Robust statistics for rejection-sampling timings

Daniel J. Bernstein[1,2]

[1] Department of Computer Science, University of Illinois at Chicago, USA
[2] Institute of Information Science, Academia Sinica, Taiwan
djb@cr.yp.to

Abstract. This paper (1) gives an example of a cost-measurement task for which medians and quartiles are neither robust nor stable; (2) suggests using $[1/8, 3/8]$, $[3/8, 5/8]$, $[5/8, 7/8]$ means as simple, stable, robust replacements for quartiles; and (3) tries this replacement on the example.

Keywords: software benchmarking, non-normal distributions

1 Prelude: the standard praise for medians

1981 Huber [18, page 107] writes in his textbook "Robust statistics" that "the so-called median absolute deviation (MAD) has emerged as the single most useful ancillary estimate of scale". The statistician is given many observations $x_1, \ldots, x_n$ from some distribution; uses the median M of $x_1, \ldots, x_n$ to estimate the location of the distribution; and uses the median of $|x_1 - M|, \ldots, |x_n - M|$ to estimate the scale (dispersion) of the distribution. "Ancillary" refers to the common situation that dispersion is "a nuisance parameter in location". The top goal is to estimate the location, but a wide distribution makes this unreliable; one estimates dispersion as an indication of the level of unreliability.

Why estimate location and dispersion using median and median absolute deviation instead of mean and standard deviation? The basic argument for medians from [18, Section 4.2] is that "the median achieves the smallest maximum bias among all translation invariant functionals". Here "bias" refers to the effect of starting from a normal distribution but corrupting a small fraction ϵ of the data. Obviously this contamination can create an arbitrarily large change in the mean no matter how small ϵ is, while the median turns out to minimize the worst-case effect of this contamination. For other robustness metrics favoring the median, see, e.g., 2025 Loh [22, Section 2] ("the robust sample median ... can be shown to have the maximal breakdown point among all translation-invariant location estimators"; "the median was shown to be the most B-robust estimator").

Permanent ID of this document: df535ec41e9f70f55c3c1bbdf81d54bd1a6d69ee. Date: 2025-07-27. This work was funded by the Taiwan's Executive Yuan Data Safety and Talent Cultivation Project (AS-KPQ-109-DSTCP). "Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s)."

2 Median coin-flip timings

Let's see how well these supposedly robust statistics do for a simple example: the distribution of the number of times you need to flip a fair coin before it comes up heads. The population distribution is a textbook geometric distribution: 1 with probability 1/2; 2 with probability 1/4; 3 with probability 1/8; etc.

The median of the population distribution is 1.5. The median of a large sample is unstable: it might occasionally match 1.5 (the probability of this is $\Theta(1/\sqrt{n})$ if there are n observations and n is even), but it has $1/2 - O(1/\sqrt{n})$ chance of being 1 and $1/2 - O(1/\sqrt{n})$ chance of being 2.

For essentially the same reason, the median of this distribution is fragile under arbitrarily small amounts of contamination: slight corruption favoring low values will push the median from 1.5 down to 1, while slight corruption favoring high values will push the median from 1.5 up to 2. In other words, the "influence function" from 1968 Hampel [**17**] is unbounded. The median absolute deviation similarly lacks stability and robustness.[3]

This is a straightforward argument against using medians and median absolute deviations whenever a quantile function might suddenly jump—being visibly discrete, for example. This doesn't contradict the calculations from [**18**, Section 4.2]: those calculations consider only a limited class of starting distributions. Also, this argument for avoiding the median is not an argument for going back to the dark ages and using the mean:[4] one wants a statistic that is meaningful for discrete distributions *and* is robust against contamination.

3 Median software timings

For many years I have been co-managing eBACS, a project that collects benchmarks of cryptographic software on many computers; see Section 7 for more information. Software timings are often contaminated (for a variety of reasons: e.g., the computer is distracted by an incoming network packet), sometimes spoiling the mean and variance of many timings; so, from the outset, eBACS instead reported medians of many timings, along with 1st and 3rd quartiles. This is a simple success story for robust statistics, right?

No, it's not that simple. More and more of the software that has been added to eBACS over the years relies on rejection-sampling loops. The rejection probabilities vary but sometimes, like the coin-flip example, create large jumps in timings close to the 25th, 50th, or 75th percentile. Then the quartiles aren't even stable, never mind robust.

As a concrete example, consider the sample quantile function displayed in Figure 3.1. This graph shows 93 measurements,[5] sorted into increasing order,

[3] 2022 Akinshin [**2**, Section 2.1] gives a more complicated example of the instability of the median absolute deviation—but still claims that the median absolute deviation is a "robust measure of statistical dispersion".

[4] Regressing to the mean, one might say.

[5] The measurement program has a 32-iteration loop of checking the clock and calling the function being timed, producing 31 timings. The program was run 3 times.

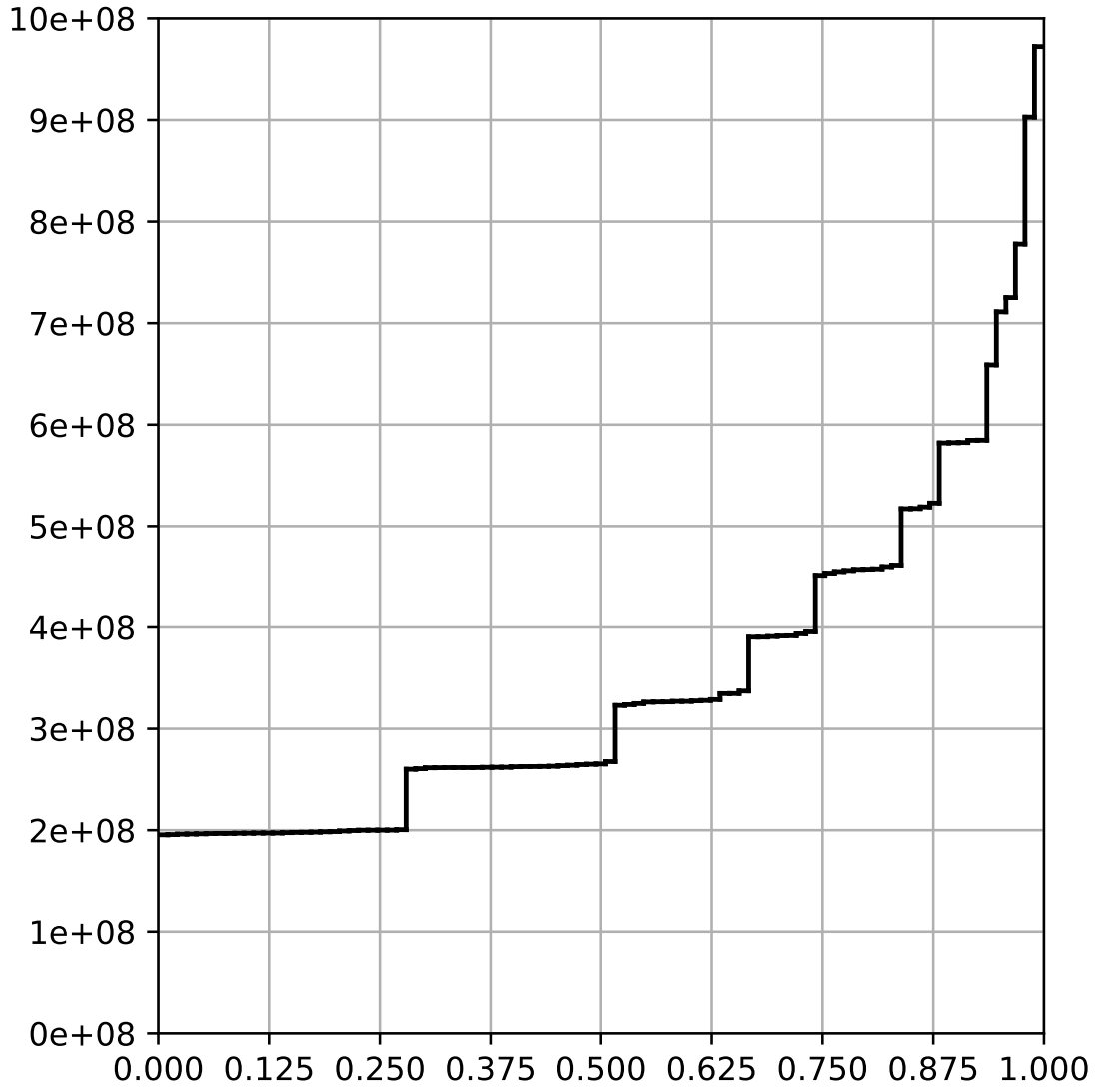

Fig. 3.1. Quantile function (transposed cdf) for 93 observations of cycle counts for `mceliece6960119` key generation on one core of a 3GHz Intel Xeon E3-1220 v5 with overclocking disabled.

of the time to generate keys for a cryptosystem called `mceliece6960119`. This cryptosystem, defined in [10] based on work going back to 1978 McEliece [23], has been deployed and is currently under consideration for standardization by ISO; see generally [5].

The software actually relies on multiple rejection-sampling loops, but the most important rejection-sampling loop tries to invert a random-looking $d \times d$ matrix with entries in the field of 2 elements, specifically with $d = 1547$. The probability p_d of invertibility for a $d \times d$ matrix generated uniformly at random

is $\prod_{1\le i\le d}(1-1/2^i)$ by, e.g., [12, page 77, Theorem 99], and one has

$$\lim_{d\to\infty} p_d = \prod_{i\ge 1}(1-1/2^i) = \sum_{k\in\mathbb{Z}}(-1)^k 2^{-k(3k+1)/2} \approx 0.288788$$

by Euler's [13] pentagonal-number theorem; p_d is only negligibly different from the limit for $d \ge 20$. The population distribution thus jumps at $p_d \approx 0.288788$, at $1-(1-p_d)^2 \approx 0.494178$, at $1-(1-p_d)^3 \approx 0.640253$, at $1-(1-p_d)^4 \approx 0.744144$, etc. Sampling randomly nudges the jumps, so there is nothing surprising about the second jump in Figure 3.1 being after 0.5 instead of before 0.5, producing a sample median around $2.7 \cdot 10^8$ where the population median is around $3.2 \cdot 10^8$. (The population mean is around $3.5 \cdot 10^8$.)

2008 Larocque–Randles [21], as part of justifying a simplified definition of the population median, comment that "it is a rare discrete population" for which the image of the cdf includes exactly 0.5. Formally, the rejection-sampling loop described above is not an exception: 0.494178 is not 0.5. But 0.494178 is close enough to 0.5 to create instability of the sample median until the sample size is very large. Even with 10000 observations, there will be variations on the scale of 0.01 starting from 0.494178, so both $2.7 \cdot 10^8$ and $3.2 \cdot 10^8$ have a good chance of appearing as the sample median. Similar comments apply to other quartiles, the median absolute deviation, etc.

4 Solutions in the literature

The literature on robust statistics considers many location statistics other than the median. For example, 1960 Tukey [28, Section 17] writes the following: "In large samples the sample mean is not nearly so safe an indicator of location as is the mean of the observations which remain after a small percentage of the highest, and an equal percentage of the lowest, have been set aside (use of a lightly truncated mean)." An example is the interquartile mean, although Tukey's examples of "light" truncation are at most 6% rather than 25%.

1920 Daniell [11] already considered "discard averages", as noted by, e.g., 2010 Stigler [27].[6] Today "trimmed means" are well known, even if not as well known as medians.[7] An interquartile mean is simple, easy to compute, and not

[6] Daniell in turn cites 1912 Poincaré [25, page 211] for discarding outliers ("rejeter une observation qui présente avec toutes les autres une divergence exagérée"). Poincaré [25, pages 212ff] considers different models for a contaminated distribution and obtains different rules for which outliers to discard; Daniell starts with more general approaches such as "quartile-discard averages" (interquartile means), and analyzes how well those approaches work for various distributions.

[7] I tried Google Scholar searches on 19 July 2025 for articles since 2024, putting search phrases into quotes. Google Scholar reported ("about") 11600 results for "trimmed mean"; 477 results for "truncated mean"; 319 results for "interquartile mean"; and 63500 results for "median", although a skim rapidly found that some of these were for, e.g., "median sternotomy".

afraid of discrete distributions. For the coin-flip example from Section 2, the interquartile mean of a large sample is consistently close to 1.5.

There are other choices in the literature, including choices that have advantages over trimmed means. For example, 1974 Stigler [**26**] shows that samples of a smoothed trimmed mean such as $16 \int_{0.25}^{0.75} (0.25 - |x - 0.5|) Q(x)\, dx$ are asymptotically normal under weak assumptions on the quantile function Q, while a similar result for samples of the interquartile mean $2 \int_{0.25}^{0.75} Q(x)\, dx$ requires more stringent assumptions about the behavior of Q at 0.25 and at 0.75.

On the other hand, trimmed means are easier to explain. Explainability is an important feature for someone like me choosing statistics to use in an application, whereas it's not clear why this application needs Stigler's asymptotic normality result—yes, Q in Figure 3.1 jumps close to 0.25 and close to 0.75; one can't expect the distribution of the sample interquartile mean to be close to normal; so what? Why should normality be a goal when it does not arise naturally? None of the other robust-even-for-the-discrete-case location statistics seem to be as popular as simple trimmed means.

What about dispersion statistics? Again the literature provides various options—for example, Tukey writes that "the use of truncated variances is likely to be quite satisfactory"—but these seem far less widely used than (untrimmed) standard deviations, quartiles, and the interquartile range,[8] all of which are unsatisfactory.

In short, the literature has solutions to the problem at hand, but it seems that the solutions are *popular* only for location, not for dispersion. I see this as another indication of the importance of explainability. I'm not satisfied saying "you know about mean and standard deviation already; I'll use trimmed mean and trimmed standard deviation". Even without trimming, standard deviation is more complicated than mean: consider the squares, the square roots, the common variants that differ noticeably from each other when the sample size is small, the tricky visual interpretation. This is all to make some formulas work out nicely, but trimming breaks those formulas. What do I say to a reader who complains that a trimmed standard deviation is neither comprehensible nor standard?

Furthermore, I want to have three statistics rather than two, to see not just location and dispersion but also asymmetry—the skewness of Figure 3.1, for example. Can I really explain replacing the 1st quartile with, say, the interquartile mean of the variable number of observations below the interquartile mean of the sample? I'd rather have something simpler.

[8] More Google Scholar searches on 19 July 2025 for articles since 2024: 49800 results for "standard deviation"; 22600 results for "interquartile range"; 18900 results for "quartile"; 12 results for "truncated standard deviation"; 45 results for "trimmed standard deviation"; 73 results for "truncated variance"; 15 results for "trimmed variance". Note that "interquartile range" typically refers to the difference between the 3rd quartile and the 1st quartile, rather than the two statistics separately—this is, as 2024 Haanappel–Voor in 't holt [**16**] note, bad terminology that should be fixed—but, either way, just a few quantiles are being used.

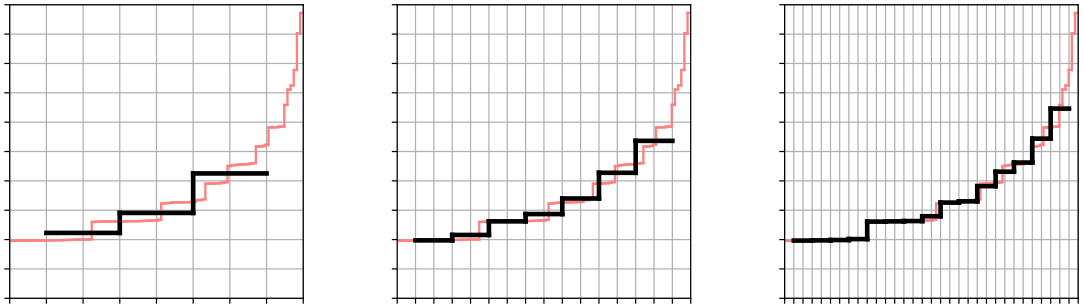

Fig. 5.1. First graph, in black: averages on $[1/8, 3/8]$ and $[3/8, 5/8]$ and $[5/8, 7/8]$ of the quantile function from Figure 3.1. Second graph, in black: averages on $[1/16, 3/16]$ etc. Third graph, in black: averages on $[1/32, 3/32]$ etc. Original quantile function is shown in light red.

5 Stabilized quartiles

The first black graph in Figure 5.1 is a graph of the following three numbers derived from the sample in Figure 3.1:

- "$[1/8, 3/8]$ mean" or "StQ_1": the mean between the 1st and 3rd octiles, as a stabilized substitute for the 1st quartile.
- "$[3/8, 5/8]$ mean" or "StQ_2": the mean between the 3rd and 5th octiles, as a stabilized substitute for the 2nd quartile.
- "$[5/8, 7/8]$ mean" or "StQ_3": the mean between the 5th and 7th octiles, as a stabilized substitute for the 3rd quartile.

Figure 5.2 shows how easy it is to compute these three statistics in Python. One can use these statistics in the same way as quartiles: computing the difference of StQ_3 and StQ_1 as an overall estimate of dispersion, computing appropriate ratios as an estimate of asymmetry, etc. For example, where [29] uses Bowley's skewness coefficient $(\mathrm{Q}_3 + \mathrm{Q}_1 - 2 \cdot \mathrm{Q}_2)/(\mathrm{Q}_3 - \mathrm{Q}_1)$, one can instead use $(\mathrm{StQ}_3 + \mathrm{StQ}_1 - 2 \cdot \mathrm{StQ}_2)/(\mathrm{StQ}_3 - \mathrm{StQ}_1)$.

For a normal distribution D, these stabilized quartiles are numerically close to quartiles: an integration exercise concludes that $\mathrm{StQ}_3(D) - \mathrm{StQ}_1(D)$ is

$$2^{5/2}\pi^{-1/2}\left(\exp\left(-\operatorname{erf}^{-1}(1/4)^2\right) - \exp\left(-\operatorname{erf}^{-1}(3/4)^2\right)\right) \approx 1.3867336971$$

times the standard deviation, while $\mathrm{Q}_3(D) - \mathrm{Q}_1(D)$ is $2^{3/2}\operatorname{erf}^{-1}(0.5) \approx 1.3489795004$ times the standard deviation.

These stabilized quartiles are robust against contamination. Their breakdown points (1/8 for the 1st and 3rd, 3/8 for the 2nd) are not as high as for quartiles (1/4, 1/2) or median absolute deviation (1/2); but, again, influence functions and sampling show that quartiles and median absolute deviation are fragile and unstable starting from the distribution in Section 2.

Asymmetrically trimmed means aren't new (for example, 1998 Kearns [20] found them useful as predictors of inflation), but I haven't found literature

```
def mean(S):
  S = list(S)
  return sum(S)/len(S)

def stq(S):
  S = sorted(8*list(S))
  n = len(S)//8
  return mean(S[n:3*n]),mean(S[3*n:5*n]),mean(S[5*n:7*n])
```

Fig. 5.2. Python 3 function `stq` to compute the three stabilized quartiles of a sample. In applications where the sample already has length divisible by 8, one can skip the initial multiplication by 8; one can also, by adding code to handle edges, avoid the initial multiplication in all cases.

proposing these statistics as an easy improvement over quartiles. I think the description above in terms of octiles is simpler than a description as interquartile means of (1) the observations below the median, (2) the interquartile observations, and (3) the observations above the median.

Some further options. The second and third graphs in Figure 5.1 are similar to the first but show stabilized octiles and stabilized hexadeciles.[9] The narrower spacing makes each statistic less stable, and (because the spacing comes closer to the edges) makes the outer statistics more vulnerable to contamination; but using more statistics provides more information, coming closer and closer to showing the full sample distribution, which in turn says something useful about the population distribution if there are enough samples.

One can instead work directly with the full sample distribution. I normally graph any distribution that I want to study; sometimes I compare it directly to a model distribution. But I also want to compress this information to a few statistics. For numerical tables summarizing benchmark results, I continue to think that 3 statistics are the right level of detail, so I'm switching to stabilized quartiles, as in the first graph in Figure 5.1. See Section 6 for an example of the stability of stabilized quartiles, and Section 7 for the impact of stabilized quartiles on eBACS.

A histogram also compresses a sample distribution to fewer numbers, but a histogram starts by choosing a spacing of *values*. The numbers here, like conventional quartiles, start by choosing a spacing of *probabilities*. A separate issue is that a histogram traditionally displays density, which is less stable than distribution. Differences of quartiles (and ratios of those differences) are similarly less stable than the quartiles per se.

[9] Note that stabilization pays attention to denominators: the jth stabilized d-ile is the $[(2j-1)/2d,(2j+1)/2d]$ mean, so the 4th stabilized octile is the $[7/16,9/16]$ mean, the 2nd stabilized quartile is the $[3/8,5/8]$ mean, and the stabilized median is the $[1/4,3/4]$ mean, i.e., the usual interquartile mean.

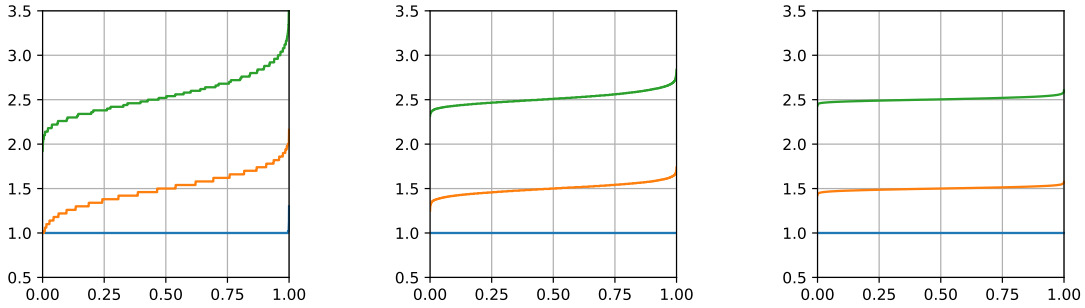

Fig. 6.1. Distribution of StQ_1 (bottom, blue), StQ_2 (middle, orange), and StQ_3 (top, green) for n observations of the number of coin flips to obtain heads, where $n = 100$ (left), $n = 1000$ (middle), or $n = 10000$ (right). Each distribution is computed from a size-10000 sample of size-n samples, and is displayed as a quantile function.

If stability were the only goal then by default I would take every opportunity to add another layer of integration: not just switching from trimmed means to Stigler's smoothed trimmed means but also switching the smoothing function from a triangle to something infinitely differentiable, such as the function $p \mapsto \exp(1/(16(p-1/2)^2-1))$ on $[1/4, 3/4]$. But this doesn't score well on simplicity.

6 Stability evaluation

Let's evaluate the stability of StQ_1, StQ_2, and StQ_3 on a specific distribution D for which medians are unstable. For simplicity and reproducibility, let's take D as the coin-flip distribution from Section 2. The stabilized quartiles of D are 1, 1.5, and 2.5. The hope is that the stabilized quartiles of a sample from D will be not just stable but unbiased, close to the stabilized quartiles of D; but one also expects deviations on the scale of $1/\sqrt{n}$ for size-n samples.

The three graphs in Figure 6.1 are produced by a simple Python script available from [**6**] as a supplement to this paper. The script calculates stabilized quartiles for a size-10000 sample of size-100 samples from D, and plots the resulting observed distributions of StQ_1, StQ_2, and StQ_3, obtaining the three curves in the left graph in Figure 6.1. It then does the same for a size-10000 sample of size-1000 samples from D, and for a size-10000 sample of size-10000 samples from D, obtaining the other two graphs.

One sees from the $n = 100$ graph that StQ_1 is occasionally above 1 (the right side of the curve shows occasional samples where $\geq$2 appeared before the 3rd octile), but is almost always 1. StQ_2 is occasionally 1, and occasionally 2 or higher, but between 1.4 and 1.6 about half the time. StQ_3 has a broader range and an evident asymmetry (the right side is pushed up by the frequent cases where $\geq$4 appears before the 7th octile) but is between 2.4 and 2.7 about half the time.

The $n = 1000$ and $n = 10000$ graphs in Figure 6.1 show that, unsurprisingly, the observed distribution of StQ_i for size-n samples from D tightens around

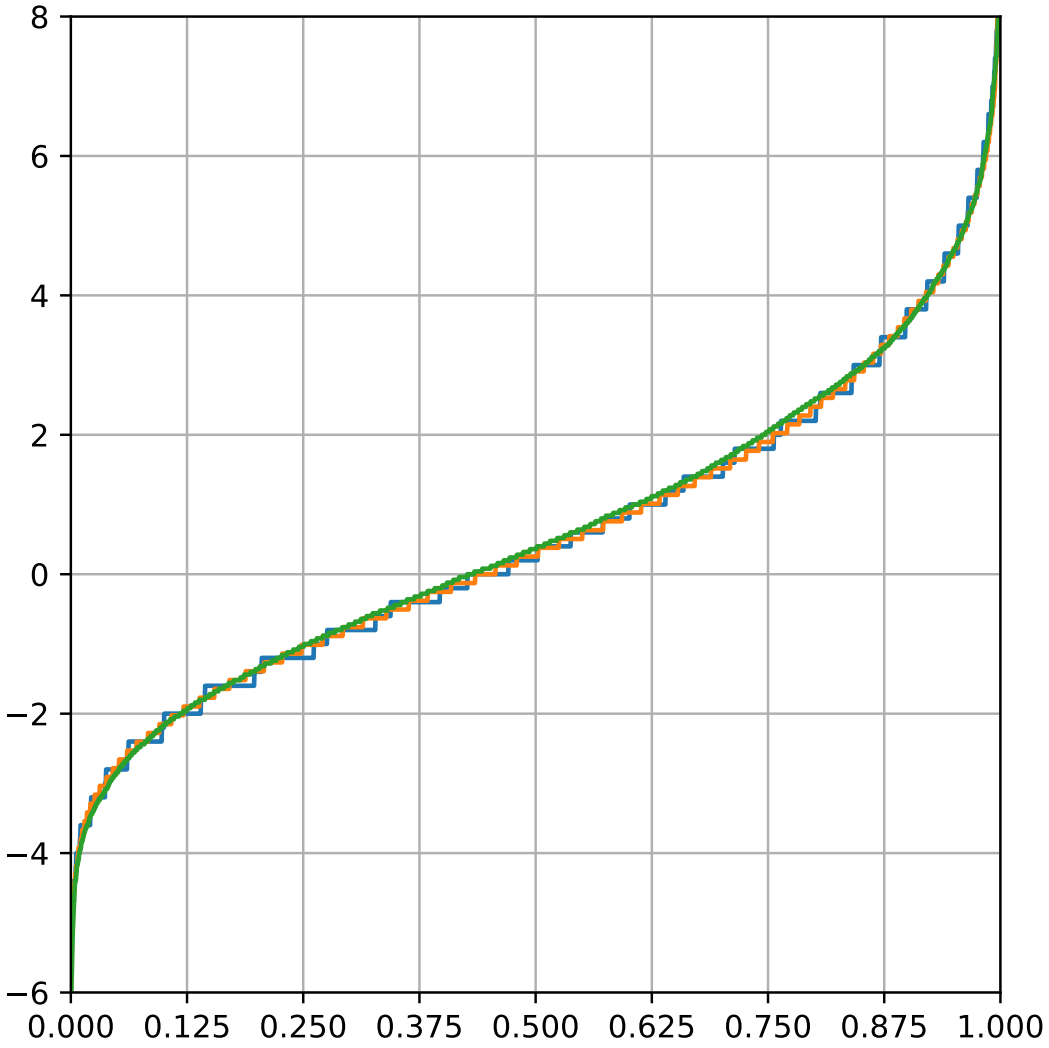

Fig. 6.2. Distribution of $\sqrt{n}(\mathrm{StQ}_3 - 2.5)$ for n observations of the number of coin flips to obtain heads, where $n = 100$ (blue), $n = 1000$ (orange), $n = 10000$ (green). Each distribution is computed from a size-10000 sample of size-n samples, and is displayed as a quantile function.

$\mathrm{StQ}_i(D)$ as n increases. As a closer look at this tightening—with the caveat that taking a size-10000 sample of statistics for size-n samples can still produce visible deviations from the population distribution of statistics for size-n samples—Figure 6.2 plots the observed distributions of $\sqrt{n}(\mathrm{StQ}_3 - 2.5)$ for $n \in \{100, 1000, 10000\}$. These distributions are very close to each other. Given that the jump in D at the 7th octile is at the edge of StQ_3, and given the aforementioned asymptotic results from [**26**], it is also unsurprising that the StQ_3 asymmetry visible in Figure 6.1, with the right tail thicker than the left tail, persists in Figure 6.2 as n increases.

7 Impact on a large collection of benchmarks

Finally, let's look at what happens when quartiles are replaced with stabilized quartiles in cryptographic benchmarks.

Status quo, part 1: context. Speed evaluations play a major role in cryptography; see [4] for examples and references. Often people carry out their own ad-hoc speed evaluations, but hundreds of people have contributed thousands of cryptographic implementations to eBACS for central evaluation reported by [9] with continual updates. Google Scholar says there are 398 citations of eBACS, and a Bing search for `bench.cr.yp.to` in quotes currently says "About 61,700 results".

Status quo, part 2: the scale of eBACS. The latest release [8] of the eBACS benchmarking toolkit is 45MB and contains 4601 implementations of 1430 cryptographic "primitives" in hundreds of different families.[10] The eBACS web pages currently show results from this toolkit on 44 computers, plus results from some stragglers using older versions of the toolkit.

For concreteness, I'll focus here on a computer named `samba`. The dataset collected by `samba` (155MB compressed, 1.5GB uncompressed) is available from [1], using the format described in [9]. This dataset contains samples from 2499504 time distributions, along with various other test results that are not relevant to this paper.

There are two basic reasons that the number of time distributions exceeds the number of primitives. First, a single primitive usually provides multiple functions: e.g., one function to sign a message and another to verify a signature. Second, a function is often measured for many different input sizes: e.g., different lengths of messages to be signed.

The online comparison table for, e.g., signing a 59-byte message shows 183 signature primitives sorted by 2nd quartiles. Equal table space is given to the other quartiles: the objective is for large dispersion—discounting contamination—to be visible as a large interquartile distance, and for asymmetry to also be visible. If $Q_1 < 0.8\,Q_3$ then all three quartiles are marked in red with question marks in the table.

Status quo, part 3: sample sizes. There is pressure on eBACS to keep sample sizes small so as to keep total benchmarking time under control. On the other hand, reliably locating the center of a broad distribution requires large sample sizes.

One reason for variations in software timings is rejection sampling, as in Figure 3.1. Another reason mentioned in Section 3 is contamination from the computer taking time for unrelated activities.

[10] One family typically includes multiple primitives at different sizes, where the larger sizes are slower but hopefully more resistant to attack. Each size is benchmarked separately.

One can see more reasons from, e.g., [14]. For example, code that has not been run yet has to be loaded into the CPU's cache, producing a large slowdown when a program runs a large function for the first time. One can thus think of timings as a mix of two distributions, namely uncached timings and cached timings. Uncached timings are sometimes of interest, but robustly measuring them—while feasible for individual functions, as [3, Appendix A] illustrates—would be costly for the number of functions handled in eBACS. The priority in eBACS is to measure cached timings, and then uncached timings are just another form of contamination.

Code also slows down whenever the CPU mispredicts a branch taken by the code. Prediction, in turn, depends on recent history and, more subtly, on virtual-memory mappings that are chosen randomly by the operating system whenever a program is run. Experience indicates that this variation is usually—but not always—small. To have a *chance* of detecting mapping-dependent issues, the eBACS toolkit runs each measurement program 3 times and collects the observations across the runs into a single sample.

For functions whose performance is expected to be stable, the benchmarking toolkit chooses sample size just 21, coming from 7 measurements per run times 3 runs. Notice that cache-related slowdowns in the first measurement in a run will then increase 1/7 of the sample; this is below the breakdown point for Q_3 (and for StQ_2), but it has a $4(1/7 - 1/8) = 1/14$ effect on StQ_3.

The toolkit uses larger sample sizes for some functions, as illustrated by the 93 timings used for Figure 3.1 (which is from `samba`). The average sample size in the `samba` data set is slightly over 31, for 78400674 cycle counts overall.

Trying stabilized quartiles. Figure 7.1 plots the StQ_i/Q_i ratios against the ratio StQ_1/StQ_3. These plots are produced by another supplement to this paper, namely [7], again a simple Python script. This script takes the uncompressed `samba` dataset as input. It restricts to $StQ_1/StQ_3 \leq 0.95$ to keep plotting time under control, and restricts to $StQ_1/StQ_3 \geq 0.75$ for visibility of the remaining data, so there are only 56561 dots in each graph. Separately from the graphs, the script outputs data about all 2499504 samples for further analysis.

The blue fin-like shapes around horizontal position 0.92 in the middle graph show some cases where StQ_2 moves 4% away from Q_2 while StQ_3 and StQ_1 are separated by 8%. Manual investigation of the blue corners found measurements of `siphash24` (for some input sizes) concentrated on two modes. If the modes are roughly balanced then StQ_1 is stably the smaller mode, StQ_3 is stably the larger mode, StQ_2 is stably in between, and Q_2 unstably picks one mode or the other.

Larger gaps between StQ_2 and Q_2 appear as StQ_1/StQ_3 drops. Beyond the graph, the minimum StQ_2/Q_2 ratio is 0.78546 and the maximum is 1.39989; there are 2601 samples below 0.99 and 3301 samples above 1.01.

The smallest and largest StQ_2/Q_2 ratios appear to be explained by rejection sampling. Specifically, graphs for various samples for `3icp` signing, `dilithium*` signing, `falcon512tree` keygen, `*gemss*` signing, `haetae*` keygen and signing, `lattisigns512` signing, `leda*` keygen, `mceliece*` keygen (as in Figure 3.1) and

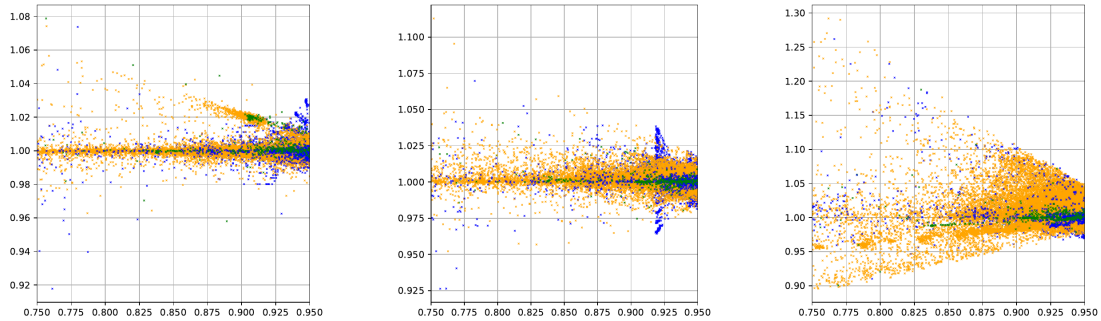

Fig. 7.1. Scatterplot of $\mathrm{StQ}_1/\mathrm{Q}_1$ (left), $\mathrm{StQ}_2/\mathrm{Q}_2$ (middle), and $\mathrm{StQ}_3/\mathrm{Q}_3$ (right) against $\mathrm{StQ}_1/\mathrm{StQ}_3$ (horizontal axis) for the 56561 samples in the `samba` data set having $0.75 \le \mathrm{StQ}_1/\mathrm{StQ}_3 \le 0.95$. Vertical range is chosen independently by `matplotlib` for each graph and is not restricted. Sample sizes are 21 (orange), 45 (blue), or 90 or larger (green).

enc, `nccsign*` signing, `ntruplus*` keygen, `pass*` signing, `qtesla*` keygen and signing, `ronald*` keygen, `rsa*` keygen, `rwb0fuz1024` keygen, and `sikep*comp` keygen suggest that all of these use rejection sampling; these account for 1804 of the 2601 + 3301 samples. Removing those cases leaves 4098 ratios $\mathrm{StQ}_2/\mathrm{Q}_2$ outside $[0.99, 1.01]$, with minimum 0.95214 and maximum 1.19968.

Only 48 of those 4098 ratios are above 1.05, with 39 (including the top 12) from a single cryptographic primitive, `aeadaes192ocbtaglen128v1`. These all use sample size 21, and show a consistent pattern of the first, second, and seventh timings in each run of 7 being higher than the others. Having 3/7 of the sample larger than the rest easily explains StQ_2 being above Q_2.

Another interesting example above 1.05 is `skinnyaeadtk3128128plusv1` specifically for encrypting 1-byte messages. The cycle counts in this sample are

$$13538, 6940, 6861, 6804, 6828, 6804, 6845,$$
$$13890, 4337, 4316, 4304, 4210, 4187, 4197,$$
$$11515, 4426, 4293, 4289, 4259, 4255, 4300$$

where the larger first column is from (common) cache effects and the larger first row is from (rare) mapping-dependent effects.

None of the investigated cases show Q_2 being more meaningful than StQ_2, while the rejection-sampling cases show an instability of Q_2 corrected by StQ_2.

As for the other quartiles: The left graph in Figure 7.1 shows $\mathrm{StQ}_1/\mathrm{Q}_1$ tending to be closer to 1 than $\mathrm{StQ}_2/\mathrm{Q}_2$ is, but sometimes being pushed upwards by what look like a few different effects. The blue spikes are again explained by bimodal measurements for `siphash24`, this time with the jump being around 25% rather than 50%.

Beyond the graph, $\mathrm{StQ}_1/\mathrm{Q}_1$ has minimum 0.79105 and maximum 1.48806. The extremes are mostly explained by rejection sampling, examples of StQ_1 being more meaningful than Q_1. There were two `edonk*` dec samples showing trimodal mapping-dependent effects.

Finally, the right graph in Figure 7.1 shows more variability in StQ_3/Q_3, especially upwards; note that an increase in StQ_3 also reduces StQ_1/StQ_3. Beyond the graph, StQ_1/Q_1 has minimum 0.77764 and maximum 1.53344. Manual investigation again found a mixture of the effects described above, including many examples of Q_3 being obviously unstable. This does not mean that StQ_3 is perfectly stable, especially for small sample sizes.

References

[1] — (no editor), "`supercop-20250415` results from `samba`" (2025). URL: https://zenodo.org/records/16281645. Cited in §7.

[2] Andrey Akinshin, "Quantile absolute deviation" (2022). URL: https://arxiv.org/abs/2208.13459. Cited in §2.

[3] Daniel J. Bernstein, "The Poly1305-AES message-authentication code", in Gilbert and Handschuh [15] (2005), 32–49. URL: https://cr.yp.to/papers.html#poly1305. Cited in §7.

[4] Daniel J. Bernstein, "Cryptographic competitions", Journal of Cryptology **37** (2024), article 7. URL: https://cr.yp.to/papers.html#competitions. Cited in §7.

[5] Daniel J. Bernstein, "McEliece standardization" (2025). URL: https://blog.cr.yp.to/20250423-mceliece.html. Cited in §3.

[6] Daniel J. Bernstein, "`coin.py`" (2025). URL: https://cr.yp.to/2025/20250722-coin.py. Cited in §6.

[7] Daniel J. Bernstein, "`stqebacs.py`" (2025). URL: https://cr.yp.to/2025/20250722-stqebacs.py. Cited in §7.

[8] Daniel J. Bernstein and Tanja Lange (editors), "`supercop-20250415`" (2025). URL: https://bench.cr.yp.to/supercop/supercop-20250415.tar.xz. Cited in §7.

[9] Daniel J. Bernstein and Tanja Lange (editors), "eBACS: ECRYPT Benchmarking of Cryptographic Systems" (2025), accessed 19 July 2025. URL: https://bench.cr.yp.to. Cited in §7, §7.

[10] Classic McEliece Team, "Classic McEliece: conservative code-based cryptography: cryptosystem specification" (2022). URL: https://classic.mceliece.org/mceliece-spec-20221023.pdf. Cited in §3.

[11] Percy John Daniell, "Observations weighted according to order", American Journal of Mathematics **42** (1920), 222–236. URL: https://www.jstor.org/stable/pdf/2370465.pdf. Cited in §4.

[12] Leonard Eugene Dickson, "Linear groups with an exposition of the Galois field theory", Teubner, 1901. URL: https://archive.org/details/lineargroupswith00ledi/. Cited in §3.

[13] Leonhard Euler, "De mirabilis proprietatibus numerorum pentagonalium", Acta Academiae Scientiarum Imperialis Petropolitanae **1780:I** (1783), 56–75. URL: https://scholarlycommons.pacific.edu/euler-works/542/. Cited in §3.

[14] Agner Fog, "Instruction tables: Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel, AMD and VIA CPUs" (2024). URL: https://agner.org/optimize/. Cited in §7.

[15] Henri Gilbert and Helena Handschuh (editors), "Fast software encryption: 12th international workshop, FSE 2005, Paris, France, February 21–23, 2005, revised selected papers", Lecture Notes in Computer Science, 3557, Springer, 2005. ISBN 3-540-26541-4. See [3].

[16] Cynthia P. Haanappel and Anne F. Voor in 't holt, "Using the interquartile range in infection prevention and control research", Infection Prevention in Practice **6** (2024), 100337. URL: https://pure.eur.nl/en/publications/using-the-interquartile-range-in-infection-prevention-and-control. Cited in §4.

[17] Frank Rudolf Hampel, "Contributions to the theory of robust estimation", Ph.D. thesis, University of California, Berkeley, 1968. Cited in §2.

[18] Peter J. Huber, "Robust statistics", Wiley, 1981; see also newer version from Huber and Ronchetti [**19**]. ISBN 978-0-47141805-4. DOI: 10.1002/0471725250. Cited in §1, §1, §2.

[19] Peter J. Huber and Elvezio M. Ronchetti, "Robust statistics", 2nd edition, Wiley, 2009; see also older version from Huber [**18**]. ISBN 978-0-47012990-6. DOI: 10.1002/9780470434697.

[20] Jonathan Kearns, "The distribution and measurement of inflation" (1998). URL: https://www.rba.gov.au/publications/rdp/1998/pdf/rdp9810.pdf. Cited in §5.

[21] Denis Larocque and Ronald H. Randles, "Confidence intervals for a discrete population median", The American Statistician **62** (2008), 32–39. Cited in §3.

[22] Po-Ling Loh, "A theoretical review of modern robust statistics", Annual Review of Statistics and Its Application **12** (2025), 477–496. DOI: 10.1146/annurev-statistics-112723-034446. Cited in §1.

[23] Robert J. McEliece, "A public-key cryptosystem based on algebraic coding theory" (1978), 114–116, JPL DSN Progress Report. URL: https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF. Cited in §3.

[24] Ingram Olkin, Sudhist G. Ghurye, Wassily Hoeffding, William G. Madow, and Henry B. Mann (editors), "Contributions to probability and statistics: Essays in honor of Harold Hotelling", Stanford Studies in Mathematics and Statistics, 2, Oxford University Press, 1960. See [28].

[25] Henri Poincaré, "Calcul des probabilités", 2nd edition, Gauthier-Villars, 1912. URL: https://archive.org/details/calculdeprobabil00poinrich. Cited in §4, §4.

[26] Stephen M. Stigler, "Linear functions of order statistics with smooth weight functions", The Annals of Statistics **2** (1974), 676–693. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-2/issue-4/Linear-Functions-of-Order-Statistics-with-Smooth-Weight-Functions/10.1214/aos/1176342756.pdf. Cited in §4, §6.

[27] Stephen M. Stigler, "The changing history of robustness", The American Statistician **64:4** (2010), 277–281. DOI: 10.1198/tast.2010.10159. Cited in §4.

[28] John W. Tukey, "A survey of sampling from contaminated distributions", in Olkin et al. [**24**] (1960), 448–485. URL: https://jugander.github.io/rare/Tukey1960_ContaminatedDistributions.pdf. Cited in §4.

[29] Randal Verbrugge and Saeed Zaman, "Improving inflation forecasts using robust measures", International Journal of Forecasting **40** (2024), 735–745. URL: https://www.clevelandfed.org/-/media/project/clevelandfedtenant/clevelandfedsite/publications/working-papers/2023/wp2223r.pdf. DOI: 10.1016/j.ijforecast.2023.05.003. Cited in §5.