\right)$ for all u in an open and convex neighborhood B of u_0. It means that the objective function $u \to \mathrm{MOM}_K\left(\ell_{f_u}\right)$ is equal to the empirical risk function over the same block of data $B_{k_{med}}$: $u \to P_{B_{k_{med}}}\ell_{f_u}$, on B. Since B is an open set and that $u \to P_{B_{k_{med}}}\ell_{f_u}$ is differentiable in u_0 then the objective function $u \to \mathrm{MOM}_K\left(\ell_{f_u}\right)$ is also differentiable in u_0 and the two gradients coincide:

$$\nabla\left(u \to \mathrm{MOM}_K\left(\ell_{f_u}\right)\right)_{|u_0} = \nabla\left(u \to P_{B_{k_{med}}}\ell_{f_u}\right)_{|u_0}. \tag{7}$$

Under Assumption 4, Algorithm 1 is indeed a gradient descent algorithm performed on the objective function $u \in \mathbb{R}^p \to \mathrm{MOM}_K\left(\ell_{f_u}\right)$.

Let us give an example where Assumption 4 is satisfied. Let $B_1 \cup \cdots \cup B_K = \{1, \ldots, N\}$ be an equipartition and let ψ be defined for all $x = (x_i)_{i=1}^N \in \mathbb{R}^N$ and $u \in \mathbb{R}^p$ by,

$$\psi_u(x) = \mathrm{MOM}_K\left(f_u(x)\right) = \mathrm{median}\left(\frac{K}{N}\sum_{i\in B_k} f_u(x_i), 1 \le k \le K\right) = P_{B(K/2)(u)}(f_u),$$

where for all blocks $B \subset \{1, \ldots, N\}$, $P_B f_u = |B|^{-1}\sum_{i\in B} f_u(x_i)$ and the blocks $B(k)(u), k = 1, \ldots, K$ are rearranged blocks defined such that $P_{B(1)(u)}(f_u) \ge \cdots \ge P_{B(K)(u)}(f_u)$. Proposition 1 below shows that Assumption 4 is satisfied in several situations. Its proof can be found in Section 6.

Proposition 1 *Let $X_1, \ldots, X_N$ be N real-valued random variables, suppose K is odd and N is a multiple of K. Let $(f_u)_{u\in\mathbb{R}^d}$ be a family of functions with values in $\mathbb{R}$. Assume that for all $x \in \mathbb{R}$, the function $u \mapsto f_u(x)$ is Lipschitz and the probability distribution of $f_u(X_1)$ has a law absolutely continuous with respect to Lebesgue measure. Then, with probability 1, Assumption 4 is satisfied, in particular, the partial derivative of $u \mapsto \psi_{f_u}((X_i)_{i=1}^N) = MOM_K(f_u((X_i)_{i=1}^N))$ with respect to the jth coordinate is given for almost all $X_1, \ldots, X_N$ by*

$$\partial_j \psi_{f_u}((X_i)_{i=1}^N) = \frac{K}{N}\sum_{i\in B(\lceil K/2\rceil)(u)} \partial_j f_u(X_i)$$

where ∂_j denote the derivative with respect to the jth coordinate of u.

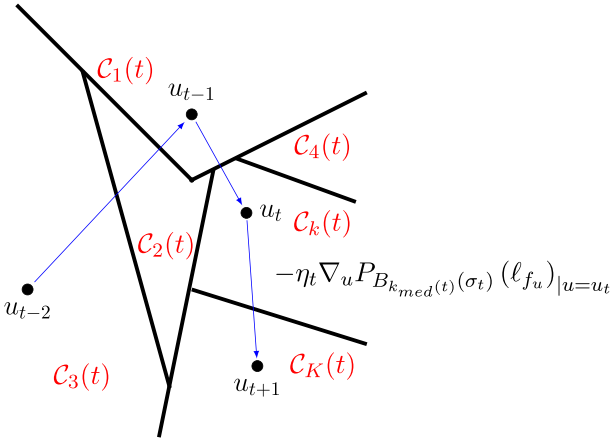


Fig. 2 Partition of $\mathbb{R}^p$ at step t by the median operator and iteration number $t-2$, $t-1$, t and $t+1$ of the MOM gradient descent algorithm. Under Assumption 4, there is a natural descent direction given at step t by $-\nabla_u(u \to P_{B_{k_{med}(t)}(\sigma_t)}(\ell_{f_u}))_{|u=u_t}$

Under Assumption 4, the picture of the MOM gradient descent algorithm is pretty simple and depicted in Fig. 2. At every step t, the median operator makes a partition of $\mathbb{R}^p$ into K cells $\mathcal{C}_k(t) = \{u \in \mathbb{R}^p : \mathrm{MOM}_K(\ell_{f_u}) = P_{B_k}\ell_{f_u}\}$ for $k = 1, \ldots, K$—this partition changes at every step because the blocks $B_1, \ldots, B_K$ are chosen randomly at the beginning of every step according to the random partition σ_t. We want every iteration u_t of the MOM algorithm to be in the interior of a cell and not on a frontier in order to differentiate the objective function $u \to \mathrm{MOM}_K(\ell_{f_u})$ at u_t. This is indeed the case under Assumption 4, given that in that case, there is an open neighbor B of u_t such that for all $v \in B$, $\mathrm{MOM}_K(\ell_{f_v}) = P_{B_k}\ell_{f_v}$ where the index $k = k_{med}$ of the block is common to every $v \in B$. Therefore, to differentiate the objective function $u \to \mathrm{MOM}_K(\ell_{f_u})$ at u_t one just needs to differentiate $u \to P_{B_k}\ell_{f_u}$ at u_t. The objective function to minimize is differentiable almost everywhere under Assumption 4 and a gradient of the objective function is given by $\nabla(u \to P_{B_k}\ell_{f_u})_{|u=u_t}$, that is ∇_t from step **5** of Algorithm 1.

Under Assumption 4, the importance of partitioning the dataset at each new iteration is more transparent. Indeed, if we were to perform the MOM gradient descent such as in Algorithm 1 but without a new partition at each step then local minima of the K empirical risks $u \to P_{B_k}\ell_{f_u}, k \in [K]$ may mislead the descent algorithm. Indeed, if a minimum of $u \to P_{B_k}\ell_{f_u}$ for some $k \in [K]$ is in the cell $\mathcal{C}_k$ then the algorithm will reach this minimum without noticing that a " better" minimum is in another cell. That is why re-partitioning the dataset of every iteration avoid this effect and speed up the convergence (see Lecué and Lerasle 2017 for experiments).

4.3 Complexity of MOM risk minimization algorithms

In this section, we compute the computational cost of several MOM versions of some classical algorithms. Let $C(m)$ be the computational complexity of a single standard gradient descent update step on a dataset of size m and let $L(m)$ be the computational complexity of the evaluation of the empirical risk $(1/m)\sum_{i\in B}\ell_f(X_i, Y_i)$ of some $f \in F$ on a dataset

B containing m data. Here the computational complexity is simply the number of basic operations needed to perform a task (Arora and Barak 2009).

For each epoch, we begin by computing the "MOM empirical risk". We perform K times N/K evaluations of the loss function, then we sort the K means of these blocks of loss to finally get the median. The complexity of this step is then $O(KL(N/K) + K\ln(K))$, assuming that the sort algorithm is in $O(K\ln(K))$ (like *quick sort* (Hoare 1962)). Then we do the gradient step on a sample of size N/K. Hence, the time complexity of this algorithm is

$$O(T(KL(N/K) + K\ln(K) + C(N/K))).$$

Example 2 (Linear complexity " ERM version" algorithms) For example, if the standard gradient step and the loss function evaluation have linear complexity—like Perceptron or Logistic Regression—the complexity of the MOM algorithm is $O(T(N+K\log(K)))$ against $O(TN)$ for the ERM algorithm. Therefore, the two complexities are of the same order and the only advantage of MOM algorithms lies in their robustness to outliers and heavy-tailed properties.

Example 3 (Super-linear complexity " ERM version" algorithms) If, on the other hand, the complexity is more than linear as for Kernel Logistic Regression (KLR), taking into account the matrix multiplications whose complexity can be found in Le Gall (2014), the complexity of the MOM version of KLR, due to the additional need of the computation of the kernel matrix, is $O(N^2+T(N^2/K+K\log(K)+(N/K)^{2.373}))$ against $O(TN^{2.373})$ for the standard " ERM version". MOM versions of KLR are therefore faster than the classical version of KLR on top of being more robust. This advantage comes from the fact that MOM algorithms work on blocks of data instead on the entire dataset at every step. More informations about Kernel Logistic Regression can be found in Roth (2001) for example.

In this last example, the complexity comes in part from the evaluation of the kernel matrix that can be computationally expensive. Following the idea that MOM algorithms are performing ERM algorithm restricted to a wisely chosen block of data, then one can modify our generic strategy in this particular example to reduce drastically its complexity. The idea here is that we only need to construct the kernel matrix on the median block. The resulting algorithm, called Fast KLR MOM is described in Fig. 2.

In Fig. 2, we compute only the block kernel matrices, denoted by $N^1, \ldots, N^k$ and constructed from the samples in the block B_k. We also denote by N_i^k the ith row in N^k.

input : $\alpha_0 \in \mathbb{R}^p$, $K \in [\![3, N/2]\!]$, $T \in \mathbb{N}^*$, $(\eta_t)_{t\in\{0,\dots,T-1\}} \in \mathbb{R}_+^T$, $\beta \in \mathbb{R}_+^*$, $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ a positive definite kernel and a bloc decomposition $B_1, \dots, B_K$ of $\{1, \dots, N\}$.
output: a MOM version of KLR classifiers

Construct the bloc Kernel matrices $N^k = (\kappa(X_i, X_j))_{i,j\in B_k}$ for $1 \le k \le K$,
for $t = 0, \cdots, T-1$ **do**
  find a median block: $k_{med}(t)$ s.t. $\mathrm{MOM}_K\left(\ell_{f_{\alpha_t}}\right) = P_{B_{k_{med}(t)}}\left(\ell_{f_{\alpha_t}}\right)$ with

$$P_{B_k}(\ell_{f_{\alpha_t}}) = \frac{1}{|B_k|}\sum_{i\in B_k} \ln(1 + e^{-N_i^k \alpha_t^k Y_i}) + \beta \sum_{k=1}^{K} (\alpha_t^k)^T N^k \alpha_t^k,$$

  where α_t^k is the vector in $\mathbb{R}^{|B_k|}$ made of the coordinates of α_t with indices in B_k.
  Do an IRLS descent step for KLR with weight matrice $W_{k_{med}(t)}$, design matrice $X_{k_{med}(t)}$ and labels $y_{k_{med}(t)}$ on $B_{k_{med}(t)}$

$$\alpha_{t+1}^{k_{med}(t)} = \alpha_t^{k_{med}(t)}(1-\eta_t) + \eta_t (X_{k_{med}(t)}^T W_{k_{med}(t)} X_{k_{med}(t)})^{-1} X_{k_{med}(t)}^T W_{k_{med}(t)} y_{k_{med}(t)}.$$

  $\alpha_{t+1}^k = \alpha_t^k (1-\eta_t), \quad \forall k \neq k_{med}(t).$
end
Return α_T, $N_{k_{med}(T)}$

Algorithm 2: Description of Fast KLR MOM algorithm.

There are several drawbacks in the approach of Algorithm 2. First, the blocks are fixed at the beginning of the algorithm; therefore the algorithm needs a bigger dataset to work well and it may converge to a local minimum. Nonetheless, from the complexity point of view, this algorithm will be much faster than both the classical KLR and MOM KLR (see below for a computation of its complexity) which is important given the growing use of kernel methods on very large databases for example in biology. The choice of K should ultimately realize a trade-off between complexity and performance (in term of accuracy for example) when dealing with big databases containing few outliers.

Example 4 (Complexity of Fast KLR-MOM algorithm) The complexity of Fast KLR-MOM is $O(N^2/K + T(N^2/K + K\log(K) + (N/K)^{2.373}))$ against $O(TN^{2.373})$ for the ERM version.

5 Implementation and simulations

5.1 Basic results on a toy dataset

The toy model we consider models outliers due to human or machine errors we would like to ignore in our learning process. It is also a dataset corrupted to make linear classifiers fail. The dataset is a 2D dataset constituted of three "labeled Gaussian distribution". Two informative Gaussians $\mathcal{N}((-1,-1), 1.4I_2)$ and $\mathcal{N}((1,1), 1.4I_2)$ with label respectively 1 and -1 and one outliers Gaussian $\mathcal{N}((24,8), 0.1I_2)$ with label 1. In other words, the distribution of informative data is given by $\mathcal{L}(X|Y=1) = \mathcal{N}((-1,-1), 1.4I_2)$, $\mathcal{L}(X|Y=-1) = \mathcal{N}((1,1), 1.4I_2)$ and $\mathbb{P}(Y=1) = \mathbb{P}(Y=-1) = 1/2$. Outliers data have distribution given by $Y = 1$ a.s. and $X \sim \mathcal{N}((24,8), 0.1I_2)$.

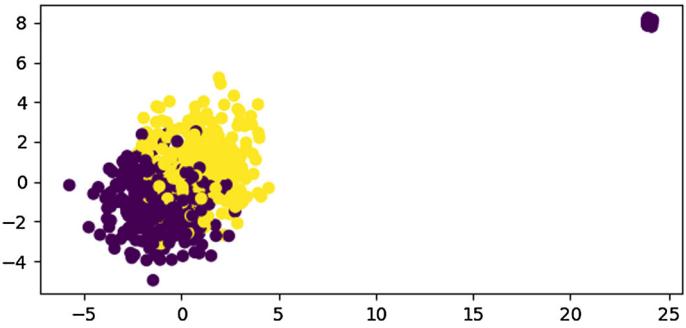


Fig. 3 Scatter plot of 630 samples from the training dataset (600 informative data, 30 outliers), the color of the points correspond to their labels (Color figure online)

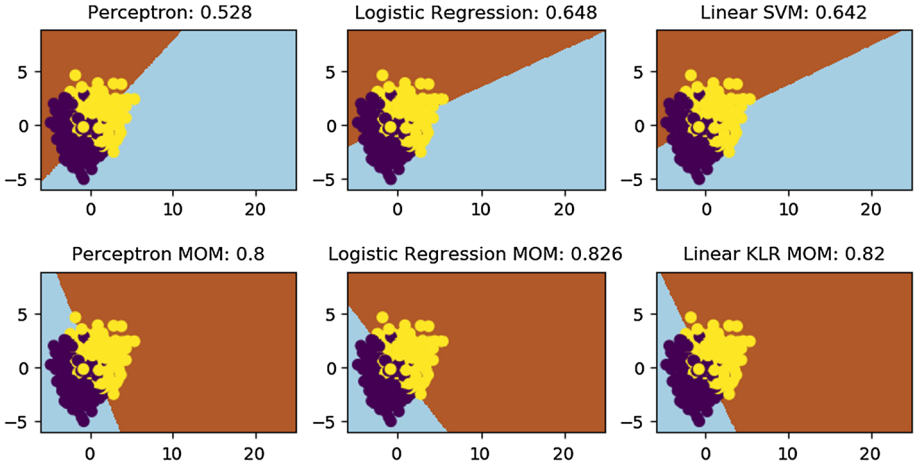


Fig. 4 Scatter plot of 500 samples from the test dataset (500 informative data), the color of the points correspond to their labels and the background color correspond to the prediction. The score in the title of each subfigure is the accuracy of the algorithm (Color figure online)

The algorithms we study are the MOM adaptations of Perceptron, Logistic Regression and Kernel Logistic Regression.

Based on our theoretical results, we know that the number of blocks K has to be larger than 4 times the number of outliers for our procedure to be on the safe side. The value $K = 120$ is therefore used in all subsequent applications of MOM algorithms on the toy dataset except when told otherwise. To quantify performance, we compute the miss-classification error on a clean dataset made of data distributed like the informative data.

For Kernel Logistic Regression, we study here a linear kernel because outliers in this dataset are clearly adversarial when dealing with linear classifiers. The algorithm can also use more sophisticated kernels, a comparison of the MOM algorithms with similar ERM algorithms is represented in Fig. 4, the ERM algorithms are taken from the python library scikit-learn (Pedregosa et al. 2011) with their default parameters.

Figure 4 illustrates resistance to outliers of MOM's algorithms compared to their classical version.

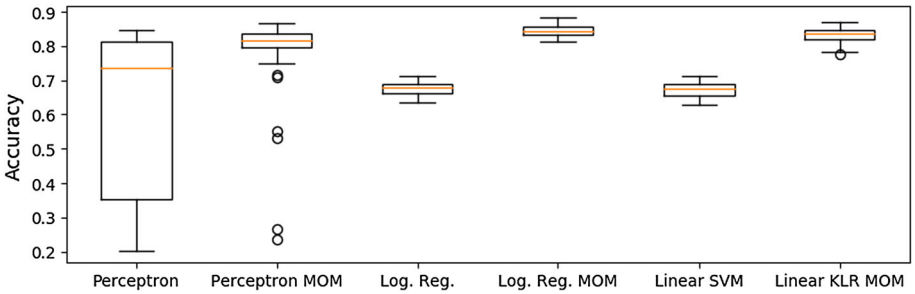


Fig. 5 Comparison of the MOM algorithms and their counterpart with the boxplots of the accuracy on the test dataset from 50 runs of the algorithms on 50 sample of the training/test toy dataset (one run for each dataset sampled)

Table 1 Time of different algorithms on a simulated dataset

Algorithm	Perceptron MOM	Log. Reg. MOM	KLR MOM	Fast KLR MOM
Time (s)	1.06	1.05	13.6	1.2
Algorithm	Rand. Forest	SVM	SGD Hub. loss	
Time (s)	0.21	9.0	0.0078	

These first results are completed in Fig. 5 where we computed accuracy on several run of the algorithms. These results confirm the visual impression of our first experiment.

Finally, we illustrate our results regarding complexities of the algorithms on a simulated example. MOM algorithms have been computed together with state-of-the art algorithms from scikit-learn (Pedregosa et al. 2011) (we use Random forest, SVM classifier as well as SGD classifier optimizing Huber loss which entail a robustness in Y but not in X, see Huber and Ronchetti 2009, Chapter 7) on a simulated dataset composed of two Gaussian blobs $\mathcal{N}((-1,-1), 1.4I_2)$ and $\mathcal{N}((1,1), 1.4I_2)$ with label respectively 1 and -1. We sample 20000 points for the training dataset and 20000 for the test dataset. The parameters used in the algorithms are those for which we obtained the optimal accuracy, (this accuracy is illustrated in the next section). Time of training plus time of evaluation on the test dataset are gathered in Table 1.

Not surprisingly, very efficient versions of linear algorithms from Python's library are extremely fast (results are sometimes provided before we even charged the dataset in some experiments). The performance of our algorithm are nevertheless acceptable in general (around 5 times longer than random forest for example). The important fact here is that non linear algorithms such as SVM take much more time to provide a result. FAST KLR MOM is able to reduce substantially the execution time of SVM with comparable predictive performance.

5.2 Applications on real datasets

We used the HTRU2 dataset, also studied in Lyon et al. (2015), that is provided by the UCI Machine Learning Repository. The goal is to detect pulsars (a rare type of Neutron star) based on radio emission detectable on earth from which features are extracted to gives us

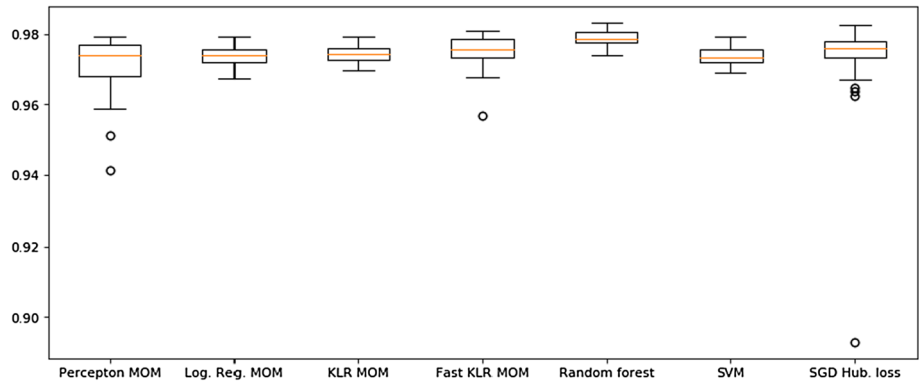


Fig. 6 Comparison of the MOM algorithms and common algorithms with the boxplots and the medians of the accuracy $\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\{\hat{f}(x_i) = y_i\}$ on the test dataset from 50 runs of the algorithms on 100 sample of a 4/5 cut of the dataset HTRU2 (one run is trained on a sample of 4/5 of the dataset and tested on the remaining 1/5)

this dataset. The problem is that most of the signal comes from noise and not pulsar, the goal is then to classify pulsar against noise, using the 17 898 points in the dataset.

The accuracy of different algorithms is obtained using on several runs of the algorithms each using 4/5 of the datasets for training and 1/5 for testing algorithms. Boxplots presenting performance of various algorithms are displayed in Fig. 6. To improve performance, RBF kernel was used both for KLR MOM and Fast KLR MOM.

5.3 Outlier detection with MOM algorithms

When we run MOM version of a descent algorithm, we select at each step a block of data points realizing the median of a set of "local/block empirical risk" at the current iteration of the algorithm. The number of times a point is selected by the algorithm can be used as a depth function measuring reliability of the data. Note that this definition of depth of a data point has the advantage of taking into account the learning task we want to solve, that is the loss ℓ and the class F. It means that outliers are considered w.r.t. the problem we want to solve and not w.r.t. some a priori notion of centrality of points in $\mathbb{R}^d$ unrelated with the problem considered at the beginning.

We apply this idea on the toy dataset with the Logistic Regression MOM algorithm. Results are gathered in a sorted histogram given in Fig. 7. Red bars represent outliers in the original datasets.

Quite remarkably, outliers are in fact those data that have been used the smallest number of times. The method targets a very specific type of outliers, those disturbing the classification task at hand. If there was a point very far away from the bulk of data but in the half-space of its label, it wouldn't be detected.

This detection algorithm doesn't scale well when the dataset gets bigger as a large number of iterations is necessary to choose each point a fair number of times. For bigger datasets, we suggest to adapt usual outlier detection algorithms (Aggarwal 2013). We emphasize that clustering techniques and K-Means are rather easy to adapt in a MOM algorithm and detect points far from the bulk of data. This technique might greatly improve usual K-Means as MOM K-Means is more robust.

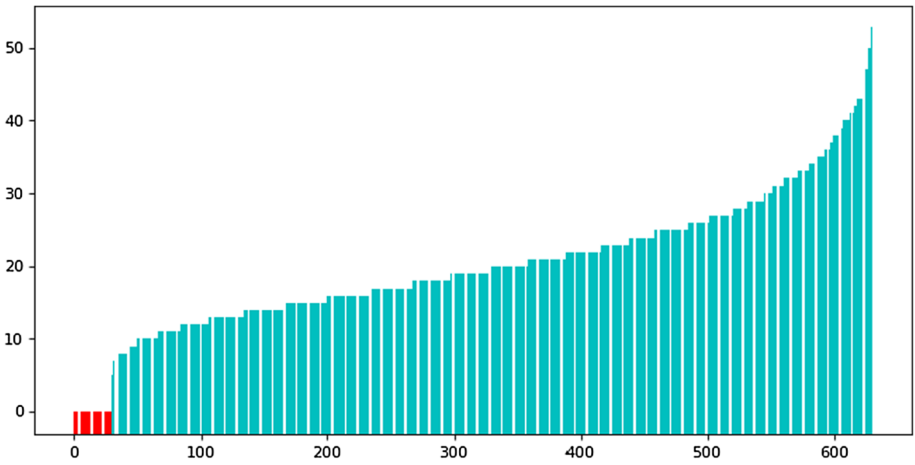


Fig. 7 Sorted Histogram of the score (number of times a data belongs to the selected median block) of each points in a Logistic Regression MOM algorithm on a toy dataset. Red is an outlier and blue is an informative sample. $K = 120$ and $T = 2000$ (Color figure online)

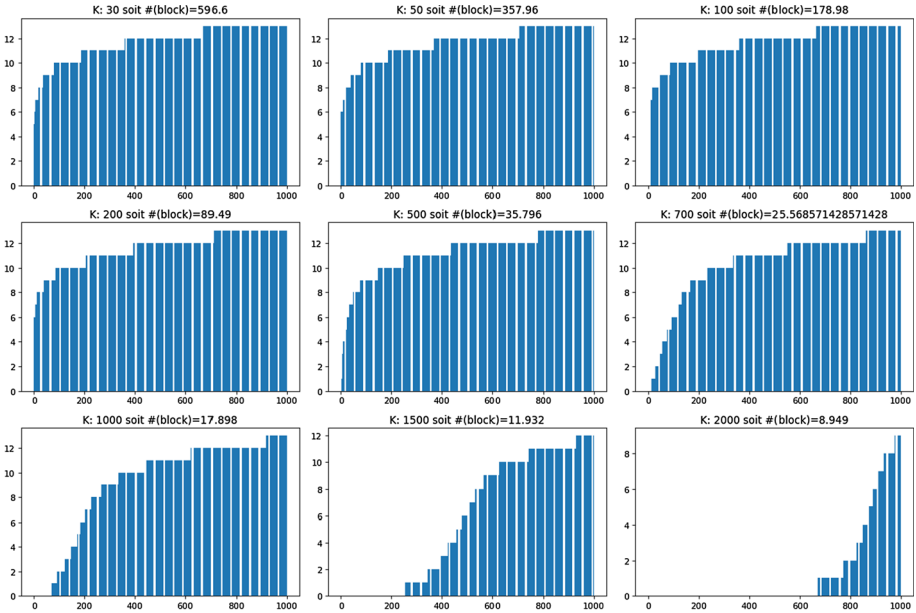


Fig. 8 Sorted Histogram on the score (number of times a data is selected in a median block) of each points in a Logistic Regression MOM algorithm on the pulsar dataset for various values of K and $T = 20 \times K$ (only the 1000 smaller counts among the 17898 sample of the pulsar dataset are represented)

Let us now analyze the effect of K on the outlier detection task. The histogram of the 1000 smaller counts of points of HTRU2 dataset as K gets bigger is plotted in Fig. 8.

It appears from Fig. 8 that K measures the sensitivity of the algorithm. Severe outliers (as in the toy example) are detected for small K while mild outliers are only discovered as K gets bigger.

It seems therefore that the optimal choice of K in MOM depends on the task one is interested in. For classification, K should be as small as possible to get better risk bounds (but it still should be larger than the number of outliers) whereas for detecting outliers we may want to choose K much larger to even detect an outlier, (but it should also be small enough for the underlying classification to perform correctly). As a proof of concept, for Pulsar database, we got optimal results choosing $K = 10$ for classification whereas we only detect a significant amount of outliers when K is around 1000.

6 Proofs

6.1 Proof of Theorem 1

We adapt Vapnik–Chervonenkis's classical analysis (Vapnik 1998) of excess risk bound of ERM to a dataset corrupted by outliers. We first recall that $f^* \in \operatorname{argmin}_{f\in\mathcal{F}} R^{0-1}(f)$ and for all $f \in \mathcal{F}$, the excess loss function of f is $\mathcal{L}_f^{0-1} = \ell_f^{0-1} - \ell_{f^*}^{0-1}$. For simplicity we denote $\hat{f} = \hat{f}_{\mathrm{ERM}}^{0-1}$ and for all $f \in \mathcal{F}$, $\mathcal{L}_f^{0-1} = \mathcal{L}_f$ and $R(f) = R^{0-1}(f)$.

It follows from the definition of the ERM estimator that $P_N\mathcal{L}_{\hat{f}} \leq 0$. Therefore, if we denote by $P_{\mathcal{I}}$ (resp. $P_{\mathcal{O}}$) the empirical measure supported on $\{(X_i, Y_i) : i \in \mathcal{I}\}$ (resp. $\{(X_i, Y_i) : i \in \mathcal{O}\}$), we have

$$\begin{aligned} R(\hat{f}) - R(f^*) &= (P - P_N)\mathcal{L}_{\hat{f}} + P_N\mathcal{L}_{\hat{f}} \leq (P - P_N)\mathcal{L}_{\hat{f}} \\ &= \frac{|\mathcal{I}|}{N}(P - P_{\mathcal{I}})\mathcal{L}_{\hat{f}} + \frac{|\mathcal{O}|}{N}(P - P_{\mathcal{O}})\mathcal{L}_{\hat{f}} \\ &\leq \frac{|\mathcal{I}|}{N}\sup_{f\in\mathcal{F}}(P - P_{\mathcal{I}})\mathcal{L}_f + \frac{2|\mathcal{O}|}{N} \end{aligned}$$

because $|\mathcal{L}_{\hat{f}}| \leq 1$ a.s.. Then, by the bounded difference inequality (Boucheron et al. 2013, Theorem 6.2), since all $f \in \mathcal{F}$ satisfies $-1 \leq \mathcal{L}_f \leq 1$, one has, for any $x > 0$,

$$\mathbb{P}\left(\sup_{f\in\mathcal{F}}(P - P_{\mathcal{I}})\mathcal{L}_f \geq \mathbb{E}[\sup_{f\in\mathcal{F}}(P - P_{\mathcal{I}})\mathcal{L}_f] + x\right) \leq e^{-2|\mathcal{I}|x^2} .$$

Furthermore, by the symmetrization argument (cf. Chapter 4 in Ledoux and Talagrand 2011),

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}}(P - P_{\mathcal{I}})\mathcal{L}_f\right] \leq 2\frac{\mathcal{R}(\mathcal{L}_{\mathcal{F}})}{|\mathcal{I}|} .$$

Therefore, for any $x > 0$, with probability larger than $1 - e^{-2|\mathcal{I}|x^2}$,

$$R(\hat{f}) - R(f^*) \leq \frac{2\mathcal{R}(\mathcal{L}_{\mathcal{F}})}{N} + x + \frac{2|\mathcal{O}|}{N} .$$

The proof is completed by choosing $x = \sqrt{K/(2|\mathcal{I}|)}$.

6.2 Proof of Theorem 2

Let $f^* \in \operatorname{argmin}_{f\in F} P\ell_f$. By definition, one has $\mathrm{MOM}_K\left(\ell_{\widehat{f}_{\mathrm{MOM},K}}\right) \le \mathrm{MOM}_K\left(\ell_{f^*}\right)$, therefore,

$$R(\widehat{f}_{\mathrm{MOM},K}) - R(f^*) \le P\ell_{\widehat{f}_{\mathrm{MOM},K}} - \mathrm{MOM}_K\left(\ell_{\widehat{f}_{\mathrm{MOM},K}}\right) - \left(P\ell_{f^*} - \mathrm{MOM}_K\left(\ell_{f^*}\right)\right) \ . \quad (8)$$

Let us now control the two expressions in the right-hand side of (8). Let $x > 0$. We have

$$\begin{aligned}\mathbb{P}\left[P\ell_{f^*} - \mathrm{MOM}_K\left(\ell_{f^*}\right) > x\right] &= \mathbb{P}\left[\sum_{k=1}^{K} I(P\ell_{f^*} - P_{B_k}\ell_{f^*} > x) \ge \frac{K}{2}\right] \\ &= \sum_{k=K/2}^{K} \binom{K}{k} p^k (1-p)^{K-k} \le p^{K/2} 2^K\end{aligned}$$

where $p = \mathbb{P}[P\ell_{f^*} - P_{B_k}\ell_{f^*} > x]$. Using Markov inequality together with $\mathrm{var}(\ell_{f^*}) \le 2L^2\mathbb{E}(f^*(X))^2 \le 2L^2\theta^2$, we obtain

$$\mathbb{P}\left[P\ell_{f^*} - \mathrm{MOM}_K\left(\ell_{f^*}\right) > x\right] \le \left(\frac{4\mathrm{var}(\ell_{f^*})K}{Nx^2}\right)^{K/2} \le \left(\frac{8L^2\theta^2K}{Nx^2}\right)^{K/2} = \exp(-K/2)$$

when $x = 2L\theta\sqrt{2eK/N}$.

Now, for any $x > 0$, one has $\sup_{f\in F} \mathrm{MOM}_K\left(P\ell_f - \ell_f\right) > x$ iff

$$\sup_{f\in F} \sum_{k=1}^{K} I\left\{(P - P_{B_k})\ell_f > x\right\} \ge \frac{K}{2} \ . \quad (9)$$

Let us now control the probability that (9) holds via an adaptation of the small ball method (Koltchinskii and Mendelson 2015; Mendelson 2015). Let $x > 0$ and let $\phi(t) = (t-1)I\{1 \le t \le 2\} + I\{t \ge 2\}$ be defined for all $t \in \mathbb{R}$. As $\phi(t) \ge I\{t \ge 2\}$, one has

$$\begin{aligned}&\sup_{f\in F} \sum_{k=1}^{K} I\left\{(P - P_{B_k})\ell_f > x\right\} \\ &\le \sup_{f\in F} \sum_{k\in\mathcal{K}} \mathbb{E}\left[\phi\left(2(P - P_{B_k})\ell_f/x\right)\right] + |\mathcal{O}| \\ &+ \sup_{f\in F} \sum_{k\in\mathcal{K}} \left(\phi\left(2(P - P_{B_k})\ell_f/x\right) - \mathbb{E}\left[\phi\left(2(P - P_{B_k})\ell_f/x\right)\right]\right)\end{aligned}$$

where we recall that $\mathcal{K} = \{k \in \{1, \cdots, K\} : B_k \cap \mathcal{O} = \emptyset\}\}$.

Since, $\phi(t) \le I\{t \ge 1\}$ and for all $f \in F$, $\mathrm{Var}(\ell_f) \le 2L^2\mathbb{E}f(X)^2 \le 2L^2\theta_2^2$, we have for all $f \in F$ and $k \in \mathcal{K}$,

$$\mathbb{E}\left[\phi\left(2(P - P_{B_k})\ell_f/x\right)\right] \le \mathbb{P}\left((P - P_{B_k})\ell_f \ge \frac{x}{2}\right) \le \frac{4\mathrm{Var}(\ell_f)}{x^2|B_k|} \le \frac{8L^2\theta_2^2K}{x^2N} \ .$$

One has therefore

$$\sup_{f\in F}\sum_{k=1}^{K} I\left\{(P-P_{B_k})\ell_f > x\right\} \leq K\left(\frac{8L^2\theta_2^2K}{x^2N}+\frac{|\mathcal{O}|}{K}+\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\left(\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)-\mathbb{E}\left[\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)\right]\right)\right) .$$

As $0\leq\phi(\cdot)\leq 1$, by the bounded-difference inequality, for any $y>0$, with probability larger than $1-e^{-2y^2K}$,

$$\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\left(\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)-\mathbb{E}\left[\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)\right]\right) \leq \mathbb{E}\left[\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\left(\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)-\mathbb{E}\left[\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)\right]\right)\right]+y .$$

Now, by the symmetrization inequality,

$$\mathbb{E}\left[\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\left(\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)-\mathbb{E}\left[\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)\right]\right)\right] \leq 2\mathbb{E}\left[\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\epsilon_k\phi\left(\frac{2(P-P_{B_k})\ell_f}{x}\right)\right] .$$

Since ϕ is 1-Lipschitz and $\phi(0)=0$, by the contraction principle (see Ledoux and Talagrand 2011, Chapter 4 or more precisely equation (2.1) in Koltchinskii 2011),

$$\mathbb{E}\left[\sup_{f\in F}\frac{1}{K}\sum_{k\in\mathcal{K}}\epsilon_k\phi\left(\frac{(P-P_{B_k})\ell_f}{x}\right)\right]\leq\mathbb{E}\left[\sup_{f\in F}\frac{1}{xK}\sum_{k\in\mathcal{K}}\epsilon_k(P-P_{B_k})\ell_f\right] .$$

By the symmetrization principle,

$$\mathbb{E}\left[\sup_{f\in F}\frac{2}{xK}\sum_{k\in\mathcal{K}}\epsilon_k(P-P_{B_k})\ell_f\right]\leq\frac{2}{xN}\mathbb{E}\left[\sup_{f\in F}\sum_{i\in\mathcal{J}}\epsilon_i\ell_f(X_i,Y_i)\right] .$$

Finally, since ℓ is L-Lipschitz, by the contraction principle (see equation (2.1) in Koltchinskii 2011),

$$\mathbb{E}\left[\sup_{f\in F}\sum_{i\in\mathcal{J}}\epsilon_i\ell_f(X_i,Y_i)\right]\leq 2L\mathcal{R}(F) .$$

Thus, for any $y>0$, with probability larger than $1-\exp(-2y^2K)$,

$$\sup_{f\in F}\sum_{k=1}^{K} I\left\{(P-P_{B_k})\ell_f > x\right\}\leq K\left(\frac{8L^2\theta_2^2K}{x^2N}+\frac{|\mathcal{O}|}{K}+y+\frac{4L\mathcal{R}(F)}{xN}\right) .$$

Let $\Delta = 1/4 - |\mathcal{O}|/K$ and let $y = \Delta$ and $x = 8L \max\left(\theta_2\sqrt{K/N}, 4\mathcal{R}(F)/N\right)$ so

$$\mathbb{P}\left(\sup_{f\in F}\sum_{k=1}^{K} I\left\{(P - P_{B_k})\ell_f > x\right\} < \frac{K}{2}\right) \geq 1 - e^{-\Delta^2 K/8} .$$

Going back to (9), this means that

$$\mathbb{P}\left(\sup_{f\in F} \mathrm{MOM}_K\left(\ell_f - P\ell_f\right) \leq 4L \max\left(\theta_2\sqrt{\frac{K}{N}}, \frac{4\mathcal{R}(F)}{N}\right)\right) \geq 1 - \exp(-2\Delta^2 K) . \quad (10)$$

Plugging this result in (8) concludes the proof of the theorem.

6.3 Proof of Proposition 1

We denote by $B(1)(u), \ldots, B(K)(u)$ the blocks such that the corresponding empirical means $P_{B(k)(u)}(f_u(X_1^N))$, $k = 1, \ldots, K$ are sorted: $P_{B(1)(u)}(f_u(X_1^N)) \geq \cdots \geq P_{B(K)(u)}(f_u(X_1^N))$. Denote $J \in \mathbb{N}$ such that $K = 2J + 1$.

The goal is to show that $u \mapsto \psi_{f_u}((X_i)_{i=1}^N) = MOM_K(f_u((X_i)_{i=1}^N))$ is differentiable and to compute its partical derivatives. To that end, it suffices to show that for all ε with $\|\varepsilon\|_2$ sufficiently small, we have $B(J)(u) = B(J)(u + t\varepsilon)$ for all $t \in [0, 1]$ and for that it is sufficient to check that the same order of the K empirical means is preserved for all $f_{u+t\varepsilon}$:

$$\forall 1 \leq k \leq K - 1, \forall t \in [0, 1], \quad P_{B(k)(u)}(f_{u+t\varepsilon}) - P_{B(k+1)(u)}(f_{u+t\varepsilon}) > 0. \quad (11)$$

We decompose this difference in three parts,

$$\begin{aligned} P_{B(k)(u)}(f_{u+t\varepsilon}) - P_{B(k+1)(u)}(f_{u+t\varepsilon}) \geq & P_{B(k)(u)}(f_u) - P_{B(k+1)(u)}(f_u) \\ & - \left|P_{B(k)(u)}(f_u) - P_{B(k)(u)}(f_{(u+t\varepsilon)})\right| \\ & - \left|P_{B(k+1)(u)}(f_{u+t\varepsilon}) - P_{B(k+1)(u)}(f_u)\right| \end{aligned}$$

The two last terms are controlled by the Lipshitz property of $u \mapsto f_u$,

$$\begin{aligned} \forall t \in [0, 1], \quad & P_{B(k)(u)}(f_{u+t\varepsilon}) - P_{B(k+1)(u)}(f_{u+t\varepsilon}) \\ & \geq P_{B(k)(u)}(f_u) - P_{B(k+1)(u)}(f_u) - 2tL\|\varepsilon\|_2. \end{aligned}$$

We denote by

$$h_k(\|\varepsilon\|_2) = \mathbb{P}\left(\forall t \in [0, 1], \quad P_{B(k)(u)}(f_u) - P_{B(k+1)(u)}(f_u) - 2tL\|\varepsilon\|_2 \geq 0\right)$$

for all $1 \leq k \leq K - 1$, h_k is an non-decreasing function. Because for all $1 \leq k \leq K$, $P_{B(k)(u)}(f_u)$ has a uniformly continuous law with respect to the Lebesgue measure (because its density is a convolution of several copies of the density of $f_u(X)$), there is no jump in the c.d.f and then h_k verifies that

$$h_k(\|\varepsilon\|_2) \xrightarrow[\|\varepsilon\|_2 \to 0]{} 1.$$

And again because for all $1 \leq k \leq K$, $P_{B(k)(u)}(f_u)$ has a uniformly continuous law with respect to the Lebesgue measure, we also have that

$$h_k(\|\varepsilon\|_2) = \mathbb{P}\left(\forall t \in [0, 1], \quad P_{B(k)(u)}(f_u) - P_{B(k+1)(u)}(f_u) - 2tL\|\varepsilon\|_2 > 0\right).$$

Then, taking the union bound for $1 \leq k \leq K-1$,

$$h(\|\varepsilon\|_2) := \mathbb{P}\left(\forall 1 \leq k \leq K-1, \forall t \in [0,1], \quad P_{B(k)(u)}(f_u) - P_{B(k+1)(u)}(f_u) - 2tL\|\varepsilon\|_2 > 0\right)$$
$$\geq 1 - \sum_{k=1}^{K-1} (1 - h_k(\|\varepsilon\|_2)).$$

Moreover, h can be rewritten as a probability that the blocks don't change using the reasoning leading to Eq. (11), hence

$$h(\|\varepsilon\|_2) = \mathbb{P}(\forall 1 \leq k \leq K-1, \quad B(k)(u) = B(k)(u + t\varepsilon)) \leq \mathbb{P}(\forall t \in [0,1], \\ B(J)(u) = B(J)(u + t\varepsilon)).$$

We now compute the partial derivatives of the median of means ψ_{f_u}. Let $e_1, \ldots, e_p \in \mathbb{R}^p$ be the canonical basis of $\mathbb{R}^p$. For all $m \in \mathbb{N}$, we define $\varepsilon_m^j = \delta_m e_j$ with $(\delta_m)_m$ a decreasing sequence of $\mathbb{R}_+^*$ such that for all $1 \leq k \leq K-1$ we have $h_k(\delta_m) \geq 1 - 2^{-m}$, δ_m exists because $h_k(\delta) \to 1$ when $\delta \to 0$. Then,

$$h(\|\varepsilon_m^j\|_2) \geq 1 - K2^{-m}. \tag{12}$$

We denote by A_m^j the event $A_m^j := \left\{\forall t \in [0,1], \quad B_{(J)}(u) = B_{(J)}(u + t\varepsilon_{m'}^j)\right\}$ and we study the limiting event $\Omega^j = \overline{\lim}_{m\to\infty} A_m^j$.

First, let us note that for all $1 \leq j \leq p$, the sequence of set $(A_m^j)_n$ is non-increasing, hence

$$\Omega^j = \overline{\lim}_{m\to\infty} A_m^j = \underline{\lim}_{m\to\infty} A_m^j = (\overline{\lim}_{m\to\infty} (A_m^j)^c)^c,$$

then, for all $1 \leq j \leq d$, we can study the $\overline{\lim}_{m\to\infty}(A_m^j)^c$ with Borel-Cantelli Lemma. Indeed, we have from Eq. (12), $\mathbb{P}((A_m^j)^c) \leq K2^{-m}$. Hence, the series $\sum_m \mathbb{P}((A_m^j)^c)$ converges and by Borel Cantelli Lemma, $\mathbb{P}(\overline{\lim}_{m\to\infty}(A_m^j)^c) = 0$, then for all $1 \leq i \leq p$, $\mathbb{P}(\Omega^j) = 1$. In other words, we have that for all $\omega \in \Omega^j$, there exists $m \geq 1$ such that $\omega \in A_m^j$. Hence, there exists $m \geq 1$ such that for all $t \in [0,1]$, $B(J)(u) = B(J)(u + t\varepsilon_m^j)$, which implies that for all $1 \leq j \leq p$,

$$\partial_j \psi_{f_u}(X) = \lim_{t\to 0} \frac{\psi_{f_{u+t\epsilon_m^j}}(X) - \psi_{f_u}(X)}{t} = \lim_{t\to 0} \frac{P_{B(J)(u)}(f_{u+t\varepsilon_m^j}) - P_{B(J)(u)}(f_u)}{t}$$
$$= \frac{1}{N/K} \lim_{t\to 0} \sum_{i \in B(J)(u)} \frac{f_{u+t\varepsilon_m^j}(X_i) - f_u(X_i)}{t} = \frac{1}{N/K} \sum_{i \in B(J)(u)} \partial_j f_u(X_i).$$

7 Annex

7.1 Choice of the number of blocks

Let us study the behaviour of our algorithms when the number of blocks changes. We plot the accuracy as a function of K averaged on 50 runs to have a good idea of the evolution of the performance with respect to K, the result is represented in Fig. 9.

There is a clear separation around $2|\mathcal{O}| = 60$ that is consistent with the theory. On the other hand the accuracy doesn't decrease when K gets bigger one would expect. This may

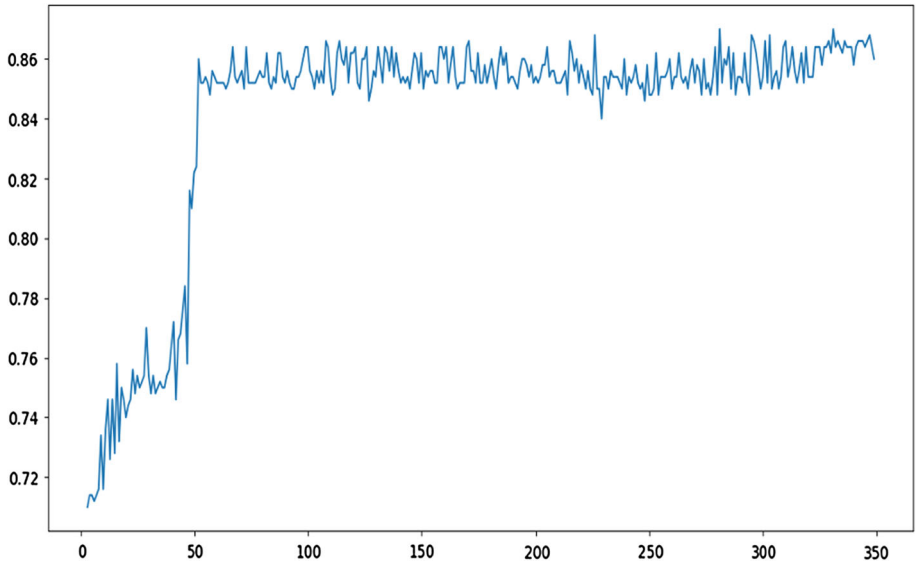


Fig. 9 Plot of the accuracy on the toy dataset of Logistic Regression MOM as a function of K

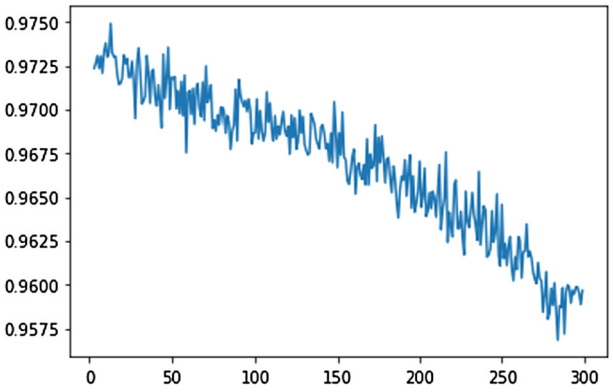


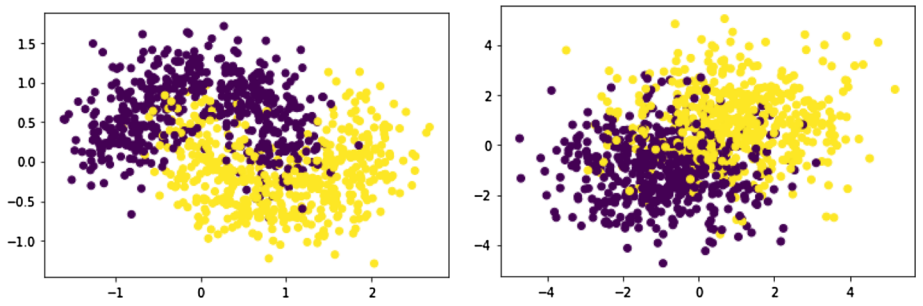
Fig. 10 Plot of the accuracy on HTRU2 dataset of Logistic Regression MOM as a function of K

be due to the symmetry of the dataset. If we run the same experiment on the real dataset, we get a much more regular plot, see Fig. 10.

Figure 10 confirms our predictions on clean datasets, the accuracy getting better as K gets smaller (the MOM minimizer is the ERM estimator when $K = 1$ and ERM is optimal in the i.i.d. setup, Lecué and Mendelson 2013). This may be due to the small number of outliers in this dataset.

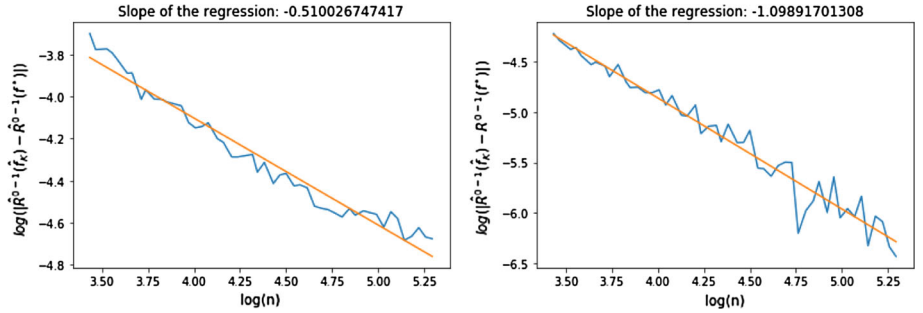
7.2 Illustration of convergence rate

In this section, we estimate the rate of convergence of the MOM risk minimization algorithm Logistic Regression on two databases (see Fig. 11). The first dataset is composed of points located on two interlaced half-circle with a Gaussian noise of standard deviation 0.3, the



(a) Scatter plot of the Moons dataset **(b)** Scatter plot of the Gaussians dataset

Fig. 11 Scatter plot of the two dataset used in this section, the color represent the class of the points (Color figure online)



(a) Convergence rate for the Moons dataset **(b)** Convergence rate for the Gaussians dataset

Fig. 12 Plot of the logarithm of the excess risk as a function of $\log(n)$ in two cases: **a** where the margin assumption does not hold and **b** where the margin assumption holds. A linear regression is fitted on the curve, its slope is printed at the top of each figure revealing a slow $n^{-0.51}$ rate of convergence in case **a** and a fast $n^{-1.1}$ in case **b**

two "moons" are each of a different class. We assume that these moons don't satisfy the margin property (we checked that the rate was slow for ERM algorithms, using the vanilla logistic regression). The second dataset is composed of two Gaussians $\mathcal{N}((-1,-1), 1.4^2 I_2)$ and $\mathcal{N}((1,1), 1.4^2 I_2)$ with respective label 1 and 0, we can prove that this dataset verifies the margin property needed to obtain fast rate in ERM

There are no outliers in the datasets because we only want to test the rate of convergence. To illustrate the rates of convergence of our algorithms, we plot the curve $\log\left(\left|\hat{R}^{0-1}(\hat{f}_K)) - \hat{R}^{0-1}(f^*)\right|\right)$ as a function of $\log(n)$ where the risk is estimated by Monte-Carlo. The figure obtained for Logistic Regression MOM is represented in Fig. 12. It seems that MOM minimizers can achieve fast rates of convergence even if we did not prove them.

Remark 5 We used random blocks sampled at each iteration for this application because it is the algorithm that we described earlier but even if we use one partition of blocks for the whole algorithm (as in the theory we developed) we obtain nonetheless fast rate for the Gaussians dataset.

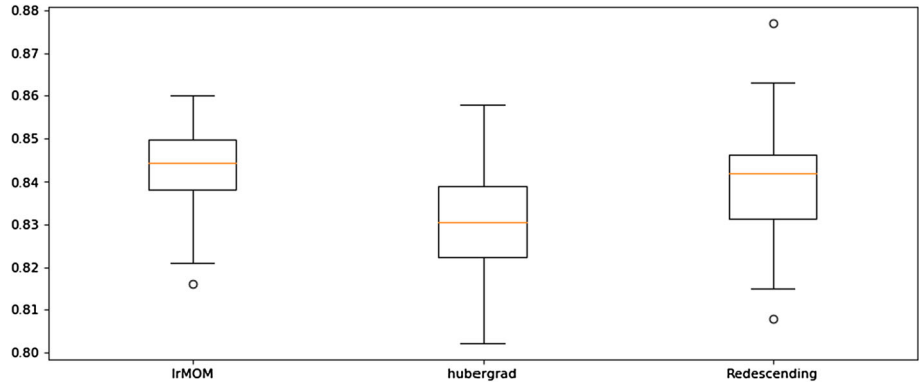


Fig. 13 Boxplot of the accuracy obtained on 50 training/test run (1000 training sample, 2% corruption) of each algorithms on a 2-dimensional toy dataset

7.3 Comparison with robust algorithms based on M-estimators

In this section we compare the algorithm Logistic Regression MOM with two other algorithms based on M-estimators, these algorithms are studied on the toy dataset presented in Section 5.

One algorithm is a gradient on the Huber estimation of the loss function, it follows the same reasoning as MOM risk minimization and minimizes $\mathbb{E}[l_f(X, Y)]$ using as a proxy the Huber estimator for this quantity. The Huber estimator is then defined as a M-estimator, denoted here $\hat{\mu}_f$, solution of

$$\sum_{i=1}^{n} \psi_c(\hat{\mu}_f - l(f(X_i), Y_i)) = 0$$

where $\psi_c = \max(-c, \min(c, x))$ is the Huber function, $c > 0$. Using this definition of $\hat{\mu}_f$, it is then easy to compute the gradient $\nabla\hat{\mu}_f$ and then use a gradient descent algorithm. The theory behind this algorithm is studied further in Brownlees et al. (2015).

The second algorithm uses a "redescending" loss function, in short we do ERM with a bounded loss function. Here we use Tukey biweight loss function rescaled by MADN scale estimator and IRLS algorithm to optimize the empirical risk.

Figure 13 shows that all algorithms perform similarly on this easy, low dimensional dataset. The situation is quite different in higher dimension. In Fig. 14 we used a 200 dimensional dataset and the algorithm using a redescending loss function does not perform well. This may be due to local minima in which the algorithm gets stuck, as local minima are multiplied when the dimension gets higher. The other algorithms don't suffer this drawback since they use a "projection by the loss function" that makes the problem one dimensional.

The algorithm using redescending loss functions is a simple gradient descent that has linear complexity. The Huber gradient algorithm estimates at each iteration a Huber estimator of location. The complexity of this estimator depends on the algorithm used but for most M-estimators a commonly used algorithm is an iteratively reweighted algorithm whose complexity is linear in the sample size. In practice we can nonetheless notice a great complexity of the Huber estimator in some cases where data are not well spread. In most cases, Logistic Regression MOM is the fastest among these three algorithms and the gradient Huber is the slowest, even though logistic regression may need a lot more iterations than the other algorithms.

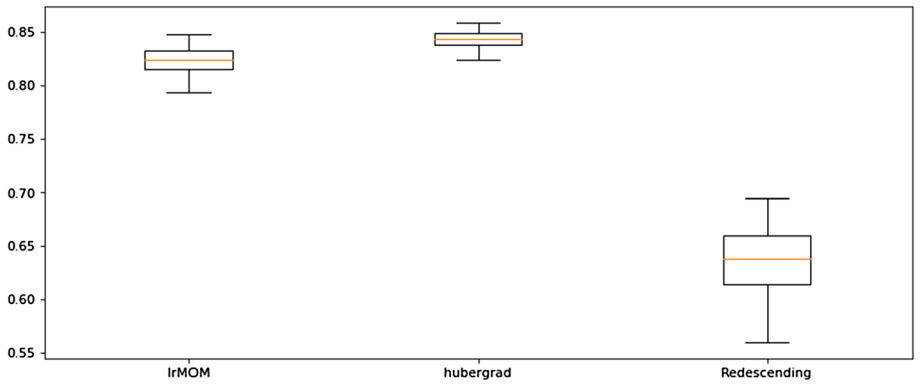


Fig. 14 Boxplot of the accuracy obtained on 50 training/test run (2000 training sample, 2% corruption) of each algorithms on a 200-dimensional toy dataset

References

Aggarwal, C. C. (2013). *Outlier analysis*. Berlin: Springer.

Alon, N., Matias, Y., & Szegedy, M. (1999). The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences, 58*(1, part 2), 137–147. *Twenty-eighth Annual ACM Symposium on the Theory of Computing* (Philadelphia, PA, 1996).

Arora, S., & Barak, B. (2009). *Computational complexity*. Cambridge: Cambridge University Press. A modern approach.

Audibert, J.-Y., & Catoni, O. (2011). Robust linear least squares regression. *Annals of Statistics*, *39*(5), 2766–2794.

Bach, F., Jenatton, R., Mairal, J., & Obozinski, G. (2012). Structured sparsity through convex optimization. *Statistical Science*, *27*(4), 450–468.

Baraud, Y., Birgé, L., & Sart, M. (2017). A new method for estimation and model selection: ρ-estimation. *Inventiones Mathematicae*, *207*(2), 425–517.

Bartlett, P. L., & Mendelson, S. (2002). Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research, 3*(Spec. Issue Comput. Learn. Theory), 463–482.

Bartlett, P. L., & Mendelson, S. (2006). Empirical minimization. *Probability Theory and Related Fields*, *135*(3), 311–334.

Birant, D., & Kut, A. (2007). St-dbscan: An algorithm for clustering spatial-temporal data. *Data & Knowledge Engineering*, *60*(1), 208–221.

Boucheron, S., Bousquet, O., & Lugosi, G. (2005). Theory of classification: a survey of some recent advances. *ESAIM: Probability and Statistics*, *9*, 323–375.

Boucheron, S., Lugosi, G., & Massart, P. (2013). *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge: Cambridge University Press.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140.

Brownlees, C., Joly, E., & Lugosi, G. (2015). Empirical risk minimization for heavy-tailed losses. *Annals of Statistics*, *43*(6), 2507–2536. 12.

Bubeck, S. (2015). Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, *8*(3–4), 231–357.

Bühlmann, P., & Bin, Y. (2002). Analyzing bagging. *The Annals of Statistics*, *30*(4), 927–961.

Catoni, O. (2012). Challenging the empirical mean and empirical variance: A deviation study. *Annales de Institut Henri Poincaré Probability and Statistics*, *48*(4), 1148–1185.

Chen, M., Gao, C., & Ren, Z. (2017). *Robust covariance and scatter matrix estimation under huber's contamination model*. Technical report, University of Chicago and University of Pittsburgh. Preprint available on arXiv:1506.00691.

Chen, M., Gao, C., & Ren, Z. (2018). Robust covariance and scatter matrix estimation under Huber's contamination model. *Annals of Statistics*, *46*(5), 1932–1960.

Cheng, Y., Diakonikolas, I., & Ge, R. (2019). High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the thirtieth annual ACM-SIAM symposium on discrete algorithms* (pp. 2755–2771). SIAM.

Chinot, G., Lecué, G., & Lerasle, M. (2019). Robust statistical learning with Lipschitz and convex loss functions. *Probability Theory and Related Fields* (to appear).

Christophe, C., & Catherine, D. (2001). Robust linear discriminant analysis using s-estimators. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, *29*(3), 473–493.

Devroye, L., Györfi, L., & Lugosi, G. (1997). *A probabilistic theory of pattern recognition*, volume 31 of *Applications of Mathematics*. Springer, corrected 2nd edition, missing.

Devroye, L., Lerasle, M., Lugosi, G., & Oliveira, R. I. (2016a). Sub-Gaussian mean estimators. *Annals of Statistics*, *44*(6), 2695–2725.

Devroye, L., Lerasle, M., Lugosi, G., & Oliveira, R. I. (2016b). Sub-Gaussian mean estimators. *The Annals of Statistics*, *44*(6), 2695–2725.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., & Stewart, A. (2016). Robust estimators in high dimensions without the computational intractability. In *57th Annual IEEE symposium on foundations of computer science—FOCS 2016* (pp. 655–664). IEEE Computer Soc., Los Alamitos, CA.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., & Stewart, A. (2017). Being robust (in high dimensions) can be practical. In *Proceedings of the 34th international conference on machine learning-volume 70* (pp. 999–1008). JMLR.org.

Donoho, D., & Montanari, A. (2015). Variance breakdown of huber (m)-estimators: $n/p \to m \in (1, +\infty)$. *Technical report*, Stanford University, Preprint available on arXiv:1503.02106.

Fan, J., & Kim, D. (2018). Robust high-dimensional volatility matrix estimation for high-frequency factor model. *Journal of the American Statistical Association*, *113*(523), 1268–1283.

Feldman, V., Guruswami, V., Raghavendra, P., & Yi, W. (2012). Agnostic learning of monomials by halfspaces is hard. *SIAM Journal on Computing*, *41*(6), 1558–1590.

Gao, C. (2017). *Robust regression via mutivariate regression depth*. Technical report, University of Chicago, Preprint available on arXiv:1702.04656.

Gao, C., Liu, J., Yao, Y., & Zhu, W. (2018). *Robust estimation and generative adversarial nets*. arXiv:1810.02030.

Guruswami, V., & Raghavendra, P. (2009). Hardness of learning halfspaces with noise. *SIAM Journal on Computing*, *39*(2), 742–765.

Gunduz, N., & Fokoué, E. (2015). *Robust classification of high dimension low sample size data*. arXiv:1501.00592.

Hampel, F. R. (1971). A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, *42*(6), 1887–1896.

Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, *69*(346), 383–393.

Han, Q., & Wellner, J. (2017). *A sharp multiplier inequality with applications to heavy-tailed regression problems*. arXiv:1706.02410.

He, X., & Fung, W. K. (2000). High breakdown estimation for multiple populations with applications to discriminant analysis. *Journal of Multivariate Analysis*, *72*(2), 151–162.

Hoare, C. A. R. (1962). Quicksort. *The Computer Journal*, *5*(1), 10–16.

Huber, P. J. (1964). Robust estimation of a location parameter. *Annals of Mathematical Statistics*, *35*, 73–101.

Huber, P. J. (1967). The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (vol. 1, pp. 221–233). Berkeley, CA.

Huber, P. J, & Ronchetti, E. M. (2009). *Robust statistics*, 2nd ed. Wiley Series in Probability and Statistics. Wiley, Hoboken, NJ.

Hubert, M., & Van Driessen, K. (2004). Fast and robust discriminant analysis. *Computational Statistics & Data Analysis*, *45*(2), 301–320.

Hubert, M., & Van Der Veeken, S. (2010). Robust classification for skewed data. *Advances in Data Analysis and Classification*, *4*(4), 239–254.

Jerrum, M. R., Valiant, L. G., & Vazirani, V. V. (1986). Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, *43*(2–3), 169–188.

Jordan, M. I. (2013). On statistics, computation and scalability. *Bernoulli*, *19*(4), 1378–1390.

Koltchinskii, V. (2008). *Oracle inequalities in empirical risk minimization and sparse recovery problems, volume 2033 of Lecture Notes in Mathematics*. Springer, Heidelberg, 2011. Lectures from the 38th Probability Summer School held in Saint-Flour, 2008, École d'Été de Probabilités de Saint-Flour. [Saint-Flour Probability Summer School].

Koltchinskii, V. (2011). *Oracle inequalities in empirical risk minimization and sparse recovery problems*.

Koltchinskii, V., & Mendelson, S. (2015). Bounding the smallest singular value of a random matrix without concentration. *International Mathematics Research Notices*, *23*, 12991–13008.
Lecué, G., & Lerasle, M. (2017). *Robust machine learning by median-of-means: Theory and practice*. Technical report, CNRS, ENSAE, Paris-sud. Preprint available on arXiv:1711.10306.
Lecué, G., & Lerasle, M. (2019). *Learning from mom's principle : Le cam's approach*. Technical report, CNRS, ENSAE, Paris-sud. Preprint available on arXiv:1701.01961.
Lecué, G., & Mendelson, S. (2013). *Learning subgaussian classes: Upper and minimax bounds*. Technical report, CNRS, Ecole polytechnique and Technion.
Ledoux, M., & Talagrand, M. (2011). *Probability in Banach spaces. Classics in Mathematics*. Springer, Berlin, 2011. Isoperimetry and processes, Reprint of the 1991 edition.
Le Gall, F. (2014). Powers of tensors and fast matrix multiplication. *CoRR*, arXiv:1401.7714.
Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *Eighth IEEE international conference on data mining, 2008. ICDM'08* (pp. 413–422). IEEE.
Lugosi, G., & Mendelson, S. (2017). *Risk minimization by median-of-means tournaments*. Preprint available on arXiv:1608.00757.
Lugosi, G., & Mendelson, S. (2019a). *Regularization, sparse recovery, and median-of-means tournaments*. Preprint available on arXiv:1701.04112.
Lugosi, G., & Mendelson, S. (2019b). *Sub-gaussian estimators of the mean of a random vector*. Preprint available on arXiv:1702.00482.
Lyon, R. J., Stappers, B. W., Cooper, S., Brooke, J. D., & Knowles, J. M. (2015). Fifty years of pulsar candidate selection: From simple filters to a new principled real-time classification approach. *MNRAS*, *000*, 000–000.
Mammen, E., & Tsybakov, A. B. (1999). Smooth discrimination analysis. *Annals of Statistics*, *27*(6), 1808–1829.
Mendelson, S. (2014). Learning without concentration. In *Proceedings of the 27th annual conference on Learning Theory COLT14* (pp. 25–39).
Mendelson, S. (2015). Learning without concentration. *J. ACM*, 62(3):Art. 21, 25.
Mendelson, S. (2017). *An optimal unrestricted learning procedure*. Preprint available on arXiv:1707.05342.
Minsker, S. (2015). Geometric median and robust estimation in Banach spaces. *Bernoulli*, *21*(4), 2308–2335.
Minsker, S. (2019). Distributed statistical estimation and rates of convergence in normal approximation. *Electronic Journal of Statistics*, *13*(2), 5213–5252.
Moulines, E., & Bach, F. R. (2011). Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Advances in neural information processing systems* (pp. 451–459).
Nemirovsky, A. S., & Yudin, D. B. (1983). *Problem complexity and method efficiency in optimization*. A Wiley-Interscience Publication. Wiley, New York. Translated from the Russian and with a preface by E. R. Dawson, Wiley-Interscience Series in Discrete Mathematics.
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.
Rosenblatt, F. (1958). *The perceptron: A theory of statistical separability in cognitive systems*. Cornell Aeronautical Laboratory, Inc., Rep. No. VG-1196-G-1. U.S. Department of Commerce, Office of Technical Services, PB 151247.
Roth, V. (2001). Probabilistic discriminative kernel classifiers for multi-class problems. In *Joint pattern recognition symposium* (pp. 246–253). Springer.
Saumard, A. (2018). On optimality of empirical risk minimization in linear aggregation. *Bernoulli*, *24*(3), 2176–2203.
Tukey, J. W. (1960). A survey of sampling from contaminated distributions. *Contributions to Probability and Statistics*, *2*, 448–485.
Tukey, J. W. (1962). The future of data analysis. *The Annals of Mathematical Statistics*, *33*(1), 1–67.
Vapnik, V. N. (1998). *Statistical learning theory. Adaptive and Learning Systems for Signal Processing, Communications, and Control*. Wiley, New York.
Vapnik, V. N. (2000). *The nature of statistical learning theory. Statistics for Engineering and Information Science* (second ed.). New York: Springer.