Robust Regression with Asymmetric Heavy-Tail Noise Distributions

Technical Report 1198
Université de Montréal, DIRO
CP 6128, Succ. Centre-Ville Montréal, Québec, Canada

Ichiro Takeuchi, Yoshua Bengio
Université de Montréal, DIRO
CP 6128, Succ. Centre-Ville
Montréal, Québec, Canada
{takeuchi, bengioy}@iro.umontreal.ca

Takafumi Kanamori*
Dept. of Mathematical and Computing Sciences,
Tokyo Institute of Technology
Ookayama 2-12-1, Meguro-ku, Tokyo 152-8552, Japan
kanamori@is.titech.ac.jp

July 6th, 2001

Abstract

In the presence of a heavy-tail noise distribution, regression becomes much more difficult. Traditional robust regression methods assume that the noise distribution is symmetric and they downweight the influence of so-called outliers. When the noise distribution is asymmetric these methods yield strongly biased regression estimators. Motivated by data-mining problems for the insurance industry, we propose in this paper a new approach to robust regression that is tailored to deal with the case where the noise distribution is asymmetric. The main idea is to learn most of the parameters of the model using conditional quantile estimators (which are biased but robust estimators of the regression), and to learn a few remaining parameters to combine and correct these estimators, to minimize the average squared error. Theoretical analysis and experiments show the clear advantages of the approach. Results are on artificial data as well as real insurance data, using both linear and neural-network predictors.

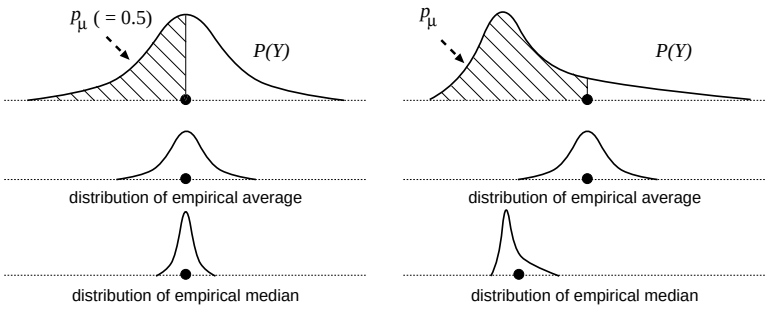
* This work has been done while Takafumi Kanamori was at Université de Montréal, DIRO, CP 6128, Succ. Centre-Ville, Montréal, Québec, Canada.

1 Introduction

In a variety of practical applications, we often find data distributions with an asymmetric heavy tail extending out towards more positive values, as in Figure 1 (ii). Modeling data with such an *asymmetric heavy-tail distribution* is essentially difficult because *outliers*, which are sampled from the tail of the distribution, have a strong influence on parameter estimation. When the distribution is *symmetric* (around the mean), the problems caused by outliers can be reduced using *robust* estimation techniques [1, 2, 3] which basically intend to ignore or put less weights on outliers. Note that these techniques do not work for an asymmetric distribution: most outliers are on the same side of the mean, so downweighting them introduces a strong bias on its estimation.
Our goal is to estimate the conditional expectation $E[Y|X]$ (measuring performance in the least-square sense). Regression can also suffer from the effect of outliers when the distribution of the noise (the variations of the output variable Y that are not explainable by the input variable X) has an asymmetric heavy tail. As in the unconditional case, the robust methods which downweight outliers[3] do not work for asymmetric noise distributions. We propose a new robust regression method which can be applied to data with an asymmetric heavy-tail noise distribution. The regressor can be linear or non-linear (e.g., approximating the desired class of functions with a multi-layer neural network). The proposed method can provably approximate the conditional expectation (provided the approximating class is large enough) for a wide range of noise structures including additive noise, multiplicative noise and combinations of both. Different variants of the proposed method are compared both theoretically and experimentally to Least Squares (LS) regression (both linear and non-linear). We demonstrate the effectiveness of the proposed method with numerical experiments on artificial datasets and with an application to a auto-insurance premium estimation problem in which the data have an asymmetric heavy-tail noise distribution. Throughout the paper, we use the following notations: X and Y for the input and the output random variables respectively, $F_W(\cdot)$ and $P_W(\cdot)$ for, respectively, the cumulative distribution function (cdf) and the probability density function (pdf) of random variable W.

2 Robust Methods

Let us first consider the easier task of estimating from a finite sample the *unconditional* mean $E[Y]$ of a heavy-tail distribution (i.e., the density decays slowly to zero when going towards ∞ or $-\infty$). The empirical average may here be a poor estimator because the few points sampled from the tails are highly variable and influence greatly the empirical average. In the case of a symmetric distribution, we can downweight or ignore the effect of these outliers in order to greatly reduce this variability. For example, the median estimator is much less sensitive to outliers than the empirical average for heavy-tail distributions.
This idea can be generalized to *conditional* estimators, for example one can estimate the regression $E[Y|X]$ from an estimated *conditional median* by minimizing



(i: symmetric distribution) (ii: asymmetric distribution)

Figure 1: The schematic illustration of empirical averages and empirical medians for (i) symmetric distribution and for (ii) asymmetric distribution: Distributions of a heavy-tail random variable Y, empirical average and empirical median and their expectations (black circles) are illustrated. Note that in (i) those expectations coincide, while in (ii) they do not. As indicated by dashed areas, we define p_μ as the order of the quantile coinciding with the mean. In (i), $p_\mu = 0.5$.

absolute errors:

$$\hat{f}_{0.5} = \underset{f \in \mathcal{F}}{\operatorname{argmin}} \sum_i |y_i - f(x_i)|, \tag{1}$$

where $\mathcal{F}$ is a set of functions (e.g., a class of neural networks), $\{(x_i, y_i), i = 1, 2, \cdots, N\}$ is the training sample, and a $\hat{}$ *hat* denotes estimation on data. Minimizing the above over $P(X, Y)$ and a large enough class of functions yields the conditional median $f_{0.5}$, i.e., $P(Y < f_{0.5}(X)|X) = 0.5$. For regression problems with a heavy tail noise distribution, the estimated *conditional median* is much less sensitive to outliers than least squares (LS) regression (which provides an estimate of the *conditional average* of Y given X).

Unfortunately, this method and other methods which downweight the influence of outliers [1, 2, 3] do not work for *asymmetric distributions*. Whereas removing outliers symmetrically on both sides of the mean does not change the average, removing them on one side changes the average considerably. For example, the median of an asymmetric distribution does not coincide with its mean (and the more asymmetric the distribution, the more they differ). Note that, instead of the median, there is **another quantile that coincides with the mean**. We call its order p_μ: i.e. for a distribution $P(Y)$, we define $p_\mu \triangleq F_Y(E[Y]) = P(Y < E[Y])$. Note that $p_\mu > 0.5$ (< 0.5) suggests $P(Y)$ is positively (negatively) skewed. When the distribution is symmetric, $p_\mu = 0.5$. See Figure 1.

For regression problems with an *asymmetric noise distribution*, we may extend median regression eq. (1) to p-th quantile regression [4] that estimates the conditional p-th quantile, i.e. we would like $P(Y < \hat{f}_p(X)|X) = p$:

$$\hat{f}_p = \underset{f \in \mathcal{F}_q}{\operatorname{argmin}} \left\{ \sum_{i: y_i \geq f(x_i)} p|y_i - f(x_i)| + \sum_{i: y_i < f(x_i)} (1-p)|y_i - f(x_i)| \right\}, \tag{2}$$

where $\mathcal{F}_q$ is a set of quantile regression functions. One could be tempted to obtain a robust regression with asymmetric noise distribution by estimating f_{p_μ}, instead of $f_{0.5}$. But what is p_μ for regression problems ? It must be defined as $p_\mu(x) \triangleq F_{Y|X}(E[Y|X=x])$=$P(Y < E[Y|X]|X=x)$. So the above idea raises 3 problems, which we will address with the algorithm proposed in the next section: (i) $p_\mu(x)$ of $P(Y|x)$ may depend on x in general, (ii) unless the noise distribution is known, p_μ itself must be estimated, which is maybe as difficult as estimating $E[Y|X]$, (iii) if the noise density at p_μ is low (because of the heavy tail and large value of p_μ), the estimator in eq. (2) may itself be very unstable. See Figure 3.

3 Robust Regression for Asymmetric Tails

3.1 Introductory Example

To explain the proposed method, we first consider an introductory example, a simple scalar linear regression problem with additive noise:

$$Y = aX + b + Z, \tag{3}$$

where Z is a zero-mean noise random variable (independent of X) with an asymmetric heavy-tail distribution whose (unknown) p_μ is 0.9, and a and b are parameters to estimate. Let us consider the 3 problems raised in the previous section. Consider (i): is p_μ of $P(Y|X)$ independent of X? yes: $P(Y < E[Y|X]|X) = P(aX + b + Z < aX + b) = P(Z < 0) = p_\mu$ does not depend on X. Concerning (ii) and (iii) we do not know a priori the value of p_μ, and even if we knew it, the quantile estimation at $p = 0.9$ might be very unstable (the 0.9-th empirical quantile sample is highly variable if it is in the tail). Therefore we propose to estimate a quantile regression at $p = 0.5$, ideally *near the mode of the distribution.* Let us call f_{p_1} this ideal quantile regression and $\hat{f}_{p_1}$ the estimated function. It can be noted that $f_{p_1}(x)$ and $E[Y|x]$ can both be written as linear functions of x with the same coefficient a (but a different intercept b' and b). This suggests the following strategy: (1) estimate a using $p = 0.5$ quantile regression, (2) keeping $\hat{a}$ fixed, estimate b to minimize the least squares error. In this way, b might not be estimated any better, but at least a is *. This idea is illustrated in Figure 3.

3.2 Algorithm

To overcome the difficulties raised above, we propose a new algorithm, RRAT, for **Robust Regression for Asymmetric Tails**. The main idea is to learn most of the parameters of the model using conditional quantile estimators (which are biased but robust), and to learn a few remaining parameters to combine and correct these estimators.

*parameter a is B-robust [2] as shown later in appendix ??

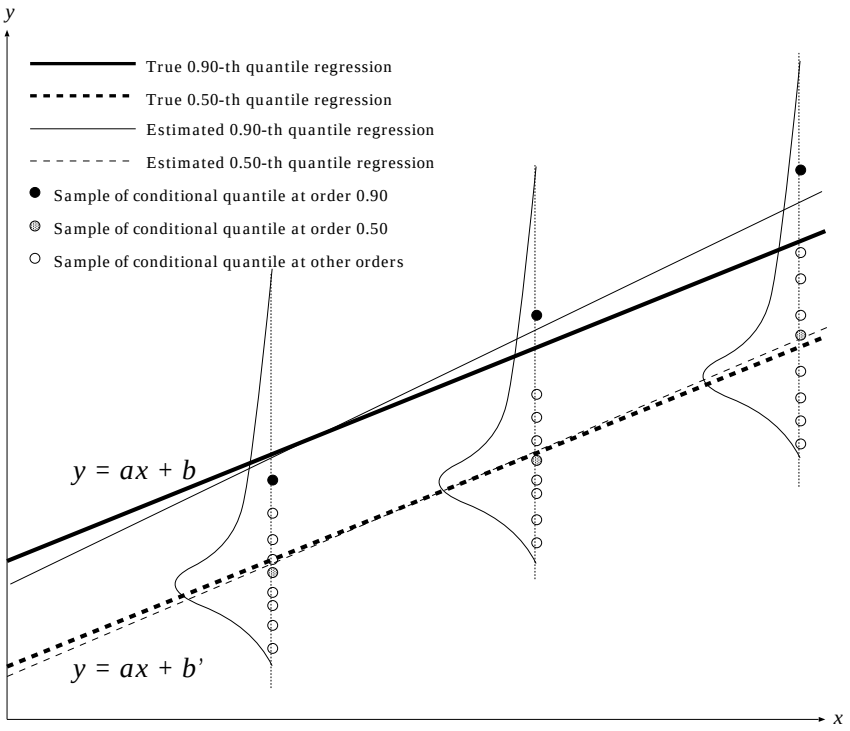


Figure 2: Example of eq. (**??**) with order $p_\mu = 0.9$ and $p_1 = 0.5$. Small circles are samples at a few values of x (fixed for illustration), y is drawn from indicated pdfs. Note that slope parameter (a) is common to both p_μ and p_1-th quantile regressions and only intercept parameters (b and b') are different. Note also that y values around order p_μ (black circles) are more variable than those around order p_1 (grey circles). It suggests that p_μ-th quantile regression yields worse estimates than p_1-th quantile regression.

Algorithm RRAT(n)

Input: data pairs $\{(x_i, y_i)\}$, quantile orders $(p_1, \cdots, p_n)$, function classes $\mathcal{F}_q$ and $\mathcal{F}_c$.
(1) Fit n quantile regressions at orders $p_1, p_2, \cdots p_n$, each as in eq. (2), yielding functions $\hat{f}_{p_1}, \hat{f}_{p_2}, \cdots, \hat{f}_{p_n}$, with $\hat{f}_{p_i} : \mathbb{R}^d \to \mathbb{R}$, with $\hat{f}_{p_i} \in \mathcal{F}_q$.
(2) Fit a least squares regression with inputs $q(x_i) = (\hat{f}_{p_1}(x_i), \cdots \hat{f}_{p_n}(x_i))$ and targets y_i, yielding a function $\hat{f}_c : \mathbb{R}^n \to \mathbb{R}$, with $\hat{f}_c \in \mathcal{F}_c$.
Output: conditional expectation estimator $\hat{f}(x) = \hat{f}_c(q(x))$.

Some of the parameters are estimated through conditional quantile estimators $f_{p_1}, \cdots, f_{p_n}$ and the latter are ***C***ombined and ***C***orrected by the function f_c in order to estimate $E[Y|X]$. In the above example $n = 1$ and $p_1 = 0.5$ is chosen,

$\hat{f}_{p_1}$ is linear in x, and $\hat{f}_c$ just has one additive parameter. In general, we believe that RRAT yields more robust regressions when the number of parameters required to characterize $\hat{f}_c$ is small (because they are estimated with "non-robust" least squares) compared to the number of parameters required to characterize the quantile regressions f_{p_i}.
Problem (ii) above is dealt with by doing quantile regressions of orders $p_1 \cdots p_n$ not necessarily equal to p_μ. Problem (iii) is dealt with if $p_1 \cdots p_n$ are in high density areas (where estimation will be robust). The issue raised with the remaining problem (i) will be discussed in the next subsection.

3.3 Applicable Class of Problems

A class of regression problems for which the above strategy works (in the sense that the analog of problem (i) above is eliminated) can be described as follows:

$$Y = g_\mu(X) + Z\, g_\sigma(g_\mu(X)), \tag{4}$$

where Z is a zero-mean random variable drawn from any form of (possibly asymmetric) continuous distribution, independent of X. The conditional expectation is characterized by an arbitrary function g_μ and the conditional standard deviation of the noise distribution is characterized by an arbitrary positive range function g_σ. Note that the regression in eq. (3) is a subclass of heteroscedastic regression [5], i.e. the standard deviation of the noise distribution is not directly conditioned on X, but only on $g_\mu(X)$. This specification narrows the class of applicable problems, but eq. (3) still covers a wide variety of noise structures as explained later.
In eq. (3), $E[Y|X] = E[g_\mu(X) + Z\, g_\sigma(g_\mu(X))|X] = g_\mu(X)$ and p_μ of the distribution $P(Z)$ coincides with p_μ of $P(Y|X)$ and does not depend on x, i.e. $P(Y < E[Y|X]|X) = P(Z\, g_\sigma(g_\mu(X)) < 0 \mid X) = P(Z < 0 \mid X) = P(Z < 0)$. Since the conditional expectation $E[Y|X]$ coincides with p_μ-th quantile regression of $P(Y|X)$, we have $g_\mu(x) \equiv f_{p_\mu}(x)$.

The following two theorems show that RRAT(n) works if appropriate choices of $p_1, \cdots, p_n$ and large enough classes of $\mathcal{F}_q$ and $\mathcal{F}_c$ are provided, by guaranteeing the existence of the function f_c that transforms the outputs of p_i-th quantile regressions $f_{p_i}(x)$ $(i = 1, \cdots, n)$ into the conditional expectation $E[Y|x]$ for all x. The cases of $n = 1$ and $n = 2$ are explained in theorem 1 and theorem 2, respectively.
The applicable class of problems with only one quantile regression f_{p_1}, i.e. RRAT(1), is smaller than the class satisfying eq. (3), but it is very important for practical applications (see subsection 3.3).

> **Theorem 1.** If the noise structure is as in eq. (3) then there exists a function f_c such that $E[Y|X] = f_c(f_{p_1}(X))$, where $f_{p_1}(X)$ is the p_1-th quantile regression, if and only if function $h(\bar{y}) = \bar{y} + F_Z^{-1}(p_1) \cdot g_\sigma(\bar{y})$ is monotonic with respect to $\bar{y} \in \{E[Y|x] \mid \forall\, x\}$ (proof in appendix A).

With the use of two quantile regressions f_{p_1} and f_{p_2}, we show that RRAT(2), covers the whole of the class satisfying eq. (3).

Theorem 2. If the noise structure is as in eq. (3) and

$$p_1 \neq p_\mu, \quad p_2 \neq p_\mu, \tag{5}$$

$$p_1 \neq p_2, \tag{6}$$

then there exists a function f_c such that $E[Y|X] = f_c(f_{p_1}(X), f_{p_2}(X))$, where $f_{p_i}(X)$ are the p_i-quantile regressions ($i = 1, 2$) (proof in appendix B).

In comparison to Theorem 1, we see that when using $n = 2$ quantile regressions, the monotonicity condition can be dropped. We **conjecture** that even the assumption of noise structure eq. (3) can be dropped when combining a sufficient number of quantile regressions. However, this may add more complexity (and parameters) to f_c, thereby reducing the gains brought by the approach.
One of the likely advantages to use a number of quantile regressions is that it increases the possibilities of choosing "good" p_i in the sense that (I) the probability density at p_i, i.e. $P_Z(F_Z^{-1}(p_i))$, is large enough, **and** that (II) p_i is near p_μ. The second property (II) is appreciable when eq. (3) does not hold globally to the whole of the conditional distribution $P(Y|X)$ but only does locally to the part covering p_i and p_μ-th quantiles of $P(Y|X)$. For example, in application to insurance premium estimation, the noise distribution Z is not continuous at zero because most customers do not file any claims (the claim amounts of those are zero). We can avoid this problem by being careful when we choose p_i so that f_{p_i} is never zero.

3.4 Some Properties for Practical Applications

Consider the case where g_σ (in eq. (3)) is affine in $g_\mu(x)$, and $g_\mu(x) \geq 0$ for all x,

$$Y = g_\mu(X) + Z \times (c + d\, g_\mu(X)), \tag{7}$$

where c and d are constants such that $c \geq 0,\ d \geq 0,\ (c, d) \neq (0, 0)$. The conditions of theorem 1 (including monotonicity of $h(y)$) are verified for **additive noise** ($d = 0$), **multiplicative noise** ($c = 0$), or combinations of both ($c > 0, d > 0$), for **any** form of noise distribution $P(Z)$ (continuous and independent of X).

Property 1. If the noise structure is affine eq. (6) and $g_\mu(x) \geq 0$ for all x, then a linear function f_c is sufficient to map f_{p_1} to f, i.e., only two parameters need to be estimated by least squares (proof in appendix C).

Note that additive/multiplicative noise covers a very large variety of practical problems[†], so this result shows that RRAT(1) already enjoys wide applicability. Figure **??** illustrates the above discussion.
Let us call **risk** of an estimator $\hat{f}(X)$ the expected squared difference between $E[Y|X]$ and $\hat{f}(X)$ (expectation over X, Y and the training set). Let us write

[†]The specification of the conditional expectations to be non-negative is generally a trivial problem because we can shift the output data. Also note that in many practical applications with asymmetric heavy-tail noise distributions (including our application to insurance premium estimation), the output range is non-negative.

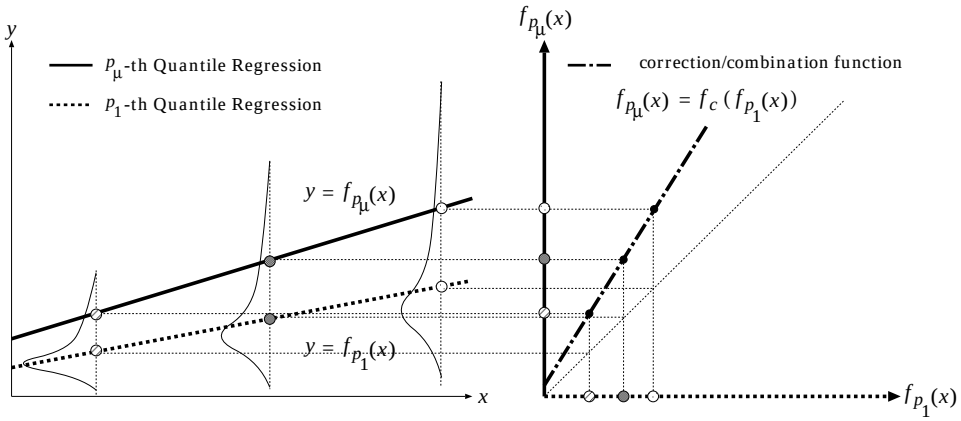


Figure 3: A schematic illustration of theorem 1 and property 1: The left figure illustrates p_μ-th quantile regression, i.e. $E[Y|X]$, and p_1-th quantile regression which we actually estimate in step (1) of RRAT algorithm. Right figure illustrates the correction/combination function f_c, which maps from $f_{p_1}(x)$ to $f_{p_\mu}(x)$. Theorem 1 and property 1 gurantees the existence of f_c under a certain condition. The figure demonstrates that some outputs of $f_{p_1}(x)$ (indicated by various types of circles) can be transformed into the corresponding output of $f_{p_1}(x) \equiv E[Y|x]$ (indicated by the circle of the same type) by function f_c.

$\hat{f}_c(\hat{f}_{p_1}(X))$ for the conditional expectation obtained by RRAT(1) with finite variance.

Property 2. Consider the class of distributions $Y = f(X;\alpha^*) + \beta^* + Z$, where f is an arbitrary function characterized by a set of parameters $\alpha^* \in \mathbb{R}^d$ (with the assumption that the second derivative matrix of $f(X;\alpha^*)$ with respect to α^* is full-rank.) and $\beta^* \in \mathbb{R}$ is a scalar parameter, and where Z is the zero-mean noise random variable drawn from a (possibly asymmetric) heavy-tail distribution (independent of X). Then, the risk of LS regression is given by $\frac{1}{n}(d+1)Var[Z]$ and the risk of RRAT(1) is given by $\frac{1}{n}(Var[Z] + d\frac{p_1(1-p_1)}{P_Z^2(F_Z^{-1}(p_1))})$. It follows that the risk of RRAT(1) is less than that of LS regression if and only if $\frac{p_1(1-p_1)}{P_Z^2(F_Z^{-1}(p_1))} < Var[Z]$ (proof in appendix D).

For instance, as we have also verified numerically, if Z is *log-normal* (see sec. 4) RRAT beats LS regression in average when $p_\mu > 0.608$ (recall that for symmetric distributions $p_\mu = 0.5$).

4 Numerical Experiments

To investigate and demonstrate the proposed RRAT algorithm, we did a series of numerical experiments using artificial data as well as actual insurance data.

The experiments are designed for two major objectives. The first objective is to understand *when* RRAT(n) works better than LS, i.e. for which classes of g_μ, g_σ and Z in eq. (3) it works. The second objective is to figure out *which* RRAT works better than the other among variants, i.e. the choices of n, $p_1, p_2, \cdots, p_n$, $\mathcal{F}_q$ and $\mathcal{F}_c$. This section focusses on the synthetic data experiments, while the next section presents the results on actual insurance data.

4.1 Overall Experimental Setting

In the synthetic data experiments, sample pairs (x_i, y_i) are generated as per eq. (3) with:

$$x_i \sim U[0,1]^d, \tag{8}$$

$$y_i = g_\mu(x_i) + z_i(p_\mu)\, g_\sigma(g_\mu(x_i)), \tag{9}$$

where $g_\mu : \mathbb{R}^d \to \mathbb{R}$, $g_\sigma : \mathbb{R} \to (0, \infty)$ and $z_i(p_\mu)$ is a random sample from log-Normal distribution $\mathrm{LN}(z_0(p_\mu), 0, \sigma^2(p_\mu))$ [7]: $Z(p_\mu) = z_0(p_\mu) + e^{W(p_\mu)}$, where $z_0(p_\mu)$ is the location parameter so that $E[Z(p_\mu)] = 0$ and where $W(p_\mu)$ follows a Normal distribution $N(0, \sigma^2(p_\mu))$ with $\sigma^2(p_\mu)$ so that $p_\mu = P(Z(p_\mu) < E[Z(p_\mu)])$. We tried several choices of g_μ, g_σ and p_μ, which allowed us to investigate the performance of the proposed RRAT algorithm with respect to (1) the complexity of the approximated conditional expectation function, (2) the noise structures and (3) the degree of "asymmetry" of the noise distribution.

The RRAT algorithm requires us to specify the number of quantile regressions n, each order $p_i, i = 1, \cdots, n$ and classes of functions $\mathcal{F}_p$ and $\mathcal{F}_c$. We tried several choices of n and p_i to find out how they affect the performance and to get some implications for practical applications. In terms of the choice of $\mathcal{F}_p$, we used well-specified affine models to approximate a linear conditional expectation function, and a multi-layer neural network (NN) model to approximate a non-linear conditional expectation function. (NNs can be shown to be good approximators not only for ordinary LS regression but also for quantile regression [8].) We always used affine models for $\mathcal{F}_c$ because parameter estimation for f_c is not robust and it is thus preferable for it to have as few parameters as possible.

We used 1000 training samples for parameter estimation. The parameters of f_{p_i} are estimated iteratively using the conjugate gradient descent method for affine models and using the stochastic gradient descent method for NN models. The parameters of f_c are estimated analytically. The performance of the model is quantified with the test-set average of **model squared error**, i.e. the squared difference between true $E[Y|X](\equiv g_\mu(X))$ and estimated $\hat{f}_c(\hat{f}_{p_1}(X), \cdots, \hat{f}_{p_n}(X))$ on 1000 test samples. The performance of RRAT was compared with that of Least Square (LS) regression. In each comparison, the same classes of models (e.g. affine, NN) were used. The parameters of the compared models were estimated on the same training samples analytically when they are linear (with respect to their parameters) and iteratively by stochastic gradient descent method when they are non-linear. The models were compared on the basis of model squared error on the same test samples. In each experimental setting, a number of experiments are repeated using completely independent datasets. The statistical comparisons

between RRAT and LS regression are given by a Wilcoxon signed rank test[‡] with a null hypothesis that the model squared error in each experiment is not statistically different. (In this section, we use the term "significant" for 0.01 level.)

4.2 Experiment 1 (When does RRAT work?)

A series of experiments are designed to investigate the performance of RRAT with respect to g_μ, g_σ and p_μ in eq. (8). We chose g_μ either from a class of affine functions with 2, 5 or 10 inputs (with parameters sampled from $U[0,1]$ in each experiment), or from the class of 6-input non-linear functions [§] described in [9]. g_σ was chosen to be either additive($g_\sigma(w) = 1 + w$), multiplicative($g_\sigma(w) = 0.1w$), or the combination of both($g_\sigma(w) = 1+0.1w$). We tried positive skews, with $p_\mu \in \{0.55, 0.60, \cdots, 0.95\}$, without loss of generality. In the series of experiments explained in this subsection, we fixed $n = 1$ and $p_1 = 0.50$. We used well specified f_c, i.e. in additive noise case: $f_c(w) = c + w$, in multiplicative noise case; $f_c(w) = d\,w$, and in combination case: $f_c(w) = c + d\,w$. Parameter estimation both for f_{p_1} and f_c were done without specific capacity control (no weight decay and the number of hidden units in NN models are fixed to 10 both for quantile regression and for LS regression.). The number of independent experiments is 500 for the linear models and 50 for the non-linear models. The results are summarized in Table **??** and Figure 3.
Table **??** shows the results of statistical comparisons of all combinations of g_μ, g_σ

[‡] We used a non-parametric test rather than a Student t-test because the normality assumption in the t-test does not hold in the presence of heavy tails.

[§] $g_\mu(x) = 10\sin(\pi x_1 x_2) + 20\sin(x_3 - 0.5)^2 + 10x_4 + 5x_5$ (does not depend on x_6).

g_μ	g_σ	d	p_μ (degree of asymmetry)								
			.55	.60	.65	.70	.75	.80	.85	.90	.95
affine	additive	2	$*$	-	○	○	○	○	○	○	○
		5	$*$	-	○	○	○	○	○	○	○
		10	$*$	$*$	○	○	○	○	○	○	○
	multipl.	2	-	-	○	○	○	○	○	○	○
		5	$*$	-	○	○	○	○	○	○	○
		10	$*$	-	○	○	○	○	○	○	○
	combin.	2	$*$	-	○	○	○	○	○	○	○
		5	$*$	-	○	○	○	○	○	○	○
		10	$*$	$*$	○	○	○	○	○	○	○
non-linear	additive	6	$*$	-	○	○	○	○	○	○	-
	multipl.	6	$*$	○	○	○	○	○	○	○	-
	combin.	6	$*$	○	○	○	○	○	○	-	$*$

Table 1: Statistical comparisons by a Wilcoxon signed rank test. '$*$': LS-regression significantly better than RRAT, '-': no significant difference, '○': RRAT significantly better than LS. d = number of inputs. In most cases, RRAT is significantly (0.01 level) better than LS when $p_\mu > 0.608$ as analytically expected.

and p_μ. As expected from the asymptotic analysis of property 2, RRAT(1) works better than LS-regression when p_μ (the degree of asymmetry) is more than 0.608 in most cases. There are few cases where the differences were not significant or LS regression works better than RRAT when the predictor is non-linear and p_μ is large.
This is not what asymptotic analysis suggests and the explanation is maybe related to the overwhelming negative effects of the heavy tail on the estimation of a few parameters by LS (step 2 of the algorithm). Figure 3 shows graphically the effect of p_μ and the number of parameters of f_{p_1} on the difference between the performance of RRAT and LS regression, in terms of the logarithm of the difference in average model squared error between RRAT and LS regression. (Note that the corresponding statistical significances are given in Table **??**.) In Figure 3, (i), (ii) and (iii) are for additive, multiplicative and combination of additive/multiplicative noise structures, respectively. In each figure, experimental curves for 2, 5 and 10-dimensional affine models and for the non-linear models are given. In additive noise case (i), the theoretical curves derived from property 2 are also indicated. Note that the more asymmetric the noise distribution is (for larger p_μ), the better RRAT (relatively) works. Note also that as the number of parameters estimated through quantile regression f_{p_1} increases, the relative improvement brought by RRAT over LS regression increases. [¶] The relative improvements decrease when the predictor is non-linear and p_μ is fairly large (but we do not have a good explanation yet).

4.3 Experiment 2 (Which RRAT works)

Another series of experiments are designed to compare the performance of RRAT for varying choices of n, $p_1, \cdots, p_n$. We tried $n \in \{1, 2, 3\}$ and $p_i \in \{0.2, 0.5, 0.8\}$. In the series of experiments in this subsection, we fixed g_μ a 2-dimensional affine function, g_σ additive ($g_\sigma(w) = 1.0 + w$) and $p_\mu = 0.75$. $\mathcal{F}_p$ is the class of 2-dimensional affine models and $\mathcal{F}_c$ is the class of additive constant models, i.e. they are well-specified. For parameter estimation, we introduced capacity control with a weight decay parameter w_d (penalty on the 2-norm of the parameters) chosen from $\{10^{-5}, 10^{-4}, \cdots, 10^{+5}\}$. The best weight decay was chosen with another 1000 validation samples that is independent of training and test samples. From 100 independent experiments, we obtained the results summarized in Figure **??**.
Figure **??** shows the mean and its standard error of the model squared error in each method. The p-values for the Wilcoxon signed rank test (null hypothesis of no difference) are also indicated. From the given p-values, it is clear that all variants of RRAT work significantly better than LS regression. Note that the choice of p_i does not change the performance considerably. In RRAT(1), the true $p_\mu = 0.75$, RRAT(1) with $p_1 = 0.20$ or 0.50 also worked as well as RRAT(1) with $p_1 = 0.80$. On the other hand, the choice of n changes the performance considerably. (The p-values of the significant difference between RRAT(1) and RRAT(2) were in the range: $[4.31 \times 10^{-9}, 8.65 \times 10^{-8}]$, those between RRAT(1) and RRAT(3) were in the range: $[7.73 \times 10^{-8}, 1.62 \times 10^{-7}]$ and those between RRAT(2) and RRAT(3) were in the range: $[3.50 \times 10^{-1}, 4.67 \times 10^{-1}]$.) As we assumed additive noise structure

[¶] The number of parameters of affine models are 3, 6 and 11, respectively and 81 for the NN.

here, RRAT(1) is sufficient and RRAT(n), $n \geq 2$, are redundant as explained in property 1. When the noise structure is more complicated, RRAT(1) might not be sufficient and RRAT(n) with larger n might be more suitable, at the cost of reducing the gains brought by RRAT.

5 Application to Insurance Premium Estimation

We have applied RRAT to an automobile insurance premium estimation: estimate the risk of a driver given his/her profile (age, type of car, etc.). One of the challenges in this problem is that the premium must take into account the tiny fraction of drivers who cause serious accidents and file large claim amounts. That is, the data (claim amounts) has noise distributed with an asymmetric heavy tail extending out towards positive values.
The number of input variables of the dataset is 39, all discrete except one. The discrete variables are one-hot encoded, yielding input vectors with 266 elements. We repeated the experiment 10 times using each time an independent dataset, by randomly splitting a large data set with 150,000 samples into 10 independent subsets with 15,000 samples. Each subset is then randomly split in 3 equal sub-

LS regression	
mean	8.96×10^{-2}
standard error	1.04×10^{-2}

RRAT(1)			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	3.45×10^{-2}	3.41×10^{-2}	3.45×10^{-2}
standard error	3.54×10^{-3}	3.51×10^{-3}	3.52×10^{-3}
p-value	9.76×10^{-15}	8.31×10^{-15}	9.25×10^{-15}

RRAT(2)			
	$p_1 = .2$, $p_2 = .5$	$p_1 = .2$, $p_2 = .8$	$p_1 = .5$, $p_2 = .8$
mean	4.56×10^{-2}	4.61×10^{-2}	4.52×10^{-2}
standard error	4.22×10^{-3}	4.11×10^{-3}	4.20×10^{-3}
p-value	1.73×10^{-11}	4.87×10^{-11}	2.34×10^{-11}

RRAT(3)	
	$p_1 = .2$, $p_2 = .5$, $p_3 = .8$
mean	4.66×10^{-2}
standard error	4.17×10^{-3}
p-value	3.97×10^{-11}

Table 2: The mean and its standard error of the average model squared error in each method. In the tables for RRAT(n), $n = 1, 2, 3$, the p-values for the Wilcoxon signed rank test are also indicated. All variants of RRAT work significantly better than LS regression. Note that the choice of p_i does not change the performance considerably, but the choice of n does.

sets with 5000 samples respectively for training, validation (model selection), and testing (model comparison).
In the experiment, we compared RRAT and LS regression using linear and NN quantile predictors, i.e., $\mathcal{F}_q$ is affine or a NN, always fitted using conjugate gradients. Capacity is controlled via weight decay $\in \{10^{-5}, 10^{-4}, \cdots 10^{5}\}$ (and the number of hidden units $\in \{5, 10, \cdots, 25\}$ and early stopping in the case of the NN), and selected using the validation set. The correction/combination function f_c is always affine (also with weight decay chosen using the validation set) and its parameters are estimated analytically. We tried RRAT(n) with $n = 1, 2, 3$ and $p_i = 0.20, 0.50, 0.80$. For RRAT(1), we tried either additive, affine, or quadratic correction/combination functions: $f_c \in \{c_0 + f_{p_1}, c_0 + c_1 f_{p_1}, c_0 + c_1 f_{p_1} + c_2 f_{p_1}^2\}$. For RRAT($n$), $n \geq 2$, we tried affine $f_c \in \{c_0 + c_1 f_{p_1} + c_2 f_{p_2} + \cdots\}$.
Figure 4 (linear model cases) and Figure 5 (NN model cases) show the mean (and its standard error) of the average squared error in each method as well as the p-values for the Wilcoxon signed rank test, where '$*$' ('$**$') denotes LS regression being significantly better than RRAT at 0.05 (0.01) level, '$-$' denotes no significant difference between them and '$\circ$' ('$\circ\circ$') denotes RRAT being significantly better than LS regression at 0.05 (0.01) level [‖]. In RRAT(1), the choice of $p_1 = 0.20$ does not work, which suggests either that the underlying distribution of the dataset is out of the class of eq .(3) **or** the noise structure is more complicated than those tried. When $p_1 = 0.50$, the choice of f_c significantly changed the performance. The worse performance of additive f_c and better performance of affine f_c suggests that the noise structure of the dataset is more multiplicative than additive. When $p_1 = 0.80$, RRAT worked better independently of the choice of f_c, that suggests the "true" p_μ of the dataset (if it does not vary too much with x) stays around 0.80. Note that RRAT(n), $n \geq 2$ always worked better than LS, even though one of the components ($f_{0.20}$) was very poor by itself. The choice of p_i was therefore not critical to the success of the application, i.e. we can choose several p_i and combine them. Furthermore, when we pick the best of the RRAT models (choice of n, p_i and f_c from above) based on the validation set average squared error, even better results are obtained.

6 Conclusion

We have introduced a new robust regression method, RRAT(n), that is well suited to asymmetric heavy-tail distributions (where previous robust regression methods are not well suited). It can be applied to both linear and non-linear regressions. A large class of generating models for which a universal approximation property holds has been characterized (and it includes additive and multiplicative noise with arbitrary conditional expectation). Theoretical analysis of a large class of asymmetric heavy-tail noise distributions reveals when the proposed method beats least-square regression. The proposed method has been tested and analyzed both on synthetic

[‖] The standard errors are fairly large because the effect of outliers significantly varies the mean in each dataset. However most of p-values are small enough because the statistical tests are on paired samples. The outliers in each dataset affect in a consistent way both RRAT and LS regression.

data and on insurance data (which were the motivation for this research), showing it to significantly outperform least squares regression.

Acknowledgements

The authors would like to thank Léon Bottou and Christian Léger for stimulating discussions, as well as the following agencies for funding: NSERC, MITACS, and Japan Society for the Promotion of Science.

A Proof of Theorem 1

proof: For all x, $f_{p_1}(x)$ is represented as a function of $f_{p_\mu}(x)$ as follows:

$$\begin{aligned}\forall\, x,\ \ f_{p_1}(x) &= f_{p_\mu}(x) - \left(F^{-1}_{Z\cdot g_\sigma\{g_\mu(x)\}}(p_\mu) - F^{-1}_{Z\cdot g_\sigma\{g_\mu(x)\}}(p_1)\right) \\ &= f_{p_\mu}(x) + F^{-1}_{Z\cdot g_\sigma\{g_\mu(x)\}}(p_1) \\ &= f_{p_\mu}(x) + F^{-1}_{Z}(p_1)\cdot g_\sigma\{g_\mu(x)\} \\ &= g_\mu(x) + F^{-1}_{Z}(p_1)\cdot g_\sigma\{g_\mu(x)\}, \qquad (10)\end{aligned}$$

where note that $F^{-1}_W(p)$ is the p-th quantile of random variable W. In eq. (9) from the first line to the second, we used $E[Z\ g_\sigma\{g_\mu(X)\}|X] = 0$. From the second to third, we used the following property of cdfs: if $F_{aW}(av) = F_W(v)$ then $F^{-1}_{aW}(p) = a\cdot F^{-1}_W(p)$, where $a > 0$, $0 < p < 1$ are constants and W is a random variable drawn from a continuous distribution. From the third to fourth, we used $f_{p_\mu}(x) \equiv g_\mu(x)$ for all x.
If the monotonicity of $h(\bar{y})$ in the theorem holds, there is a one-to-one mapping between $f_{p_1}(x)$ and $g_\mu(x)$ $(= E[Y|x])$. It follows that there exists a function f_c such that $E[Y|X] = f_c(f_{p_1}(X))$. **Q.E.D**.

B Proof of Theorem 2

proof: As in eq. (9) in the proof of theorem 1, $f_{p_1}(x)$ and $f_{p_2}(x)$ are given, for all x, by

$$\begin{aligned}f_{p_1}(x) &= g_\mu(x) + F^{-1}_Z(p_1)\cdot g_\sigma(g_\mu(x)), \qquad (11)\\ f_{p_2}(x) &= g_\mu(x) + F^{-1}_Z(p_2)\cdot g_\sigma(g_\mu(x)) \qquad (12)\end{aligned}$$

As in theorem 1, it is necessary and sufficient to prove that the mapping between $g_\mu(x)$ and a pair of $\{f_{p_1}(x), f_{p_2}(x)\}$ characterized by eq. (10) and eq. (11) is one-to-one for all x.
Assume that it is **not** one-to-one, then there is at least one case where two different values $\bar{y}$ and $\bar{y}'$ are mapped to identical $\{f_{p_1}(x), f_{p_2}(x)\}$, i.e.

$$\begin{aligned}\bar{y} - F^{-1}_Z(p_1)\cdot g_\sigma(\bar{y}) &= \bar{y}' - F^{-1}_Z(p_1)\cdot g_\sigma(\bar{y}'), \qquad (13)\\ \bar{y} - F^{-1}_Z(p_2)\cdot g_\sigma(\bar{y}) &= \bar{y}' - F^{-1}_Z(p_2)\cdot g_\sigma(\bar{y}'). \qquad (14)\end{aligned}$$

Dividing both sides of eqn.(12) and (13) by $F_Z^{-1}(p_1)$ and $F_Z^{-1}(p_2)$, respectively (from the continuity of Z and eq. (5), $F_Z^{-1}(p_1)$ and $F_Z^{-1}(p_2)$ are not zero) and substracting eqn.(12) from (13) yields

$$(\frac{1}{F_Z^{-1}(p_1)} - \frac{1}{F_Z^{-1}(p_2)})(\bar{y} - \bar{y}') = 0 \tag{15}$$

From the continuity of Z and eq. (5), $\frac{1}{F_Z^{-1}(p_1)} \neq \frac{1}{F_Z^{-1}(p_2)}$ and from the assumption $\bar{y} \neq \bar{y}'$, we find that the assumption that the mapping between $g_\mu(x)$ and $\{f_{p_1}(x), f_{p_2}(x)\}$ is **not** one-to-one is **false**.
It follows that there exists a function f_c such that $E[Y|X] = f_c(f_{p_1}(X), f_{p_2}(X))$. **Q.E.D**.

C Proof of property 1

proof: By applying g_σ in eqn.(6) into eqn.(9), we get

$$f_{p_1}(x) = g_\mu(x) - F_Z^{-1}(p_1) \cdot (c + d \cdot g_\mu(x)). \tag{16}$$

We can exactly obtain a linear function f_c:

$$\begin{aligned} g_\mu(x) &= f_c(f_{p_1}(x)) \\ &= \frac{-F_Z^{-1}(p_1) \cdot c}{1 - F_Z^{-1}(p_1) \cdot d} + \frac{1}{1 - F_Z^{-1}(p_1) \cdot d} \cdot f_{p_1}(x) \end{aligned} \tag{17}$$

except the case where the denominator appearing in the right-hand side of the second line in eq. (16) is zero ** **Q.E.D**

D Proof of property 2

proof: We consider a class of regression problems in the following form:

$$Y = f(X; \alpha^*) + \beta^* + Z, \tag{18}$$

where X and Y are the input and output random variables, respectively. f is an arbitrary function characterized by a set of parameters $\alpha^* \in \mathbb{R}^d$ and $\beta^* \in \mathbb{R}$ is a scalar parameter. Z is the (zero-mean) noise random variable drawn from a (possibly) asymmetric heavy-tail distribution (independent of X). We assume that the model is well specified, i.e. the true function $f(x; \alpha^*) + \beta^*$ is a member of the class of parametric models: $M = \{f(x; \alpha) + \beta : \alpha \in \Theta \subset \mathbb{R}^d, \beta \in \mathbb{R}\}$, and that the training data is i.i.d.

This problem can be solved just by choosing another order $p_2 \neq p_1$. In application of this method, we suggest to try several orders $p_1, p_2, \cdots, p_n$ and choose the best one or use RRAT(n), $n \geq 2$. This is done in sec. 4 and 5. Note that this problem happens, in the schematic illustration in Figure **??, when p_1-th quantile regression is parallel to horizontal axis, i.e. p_1-th quantile regression does not provide any dependencies between x and y.

The risk of the model by least squares (LS) regression is given by standard calculation of asymptotic statistics:

$$\begin{aligned} \mathrm{risk}_{\mathrm{LS}} &= E_{Data}\left\{E_X\{(f(x;\alpha^*)+\beta^*-f(x;\hat{\alpha}_{\mathrm{LS}})-\hat{\beta}_{\mathrm{LS}})^2\}\right\} \\ &= \frac{1}{n}\{Var(Z)+d\,Var(Z)\}+o\left(\frac{1}{n}\right) \end{aligned} \tag{19}$$

where n is the number of training data and $\hat{\alpha}_{\mathrm{LS}}$, $\hat{\beta}_{\mathrm{LS}}$ denotes the corresponding estimated parameters by LS-regression.
As explained, RRAT provides the following estimates:

$$\begin{aligned} \{\hat{\alpha}_{\mathrm{RRAT}},\hat{\gamma}_{\mathrm{RRAT}}\} &= \underset{\alpha,\gamma}{\mathrm{argmin}}\left\{\sum_{i:y_i\geq f(x_i;\alpha)+\gamma} p_1\,|y_i-f(x_i;\alpha)-\gamma| \right. \\ &\qquad \left. +\sum_{i:y_i<f(x_i;\alpha)+\gamma}(1-p_1)\,|y_i-f(x_i;\alpha)-\gamma|\right\}, \end{aligned} \tag{20}$$

$$\hat{\beta}_{\mathrm{RRAT}} = \frac{1}{n}\sum_{i=1}^{n}\{y_i-f(x_i;\hat{\alpha}_{\mathrm{RRAT}})\}. \tag{21}$$

Using these estimated parameters, we will show that the risk of RRAT(1) is

$$\begin{aligned} \mathrm{risk}_{\mathrm{RRAT}} &= E_{Data}\left\{E_X\{(f(x;\alpha^*)+\beta^*-f(x;\hat{\alpha}_{\mathrm{RRAT}})-\hat{\beta}_{\mathrm{RRAT}})^2\}\right\} \\ &= \frac{1}{n}\left\{Var(z)+d\cdot\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}\right\}+o\left(\frac{1}{n}\right), \end{aligned} \tag{22}$$

where P_Z and F_Z are the pdf and cdf of the noise distribution.
From eq. (18) and eq. (21), RRAT yields more efficient estimates than LS-regression when

$$Var(Z)>\frac{p_1(1-p_1)}{P_Z^2(F_Z^{-1}(p_1))}. \tag{23}$$

To prove eq. (21), let us introduce some notation. We define $|w|^+=\max(0,w)$. We also omit the subscript $_{\mathrm{RRAT}}$ for parameters estimated by RRAT.
First of all we define the matrix K as

$$K=\int\begin{pmatrix}\frac{d}{d\alpha}f(x;\alpha^*)\frac{d}{d\alpha}f(x;\alpha^*)' & \frac{d}{d\alpha}f(x;\alpha^*)\\ \frac{d}{d\alpha}f(x;\alpha^*)' & 1\end{pmatrix}p(x)dx=\begin{pmatrix}M_2 & m_1\\ m_1' & 1\end{pmatrix},$$

where M_2 is a $d\times d$ matrix and m_1 is a $d\times 1$ vector. The risk of the RRAT is written as

$$\mathrm{risk}=\mathrm{Tr}\;Var(\hat{\alpha},\hat{\beta})K+o\left(\frac{1}{n}\right).$$

To obtain the variance of $\hat{\alpha}$ we can use the following relation between the variance and the influence function [2]:

$$\lim_{n\to\infty}nVar(\hat{\alpha},\hat{\gamma})=\int IF(x,y)IF(x,y)'p(y|x)p(x)dydx,$$

where $IF(x,y)$ is the influence function of the estimator $(\hat{\alpha},\hat{\gamma})$. The influence function is defined as

$$IF(\tilde{x},\tilde{y}) = \lim_{\kappa\to 0}\frac{(\alpha_\kappa,\gamma_\kappa)-(\alpha^*,\gamma^*)}{\kappa} \tag{26}$$

where $(\alpha_\kappa,\gamma_\kappa)$ is given by minimizing the following with respect to (α,γ):

$$\begin{aligned}&(1-\kappa)\int\left\{p_1\left|y-f(x;\alpha)-\gamma\right|^+ + (1-p_1)\left|f(x;\alpha)+\gamma-y\right|^+\right\}p(y|x)p(x)dydx\\ &+\kappa\left\{p_1|\tilde{y}-f(\tilde{x};\alpha)-\gamma|^+ + (1-p_1)|f(\tilde{x};\alpha)+\gamma-\tilde{y}|^+\right\}.\end{aligned}$$

To obtain the influence function we use $\frac{d}{dx}|x|^+ = \sigma(x)$ [††] and $\frac{d}{dx}\sigma(x) = \delta(x)$ [‡‡]. We obtain $(\alpha_\kappa,\gamma_\kappa)$ as follows:

$$\begin{aligned}(\alpha_\kappa,\gamma_\kappa)' &= (\alpha^*,\gamma^*)' - \kappa\frac{1}{P_Z(F_Z^{-1}(p_1))}\left\{1-p_1-\sigma(\tilde{y}-f(\tilde{x};\alpha^*)-\gamma^*)\right\}\\ &\quad\cdot K^{-1}\begin{pmatrix}\frac{d}{d\alpha}f(\tilde{x};\alpha^*)\\ 1\end{pmatrix}+o(\kappa).\end{aligned} \tag{27}$$

The influence function is

$$IF(\tilde{x},\tilde{y}) = \frac{1}{P_Z(F_Z^{-1}(p_1))}\left\{1-p_1-\sigma(\tilde{y}-f(\tilde{x};\alpha^*)-\gamma^*)\right\}K^{-1}\begin{pmatrix}\frac{d}{d\alpha}f(\tilde{x};\alpha^*)\\ 1\end{pmatrix}$$

and

$$Var(\hat{\alpha},\hat{\gamma}) = \frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}K^{-1}+o\left(\frac{1}{n}\right).$$

We decompose K^{-1} as

$$K^{-1} = \begin{pmatrix}H & t\\ t' & u\end{pmatrix}$$

and write

$$Var(\hat{\alpha}) = \frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}H+o\left(\frac{1}{n}\right). \tag{28}$$

[††] $\sigma(x)$ is 1 when $x\geq 0$ and 0 when $x<0$.

[‡‡] $\delta(x)$ is Dirac's delta function.

Next we calculate the variance of $\hat{\beta}$ in (20):

$$\begin{aligned} Var(\hat{\beta}) &= Var\left(\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i|\hat{\alpha}))\right) \\ &= \frac{1}{n^2}\sum_{i,j} E_{Data}\{z_i z_j\} \qquad (29) \\ &\quad + \frac{1}{n^2}\sum_{i,j} E_{Data}\left\{\frac{d}{d\alpha}f(x_i;\alpha^*)'(\hat{\alpha}-\alpha^*)(\hat{\alpha}-\alpha^*)'\frac{d}{d\alpha}f(x_j;\alpha^*)\right\} \qquad (30) \\ &\quad - \frac{2}{n^2}\sum_{i,j} E_{Data}\left\{z_i(\hat{\alpha}-\alpha^*)'\frac{d}{d\alpha}f(x_j;\alpha^*)\right\} \qquad (31) \\ &\quad - \frac{2}{n^2}\sum_{i,j} E_{Data}\left\{z_i(\hat{\alpha}-\alpha^*)'\frac{\partial^2 f(x_j;\alpha^*)}{\partial\alpha^2}(\hat{\alpha}-\alpha^*)\right\} \qquad (32) \\ &\quad + o\left(\frac{1}{n}\right) \end{aligned}$$

The first term (28) is equal to $\frac{Var(z)}{n}$. The second term (29) is calculated as follows. First,

$$E_{z|X}\left\{(\hat{\alpha}-\alpha^*)(\hat{\alpha}-\alpha^*)'\right\} = \frac{1}{n}\frac{p_1(1-p_1)}{p_z(m_1)^2}H + o_p\left(\frac{1}{n}\right), \qquad (33)$$

where $o_p(\cdot)$ is the probabilistic order with respect to $p(x_1,\ldots,x_n)$. Substituting (32) into (29),

$$\begin{aligned} &\frac{1}{n^2}\sum_{i,j} E_{Data}\left\{\frac{d}{d\alpha}f(x_i;\alpha^*)'(\hat{\alpha}-\alpha^*)(\hat{\alpha}-\alpha^*)'\frac{d}{d\alpha}f(x_j;\alpha^*)\right\} \\ = &\frac{1}{n^2}\sum_{i,j}\mathrm{Tr}\left\{\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}HE_X\left\{\frac{d}{d\alpha}f(x_i;\alpha^*)\frac{d}{d\alpha}f(x_j;\alpha^*)'\right\} + o\left(\frac{1}{n}\right)\right\} \\ = &\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}m_1'Hm_1 + o\left(\frac{1}{n}\right). \qquad (34) \end{aligned}$$

The third term (30) is calculated as follows. First calculate $E_{z|X}\{z_i(\hat{\alpha}-\alpha^*)\}$. Then define $(\hat{\alpha}_{(i)},\hat{\gamma}_{(i)})$ as the estimator which is obtained from all the training data except (x_i,y_i). By definition $(\hat{\alpha}_{(i)},\hat{\gamma}_{(i)})$ is independent from z_i. Thus

$$\begin{aligned} \begin{pmatrix}\hat{\alpha}\\ \hat{\gamma}\end{pmatrix} &= \begin{pmatrix}\hat{\alpha}_{(i)}\\ \hat{\gamma}_{(i)}\end{pmatrix} - \frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left\{1-p_1-\sigma(y_i-f(x_i|\hat{\alpha})-\hat{\gamma})\right\}K^{-1}\begin{pmatrix}\frac{d}{d\alpha}f(x_i;\alpha^*)\\ 1\end{pmatrix} \\ &\quad + o_p\left(\frac{1}{n}\right), \qquad (35) \end{aligned}$$

and

$$\begin{aligned} \hat{\alpha}-\alpha^* &= \hat{\alpha}_{(i)}-\alpha^* - \frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left\{1-p_1-\sigma(y_i-f(x_i|\hat{\alpha})-\hat{\gamma})\right\}\left(H\frac{d}{d\alpha}f(x_i;\alpha^*)+t\right) \\ &\quad + o_p\left(\frac{1}{n}\right). \qquad (36) \end{aligned}$$

Substituting (35) in $E_{z|X}\{z_i(\hat{\alpha}-a^*)\}$ we obtain

$$E_{z|X}\{z_i(\hat{\alpha}-a^*)\} = \frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left(\int_0^\infty \tilde{z}p_z(\tilde{z})d\tilde{z}\right)\left(H\frac{d}{d\alpha}f(x_i;\alpha^*)+t\right)+o_p\left(\frac{1}{n}\right) \quad (37)$$

and

$$\begin{aligned}
& -\frac{2}{n^2}\sum_{i,j} E_{Data}\left\{z_i(\hat{\alpha}-\alpha^*)'\frac{d}{d\alpha}f(x;\alpha^*)\right\} \\
= & -\frac{2}{n^2}\frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left(\int_0^\infty \tilde{z}p_z(\tilde{z})d\tilde{z}\right)\sum_{i,j} E_X\left\{\mathrm{Tr}\left(H\frac{d}{d\alpha}f(x_i;\alpha^*)+t\right)\frac{d}{d\alpha}f(x_j;\alpha^*)'\right\}+o\left(\frac{1}{n}\right) \\
= & -\frac{2}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left(\int_0^\infty \tilde{z}p_z(\tilde{z})d\tilde{z}\right)\left(m_1'Hm_1+m_1't\right)+o\left(\frac{1}{n}\right) \\
= & \ o\left(\frac{1}{n}\right). \qquad (38)
\end{aligned}$$

The last equation is obtained from the definition of H and t, that is, we can find that $Hm_1+t=0$.
The fourth term (31) is also $o\left(\frac{1}{n}\right)$. This can be verified by substituting (35) into (31). From the previous discussion we obtain $\hat{\beta}$ as

$$Var(\hat{\beta}) = \frac{Var(z)}{n}+\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}m_1'Hm_1+o\left(\frac{1}{n}\right). \quad (39)$$

Next we calculate the covariance between $\hat{\alpha}$ and $\hat{\beta}$. $\hat{\beta}-\beta^*$ is written as

$$\hat{\beta}-\beta^* = \frac{1}{n}\sum_{i=1}^n z_i - \frac{1}{n}\sum_{i=1}^n(\hat{\alpha}-\alpha)'\frac{d}{d\alpha}f(x_i;\alpha^*)+o_p\left(\frac{1}{\sqrt{n}}\right) \quad (40)$$

Substituting the above equation into $E_{Data}\{(\hat{\beta}-\beta^*)(\hat{\alpha}-\alpha^*)\}$,

$$\begin{aligned}
& E_{Data}\{(\hat{\beta}-\beta^*)(\hat{\alpha}-\alpha^*)\} \\
= & \frac{1}{n}\sum_{i=1}^n E_X\left\{E_{z|X}\left\{z_i(\hat{\alpha}-\alpha^*)\right\}\right\}-\frac{1}{n}\sum_{i=1}^n E_X\left\{E_{z|X}\left\{(\hat{\alpha}-\alpha^*)(\hat{\alpha}-\alpha^*)'\right\}\frac{d}{d\alpha}f(x_i;\alpha^*)\right\} \\
& +o\left(\frac{1}{n}\right) \\
= & \frac{1}{n}\sum_{i=1}^n E_X\left\{\frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left(\int_0^\infty \tilde{z}p_z(\tilde{z})d\tilde{z}\right)\left(H\frac{d}{d\alpha}f(x_i;\alpha^*)+t\right)\right\} \\
& -\frac{1}{n}\sum_{i=1}^n E_X\left\{\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}H\frac{d}{d\alpha}f(x_i;\alpha^*)\right\}+o\left(\frac{1}{n}\right) \\
= & \frac{1}{n}\frac{1}{P_Z(F_Z^{-1}(p_1))}\left(\int_0^\infty \tilde{z}p_z(\tilde{z})d\tilde{z}\right)(Hm_1+t)-\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}Hm_1+o\left(\frac{1}{n}\right) \\
= & -\frac{1}{n}\frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2}Hm_1+o\left(\frac{1}{n}\right) \qquad (41)
\end{aligned}$$

Now we have all the elements for calculating the risk of the estimator $(\hat{\alpha}, \hat{\beta})$. The variance of $(\hat{\alpha}, \hat{\beta})$ is

$$Var(\hat{\alpha}, \hat{\beta}) = \frac{1}{n} \frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2} \begin{pmatrix} H & -Hm_1 \\ -m_1'H & \frac{P_Z(F_Z^{-1}(p_1))^2}{p_1(1-p_1)} Var(z) + m_1'Hm_1 \end{pmatrix} + o\left(\frac{1}{n}\right). \quad (42)$$

The risk is calculated as $\mathrm{Tr} Var(\hat{\alpha}, \hat{\beta}) K + o\left(\frac{1}{n}\right)$:

$$\begin{aligned} \mathrm{Tr} Var(\hat{\alpha}, \hat{\beta}) K &= \frac{1}{n} \frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2} \mathrm{Tr} \begin{pmatrix} H & -Hm_1 \\ -m_1'H & \frac{P_Z(F_Z^{-1}(p_1))^2}{p_1(1-p_1)} Var(z) + m_1'Hm_1 \end{pmatrix} \begin{pmatrix} M_2 & m_1 \\ m_1 & 1 \end{pmatrix} \\ &= \frac{1}{n} \frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2} \left\{ \mathrm{Tr} H M_2 - m_1'Hm_1 + \frac{P_Z(F_Z^{-1}(p_1))^2}{p_1(1-p_1)} Var(z) \right\} \\ &= \frac{1}{n} \left\{ Var(z) + d \cdot \frac{p_1(1-p_1)}{P_Z(F_Z^{-1}(p_1))^2} \right\}, \end{aligned} \quad (43)$$

where we use the relation

$$H = M_2^{-1} + \frac{1}{1 - m_1'M_2^{-1}m_1} M_2^{-1} m_1 m_1' M_2^{-1}.$$

Thus we obtain the assertion of eq. (21). **Q.E.D**.

References

[1] P.J. Huber. *Robust Statistics*. John Wiley & Sons Inc., 1982.

[2] F.R.Hampel, E.M.Ronchetti, P.J.Rousseeuw, and W.A.Stahel. *Robust Statistics, The Approach based on Influence Functions*. John Wiley & Sons, 1986.

[3] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons Inc., 1987.

[4] R. Koenker and Jr. G. Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

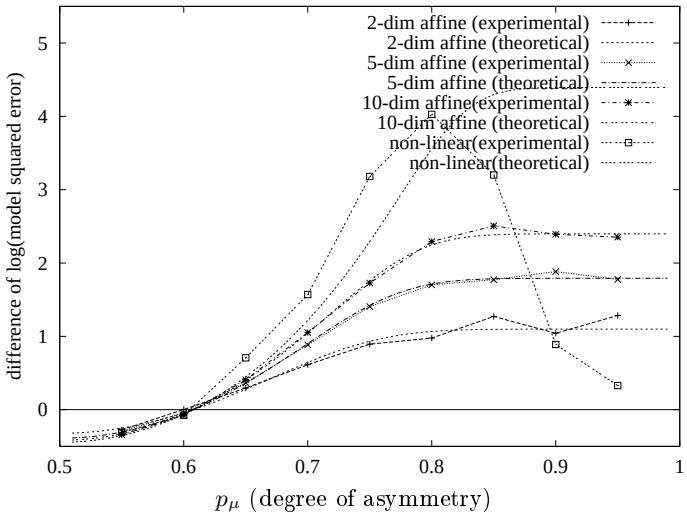
[5] W.H.Greene. *Econometric Analysis 3rd edition*. Prentice Hall, Inc., 1997.

[6] Ichiro Takeuchi, Yoshua Bengio, and Takafumi Kanamori. Robust regression with asymmetric heavy-tail noise. Technical Report 1198, Dept. IRO, Université de Montréal, 2001.

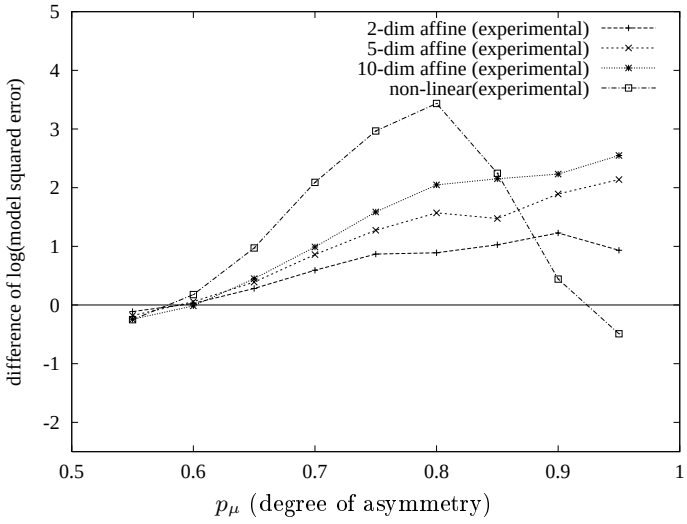
[7] C.E.Antle. *Lognormal Distribution*, volume 5, pages 134–136. John Wiley & Sons, 1985.

[8] H. White. Nonparametric estimation of conditional quantiles using neural networks. In *Proceedings of 23rd Symposium on the Interface, Computer Science and Statistics*, pages 190–199, 1991.

[9] J.H.Friedman, E.Grosse, and W.Suetzle. Multidimensional additive spline approximation. *SIAM Journal of Scientific and Statistical Computing*, 4(2):291–301, 1983.

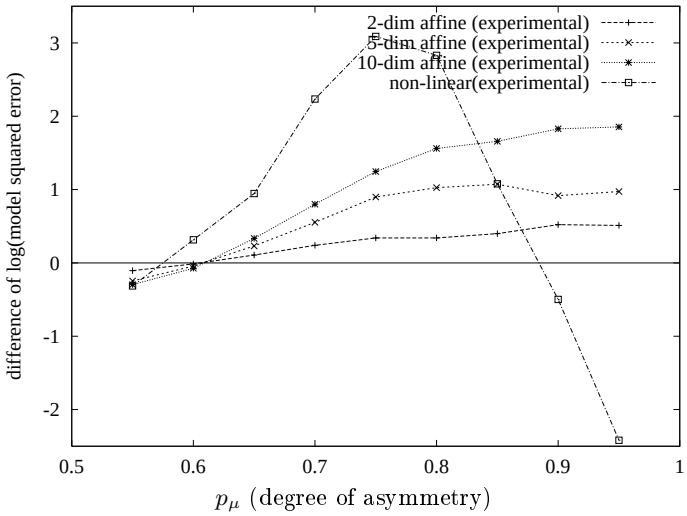


(i: additive noise structure)



(ii: multiplicative noise structure)

Figure 4: Effect of p_μ and number of parameters in f_{p_1} on relative performance of RRAT and LS regression when noise structure is (i) additive or (ii) multiplicative: values above 0 suggest RRAT is better than LS-regression. In (i), the theoretical curves are derived from property 2. More asymmetry and more parameters in f_{p_1} both generally increase the relative merit of RRAT. These numbers are 3,6, 11,respectively for affine models and 81 for the neural network.



(iii: combination of additive and multiplicative noise structure)

Figure 4 (continued): Effect of p_μ and number of parameters in f_{p_1} on relative performances of RRAT and LS regression when noise structure is (iii) combination of additive and multiplicative.

LS regression			
mean	5.289×10^2		
standard error	1.620×10^2		
RRAT(1), $f_c(f_{p_1}) = c_0 + f_{p_1}$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.366×10^2	5.346×10^2	5.275×10^2
standard error	1.643×10^2	1.640×10^2	1.619×10^2
p-value	8.302×10^{-3} $**$	2.967×10^{-2} $*$	2.343×10^{-2} $\circ$
RRAT(1), $f_c(f_{p_1}) = c_0 + c_1 f_{p_1}$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.318×10^2	5.277×10^2	5.266×10^2
standard error	1.615×10^2	1.618×10^2	1.618×10^2
p-value	1.013×10^{-1} -	3.723×10^{-2} $\circ$	3.455×10^{-3} $\circ\circ$
RRAT(1), $f_c(f_{p_1}) = c_0 + c_1 f_{p_1} + c_2 f_{p_1}^2$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.312×10^2	5.269×10^2	5.264×10^2
standard error	1.613×10^2	1.618×10^2	1.618×10^2
p-value	1.931×10^{-1} -	4.672×10^{-3} $\circ\circ$	2.531×10^{-3} $\circ\circ$
RRAT(2), $f_c(f_{p_1}, f_{p_2}) = c_0 + c_1 f_{p_1} + c_2 f_{p_2}$			
	$p_1 = .2$, $p_2 = .5$	$p_1 = .2$, $p_2 = .8$	$p_1 = .5$, $p_2 = .8$
mean	5.275×10^2	5.265×10^2	5.266×10^2
standard error	1.617×10^2	1.617×10^2	1.618×10^2
p-value	1.421×10^{-2} $\circ$	3.455×10^{-3} $\circ\circ$	3.455×10^{-3} $\circ\circ$
RRAT(3), $f_c(f_{p_1}, f_{p_2}, f_{p_3}) = c_0 + c_1 f_{p_1} + c_2 f_{p_2} + c_3 f_{p_3}$			
	$p_1 = .2$, $p_2 = .5$, $p_3 = .8$		
mean	5.265×10^2		
standard error	1.617×10^2		
p-value	2.531×10^{-3} $\circ\circ$		
Best model on validation			
mean	5.265×10^2		
standard error	1.618×10^2		
p-value	2.531×10^{-3} $\circ\circ$		

Figure 5: Insurance experiment comparison of RRAT and LS regression on linear predictors: The mean and its standard error of the average squared error in each method as well as the p-values from Wilcoxon signed rank test are indicated, where '$*$' ('$**$') denotes LS regression being significantly better than RRAT at 0.05 (0.01) level, '$-$' denotes no significant difference between them and '$\circ$' ('$\circ\circ$') denotes RRAT being significantly better than LS regression at 0.05 (0.01) level. **Note that RRAT(n), $n \geq 2$, always beat LS regression**.

LS regression			
mean	5.310×10^2		
standard error	1.633×10^2		
RRAT(1), $f_c(f_{p_1}) = c_0 + f_{p_1}$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.359×10^2	5.301×10^2	5.271×10^2
standard error	1.645×10^2	1.628×10^2	1.614×10^2
p-value	1.832×10^{-2} $*$	1.664×10^{-1} -	1.091×10^{-2} $\circ$
RRAT(1), $f_c(f_{p_1}) = c_0 + c_1 f_{p_1}$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.307×10^2	5.272×10^2	5.271×10^2
standard error	1.614×10^2	1.616×10^2	1.615×10^2
p-value	3.994×10^{-1} -	4.672×10^{-3} $\circ\circ$	1.833×10^{-2} $\circ$
RRAT(1), $f_c(f_{p_1}) = c_0 + c_1 f_{p_1} + c_2 f_{p_1}^2$			
	$p_1 = .2$	$p_1 = .5$	$p_1 = .8$
mean	5.301×10^2	5.272×10^2	5.271×10^2
standard error	1.611×10^2	1.616×10^2	1.616×10^2
p-value	4.392×10^{-1} -	4.672×10^{-3} $\circ\circ$	1.421×10^{-2} $\circ$
RRAT(2), $f_c(f_{p_1}, f_{p_2}) = c_0 + c_1 f_{p_1} + c_2 f_{p_2}$			
	$p_1 = .2$, $p_2 = .5$	$p_1 = .2$, $p_2 = .8$	$p_1 = .5$, $p_2 = .8$
mean	5.269×10^2	5.270×10^2	5.267×10^2
standard error	1.617×10^2	1.616×10^2	1.616×10^2
p-value	3.455×10^{-3} $\circ\circ$	1.421×10^{-2} $\circ$	6.258×10^{-3} $\circ\circ$
RRAT(3), $f_c(f_{p_1}, f_{p_2}, f_{p_3}) = c_0 + c_1 f_{p_1} + c_2 f_{p_2} + c_3 f_{p_3}$			
	$p_1 = .2$, $p_2 = .5$, $p_3 = .8$		
mean	5.267×10^2		
standard error	1.616×10^2		
p-value	8.303×10^{-3} $\circ\circ$		
Best model on validation			
mean	5.267×10^2		
standard error	1.616×10^2		
p-value	4.672×10^{-3} $\circ\circ$		

Figure 6: Insurance experiment comparison of RRAT and LS regression on NN predictors: The mean and its standard error of the average squared error in each method as well as the p-values from a Wilcoxon signed rank test are indicated, where '$*$' ('$**$') denotes LS regression being significantly better than RRAT at 0.05 (0.01) level, '$-$' denotes no significant difference between them and '$\circ$' ('$\circ\circ$') denotes RRAT being significantly better than LS regression at 0.05 (0.01) level. **Note that RRAT(n), $n \geq 2$, always beat LS regression**.