Efficient Mergeable Quantile Sketches using Moments

Extended Abstract

Edward Gan, Jialin Ding, Peter Bailis
Stanford Infolab

ABSTRACT

The collection of increasingly high volume data from heterogeneous sources motivates the use of data sketches with very low storage and update overhead. To address these needs, we draw a connection between sketching and statistical moments to develop an efficient and mergeable sketch for approximate quantile queries, referred to as the moments sketch. The moments sketch operates with a small memory footprint (< 200 bytes) and computationally efficient (< 30 ns) merges and updates by tracking only the summary statistics of a data stream. We can then efficiently recover quantile estimates from the sketch by applying the method of moments and the maximum entropy principle. This allows the moments sketch to match the accuracy of alternative sketches while operating with much less storage and runtime overhead. Efficient maximum entropy solvers and fast merge times then enable significant end to end speedups in large-scale analytics.

1 INTRODUCTION

The tremendous growth of log data from sources such as mobile phones, sensors, and datacenters makes it possible to monitor complex application deployments at a per-device and per-minute resolution. Engineers rely on being able to estimate the latency quantiles (i.e. p99) for hundreds of thousands of device-type and app-version combinations over weeks and months [11]. These quantiles are then used to power both real-time dashboards as well as downstream classification and feature selection queries.

However, the volume of data in modern deployments makes it difficult to compute quantiles over fine-grained subpopulations at interactive speeds. Given billions of log events, selection or sorting on raw values is expensive. Alternatively, *mergeable summaries* provide compact synposes of data that can be combined without loss of accuracy [2, 10]. Simple mergeable summaries can include samples or histograms, and they are used in data aggregation engines such as Druid and Spark [5, 9, 27] to calculate quantiles without keeping raw data in memory, to distribute work across nodes, and to pre-aggregate data by materializing summaries at ingest time. However, the memory overheads and merge times of existing sketches still become prohibitive when many sketches must be aggregated.

To illustrate these system requirements, we describe our experience working with a large industrial data stream which deploys the MacroBase feature selection engine [6] on top of a system very similar to Druid [27]. In this deployment, real time telemetry metrics arrive at a rate of billions of events per day. As they arrive, the Druid-like engine groups them by dimensions such as application version, OS version, and hardware model, and pre-aggregates them into five-minute windows, maintaining a summary for each sub-group. The engine calculates quantiles for specific time-ranges and dimensions by merging the appropriate summaries. Even so, telemetry deployments can have hundreds of thousands of dimension combinations, so calculating a quantile over a 2-week range requires loading and merging half a billion summaries. We have found existing summaries generally operate with either microsecond-level merge times or kilobyte-level space overheads, leading to multiple minute query times and terabyte-level memory usage.

The above scenarios motivate quantile summaries that are:

- **Compact**, to limit network and memory overhead,
- **Mergeable** [2], so sketches can be aggregated, and
- **Fast**, to enable interactive latencies on large datasets.

To address these requirements we draw a connection between mergeable sketches and the statistical method-of-moments [26]. We propose a simple data summary, referred to as the *Moments Sketch* (M-sketch), which tracks summary statistics such as mean, higher moments, and min and max. Using the method of moments we can construct a maximum entropy distribution that matches the moments in an M-sketch, and provide empirically accurate quantile estimates. This data structure is easily mergeable, and we find that the M-sketch can achieve the same accuracy with less memory and orders of magnitude faster merge and update times compared with existing sketches.

In summary, we identify and evaluate the use of higher moments for fast, compact, and mergeable quantile sketches, and we develop efficient methods for estimating quantiles from moments.

2 METHODS

An order-k M-sketch consists of $k + 3$ double-precision floating point numbers summarizing a dataset $X = \{x : x \in \mathbb{R}\}$: the min, max, count, and the first k moments. The k-th moment (denoted m_k) of a given dataset X is the summary statistic given by $m_k = \frac{1}{|X|}\sum_{x \in X} x^k$. The first moment is the mean, the second moment determines the standard deviation, and higher moments describe a distribution's skew and kurtosis. Note that these are the moments of the empirical distribution, not those of the unknown underlying distribution. In practice we extend the M-sketch to track other statistics such as the log-moments $\frac{1}{|X|}\sum_{x \in X} (\log x)^k$ to support heavy-tailed distributions, though we omit this extension in this abstract due to space constraints. Merge and update operations on an M-sketch operate over the components independently and consist of a handful of floating point operations per component.

To query an M-sketch, we must first estimate the underlying distribution from its moments. The method of moments allows us to estimate distribution parameters from observed moments [4, 7, 16, 26]. However, in general, the first k moments are insufficient to uniquely determine an arbitrary distribution [3], so we use the maximum-entropy principle [15] to select a single distribution estimate from

SysML2018, February 2018, Stanford, CA
2018. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

the space of possible distributions. Specifically, a maximum entropy distribution with pdf f is a distribution with maximal differential entropy $H = -\int_{X \subset \mathbb{R}} f(x) \log f(x) dx$ under certain constraints – in this case observed moments. Maximizing entropy is a means of "making inferences on the basis of partial information" while minimizing the bias of unfounded assumptions [15], and provides useful estimates in a variety of domains [21, 22]. In the case of quantile estimation, this principle is particularly applicable to smooth, continuous datasets without unusual gaps and spikes. Thus though M-sketch provides a small amount of robustness to outliers, performance degrades when working with large anomalous values.

The M-sketch tracks a finite set of moments, so we can use functional optimization to show that a maximum entropy distribution with these moments must be of the form $f(x) = \exp\left(\sum_{i=0}^{k} \lambda_i x^i\right)$. Then, we can use Newton's method to solve for the parameters λ_i such that $\int f(x) x^i \, dx = m_i$, as described in [20]. To improve the convergence of Newton's method, we express the polynomial $\sum \lambda_i x^i$ as a sum of Chebyshev polynomials [23]. Since these polynomials form an orthogonal basis, small changes in $f(x)$ map more directly to small changes in the coefficients, and they yield better-conditioned Hessians [25]. Letting $T_i(x)$ denote the i-th Chebyshev polynomial of the first kind, the maximum entropy distribution we seek then has the form $f(x) = \exp\left(\sum_{i=0}^{k} c_i T_i(x)\right)$. We solve for c_i by minimizing the potential $P(\vec{c}) = \int f(x) \, dx - \sum_{i=0}^{k} c_i \mu_i^c$ where μ_i^c are the Chebyshev data moments $\mu_i^c = \frac{1}{|X|} \sum_{x \in X} T_i(x)$.

This potential is convex so we minimize it using Newton's method with backtracking [8]. As part of Newton's method, we compute ∇P and $\nabla^2 P$ efficiently with a modification of Clenshaw-Curtis integration [23]. In our evaluation datasets, for $k \leq 10$ our implementation converges in less than 20 Newton steps. Given $f(x)$, we can finally calculate quantiles using numeric integration.

Accuracy Bounds. Given a ϕ-quantile estimate $\hat{q}_\phi$, its quantile error $\epsilon \in [0, 1]$ is $|\phi - F^{-1}(\hat{q}_\phi)|$ where F^{-1} is the inverse cdf. Along with a quantile estimate, the M-sketch provides a worst case upper bound on the error ϵ of its estimate $\hat{q}_p$ using the techniques in [3, 19, 24]. These bounds are analogous to stronger versions of the Markov and Chebyshev inequalities. Though they are useful, they are orders of magnitude more conservative ($\approx O(1/k)$ for k moments [17]) than the observed error for the relatively high-entropy distributions we observe in practice.

3 EVALUATION

To compare the accuracy and merge times of the M-sketch across a variety of sketch sizes, we obtained 81 million log-scale internet traffic measurements from the Milan telecom dataset [14] for November 2013, divided them into subsets of 100 points each, and pre-aggregated sketches on each subset. Then, to replicate the aggregation workflows discussed earlier, we merge all of the sketches and estimate quantiles over the merged sketch, measuring the average quantile error ϵ for 21 equally spaced quantiles between 0.01 and 0.99.

We measured performance on an Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz processor with 1TB of RAM. We obtained timings for a single merge by dividing the total merge time by the number of sketches merged, averaged across 10 runs. The M-sketch is implemented in Java, and we compare against a Java equi-width histogram (EW-Histogram), the Greenwald-Khanna (GK) sketch [13] as implemented in Spark-SQL [5], the T-Digest sketch [12], reservoir sampling, and the low-discrepancy mergeable sketch (Merge12) from [2], both implemented in the Yahoo! datasketches library [1].

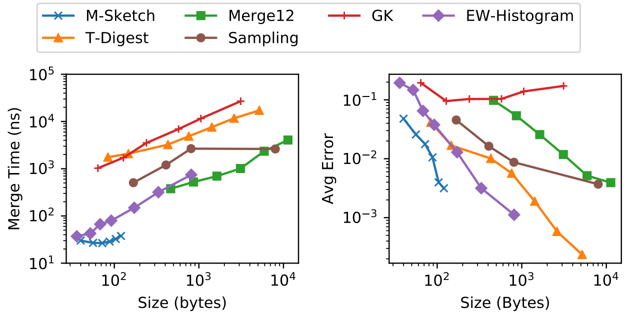

Figure 1: The M-sketch achieves sub-1% error with fast merges and low space usage.

Figure 1 compares the merge time (per sketch) and accuracy of the M-sketch with other summaries. Since each sketch is parameterized by its size, we instantiate each sketch at a range of sizes. The M-sketch has faster merge times and smaller space usage than other sketches that achieve the same ϵ error. Going back to the scenario in the Introduction, for quantile queries over a half-billion summaries on a single node an order 8 M-sketch would take 30ns * 0.5 bln = 15 seconds to merge, and up to an additional 5ms to solve, yielding much faster total times than sketches with microsecond merge times.

The GK sketch is not fully mergeable and suffers accuracy loss on every merge, which is why its error does not go to 0 even as it is given more space. We observe that histograms and the M-sketch both provide the fast merge times required for scalable analytics. However, equi-width histograms perform very poorly in the presence of even a single large outlier. In separate evaluations, we have observed that an M-sketch with 10 moments consistently achieves $\epsilon \approx .01$ accuracy on a variety of real-valued datasets including the HEPMASS and Occupancy dataset from UCI [18] and synthetic Gaussian datasets an outlier at $x = 20\sigma$.

4 DISCUSSION AND FUTURE WORK

Interactive analytics are increasingly bottlenecked on their ability to aggregate large volumes of data. The moments sketch provides one approach to achieving high performance merge times and low storage sizes when estimating distributions, allowing interactive modeling on multi-faceted populations.

The cost of solving for quantiles from moments can be further reduced by using a cascade of cheaper bounds to prune subpopulations as part of threshold or top-k queries, for example in queries for the set device types with the top p99 latencies. The cost of storing the M-sketch can also be reduced using low-precision floating point. Looking forward, we hope to apply our techniques directly to downstream modeling tasks that depend on distribution estimation, including regression and classification.

Acknowledgements. We thank Kai Sheng Tai and Vatsal Sharan for their feedback in this work. This research was supported in part by affiliate members and other supporters of the Stanford DAWN project - Google, Intel, Microsoft, Teradata, and VMware - as well as DARPA under No. FA8750-17-2-0095 (D3M), industrial gifts and support from Toyota Research Institute, Juniper Networks, Keysight Technologies, Hitachi, Facebook, Northrop Grumman, NetApp, and the NSF under grant DGE-114747.

REFERENCES

[1] 2017. Yahoo! Data Sketches Library. (2017). https://datasketches.github.io/.

[2] Pankaj K. Agarwal, Graham Cormode, Zengfeng Huang, Jeff Phillips, Zhewei Wei, and Ke Yi. 2012. Mergeable Summaries. In *PODS*.

[3] N.I. Akhiezer. 1965. *The Classical Moment Problem and Some Related Questions in Analysis*. Oliver & Boyd.

[4] Animashree Anandkumar, Daniel Hsu, and Sham M Kakade. 2012. A method of moments for mixture models and hidden Markov models. In *Conference on Learning Theory*. 33–1.

[5] Michael Armbrust, Reynold S Xin, Cheng Lian, Yin Huai, Davies Liu, Joseph K Bradley, Xiangrui Meng, Tomer Kaftan, Michael J Franklin, Ali Ghodsi, et al. 2015. Spark sql: Relational data processing in spark. In *SIGMOD*. ACM, 1383–1394.

[6] Peter Bailis, Edward Gan, Samuel Madden, Deepak Narayanan, Kexin Rong, and Sahaana Suri. 2017. MacroBase: Prioritizing attention in fast data. In *SIGMOD*. ACM, 541–556.

[7] Mikhail Belkin and Kaushik Sinha. 2010. Polynomial Learning of Distribution Families. In *Foundations of Computer Science (FOCS '10)*. 10.

[8] Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press, New York, NY, USA.

[9] Mihai Budiu, Rebecca Isaacs, Derek Murray, Gordon Plotkin, Paul Barham, Samer Al-Kiswany, Yazan Boshmaf, Qingzhou Luo, and Alexandr Andoni. 2016. Interacting with Large Distributed Datasets Using Sketch. In *Eurographics Symposium on Parallel Graphics and Visualization*.

[10] Graham Cormode, Minos Garofalakis, Peter J Haas, and Chris Jermaine. 2012. Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends in Databases* 4, 1–3 (2012), 1–294.

[11] Jeffrey Dean and Luiz André Barroso. 2013. The Tail at Scale. *Commun. ACM* 56, 2 (Feb. 2013), 74–80.

[12] Ted Dunning and Otmar Ertl. 2017. Computing extremeley accurate quantiles using t-digests. https://github.com/tdunning/t-digest. (2017).

[13] Michael Greenwald and Sanjeev Khanna. 2001. Space-efficient online computation of quantile summaries. In *SIGMOD*, Vol. 30. ACM, 58–66.

[14] Telecom Italia. 2015. Telecommunications - SMS, Call, Internet - MI. (2015). https://doi.org/10.7910/DVN/EGZHFV

[15] E. T. Jaynes. 1957. Information Theory and Statistical Mechanics. *Phys. Rev.* 106 (May 1957), 620–630. Issue 4.

[16] Adam Tauman Kalai, Ankur Moitra, and Gregory Valiant. 2010. Efficiently learning mixtures of two Gaussians. In *Symposium on Theory of computing*. ACM, 553–562.

[17] Weihao Kong and Gregory Valiant. 2017. Spectrum estimation from samples. *Ann. Statist.* 45, 5 (10 2017), 2218–2247.

[18] M. Lichman. 2013. UCI Machine Learning Repository. (2013). http://archive.ics.uci.edu/ml

[19] Bruce G. Lindsay and Prasanta Basak. 2000. Moments Determine the Tail of a Distribution (But Not Much Else). *The American Statistician* 54, 4 (2000), 248–251.

[20] Lawrence R. Mead and N. Papanicolaou. 1984. Maximum entropy in the problem of moments. *J. Math. Phys.* 25, 8 (1984), 2404–2417.

[21] Kamal Nigam. 1999. Using maximum entropy for text classification. In *In IJCAI-99 Workshop on Machine Learning for Information Filtering*. 61–67.

[22] Steven J. Phillips, Miroslav Dudík, and Robert E. Schapire. 2004. A Maximum Entropy Approach to Species Distribution Modeling. In *ICML (ICML '04)*. ACM, New York, NY, USA, 83–.

[23] William H Press. 2007. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press.

[24] Sandor Racz, Arpad Tari, and Miklos Telek. 2006. A moments based distribution bounding method. *Mathematical and Computer Modelling* 43, 11 (2006), 1367 – 1382.

[25] RN Silver and H Röder. 1997. Calculation of densities of states and spectral functions by Chebyshev recursion and maximum entropy. *Physical Review E* 56, 4 (1997), 4822.

[26] Larry Wasserman. 2010. *All of Statistics: A Concise Course in Statistical Inference*. Springer Publishing Company, Incorporated.

[27] Fangjin Yang, Eric Tschetter, Xavier Léauté, Nelson Ray, Gian Merlino, and Deep Ganguli. 2014. Druid: A Real-time Analytical Data Store. In *SIGMOD*. 157–168.