Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds.

LOW-STORAGE ONLINE ESTIMATORS FOR QUANTILES AND DENSITIES

Soumyadip Ghosh

Business Analytics and Math Sciences
T.J. Watson IBM Research Center
Yorktown Heights, NY 10598, USA

Raghu Pasupathy

Industrial and Systems Engineering
Virginia Tech
Blackburg, VA, 24061, USA

ABSTRACT

The traditional estimator $\hat{\xi}_{p,n}$ for the p-quantile ξ_p of a random variable X, given n observations from the distribution of X, is obtained by inverting the empirical cumulative distribution function (cdf) constructed from the obtained observations. The estimator $\hat{\xi}_{p,n}$ requires $O(n)$ storage, and it is well known that the mean squared error of $\hat{\xi}_{p,n}$ (with respect to ξ_p) decays as $O(n^{-1})$. In this article, we present an alternative to $\hat{\xi}_{p,n}$ that seems to require dramatically less storage with negligible loss in convergence rate. The proposed estimator, $\tilde{\xi}_{p,n}$, relies on an alternative cdf that is constructed by accumulating the observed random variates into variable-sized bins that progressively become finer around the quantile. The size of the bins are strategically adjusted to ensure that the increased bias due to binning does not adversely affect the resulting convergence rate. We present an "online" version of the estimator $\tilde{\xi}_{p,n}$, along with a discussion of results on its consistency, convergence rates, and storage requirements. We also discuss analogous ideas for density estimation. We limit ourselves to heuristic arguments in support of the theoretical assertions we make, reserving more detailed proofs to a forthcoming paper.

1 INTRODUCTION

For a random variable X with cumulative distribution function (cdf) F, the p-quantile ξ_p is defined as $\xi_p = \underset{x}{\text{argmin}}\{F(x) \geq p\}$. This is also the definition of the general inverse of F, and so $\xi_p = F^{-1}(p)$. Estimation of quantiles is of natural interest in numerous contexts: financial applications often require estimation of the "Value-at-Risk" of a financial instrument (Duffie and Pan 1997); service systems such as call-centers often provide quality-of-service guarantees in terms of the 95-th percentile of the response time being within specified bounds (Goldenberg, Qiuy, Xie, Yang, and Zhang 2004, Ghosh and Ghosh 2012); the service metric of an information technology service provider may be tied to the 95-th percentile of specific measurable quantities such as network latency (Goldenberg, Qiuy, Xie, Yang, and Zhang 2004), or CPU latency for cloud service providers (Ghosh and Ghosh 2012); and in a simulation context, "systems" however defined, may be compared based on their performance expressed in terms of a quantile (Bekki, Fowler, Mackulak, and Nelson 2007, Pasupathy, Szechtman, and Yücesan 2010).

Our primary interest in this paper is that of estimating the p-quantile of a random variable X, given independent and identically distributed (iid) random variates $X_i, i = 1,2,\ldots$ from the distribution $F(\cdot)$ of X. The data $X_i, i = 1,2,\ldots$ used to construct the quantile estimator are "observed" sequentially, that is, they are made available not simultaneously, but one at a time. Furthermore, the estimation contexts of interest are "online" implying that the storage and the computational complexity of the quantile estimator are of particular importance.

To illustrate the latter issue of storage and computational complexity, consider estimating the mean $\mu = \mathbb{E}[X]$ of the random variable X. The traditional estimator in this case is the simple sample mean

978-1-4799-2076-1/13/$31.00 ©2013 IEEE

$\overline{X}(n) = n^{-1}\sum_{i=1}^{n} X_i$ which can be recursively estimated as

$$\overline{X}(n) = \frac{n-1}{n}\overline{X}(n-1) + \frac{1}{n}X_n, \quad n = 2, 3, \ldots.$$

It is clear from the above recursion that, irrespective of the value of n, $\overline{X}(n)$ can obtained through a manipulation of only the previous estimator $\overline{X}(n-1)$ and the most recent observation X_n, implying $O(1)$ storage and computational complexities.

What are the corresponding complexities for the traditional estimator $\hat{\xi}_p$ for the p-quantile $\xi_p = F_X^{-1}(p)$ of X, obtained by inverting the empirical cdf constructed from the iid observations $\{X_i, i = 1, 2, \ldots, n\}$? Suppose the set $\{X_i, i = 1, 2, \ldots, n\}$ is sorted in ascending order to form the list $\{X_{(1)}, X_{(2)}, \ldots, X_{(n)}\}$. Then, assuming no ties, the inverse of the empirical cdf $\hat{F}_n$ at p is the same as $\lceil np \rceil$-th value in the sorted list, that is, $\hat{F}_n = X_{(\lceil np \rceil)}$. This estimator requires that the entire sorted list be available, and so storage required is $O(n)$. The best algorithms available to maintain a list of values in ascending sorted order incur $O(\log n)$ computational complexity for each step that updates the sorted list with the next sample X_n and retrieves the latest estimate $\hat{\xi}_{p,n}$ as the $\lceil np \rceil$-th order-statistic (Bayer 1972). Thus, the complexity of the traditional quantile estimator as an "online" context is $O(n)$ for storage and $O(\log n)$ for the estimate update computation.

We present a competing quantile estimator $\tilde{\xi}_p$ that appears to require much less storage and computational complexity (provably $O(\log n)$ and $O(\log\log n)$, respectively) than the traditional estimator $\hat{\xi}_p$. Furthermore, such gains in storage and computation appear to come at little cost in terms of convergence characteristics. Specifically, it is well-known that the traditional quantile estimator $\hat{\xi}_p$ is a consistent estimator whose mean-squared error converges to zero as $O(n^{-1})$ (Serfling 1980). The alternative estimator $\tilde{\xi}_p$ that we propose, while enjoying reduced storage and computational complexity, retains the convergence rate of the traditional estimator. The main idea underlying the alternative estimator $\tilde{\xi}_p$ is the construction of an alternative empirical cdf $\tilde{F}_n(\cdot)$ that is sensitive to data storage and computation. The cdf $\tilde{F}_n(\cdot)$ is constructed using bins that progressively become finer around the location of the quantile. This essentially implies that all observations except those that lie close to the quantile are grouped, thereby reducing storage. The challenge lies in deciding how to strategically reduce the bin sizes around the quantile, so as to reduce storage and computational complexity but to maintain the increased bias at a level that does not affect the convergence rate of the resulting estimator.

We also show that an analogous "online" estimator can be constructed for the context of estimating densities. Using ideas akin to those discussed for constructing low-storage quantile estimation, we outline a method by which the traditional kernel density estimators can be modified to achieve less storage and computation while retaining the traditional kernel density estimator's convergence rate. Our treatment of online density estimators is cursory because much of the ideas we present follow from results in Hall and Wand (1996).

The following is an organization of the rest of the paper. Section 2.1 provides an overview of the traditional quantile estimator, its properties and the variations that have been developed to address certain shortcomings. Section 2.2 presents the proposed low-storage online estimator, with the main algorithm being defined in Section 2.2.1, and its key properties in Section 2.2.2. Other related quantities that can be speedily estimated in a similar fashion are described in Section 2.2.4. For densities, Section 3 provides a brief overview of traditional density estimation and Section 3.1 briefly describes the low-storage version of the estimator, along with its key properties.

2 QUANTILE ESTIMATION

In this section, we start with a brief overview of the traditional quantile estimator. The literature on quantile estimation is enormous and we limit ourselves to listing the bare minimum that is needed understand the online estimator that we describe in Section 2.2. As noted earlier, while we provide a detailed algorithm listing and numerical performance for the estimator we propose, we support the theoretical assertions we make using only heuristic arguments. Detailed proofs for these assertions are reserved for a forthcoming paper.

2.1 Standard Estimators

The traditional quantile estimator or the crude Monte Carlo estimator (henceforth CMC estimator) for the p-quantile of the random variable X with cdf F rank-orders n observations $\{X_i,\ i = 1,\ldots,n\}$ of X from the cdf F and returns the $\lceil p \rceil$-th order statistic. This is equivalent to constructing the *empirical cdf*

$$\hat{F}_n(x) = \sum_n I(X_n \leq x), \tag{1}$$

and then estimating the quantile ξ_p as the inverse of the empirical cdf: $\hat{\xi}_{n,p} = \hat{F}_n^{-1}(p)$. The statistical properties of $\hat{\xi}_{n,p}$ are well-understood and derived from the famous Bahadur representation of $\hat{\xi}_{n,p}$ (Chu and Nakayama 2012), given as

$$\hat{\xi}_{p,n} = \xi_p - \frac{F_n(\xi_p) - p}{f(\xi_p)} + R_n, \tag{2}$$

where $R_n = O(n^{-3/4}(\log\log n)^{3/4})$. To see why this relationship may hold, consider the following heuristic argument. For a large enough sample size n, and assuming the density $f = \partial F/\partial x$ exists, one can expect

$$p \approx F(\hat{\xi}_{p,n}) \approx F(\xi_p) + f(\xi_p)(\hat{\xi}_{p,n} - \xi_p) \approx \hat{F}_n(\xi_p) + f(\xi_p)(\hat{\xi}_{p,n} - \xi_p),$$

where the first approximation uses a first-order Taylor series expansion and the second is implied by the convergence of $\hat{F}_n$ to F. Now rearrange terms to arrive at an expression that resembles (2).

The expression in (2) tells us that the bias in the estimator $\hat{\xi}_{p,n}$ is of the same order as the bias in the empirical cdf $\hat{F}_n$, which in great generality is known to be $O(n^{-1})$ (Serfling 1980). The representation in (2) can also be used to establish a CLT for $\hat{\xi}_{p,n}$:

$$\sqrt{n}(\hat{\xi}_{p,n} - \xi_p) \xrightarrow{d} N(0, \frac{p(1-p)}{f^2(\xi_p)}), \tag{3}$$

where $\xrightarrow{d}$ denotes convergence in distribution, $N(\mu,\sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2, and f is the density corresponding to the cdf F, assumed to obey $f(\xi_p) > 0$. Thus, the variance of the CMC estimator, just like its bias, decays as $O(n^{-1})$.

Remark 1 The expression $p(1-p)$ appearing in the variance constant of the CLT in (3) implies that as p approaches either boundary of the domain $(0,1)$, the CMC estimator has a diverging variance. This issue has inspired alternate estimators that use variance reduction techniques from the rare-event mean or moment estimation literature to more efficiently estimate the cdf F as p approaches 0 or 1; examples include control variates (Hsu and Nelson 1990), importance sampling (Glynn 1996) and stratified sampling (Avramidis and Wilson 1995). Chu and Nakayama (2012) provide a unified theoretical framework for the asymptotic convergence of the variance constant for such estimators.

Two simple ideas, *batching* and *sectioning*, are worthy of mention due to their connection with the estimator we propose. Note that the CLT in (3) cannot be directly used in constructing a confidence interval on $\hat{\xi}_{p,n}$ because the term $f(\xi_p)$ appearing as part of the variance constant is unknown. Batching and sectioning are ideas to circumvent this issue (Nakayama 2012). Both batching and sectioning essentially divide the n available observations into M non-overlapping sub-sets of $s = n/M$ observations. Empirical cdf's $\hat{F}_s^{(m)}$ and quantile estimates $\hat{\xi}_{p,s}^{(m)} = (\hat{F}_s^{(m)})^{-1}(p)$ are then constructed from each of the M subsets. The batching procedure then constructs the variance estimate as

$$\sigma^2_{n,batch} = \sum_{m=1}^{M} \left(\hat{\xi}_{p,s}^{(m)} - \bar{\hat{\xi}}_{p,s} \right)^2,$$

where $\bar{\hat{\xi}}_{p,s} = \sum_m \hat{\xi}_{p,s}^{(m)}/M$ is the average of the M quantiles from the M batches. The sectioning procedure constructs the slightly different variance estimate

$$\sigma^2_{n,section} = \sum_{m=1}^{M} \left(\hat{\xi}_{p,s}^{(m)} - \hat{\xi}_{p,n} \right)^2,$$

where $\hat{\xi}_{p,n}$ is the quantile estimate from the empirical cdf $\hat{F}_n$ constructed with the entire sample set of size n. Nakayama (2012) finds that the $\sigma^2_{n,section}$ provides a better estimate of the variance constant for a variety of quantile estimators, including the crude Monte Carlo and variance reducing estimators.

2.2 Low-Storage Online Estimator

The fundamental idea underlying the estimator we propose stems from a single important fact: the squared bias of the CMC estimator discussed in the previous section decays as $O(n^{-2})$ while its variance decays as $O(n^{-1})$. Since the mean squared error is the sum of the squared bias and the variance, alternative estimators that can trade-off increased bias in return for less total storage and computational complexity will be superior as long as the new bias does not exceed $O(n^{-1/2})$.

The procedure we propose groups data into appropriately sized bins, from which an empirical cdf is constructed and inverted to obtain the alternative quantile estimator. The grouping of observations causes increased bias, but since the bins are strategically sized to ensure that the resulting bias does not exceed $O(n^{-1/2})$, the canonical rate $O(n^{-1})$ is retained while dramatically reducing the storage and computational requirements. In what follows, we provide complete details of the proposed estimator.

2.2.1 Algorithm

The proposed estimator achieves a balance of the asymptotic rate of drop in estimator bias and variance by replacing the empirical cdf $\hat{F}_n$ in the CMC estimator with an empirical cdf $\tilde{F}_n$ constructed from a binned histogram. The histogram is created over a partition $\mathscr{P} = \{a_0, \ldots, a_{k+1}\}$ of the support of the cdf F, where the end points may be $\pm\infty$. The mass $\tilde{p}_{l,n}$ assigned to the l-th histogram bin $B_l = (a_{l-1}, a_l]$ is calculated as $\tilde{p}_{l,n} = \sum_{i=1}^{n} I(X_i \in B_l)$. The binned empirical cdf $\tilde{F}_n$ is calculated by interpolating the mass p_l uniformly throughout the l-th bin. Thus, $\tilde{F}_n(x) = \sum_{i=1}^{l-1} \tilde{p}_{i,n} + g_l(x)\tilde{p}_{l,n}$, where $x \in B_l$. The interpolation function $g_l(x) = (x - a_{l-1})/(a_l - a_{l-1})$ is linear for all finite bins B_l. An appropriate tail interpolation g_l will have to be used for the right- and left-end bins if the end-points are $\pm\infty$ such that it satifies $\int_{B_{l,n}} g_l(x)dx < \infty$. The quantile ξ_p is then estimated as $\tilde{\xi}_{p,n} = \tilde{F}_n^{-1}(p)$.

Algorithm 1 describes the online procedure used by the proposed estimator to maintain the binned empirical cdf $\tilde{F}_n$ as the sample size n grows. The cdf $\tilde{F}_n$ has a support set $\mathscr{P}_{k(n)}$ with a total number of bins $k(n)$ that is dynamically updated based on the statistical properties of the quantile estimate $\tilde{\xi}_{p,n}$. The procedure need only store $O(k(n))$ amount of data to represent $\tilde{F}_n$. The size of the histogram $k(n)$ is adjusted according to the bias and variance of the estimator $\tilde{\xi}_{p,n}$.

Let $l(n)$ represent the index of the bin in the support $\mathscr{P}_{k(n)}$ of the empirical cdf $\tilde{F}_n$ that contains the estimate $\tilde{\xi}_{p,n}$, that is $\tilde{\xi}_{p,n} \in B_{l(n)}$. The bias in the estimate $\tilde{\xi}_{p,n}$ is bounded above by the size b_n of the bin $B_{l(n),n}$; see Claim 1 in Section 2.2.2. Algorithm 1 estimates the variance of the estimate $\tilde{\xi}_{p,n}$ by implementing an online version of the sectioning concept described by Nakayama (2012): the incoming

Algorithm 1 Online Low-Storage Quantile Estimation Procedure

Problem Parameters: probability $p \in (0,1)$ for which quantile ξ_p is to be estimated; an oracle to sample from cdf F

Algorithm Parameters: an initial k_0-partition $\mathscr{P}_0 = \{a_0, \ldots, a_{k_0+1}\}$ for empirical cdf construction; values for parameters M (number of sections for variance computation), γ (variance-bias comparison)

```
Initialization
Set $n = 0$                                                          ▷ total sample counter
Set $s = 0$                                                          ▷ section sample counter
Set $k = 0$                                                          ▷ additional points in $\mathscr{P}$
Initialize set of $M+1$ empirical cdfs $\{\tilde{F}_n, \tilde{F}_s^{(1)}, \ldots, \tilde{F}_s^{(M)}\}$ using an initial sample size to calculate
  appropriate probability masses $\tilde{p}_{l,n}$ over the initial partition $\mathscr{P}_0$.
Estimation
while further estimation is required do
    Gather the next set of $M$ samples $\{X_i,\ i = sM+1, \ldots, (s+1)M\}$.
    Update empirical cdf $\tilde{F}_n$ with the $M$ new values.
    Re-calculate the estimate $\tilde{\xi}_{p,n}$.
    Let $n = n + M$.
    For each $m = 1, \ldots, M$, add $X_{sM+m}$ to $\tilde{F}_s^{(m)}$
                        ▷ Update each of the $M$ variance-calculation cdfs $\tilde{F}_s^{(m)}$ with one value each.
    Set $s = s + 1$.
    Re-calculate the section estimates $\tilde{\xi}_{p,s}^{(m)}$.
    Update the variance estimate
        $$\sigma^2_{n,section} = \frac{1}{M-1} \sum_{m=1}^{M} (\tilde{\xi}_{p,s}^{(m)} - \tilde{\xi}_{p,n})^2.$$
    Set $l(n)$ as the index of the bin $B_l$ that contains $\tilde{\xi}_{p,n}$.
    Set $b_{l(n)} = a_{l(n)} - a_{l(n)-1}$.                                  ▷ length of the bin $B_{l(n)}$.
    if $\sigma^2_{n,section} < \gamma\, b^2_{l(n)}$ , then
        Set $a^{(k+1)} = a_{l(n)-1,n} + (a_{l(n),n} - a_{l(n)-1,n})/2$.
        Set $\mathscr{P}_{k+1} = \mathscr{P}_k \cup \{a^{(k+1)}\}$.   ▷ Introduce a new support point to all the $M+1$ empirical cdfs.
        Set probability mass $\tilde{p}_{l,n}$ of $\tilde{F}_n$ for newly created bins as $p_{l(n)}/2$, and similarly for each $\tilde{F}_s^{(m)}$.
        Set $k = k + 1$.
    end if
    Report $\{\tilde{\xi}_{p,n}, \sigma^2_{n,section}\}$ as the current estimate of quantile and its variance.
                        ▷ These can be used to calculate an appropriate confidence interval.
end while
```

samples X_i are partitioned into M sub-streams and a set of quantile estimates $\{\tilde{\xi}_{p,s}^{(m)},\ m = 1,\ldots,M\}$ are developed from the sub-streams that each use the same bin-support as the overall estimator $\tilde{\xi}_{p,n}$, where $s = n/M$. Algorithm 1 monitors the variance $\sigma^2_{n,section}$ thus calculated for the estimate $\tilde{\xi}_{p,n}$. Once variance is of the same order of the square of the bias (where, again, bias is represented by the size b_n of the bin $B_{l(n)}$), the procedure divides the bin $B_{l(n),n}$ into two new bins of half the size b_n by introducing a new separation point $a^{(k)} = a_{l(n)-1} + (a_{l(n)} - a_{l(n)-1})/2$.

2.2.2 Properties

The empirical cdf $\tilde{F}_n$ that provides the quantile estimate in Algorithm 1 is a generalized histogram with a support set $\mathscr{P}_k$ that implies unequal sized bins. Specifically, the bins closest to the local region around the quantile estimate $\tilde{\xi}_{p,n}$ dynamically adjust themselves as the sample size n grows in order to give a finer approximation of the empirical cdf. Thus, the binned empirical cdf $\tilde{F}_n$ is an approximation of the regular empirical cdf $\hat{F}_n$ tailored to approximate the function well only in the neighbourhood of the quantile ξ_p. We will now examine why this estimator can be expected to have the same asymptotic properties as the standard quantile estimators, and also estimate the order of the information stored by this estimator as sample size n grows. We provide only a heuristic sketch of proofs for the results presented here.

Define as $n(k)$ the total number of observations when the k-th additional o is adbservation $\mathscr{P}_{k-1}$ and let $\Delta n(k) = n(k) - n(k-1)$. For a fixed support set $\mathscr{P}_k$, let $\bar{F}$ represent the cdf constructed by linear interpolation of the true cdf values $F(a_l)$ at the support points $a_l \in \mathscr{P}_k$. Further, let $\bar{\xi}_p$ be the p-quantile of this interpolated cdf.

Consider first the estimate $\tilde{F}_n$ of the true cdf F for a fixed support $\mathscr{P}_k$.

Claim 1 Let $\tilde{F}_n$ be a binned empirical cdf defined over the support set $\mathscr{P}_k$ and let $\tilde{\xi}_{p,n} = \tilde{F}_n^{-1}(p)$. Further, assume that the set $\mathscr{P}_k$ is sufficiently fine that the length b_n of the bin indexed $l(n)$ that contains $\tilde{\xi}_{p,n}$ is finite. Then, $\mathbb{E}[\tilde{\xi}_{p,n} - \xi_p] = O(b_n)$.

Claim 1 asserts that the bias in estimating the quantile from a binned empirical cdf over the (fixed) support $\mathscr{P}_k$ is of the order of the size of the bin that contains the estimate $\tilde{\xi}_{p,n}$. To see why this can be expected, note that as n gets large, the estimate of the $\tilde{F}_n(a_l) \approx F(a_l)$ for all the support points $a_l \in \mathscr{P}_k$. Thus, the inverses of these functions also coincide at these support points, and the error in estimating the inverse of the cdf between these support points is induced by the linear interpolation of the cdf over the finite interval $(a_{l(n)-1}, a_{l(n)}]$ that contains $\tilde{\xi}_{p,n}$. This is in turn bounded above by the length $b_n = (a_{l(n)} - a_{l(n)-1})$ of the interval.

Algorithm 1 maintains the support set $\mathscr{P}_{k(n)}$ of the binned empirical cdf until a sample size n that yields a sample variance estimate $\sigma^2_{n,section}$ of the quantile estimator that is of the same order as the squared bias $b^2_{l(n)}$ of the estimation. For a fixed support $\mathscr{P}_k$, there exists a CLT of the form $\sqrt{n}|\tilde{\xi}_{p,n} - \bar{\xi}_p| \to \sigma N(0,1)$, where $\bar{\xi}_p$ is the inverse of $\bar{F}$, the interpolated cdf derived from the true cdf F at the support points of $\mathscr{P}_k$. The sectioning estimator $\sigma^2_{n,section}$ provides an estimate for this CLT's variance constant σ^2.

Step 16 ensures that the support set is not modified till $\sigma^2_{n,section} \sim b^2_{l(n)}$, which implies that the number of additional samples $\Delta n(k)$ used before the support set is updated is $O(b^{-4}_{l(n)})$. Thus, the total number of observations $n(k+1)$ when the $(k+1)$th observation is collected is $O(\sum_{i=1}^{k} b^{-4}_{l(n(i))})$. In the worst case, the algorithm may keep bisecting the same interval successively in Step 16. Thus,

$$n(k+1) = \sum_i \Delta n(i) = O(\sum_{i=1}^{k+1} b_0 2^{4(i-1)}).$$

Claim 2 The sample size $n(k+1)$ when the $(k+1)$th support point is added to the support set $\mathscr{P}_k$ satisfies

$$n(k+1) = \sum_{i=1}^{k+1} 2^{4(i-1)} = O(2^{4k}).$$

Claim 2 asserts that the series 2^{4k} is growing so fast that the summation is essentially the same order of magnitude as the last term.

Thus, the binned estimator $\tilde{\xi}_{p,n}$ balances the variance and squared bias before its storage needs grow. When sample sizes warrant a change in the support, that is $n = n(k)$, the mean squared error is $O(b_k^2) = O(\Delta n(k)^{-1}) = O(2^{-4k})$. On the other hand, the mean squared error of a standard quantile estimator at the end of $n = O(2^{4k})$ samples is the same as the rate of drop in the variance term of the estimator, which is as per the canonical rate of convergence of $O(n^{-1}) = O(2^{-4k})$. These heuristic arguments lead to Claim 3.

Claim 3 The mean squared error of the binned estimator $\tilde{\xi}_{p,n}$ decays to zero as $O(n^{-1})$.

Now consider the amount of data that needs to be stored by each estimator. The standard quantile estimator $\hat{\xi}_{p,n}$ needs all of the n observations in order to perform its calculations. On the other hand, the binned estimator $\tilde{\xi}_{p,n}$ introduces only one extra bin every $n(k)$ observations. So, if n were the total number of observations, the corresponding number of bins k can be obtained by solving $n = 2^{2k}$, to give $k = O(\log n)$.

Claim 4 The storage complexity of the binned estimator $\tilde{\xi}_{p,n}$ is $O(\log n)$.

Remark 2 Algorithm 1 is described in this article as an analog of the CMC estimator $\hat{\xi}_{p,n}$, in that the binned empirical cdf is an approximation of the canonical empirical cdf. We expect that the same approach can be employed to approximate the empirical cdf constructed in conjunction with variance reduction techniques (for instance those described in Chu and Nakayama (2012)).

2.2.3 NUMERICAL RESULTS

This section tests the efficacy of the proposed low-storage estimator over a range of input values, focusing on the online aspect of quantile estimation. To recap, the standard estimator (Section 2.1) uses the $\lceil p \rceil$-th order statistic as the estimate for the p-quantile. In order to provide an online estimator for this value, the available set of observations $X_i, i = 1, 2, \dots$ needs to be constantly re-sorted. The proposed low-storage estimator (Section 2.2), on the other hand, only maintains a small data-structure to provide the estimate of the quantile.

Figure 1 establishes the statistical convergence properties of the two methods by plotting each method's mean squared error as a function of the number of collected observations. The standard estimator and the proposed low-storage estimator were both required to estimate the 0.95-quantile $\xi_p = 1.81246112281168$ of the Student's T distribution with 10 degrees of freedom. Under both cases, the mean squared error was calculated using 100 independent replications of the quantile estimation procedure.

As Figure 1 illustrates, the graph for the CMC estimator could be obtained only for a maximum $n = 1.75 \times 10^5$ due to storage and computational issues. The corresponding graph for the proposed estimator could be obtained for a maximum $n = 1 \times 10^8$. The tests were conducted on a desktop machine with 3GHz Intel Xeon processors and 4Gb of memory per core. Executing the CMC estimator for n up to 1.75×10^5 took about 30 minutes of computation time per replication. In comparison, for the proposed estimator, the computation time required for the entire set of 100 replications, each of which executed to $n = 1 \times 10^8$, was about an hour.

Figure 1 illustrates the $O(n^{-1})$ decay of the mean squared error, and the $O(\log n)$ storage requirement for the proposed estimator. As an example of the latter complexity, only 42 bins are required on average with $n = 1 \times 10^8$. Figure 2 plots a single sample-path of the number of bins required by the proposed

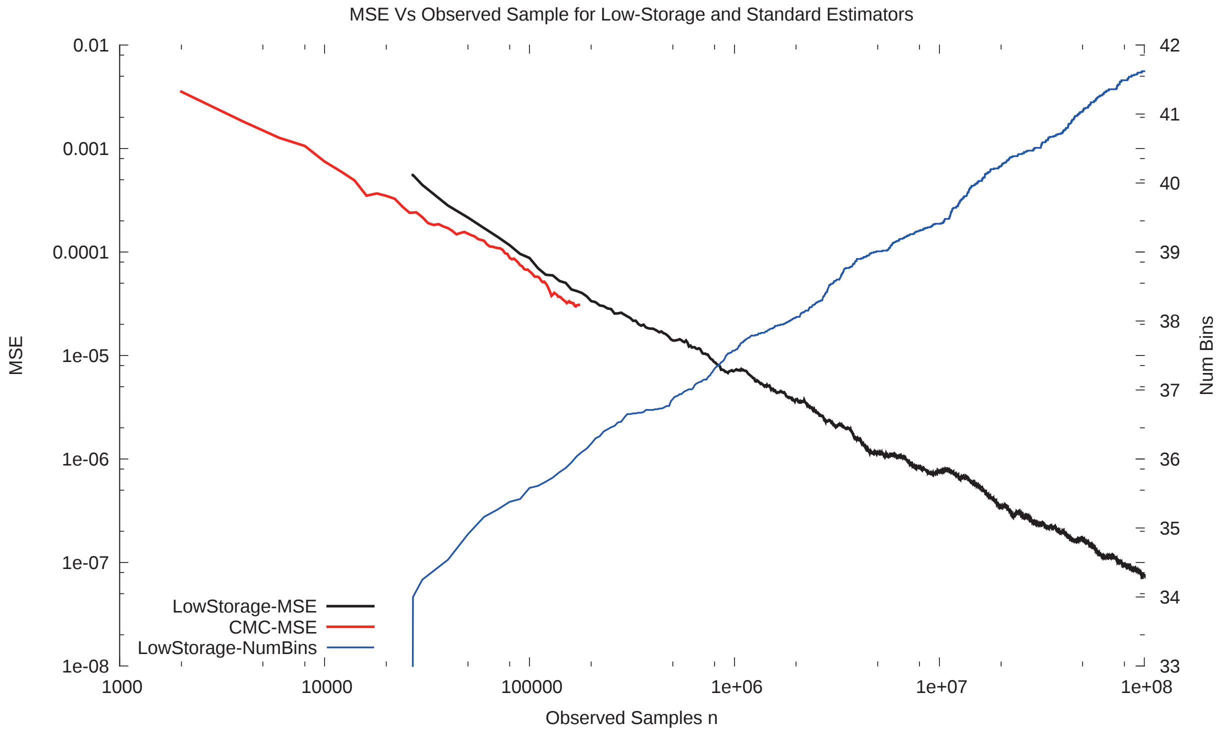

Figure 1: The left ordinate is the mean squared error plotted as a function of the number of observations. The mean squared error was calculated over 100 realizations of the traditional and the proposed quantile estimator. The quantile is for $p = 0.95$ for the Student's T distribution with 10 degrees of freedom. The right ordinate is the number of bins used by the proposed estimator as a function of the number of collected observations.

algorithm for a variety of distributions and p-values. Under all such combinations of settings, the method shows a robust $O(\log n)$ growth in the number of bins.

2.2.4 Estimating CVAR

The problem of estimating the quantile ξ_p underlies the estimation of some important performance measures. One such important measure is the *mean-excess* $\eta_p = E[X|X > \xi_p]$, also known as the *Conditional Value-at-Risk* (CVAR) in financial applications (Glasserman 2004). Estimating η_p from simulation outputs poses the same storage/memory issues as estimating ξ_p does, since one needs to simultaneously estimate the two quantities ξ_p, η_p. Our approach for a low-storage online estimator readily extends to estimating measures such as η_p. For the specific case of η_p, an additional set of data is recorded for each bin:

$$\tilde{\eta}_{l,n} = \sum_{i=1}^{n} X_i I(X_i > a_l), \qquad \forall a_l \in \mathscr{P}_{k(n)}.$$

If $l(n)$ is the index of the bin that contains the quantile estimate $\tilde{\xi}_{p,n}$, then the mean-excess η_p can be estimated as either of the values $\tilde{\eta}_{l(n)-1,n}$ or $\tilde{\eta}_{l(n),n}$ or a linear interpolation of both. In the CVAR context, we expect similar storage and computational gains as the proposed quantile estimator.

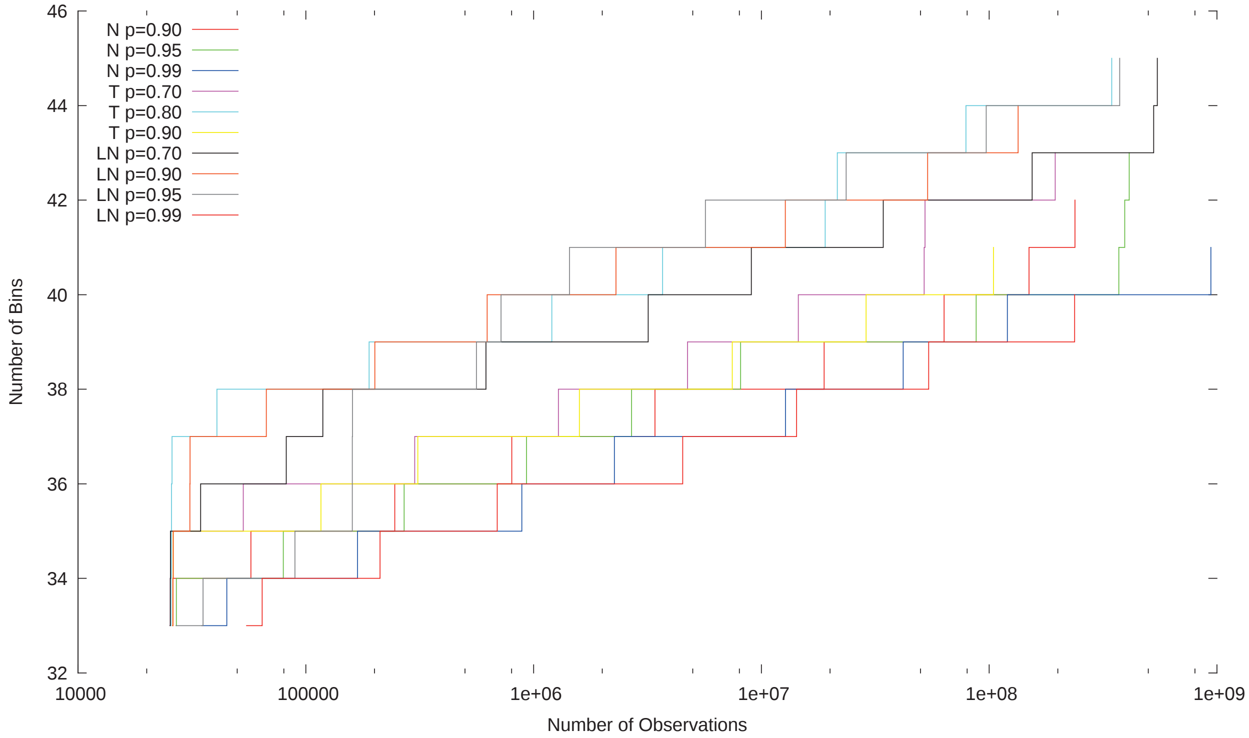

Figure 2: The curves plot the number of bins used by the proposed estimator as a function of the number of collected observations (plotted in log-scale). Three distributions were tested for $p = 0.7, 0.9, 0.99$; the distributions were $N(0,1)$, LogNormal (mean = 1 and variance = 20) and the Student's T distribution with 10 degrees of freedom. Note that only a single sample path is plotted for each setting.

3 DENSITY ESTIMATION

A context analogous to quantile estimation of a random variable X is *density estimation.* Here, the loosely stated question is to estimate the density function $f(\cdot)$ of the random variable X (assuming it exists), given only iid observations from the distribution of X. This problem has been studied exhaustively over the last several decades. (See Silverman (1986), Hall and Wand (1996) for an account and for key entry points into the literature.) One of the dominant methods for solving this problem is what has traditionally been called *kernel density estimation* (henceforth KDE). KDE methods are easily understood. After obtaining n iid random variates distributed according to the density f, construct the estimator of f as an n-equiprobable mixture of kernels "placed" on each of the sampled variates $X_i, i = 1, 2, \dots, n$. Formally, the KDE estimator is given as

$$\hat{f}(x) := n^{-1} \sum_{i=1}^{n} \frac{1}{h} K(\frac{x - X_i}{h}), \tag{4}$$

where $K(\cdot)$ is the chosen kernel, and $h(n)$ is the chosen "bandwidth parameter" that controls the amount of "smoothing" present in the estimator. The multivariate analogue of (4) follows in a straightforward fashion.

The asymptotic and small-sample properties of the estimator in (4) have been studied extensively. For instance, when the kernel $K(\cdot)$ is infinitely differentiable, the bias and variance of the estimator $\hat{f}(\cdot)$ (in one dimension) are known to be

$$\mathbb{E}[\hat{f}(x)] - f(x) = \frac{1}{2} h^2(n) \left(\int_a^b u^2 K(u)\, du \right) f(2)(x) + o(h^2(n)) \tag{5}$$

and

$$\mathrm{Var}(\hat{f}(x)) = n^{-1}h^{-1}\int_a^b K^2(u)\,du + o(n^{-1}h^{-1}) \tag{6}$$

respectively. The expressions in (5) and (6) together determine the best achievable convergence rate for KDE estimators. Specifically, it can be easily shown that the fastest achievable convergence rate of the mean integrated squared error $\int_a^b \mathbb{E}\left[(\hat{f}(x)-f(x))^2\right]\,dx$ is $n^{-4/5}$ achieved when when the bandwidth parameter $h(n)$ is chosen as

$$h = \left\{\int_a^b u^2K(u)\,du\right\}^{-2/5}\left\{\int_a^b K^2(u)\,du\right\}^{1/5}\left\{\int_a^b (f^{(2)}(u))^2\,du\right\}^{-1/5} n^{-1/5}. \tag{7}$$

Analogous results are available for the multivariate contexts, and for settings where rough kernels $K(\cdot)$ are used in the construction of the estimator $\hat{f}(\cdot)$.

In this section, our objective is somewhat modest. Following our treatment of online estimators for quantiles, we wish to illustrate corresponding estimators for the context of estimating densities. Unlike the quantile context, however, the theory for online density estimators follows in a rather straightforward fashion from the results in Hall and Wand (1996). Accordingly, our treatment is cursory and limited to one dimension.

3.1 An Online Density Estimator

To motivate online estimators for the context of density estimation, a crucial fact about the estimator in (4) is noteworthy. The estimator $\hat{f}$ requires $O(n)$ storage and computation since it involves the manipulation of all n observed random variates $X_1, X_2, \ldots, X_n$. Our interest is an alternate "online density estimator" that retains the $n^{-4/5}$ convergence rate of the traditional KDE estimator; however, in the same sense as the online quantile estimator, we seek an online density estimator that requires much less storage and computation. We accomplish this through two strategies: (i) binning, where instead of storing every observation $X_i, i = 1, 2, \ldots, n$, we accumulate frequencies of the observed random variates over a prespecified grid; (ii) choosing the grid in (i) in such a way as to ensure that the increased bias due to binning does not degrade the convergence rate of the resulting binned estimator. Towards detailing this estimator in more rigorous terms, let

$$\mathscr{P}(n) := \left\{a, a+\delta(n), a+2\delta(n), \ldots, a+\left\lfloor\frac{b-a}{\delta(n)}\right\rfloor\delta(n)\right\}$$

denote a partition of the interval $[a,b]$, and $\delta(n) \in (0,\infty)$ a chosen "window width" expressed as a function of the number of observed random variates n. (Recall that $[a,b]$ is the region where the density f is to be estimated.) Also, let $\alpha_j(n), j \in \{1, \ldots, \lfloor\frac{b-a}{\delta(n)}\rfloor\}$ denote the fraction of random variates that lie closest to $a + j\delta(n)$, that is, $\alpha_j = n^{-1}\sum_{i=1}^n \mathbb{I}\{[X_i] = a + j\delta(n)\}$, where $[z]$ is the element in $\mathscr{P}(n)$ closest to z. Then, the *online density estimator* $\tilde{f}(x)$ is given as

$$\tilde{f}(x) := \sum_{y\in\mathscr{P}(n)} \alpha_j(n)\frac{1}{h}K(\frac{x-y}{h}). \tag{8}$$

It is easily seen that the storage and computation for $\tilde{f}$ is $O(\delta(n))$ as opposed to $O(n)$. To get a sense of the mean integrated squared error of $\tilde{f}(x)$, we appeal to the results in Hall and Wand (1996). When the chosen kernel $K(\cdot)$ is an infinitely differentiable probability density (e.g., $K(\cdot)$ is a Gaussian or Student's T distribution), it can be shown that the integrated squared bias of $\tilde{f}(\cdot)$ is given as

$$\int_a^b \left(\mathbb{E}\left[\hat{f}(x)\right]-f(x)\right)^2 dx \;=\; \left(\frac{1}{576}\delta^4(n)+\frac{1}{4}h^4(n)\left(\int_a^b u^2K(u)\,du\right)\right)\left(\int_a^b f^{(2)}\,dx\right)^2 + o(\delta^4(n)+h^4(n)) \tag{9}$$

as $h(n) \to 0$ and assuming $\delta(n) < h(n)$. Similarly, the integrated variance can be shown to be

$$\int_a^b \mathrm{Var}(\tilde{f}(x))\,dx = n^{-1}h^{-1}(n)\left(\int_a^b K^2(u)\,du\right) + o\left(n^{-1}h^{-1}(n)\right) \tag{10}$$

when $\delta(n) < h(n)$.

Two observations about (9) and (10) are noteworthy. First, the window width $\delta(n)$ has no effect on the integrated variance, and only an additive effect on the integrated squared bias. Second, the integrated mean-squared error of the estimator $\tilde{f}(\cdot)$ can be estimated by neglecting the $o(\cdot)$ terms and substituting $f(\cdot)$ with $\tilde{f}(\cdot)$ in the expressions (9) and (10). These observations inspire an online estimation algorithm that has a similar flavor to what was outlined for the case of estimating quantiles. The algorithm is characterized by three repeating steps:

S.1 Obtain random variate(s).

S.2 Estimate the integrated squared bias and variance of the estimator $\tilde{f}(x)$ by neglecting $o(\cdot)$ terms, and substituting $\tilde{f}(x)$ for $f(x)$, in (9) and (10).

S.3 If the estimated integrated squared bias is less than the integrated variance, then reduce the window width $\delta(n)$ and the bandwidth $h(n)$ by a constant factor and then go back to S.1. Otherwise, simply go back to S.1.

As in the context of online quantile estimators, the step S.2 plays the crucial role of keeping the integrated squared bias and variance in lock-step, thereby ensuring that the integrated mean-squared error of $\tilde{f}(\cdot)$ decreases at the fastest rate possible. The window size $\delta(n)$ and the bandwidth $h(n)$ are kept in lock-step to ensure that the expressions in (9) and (10) are valid. Furthermore, the window size $\delta(n)$ and the bandwidth $h(n)$ are changed only infrequently, whenever the integrated variance "catches up" with the integrated bias. While such infrequent updates cause inefficiencies, they reduce total computation involved in estimating $\tilde{f}(x)$ by changing the partition $\mathscr{P}(n)$ only infrequently.

It is easy to guess the rates at which $\delta(n)$ and $h(n)$ reduce to zero, and the corresponding rate at which the integrated mean-squared error of $\tilde{f}$ tends to zero, under the proposed scheme. For instance, since the integrated squared bias and variance are effectively equated, the expressions in (9) and (10) yield that $h(n) \approx \delta(n) \approx n^{-1/5}$. Plugging this back in (9) and (10), one finds since the integrated mean squared error of $\tilde{f}$ is $O(n^{-4/5})$. This is the same as traditional KDE under similar conditions, but with only fifth root of the corresponding storage.

REFERENCES

Avramidis, A. N., and J. R. Wilson. 1995. "Correlation-induction techniques for estimating quantiles in simulation experiments". In *Proceedings of the 27th conference on Winter simulation*, WSC '95, 268–277. Washington, DC, USA: IEEE Computer Society.

Bayer, R. 1972. "Symmetric binary B-Trees: Data structure and maintenance algorithms". *Acta Informatica* 1 (4): 290–306.

Bekki, J. M., J. W. Fowler, G. T. Mackulak, and B. L. Nelson. 2007. "Using quantiles in ranking and selection procedures". In *Winter Simulation Conference*, edited by S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 1722–1728: WSC.

Chu, F., and M. K. Nakayama. 2012. "Confidence intervals for quantiles when applying variance-reduction techniques". *ACM Trans. Model. Comput. Simul.* 22 (2): 10.

Duffie, D., and J. Pan. 1997, Spring. "An Overview of Value at Risk". *The Journal of Derivatives* 4 (3): 7–49.

Ghosh, S., and S. Ghosh. 2012. "A strong law for the rate of growth of long latency periods in a cloud computing service.". *Adv. Appl. Probab.* 44 (4): 995–1017.

Glasserman, P. 2004. *Monte Carlo Methods in Financial Engineering*. New York, NY.: Springer.

Glynn, P. 1996. “Importance sampling for Monte Carlo estimation of quantiles”. In *Stochastic Simulation and Experimental Design: Proceedings of the 2nd St. Petersburg Workshop on Simulation*, 180–185: Publishing House of St. Petersburg University.

Goldenberg, D. K., L. Qiuy, H. Xie, Y. R. Yang, and Y. Zhang. 2004. “Optimizing cost and performance for multihoming”. In *ACM SIGCOMM Computer Communication Review*, Volume 34, 79–92. ACM.

Hall, P., and M. P. Wand. 1996. “On the accuracy of binned kernel density estimators”. *Journal of Multivariate Analysis* 56:165–184.

Hsu, J. C., and B. L. Nelson. 1990, July. “Control variates for quantile estimation”. *Manage. Sci.* 36 (7): 835–851.

Nakayama, M. K. 2012. “Using sectioning to construct confidence intervals for quantiles when applying importance sampling”. In *Winter Simulation Conference*, edited by O. Rose and A. M. Uhrmacher, 9: WSC.

Pasupathy, R., R. Szechtman, and E. Yücesan. 2010. “Selecting small quantiles”. In *Winter Simulation Conference*, 2762–2770: WSC.

Serfling, R. J. 1980. *Approximation Theorems of Mathematical Statistics*. New York, NY.: John Wiley & Sons, Inc.

Silverman, B. W. 1986. *Density Estimation for Statistics and Data Analysis*. New York, NY.: Chapman and Hall.

AUTHOR BIOGRAPHIES

RAGHU PASUPATHY is an Associate Professor in the Grado Department of Industrial and Systems Engineering at Virginia Tech. His research interests lie broadly in Monte Carlo methods with a specific focus on simulation optimization. He is a member of INFORMS, IIE, and ASA, and serves as Department Editor for the simulation desk at IIE Transactions and Associate Editor for *Operations Research*, *ACM TOMACS* and *INFORMS Journal on Computing*. His email address is pasupath@vt.edu and his web page is https://filebox.vt.edu/users/pasupath/pasupath.htm.

SOUMYADIP GHOSH is a Research Staff Member in the Business Analytics and Mathematical Sciences division at the IBM TJ Watson Research Center. His current research interests lie in simulation based optimization techniques for stochastic optimization problems, with a focus on applications in Energy and Power systems and supply chain management. His email is ghoshs@us.ibm.com and his web page is at https://researcher.ibm.com/researcher/view.php?person=us-ghoshs.