CrossMark

A new quantile tracking algorithm using a generalized exponentially weighted average of observations

Hugo Lewi Hammer[1] · **Anis Yazidi**[1] · **Håvard Rue**[2]

Abstract

The Exponentially Weighted Average (EWA) of observations is known to be a state-of-art estimator for tracking *expectations* of dynamically varying data stream distributions. However, how to devise an EWA estimator to track *quantiles* of data stream distributions is not obvious. In this paper, we present a lightweight quantile estimator using a generalized form of the EWA. To the best of our knowledge, this work represents the first reported quantile estimator of this form in the literature. An appealing property of the estimator is that the update step size is adjusted online *proportionally* to the difference between current observation and the current quantile estimate. Thus, if the estimator is off-track compared to the data stream, large steps will be taken to promptly get the estimator back on-track. The convergence of the estimator to the true quantile is proven using the theory of stochastic learning. Extensive experimental results using both synthetic and real-life data show that our estimator clearly outperforms legacy state-of-the-art quantile tracking estimators and achieves faster adaptivity in dynamic environments. The quantile estimator was further tested on real-life data where the objective is efficient in online control of indoor climate. We show that the estimator can be incorporated into a concept drift detector to efficiently decide when a machine learning model used to predict future indoor temperature should be retrained/updated.

Keywords Concept drift detection · Data stream · Generalized exponentially weighted average · Quantile tracking

1 Introduction

The volumes of automatically generated data are constantly increasing [27] and such data usually requires to be analyzed in real time [18]. Unfortunately, conventional statistical and data mining techniques are not applicable for such real time analysis [20]. Analysis of automatically generated data has been transitioning from being predominantly offline (or batch) to primarily online (or streaming) [18]. A wide range of streaming algorithms are continuously being developed pointed at real time analysis, like clustering, filtering, cardinality estimation, estimation of moments or quantiles, predictions and anomaly detection.

In this paper we consider the problem of estimating quantiles of streaming data. Streaming quantile estimation has been considered for a wide range of applications like portfolio risk measurement in the stock market [1, 15], fraud detection [40], signal processing and filtering [32], climate change monitoring [41], SLA violation monitoring [30, 31], network monitoring [7, 22], Monte Carlo simulation [36], structural health monitoring [16] and non-parametric statistical testing [21],

The first and the second moments of data, i.e. the mean and variance, are most commonly used as features in machine learning. However, as it is known in many real-life applications, these features might sometimes be misleading. Quantiles are better suited to extract the different aspects of data [11, 12]. A feature selection technique can further be applied to select the most appropriate quantile features such as done in [9, 29, 35].

Suppose that we are interested in estimating the quantile related to some probability q. The natural approach is to use the q quantile of the sample distribution. Unfortunately, such conventional approach has clear disadvantages for data streams as computation time and memory requirement are linear to the number of samples received so far from the

✉ Hugo Lewi Hammer
hugo.hammer@oslomet.no

1 OsloMet – Oslo Metropolitan University, Pilestredet 35, 0166 Oslo, Norway

2 King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

data stream. Such methods thus are infeasible for large data streams.

Several algorithms have been proposed to deal with those challenges. Most of the proposed methods fall under the category of what can be called histogram or batch based methods. The methods are based on efficiently maintaining a histogram estimate of the data stream distribution such that only a small storage footprint is required. Another ally of methods are the so-called incremental update methods. The methods are based on performing small updates of the quantile estimate every time a new sample is received from the data stream. Generally, the current estimate is a convex combination of the estimate at the previous time step and a quantity depending on the current observation. A thorough review of state-of-the-art streaming quantile estimation methods is given in the related work section (Section 2).

In data stream applications, a common situation is that the distribution of the samples from the data stream varies with time. Such system or environment is referred to as a dynamical system in the literature. Given a dynamical system, two main problems are considered namely to i) dynamically update estimates of quantiles of all data received from the stream so far or ii) estimate quantiles of the current distribution of the data stream (tracking). Incremental methods are well suited to address the tracking problem ii while histogram and batch methods mainly have been used to address problem i. Histogram and batch based methods are not well suited for the tracking problem ii and incremental methods typically are the only viable lightweight alternatives [4].

To address tracking problem ii, several incremental quantile estimators have been suggested [3–5, 22, 24, 34, 38]. The intuition behind the estimators is simple. If the received sample has a value below some threshold, e.g. the current quantile estimate, the estimate is decreased. Alternatively, whenever the received sample has a value above the same threshold, the estimate is increased. Even though the estimators document state-of-the-art tracking performance [38], neither of them use the values of the received samples directly to update the estimate, but only whether the value of the samples is above or below some varying threshold. Intuitively, this seems like a waste of information received from the data stream. In this paper, we thus present an estimator that uses the values of the received samples directly separating it from *all* incremental estimators suggested in the literature. The estimator is such that the update step size is *proportional* to the distance between the current estimate and the value of the sample. Thus if the current estimate is off-track compared to the data stream, the estimator will perform large jumps to rapidly get back on-track. A theoretical proof is provided to document the convergence properties of the estimator in addition to extensive simulation experiments. The experiments show that the estimator outperforms several other legacy state-of-the-art quantile tracking algorithms.

The EWA of observations is known to be state-of-the-art estimator to track expectations of dynamically varying data streams [14]. Interestingly, we will show that the suggested quantile estimator in this paper is in fact an instance of a *generalized* EWA such that *quantiles* and *not* expectations are tracked. To the best of our knowledge, this is the first EWA based quantile estimator found in the literature.

The paper is organized as follows. In Section 3, we present the novel quantile estimator using an EWA of observations. In Section 4, we present a quantile estimation algorithms based on the estimator in Section 3. In Section 5, we perform extensive experiments that document the superiority of the suggested algorithm. Finally, in Section 6 we apply the quantile estimator on real-life data related to the problem of efficient online control of indoor climate. More specifically, the estimator is used to detect when a machine learning model should be retrained/updated which is commonly referred to as concept drift detection [13].

2 Related work

In this Section, we shall review some of the related work on estimating quantiles from data streams. However, as we will explain later, these related works require some memory restrictions which renders our work radically distinct from them. In fact, our approach requires storing only one sample value in order to update the estimate. The most representative work for this type of "streaming" quantile estimator is due to the seminal work of Munro and Paterson [25]. In [25], Munro and Paterson described a *p*-pass algorithm for selection using $O(n^{1/(2p)})$ space for any $p \geq 2$. Cormode and Muthukrishnan [8] proposed a more space-efficient data structure, called the Count-Min sketch, which is inspired by Bloom filters, where one estimates the quantiles of a stream as the quantiles of a random sample of the input. The key idea is to maintain a random sample of an appropriate size to estimate the quantile, where the premise is to select a subset of elements whose quantile approximates the true quantile. From this perspective, the latter body of research requires a certain amount of memory that increases as the required accuracy of the estimator increases [37]. Furthermore, in the case where the underlying distribution changes over time, those methods suffer from large bias in the summary information since the stored data might be stale [4].

As Arandjelovic remarks [2], most quantile estimation algorithms are not single-pass algorithms and thus are not applicable for streaming data. On the other hand, the single

pass algorithms are concerned with the exact computation of the quantile and thus require a storage space of the order of the size of the data which is clearly an unfeasible condition in the context of big data stream. Thus, we submit that all work on quantile estimation using more than one pass, or storage of the same order of the size of the observations seen so far is not relevant in the context of this paper.

When it comes to memory efficient methods that require a small storage footprint, histogram based methods form an important class. A representative work in this perspective is due to Schmeiser and Deutsch [28]. In fact, they proposed to use equidistant bins where the boundaries are adjusted online. Arandjelovic et al. [2] used a different idea than equidistant bins by attempting to maintain bins in a manner that maximizes the entropy of the corresponding estimate of the historical data distribution. Thus, the bin boundaries are adjusted in an online manner. Nevertheless, histogram based methods have problems addressing the problem of tracking quantiles of the current data stream distribution [4] and are mainly used to recursively update quantiles for all data received so far. Finally, Lou et al. [23] perform extensive experiments to compare several histogram based algorithms.

Another group of methods are incremental quantile algorithms, which are particularly suitable to track quantiles of dynamically varying data stream distributions. In [3–6], the authors proposed modifications of the stochastic approximation algorithm [33]. While Tierney [33] uses a sample mean update from previous quantile estimates, [3–6] propose an exponential decay in the usage of old estimates making them able to track quantiles of non-stationary data stream distributions. Indeed, a "weighted" update scheme is applied to incrementally build local approximations of the distribution function in the neighborhood of the quantiles. More recent approaches in this direction is the Frugal algorithm by Ma et al. [24]. The RUMIQE and DUMIQE algorithms by Yazidi and Hammer [38] represents multiplicative updates compared to the additive updates of other incremental methods. A nice property of the RUMIQE and DUMIQE algorithms and the estimator suggested in this paper that the update size is automatically adjusted dependent on the scale/range of the data. This makes the estimators robust to substantial changes in the data stream. The DQTRE and DQTRSE algorithms by Tiwari and Pandey [34] aim to achieve the same by estimating the range of the data using peak and valley detectors. However, a disadvantage with these algorithms is that several tuning parameters are required to estimate the range of the data which renders the algorithms difficult to tune.

Table 1 Key notations with their meaning

Notation	Meaning
X_n	Possible outcomes from data stream at time n
f_n	Distribution of X_n
F_n	Cumulative distribution of X_n
$Q_n(q)$	Quantile of X_n associated with probability q
$\widehat{Q}_n(q)$	Estimate of $Q_n(q)$
μ^-	$E(X_n | X_n < \widehat{Q}_n(q))$
μ^+	$E(X_n | X_n > \widehat{Q}_n(q))$

3 Quantile estimator using a generalized exponentially weighted average of observations

Let X_n denote a stochastic variable representing the possible outcomes from a data stream at time n and let x_n denote a random sample (realization) of X_n. We assume that X_n is distributed according to some distribution $f_n(x)$ that varies dynamically over time n. We denote the cumulative distribution of X_n with $F_n(x)$, i.e. $P(X_n \leq x) = F_n(x)$. Further, let $Q_n(q)$ denote the quantile associated with probability q, i.e $P(X_n \leq Q_n(q)) = F_n(Q_n(q)) = q$. A summary of the most central notation is given in Table 1.

In 2017, Hammer and Yazidi suggested the DUMIQE algorithm [38] given by

$$
\begin{aligned}
\widetilde{Q}_{n+1}(q) &\leftarrow \widetilde{Q}_n(q) + \lambda q \widetilde{Q}_n(q) && \text{if } x_n > \widetilde{Q}_n(q) \\
\widetilde{Q}_{n+1}(q) &\leftarrow \widetilde{Q}_n(q) - \lambda(1-q)\widetilde{Q}_n(q) && \text{if } x_n \leq \widetilde{Q}_n(q)
\end{aligned}
\tag{1}
$$

which documents state-of-the-art tracking performance. However, a weakness of the DUMIQE and other proposed quantile tracking algorithms is that neither of them use the values of the received samples directly to update the estimate, but only whether the value of the samples is above or below some varying threshold. Intuitively, this seems like a waste of information received from the data stream. We now propose an incremental quantile estimator where the update step size is *proportional* to the distance between the received sample and current estimate. Thus, if the current estimate is off-track compared to the data stream, the estimator will initiate large jumps to rapidly get back on-track and thus more efficient tracking is expected. The suggested estimator is described formally as follows

$$
\begin{aligned}
\widehat{Q}_{n+1}(q) &\leftarrow \widehat{Q}_n(q) + \lambda c_n \frac{q}{\mu_n^+ - \widehat{Q}_n(q)} \left| x_n - \widehat{Q}_n(q) \right| && \text{if } x_n > \widehat{Q}_n(q) \\
\widehat{Q}_{n+1}(q) &\leftarrow \widehat{Q}_n(q) - \lambda c_n \frac{1-q}{\widehat{Q}_n(q) - \mu_n^-} \left| x_n - \widehat{Q}_n(q) \right| && \text{if } x_n \leq \widehat{Q}_n(q)
\end{aligned}
\tag{2}
$$

where $\mu^+ = E(X_n | X_n > \widehat{Q}_n(q))$ and $\mu^- = E(X_n | X_n < \widehat{Q}_n(q))$. Naturally, the conditional expectations satisfy the inequality

$$\mu^- < \widehat{Q}_n(q) < \mu^+$$

such that $\mu_n^+ - \widehat{Q}_n(q) > 0$ and $\widehat{Q}_n(q) - \mu_n^- > 0$. The factors $q/(\mu_n^+ - \widehat{Q}_n(q))$ and $(1-q)/(\widehat{Q}_n(q) - \mu_n^-)$ are included to ensure that the estimator converges to the true quantile value.

The constants c_n can be any sequence of positive and bounded values. The estimator performed well when the fractions in (2) were "normalized" as follows

$$c_n = \left(\frac{q}{\mu_n^+ - \widehat{Q}_n(q)} + \frac{1-q}{\widehat{Q}_n(q) - \mu_n^-} \right)^{-1} \qquad (3)$$

Substituting (3) into (2) we get

$$\begin{aligned} \widehat{Q}_{n+1}(q) &\leftarrow \widehat{Q}_n(q) + \lambda a_n \left| x_n - \widehat{Q}_n(q) \right| && \text{if } x_n > \widehat{Q}_n(q) \\ \widehat{Q}_{n+1}(q) &\leftarrow \widehat{Q}_n(q) - \lambda(1-a_n) \left| x_n - \widehat{Q}_n(q) \right| && \text{if } x_n \leq \widehat{Q}_n(q) \end{aligned} \qquad (4)$$

where

$$a_n = \frac{q}{\mu_n^+ - \widehat{Q}_n(q)} \bigg/ \left(\frac{q}{\mu_n^+ - \widehat{Q}_n(q)} + \frac{1-q}{\widehat{Q}_n(q) - \mu_n^-} \right) \qquad (5)$$

Please note that since $\mu_n^+ - \widehat{Q}_n(q) > 0$ and $\widehat{Q}_n(q) - \mu_n^- > 0$ we have that $0 < a_n < 1$. By factoring out $\widehat{Q}_n(q)$ and x_n we get

$$\begin{aligned} \widehat{Q}_{n+1}(q) &\leftarrow (1 - \lambda a_n)\widehat{Q}_n(q) + \lambda a_n x_n && \text{if } x_n > \widehat{Q}_n(q) \\ \widehat{Q}_{n+1}(q) &\leftarrow (1 - \lambda(1-a_n))\widehat{Q}_n(q) + \lambda(1-a_n)x_n && \text{if } x_n \leq \widehat{Q}_n(q) \end{aligned}$$

which can be written as

$$\widehat{Q}_{n+1}(q) \leftarrow (1 - b_n)\widehat{Q}_n(q) + b_n x_n \qquad (6)$$

where $b_n = \lambda \left(a_n + I \left(x_n \leq \widehat{Q}_n(q) \right) (1 - 2a_n) \right)$ and $I(A)$ the indicator function returning one (zero) if A is true (false).

Now we will present a theorem that catalogs the properties of the estimator $\widehat{Q}_n(q)$ for a stationary data stream, i.e. $X_n = X \sim F(x)$, $n = 1, 2, \ldots$.

Theorem 1 *Let $Q(q) = F^{-1}(q)$ be the true quantile to be estimated. Applying the updating rule in (6), we obtain:*

$$\lim_{n\lambda \to \infty, \lambda \to 0} \widehat{Q}_n(q) = Q(q)$$

The proof of the theorem can be found in Appendix A. Although the quantile estimator $\widehat{Q}_n(q)$ given in (6) is designed to track quantiles for dynamic environments, it is an important requirement that the estimator converges to the true quantile for static data streams as verified by Theorem 1.

We end this section with a remark.

Remark 1 If the conditional expectations are symmetrically positioned on each side of the quantile estimate, then $\widehat{Q}_n(q) - \mu_n^- = \widehat{Q}_n(q) - \mu_n^-$ and $a_n = q$ which is equal to DUMIQE. In other words, we can interpret that $\widehat{Q}_n(q) - \mu_n^-$ and $\widehat{Q}_n(q) - \mu_n^-$ ensure that the update rules take into account the asymmetries of the data stream distribution on each side of the quantile.

3.1 Connection to the EWA

A simple and intuitive approach to track the expectation of a data stream distribution, i.e. $\mu_n = E(X_n)$, is the weighted moving average

$$\widehat{\mu}_n = \frac{1}{W_n} \sum_{i=0}^{n} w_i x_i \qquad (7)$$

where $W_n = \sum_{j=1}^{n} w_j$. Using $w_{n-h} = \cdots = w_n = 1$ and the other weights equal to zero, (7) reduces to the standard moving average. Intuitively, it seems more reasonable to use weights with decreasing values. The decrease should be more rapid than the standard sample mean $w_i = 1/i$ to be able to track the changes in the data stream.

Consider the following recursive update scheme

$$\widehat{\mu}_0 \leftarrow x_0 \qquad (8)$$

$$\widehat{\mu}_{n+1} \leftarrow (1-\alpha)\widehat{\mu}_n + \alpha x_n \qquad (9)$$

where the current estimate is a convex combination of the estimate at the previous time step and the observation. By substitution, we get

$$\widehat{\mu}_{n+1} = \alpha(x_n + (1-\alpha)x_{n-1} + (1-\alpha)^2 x_{n-2} + \cdots + (1-\alpha)^{n-1} x_1) + (1-\alpha)^n x_0 \qquad (10)$$

Interestingly, from (10) we see that (8)-(9) can be interpreted as an EWA of observations. The estimator is highly popular and known to be the state-of-the-art approach to track expectations of dynamically varying data streams. Inspecting the incremental update form of our quantile estimator in (6), we see that it is identical to the update form of (9), except that the $0 < b_n < 1$ varies with time. Thus by keeping the weights constant as in (9), the estimator will track the expectation of the data stream distribution, while using the weights $0 < b_n < 1$ in (6), the estimator will track a quantile of the distribution.

4 Quantile estimation algorithm

The interpretation of the update rule in (6) as an EWA of observations (recall Section 3.1) and Theorem 1 constitute some intriguing theoretical results on the link between EWA and quantile estimation. However, the update rule in (6)

cannot be used directly since the conditional expectations, μ_n^+ and μ_n^-, are unknown and need to be estimated. Probably the most natural approach is to track conditional expectations using an EWA of observations as given in (8)-(9). This results in the following update rules:

- $\widehat{Q}_{n+1}(q) \leftarrow (1 - \widehat{b}_n)\widehat{Q}_n(q) + \widehat{b}_n x_n$ (11)

- If $x_n > \widehat{Q}_n(q)$
  - $\widehat{\mu}_{n+1}^+ \leftarrow \widehat{Q}_{n+1}(q) - \widehat{Q}_n(q) + (1 - \gamma)\widehat{\mu}_n^+ + \gamma x_n$ (12)
  - $\widehat{\mu}_{n+1}^- \leftarrow \widehat{Q}_{n+1}(q) - \widehat{Q}_n(q) + \widehat{\mu}_n^-$ (13)
- Else
  - $\widehat{\mu}_{n+1}^+ \leftarrow \widehat{Q}_{n+1}(q) - \widehat{Q}_n(q) + \widehat{\mu}_n^+$ (14)
  - $\widehat{\mu}_{n+1}^- \leftarrow \widehat{Q}_{n+1}(q) - \widehat{Q}_n(q) + (1 - \gamma)\widehat{\mu}_n^- + \gamma x_n$ (15)

- $\widehat{a}_{n+1} \leftarrow \dfrac{q}{\widehat{\mu}_{n+1}^+ - \widehat{Q}_{n+1}(q)} \left/ \left(\dfrac{q}{\widehat{\mu}_{n+1}^+ - \widehat{Q}_{n+1}(q)} \right. \right.$
$$\left. + \dfrac{1 - q}{\widehat{Q}_{n+1}(q) - \widehat{\mu}_{n+1}^-} \right) \quad (16)$$

- $\widehat{b}_{n+1} \leftarrow \lambda \left(\widehat{a}_{n+1} + I\left(x_n \le \widehat{Q}_{n+1}(q) \right) (1 - 2\widehat{a}_{n+1}) \right)$ (17)

In each of the (12)-(15), the part $\widehat{Q}_{n+1}(q) - \widehat{Q}_n(q)$ is included to ensure that the conditional expectation estimates are relative to the current quantile estimate $\widehat{Q}_{n+1}(q)$.

Thus (11) tracks the overall trends of the dynamical data stream while (12)-(15) are responsible for estimating the

conditional expectations *relative* to the quantile estimate. Thus, for most dynamic data streams it is reasonable to use a value of the EWA tuning parameter, γ, that is on a smaller scale than λ [19]. This is verified in our experiments. In the rest of the paper, we denote this EWA quantile estimator approach for QEWA. We end this section with a remark.

Remark1: We evaluated a second approach based on estimating the streaming distribution, $f_n(x)$, and computing the unknown conditional expectations from the estimated distribution. The streaming distribution were estimated by tracking several quantiles $Q_n(q_1), Q_n(q_2), \ldots, \ldots, Q_n(q_K)$ and a linear spline were interpolated between the quantile estimates. However experiments showed that the QEWA approach performed better than this spline approach. The spline approach therefore is not followed any further in the paper.

5 Experiments based on synthetic data

In this section we perform a thorough comparison of the performance of the suggested algorithm QEWA and other quantile estimators in the literature. Figure 1 shows tracking of the quantile with probability $q = 0.7$ for the suggested algorithm QEWA and DUMIQE. The true quantile is given as the dashed black line. The tuning parameters are adjusted

Fig. 1 Quantile estimates in every iteration using the DUMIQE and the suggested algorithm QEWA using ratio $\gamma/\lambda = 1/100$
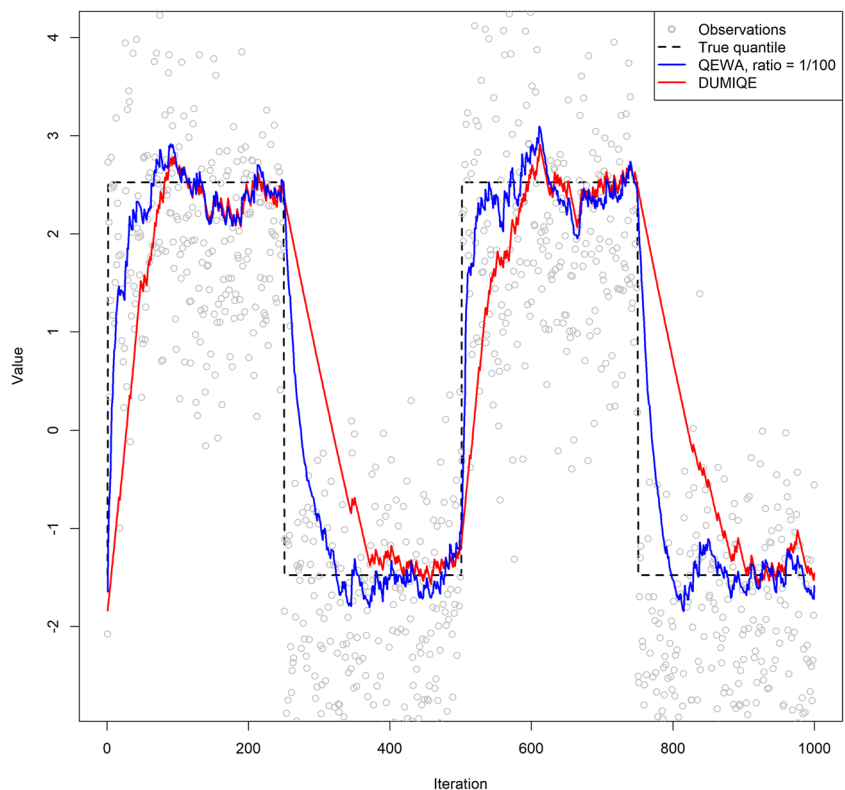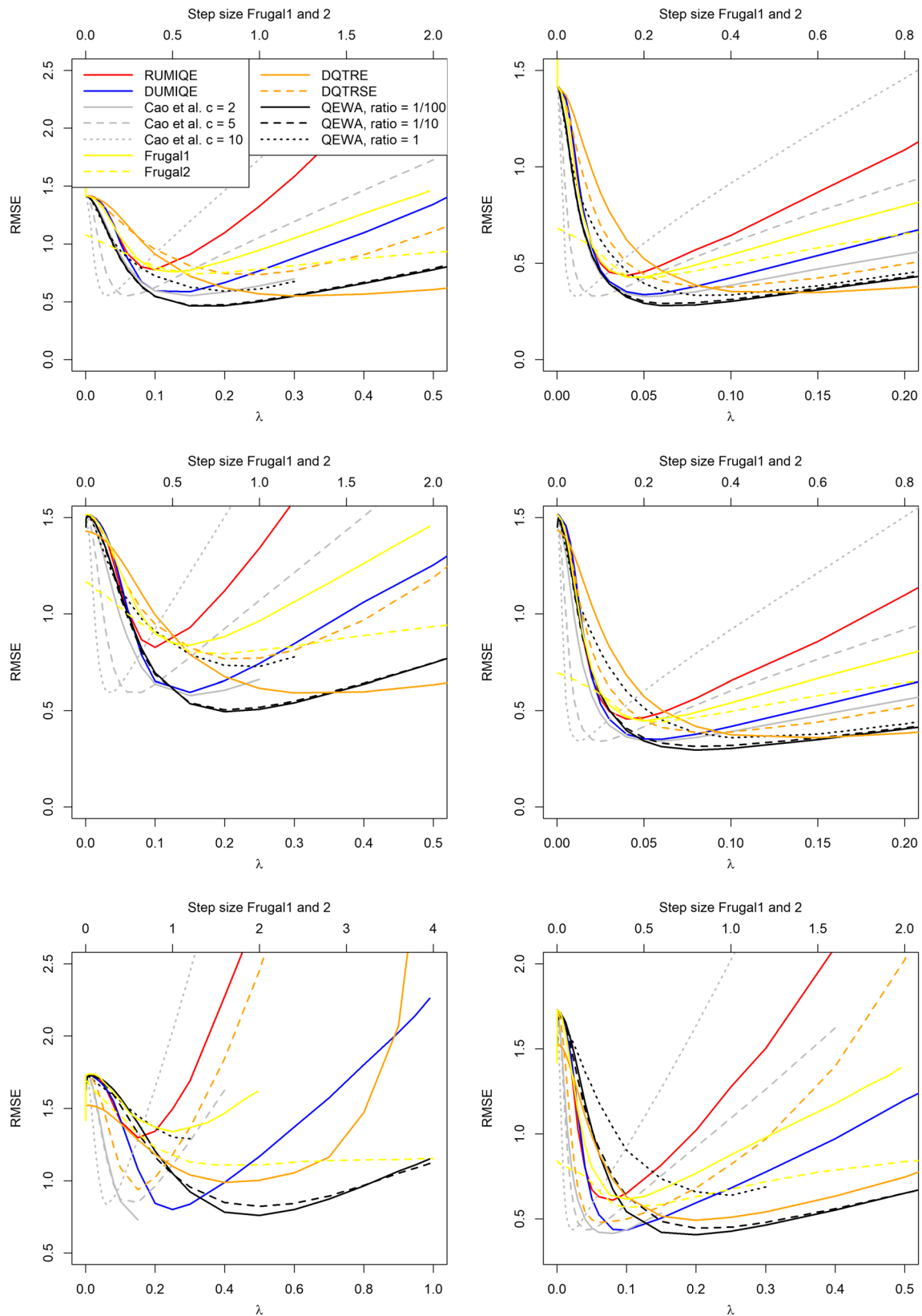
Fig. 2 Normal distribution smooth case: The left and right columns show results for $T = 100$ and $T = 500$, respectively. The rows from top to bottom show results when estimating quantile $Q_n(q = 0.5)$, $Q_n(q = 0.7)$ and $Q_n(q = 0.9)$, respectively. Ratio refers to the ratio between the tuning parameters, i.e. ratio = γ / λ. The upper x axis refers to the step size in the Frugal algorithms
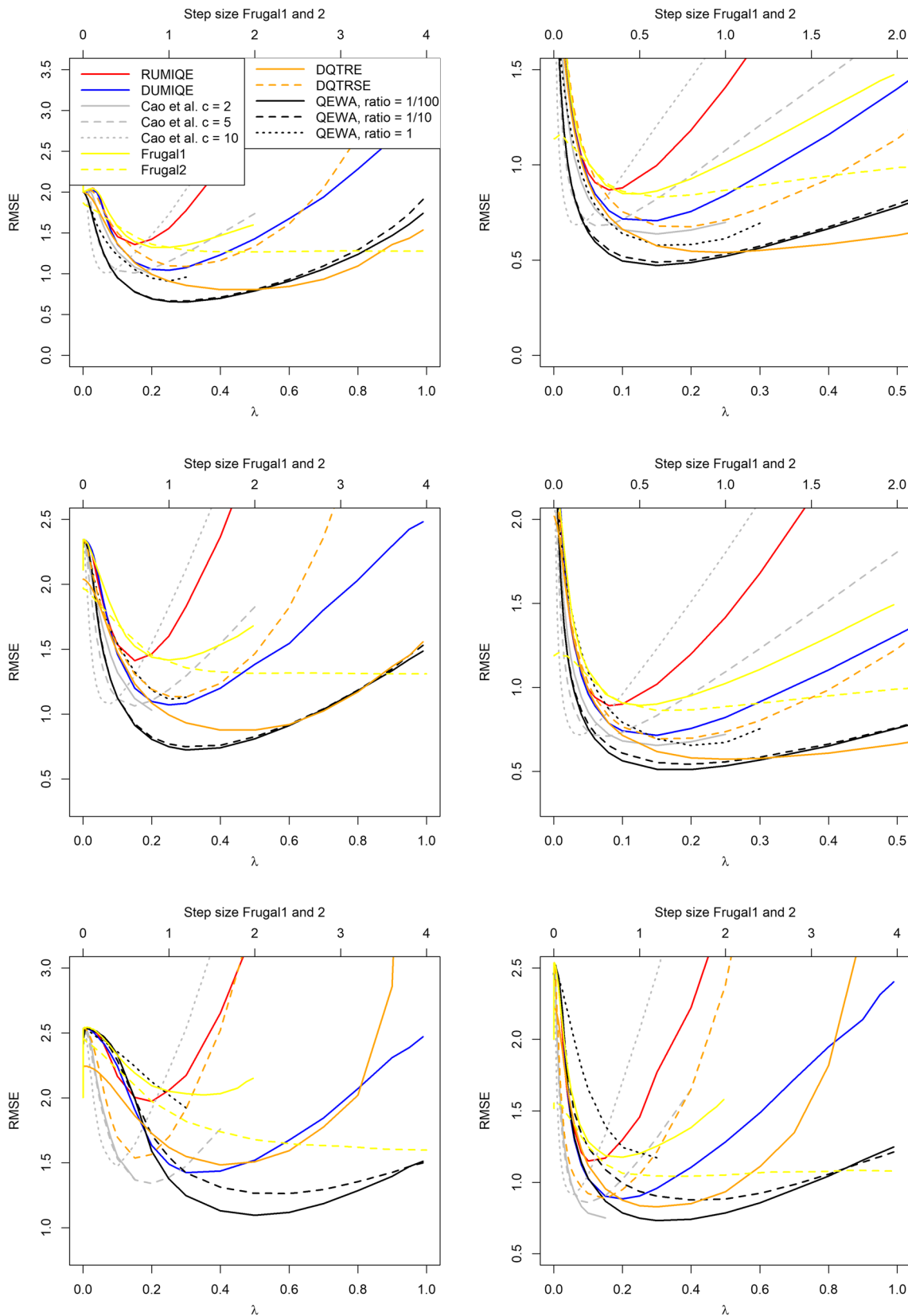
Fig. 3 Normal distribution switch case: The left and right columns show results for $T = 100$ and $T = 500$, respectively. The rows from top to bottom show results when estimating quantile $Q_n(q = 0.5)$, $Q_n(q = 0.7)$ and $Q_n(q = 0.9)$, respectively. Ratio refers to the ratio between the tuning parameters, i.e. ratio $= \gamma/\lambda$. The upper x axis refers to the step size in the Frugal algorithms

Fig. 4 χ^2 distribution smooth case: The left and right columns show results for $T = 100$ and $T = 500$, respectively. The rows from top to bottom show results when estimating quantile $Q_n(q = 0.5)$, $Q_n(q = 0.7)$ and $Q_n(q = 0.9)$, respectively. Ratio refers to the ratio between the tuning parameters, i.e. ratio = γ/λ. The upper x axis refers to the step size in the Frugal algorithms
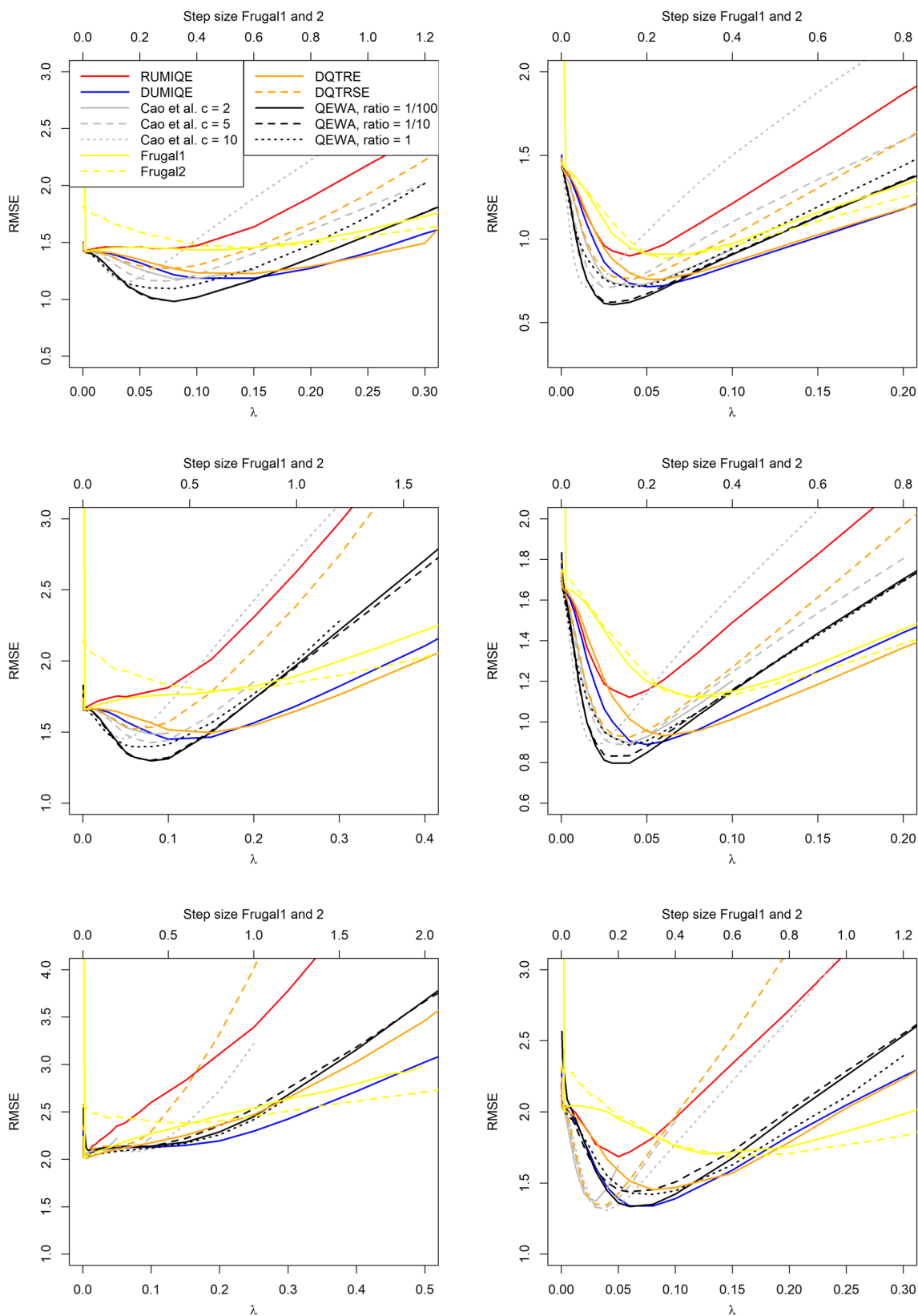
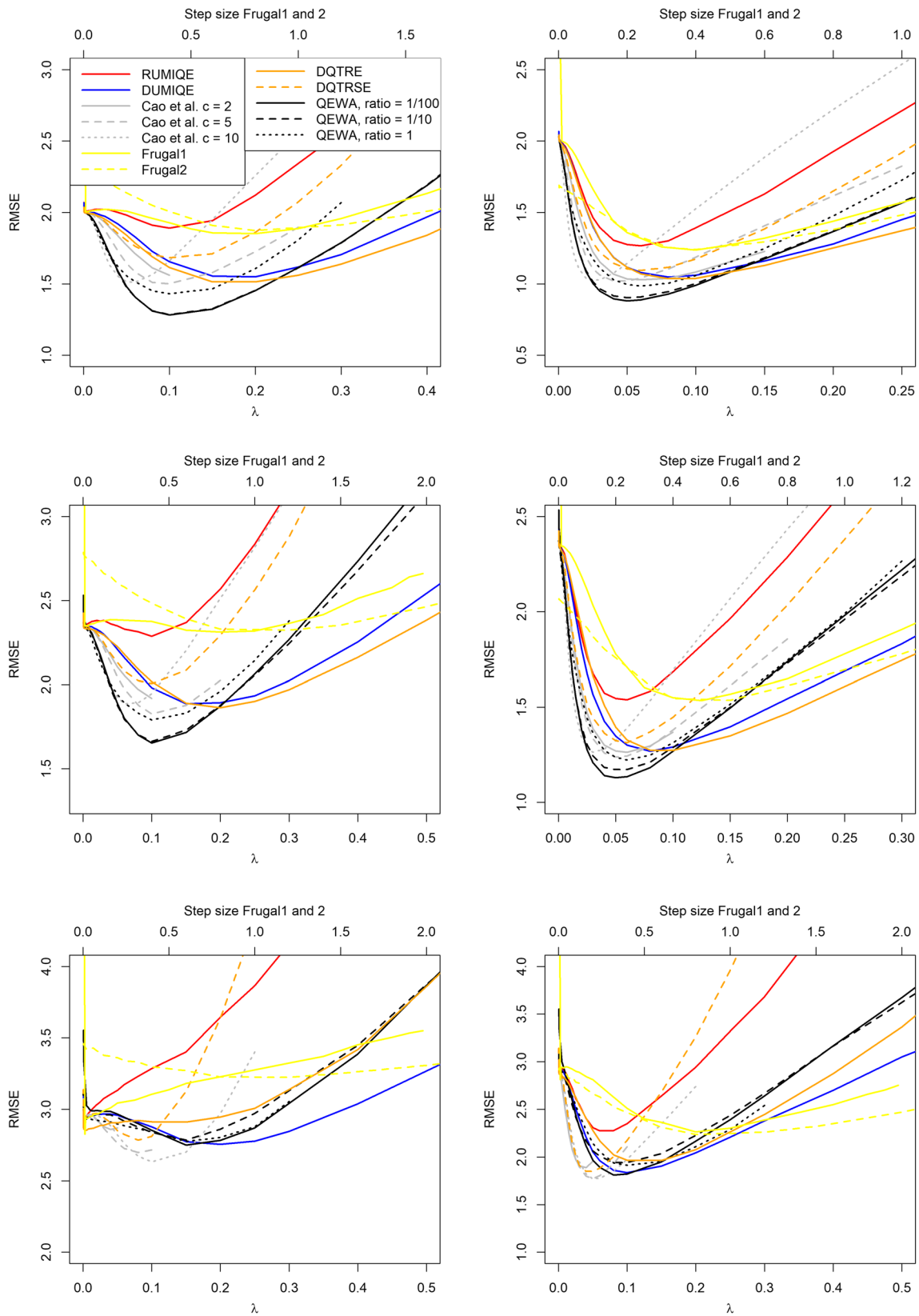**Fig. 5** χ^2 distribution switch case: The left and right columns show results for $T = 100$ and $T = 500$, respectively. The rows from top to bottom show results when estimating quantile $Q_n(q = 0.5)$, $Q_n(q =$ 0.7) and $Q_n(q = 0.9)$, respectively. Ratio refers to the ratio between the tuning parameters, i.e. ratio = γ/λ. The upper x axis refers to the step size in the Frugal algorithms

Table 2 Normal distribution smooth case: Root mean squared estimation error for the selection algorithm [17]

	$q = 0.5$	$q = 0.7$	$q = 0.9$
$T = 100$	1.4278	1.5279	1.7646
$T = 500$	1.4233	1.5433	1.7342

Table 4 χ^2 distribution smooth case: Root mean squared estimation error for the selection algorithm [17]

	$q = 0.5$	$q = 0.7$	$q = 0.9$
$T = 100$	1.4441	1.7423	2.4316
$T = 500$	1.4386	1.7273	2.6951

such that the estimation error in the stationary parts after convergence is the same for the two algorithms. We see that the proposed algorithm QEWA tracks the true quantile more efficiently after a switch than the DUMIQE. For the suggested algorithm, the step size is proportional to the difference between the observations and the quantile estimate (recall (4)). After a switch, these differences are large, and our devised algorithm makes large steps to get back on track. The DUMIQE, and the other state-of-the art incremental algorithms, use the same step size independent of these difference, resulting in poorer tracking.

The results below show a more systematic evaluation of the performance of the suggested algorithm against seven state-of-the-art quantile estimators namely the DUMIQE and RUMIQE by Yazidi and Hammer [38], the estimator due to Cao et al. [3], the Frugal approach by Ma et al. [24], the selection algorithm by Guha and McGregor [17] and the DQTRE and DQTRSE algorithms by Tiwari and Pandey [34]. For the DQTRE and DQTRSE algorithms we used values of the tuning parameters recommended in [34], namely $\alpha = 0.1$, $\beta = (1 - \alpha)^\lambda$, $p_b = 1/10$ and $l = 1/4$ which performed well in our experiments.

The estimator in this paper is designed to perform well for dynamically changing data streams and the experiments will focus on such streams.

We considered four different data cases. For the first case, the data stream distributions were normally distributed and the expectations, μ_n, varied smoothly as follows

$$\mu_n = a \sin\left(\frac{2\pi}{T}n\right), \quad n = 1, 2, 3, \ldots$$

which is a sinus function with period T. For the second case, the data stream distributions were also normally distributed, but the expectation switched between values a and $-a$

$$\mu_n = \begin{cases} a & \text{if } n \bmod T \leq T/2 \\ -a & \text{else} \end{cases}$$

We assumed that the standard deviation of the normal distributions did not vary with time and was equal to one. For the two remaining cases, the data stream distributions were χ^2 distributed, one with smooth changes and one with rapid switches. For the smooth case the number of degrees of freedom, ν_n, varied with time as follows

$$\nu_n = a \sin\left(\frac{2\pi}{T}n\right) + b, \quad n = 1, 2, 3, \ldots$$

where $b > a$ such that $\nu_n > 0$ for all n. For the switch case, the number of degrees of freedom switched between values $a + b$ and $-a + b$

$$\mu_n = \begin{cases} a + b & \text{if } n \bmod T \leq T/2 \\ -a + b & \text{else} \end{cases}$$

In the experiments we used $a = 2$ and $b = 6$.

We estimated quantiles of both the normally and χ^2 distributed data streams above using two different periods, namely $T = 100$ (rapid variation) and $T = 500$ (slow variation), i.e. in total eight different data streams. For each data stream we estimated the 50, 70 and 90% quantiles ending up with a total of 24 different estimation tasks.

To measure estimation error, we used the root mean squares error (RMSE) for each quantile given as:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} \left(Q_n(q) - \widehat{Q}_n(q)\right)^2} \tag{18}$$

where N is the total number of samples received from the data stream. In the experiments, we used $N = 10^6$ which efficiently removed any Monte Carlo errors in the experimental results. In order to obain a good overview of the performance of the algorithms, we measured the estimation error for a large set of different values of the tuning parameters of the algorithms.

Figures 2, 3, 4 and 5 illustrate the results of our experiments. For the normal distribution period case (Fig. 2), we see that the QEWA algorithm outperforms all the algorithms in the literature. In accordance with the

Table 3 Normal distribution switch case: Root mean squared estimation error for the selection algorithm [17]

	$q = 0.5$	$q = 0.7$	$q = 0.9$
$T = 100$	2.0541	2.3171	2.5479
$T = 500$	2.0947	2.3489	2.5427

Table 5 χ^2 distribution switch case: Root mean squared estimation error for the selection algorithm [17]

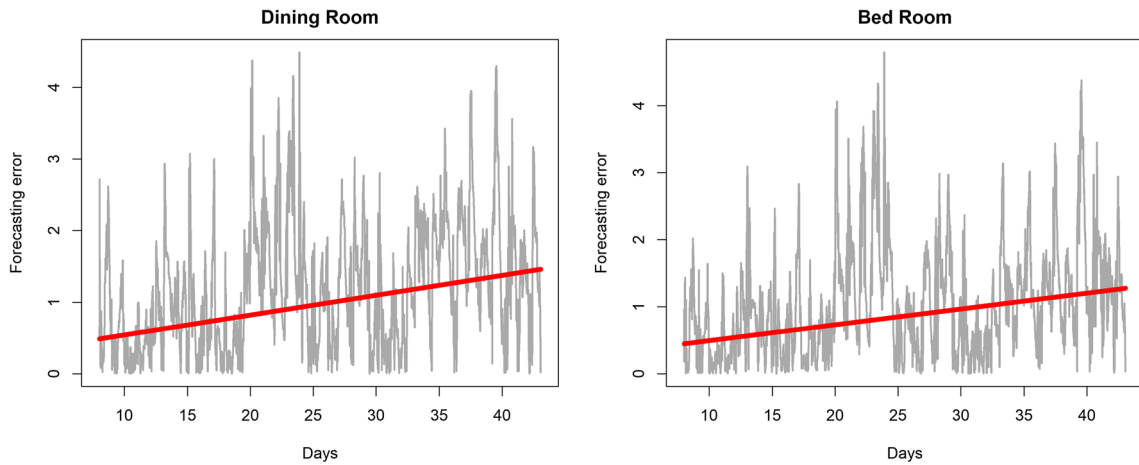|	$q = 0.5$	$q = 0.7$	$q = 0.9$
$T = 100$	2.0367	2.3913	3.3717
$T = 500$	2.0462	2.4137	3.1166

Dining Room

Bed Room

Fig. 6 The left and right panels refer to dining room and bed room, respectively. The x-axis refers to the number of days since the observation started. The gray curves show the forecasting error predicting 15 minutes into the future. The red curves show the linear trends in the forecasting error

analysis in Section 4, the QEWA algorithm performed the best using a small value of the ratio γ/λ. The Cao et al. algorithm struggled with numerical problems for some choices of the tuning parameters and therefore some of the curves are short.

For the normal distribution switch case (Fig. 3), we see that the QEWA algorithm again outperforms all the algorithms in the literature. Again we see that the QEWA performs best using a small value of the ratio γ/λ.

For the χ^2 distribution cases we see that the QEWA algorithm also here outperforms the other algorithms. For $q = 0.9$, the QEWA algorithm documents competitive results to the best performing alternative algorithms. Also here a small value of the ratio γ/λ is the preferable choice.

Among the alternative algorithms there were no consistency in which algorithms were closest to the performance of the QEWA, but overall the DUMIQE and DQTRE seem to be closest. However, all the alternative algorithms suffer with significantly poorer results than the QEWA for at least some cases. E.g. DQTRE performed poorly when estimating quantiles in the tails ($q = 0.9$) and DUMIQE for the switch cases.

Tables 2, 3, 4 and 5 show results for the selection algorithm [17]. The algorithm does not have any tuning parameters and the results are presented in the tables. We see that QEWA outperforms the selection algorithm with a clear margin for all the different cases.

In summary the QEWA algorithm outperforms all the different state-of-the-art algorithms from the literature. Best
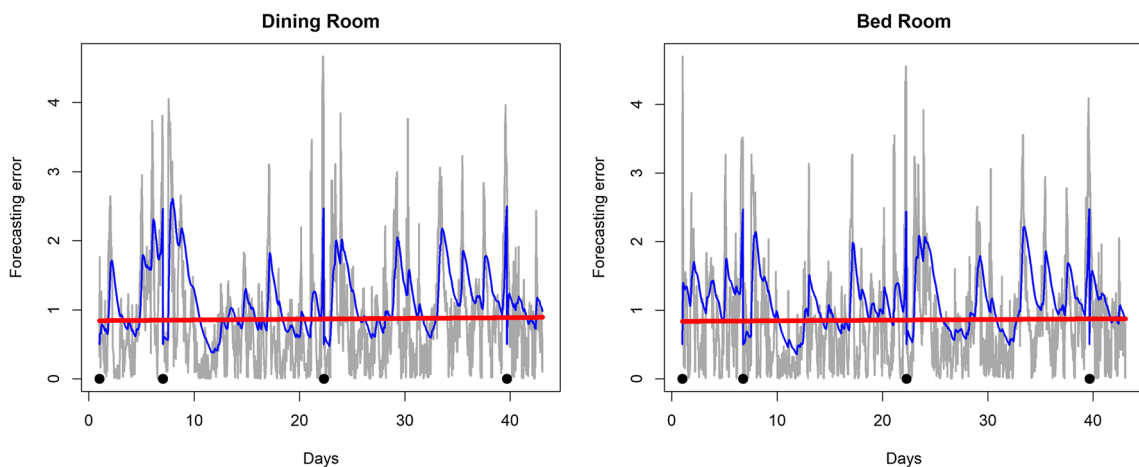
Dining Room

Bed Room

Fig. 7 The left and right panels refer to dining room and bed room, respectively. The x-axis refers to the number of days since the observation started. The gray curves show the forecasting error predicting 15 minutes into the future. The blue curves show tracking of the 80% quantiles of the forecasting error data streams. The black dots along the x-axis show when the model was retrained. The red show the linear trends in the forecasting error

performance is achieved using a small value of the ratio γ/λ.

6 Real-life data experiments – concept drift detection

In most challenging data prediction tasks, the relation between input and output data evolves over time. Thus if static relationships are assumed, prediction performance will degrade with time. In the field of machine learning and data mining this phenomenon is referred to as concept drift [13]. Different strategies have been suggested to detect when the performance of the predictive model degrades and thus should be retrained/updated [13]. Current state-of-the-art strategies monitor the *average* predictive error, but for real-life applications it is often more relevant to control that the prediction error rarely *goes above* some critical threshold. In this example we demonstrate how to perform concept drift detection and adaptation on such a critical threshold by tracking an upper quantile of the prediction error distribution, e.g. the 80% quantile. As an application domain, we investigate the case of efficient control of indoor climate.

Heating, ventilation and air conditioning (HVAC) systems typically control indoor climate by reacting on the current room conditions such as indoor temperature. However, given the time required for a HVAC system to adjust to changes in the indoor climate, such strategies always will lag behind resulting in poor control of indoor climate and energy usage. This raises the need for building models that forecast future indoor climate temperature and to use this as input to the HVAC system. Zamora-Martínez et al. [39] propose to use artificial neural network (ANN) models to forecast future indoor temperature based on a total of 20 features including outdoor climate variables such as temperature and precipitation amounts and indoor climates variables such as CO_2 level. Since more observations are received with time and the relation between input and output may evolve with time, the model is retrained in an online manner. The authors however do not take advantage of concept drift detection in order to efficiently decide when to retrain the model.

We now demonstrate how the suggested quantile estimator in this paper can be used for concept drift detection for the online indoor temperature forecasting problem described above. We consider the same dataset as in [39] where new observation of input and output variables is received every 15 minutes. We forecasted indoor temperature 15 minutes into the future using an autoregressive (AR) model of order one. In addition to the current indoor temperature, the current value of the other 20 features were used as input to the forecasting model. Given

the large number of features, regularization of the model parameters was required to get a reliable forecasts and we relied on LASSO regularization [10][3].

First, we trained the LASSO AR model based on eight days of observations and used the model to predict 15 minutes into the future each time a new observation was received. The results are shown in Fig. 6. The figure demonstrates that if the model is not retrained after day eight, the forecasting error gradually increases with time (the red line). In other words, the data is subject to concept drift and the forecasting model should be retrained as more observations are received. Instead of retraining the model regularly according to a fixed periodicity which is clearly ineffective, a sophisticated approach consists of retraining the model only if concept drift is detected.

We now build a concept drift and model retraining procedure based on quantile tracking. We required that the indoor temperature forecasting error rarely should go above two degrees centigrade. We used the QEWA estimator to track the 80% quantile of the forecasting error data stream (the gray curves in Fig. 6). If the quantile estimate went above two degrees centigrade, the model was retrained. We trained the model for the first time after 24 hours of observations. The results are shown in Fig. 7. After the initial training after 24 hours of observations, the 80% quantile estimate of the forecasting error distribution went above two degrees three times and each time the model was retrained. The results demonstrates that by a few selected retrainings of the model, the forecasting error is controlled, indicated by a horizontal linear trend (red curves) in Fig. 7.

In conclusion, the example demonstrates how the suggested quantile estimator can be useful for concept drift detection and model adaptation.

7 Closing remarks

The exponentially weighted moving average of observations is known to be the state-of-art estimator to track the expectation of dynamically varying data stream distributions. In this paper, we have presented an incremental quantile estimator that is in fact a generalized exponential weighted moving average estimator. To the best of our knowledge, this is the first quantile estimator in the literature that falls within this well-known class of efficient estimators. The experiments show that the estimator outperforms state-of-the-art quantile estimators in the literature.

---

[3]This model is a simple and natural forecasting model, but other and more advanced machine learning models that predict on the continuous scale, like ANN models, could also be used.

We demonstrate how tracking of quantiles has application in the field of machine learning. More particularly, we show how the suggested estimator can be used for tracking quantiles of the prediction error distribution in order to detect when a machine learning model should be retrained.

A potential ally for future research is to extend the QEWA estimator to simultaneously track multiple quantiles. One could of course, just run the QEWA estimator for each quantile of interest, but this could potentially lead to a violation of the monotone property of quantiles. The monotone property of quantiles, refers to the requirement that an estimate of a higher quantile should be always bigger than an estimate of a lower quantile e.g. the 50% quantile should always be above the 30% quantile.

Appendix A: Proof of theorem 1

We will first present a theorem due to Norman [26] that will be used to prove Theorem 1. Norman [26] studied distance "diminishing models". The convergence of $\widehat{Q}_n(q)$ to $Q(q)$ is a consequence of this theorem.

Theorem 2 *Let $x(t)$ be a stationary Markov process dependent on a constant parameter $\theta \in [0, 1]$. Each $x(t) \in I$, where I is a subset of the real line. Let $\delta x(t) = x(t + 1) - x(t)$. The following are assumed to hold:*

1. *I is compact*
2. *$E[\delta x(t)|x(t) = y] = \theta w(y) + O(\theta^2)$*
3. *$Var[\delta x(t)|x(t) = y] = \theta^2 s(y) + O(\theta^2)$*
4. *$E[\delta x(t)^3|x(t) = y] = O(\theta^3)$ where $sup_{y \in I} \frac{O(\theta^k)}{\theta^k} < \infty$ for $k = 2, 3$ and $sup_{y \in I} \frac{o(\theta^2)}{\theta^2} \to 0$ as $\theta \to 0$*
5. *$w(y)$ has a Lipschitz derivative in I*
6. *$s(y)$ is Lipschitz I.*

If Assumptions 1 to 6 above hold, $w(y)$ has a unique root y^* in I and $\frac{dw}{dy}\big|_{y=y^*} \leq 0$ then

1. *$var[\delta x(t)|x(0) = x] = O(\theta)$ uniformly for all $x \in I$ and $t \geq 0$. For any $x \in I$, the differential equation $\frac{dy(\tau)}{d\tau} = w(y(t))$ has a unique solution $y(\tau) = y(\tau, x)$ with $y(0) = x$ and $E[\delta x(t)|x(0) = x] = y(t\theta) + O(\theta)$ uniformly for all $x \in I$ and $t \geq 0$.*
2. *$\frac{x(t) - y(t\theta)}{\sqrt{\theta}}$ has a normal distribution with zero mean and finite variance as $\theta \to 0$ and $t\theta \to \infty$.*

Having presented Theorem 2, we are now ready to prove Theorem 1.

Proof We now start by showing that the Markov process based on the updating rules in (6) and Theorem 1 satisfies

the assumptions 1 to 6 in Theorem 2. We start by verifying assumption 2

$$
\begin{aligned}
& E\left(\delta\widehat{Q}_n(q) \mid \widehat{Q}_n(q)\right) \\
= & E\left(\delta\widehat{Q}_n(q) \mid \widehat{Q}_n(q) \geq X\right) P\left(\widehat{Q}_n(q) \geq X\right) \\
& + E\left(\delta\widehat{Q}_n(q) \mid \widehat{Q}_n(q) < X\right) P\left(\widehat{Q}_n(q) < X\right) \\
= & \lambda c_n\left(\widehat{Q}_n(q)\right) \frac{q}{\mu_n^+ - \widehat{Q}_n(q)}\left(\mu_n^+ - \widehat{Q}_n(q)\right)\left(1 - F\left(\widehat{Q}_n(q)\right)\right) \\
& - \lambda c_n\left(\widehat{Q}_n(q)\right) \frac{1-q}{\widehat{Q}_n(q) - \mu_n^-}\left(\widehat{Q}_n(q) - \mu_n^-\right) F\left(\widehat{Q}_n(q)\right) \\
= & \lambda c_n\left(\widehat{Q}_n(q)\right)\left(q - F\left(\widehat{Q}_n(q)\right)\right)
\end{aligned}
\tag{19}
$$

where $c_n\left(\widehat{Q}_n(q)\right)$ is as given in (3). We now let $\theta = \lambda$, $y = \widehat{Q}_n(q)$ and $w\left(\widehat{Q}_n(q)\right)$ equal to "everything" in (19) except λ. It is easy to see that assumption 2 in Theorem 2 is satisfied. Further, since $\mu_n^+ - \widehat{Q}_n(q) > 0$ and $\widehat{Q}_n(q) - \mu_n^- > 0$, $w\left(\widehat{Q}_n(q)\right)$ has a Lipschitz derivative and assumption 5 is satisfied.

Next we turn to assumption 3.

$$
\begin{aligned}
& E\left(\delta\widehat{Q}_n(q)^2 \mid \widehat{Q}_n(q)\right) = E\left(\delta\widehat{Q}_n(q)^2 \mid \widehat{Q}_n(q) \geq X\right) \\
& P\left(\widehat{Q}_n(q) \geq X\right) + E\left(\delta\widehat{Q}_n(q)^2 \mid \widehat{Q}_n(q) < X\right) \\
& P\left(\widehat{Q}_n(q) < X\right) = \lambda^2\left(c_n\left(\widehat{Q}_n(q)\right) \frac{q}{\mu_n^+ - \widehat{Q}_n(q)}\right)^2 \\
& \times\left(\mu_n^{2,+} - 2\widehat{Q}_n(q)\mu_n^+ + \widehat{Q}_n(q)^2\right)\left(1 - F\left(\widehat{Q}_n(q)\right)\right) \quad (20) \\
& + \lambda^2\left(c_n\left(\widehat{Q}_n(q)\right) \frac{q}{\widehat{Q}_n(q) - \mu_n^-}\right)^2 \\
& \times\left(\widehat{Q}_n(q)^2 - 2\widehat{Q}_n(q)\mu_n^- + \mu_n^{2,-}\right) F\left(\widehat{Q}_n(q)\right)
\end{aligned}
$$

where $\mu_n^{2,+} = E(X_n^2|X_n > \widehat{Q}_n(q))$ and $\mu_n^{2,-} = E(X_n^2|X_n < \widehat{Q}_n(q))$. Further we know that

$$
\begin{aligned}
Var\left(\delta\widehat{Q}n(q) \mid \widehat{Q}_n(q)\right) = & E\left(\delta\widehat{Q}n(q)^2 \mid \widehat{Q}n(q)\right) \\
& - E\left(\delta\widehat{Q}_n(q) \mid \widehat{Q}_n(q)\right)^2
\end{aligned}
\tag{21}
$$

By substituting (19) and (20) into (21), we see that assumption 3 is satisfied with $s\left(\widehat{Q}_n(q)\right)$ equal to everything in (21) except λ^2. Since $\mu_n^+ - \widehat{Q}_n(q) > 0$ and $\widehat{Q}_n(q) - \mu_n^- > 0$, $s\left(\widehat{Q}_n(q)\right)$ is Lipschitz and assumption 6 is also satisfied. Assumption 4 can now be proved in the same manner.

We will use the results of Norman to prove the convergence. It is easy to see that $w\left(\widehat{Q}_n(q)\right)$ in (19) admits one unique root $\widehat{Q}_n(q) = F^{-1}(q) = Q(q)$ (note $c_n\left(\widehat{Q}_n(q)\right) > 0 \; \forall\widehat{Q}_n(q)$).

We now differentiate to get:

$$\frac{d\,w\left(\widehat{Q}_n(q)\right)}{d\,\widehat{Q}_n(q)} = c'_n\big(\widehat{Q}_n(q)\big)\big(q - F\big(\widehat{Q}_n(q)\big)\big) - c_n\big(\widehat{Q}_n(q)\big)\,f\big(\widehat{Q}_n(q)\big)$$

We substitute the unique root $Q(q)$ for $\widehat{Q}_n(q)$ and get

$$\frac{d\,w\left(\widehat{Q}_n(q)\right)}{d\,\widehat{Q}_n(q)}\bigg|_{\widehat{Q}_n(q)=Q(q)} = c'_n\left(Q_n(q)\right)\left(q - F\left(Q_n(q)\right)\right)$$
$$- c_n\left(Q_n(q)\right)f\left(Q_n(q)\right)$$
$$= 0 - c_n\left(Q_n(q)\right)f\left(Q_n(q)\right) < 0$$

This gives

$$\lim_{n\lambda\to\infty,\lambda\to 0} E\left(\widehat{Q}_n(q)\right) = Q(q) + O(\lambda)$$

and

$$Var\left(\widehat{Q}_n(q)\right) = O(\lambda)$$

Consequently

$$\lim_{n\lambda\to\infty,\lambda\to 0}\widehat{Q}_n(q) = Q(q)$$

$\square$

References

1. Abbasi B, Guillen M (2013) Bootstrap control charts in monitoring value at risk in insurance. Expert Syst Appl 40(15):6125–6135
2. Arandjelovic O, Pham D-S, Venkatesh S (2015) Two maximum entropy-based algorithms for running quantile estimation in nonstationary data streams. IEEE Trans Circ Syst Video Technol 9:1469–1479
3. Cao J, Li L, Chen A, Bu T (2010) Tracking quantiles of network data streams with dynamic operations. In: INFOCOM Proceedings IEEE. IEEE, pp 1–5
4. Cao J, Li EL, Chen A, Bu T (2009) Incremental tracking of multiple quantiles for network monitoring in cellular networks. In: Proceedings of the 1st ACM workshop on mobile internet through cellular networks. ACM, pp 7–12
5. Chambers JM, James DA, Lambert D, Wiel SV et al (2006) Monitoring networked applications with incremental quantile estimation. Stat Sci 21(4):463–475
6. Chen F, Lambert D, Pinheiro JC (2000) Incremental quantile estimation for massive tracking. In: Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 516–522
7. Choi B-Y, Moon S, Cruz R, Zhang Z-L, Diot C (2007) Quantile sampling for practical delay monitoring in internet backbone networks. Comput Netw 51(10):2701–2716
8. Cormode G, Muthukrishnan S (2005) An improved data stream summary: the count-min sketch and its applications. J Algorithm 55(1):58–75
9. Espinosa HP, García CAR, Pineda LV (2010) Features selection for primitives estimation on emotional speech. In: 2010 IEEE international conference on acoustics speech and signal processing (ICASSP). IEEE, pp 5138–5141
10. Friedman J, Hastie T, Tibshirani R (2010) Regularization paths for generalized linear models via coordinate descent. J Stat Softw 33(1):1
11. Gaber MM, Gama J, Krishnaswamy S, Gomes JB, Stahl F (2014) Data stream mining in ubiquitous environments: state-of-the-art and current directions. Wiley Interdiscip Rev: Data Min Knowl Discov 4(2):116–138
12. Gama J (2013) Data stream mining: the bounded rationality. Informatica 37(1)
13. Gama J, Zliobaite I, Bifet A, Pechenizkiy M, Bouchachia A (2014) A survey on concept drift adaptation. ACM Comput Surv (CSUR) 46(4):44:1–44:37. https://doi.org/10.1145/2523813
14. Everette S (2006) Gardner. Exponential smoothing: the state of the art, part II. Int J Forecast 22(4):637–666
15. Gilli M et al (2006) An application of extreme value theory for measuring financial risk. Comput Econ 27(2-3):207–228
16. Gregory A, Lau F, Butler L (2018) A quantile-based approach to modelling recovery time in structural health monitoring. arXiv:1803.08444
17. Guha S, McGregor A (2009) Stream order and order statistics: quantile estimation in random-order streams. SIAM J Comput 38(5):2044–2059
18. Kejariwal A, Kulkarni S, Ramasamy K (2015) Real time analytics: algorithms and systems. Proc VLDB Endowment 8(12):2040–2041
19. Konda VR, Tsitsiklis JN (2004) Convergence rate of linear two-time-scale stochastic approximation. Ann Appl Probab 14(2):796–819
20. Krempl G, Žliobaite I, Brzeziński D, Hüllermeier E, Last M, Lemaire V, Noack T, Shaker A, Sievi S, Spiliopoulou M et al (2014) Open challenges for data stream mining research. ACM SIGKDD Explor Newsl 16(1):1–10
21. Lall A Data streaming algorithms for the kolmogorov-smirnov test. In: 2015 IEEE international conference on big data (Big Data). IEEE, pp 95–104
22. Liu J, Zheng W, Zheng L, Lin N (2018) Accurate quantile estimation for skewed data streams using nonlinear interpolation. IEEE Access
23. Luo G, Wang L, Yi K, Cormode G (2016) Quantiles over data streams: experimental comparisons, new analyses, and further improvements. The VLDB Journal–The International Journal on Very Large Data Bases 25(4):449–472
24. Ma Q, Muthukrishnan S, Sandler M (2013) Frugal streaming for estimating quantiles. In: space-efficient data structures, streams, and algorithms. Springer, pp 77–96
25. Ian Munro J, Paterson MS (1980) Selection and sorting with limited storage. Theor Comput Sci 12(3):315–323
26. Frank Norman M (1972) Markov processes and learning models, vol 84. Academic Press, New York
27. Ramírez-Gallego S, Krawczyk B, García S, Woźniak M, Herrera F (2017) A survey on data preprocessing for data stream mining: current status and future directions, vol 239
28. Schmeiser BW, Deutsch SJ (1977) Quantile estimation from grouped data: the cell midpoint. Commun Stat Simul Comput 6(3):221–234
29. Sen R, Maurya A, Raman B, Mehta R, Kalyanaraman R, Singh A (2014) Road-rfsense: a practical rf sensing–based road traffic estimation system for developing regions. ACM Trans Sensor Netw (TOSN) 11(1):4
30. Sommers J, Barford P, Duffield N, Ron A (2007) Accurate and efficient sla compliance monitoring. In: ACM SIGCOMM computer communication review. ACM, vol 37-4, pp 109–120
31. Sommers J, Barford P, Duffield N, Ron A (2010) Multiobjective monitoring for sla compliance. IEEE/ACM Trans Netw (TON) 18(2):652–665
32. Stahl V, Fischer A, Bippus R (2000) Quantile based noise estimation for spectral subtraction and wiener filtering. In: acoustics, speech, and signal processing, 2000. ICASSP'00.

Proceedings IEEE International Conference on. IEEE, vol 3, pp 1875–1878
33. Tierney L (1983) A space-efficient recursive procedure for estimating a quantile of an unknown distribution. SIAM J Sci Stat Comput 4(4):706–711
34. Tiwari N, Pandey PC (2018) A technique with low memory and computational requirements for dynamic tracking of quantiles. Journal of Signal Processing Systems. https://doi.org/10.1007/s11265-017-1327-6
35. Vogt T, André E (2005) Comparing feature sets for acted and spontaneous speech in view of automatic emotion recognition. In: 2005 ICME 2005 IEEE international conference on multimedia and expo. IEEE, pp 474–477
36. Wang W, Ching W-K, Wang S, Yu L (2016) Quantiles on stream An application to monte carlo simulation. J Syst Sci Inf 4(4):334–342
37. Weide B (1978) Space-efficient on-line selection algorithms. In: Computer science and statistics: proceedings of the eleventh annual symposium on the interface, pp 308–311
38. Yazidi A, Hammer HL (2017) Multiplicative Update Methods for Incremental Quantile Estimation. IEEE Transactions on Cybernetics (accepted)
39. Zamora-Martínez F, Romeu P, Botella-Rocamora P, Pardo J (2014) On-line learning of indoor temperature forecasting models towards energy efficiency. Energy Build 83:162–172
40. Zhang L, Guan Y (2008) Detecting click fraud in pay-per-click streams of online advertising networks. In: 28th international conference on distributed computing systems ICDCS'08
41. Zhang X, Alexander L, Hegerl GC, Jones P, Tank AK, Peterson TC, Trewin B, Zwiers FW (2011) Indices for monitoring changes in extremes based on daily temperature and precipitation data. Wiley Interdiscip Rev Clim Chang 2(6):851–870

Anis Yazidi received the M.Sc. and Ph.D. degrees from the University of Agder, Grimstad, Norway, in 2008 and 2012, respectively. He is currently an Associate Professor with the Department of Computer Science OsloMet - Oslo Metropolitan University, Oslo, Norway. Before joining OsloMet, he worked as a researcher with Teknova AS, Grimstad, Norway. His current research interests include machine learning, learning automata, stochastic optimization, software-defined networks and cloud computing. He led the research group Autonomous Systems and Networks at OsloMet from 2015 to 2018. He is currently the leader of the research group Applied Artificial Intelligence at OsloMet.

Håvard Rue received the M.Sc. and Ph.D. degrees from the Norwegian Institute of Technology, in 1988 and 1993, respectively. He is currently a professor in statistics with the Computer, Electrical and Mathematical Science and Engineering Division, King Abdullah University of Science and Technology (KAUST), Saudi Arabia. Before joining KAUST, he worked as a Professor in statistics at Norwegian University of Science and Technology. His research interests lie in computational Bayesian statistics and Bayesian methodology such as priors, sensitivity and robustness. His main body of research is built around the R-INLA project (www.r-inla.org), which aims to provide a practical tool for approximate Bayesian analysis of latent Gaussian models, often at extreme data scales. This project also includes efforts to use stochastic partial differential equations to represent Gaussian fields, for the use in spatial statistics. He is the leader of the research group Bayesian Computational Statistics and Modeling.

Hugo Lewi Hammer received the M.Sc. and Ph.D. degrees from the Norwegian University of Science and Technology, in 2003 and 2008, respectively. He is currently an Associate Professor with the Department of Computer Science, OsloMet - Oslo Metropolitan University, Oslo, Norway. Before joining OsloMet, he worked as a researcher with Norwegian Computer Center, Oslo, Norway. His research interests include computer intensive statistical methods, machine learning, learning automata and stochastic optimization.