How to estimate quantiles easily and reliably

Keith Briggs, BT Technology, Services and Operations[*]
Fabian Ying, Mathematical Institute, University of Oxford[†]

Last modified 2017-01-13 10:55

1 Abstract

We describe a practical recipe for determining when to stop a simulation which is intended to estimate quantiles of the unknown distribution of a real-valued random variable. The ability to reliably estimate quantiles translates into an ability to estimate cumulative distribution functions. The description focusses on the practical implementation of the method, and in particular on stopping criteria. Both authors have found the method useful in their day-to-day work.

2 Introduction

Much work in applied mathematics nowadays makes use of computer simulations, and very often these simulations are set up to estimate the distribution of a random variable. If the probability density function (pdf) is required, the usual technique is to construct a histogram. But this has disadvantages, at least for the usual direct method: lower and uppers limits for the histogram bins, as well as their width, have to be decided before the simulation is started. Thus one might instead choose to estimate the cumulative distribution function (cdf). For either pdf or cdf estimation, a stopping criterion will be needed, in order to determine how many samples of the random variable need to be taken before some pre-determined accuracy is met. This brings us to the point of this short article: we argue that cdf estimation is easier to set up, and there is a much simpler way to apply a stopping criterion than for histograms; and moreover it is one which is independent of the unknown distribution being estimated. We suggest also that for many applications, such as estimating tail probabilities, knowledge of the cdf rather than the pdf is more directly useful. Nothing presented here is entirely novel, but the technique does not seem to be as widely known as it deserves to be, and it is perhaps not appreciated that this method can easily be applied as a wrapper around existing simulation code.

3 Quantile estimation via simulation

Our method for cdf estimation depends on estimation of a number of quantiles, the number depending on how precisely the knowledge of the cdf is needed. We set up the theory in a way that applies to both continuous and discrete random variables, though our main interest is the continuous case. Our notation and terminology is generally consistent with that of Grimmett and Stirzaker (1992). The p-quantile ξ_p of a random variable X is defined to be the smallest ξ_p such that Prob $[X \leqslant \xi_p] = p$. For

[*]keith.briggs@bt.com
[†]fabian.ying@maths.ox.ac.uk

example, sometimes just the median or 0.5-quantile may be needed; or, for risk estimation, a 0.95 or 0.99 quantile. But for a very precise estimation of the cdf, we might choose a range of equally spaced quantiles, such as i/n-quantiles for $n{=}100$ and $i{=}1,2,\ldots,99$. We assume that in all cases a "blackbox" function f is available which returns a sample from the unknown distribution. Successive calls of the function are assumed to return independent and identically distributed samples. Even better (as presented in the pseudocode below), is to arrange that the sampler f can return a batch of samples of specified size. There will typically be a trade-off here; larger batches may be more efficient to generate, but the stopping criterion will then be applied less often, and we may end up using more samples than strictly necessary,

To estimate the quantile ξ_p, we take samples $X_1,X_2,\ldots,X_n$ of X, and then sort into increasing order as $X_{(1)}\leqslant X_{(2)}\leqslant\ldots\leqslant X_{(n)}$. Then our estimator of ξ_p is $\hat{\xi}_p\equiv X_{\lceil np\rceil}$. There are two main practical difficulties: (1) all data will be stored, so the storage needed is potentially large, but also sorting becomes progressively slower as the set of samples becomes larger; (2) we need a stopping criterion in order to minimize the amount of computation to achieve a pre-specified accuracy (say, a 95% or 99% confidence interval for quantile of sufficiently narrow width). The first difficulty is not serious on modern computers; we might typically need to store a maximum of about 10^6 samples, and fast sorting algorithms are commonplace. If reduction of storage is required, the alternative method of Ghosh and Pasupathy (2013) can be used.

For the second difficulty, we use the following result on distribution-free confidence intervals to decide when to stop acquiring samples (David 1970, p. 13). Let X be the random variable of interest, and let $X_{(1)}\leqslant\ldots\leqslant X_{(n)}$ be our sorted samples. Then for $r<s$,

$$\text{Prob}\left[X_{(r)}\leqslant\xi_p<X_{(s)}\right]=\sum_{i=r}^{s-1}\binom{n}{i}p^i(1-p)^{n-i}.$$

To see this, consider

$$\text{Prob}\left[X_{(r)}\leqslant\xi_p<X_{(s)}\right]=\text{Prob}\left[X_{(r)}\leqslant\xi_p\right]-\text{Prob}\left[X_{(s)}\leqslant\xi_p\right]. \tag{1}$$

So we need to find an expression for Prob $\left[X_{(k)}\leqslant\xi_p\right]$ for any k. $X_{(k)}\leqslant\xi_p$ is true precisely when at least k values in the sample are less than ξ_p. Each realization of X has independent and identical probability of being smaller than ξ_p given by

$$\text{Prob}\left[X\leqslant\xi_p\right]=p$$

by the definition of a quantile. Hence the number N_p of values in the sample that are at most ξ_p is binomially distributed, that is, $N_p\sim\text{Binomial}(n,p)$. Hence,

$$\begin{aligned}\text{Prob}\left[X_{(k)}\leqslant\xi_p\right]&=\text{Prob}\left[N_p\geqslant k\right] &(2)\\ &=\sum_{i=k}^{n}\binom{n}{i}p^i(1-p)^{n-i}. &(3)\end{aligned}$$

The result now follows by applying equation (3) to equation (1). Remarkably, this result is distribution-free; it does not depend on the distribution of X.

Algorithm 1 Quantile estimation algorithm

```
1: procedure QUANTILEESTIMATOR
2: Input: sampling function $f$, the required quantiles $p_1, \dots, p_m$, batch size $n_{\text{batch}}$, absolute error
   tolerance $\epsilon_{\text{abs}}$, relative error tolerance $\epsilon_{\text{rel}}$, confidence level $\alpha$
3: Initialize: S={}
4: Algorithm:
5:     repeat
6:         Generate a new sample of size $n_{\text{batch}}$ using $f$ and append to existing sample $S$
7:         Sort $S$ into increasing order as $X_1 \leqslant X_2 \leqslant \cdots \leqslant X_{|S|}$
8:         for $k=1,2,\dots,m$ do
9:             Find $r_k$, $s_k$ with $s_k - r_k$ minimal, subject to $\sum_{i=r_k}^{s_k-1} \binom{n}{i} p_k^i (1-p_k)^{n-i} > 1-\alpha$
10:    until $\max_k (X_{s_k} - X_{r_k}) < \epsilon_{\text{abs}}$ and $\max_k (X_{s_k} - X_{r_k}) < \epsilon_{\text{rel}} (X_{|S|} - X_1)$
11: Output: $(1-\alpha)$-confidence intervals $[X_{r_k}, X_{s_k}]$ and estimator $X_{\lfloor |S| p_k \rfloor}$ for quantile $p_k$ for $k =$
    $1, \dots, m$
```

Our basic stopping condition will require that the confidence interval for each quantile is no wider than some pre-specified width ϵ_{abs}. Suppose we want a $(1-\alpha)$-confidence interval on the p-quantile ξ_p from a sample of size n. Typically we might want 95% confidence intervals, so we take $\alpha = 0.05$. We then need to find indices r and s, with $1 \leqslant r < s \leqslant n$, and such that

$$\text{Prob}\,[r \leqslant N < s] = \sum_{i=r}^{s-1} \binom{n}{i} p^i (1-p)^{n-i} > 1-\alpha,$$

where $N \sim \text{Binomial}(n,p)$. This can be achieved by first setting r and s to the mode of the binomial distribution, which is either $\lfloor (n+1)p \rfloor$ or $\lceil (n+1)p \rceil - 1$, and then alternately decrementing r and incrementing s until the condition is met. If the resulting confidence interval is still too wide, the procedure can continue by taking yet more samples. This gives us an effective and reliable stopping criterion, and we thus have all the required ingredients for our method. A pseudocode for the algorithm (incorporating also a relative stopping condition) is given in Algorithm 1. In practice, we typically take samples in batches of size 1000, and after each batch we sort the full set of samples and apply the stopping criterion. When more than one quantile is being estimated simultaneously, we require all confidence intervals to be narrower than our pre-specified width.

We have presented the algorithm with a fully general relative and absolute error tolerance in the stopping condition, since in practice we may not have much *a priori* information about the range of the random variable being studied. This means that choosing an appropriate pre-specified width ϵ_{abs} of the confidence interval may not be straightforward. For example, a random variable which varies between 0 and 1000 will likely require a much larger ϵ_{abs} than a random variable between 0 and 1. To get around this, we can set ϵ_{abs} to some large value (effectively disabling the absolute error stopping criterion) and impose that confidence intervals are no wider than some pre-specified fraction ϵ_{rel} of the whole (estimated) range of X. The algorithm estimates the range of X by the difference between the maximal and minimal sampled values. A final nicety is that for general-purpose use, one would also want to add "bail-out" checks, which abort the algorithm if a specified maximum sample size is exceeded, or the running time exceeds a specified limit; such checks should catch cases where for some reason convergence is not being attained.

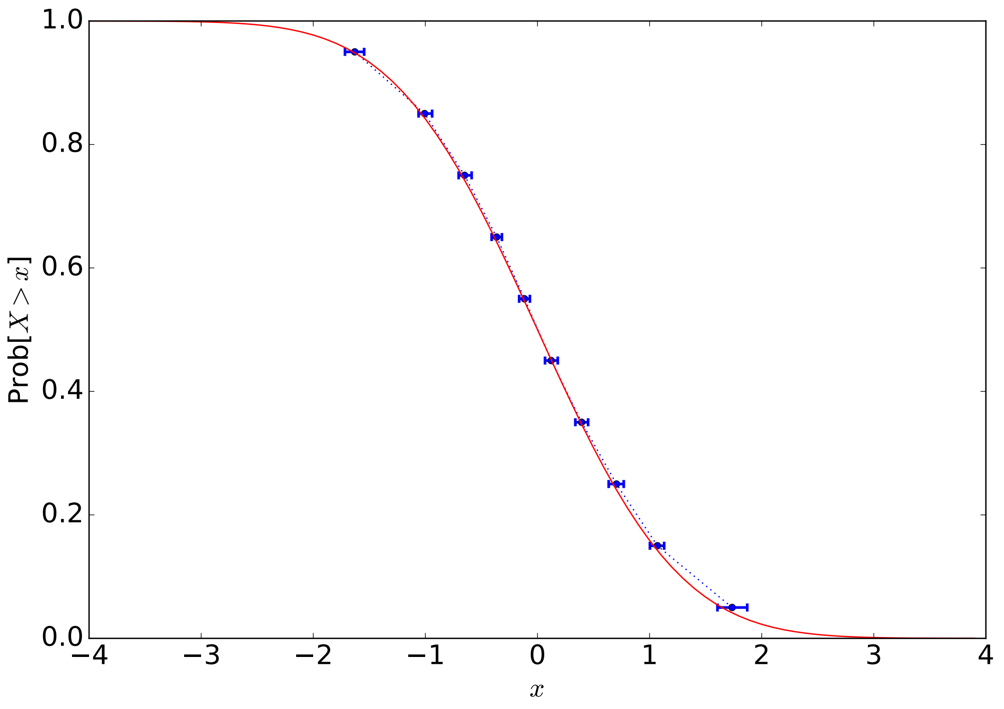

Figure 1: Estimated complementary cdf for a standard normal distribution $N(0,1)$, with 95% confidence intervals for the quantiles (blue rectangles, joined by a blue dotted line), compared with the exact result (which is $(1-\mathrm{erf}(x/\sqrt{2}))/2$) in red. In all cases the computed confidence intervals enclose the true value. 2000 samples were needed to achieve a relative accuracy of 5% (ϵ_{rel}=0.05).

As simple example, the estimated ccdf (complementary cdf, the probability that the random variable exceed a given value on the x-axis) for standard normal distribution is shown in Figure 1, with horizontal error bars showing the 95% confidence intervals. In the next sections we give three more examples: The first two are more complex real-world examples and the third one addresses the case when the our random variable is discrete.

4 Example 1: railway delays

It is known that train delays are very closely modelled by q-exponential random variables, which are a two-parameter family with pdf $f_{q,\beta}(t)=(1+(-\beta t+c)(q-1))^{1/(1-q)}$ (Briggs and Beck 2006). Here t is the delay in minutes of a scheduled departure time, and the parameters q and β vary depending on the route, but may be accurately estimated from data on delays in the past. This describes the distribution of delay on a single leg of a journey, but of course when a passenger transfers to a new train at a station, there is an additional delay while he or she waits for the connecting train (which might itself be delayed). We typically plot the ccdf, giving the probability for the total trip time to exceed a given

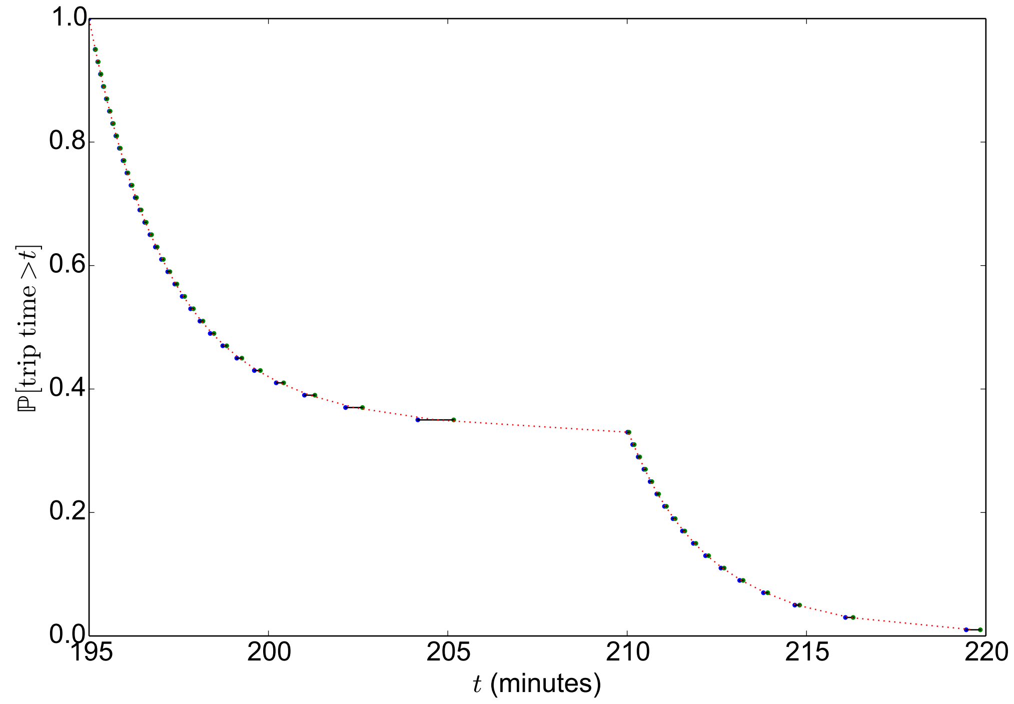

Figure 2: Estimated complementary cdf of total trip time for a hypothetical two-leg rail journey. The delays are governed by the q-exponential distribution with parameters $q{=}1.05$, $\beta{=}0.6$ for the first leg, and $q{=}1.1$, $\beta{=}0.5$ for the second leg. The probability of the trip taking more than 215 minutes is thus estimated to be about 7%. A relative accuracy of 1% was imposed (ϵ_{rel}=0.01), and 177000 samples were needed.

value on the time axis. Figure 2 shows the resulting overall ccdf for a hypothetical example. The passenger is assumed to start the first leg at time 0, and the first leg takes time 58 minutes plus any delay. The second leg departs the transfer station at times 60, 75, 90, $\ldots$, and takes 135 minutes plus the random delay. So if the delay on the first leg is less than two minutes, the train at time 60 is caught and the trip takes 195 minutes, plus second-leg delay. This gives the first part of the curve, to 210 minutes. But is the first-leg delay more than two minutes and less than 17 minutes, the first train at time 60 is missed and the train at time 75 is the one caught on the second leg. This gives the second part of the ccdf curve, to about 220 minutes.

5 Example 2: cellular wireless communication systems

The performance of a cellular wireless communication system is strongly determined by the distribution of signal-to-interference ratio (SIR) at the receiving devices (for example, laptops, mobile phones, or tablets). The SIR of a device is given by the ratio of the received signal power from the serving base station (for example, a cellular tower) to the total received interference power from all other base stations which transmit in the same frequency band. The SIR determines by Shannon's

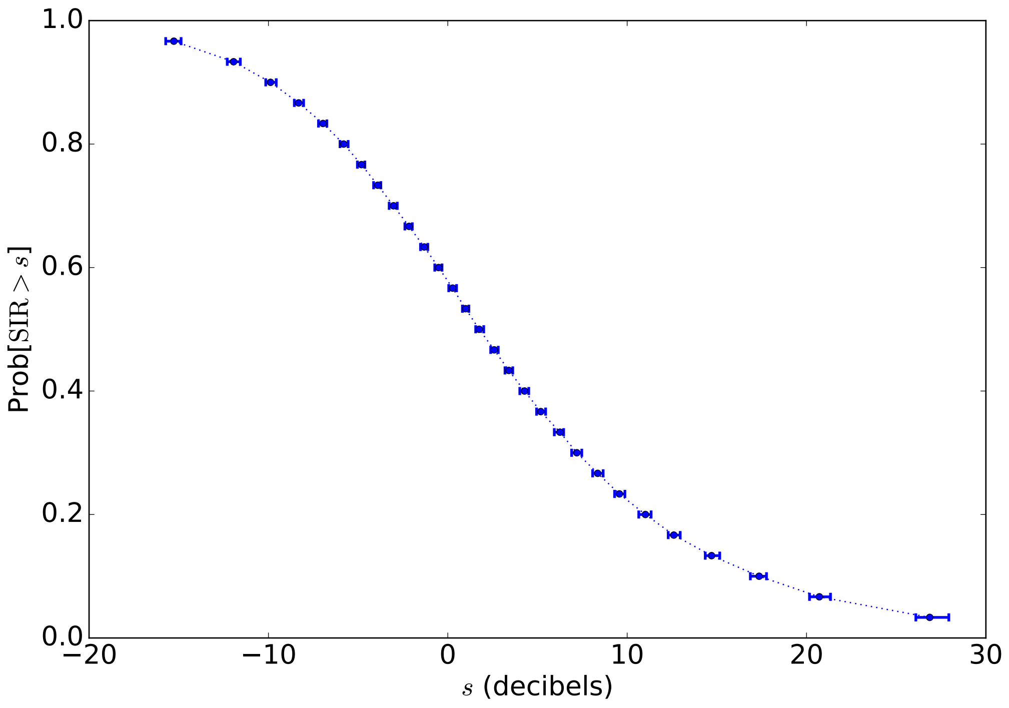

Figure 3: Estimated complementary cdf of signal-to-interference ratio (SIR) for a typical device in a home femtocell deployment, with 95% confidence intervals (the sample size was 7000). A decibel (dB) is a logarithmic ratio unit, here equal to $10 \log_{10}(\text{SIR})$. The computed confidence intervals are narrower at steeper sections of the cdf, as we can see when comparing the confidence intervals near 0 dB and near 25 dB. A relative accuracy of 1% was imposed (ϵ_{rel}=0.01).

Theorem the maximal achievable data-rate (in bits per second per unit of bandwidth) for a given device. The distribution of the SIR is typically visualized by the ccdf of SIR, giving the probability that a typical device achieves an SIR greater than a given value on the x-axis. For example, to find the performance of a home femtocell deployment (home femtocells are small, low-power 4G base stations placed in homes in a similar way to traditional wifi home hubs), one can perform simulations using openly available housing data and simplified propagation models to find the ccdf of the SIR. For a 2km by 2km suburban area, we compute the estimated distribution of SIR for a randomly placed device in a home femtocell deployment (see Figure 3). We assume that a femtocell is located at each house and devices connect to the nearest femtocell with all other femtocells as interferers. Connection to the network is considered possible for SIR greater than about -4 decibels, so in the example of Figure 3, only about 60% of the users can connect. Performance (in terms of bits of data transmitted per second), however, will be poor for negative SIR values. In network planning, one would aim to adjust parameters to push the SIR ccdf curve as far as possible to the right.

6 Example 3: Discrete random variables

So far, we have only considered estimating quantiles for continuous random variables, but the method applies to discrete random variables in exactly the same way. In this case the true cdf or ccdf is piece-wise constant. We compute some quantiles of a Poisson random variable in Figure 4. From the plot, we can see two main differences to the continuous case: firstly, the true ccdf has jumps, which are represented by several quantiles at the same x value. Secondly, the width of the confidence intervals are, at least in this case, precisely zero. In our case, this is due to the fact that we imposed $\epsilon_{\text{abs}}=0.5$. As described above, the simulation only stops when the lower and upper limit of each confidence interval is smaller than ϵ_{abs} and the limits of the confidence intervals are always values from the simulated sample. Since a Poisson random variable is integer-valued, the simulation will only stop when the lower and upper limits of the confidence intervals coincide, so the algorithm returns zero-width confidence intervals. If we were instead to set ϵ_{abs} so that $1 \leqslant \epsilon_{\text{abs}} < 2$, we would see confidence intervals of unit width or zero width. For most applications, it would arguably make more sense to estimate the probability mass function (pmf) instead of the cdf for discrete random variables, especially since the problems with the selection of histogram bins mentioned in the introduction do not apply for discrete random variables; each discrete value of the random variable represents its own bin. But should you wish to estimate the cdf instead of pmf, our method can be directly applied to discrete random variables with no added effort.

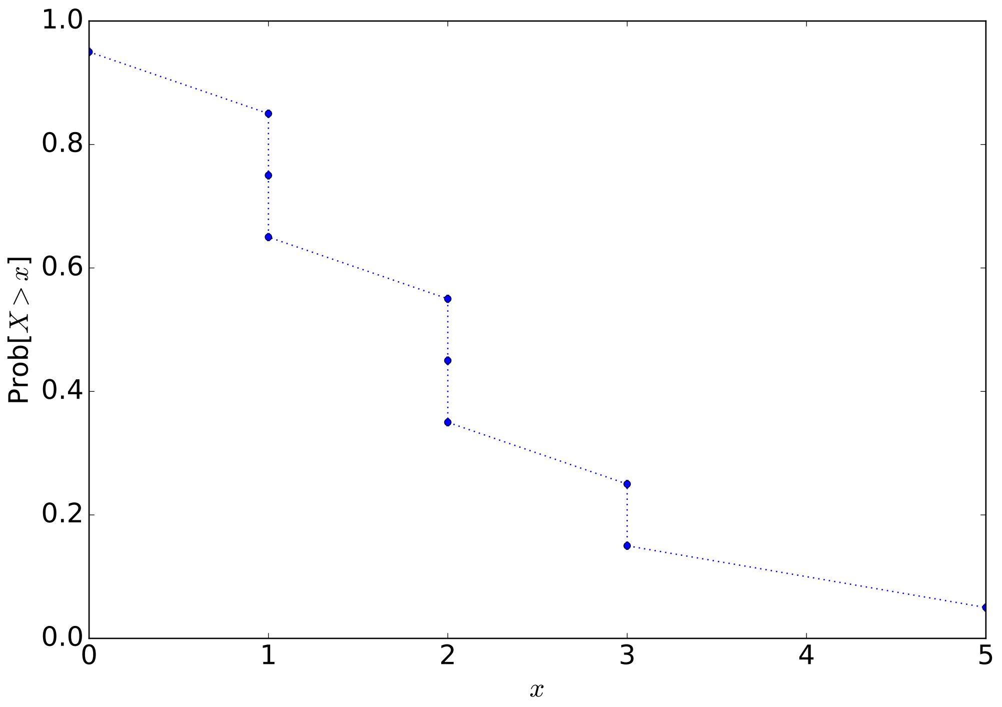

Figure 4: Estimated complementary cdf of a Poisson random variable with intensity $\lambda=2$ with 95% confidence intervals. A relative accuracy of 1% was imposed (ϵ_{rel}=0.01), and 15000 samples were needed.

7 Conclusion

Reliably estimating the distribution of an unknown random variable is useful in many different fields of science, and we have presented several examples above. In this note, we in particular address the practical question of “How many runs of the simulation should you do before your estimation is accurate enough?”. We have described a practical method for estimating quantiles (and therefore points on the cumulative distribution curve) for any unknown (discrete or continuous) random variable, giving a stopping criterion for simulations based on the desired accuracy for the quantiles. The strength of this method is that it requires only a black-box function that returns samples from the unknown distribution; the method does not require any other knowledge of the random variable. The method is therefore distribution-free and can be used as a wrapper around existing simulation code to estimate quantiles and hence the cdf of any real random variable. An implementation of the method in python is available at keithbriggs.info/software.html.

Acknowledgments The initial version of this technique was developed in collaboration with Jamie Fairbrother of Lancaster University in a project on radio spectrum management.

References

Briggs, Keith and Christian Beck (2006). “Modelling train delays with q-exponential functions”. In: *Physica A: Statistical Mechanics and its Applications* 378.2, pp. 498–504.

David, H. A. (1970). *Order Statistics*. New York: Wiley, (1970).

Ghosh, Soumyadip and Raghu Pasupathy (2013). ““Online” quantile and density estimators”. In: *Winter Simulation Conference (WSC), Washington*. (2013).

Grimmett, G. R. and D. R. Stirzaker (1992). *Probability and random processes*. 2nd ed. Oxford: Clarendon Press, (1992).