Set-Valued and Variational Analysis manuscript No.
(will be inserted by the editor)

Superquantiles at Work: Machine Learning Applications and Efficient Subgradient Computation

Yassine Laguel · Krishna Pillutla · Jérôme Malick · Zaid Harchaoui

submission to special issue for T. Rockafellar

Abstract R. Tyrell Rockafellar and his collaborators introduced, in a series of works, new regression modeling methods based on the notion of superquantile (or conditional value-at-risk). These methods have been influential in economics, finance, management science, and operations research in general. Recently, they have been subject of a renewed interest in machine learning, to address issues of distributional robustness and fair allocation. In this paper, we review some of these new applications of the superquantile, with references to recent developments. These applications involve nonsmooth superquantile-based objective functions that admit explicit subgradient calculations. To make these superquantile-based functions amenable to the gradient-based algorithms popular in machine learning, we show how to smooth them by infimal convolution and detail numerical procedures to compute the gradients of the smooth approximations. We put the approach into perspective by comparing it to other smoothing techniques and by illustrating it on toy examples.

Yassine Laguel
Univ. Grenoble Alpes, Grenoble INP, LJK, Grenoble, France
E-mail: yassine.laguel@univ-grenoble-alpes.fr

Krishna Pillutla
University of Washington, Seattle, USA
E-mail: pillutla@cs.washington.edu

J. Malick
Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, Grenoble, France
E-mail: jerome.malick@univ-grenoble-alpes.fr

Z. Harchaoui
University of Washington, Seattle, USA
E-mail: zaid@uw.edu

1 Introduction

1.1 Superquantiles at Work: Old and New

Risk measures play a crucial role in optimization under uncertainty, involving problems with an aversion to worst-cases scenarios. Among popular convex risk measures, the superquantile – also called the Conditional Value at Risk, Tail Value at Risk, Mean Excess Loss, or Mean Shortfall – has received special attention. The superquantile has been extensively studied from a convex analysis perspective: we refer for instance to [44] for a variational formulation of the superquantile, to [5] for its generalization to a larger class of risk measures, to [12] for a dual formulation (also later generalized in [46] or [43]) and [41] for additional convex properties. The superquantile can be traced back to the paper [4]. These nice theoretical properties have given interesting results in various applications, ranging from finance [47] to energy planning [13]; for a thorough discussion and many references, we refer to the seminal work [44], the classical textbook [50, Chap. 6], or the tutorial paper [40].

More recently, the superquantile has also drawn an increasing attention in machine learning. In this paper, we give an overview of some of the new applications of the superquantile in machine learning: we discuss the use of the superquantile for distributionally robust learning, fair learning, federated learning, adversarial classification, and reinforcement learning; we also give toy illustrations and pointers to recent exciting developments.

Superquantile optimization problems are nonsmooth, possibly non-convex, but also highly structured. In financial or operations research applications, these nonsmooth optimization problems are usually solved using one of two approaches: (a) extending specific algorithms (e.g., progressive hedging for risk-averse multi-stage programming [39]), or, (b) relying on convex programming (e.g., linear programming coupled with Monte Carlo simulations for portfolio management [44]). We refer to [42] and [29] for discussions on computational approaches. In machine learning, recent papers propose to use stochastic first-order optimization algorithms for superquantile learning; see e.g., [8, 25] and references therein.

In this paper, we propose a simple alternative. We study the smoothing of superquantile by infimal-convolution, extending and clarifying the results of [22, Sec. 3]. This opens the way for using first-order methods for smooth optimization: this is of special interest for machine learning applications where standard algorithms and software rely heavily rely on gradient-based optimization [1, 35]. In fact, optimization guarantees in this context are typically given for smooth surrogates of the superquantile, e.g., [23, 25]; which we study and clarify. In view of these applications, we pay attention to provide efficient procedures for computing gradients of smooth approximations of superquantile-based functions. We illustrate these smoothed oracles combined with quasi-Newton methods on simple problems with synthetic or real data.

We refer to our recent work [22, 23] for more computational experiments, using particular cases of such efficient smoothed oracles.

More specifically, the contributions of this paper are multiple and can be pointed out, section by section, as follows:

- We formalize, in Section 2, the existing notion of empirical superquantile minimization and provide a convergence result of supervised learning.
- We propose, in Section 3, an overview of recent machine learning applications of superquantiles.
- We study in Section 4 the (sub)gradient calculus of superquantile-based functions with a focus on computational efficiency. In particular, Section 4.2 studies generalized subgradients of superquantile-based functions and Section 4.3 considers gradients of smooth approximations of the superquantile by inf-convolution. Finally, we establish in Section 4.4 the equivalence between different inf-convolution schemes, as well as the smoothing by convolution. We propose to use quasi-Newton algorithms to minimize these smoothed approximations.

1.2 Superquantiles: Review and Notation

We recall basic definitions and properties used in this paper. Our notation and terminology follow closely the ones of [46] and [40]; we refer to these papers for more details and references.

Consider a probability space Ω, with probability denoted $\mathbb{P}$. For $p \in (0,1)$, the p-quantile of a random variable $U\colon \Omega \to \mathbb{R}$, denoted by $Q_p(U)$, is the inverse of the cumulative distribution function of U: for all $t \in \mathbb{R}$ we have

$$Q_p(U) \le t \iff \mathbb{P}(U \le t) \ge p\,. \tag{1}$$

When e.g., $p = 1/2$, the p-quantile corresponds the median value of the random variable. For $p \in [0,1)$, the p-superquantile of U is then defined as the mean of values of quantiles greater than the threshold p:

$$\mathbb{S}_p(U) = \frac{1}{1-p} \int_p^1 Q_{p'}(U)\mathrm{d}p'\,. \tag{2}$$

The analogue to (1) for the superquantile is stronger:

$$\mathbb{S}_p(U) \le t \iff U \text{ is lower than } t \text{ on average in its } p\text{-tail.}$$

The superquantile is thus interpreted as a measure of the upper tail of the distribution of U. Another interpretation comes from the dual formulation of superquantiles [12]: $\mathbb{S}_p(U)$ can be written as a maximal expectation of U

with respect to probability measures having a (Radon-Nykodim) derivative bounded by $1 - p$

$$\mathbb{S}_p(U) = \max_{\substack{0 \le q(\cdot) \le \frac{1}{1-p} \\ \int_\Omega q \, d\,\mathbb{P}(\omega) = 1}} \int_{\omega \in \Omega} U(\omega) q(\omega) \mathrm{d}\mathbb{P}(\omega)\,. \tag{3}$$

When U is a discrete random variable, the above expression simplifies; we come back to this in Section 4. Finally, for an optimization perspective, the superquantile also has a nice variational formulation [44].

$$\mathbb{S}_p(U) = \min_{\eta \in \mathbb{R}} \left\{ \eta + \frac{1}{1-p} \mathbb{E}[\max(U - \eta, 0)] \right\}. \tag{4}$$

In this expression, the quantile $Q_p(U)$ is obtained as the left end-point of the solution. This last expression also reveals an important advantage of $\mathbb{S}_p(U)$ over $Q_p(U)$ as a measure of the tail of U, from both theoretical and practical points of view: the superquantile is convex, positively homogeneous, monotonic, translation invariant; see, e.g.,, the tutorial article [40].

2 Standard and Superquantile Machine Learning

Optimization is at the heart of machine learning, through the paradigm of empirical risk minimization, which we briefly recall in Section 2.1. In Section 2.2, we discuss superquantile learning, where the risk measure of the learning model is the superquantile. The material of this section also serves as a gentle introduction to the recent developments outlined in the next section.

2.1 Supervised Learning Review

We recall here the notation and basic notions of supervised learning; we refer to standard textbooks [7] or [49] for more details. In the training phase of supervised learning, we have access to n data-points: each data-point is a pair (x, y), where $x \in X$ is a feature vector and $y \in Y$ is its corresponding target. For instance, for a binary classification task, y is a Boolean encoding the membership of the image x to one of the two classes. From this training data, the aim is to learn a parameter $w \in W \subset \mathbb{R}^{\mathrm{d}}$ as "weights" of a given prediction function $z = \varphi(w, x)$ that produces, for an input $x \in X$, a prediction $z \in Z$ of the associated target $y \in Y$. Typical examples of prediction functions include simple linear models $\varphi(w, x) = w^\top x$, polynomial models (as in Example 1 below), or artificial neural networks

$$\varphi(w, x) = w_s^\top \sigma(\cdots \sigma(w_1^\top x))\,, \tag{5}$$

which are successive compositions of linear models w_j and non-linear activations σ. The prediction error is then measured by a loss function $\ell : Y \times Z \to \mathbb{R}$.

Typical examples of loss functions include the least-squares loss ($Y = \mathbb{R}, Z = \mathbb{R}$) or the logistic loss ($Y = \{-1, 1\}, Z = \mathbb{R}$), defined respectively as

$$\ell(y, z) = \frac{1}{2}(y - z)^2 , \qquad \text{and,} \qquad \ell(y, z) = \log(1 + \exp(-y\, z)) . \tag{6}$$

Assuming[1] that the training data are generated from a given distribution P over $X \times Y$, the "best" model parameter w solves the optimization problem

$$\min_{w \in W} \left[R(w) = \ \mathbb{E}_{(x,y)\sim P} \left[\ell(y, \varphi(w, x)) \right] \right] . \tag{7}$$

However, we can only access P via i.i.d. samples $\{(x_i, y_i)\}_{1 \le i \le n}$. So we consider instead the empirical risk minimization approach, which solves the following optimization problem, analogous to (7) but where the expectation is taken over P_n, the empirical measure over the training examples:

$$\min_{w \in W} \left[R_n(w) = \ \mathbb{E}_{(x,y)\sim P_n} \left[\ell(y, \varphi(w, x)\right] = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \varphi(w, x_i) \right] . \tag{8}$$

Under suitable conditions, we have that the minimizer $w_n^\star$ of (8) converges almost surely in mean error to the best population error as $n \to \infty$, i.e.,

$$R(w_n^\star) \underset{n \to \infty}{\longrightarrow} R(w^\star) \qquad \text{almost surely.} \tag{9}$$

For concreteness, we instantiate this general framework with a simple regression task, which will also be used in illustrations in subsequent sections.

Example 1 (Least-squares regression) Consider a dataset $D = (x_i, y_i)_{1 \le i \le n} \in (\mathbb{R} \times \mathbb{R})^n$ generated by noisy observations of a quadratic function: we have

$$y_i = \bar{w}_0 + \bar{w}_1 \, x_i + \bar{w}_2 \, x_i^2 + \varepsilon_i \, , \qquad \text{where } \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \, , \tag{10}$$

for an unknown vector $\bar{w} = (\bar{w}_0, \bar{w}_1, \bar{w}_2) \in \mathbb{R}^3$ that we would like to approximate. In this case, (8) instantiates as the ordinary least-squares problem

$$\min_{w \in \mathbb{R}^3} \mathbb{E}_{(x,y)\sim P_n} \left[(y_i - (w_2 x_i^2 + w_1 x_i + w_0))^2 \right] , \tag{11}$$

with a quadratic model $\varphi(w, \cdot)$ and the square loss. □

[1] When the assumption of the existence of an underlying distribution P is not realistic, the usual approach is to still use the empirical risk minimization (8) from the given training dataset $P_n = \{(x_i, y_i)\}_{1 \le i \le n}$.

2.2 Superquantile Learning

The standard framework, recalled above, is currently challenged by important domain applications [e.g., 18, 38], in which several of the standard assumptions turn out to be limiting. Indeed classical supervised learning assumes that, at training time, the examples $(x_1, y_1), \ldots, (x_n, y_n)$ are drawn i.i.d. from a given distribution P, and that, at testing time, we face a new example x', also drawn from the same distribution P. However, recent failures of learning systems when operating in unknown environments [21, 27] underscore the importance of taking into account that we may not face the same distribution at test/prediction time.

Distributionally robust learning aims to bolster the safety of learning systems by enforcing robustness to heterogeneous data. This notion of robustness is aligned with the one in robust optimization [3]; it is, however, different from the notion of robustness in robust statistics [3, Sec. 12.6]. Here, we assume that the dataset is preprocessed to remove outliers such that the extreme data in the dataset is relevant to the learning process.

The superquantile can be used to build distributionally robust machine learning models, as studied recently in [8, 22, 25, 51] among others. From the dual formulation (3), superquantiles are expected to produce models that perform better in case of changes in underlying distributions, compared to models trained using standard empirical risk minimization. Therefore, a natural approach to distributionally robust learning consists in replacing the expectation in (7) by the superquantile (2). The resulting objective function is

$$\min_{w \in W} \left[S^p(w) = [\mathbb{S}_p]_{(x,y)\sim P} \left[\ell(y, \varphi(w,x)) \right] \right] ,$$

as well as its empirical version analogous to (8)

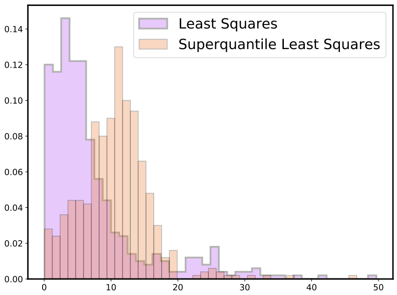
$$\min_{w \in W} \left[S_n^p(w) = [\mathbb{S}_p]_{(x,y)\sim P_n} \left[\ell(y, \varphi(w,x)) \right] \right] . \tag{12}$$

As we establish in the forthcoming Theorem 1, the convergence property (9) also holds for S_n^p and S^p. We refer to [10, 48] for further discussions on statistical aspects of distributionally robust learning, and to [24, 28, 51] for superquantile learning in particular.

In practice, superquantile has been shown experimentally to produce models more robust to distribution shifts in various contexts; we refer to [8, 11, 20, 22, 25, 51]. For illustration, we include numerical experiments inspired from the ones of [8] in the Appendix. We also include a short toy example here.

Example 2 (Superquantile regression) We illustrate the interest of superquantile learning in presence of heterogeneous data, on a variant of the regression task of Example 1. Consider a dataset gathering two different subgroups: 80% of the points are generated by (10) and the remaining 20% are also generated by (10) but with completely different parameters $\bar{w}$. Then we can compare the usual approach using ordinary least-squares (11) with its superquantile counterpart of the form (12) for $p = 0.9$.

We report on Figure 1 the distribution of residuals $r_i = |y_i - (w_2 x_i^2 + w_1 x_i + w_0)|$ for models (11) and (12). The superquantile model (12) shows an improvement of 90/95th quantiles of the distribution of residuals, which appears on histograms as a shift of the upper tail to the left. This comes at the price of a degraded performance on average, which appears on the figure as the shift of the peak of residuals to the right. □



Metric	model (11)	model (12)
Mean	**7.23**	10.62
80^{th} perc.	**9.73**	13.94
90^{th} perc.	17.29	**15.93**
95^{th} perc.	24.00	**17.83**

Fig. 1: Superquantile regression improves over worst-case datapoints. **Left figure**: histograms of residuals $r_i = |y_i - (w_2 x_i^2 + w_1 x_i + w_0)|$ for model (11) (in violet) and model (12) (in orange). **Right table**: x^{th} perc. stands for x-th percentile of final distribution of the residuals r_i.

We finish this section on superquantile learning with an asymptotic result generalizing (9) for the superquantile. We present an elementary self-contained proof: we follow the general approach (see e.g. the monograph [7]) that we combine with the specific expression of the superquantile.

We start with the mathematical framework. The prediction space Z is equipped with a norm $\|\cdot\|$. We consider the uniform pseudometric dist_φ on the parameter set W

$$\mathrm{dist}_\varphi(w, w') = \sup_{x \in X} \|\varphi(w, x) - \varphi(w', x)\|.$$

We assume W is bounded and we further make the following assumption on its size with respect to dist_φ.

Assumption 1 (On the "size" of W)**.** *For any $\varepsilon > 0$, there exists a finite set $T \subset W$ such that for every $w \in W$, there exists a $w' \in T$ with $\mathrm{dist}_\varphi(w, w') \leq \varepsilon$. Such a T is called a ε-cover of W, and the size of the smallest such a T is denoted $N(\varepsilon)$.*

For example for the set of d-dimensional linear functions $\varphi(w, x) = w^\top x$ for $\|w\|_2 \leq 1$. If we take the norm $\|z\| = |z|$ on the real line $Z = \mathbb{R}$, one can prove that $\log N(W, \mathrm{dist}_\varphi, \varepsilon) \leq C\, d \log(1/\varepsilon)$ for some absolute constant C and normalized data (see e.g., [56, Lemma 5.7]). The second standard assumption that we consider is on the loss function $\ell(\cdot, \cdot)$.

Assumption 2 (On the loss)**.** *The loss function ℓ is P-almost surely*

- *bounded, i.e., $0 \leq \ell(y, \varphi(w, x)) \leq B$ for each $w \in W$,*
- *M-Lipschitz in the second argument, i.e., $|\ell(y, z) - \ell(y, z')| \leq M\|z - z'\|$ for every $z, z' \in Z$.*

We have the following result generalizing (9).

Theorem 1. *Let Assumptions 1 and 2 hold. Fix $p \in (0, 1)$ and assume that the minimizers of S^p and S^p_n are attained:*

$$w^\star \in \operatorname*{arg\,min}_{w \in W} S^p(w) \quad \textit{and} \quad w^\star_n \in \operatorname*{arg\,min}_{w \in W} S^p_n(w).$$

Then, we have that $S^p(w^\star_n) \to S^p(w^\star)$ almost surely.

Proof Sketch. We give a sketch here and defer technical details to Appendix.

The key step in the proof is to show the uniform convergence

$$S^p_n(w) \to S^p(w) \text{ almost surely for all } w \in W.$$

Indeed, once we have this, the result immediately follows as

$$\begin{aligned} 0 \leq S^p(w^\star_n) - S^p(w^\star) &= S^p(w^\star_n) - S^p_n(w^\star_n) + S^p_n(w^\star_n) - S^p_n(w^\star) + S^p_n(w^\star) - S^p(w^\star) \\ &\leq 2 \sup_{w \in W} |S^p_n(w) - S^p(w)| \to 0, \end{aligned}$$

where we use $S^p_n(w^\star_n) \leq S^p_n(w^\star)$ in the second inequality.

The proof of the uniform convergence follows the general approach (see e.g. [7]) adapted to variational expression of the superquantile (4). We introduce

$$\bar{S^p}(w, \eta) = \eta + \frac{1}{1-p} \mathbb{E}_{(x,y)\sim P}[\max(\ell(y, \varphi(w, x)) - \eta, 0)]\,,$$

to write

$$S^p(w) = \min_{\eta \in [0,B]} \bar{S^p}(w, \eta),$$

as well as analogous empirical version $\bar{S^p_n}(w)$. The proof now consists in two steps, for a given $\varepsilon > 0$

- to construct a cover T of $W\times[0,B]$ from a cover of W (given by assumption);
- to control the convergence over the points of T, more precisely to control the probability of the event

$$E_n(\varepsilon) = \bigcap_{(w,\eta)\in T} \left\{ \bar{S^p_n}(w, \eta) - \bar{S^p}(w, \eta) \leq \varepsilon/2 \right\}\,.$$

In fact, we show that

$$\sum_{n=1}^{\infty} \mathbb{P}\Big(\exists w \,:\, |S_n^p(w) - S^p(w)| > \varepsilon\Big) \le \sum_{n=1}^{\infty} \mathbb{P}\big(\overline{E}_n(\varepsilon)\big) < \infty,$$

from which we conclude the uniform convergence $S_n^p(w) \to S^p(w)$ for all $w \in W$ using the Borel-Cantelli Lemma. □

3 Recent Applications of the Superquantile in Machine Learning

In this section, we give brief introductions to some of recent applications in machine learning, involving the superquantile.

3.1 Conformity in Distributed Learning on Mobile Devices

The superquantile can be leveraged in distributed learning on mobile devices to model conformity to the population [23]. Each mobile device contains the data generated by a single user, and thus the data distribution across devices is highly heterogeneous.

Concretely, suppose we have m training devices with respective data distribution q_i and losses $L_i(w) = \mathbb{E}_{(x,y)\sim q_i}[\ell(y, \varphi(w,x))]$. *Federated learning* [18] is a distributed learning paradigm which aims to collaboratively learn a common model across all devices without moving data between devices. The empirical risk minimization approach to federated learning consists in assigning a weight $\alpha_i > 0$ to each device to minimize the aggregate loss, which corresponds to an expectation over a mixture p_α of the training distributions q_i:

$$L(w) = \sum_{i=1}^{m} \alpha_i L_i(w) = \mathbb{E}_{(x,y)\sim p_\alpha}[\ell(y, \varphi(w,x))] \quad \text{with} \begin{cases} p_\alpha = \sum_{i=1}^m \alpha_i q_i \,, \\ \sum_{i=1}^n \alpha_i = 1 \,. \end{cases}$$

While minimizing such an objective might offer good performance for test devices which conform to the population of training devices (i.e., distribution q of the test device is close to p_α), one can expect poor predictive performance when q largely departs from p_α. A alternative is to model the heterogeneity of devices by considering data distribution p_π written as a convex combinaison of the training distributions, but with weights π_i different from α_i:

$$p_\pi = \sum_{i=1}^{m} \pi_i q_i, \qquad \text{with } 0 \le \pi_i \le 1 \text{ and } \sum_{i=1}^{m} \pi_i = 1.$$

In this context, [23] proposes to measure how close a test device's distribution p_π is to the training distribution p_α by the so-called conformity level

$$\mathrm{conf}(p_\pi) = \min_{1 \le i \le m} \alpha_i / \pi_i \quad \in (0, 1].$$

We see that the closer the conformity level is to 1, the closer p_π is to p_α, and thus the device and its user tightly conform to the population trend. To learn a robust model w performing well on reasonably non-conforming devices, [23] proposes to find the best w for the set of devices with a conformity of at least a given threshold $c \in (0,1)$; this leads to the optimization problem

$$\min_{w\in\mathbb{R}^d} \max_{p_\pi\in\mathcal{P}} \mathbb{E}_{(x,y)\sim p_\pi}[\ell(y,\varphi(w,x))]\,, \quad \text{with} \quad \mathcal{P}=\{p_\pi : \mathrm{conf}(p_\pi)\geq c\}\,. \tag{13}$$

We observe now that the condition $\mathrm{conf}(p_\pi)\geq c$ can be written $\pi_i/\alpha_i \leq 1/c$ for all i. Thus this constraint coincides, for the level $p=1-c$, with the constraint $q_i \leq \frac{1}{m(1-p)}$ in the dual formulation of the superquantile (3); see more precisely the discrete version of the dual formulation (19). The extensive computational experiments of [23, Sec. 4] show that such superquantile federated learning has, as expected, superior performances for heterogeneous devices. Here we provide a toy example illustrating the interest of the approach.

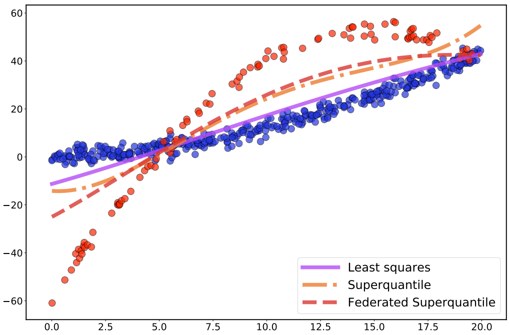


Fig. 2: Comparison of the three regressions for the toy federated learning setting described in Example 3. We want commensurate performances among users, which means graphically a curve at the same distance from the datapoints of the conforming users (in blue) and the non-conforming user (red).

Example 3 (Federated regression) Consider a specific instance of Example 2 in a federated setting. We consider that 80% of the data corresponds to four devices having the same data distribution following (10), while the remaining 20% corresponds to a fifth device having its own distribution. In this example, the solution to federated regression problem with α being the uniform distribution over the five devices coincides with the ordinary least squares model on the whole dataset. Figure 2 shows this bivalent dataset: the blue points correspond to the data of the four first devices, and the red points correspond to the last device. We would like to have a regression that captures worst-cases for both behaviours. We take the conformity level of $c = 1-p = 1/5$, using the knowledge on how the dataset is constructed.

We plot on Figure 2 the regressions given by (11), (12), and the (13). We can make three observations. First the standard model (11) (in purple) tends to follow the trend imposed by the first four devices. Second, the superquantile model (12) (in orange) has better regression on worst-case data, but regardless the group of the data point. Finally the federated superquantile model (13) finds, in contrast, a compromise between the two trends. Thus, federated superquantile regression better captures the (red) data points of the non-conforming user. □

3.2 Fairness-aware Machine Learning

The superquantile naturally appears when considering the notion of fairness in machine learning (see e.g., [16, 19]) as presented below.

Fairness in machine learning is studied with reference to a sensitive attribute, such as race or gender; see e.g., [19]. Suppose that we have a population that can be partitioned unambiguously between G subgroups with respect to this attribute. We denote $L(w) = (L_1(w), \dots, L_G(w))$ the vector of losses on each of the subgroups of the training set. Fairness in such situation would require independence between the sensible attribute and either predicted value or averaged losses per group $L_i(w)$. An *ideal group fairness* of the model w would then imply that $L_1(w) = \cdots = L_n(w)$ [57, Def. 1]; but such a model could be no better than random guessing in the worst case. So [57] considers *approximate group fairness* and introduces the notion of fairness risk measures. As explained in detail in Section 4 and supplementary material of [57], key properties for fairness risk measures includes convexity, positively homogeneity and monotonicity: the superquantile is thus a prominent example of such a measure. Experiments in [57, Sec. 7] show that superquantile indeed allows for a good balance between predictive accuracy and fairness violation. For completeness, we provide here a simple illustration in the context of Example 3.

Example 4 (Fair regression) Let us come back to the toy example of Example 3. We look at it with the perspective of fairness between the predominant group (the four blue users) and the minority group (the fifth "red" user). Table 1 compares (i) the average performance over the predominant group and (ii) the average performance on the minority group. We observe that the difference between these performance is minimal for the user-level superquantile model provided by (13), achieving better approximate group fairness. □

Model	$L_1(w)$ (blue subgr.)	$L_2(w)$ (red subgr.)
least-square (11)	4.59	17.76
superquantile (12)	9.88	13.62
federated superqu. (13)	10.87	11.46

Table 1: Average performances of each model over both subgroups.

3.3 Adversarial Classification

The superquantile also appears in generalized classification tasks when studying robustness to perturbations of data distributions [15].

In binary classification, we have $Y = \{-1, +1\}$ and the prediction function $\varphi(w, x)$ correctly classifies a data point (x, y) if

$$y\, \varphi(w, x) > 0.$$

For an underlying data distribution P, we may want to choose w so as either to minimize the probability $\mathbb{P}_{(x,y)\sim P}\left(y\, \varphi(w, x) \leq 0\right)$ of encountering an error, or to control the distance $d(w, (\bar{x}, \bar{y}))$ to misclassification of a data point $(\bar{x}, \bar{y})$:

$$d(w, (\bar{x}, \bar{y})) = \inf_x \left\{ \|x - \bar{x}\|^2 \,:\, \bar{y}\, \varphi(w, x) \leq 0 \right\} .$$

In this context, robustness against perturbations of the data distribution P can be guaranteed by minimizing the worst-case error probability over a ball (e.g., for Wasserstein distance d_{W}) around P

$$\min_w \sup_{Q:\, d_{\mathrm{W}}(Q,P)\leq\varepsilon} \mathbb{P}_{(x,y)\sim Q}\big(y\, \varphi(w, x) \leq 0\big) . \tag{14}$$

Interestingly, optimal solutions of this problem coincide with those solving:

$$\min_w \; [\mathbb{S}_p]_{(\bar{x},\bar{y})\sim P}\big(- d(w, (\bar{x}, \bar{y}))\big) , \tag{15}$$

for a well-chosen p; see [15, Theorem 2.6]. When the distance function d has a computable closed form, formulation (15) is simpler to handle than (14). We refer to [15] for results in the general case and for related literature.

3.4 Risk-Sensitive Reinforcement Learning

In a framework different from the supervised learning ones considered so far, the superquantile plays a role in risk-sensitive reinforcement learning. Reinforcement learning methods attempt to find decision rules to minimize a cumulative cost [52] in a sequential decision-making setting.

Concretely, a learning agent acts in a Markov decision process using a policy π which maps a state to a distribution over an action space. The agent's aim is to minimize the total cost $c(\tau) = \sum_{i=1}^n c(s_i)$ of a trajectory $\tau = (s_1, \ldots, s_n)$ of states taken by the agent while following the policy π. Letting $\Gamma(\pi)$ denote the induced distribution over trajectories of length n under policy π, standard reinforcement learning methods minimize the expected cumulative cost as

$$\min_w \mathbb{E}_{\tau\sim\Gamma(\pi_w)}[c(\tau)] ,$$

where $w \in \mathbb{R}^d$ parameterizes the policy π_w. The so-called policy gradient methods aim to solve this by first-order optimization methods where the gradient of the objective is estimated by Monte Carlo simulations [53].

However, in safety-critical applications, we are interested in accounting for unlikely events with high cost [17]. In particular, [30] considers sensitivity to risky high-cost trajectories by minimizing the superquantile counterpart

$$\min_w \, [\mathbb{S}_p]_{\tau \sim \Gamma(\pi_w)} [c(\tau)] \, . \tag{16}$$

This risk-sensitive reinforcement learning setting thus leads to similar superquantile problems than in the supervised learning setting. We refer to [54] on how to adapt policy gradient methods to estimate the gradient of the objective with respect to the parameters.

4 Efficient (Sub)differential Calculus

The applications sketched in the previous section reveal optimization problems with objective functions[2] written as the composition of a superquantile and a general loss function

$$f(w) = \mathbb{S}_p(L(w)). \tag{17}$$

For example, (12) involves $L\colon \mathbb{R}^d \to \mathbb{R}^n$ defined component-wise for each data point by $L_i(w) = \ell(y_i, \varphi(w, x_i))$; similar expressions follow from (13), (15) and (16). We notice first that L is usually non-convex (e.g., with φ as (5)) but smooth (e.g., with ℓ as (6)).

In this section, we provide easy-to-implement expressions of subgradients of superquantile-based functions (17), in Section 4.2, and of gradients of smoothed approximations of them in Section 4.3. Finally, in Section 4.4, we compare the proposed smoothing with others considered in the literature (e.g., [6, 26]). Computing the (sub)gradients would be the first step toward using first-order optimization algorithms for solving superquantile problems. Though simple, this idea of using first-order methods is not widely used for such problems; among the few exceptions, we mention the PhD thesis [29] using subgradient algorithms (in a special case) and our conference paper [22] presenting a toolbox for using first-order methods in superquantile learning. The developments of this section detail and extend those of [22, Sec. 3].

4.1 Computing the Superquantile

For the practical developments of this section, we consider a data-driven setting where the random variable U takes equiprobable values $u_1, \dots, u_n$[3]. In this

[2] In coherence with the previous section and to comply with common notation in machine learning, we stick to the notation w for the variable of the functions.

[3] In the sequel, we make a slight abuse of notation by not distinguishing between the random variable U and the vector of its equiprobable realizations $u = (u_1, \dots, u_n)$. Thus

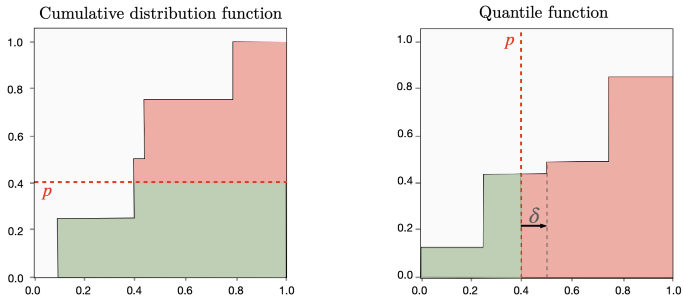


Fig. 3: Illustration of the integral expression of the superquantile. Cumulative distribution function (on the left) and quantile function (on the right) are inverse one of the other. $\mathbb{S}_p(U)$ is obtained by averaging the quantiles greater than p (red section on right graph).

setting, the three representations of superquantile recalled in Section 1.2 takes explicit forms that have special interest from a computational perspective.

- *Integral representation.* By splitting the integral, (2) can be written as

$$\mathbb{S}_p(U) = \frac{1}{n(1-p)} \sum_{i \in I_>} u_i + \frac{\delta}{1-p} Q_p(U) \quad \text{with } I_> = \{i : u_i > Q_p(U)\}. \quad (18)$$

  This expression involves the distance from p to the next discontinuity point of the quantile function $p' \mapsto Q_{p'}(U)$ (see Figure 3):

$$\delta = F_U(Q_p(U)) - p = \frac{1}{n}(n - |I_>|) - p.$$

  Thus (18) gives an efficient way to compute superquantiles from the following three step procedure: (a) compute the p-quantile with the specialized algorithm (called `quickfind`) of complexity $\mathrm{O}(n)$; (b) select all values greater or equal than the quantile; (c) average values along (18).

- *Dual representation.* The expression (3) simplifies to

$$\mathbb{S}_p(U) = \max_{q \in \Delta_p} \; q^\top u \quad \text{with } \Delta_p = \left\{ q \in \mathbb{R}_+^n : \sum_{i=1}^n q_i = 1, q_i \leq \frac{1}{n(1-p)} \right\}. \quad (19)$$

  In words, the superquantile is the support function of the intersection of the simplex with a box (see Figure 4). This problem also corresponds to a classical optimization problem, called fractional knapsack problem, which is solved, after sorting the u_i's, by a simple greedy strategy of the associated q_i's [9]. For our purposes, this expression of superquantile, as a direct max, is useful when applying dual smoothing techniques; see Section 4.3.

we consider the superquantile function $\mathbb{S}_p : \mathbb{R}^n \to \mathbb{R}$ as a function of $u \in \mathbb{R}^n$, and we study the differentiability properties of compositions with $\mathbb{S}_p$.

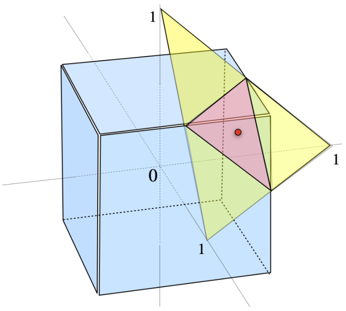


Fig. 4: Illustration of the dual expression of the superquantile. $\mathbb{S}_p$ is the support function of the red polytope. The red point represents the uniform distribution.

- *Variational representation.* The expression (4) writes

$$\mathbb{S}_p(U) = \inf_{\eta \in \mathbb{R}} \left\{ \eta + \frac{1}{n(1-p)} \sum_{i=1}^{n} \max(u_i - \eta, 0) \right\}. \tag{20}$$

  This expression is often used in solving approaches for superquantile optimization; see e.g., the progressive hedging for risk-averse multistage programming of [39]. Here, this expression will provide a nice interpretation of the infimal convolution smoothing (Corollary 6).

4.2 Subdifferentials

In this section, we provide explicit and implementable expressions of the subdifferential of superquantile-based functions. Expressions of (convex) subdifferential of superquantile are well-known in general settings; see e.g., [46] for a thorough study. Here we study non-convex subdifferentials and derive concrete expressions in the data-driven context; we give direct proofs as applications of basic definitions and properties of nonsmooth analysis.

We start by recalling the standard notions of subgradients for nonsmooth functions (in finite dimension), following the terminology of [45]. For a function $\psi\colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$, the regular (or Fréchet) subdifferential of ψ at $\bar{w}$ (such that $\psi(\bar{w}) < +\infty$) is defined by

$$\partial^R \psi(\bar{w}) = \left\{ s \in \mathbb{R}^d : \ \psi(w) \geq \psi(\bar{w}) + s^\top (w - \bar{w}) + \mathrm{o}(\|w - \bar{w}\|) \right\}.$$

The regular subdifferential thus corresponds to the set of gradients of smooth functions that are below ψ and coincide with it at $\bar{w}$. The limiting subdifferential is the set of all limits produced by regular subgradients

$$\partial^L \psi(\bar{w}) = \limsup_{w \to \bar{w}, \psi(w) \to \psi(\bar{w})} \partial^R \psi(w).$$

These notions generalize (sub)gradients of both smooth functions and convex functions: for these functions indeed, the two subdifferentials coincide, and they reduce to $\{\nabla\psi(\bar{w})\}$ when ψ is smooth and to the standard subdifferential from convex analysis when ψ is convex.

For the function (17), which is the composition of a convex function and a continuously differentiable function, we get from basic chain rules that the two subdifferentials coincide; we simply denote it by $\partial f(w)$. Moreover the dual representation (19) expressing $\mathbb{S}_p$ as a support function allows to obtain readily an expression of the subdifferential of $\partial\mathbb{S}_p$ and, as a result, of the one of f. We formalize all this in the following proposition.

Proposition 2 (Explicit subdifferential of superquantile-based functions)**.** *Consider the superquantile-based function* (17) *with L continuously differentiable. We have*

$$\partial f(\bar{w}) = \Big(\partial^L f(\bar{w}) = \partial^R f(\bar{w}) = \Big)\ \nabla L(\bar{w})^* \partial\,\mathbb{S}_p(L(\bar{w})) \tag{21}$$

where $\nabla L(\bar{w})^*$ is the adjoint of the Jacobian of L at $\bar{w}$ and $\partial\,\mathbb{S}_p(L(\bar{w}))$ the (convex) subdifferential of $\mathbb{S}_p$ taken at $L(\bar{w})$. Moreover, for $w \in \mathbb{R}^d$, compute $L(w) \in \mathbb{R}^n$ and $Q_p(L(w)) \in \mathbb{R}$. Consider $I_>$ the set of indices such that $L_i(w) > Q_p(L(w))$ and $I_=$ the set of indices such that $L_i(w) = Q_p(L(w))$. Then the subdifferential of f at w can be written with the gradients $\nabla L_i(w)$ for $i \in I_> \cup I_=$, as follows

$$\partial f(w) \;=\; \frac{1}{n(1-p)} \sum_{i \in I_>} \nabla L_i(w) \;+\; \frac{\delta}{1-p}\,\mathrm{conv}\left\{\nabla L_i(w) : i \in I_=\right\}.$$

Proof. We apply the chain rule of [45, 10.6] to the composition $\mathbb{S}_p \circ L$: we have that $\mathbb{S}_p$ is convex with full domain, which implies that the two subdifferentials[4] of f coincide (i.e., f is regular in the terminology of [45]) and we have (21).

Since $\mathbb{S}_p$ is the support function of the set Δ_p, standard subdifferential calculus [14, Cor. 4.4.4] gives that $\partial\mathbb{S}_p(L(w))$ is the set of optimal solutions of (19) with $u = L(w)$. Knowing $I_>$ and $I_=$, the so-called fractional knapsack problem (19) can be solved by the simple greedy strategy [9] of taking the largest q_i for $i \in I_>$ and completing to 1 with the q_i for $i \in I_=$. Thus

$$q \text{ solution of } (19) \iff \begin{cases} q_i = \frac{1}{n(1-p)} & \text{if } i \in I_> \\ 0 \le q_i \le \frac{1}{n(1-p)} & \text{if } i \in I_= \ \text{ s.t. } \sum_{i \in I_=} q_i = \frac{\delta}{1-p} \\ q_i = 0 & \text{otherwise.} \end{cases}$$

[4] Remark on the Clarke subdifferential: As another by-product of the chain rule [45, 10.6], the set of horizon subgradients of f is reduced to 0 since so is the one of $\mathbb{S}_p$ (convex and defined on $\mathbb{R}^n$). As a consequence, the Clarke subdifferential is the convex hull of the limiting subdifferential [45, 8.49]. Thus we have, in our case, that the three subdifferentials (regular, limiting and Clarke) coincide.

By (21), this gives:

$$\partial f(w) = \frac{1}{n(1-p)} \sum_{i \in I_>} \nabla L_i(w) + \left\{ \sum_{i \in I_=} q_i \nabla L_i(w), \text{ s.t. } \begin{cases} 0 \le q_i \ \forall i \in I_= \\ \sum_{i \in I_=} q_i = \frac{\delta}{1-p} \end{cases} \right\}.$$

Finally, introducing weights $\alpha_i = \frac{q_i(1-p)}{\delta}$ for $i \in I_=$, the right-hand term can be written as the convex hull of $\nabla L_i(w)$ for $i \in I_=$, which gives the expression. □

We observe that the expression of $\partial f(w)$ does not involve the gradients of all the L_i's, but only of those associated to the largest values. We also see that f is differentiable at w if and only if $I_=$ is reduced to a singleton. The objective function is not differentiable in general, which poses a problem for a direct application of machine-learning gradient-based algorithms.

4.3 Smoothing by Infimal Convolution

In this section, we study a smoothing of nonsmooth superquantile-based functions (17). We propose to use the infimal convolution smoothing of [31]; the comparison to other smoothing approaches is postponed to the next section.

We follow the guidelines of [2] : we smooth only the superquantile $\mathbb{S}_p$ rather than the whole function f. Thus we consider

$$f_\nu(w) = \mathbb{S}_p^\nu(L(w)) \quad \text{for } \mathbb{S}_p^\nu \text{ a smooth approximation of } \mathbb{S}_p. \tag{22}$$

Regularizing the dual representation (19) of superquantile, we consider the function, parameterized by the smoothing parameter ν,

$$\mathbb{S}_p^\nu(u) = \max_{q \in \Delta_p} \left\{ q^\top u - \nu D(q) \right\} , \tag{23}$$

for a given strongly convex function D. The following proposition establishes that the resulting function f_ν as (22) is a smooth approximation of f, as a direct application of e.g., [2, Theorem 4.1, Lemma 4.2], or [31, Theorem 1].

Proposition 3 (Smoothed approximation)**.** *In the above setting, the function f_ν provides a global approximation of f, i.e.*

$$f_\nu(w) \le f(w) \le f_\nu(w) + \frac{\nu}{2} \qquad \textit{for all } w \in \mathbb{R}^d.$$

Moreover $\mathbb{S}_p^\nu$ is differentiable, with $\nabla \mathbb{S}_p^\nu(u)$ being the argmax of (23), unique by strong convexity of D. When L is differentiable, f_ν is differentiable as well, with

$$\nabla f_\nu(w) = \nabla L(w)^* \nabla \mathbb{S}_p^\nu(L(w)). \tag{24}$$

In our quest for simple and implementable expressions, we study in the rest of this section the case of separable strongly functions of the form:

$$D(q) = \sum_{i=1}^{n} d(q_i) \qquad \text{given a strongly convex function } d\colon [0,1] \to \mathbb{R}. \tag{25}$$

We provide in Corollary 5 a general scheme to compute the gradient with explicit expressions in Examples 5 and 6 for special choices of d. Finally we discuss the role of the smoothing parameter ν on a numerical illustration.

We start with a lemma gathering the nice duality properties of (23). A one-dimensional convex function plays a special role: it is the convex conjugate of the sum of νd and the indicator of the segment $[0, 1/n(1-p)]$

$$g_\nu(s) = \left(\nu d + i_{[0,\frac{1}{n(1-p)}]}\right)^*(s) = \max_{0 \le t \le \frac{1}{n(1-p)}} \{s\,t - \nu\, d(t)\} \,. \tag{26}$$

Since d is strongly convex, standard (one-dimensional) convex analysis gives (see e.g., [14, Prop.I.6.2.2]) that g_ν is continuously differentiable with derivative $g'_\nu(s)$ being the (unique) t achieving the above max. Simple calculus yields

$$g'_\nu(s) = \begin{cases} 0 & \text{if } s \le \nu\ d'_+(0) \\ \frac{1}{n(1-p)} & \text{if } s \ge \nu\ d'_-(1/(n(1-p))) \\ (d^*)'\left(\frac{s}{\nu}\right) & \text{otherwise.} \end{cases} \tag{27}$$

where $d'_+(0) \in [-\infty, +\infty)$ and $d'_-(1/(n(1-p))) \in [-\infty, +\infty)$ are respectively the right-derivative of d at 0 and the left-derivative of d at $1/(n(1-p))$. Note finally that ${g_\nu}'$ is a non-decreasing function.

Lemma 4 (Duality)**.** *The dual problem of* (23) *can be expressed as the (smooth convex) one-dimensional problem:*

$$\min_\eta \quad \theta(\eta) = \eta + \sum_{i=1}^{n} g_\nu(u_i - \eta). \tag{28}$$

Moreover, there is no duality gap between (23) *and* (28)*. There exists a primal-dual solution* $(q^\star_\nu, \eta^\star)$ *and the unique primal solution can be written* $q^\star_\nu = (g'_\nu(u_i - \eta^\star))_{i=1,\ldots,n}$ *with the help of* (27)*.*

Proof. This lemma could be proved by applying a sequence of results from abstract Lagrangian duality [14, Chap. XII]. Instead, we provide a simple proof from the direct calculus developed so far. Consider the dualization of the constraint $\sum_{i=1}^n q_i - 1 = 0$ in Δ_p. For a primal variable $q \in B_p = \left[0, \frac{1}{n(1-p)}\right]^n$ and a dual variable $\eta \in \mathbb{R}$, we write the Lagrangian

$$L(q,\eta) = \sum_{i=1}^{n} q_i u_i - \nu d_i(q_i) - \eta\Big(\sum_{i=1}^{n} q_i - 1\Big) = \eta + \sum_{i=1}^{n} q_i(u_i - \eta) - \nu d_i(q_i)\,,$$

and the associated dual function

$$\theta(\eta) = \max_{q \in B_p} L(q, \eta) = \eta + \sum_{i=1}^{n} \max_{0 \leq q_i \leq \frac{1}{n(1-p)}} \{q_i(u_i - \eta) - \nu\, d_i(q_i)\}\ ,$$

which gives the expression of the dual function (28) from (26). Note for later that we have, by construction, the so-called weak duality inequality

$$\theta(\eta) \geq L(q, \eta) = \sum_{i=1}^{n} q_i u_i - \nu d_i(q_i) \quad \text{for all } \eta \text{ and all feasible } q \in \Delta_p. \tag{29}$$

Now recall that g_ν in (26) is differentiable and so is the dual function with

$$\theta'(\eta) = 1 - \sum_{i=1}^{n} g'_\nu(u_i - \eta)\,. \tag{30}$$

The above expression also shows that

$$\lim_{\eta \to +\infty} \theta'(\eta) = 1 \qquad \text{and} \qquad \lim_{\eta \to -\infty} \theta'(\eta) = 1 - \sum_{i=1}^{n} \frac{1}{n(1-p)} = \frac{-p}{1-p}.$$

By continuity of g'_ν and θ', this implies that there exists $\eta^\star$ such that $\theta'(\eta^\star) = 0$, i.e., there exists a dual solution $\eta^\star$. On the primal side, the compactness of B_p and strong convexity of d gives existence and uniqueness of the primal solution, denoted $q^\star_\nu$. Observe now that (30) means that the vector $(g'_\nu(u_i - \eta^\star))_{i=1,\ldots,n}$, which lies in B_p by construction, is in fact primal feasible. From (29) and uniqueness of the primal solution, this implies that $q^\star_\nu = (g'_\nu(u_i - \eta^\star))_{i=1,\ldots,n}$ and that there is no duality gap. □

From Lemma 4, we get an almost explicit expressions of values and gradients of the smooth approximation f_ν.

Corollary 5 (Oracle for smooth approximation)**.** *Consider f_ν defined by* (22) *with L differentiable. With $\eta^\star$ an optimal solution of* (28) *with $u_i = L_i(w)$,*

$$f_\nu(w) = \eta^\star + \sum_{i=1}^{n} g_\nu(L_i(w) - \eta^\star)\,,$$

$$\nabla f_\nu(w) = \sum_{i=1}^{n} g'_\nu(L_i(w) - \eta^\star)\, \nabla L_i(w)$$

where g_ν and g'_ν are given by (26) *and* (27)*.*

Proof. The no-gap result of Lemma 4 gives that $\mathbb{S}^\nu_p(u)$ is equal to the optimal value of (28). This gives directly the above expression of $f_\nu(w) = \mathbb{S}^\nu_p(L(w))$ with $\eta^\star$ an optimal solution of (28) with $u_i = L_i(w)$. Regarding the expression of the gradient, Proposition 3 states that $\nabla \mathbb{S}^\nu_p(u)$ is the optimal solution of (23), and Lemma 4 expresses it as $(g'_\nu(u_i - \eta^\star))_{i=1,\ldots,n}$. We then get the expression of $\nabla f_\nu(w)$ from (24). □

Thus the computation of the first-order oracle of f_ν boils down to solving the one-dimensional convex problem (28) with $u_i = L_i(w)$. This easy task can be done in general by bisection or higher-order schemes. Here Lemma 4 allows us to make an additional simplification with an initial interval tightening. We can indeed shrink the segment where to find $\eta^\star$ to two consecutive points in

$$N = \left\{ u_i - \nu\, d'_+(0), u_i - \nu\, d'_-\Big(\frac{1}{(n(1-p)}\Big) \;\; i = 1, \ldots, n \right\}$$

which is a set of special points regarding the structure of the dual function (recall (27) and (28)). Denoting $\underline{\eta}$ and $\bar{\eta}$, defined respectively as the largest point in N such that $\theta'(\underline{\eta}) \leq 0$ and the smallest point in N such that $\theta'(\bar{\eta}) \geq 0$, we get $\eta^\star$ by testing three cases:

- if $\theta'(\underline{\eta}) = 0$, take $\eta^\star = \underline{\eta}$; if $\theta'(\bar{\eta}) = 0$, take $\eta^\star = \bar{\eta}$;
- otherwise, compute $\eta^\star$ in the small interval $[\underline{\eta}, \bar{\eta}]$.

The initial interval tightening thus boils down to having sorted points in N, which is obtained directly from sorting the given data.

Finally we emphasize that we can sometimes go one step further ahead and obtain explicit expressions of $\eta^\star$ and thus, readily implementable expressions of $\nabla f_\nu(w)$. In the next two examples, we illustrate this for two cases of interest, when we smooth the superquantile by a divergence to the uniform probability (which is at the center of Δ_p; recall Figure 4). In particular the smoothing detailed in the forthcoming Example 5 was used in the numerical illustrations of Examples 2, 3, and 4 (where the resulting smoothed superquantile optimization problems were solved by L-BFGS).

Example 5 (Euclidean smoothing) We suggest to smooth the superquantile with the Euclidean distance to the uniform distribution

$$D(q) = \frac{1}{2}\|q - \bar{q}\|^2 \quad \text{with} \quad \bar{q} = \left(\frac{1}{n}, \ldots, \frac{1}{n}\right),$$

which consists in taking in (25)

$$d(t) = \frac{1}{2}\left(t - \frac{1}{n}\right)^2.$$

In this case, elementary calculus gives

$$d'_-(0) = -\frac{1}{n}, \quad d'_+\left(\frac{1}{n(1-p)}\right) = \frac{p}{n(1-p)}, \quad \text{and} \quad (d^*)'\left(\frac{t}{\nu}\right) = \frac{t}{\nu} + \frac{1}{n}$$

so that we get from (27) the following expression

$$g'_\nu(u_i - \eta) = \begin{cases} 0 & \text{if } \eta \geq u_i + \frac{\nu}{n} \\ \frac{1}{n(1-p)} & \text{if } \eta \leq u_i - \frac{\nu}{n}\frac{p}{1-p} \\ \frac{u_i - \eta}{\nu} + \frac{1}{n} & \text{otherwise.} \end{cases}$$

We also have that θ' is piecewise linear in this case and that

$$N = \left\{ x_i + \frac{\nu}{n}, x_i - \frac{\nu}{n}\frac{p}{1-p} \quad i = 1, \ldots, n \right\}.$$

Therefore from $\underline{\eta}$ and $\bar{\eta}$ in N, finding $\eta^\star$ in the interval $[\underline{\eta}, \bar{\eta}]$ simply reduces to interpolating linearly as

$$\eta^\star = \underline{\eta} - \frac{\theta'(\underline{\eta})(\bar{\eta} - \underline{\eta})}{\theta'(\bar{\eta}) - \theta'(\underline{\eta})}.$$

We can apply Corollary 5 to get an efficiently implemented expression of the gradient. Note that the obtained expression of $\nabla f_\nu(w)$ involves only the gradients $\nabla L_i(w)$ for largest values of $L_i(w)$ (comparable to the expression of $\partial L(w)$ in Proposition 2). □

Example 6 (KL smoothing) We use here the Kullback-Lieber divergence to the uniform probability

$$d(q) = \sum_{i=1}^{n} q_i \log(q_i/\bar{q}_i) \quad \text{with} \quad \bar{q} = \left(\frac{1}{n}, \ldots, \frac{1}{n}\right).$$

which consists in taking $d(t) = t\log(t)$ in (25). Elementary calculus then gives

$$d'_+(0) = -\infty, \quad d'_-\left(\frac{1}{n(1-p)}\right) = 1 - \log(n(1-p)), \text{ and } (d^*)'\left(\frac{t}{\nu}\right) = \exp\left(\frac{t}{\nu} - 1\right)$$

which in turn yields

$$g'_\nu(u_i - \eta) = \begin{cases} \frac{1}{n(1-p)} & \text{if } \eta \leq u_i + \nu\left(\log(n(1-p)) - 1\right) \\ \exp\left(\frac{u_i - \eta}{\nu} - 1\right) & \text{otherwise} \end{cases}$$
$$N = \left\{u_i + \nu\left(\log(n(1-p)) - 1\right) \quad i = 1, \ldots, n\right\}.$$

On the interval $[\underline{\eta}, \bar{\eta}]$, we have that

$$\theta'(\eta) = 1 - \sum_{i \in I} \frac{1}{n(1-p)} - \sum_{i \notin I} \exp\left(\frac{u_i - \eta}{\nu} - 1\right)$$

with $I = \{i, u_i + \nu\left(\log(n(1-p)) - 1\right) \leq \underline{\eta}\}$ the set of indices of points in N smaller than $\underline{\eta}$. This yields

$$\eta^\star = \nu \log\left(\frac{\sum_{i \notin I} \exp(u_i/\nu - 1)}{1 - |I|/\big(n(1-p)\big)}\right).$$

We can then apply Corollary 5 to get the smoothed gradient. □

We conclude this section on the infimal-smoothing of the superquantile with a short discussion on the impact of the smoothing parameter ν.

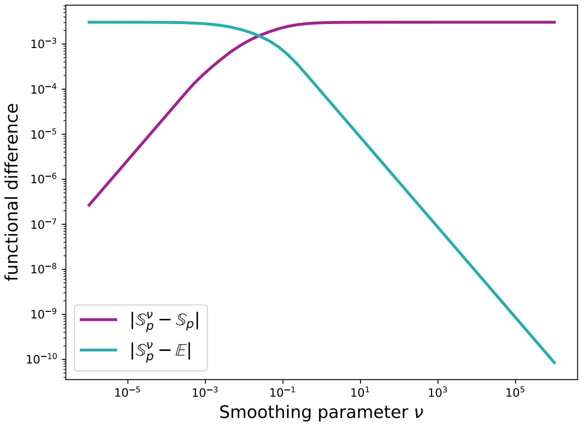


Fig. 5: Impact of the smoothing parameter ν solving a superquantile logistic regression on a classical dataset (Australian Credit dataset).

Remark 1 (Impact of the smoothing parameter) We briefly illustrate here the impact of the smoothing parameter ν: we fix a vector $\bar{w}$ and we observe the values of smoothed approximations of a superquantile-based function for different values of ν. More precisely, we consider the logistic regression problem used in Appendix B; we use the quadratic smoothing of Example 5 with $\nu = 0.1$; and we solve the problem by L-BFGS to get the reference point $\bar{w}$. Then we compute, at this point, the values of:

- the underlying superquantile-based objective (12) which corresponds to the case $\nu = 0$;
- the smoothed approximations (which corresponds to (12) with $\mathbb{S}_p^\nu$ replacing $\mathbb{S}_p$) for a sequence of ν evenly spread on a log scale;
- the usual empirical risk minimization objective (8), which corresponds to the case $\nu = +\infty$. Indeed, in this regime $\nu \to +\infty$, the impact of the quadratic penalization term $(q - \bar{q})$ increases so that the solution of (23) eventually becomes the uniform distribution $\bar{q}$, in which case $\mathbb{S}_p^\nu$ coincides with the expectation.

We observe on Figure 5 what is expected: for small values of ν, the difference between the superquantile-based objective and its smooth approximations vanishes; for large values of ν, the smoothed superquantile loss tends to the average loss and does not approximate the nonsmooth superquantile loss well.

A key benefit of smoothing the superquantile is to leverage efficient smooth optimization algorithms, such as L-BFGS, for superquantile learning. When ν is too small, the problem is almost non-smooth, which leads to numerical issues with convergence (on this instance, L-BGFS fails to converge when ν too small or when used with the nonsmooth oracle of Proposition 2 due to a line search failure). When ν is too large, the smoothed superquantile gets close to the expectation and the interest of using a superquantile approach disappears. This illustrates the interest of having a moderate ν for superquantile learning, where the smoothed objective is an reasonable approximation of the nonsmooth superquantile, while still being smooth enough to leverage fast optimization algorithms. □

4.4 Comparison to Other Smoothing Schemes

We compare the proposed infimal convolution smoothing of the superquantile (23) to other possible smoothing schemes. Classical smoothing techniques are based either on convolution or infimal convolution. For superquantile, one could either smooth the dual representation (19) or the variational representation (20). Together, this yields four natural ways to smooth the superquantile.

We first formalize the equivalence between the two infimal convolution smoothings: indeed, smoothing the dual representation considered in the preceding section corresponds to a smoothing of $\max\{\cdot, 0\}$ in the variational formulation.

Corollary 6 (Equivalence of smoothings with infimal convolution)**.** *With the notation of Section 4.3, the infimal convolution smoothing of* $\mathbb{S}_p$ *with a separable strongly convex function* (25) *is equivalent to the infimal convolution smoothing of the positive part* $\max\{\cdot, 0\}$ *as*

$$m_\nu(\eta) = \max_{0 \le t \le 1} \left\{ \eta\, t - \nu\, \tilde{d}(t) \right\} \qquad \textit{with} \;\; \tilde{d}(t) = n(1-p) d \left(\frac{t}{n(1-p)} \right). \tag{31}$$

More precisely, we have the following equality (to be compared with (20)*)*

$$\mathbb{S}_p^\nu(u) = \min_{\eta} \;\; \left\{ \eta + \frac{1}{n(1-p)} \sum_{i=1}^{n} m_\nu(u_i - \eta) \right\}.$$

Proof. A direct change of variable in (26) gives $g_\nu(u_i - \eta) = \frac{1}{n(1-p)} m_\nu(u_i - \eta)$. The proof is direct from the expression of the dual problem (28) and the no-gap result stated in Lemma 4. □

Next, we show an equivalence between the smoothing by infimal convolution (31), and by convolution, as considered in [6, 26]. Given a continuous probability density $\rho\colon \mathbb{R} \to \mathbb{R}_+$, (such that $\int_{-\infty}^{\infty} |s| \rho(s) \mathrm{d}s$ is finite) the smoothing by convolution of the function $\max\{\cdot, 0\}$ with smoothing parameter $\nu > 0$ is defined by[5]

$$\bar{m}_\nu(\eta) = \frac{1}{\nu} \int_{-\infty}^{\infty} \max\{\eta - s, 0\} \rho\left(\tfrac{s}{\nu}\right) \mathrm{d}s = \frac{1}{\nu} \int_{-\infty}^{\eta} (\eta - s) \rho\left(\tfrac{s}{\nu}\right) \mathrm{d}s\,. \tag{32}$$

[5] Applied to $\max\{x, \cdot\}$, the general smoothing by convolution as defined in (32) coincides with the double integral representation used in [6, 26]. Indeed, integrating (33) yields

$$\bar{m}_\nu(\eta) = \tfrac{1}{\nu} \int_{-\infty}^{\eta} \int_{-\infty}^{\eta'} \rho\left(\tfrac{s}{\nu}\right) \mathrm{d}s\, \mathrm{d}\eta'\,.$$

The function $\bar{m}_\nu$ is convex and smooth, with derivative

$$\bar{m}'_\nu(\eta) = \frac{1}{\nu}\int_{-\infty}^{\eta} \rho\left(\tfrac{s}{\nu}\right) \mathrm{d}s\,. \tag{33}$$

The next proposition, relating this smoothing to the previous one, involves $Q_t(\rho)$ the quantile function of a random variable with density ρ.

Proposition 7 (Equivalence of convolution/inf-convolution smoothings)**.** *With the above notation, the convolution smoothing $\bar{m}_\nu$ of (32) for $\nu = 1$ can be written as the infimal-convolution smoothing (to be compared with (31))*

$$\bar{m}_1(\eta) = \max_{0\leq t\leq 1} \left\{\eta\, t - \bar{d}(t)\right\} \quad \textit{where} \quad \bar{d}(t) = tQ_t(\rho) - \bar{m}_1(Q_t(\rho)). \tag{34}$$

Conversely, the infimal convolution smoothing m_ν of (31) for $\nu = 1$ can be written as the convolution smoothing (to be compared with (32))

$$m_1(\eta) = \lim_{s\to-\infty} m_1(s) + \int_{-\infty}^{\eta} (\eta - s)\tilde{\rho}(s)\mathrm{d}s \quad \textit{where} \quad \tilde{\rho}(s) = m''_1(s) \;\; \textit{a.e.} \tag{35}$$

Proof. For the first part, we consider the convex conjugate of $\bar{m}_1$

$$\bar{m}^*_1(t) = \sup_{\eta\in\mathbb{R}} \left\{\eta\, t - \bar{m}_1(\eta)\right\}\,.$$

If $t \notin [0, 1]$, the supremum is $+\infty$ since $|\bar{m}_1(\eta) - \max\{\eta, 0\}|$ is bounded by an absolute constant. For $t \in [0, 1]$, the concave function $\eta \mapsto \eta t - \bar{m}_1(\eta)$ is maximized at $\eta^\star$ if and only if it satisfies the first-order optimality condition

$$t = \bar{m}'_1(\eta^\star) = \int_{-\infty}^{\eta^\star} \rho(s)\mathrm{d}s.$$

Since the latter is the cumulative distribution function, $\eta^\star = Q_t(\rho)$ is the corresponding quantile function (well-defined since ρ is continuous). This yields

$$\bar{m}^*_1 = \bar{d} + i_{[0,1]}, \tag{36}$$

which in turn gives (34). Finally to establish the strong convexity of $\bar{d}$, we use again (36) together with the smoothness of $\bar{m}_1$. Thus $\bar{m}_1$ corresponds to the infimal-convolution smoothing with $\bar{d}$.

For the second part, we start by noting that since m'_1 is Lipschitz, m''_1 exists almost everywhere, and $\tilde{\rho}$ is well-defined. Since m_1 is convex, it also holds that $m''_1(s) \geq 0$, and then that we have the normalization

$$\int_{-\infty}^{\infty} \tilde{\rho}(s)\mathrm{d}s = \int_{-\infty}^{\infty} \tilde{m}''_1(s)\mathrm{d}s = \lim_{\eta\to\infty} m'_1(\eta) - \lim_{\eta\to-\infty} m'_1(\eta) = 1 - 0 = 1\,,$$

where we use $m'(\eta)$ is the (unique) optimal solution of (31). Then the proof follows from the next two claims.

Claim 1: m_1 admits a limit at $-\infty$. Convexity of m_1 gives that m_1' is non-decreasing. Since $\lim_{s\to-\infty} m_1'(s) = 0$, we get that m_1' is non-negative. Thus, m_1 is non-decreasing and, since it is bounded from below, this implies that m_1 admits a limit at $-\infty$ (that we denote $m_1(-\infty)$).

Claim 2: $\lim_{s\to-\infty} s\, m_1'(s) = 0$. For a given s, we write:

$$s\; m_1'(2s) \le \int_{2s}^{s} m_1'(t)\mathrm{d}t = m_1(s) - m_1(2s),$$

where the inequality comes from the fact that m_1' is non-decreasing. Using that m_1 admits a limit at $-\infty$ (Claim 1), we then get Claim 2.

Finally, we can conclude the proof with integration by parts:

$$\begin{aligned} m_1(\eta) &= m_1(-\infty) + \int_{-\infty}^{\eta} m_1'(s)\mathrm{d}s \\ &= m_1(-\infty) + [(s-\eta)m_1'(s)]_{-\infty}^{\eta} + \int_{-\infty}^{\eta} (\eta - s)\tilde{\rho}(s)\mathrm{d}s \\ &= m_1(-\infty) + \int_{-\infty}^{\eta} (\eta - s)\tilde{\rho}(s)\mathrm{d}s. \end{aligned}$$

This establishes (35) and ends the proof. □

Finally, we mention the smoothing of the dual representation (19) using convolution, which would write:

$$\bar{S}_p^{\nu}(u) = \frac{1}{\nu}\int_{\mathbb{R}^n} S_p(u-z)\rho\big(\tfrac{z}{\nu}\big)\,\mathrm{d}z = \mathbb{E}_{Z\sim\rho}[S_p(u-\nu Z)]\,,$$

for the density $\rho\colon \mathbb{R}^n \to \mathbb{R}$ and the parameter $\nu > 0$. We do not consider this smoothing approach because it suffers from two drawbacks in view of practical implementation. First, it usually cannot be computed in closed form, unlike the other smoothing approaches considered here. Second, the Lipschitz constant of the gradient (appearing in condition numbers, constant scalings, and rates of convergence of first-order methods [32]) scales badly: as $O(\sqrt{n}/\nu)$ for the Lipschitz constant of $\nabla\bar{S}_p^{\nu}$ [33, Lemma 2], as opposed to the dimension-independent $O(1/\nu)$ for the one of ∇S_p^{ν} [31, Theorem 1].

5 Conclusion

In this paper, we have developed two different aspects of the superquantile, a famous risk measure studied and popularized by R. T. Rockafellar and his co-authors. First, we have reviewed recent applications of superquantiles in machine learning, keeping our discussion at a high-level, omitting details, and just providing basic illustrations and pointers to recent research. Second, we have provided explicit expressions of (sub)gradients of (smoothed) superquantiles; here, in contrast, we go down to the details of computation in order to get efficient first-order oracles for superquantile-based functions. In particular, we have proved that smoothed oracles have essentially the same computational complexity as for the corresponding superquantile functions (Corollary 5 and following discussions).

These fast oracles are implemented in the toolbox[6] `spqr` build on top of the popular Python machine learning library `scikit-learn` [36]. This toolbox provides an interface for using standard first-order algorithms; we refer to our numerical experiments of [22] and [23] (see also Appendix B). From this experimental experience, we advocate the use of quasi-Newton methods (and in particular L-BGFS; see e.g., [34]) that gives good performances in practice.

Acknowledgements We acknowledge support from ANR-19-P3IA-0003 (MIAI - Grenoble Alpes), NSF DMS 2023166, DMS 1839371, CCF 2019844, the CIFAR program “Learning in Machines and Brains”, and faculty research awards.

References

1. Abadi, M., et al.: Tensorflow: A system for large-scale machine learning. In: OSDI (2016)
2. Beck, A., Teboulle, M.: Smoothing and First Order Methods: A Unified Framework. SIAM Journal on Optimization **22**(2), 557–580 (2012)
3. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust Optimization, vol. 28. Princeton University Press (2009)
4. Ben-Tal, A., Teboulle, M.: Expected utility, penalty functions, and duality in stochastic nonlinear programming. Management Science **32**(11), 1445–1466 (1986)
5. Ben-Tal, A., Teboulle, M.: An Old-New Concept of Convex Risk Measures: The Optimized Certainty Equivalent. Mathematical Finance **17**(3), 449–476 (2007)
6. Chen, C., Mangasarian, O.L.: A class of smoothing functions for nonlinear and mixed complementarity problems. Computational Optimization and Applications **5**(2), 97–138 (1996)
7. Cucker, F., Zhou, D.X.: Learning Theory: An Approximation Theory Viewpoint, vol. 24. Cambridge University Press (2007)
8. Curi, S., Levy, K.Y., Jegelka, S., Krause, A.: Adaptive sampling for stochastic risk-averse learning. NeurIPS (2020)
9. Dantzig, G.B.: Discrete-variable extremum problems. Operations research **5**(2), 266–288 (1957)
10. Duchi, J., Namkoong, H.: Learning models with uniform performance via distributionally robust optimization. arXiv preprint arXiv:1810.08750 (2018)
11. Fan, Y., Lyu, S., Ying, Y., Hu, B.G.: Learning with average top-k loss. In: NIPS (2017)

[6] The code is publicly available at https://github.com/yassine-laguel/spqr.

12. Föllmer, H., Schied, A.: Convex measures of risk and trading constraints. Finance and stochastics **6**(4), 429–447 (2002)
13. Guigues, V., Sagastizabal, C.: Risk-averse feasible policies for large-scale multistage stochastic linear programs. Mathematical Programming **138** (2012). DOI 10.1007/s10107-012-0592-1
14. Hiriart-Urruty, J.B., Lemaréchal, C.: Convex Analysis and Minimization Algorithms. Springer Verlag, Heidelberg (1993). Two volumes
15. Ho-Nguyen, N., Wright, S.J.: Adversarial classification via distributional robustness with wasserstein ambiguity. preprint arXiv:2005.13815 (2020)
16. Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudik, M., Wallach, H.: Improving fairness in machine learning systems: What do industry practitioners need? In: CHI, pp. 1–16 (2019)
17. Howard, R.A., Matheson, J.E.: Risk-sensitive markov decision processes. Management science **18**(7), 356–369 (1972)
18. Kairouz, P., et al.: Advances and open problems in federated learning. arXiv Preprint (2019)
19. Kamishima, T., Akaho, S., Asoh, H., Sakuma, J.: Fairness-Aware Classifier with Prejudice Remover Regularizer. In: ECML PKDD, pp. 35–50. Springer (2012)
20. Kawaguchi, K., Lu, H.: Ordered sgd: A new stochastic optimization framework for empirical risk minimization. In: International Conference on Artificial Intelligence and Statistics, pp. 669–679. PMLR (2020)
21. Knight, W.: A self-driving Uber has killed a pedestrian in Arizona. Ethical Tech (2018)
22. Laguel, Y., Malick, J., Harchaoui, Z.: First-order optimization for superquantile-based supervised learning. In: IEEE MLSP (2020)
23. Laguel, Y., Pillutla, K., Malick, J., Harchaoui, Z.: A Superquantile Approach to Federated Learning with Heterogeneous Devices. In: IEEE CISS (2021)
24. Lee, J., Park, S., Shin, J.: Learning Bounds for Risk-sensitive Learning. In: NeurIPS (2020)
25. Levy, D., Carmon, Y., Duchi, J.C., Sidford, A.: Large-scale methods for distributionally robust optimization. NeurIPS (2020)
26. Luna, J.P., Sagastizábal, C., Solodov, M.: An approximation scheme for a class of risk-averse stochastic equilibrium problems. Mathematical Programming **157**(2), 451–481 (2016)
27. Metz, R.: Microsoft's neo-Nazi sexbot was a great lesson for makers of AI assistants. Artificial Intelligence (2018)
28. Mhammedi, Z., Guedj, B., Williamson, R.C.: PAC-Bayesian Bound for the Conditional Value at Risk. In: NeurIPS (2020)
29. Miranda, S.I.: Superquantile regression: theory, algorithms, and applications. Tech. rep., Naval postgraduate school Monterey ca (2014)
30. Morimura, T., Sugiyama, M., Kashima, H., Hachiya, H., Tanaka, T.: Nonparametric return distribution approximation for reinforcement learning. In: ICML (2010)

31. Nesterov, Y.: Smooth minimization of non-smooth functions. Mathematical programming **103**(1), 127–152 (2005)
32. Nesterov, Y.: Introductory lectures on convex optimization: A basic course, vol. 87. Springer Science & Business Media (2013)
33. Nesterov, Y., Spokoiny, V.: Random gradient-free minimization of convex functions. Foundations of Computational Mathematics **17**(2) (2017)
34. Nocedal, J., Wright, S.: Numerical optimization. Springer Science & Business Media (2006)
35. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: NIPS-W (2017)
36. Pedregosa, F., et al.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research (2011)
37. Pollard, D.: A user's guide to measure theoretic probability. 8. Cambridge University Press (2002)
38. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? arXiv preprint arXiv:1902.10811 (2019)
39. Rockafellar, R.T.: Solving stochastic programming problems with risk measures by progressive hedging. Set-Valued and Variational Analysis **26**(4), 759–768 (2018)
40. Rockafellar, R.T., Royset, J.O.: Superquantiles and their applications to risk, random variables, and regression. In: Theory Driven by Influential Applications, pp. 151–167. INFORMS (2013)
41. Rockafellar, R.T., Royset, J.O.: Random variables, monotone relations, and convex analysis. Mathematical Programming **148**(1-2) (2014)
42. Rockafellar, R.T., Royset, J.O., Miranda, S.I.: Superquantile regression with applications to buffered reliability, uncertainty quantification, and conditional value-at-risk. European Journal of Operational Research **234**(1), 140–154 (2014)
43. Rockafellar, R.T., Uryasev, S.: Conditional value-at-risk for general loss distributions. Journal of banking & finance **26**(7), 1443–1471 (2002)
44. Rockafellar, R.T., Uryasev, S., et al.: Optimization of conditional value-at-risk. Journal of risk **2**, 21–42 (2000)
45. Rockafellar, R.T., Wets, R.J.B.: Variational analysis, vol. 317. Springer Science & Business Media (2009)
46. Ruszczyński, A., Shapiro, A.: Optimization of convex risk functions. Mathematics of operations research **31**(3), 433–452 (2006)
47. Sarykalin, S., Serraino, G., Uryasev, S.: Value-at-risk vs. conditional value-at-risk in risk management and optimization. In: State-of-the-art decision-making tools in the information-intensive age, pp. 270–294. Informs (2008)
48. Shafieezadeh-Abadeh, S., Kuhn, D., Esfahani, P.M.: Regularization via mass transportation. Journal of Machine Learning Research **20**(103), 1–68 (2019)
49. Shalev-Shwartz, S., Ben-David, S.: Understanding machine learning: From theory to algorithms. Cambridge university press (2014)

50. Shapiro, A., Dentcheva, D., Ruszczynski, A.: Lectures on Stochastic Programming: Modeling and Theory, Second Edition. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2014)
51. Soma, T., Yoshida, Y.: Statistical Learning with Conditional Value at Risk. arXiv preprint arXiv:2002.05826 (2020)
52. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
53. Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. In: Advances in neural information processing systems, pp. 1057–1063 (2000)
54. Tamar, A., Chow, Y., Ghavamzadeh, M., Mannor, S.: Policy gradient for coherent risk measures. In: Advances in Neural Information Processing Systems, pp. 1468–1476 (2015)
55. Vershynin, R.: High-Dimensional Probability: An Introduction with Applications in Data Science. Cambridge university press (2018)
56. Wainwright, M.J.: High-Dimensional Statistics: A Non-Asymptotic Viewpoint. Cambridge University Press (2019)
57. Williamson, R.C., Menon, A.K.: Fairness Risk Measures. In: International Conference on Machine Learning (2019)

A Proof of Theorem 1

In this appendix, we provide a complete proof of Theorem 1. For classical results in this spirit, we refer to the monograph [7]. For discussions on statistical aspects of statistical learning, we refer to e.g., [10, 24, 28].

The key step in the proof of Theorem 1 is to show the uniform convergence

$$S^p_n(w) \to S^p(w) \text{ almost surely for all } w \in W. \tag{37}$$

Indeed, once we have this, the result immediately follows as

$$\begin{aligned} 0 \le S^p(w^\star_n) - S^p(w^\star) &= S^p(w^\star_n) - S^p_n(w^\star_n) + S^p_n(w^\star_n) - S^p_n(w^\star) + S^p_n(w^\star) - S^p(w^\star) \\ &\le 2 \sup_{w \in W} |S^p_n(w) - S^p(w)| \to 0, \end{aligned}$$

where we use $S^p_n(w^\star_n) \le S^p_n(w^\star)$ in the second inequality.

In order to prove (37), we use the variational expression of the superquantile (4). We define

$$\bar{S^p}(w, \eta) = \eta + \frac{1}{1-p} \mathbb{E}_{(x,y)\sim P}[\max(\ell(y, \varphi(w, x)) - \eta, 0)] ,$$

so that, using that the loss is bounded by B, we can write

$$S^p(w) = \min_{\eta \in [0,B]} \bar{S^p}(w, \eta).$$

We define the analogous empirical version $\bar{S^p_n}(w, \eta)$ so that $S^p_n(w) = \min_{\eta \in [0,B]} \bar{S^p_n}(w, \eta)$.

Claim 1: Under Assumption 2, the random variable

$$\delta_n(w, \eta) := \bar{S^p_n}(w, \eta) - \bar{S^p}(w, \eta)$$

has mean zero, lies almost surely in $[-B, B]$*, and satisfies*

$$|\delta_n(w,\eta) - \delta_n(w',\eta')| \leq 2M \operatorname{dist}_\varphi(w,w') + 2|\eta - \eta'| \,. \tag{38}$$

Note first that $\mathbb{E}[\bar{S}^p_n(w,\eta)] = \bar{S^p}(w,\eta)$ and that the boundedness of δ_n comes from the boundedness of the loss function. The Lipschitzness of δ_n also comes from the one of the loss function, as follows. Using that $\max\{\cdot, 0\}$ is 1-Lipschitz and that the loss ℓ is M-Lipschitz, we get

$$\begin{aligned} |\max\{\ell(y,\varphi(w,x)) - \eta, 0\} - &\max\{\ell(y,\varphi(w',x)) - \eta', 0\}| \\ &\leq |\ell(y,\varphi(w,x)) - \ell(y,\varphi(w',x))| + |\eta - \eta'| \\ &\leq M\|\varphi(w,x) - \varphi(w',x)\| + |\eta - \eta'| \\ &\leq M \operatorname{dist}_\varphi(w,w') + |\eta - \eta'| \,. \end{aligned}$$

Then, (38) simply follows from the triangle inequality, and Claim 1 is proved.

The next step in the proof is, for a given $\varepsilon > 0$

- to construct a cover T of $W \times [0, B]$, and then
- to control the convergence over the points of T, more precisely to control the probability of the event
$$E_n(\varepsilon) = \bigcap_{(w,\eta)\in T} \{\delta_n(w,\eta) \leq \varepsilon/2\} \;.$$

First, using Assumption 1, we consider T_1 a $(\varepsilon/(8M))$-cover of W with respect to $\operatorname{dist}_\varphi$. We also consider T_2 a uniform discretization of the line segment $[0, B]$ at width $\varepsilon/8$. We can introduce the cover of $W \times [0, B]$

$$T = T_1 \times T_2 \subset W \times [0, B].$$

Since, $|T_2| = 8B/\varepsilon$, we have that $|T| = (8B/\varepsilon)N(\varepsilon/(8M))$.

To get uniform convergence, it is sufficient to control what happens at points of T. Indeed, for any (w,η), there exists a point $(w',\eta') \in T$ such that $\operatorname{dist}_\varphi(w,w') \leq \varepsilon/(8M)$ and $|\eta - \eta'| \leq \varepsilon/8$. As a consequence, if the event $E_n(\varepsilon)$ holds, then

$$\begin{aligned} \delta_n(w,\eta) &\leq \delta_n(w',\eta') + |\delta_n(w,\eta) - \delta_n(w',\eta')| \\ &\overset{(38)}{\leq} \delta_n(w',\eta') + 2M \operatorname{dist}_\varphi(w,w') + 2|\eta - \eta'| \\ &\leq \frac{\varepsilon}{2} + \frac{\varepsilon}{4} + \frac{\varepsilon}{4} = \varepsilon. \end{aligned}$$

This implies that events of interest are included in $\overline{E}_n(\varepsilon)$, the complement of $E_n(\varepsilon)$; we have indeed

$$\{\exists w \,:\, |S^p_n(w) - S^p(w)| > \varepsilon\} \subset \{\exists (w,\eta) \,:\, \delta_n(w,\eta) > \varepsilon\} \subset \overline{E}_n(\varepsilon),$$

so that we have the following bound on the sum of probabilities

$$\sum_{n=1}^{\infty} \mathbb{P}\Big(\exists w \,:\, |S^p_n(w) - S^p(w)| > \varepsilon\Big) \leq \sum_{n=1}^{\infty} \mathbb{P}\big(\overline{E}_n(\varepsilon)\big). \tag{39}$$

Claim 2: The probabilities of the complements of $E_n(\varepsilon)$ *are summable, i.e.,*

$$\sum_{n=1}^{\infty} \mathbb{P}\big(\overline{E}_n(\varepsilon)\big) < \infty \,.$$

This is a direct application of the Hoeffding's inequality (see e.g. [55, Theorem 2.2.2]) as follows. For any fixed $(w,\eta) \in W \times [0,B]$, the Hoeffding's inequality gives

$$\mathbb{P}(|\delta_n(w,\eta)| > \varepsilon/2) \leq 2\exp\left(-\frac{n\varepsilon^2}{2B^2}\right).$$

Applied to all $(w,\eta) \in T$, this yields

$$\mathbb{P}\big(\overline{E}_n(\varepsilon)\big) \leq 2|T|\exp\left(-\frac{n\varepsilon^2}{2B^2}\right) = \frac{16B}{\varepsilon}\, N\left(\frac{\varepsilon}{8M}\right)\exp\left(-\frac{n\varepsilon^2}{2B^2}\right).$$

and proves Claim 2.

Finally, we conclude on the uniform convergence (37) with the Borel-Cantelli Lemma by the classical rationale (see e.g. the textbook [37, Chap. 2, Sec. 6]): the bound (39) and Claim 2 give that the probabilities for any ε are summable; applying Borel-Cantelli with the sequence $\varepsilon_k = 1/k$ gives the uniform convergence (37), which completes the proof of the theorem.

B Numerical Illustrations

We provide simple illustrations of the interest of using superquantile for machine learning. More precisely, we reproduce the experimental framework of the computational experiments of [8] and we solve the superquantile optimization problems with the approach depicted here, by combining smoothing and quasi-Newton. For additional experiments with other datasets, metrics, and contexts, we refer to [8].

We consider two basic machine learning tasks (regression and classification) with linear prediction functions $\varphi(w,x) = w^\top x$ and with two standard datasets, from the UCI ML repository. Denoting these datasets $P_n = \{(x_i,y_i)\}_{1\leq i\leq n}$, we introduce the (regularized) empirical risk minimization

$$\min_{w\in\mathbb{R}^{\mathrm{d}}} \quad \mathbb{E}_{(x,y)\sim P_n}\left[\ell(y,w^\top x)\right] + \frac{1}{2n}\|w\|^2.$$

and its smoothed superquantile analogous

$$\min_{w\in\mathbb{R}^{\mathrm{d}}} \quad [\mathbb{S}_p^\nu]_{(x,y)\sim P_n}\left[\ell(y,w^\top x)\right] + \frac{1}{2n}\|w\|^2.$$

We solve these problems using L-BFGS via the toolbox SPQR [22] offering an simple user-interface and implementing the oracles (with the Euclidean smoothing of Example 5 for the smoothed approximation).

Regression and Least-Squares

We consider a regularised least square regression on the dataset Abalone from the UCI Machine learning repository. We perform a 80%/20% train-test split on the dataset. We minimize the least-squares loss on the training set both in expectation and with respect to the superquantile (with $p = 0.98$ and $\nu = 0.1$).

We report on Figure 6 the distribution of errors $|y_i - w^\top x_i|$ for the testing dataset for both models w (standard in blue and superquantile in red). We observe that the superquantile model exhibits a thinner upper tail than the risk-neutral model, which is quantified by the shift to the left of 0.98 quantile. This comes at the price of lower performance in expectations than the model trained with expectation, which is clear visible on the picture and quantified by the shift to the right of the mean.

Classification and Logistic regression

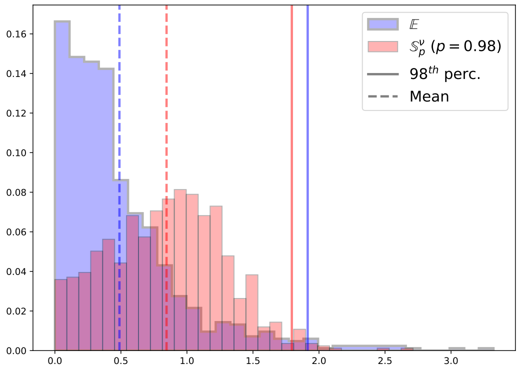


Fig. 6: Regression: Histogram of the regression errors on the testing dataset for the model learning by the superquantile approach (red) compared to the one of the classical empirical risk minimization (violet). We see a reshaping of the histogram of errors and a gain on worst-case errors.

We consider a logistic regression on the Australian Credit dataset. We randomly split the dataset with a 80%/20% train-test split for 5 different seeds. For each seed, we perform a pessimistic distributional shift on the training dataset by downsampling the majority class (similarly to what is done in [8, Sec. 5.2]). More precisely, we remove an important fraction of the majority class, randomly selected, so that it counts afterward for only 10% of the minority class. We tune then the safety level parameter p by a k-cross validation on the shifted dataset and select the safety parameter yielding the best validation accuracy. The grid we use for tuning this parameter is [0.8, 0.85, 0.9, 0.95, 0.99] We finally compute with this parameter the testing accuracy and the testing precision.

We report in Table 2 the testing accuracy and the testing precision, averaged over the 5 different seeds, with the associated standard deviation. We observe that the superquantile model brings better performance for both in terms of accuracy and precision than the standard model.

Model	Accuracy	Precision
Standard	0.65 ± 0.03	0.56 ± 0.04
Superquantile	0.69 ± 0.04	0.60 ± 0.05

Table 2: Classification: Better testing accuracy and precision for the superquantile approach, in the case of distributional shifts.