Statistics and Computing **13**: 91–100, 2003

Single-pass low-storage arbitrary quantile estimation for massive datasets

JOHN C. LIECHTY*, DENNIS K. J. LIN† and JAMES P. McDERMOTT**

**Department of Marketing, Pennsylvania State University, University Park, PA 16802, USA*
jcl12@psu.edu
†*Department of Supply Chain and Information Systems, University Park, PA 16802, USA*
dk15@psu.edu
***Department of Statistics, Pennsylvania State University, University Park, PA 16802, USA*
mcdermott@stat.psu.edu

Received April 2001 and accepted May 2002

We present a single-pass, low-storage, sequential method for estimating an arbitrary quantile of an unknown distribution. The proposed method performs very well when compared to existing methods for estimating the median as well as arbitrary quantiles for a wide range of densities. In addition to explaining the method and presenting the results of the simulation study, we discuss intuition behind the method and demonstrate empirically, for certain densities, that the proposed estimator converges to the sample quantile.

Keywords: low-storage, quantile estimation, single-pass algorithms, data mining, large datasets, tail quantile

1. Introduction

Quantile estimation for an unknown distribution is a commonly studied problem. Pfanzagl (1974) showed that when nothing is known about a distribution of interest, the sample quantile has the minimum asymptotic variance among translation invariant estimators of the population quantile. While it may be desirable, using the sample quantile as an estimate of the population quantile becomes cumbersome and in many cases impractical to obtain, both in terms of storage space and computation time, when the size of the data set becomes large. In this paper we introduce a single-pass, low-storage method of estimating an arbitrary quantile, based on a sequential scoring algorithm that combines estimated ranks and assigned weights, where the weights represent, in some sense, the information associated with each estimated rank.

Massive datasets are becoming more and more common in modern society. They arise from sources as diverse as large call centers, internet traffic data, sales transactional records, or satellite feeds. This phenomenon presents a clear need to be able to process the data accurately and efficiently so that current analyses may be performed before becoming inundated by a continually growing store of data.

Applications of the proposed method include, but are not limited to, query optimization for large databases and network routing problems. Manku, Rajagopalan and Lindsay (1998) note that it is common in the database field to keep summaries of the variables in the form of equi-depth histograms. However, creating and maintaining these histograms can be quite costly. Another possible application could be in the area of network routing. Network routing decisions, and hence quality of service for the network users (Kesidis 1999), could be improved by having more accurate summaries of the distributions of the historical traffic data, in particular of the tails of these distributions. As noted in Dunn (1991), a further application is in the computation through simulation of critical values and percentile points of new statistics whose distributions are unknown. A final application could be in the area of MCMC analysis where simulations routinely generate massive amounts of data.

In Section 2, we briefly review existing methods for estimating the median and arbitrary quantiles. We describe the proposed estimation method in Section 3. In Section 4, we demonstrate the performance of this method and highlight some of its properties by summarizing the results from a collection of simulation studies. Some theoretical considerations are discussed in Section 5 and we conclude with a discussion of the proposed method and future research.

2. Existing methods

We start our discussion by putting forward notation and definitions that will be used throughout the paper. Let $X_1, \ldots, X_n$ be a sample from a distribution F, where we assume F is continuous so that all observations are unique with probability 1. Let the order statistics $X_{(1)} < \cdots < X_{(n)}$ be the observations arranged in ascending order. The pth population quantile of a distribution F is defined as

$$\xi_p = F^{-1}(p) = \inf\{x : F(x) \geq p\},$$

and the pth sample quantile as $\hat{\xi}_p = X_{(k)}$, where $k = \lceil np \rceil$ is the smallest integer greater than or equal to np, for $0 < p < 1$. Hence a sample quantile can be attained by simply sorting the data and taking the appropriate order statistic. However, as the size of the dataset becomes large, computation and storage burdens make this method infeasible.

Hurley and Modarres (1995) offer a nice survey of current methods for estimating quantiles. Most of the methods reviewed in this survey focus on estimating the median of a distribution, and in practice only one method, the Stochastic Approximation (S.A.) method introduced in Tierney (1983), is easily extended to estimate an arbitrary quantile. In addition to reviewing current methods, Hurley and Modarres (1995) introduce a histogram based method for estimating quantiles. Their proposed method has many attractive qualities, in particular for estimation of the median. However, for estimation of quantiles other than the median their method has a non-zero probability of having to be repeated and hence may require more than one pass through the data set in order to obtain an appropriate estimate of the quantile. Extending their method so that it can be used to estimate extreme quantiles (quantiles with values of p close to 0 or 1), would result in an increased probability of requiring more than one pass through the data set, making it impractical for estimating extreme quantiles.

Pearl (1981) proposed using a minimax tree to estimate an arbitrary quantile. While this method is easy to implement and utilizes very little storage space, it has the drawback that it will only work for sample sizes that can be specified in terms of the three parameters which describe the tree. As a result this method cannot easily be extended to arbitrary quantiles without ignoring part of the data in order to get a desired sample size. Rousseeuw and Bassett (1990) proposed the remedian method of quantile estimation. As with the minimax tree method, there are restrictions on the size of data sets that can be analyzed using this method. The remedian method can be extended to other quantiles (Chao and Lin 1993), however this extension is not easily accomplished in practice.

Alternatively, the S.A. method proposed by Tierney (1983) is quite accurate, straightforward to implement for arbitrary sized datasets, and easily extensible to estimate arbitrary quantiles. The main drawbacks of the S.A. method are that its accuracy depends on an initial sample and it allows for estimates which are outside of the range of possible values. Because the accuracy depends on getting an initial sample that has a quantile that is close to the sample quantile of the entire data set, the S.A. performs poorly when estimating extreme quantiles. This is a weakness that can only be overcome by increasing the size of the initial sample which can lead to the same challenges associated with the sample quantile. With regards to the bounds of the S.A. estimator, if one were estimating a left tail quantile for data generated by a χ^2_1 distribution, there is nothing to prevent this estimator from returning a negative value for an estimate, since the method doesn't return an actual element of the data set.

There are several methods that use a variable amount of storage, for example Krutchkoff (1986) and Dunn (1991), where the authors employ probabilistic techniques based on the normal approximation to the binomial. While these methods have merit, we are interested in studying methods which can be implemented with a small fixed amount of storage. As a result we will not make direct comparisons to methods utilizing variable amounts of storage.

All of the fixed storage methods that we consider perform accurately and efficiently for median estimation. Some, such as the S.A. method and histogram based method, are well suited to handle datasets of any size whereas others are restricted to datasets with a fixed number of elements. In addition, Tierney's S.A. method can be easily extended to estimate arbitrary quantiles. While Tierney's method works well for quantiles at the center of the distribution, the variability of the estimator increases as the quantile moves toward the tails of the distribution. Our main contribution comes in presenting a new method whose performance is comparable to existing methods for quantiles in the center of a distribution and which performs appreciably better for quantiles in the tails of a distribution.

3. Proposed method

The proposed method keeps track of a small fraction of the original data set (typically from 40 to 100 data points are tracked by the algorithm for any one implementation) and uses a scoring rule based on an estimated rank and an assigned weight for each data point to determine which data points to track and which data points to ignore. The assigned weights can be viewed in two different ways. From one perspective, each weight can be seen as a measurement of the amount of information associated with a particular estimated rank. From another perspective, each weight can be viewed as an estimate of the standard deviation associated with a particular estimated rank. The second perspective provided the inspiration for the scoring rule used in the algorithm. After giving a brief overview of the algorithm, we give a detailed description of certain steps of the algorithm and give motivation for appropriate steps.

Overview of the quantile estimation algorithm

1. Sort the first m data points. Set the estimated rank, r_i, for each data point, x_i, equal to the actual rank of the initial sample

(i.e. $r_i = i$). Set the weight, w_i, for each data point to 1 (i.e. $w_i = 1$).

2. Determine the location of the next point in the data set, x_*, with respect to the m data points that are being tracked, and increment the ranks of the points that are greater than the new point, i.e. if $x_i > x_*$, then $r_i = r_i + 1$.
3. Calculate an estimated rank for the new point; see detailed discussion for appropriate formula.
4. Assign a weight to the new point; let

$$w_* = \min(r_{i+1} - r_*, r_* - r_i),$$

where $x_i < x_* < x_{i+1}$.

5. Assign a score to all of the points in the array and to the new point,

$$s_\cdot = \left| \frac{r_\cdot - target}{w_\cdot} \right|,$$

where $target = n'p$, n' is the total number of data points observed so far and p is the proportion associated with the pth population quantile.

6. If the maximum score of the points being tracked is larger than the score for the new point, remove the point with the largest score from the tracking list and insert the new point, along with its estimated rank and weight, into the tracking list.
7. Repeat Steps 2–6 until all elements of the data set have been seen.
8. The final estimate of the pth population quantile is the point in the final array with the estimated rank closest to the target rank.

The proposed method begins by sorting the first m points from the data set and assigning an *estimated rank* to each of these points. The estimated rank is a measure of where each point falls in relation to the other points previously observed. The *weight* associated with each observation is a measure of the amount of information associated with each estimated rank.

The estimated rank, which is assigned in Step 3, depends on the location of the new point with respect to the points that are being tracked. Following are the formulae used to calculate the estimated rank for a new data point.

- If the new point is a new maximum, $x_* > x_m$, the new point becomes the new maximum and the old maximum becomes the new point. Let $r_* = r_m$ and then let $r_m = r_m + 1$.
- If the new point is a new minimum, $x_* < x_1$, the new point becomes the new minimum and the old minimum becomes the new point. Let $r_* = 2$ and then let $r_1 = 1$.
- If the new point is just less than the maximum, $x_{m-1} < x_* < x_m$, then

$$r_* = r_{m-1} + \frac{r_m - r_{m-1}}{1 - \delta}\left(1 - e^{-\lambda(x_* - x_{m-1})}\right), \qquad (1)$$

where

$$\delta = e^{-\lambda(x_m - x_{m-1})}, \quad \lambda = -\frac{\log(1 - q_2(1 - \delta))}{q_1(x_m - x_{m-1})},$$

and q_1 and q_2 are set by the researcher (as will be discussed below).

- If the new point is just greater than the minimum, $x_1 < x_* < x_2$, then

$$r_* = r_2 + \frac{r_1 - r_2}{1 - \delta}\left(1 - e^{-\lambda(x_2 - x_*)}\right), \qquad (2)$$

where

$$\delta = e^{-\lambda(x_2 - x_1)}, \quad \lambda = -\frac{\log(1 - q_2(1 - \delta))}{q_1(x_2 - x_1)},$$

and q_1 and q_2 are, again, set by the researcher.

- If the new point falls any where else, $x_2 < x_* < x_{m-1}$, then

$$r_* = r_i + (r_{i+1} - r_i)\frac{x_* - x_i}{x_{i+1} - x_i},$$

where $x_i < x_* < x_{i+1}$.

To illustrate these ideas, note in Fig. 1 that if a new point falls between the current minimum and the second smallest point in the tracking array or between the current maximum and the second largest point in the tracking array (i.e. between x_1 and x_2 or between x_{m-1} and x_m) then we use the non-linear interpolation method for obtaining the estimated rank of the new point. If it falls between two other points in the tracking array (i.e. between x_i and x_{i+1}, for $2 \leq i \leq m-2$) then we use a linear interpolation to estimate the rank. If the new point represents a new maximum or minimum, we simply switch the old maximum or minimum with the new point and assign estimated ranks accordingly. If the new point is between the minimum and the second smallest point in the list that is being tracked or between the maximum and the second largest point, we use a non-linear interpolation to estimate the rank. In all other cases we use a simple linear interpolation to estimate the rank of the new point. We found that as the algorithm progresses through a data set, the distance between the maximum or minimum element of the tracked list and the next point in the tracked list tends to become very large. As a result, the values associated with the points being tracked tend to contain reasonably good information about the rank for points close to the quantile that is being estimated, but these values offer very little information about the rank of points that are not close to this quantile. The non-linear functions, as described in (1) and (2), are exponential curves which are designed so that the estimated ranks quickly go to the rank associated with the maximum or minimum element as the new point moves towards either of these elements.

The concept of fitting an exponential curve to the tail of a sorted sample is not a new one. Breiman, Gins and Stone (1979) propose taking the maximum likelihood estimator of the tail of the sample and then fitting an exponential curve with this estimated parameter value to predict unobserved tail quantiles. Ott (1995) discusses various methods for fitting a two-parameter exponential curve to the tail of the data. In our approach, we use fixed parameters for our exponential curves and force them to go through two points: the minimum (or maximum) and the second smallest (or second largest) point in the tracking array. The shape of the exponential curves is determined by the parameters q_1 and

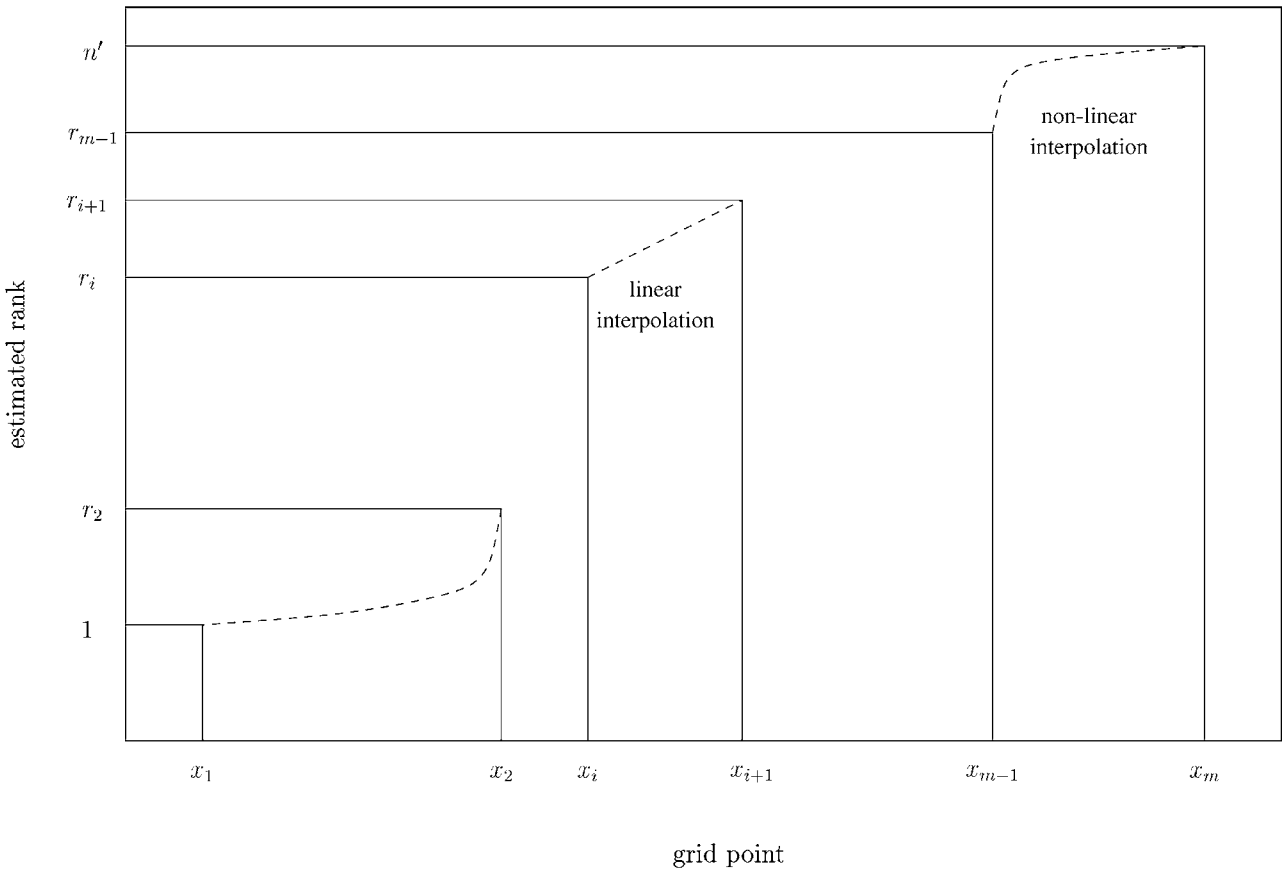

Fig. 1. *Estimation of ranks*

q_2. The rank of the new point is then interpolated from this curve. These non-linear functions serve as safeguards against heavy-tailed data. If there is no heavy tailed data, the curves have little or no effect.

The parameters q_1 and q_2 determine how quickly the estimated rank goes to the rank associated with the maximum or minimum. To illustrate, for the maximum, q_1 and q_2 are set so that

$$q_1 = \frac{x_* - x_{m-1}}{x_m - x_{m-1}}$$

and

$$q_2 = \frac{r_* - r_{m-1}}{r_m - r_{m-1}}.$$

In practice we set $q_1 = 0.1$ and $q_2 = 0.9$. The reasons for these choices are purely heuristic. Through experimentation, we have found these values to work best for the wide range of distributions considered. The result of choosing these levels of q_1 and q_2 is that if the new point were exactly equal to the next largest value being tracked plus 10% of the distance between the largest value and the next largest value, i.e. $x_* = x_{m-1} + q_1(x_m - x_{m-1})$, then the estimated rank for this new point is equal to the estimated rank associated with the next largest value plus 90% of the distance between the estimated ranks associated with the largest value and the next largest value, i.e. $r_* = r_{m-1} + q_2(r_m - r_{m-1})$.

The score, which is calculated in Step 5, can be thought of as a 'z-score' for the target rank $n'p$. The score for each point could be viewed as the 'probability' that the target rank came from a distribution with a mean equal to the estimated rank and a standard deviation equal to the assigned weight. The larger the score, the less likely it is that the target rank is similar to the estimated rank, and the less likely that the associated point is similar to the target sample quantile.

If a new point is 'close' to one of the points being tracked, then the estimated rank of the new point will be 'close' to the estimated rank of that point. When a new point is 'close' to a tracked point, the associated weight will be 'small', reflecting the idea that there is little new information associated with the rank estimate of that new point. As a result, the new point's score will tend to be 'large' and more likely to be thrown out. This makes intuitive sense since a point that is 'close' to a current point will not be giving us much new information. However, if a new point is 'close' to the middle of the two adjacent points that are being tracked, then the estimated rank will tend not to be 'close' to the estimated rank of one of the tracked points. In this case the associated weight tends to be 'large', the associated score tends to be 'small' and the new point is more likely to be kept. Again this makes sense because a point that falls farther away from the current points is giving us new information which may tend to be more useful.

4. Simulation results

In this section we present the results of four simulation studies. The first two studies (Tables 1 and 2) focus on assessing the performance of the proposed method as compared with the methods reviewed in Hurley and Modarres (1995), with regard to estimation of the median. The third study (Tables 3 and 4) focus on the performance of the proposed method compared with the S.A. method with regard to estimating a collection of different quantiles. The fourth study (Table 5) focused on studying the performance of the proposed method as the number of elements being tracked is increased. All simulation studies presented here were implemented using the C programming language using the Microsoft Visual C++ compiler to compile the code. This choice was made mainly for reasons of computational efficiency since the sizes of the datasets generated were so large.

4.1. *Description of studies*

The first simulation study (Table 1) is a reproduction of the study presented in Hurley and Modarres (1995) with our proposed method included. As in their paper, we use samples of size 50,625 and $m = 60$ with 1,000 replications for the standard normal, the standard Cauchy, the chi-square with 1 degree of freedom, and a mixture with 90% of the data coming from a standard normal and 10% of the data coming from a normal with mean 0 and variance 9. For the next three tables (Tables 2–4), we used data drawn from the same distributions except that the mixture had 90% of the observations drawn from standard normal and 10% of the observations drawn from a normal with mean 10 and variance 9. In the first two studies we used five different methods to estimate the median: S.A., remedian, histogram, minimax and the proposed method. In order to accommodate the restrictions on the size of the data set that can be used for both the minimax method and the remedian method we generated data sets with 3,748,096 elements for the second study. (It should be noted that the with this size of a data set, the minimax method is actually estimating the 50.44th percentile, not exactly the median.) In the third study we used the S.A. method and the proposed method to estimate the quantiles for $p = .001, .01, .05, .10, .25, .75, .90, .95, .99$, and .999 and we generated data sets with 10,000,000 observations. In studies 2, 3, and 4 we used 100 replications. For the fourth study we used data that was drawn from the standard normal and standard Cauchy densities for the purpose of examining the effect of increasing the value of m.

For each study and each quantity being estimated we calculated the average estimate, the mean square error (MSE) of each estimation method with regards to the population quantile and the MSE of the sample quantile with regards to the population quantile. The efficiency of each algorithm was assessed by calculating the ratio of these two MSE estimates. Our motivation for calculating this MSE ratio is to create a measure of how each estimator's variation compares to that of the sample quantile. We also present a measure called MSE^* which we define to be the average squared deviation of the proposed estimate from the true sample quantile. This measure differs from the usual mean

Table 1. *Median comparison—50,625 observations with m = 60*

Distribution	Normal	Cauchy	χ^2_1	Mixture
True median	0.00000	0.00000	0.45494	0.00000
	Avg. est.	Avg. est.	Avg. est.	Avg. est.
	mse ratio	mse ratio	mse ratio	mse ratio
Method	mse*	mse*	mse*	mse*
Histogram	0.00104	0.00138	0.45580	0.00077
	1.020	1.025	1.031	1.008
	5.8e–05	1.4e–06	7.6e–07	6.9e–05
Stochastic	0.00024	0.00027	0.45497	−0.00004
approx.	1.007	1.017	1.008	1.012
	5.7e–05	1.3e–06	2.1e–07	6.9e–05
Remedian	0.00002	−0.00054	0.45508	−0.00052
	3.383	3.351	3.451	3.733
	1.3e–04	1.2e–04	5.3e–05	1.6e–04
Minimax	−0.02252	−0.02851	0.43690	−0.02616
	79.730	78.414	148.459	80.723
	2.5e–03	3.8e–03	1.6e–03	2.9e–03
Proposed	0.00022	0.00035	0.45505	−0.00010
method	0.998	0.995	0.996	0.997
	5.7e–05	1.2e–07	5.8e–08	6.9e–05

Avg. est. = median estimates averaged over 1000 runs.
mse ratio = MSE for given estimator/MSE for sample quantile.
mse* = average squared deviation of a given estimate from sample quantile estimate.

Table 2. *Median comparison—3,748,096 observations with m = 100*

Distribution	Normal	Cauchy	χ^2_1	Mixture
True median	0.00000	0.00000	0.45494	0.13959
	Avg. est.	Avg. est.	Avg. est.	Avg. est.
	mse ratio	mse ratio	mse ratio	mse ratio
Method	mse*	mse*	mse*	mse*
Histogram	0.00002	−0.00001	0.45493	0.13950
	1.003	0.987	1.003	0.966
	3.6e–09	4.9e–09	2.3e–09	4.5e–09
Stochastic	−0.00003	−0.00007	0.45489	0.13944
approx.	1.004	0.991	1.001	0.988
	5.5e–10	7.1e–10	3.8e–10	6.9e–10
Remedian	0.00016	0.00007	0.45569	0.13952
	3.444	3.904	5.912	4.119
	1.1e–06	1.9e–06	1.4e–06	1.3e–06
Minimax	0.01346	0.01776	0.46250	0.15253
	821.465	1125.343	1683.415	895.516
	3.8e–04	7.6e–04	2.2e–04	4.0e–04
Proposed	−0.00003	−0.00007	0.45489	0.13944
method	1.007	0.998	0.996	0.999
	1.6e–10	1.9e–10	1.4e–10	2.7e–10

Avg. est. = median estimates averaged over 100 runs.
mse ratio = MSE for given estimator/MSE for sample quantile.
mse* = average squared deviation of a given estimate from sample quantile estimate.

Table 3. *Arbitrary quantile comparison—10,000,000 observations with m = 100*

	Normal			Cauchy		
	Method	S.A.	Proposed	Method	S.A.	Proposed
		Avg. est.	Avg. est.		Avg. est.	Avg. est.
		mse ratio	mse ratio		mse ratio	mse ratio
p	True	mse*	mse*	True	mse*	mse*
0.001	−3.0902	−3.0733	−3.0902	−318.31	−519.63	−318.57
		8415.2	0.993		180928.7	1.031
		7.6e–02	2.2e–08		1.5e + 06	4.7e–01
0.01	−2.3264	−2.3295	−2.3262	−31.821	−44.110	−31.808
		72.86	1.001		104023.9	1.008
		7.4e–05	1.8e–09		1.1e + 03	2.9e–05
0.05	−1.6449	−1.6448	−1.6448	−6.3138	−6.6253	−6.3137
		0.991	0.994		10020.9	0.998
		1.2e–09	2.4e–10		7.4e–01	4.7e–08
0.10	−1.2816	−1.2816	−1.2816	−3.0777	−3.0912	−3.0777
		1.005	0.995		142.71	0.999
		3.1e–10	1.6e–10		1.2e–03	3.2e–09
0.25	−0.6745	−0.6744	−0.6744	−1.0000	−1.0001	−1.0001
		1.007	0.997		1.014	0.991
		2.0e–10	6.9e–11		2.2e–09	2.0e–10
0.75	0.6745	0.6745	0.6745	1.0000	1.0000	1.0000
		1.000	0.996		0.992	1.002
		1.9e–10	6.3e–11		2.3e–09	1.8e–10
0.90	1.2816	1.2815	1.2815	3.0777	3.0916	3.0779
		0.995	1.002		212.21	1.002
		3.9e–10	1.3e–10		2.4e–03	2.7e–09
0.95	1.6449	1.6448	1.6448	6.3138	6.5340	6.3154
		0.999	1.006		12337.7	0.992
		1.2e–09	2.6e–10		1.0e–00	3.4e–08
0.99	2.3264	2.3261	2.3264	31.821	34.211	31.839
		11.56	0.996		34137.4	0.997
		1.1e–05	5.0e–09		3.6e + 02	2.5e–05
0.999	3.0902	2.9877	3.0894	318.31	1159.93	317.60
		3179.5	1.088		2009247.5	1.168
		3.1e–02	1.1e–06		2.2e + 07	2.8e + 00

Avg. est. = median estimates averaged over 100 runs.
mse ratio = MSE for given estimator/MSE for sample quantile.
mse* = average squared deviation of a given estimate from sample quantile estimate.

squared error in that we measure deviations from the sample quantile instead of the population quantile. Our motivation for including MSE^* is that we want to see how closely an estimator's performance tracks that of the sample quantile.

In the first three studies, the proposed method only kept track of 100 data points at any one time, or $m = 100$. As a result the total storage needed then was 300 memory locations; 100 for each of the arrays used in the algorithm. In these studies, we also allowed the S.A. and histogram methods to use initial samples of size 300.

4.2. *Results for median comparison*

Here we present the results of the two median comparison studies. Table 1 gives results of the recreation of the study done by Hurley and Modarres (1995). The results show similar results as given by Hurley and Modarres but with the inclusion of our proposed method. Our proposed method appears to perform as well as or slightly better than the other methods considered.

The next median comparison study was conducted with larger samples of size 3,748,096 and with appropriate parameter settings for the various methods. The simulation results agree with those in Table 1 (see Table 2). The minimax estimator consistently displayed the worst performance of all methods. However, this can be attributed to 2 facts. First, for the given parameter values, the estimator is actually estimating the 50.44th percentile. Hence all estimates will have a slight positive bias. Secondly, the height of the minimax tree is quite small, which is theoretically known to impact the performance of the minimax method. The remedian estimator had MSE ratios of between 3 and 4.

Table 4. *Arbitrary quantile comparison—10,000,000 observations with m = 100*

	Chi-square			Mixture		
	Method	S.A. Avg. est. mse ratio	Proposed Avg. est. mse ratio	Method	S.A. Avg. est. mse ratio	Proposed Avg. est. mse ratio
p	True	mse*	mse*	True	mse*	mse*
0.001	0.000002	−0.000223	0.000002	−3.0590	−2.9732	−3.0589
		36375319.0	0.967		7758.5	1.003
		5.7e–08	4.3e–17		5.7e–02	2.2e–08
0.01	0.00016	0.00016	0.00016	−2.2867	−2.2897	−2.2867
		1.060	0.993		47.9	1.011
		2.7e–13	9.2e–16		8.0e–05	1.6e–09
0.05	0.00393	0.00393	0.00393	−1.5933	−1.5933	−1.5933
		0.997	0.997		1.006	1.002
		3.6e–13	6.1e–14		2.2e–09	2.1e–10
0.10	0.01579	0.01579	0.01579	−1.2207	−1.2207	−1.2207
		1.007	0.998		0.998	0.996
		1.1e–12	3.4e–13		5.5e–10	1.1e–10
0.25	0.10153	0.10151	0.10152	−0.5895	−0.5895	−0.5895
		1.005	1.000		1.002	0.998
		1.1e–11	4.5e–12		1.8e–10	5.9e–11
0.75	1.3233	1.3233	1.3233	0.9668	0.9668	0.9668
		1.014	1.000		1.007	0.994
		8.8e–10	1.5e–10		3.5e–10	1.1e–10
0.90	2.7055	2.7055	2.7055	3.0509	4.0626	3.0518
		0.991	0.999		5492.9	1.010
		3.9e–08	8.1e–10		2.8e–00	8.2e–07
0.95	3.8415	3.8422	3.8415	10.000	9.977	9.999
		4.473	0.996		1726.8	1.000
		1.6e–05	2.7e–09		3.9e–02	1.3e–08
0.99	6.6349	6.5630	6.6342	13.845	13.803	13.844
		4237.4	1.007		6093.2	1.011
		1.0e–01	8.5e–08		1.7e–01	1.2e–07
0.999	10.828	10.115	10.821	16.979	16.472	16.976
		15156.3	1.143		15490.9	1.119
		5.2e–00	3.8e–05		1.9e–00	1.5e–05

Avg. est. = median estimates averaged over 100 runs.
mse ratio = MSE for given estimator/MSE for sample quantile.
mse* = average squared deviation of a given estimate from sample quantile estimate.

The MSE ratios for the histogram, S.A., and proposed methods were all very close to 1 for all of the distributions. The proposed method appears to have a slight advantage with regards to the MSE^* measurement. Hence for median estimation, one could employ either the histogram, S.A., or the proposed method and achieve comparable performance results.

4.3. *Results for arbitrary quantile comparison*

As mentioned earlier, for this study we only compared the S.A. method with the proposed method. Tables 3 and 4 give results regarding performance for estimating a wide range of quantiles for the four densities under study. The differences here are much more pronounced than in the median comparison. To emphasize this point, consider the results for the normal distribution with $p = 0.001$. The MSE ratio of the S.A. estimator is 8415.2 whereas the MSE ratio of our method is 0.993. Similar observations hold for the other quantiles and distributions considered.

Another interesting result is for $p = 0.001$ for the chi-square density. The averaged estimates for the S.A. method is −0.000223 which is not within the range of values allowed for the chi-square density. In addition the large MSE ratio for this quantile suggests that the S.A. method has potential problems when estimating values near the extreme tail of a density with a hard cut-off (as is the case here with the chi-square at zero).

The poor tail performance of the S.A. method could be improved by taking a larger initial sample and thereby obtaining a more accurate starting value. For example, in the above example with $p = 0.001$ for the normal density, if instead of an initial

Table 5. *Proposed method varying m for median estimation using 10,000,000 observations*

m	Normal Avg. est. mse ratio mse*	Cauchy Avg. est. mse ratio mse*
40	0.000029	−0.001270
	1.003	143.979
	2.2e–10	3.3e-05
60	0.000023	−0.000087
	0.993	3.154
	5.5e–11	4.0e-07
80	0.000023	0.000001
	0.997	1.000
	4.8e–11	9.2e-11
100	0.000024	−0.000001
	0.997	1.001
	3.1e–11	5.3e-11
500	0.000024	−0.000001
	1.000	1.002
	4.7e–12	1.1e-11
1000	0.000023	−0.000001
	1.001	1.002
	2.2e–12	4.0e-12

Avg. est. = median estimates averaged over 100 runs.
mse ratio = MSE for given estimator/MSE for sample quantile.
mse* = average squared deviation of a given estimate from sample quantile estimate.

sample of size 300 we increase the size to 10,000, the MSE ratio decreases from 8415 to 564. For the Cauchy density with $p = 0.999$, if we again allow an initial sample of size 10,000 instead of 300 the MSE ratio goes from over 2,000,000 down to approximately 1099. These are both considerable improvements, but still not close to ratios of 1. One could conceivably continue to increase the size of the initial sample, however doing so would introduce a new storage burden.

4.4. *Results for changing the number of points being tracked*

In this section, we explore the impact of varying the value of m, the size of the array of points being tracked. As can be seen in Table 5, the performance of the proposed estimator as a function of the number of data points being tracked appears to stabilize for $m = 60$ or higher for the normal density and for $m = 80$ or higher for the Cauchy density.

This study suggests that array sizes of 100 may be appropriate for most cases in practice.

5. Theoretical considerations

This algorithm raises questions regarding whether the resulting estimator converges to the target population quantile. We approach the convergence issue by monitoring the distance between our estimator and the target sample quantile as new data points are observed (see Fig. 2 for an example of a plot of this difference for the median from a dataset drawn from the Cauchy density).

Intuitively this figure suggests that the proposed method should always be in some order statistic neighborhood of the true sample quantile at any given step. Experimentation has shown that when estimating the median the proposed method is always within $\pm\sqrt[3]{m+k}$ order statistics of the true sample median for each step of the algorithm. Here k denotes the number of data points that have been observed after the first m points. If the proposed estimator always remains within this range then the following lemma demonstrates that the estimator will converge to the target population quantile.

Lemma 1. *Let $X_1, \ldots, X_n$ be a random sample from a continuous distribution function F and let $0 < p < 1$. Further assume that F is twice differentiable at $\xi_p = F^{-1}(p)$ and that $F'(\xi_p) = f(\xi_p) > 0$. Then*

$$\sqrt{n}\big(X_{(np+\sqrt[3]{n})} - X_{(np-\sqrt[3]{n})}\big) \to 0,$$

as $n \to \infty$.

Proof: Without loss of generality, we can assume $np \pm \sqrt[3]{n}$ are both integer valued indices. If they are not integer valued, we can apply the greatest integer function. Then by Bahadur's representation of the sample quantiles (see Bahadur 1966 and Serfling 1980) we have

$$X_{(np\pm\sqrt[3]{n})} = \xi_p + \frac{p \pm n^{-2/3} - F_n(\xi_p)}{f(\xi_p)} + O\big(n^{-3/4}(\log n)^{3/4}\big),$$

with probability 1 as $n \to \infty$. Therefore we have

$$X_{(np+\sqrt[3]{n})} - X_{(np-\sqrt[3]{n})} = \frac{2n^{-2/3}}{f(\xi_p)} + O\big(n^{-3/4}(\log n)^{3/4}\big),$$

and hence the result follows. □

In all of our simulations the proposed estimator has stayed within $\pm\sqrt[3]{m+k}$ order statistics of the true sample median and we conjecture that this result will hold when the proposed method is used for data that comes from an arbitrary continuous distribution.

6. Discussion

We have demonstrated the ability of our method to accurately estimate arbitrary quantiles (including tail quantiles) from an unknown distribution. Using a wide range of quantiles and distributions we have shown performance differences between our method and existing methods.

Among the methods discussed, the only estimators which are easily extended to accommodate tail quantile estimation are the sample quantile estimator, the S.A. estimator of Tierney (1983),

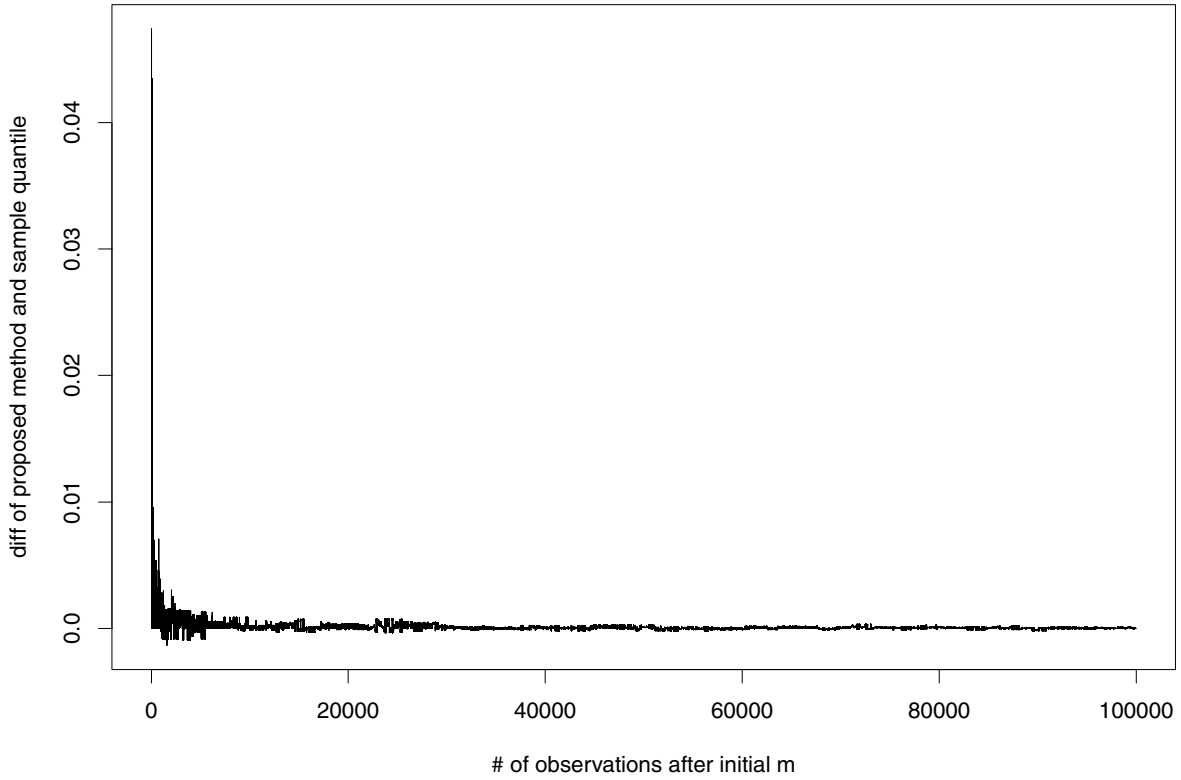

Fig. 2. *Sequential difference between proposed method and S.Q. for median estimation of a Cauchy sample*

and the proposed estimator. The sample quantile estimator becomes too costly to employ as the size of the dataset becomes large, making it unattractive or infeasible for large datasets. The S.A. method's accuracy is dependent upon the size of the initial sample taken for the starting value. The larger the initial sample, the better the starting value and hence the better the accuracy of the estimator. The proposed method's accuracy is dependent upon m, the size of the array of points being tracked. The larger the array size, the more closely it approximates the performance of the sample quantile estimator. However, based upon our experimental results, setting $m = 100$ is a reasonable choice.

When compared with the S.A. method, the proposed method has at least three advantages over other methods. First, it returns an actual observation as an estimate. In comparison there is nothing to prevent the S.A. method from returning an estimate that is outside the range of possible values for the distribution that generated the data. Second, the proposed method performs better than the S.A. method with regards to approximating the sample quantile, particularly for tail quantiles. It appears that the only way to overcome that difference in performance is to increase the size of the initial sample for the S.A. method. A third advantage has to do with using each method to estimate a collection of quantiles simultaneously. The S.A. method scales in a linear fashion with regards to computational time and storage requirements, i.e. estimating n quantiles would take n times as long as estimating one quantile. The proposed method could be easily extended in such a way that it would require less than a one-to-one increase in computational time and storage requirements. Exploring this claim is a topic of current research.

Acknowledgment

The authors wish to thank the editor and the referees for several very helpful suggestions that greatly improved the quality of the paper.

References

Bahadur R.R. 1966. A note on quantiles in large samples. Annals of Mathematical Statistics 37: 577–580.

Breiman L., Gins J., and Stone C. 1979. New Methods for Estimating Tail Probabilities and Extreme Value Distributions. Technology Service Corp. Santa Monica, CA, TSC-PD-A2261.

Chao M.T. and Lin G.D. 1993. The asympotic distribution of the remedians. Journal of Statistical Planning and Inference 37: 1–11.

Dunn C.L. 1991. Precise simulated percentiles in a pinch. The American Statistician 45(3): 207–211.

Hurley C. and Modarres R. 1995. Low-Storage quantile estimation. Computational Statistics 10(4): 311–325.

Kesidis G. 1999. Bandwidth adjustments using on-line packet-level adjustments. In: SPIE Conference on Performance and Control of Network Systems, Boston, Sept. 19–22.

Krutchkoff R.G. 1986. Percentiles by simulation: Reducing time and storage. Journal of Statistical Computation and Simulation 25: 304–305.

Manku G.S., Rajagopalan S., and Lindsay B.G. 1998. Approximate medians and other quantiles in one pass and with limited memory. In: Proc. ACM SIGMOD International Conf. on Management of Data June, pp. 426–435.

Pearl J. 1981. A space-efficient on-line method of computing quantile estimates. Journal of Algorithms 2: 164–177.

Ott W.R. 1995. Environmental Statistics and Data Analysis. Lewis Publishers.

Pfanzagl J. 1974. Investigating the quantile of an unknown distribution. Contributions to Applied Statistics, Ziegler W.J. (Ed.), Birkhauser Verlag, Basel, pp. 111–126.

Rousseeuw P.J. and Bassett G.W. 1990. The remedian: A robust averaging method for large datasets. Journal of the American Statistical Association 85(409): 97–104.

Serfling R.J. 1980. Approximation Theorems of Mathematical Statistics. Wiley, New York.

Tierney L. 1983. A space-efficient recursive procedure for estimating a quantile of an unknown distribution. SIAM Journal on Scientific and Statistical Computing 4(4): 706–711.