Current Trends in Data Summaries

Graham Cormode*
Meta AI

ABSTRACT

The research area of data summarization seeks to find small data structures that can be updated flexibly, and answer certain queries on the input accurately. Summaries are widely used across the area of data management, and are studied from both theoretical and practical perspectives. They are the subject of ongoing research to improve their performance and broaden their applicability. In this column, recent developments in data summarization are surveyed, with the intent of inspiring further advances.

1. INTRODUCTION

The data management community makes extensive use of various kinds of summaries: compact data structures that represent a large dataset, and allow queries to be answered with some guarantee of accuracy. The most common example of summaries come in the form of samples, where evaluating a query on a sample provides an approximate answer to the query on the full data set. Other popular summary types are Bloom filters [8], which approximately represent sets, and sketches [12], which approximately represent vectors, as well as other summaries targeting more specific queries. Key application areas include approximate query processing (AQP), where sampling is quite ubiquitous [43], and distributed and stream processing [25].

The design and application of summaries is now ubiquitous within the research community, and has been the subject of several tutorials and books, covering developments from the late 1970s onwards [60, 45, 56, 19]. In this column, I will give a very high-level survey of current active research directions in data summarization, with emphasis on results from the last few years. This is a very subjective and partial view, based on topics that have been the focus of recent papers in data management venues, or just ones that have caught the interest of researchers in this area. The intent is, fittingly, to draw an approximate summary of efforts in this area, rather than a precise characterization.

*gcormode@fb.com

2. SUMMARIES FOR ML

Given the high level of interest in machine learning (ML) across computer science and beyond, it should be no surprise that researchers are looking to use data summaries in order to improve the ML training process. The primary application of summaries is to try to reduce the size of ML models without sacrificing their expressivity. The most natural place to apply data summaries is in compressing the information exchanged between data owners during the training of networks. In distributed training of machine learning models (usually referred to as Federated Learning [37]), each client holds some labeled examples, and a server sends out a candidate model. Each client evaluates the candidate model on their labeled examples, and determines an update to the model, typically in the form of a gradient vector to adjust the model parameters in order to improve the accuracy of the model on their examples. The server will then update the model based on combining these gradients, often by moving in the direction of the average gradient. However, the size of the model can be very large, and sending the full gradient vector can have high computational cost for each client (in terms of uplink communication). It is natural to look to data summaries as a way to reduce the size of the communication, with the tradeoff of potentially slightly increasing the number of steps before the model converges, or of slightly reducing the accuracy of the final model that is found.

Two recent papers suggest similar approaches to reducing communication in Federated Learning with the use of sketches. In FetchSGD [52], the authors propose the use of the CountSketch summary [12] as the medium through which to convey the gradient updates. CountSketch has several attractive features: it promises to preserve the large entries of the input vector accurately, and so using sketches captures the most significant parts of the updates. In addition, it is a linear summary: sketches can be summed and subtracted, with the resulting sketch being identical to the one we would obtain if we had applied these operations to the input vectors before sketching. This means that we can treat the

sketches as if they were the full vectors, and apply various techniques from machine learning, such as momentum (including updates from previous iterations at lower weight) and error compensation. In addition, it is possible to prove results on the speed and accuracy of convergence under standard ML assumptions.

The FedSketch [28] paper follows a similar outline, also making use of CountSketch as a compression operator. It additionally considers the provision of a differential privacy guarantee, taking advantage of both linearity and the sparsity of the CountSketch transformation. Experiments and analysis demonstrate that this approach converges more quickly than other previously proposed private federated approaches. Away from the federated setting, Tai *et al.* propose the Weight-Median sketch as a tool for sketching gradients, which is applied to learn linear classifiers over streams of updates [54].

There are a number of other directions in which summarization can assist in machine learning. An orthogonal approach to handling the large size of ML models in the literature is to apply quantization to the model parameters. That is, rather than representing each parameter with a 32 or 64 bit floating point representation, they can instead be represented more crudely by a much fewer number of bits. Currently popular approaches apply fairly simple quantization encoding – for example, using 8 bits to represent values divided uniformly between a minimum and maximum value. This approach is rather coarse, and can lead to errors accumulating when multiple quantized update vectors are combined together. A more promising approach might be to use randomized representations of values, so that errors tend to cancel out on average as more vectors are aggregated [58]. Similarly, pruning is a simple way to reduce the size of an update vector. Under pruning, values in update vectors with small magnitude are pruned to zero, and can be omitted from reporting back to the server. An intriguing open research direction would be to combine pruning with techniques from data summarization (e.g., sketching), to more compactly encode the sparse pruned updates.

3. ML FOR SUMMARIES

Just as summarization can help with machine learning, so too can machine learning help summarization. A highly impactful paper from 2018 argued that rather than traditional indices (B-trees and the like), it is valuable to use compact models to access data [41]. That is, train a model to predict where to find a piece of data, by minimizing an appropriate loss function, since all indices can be interpreted as implicit models of the data layout. One way to "train" a Bloom filter is to optimize the hash functions: to define a hash function via a machine learning model (a neural network), which is optimized to reduce the number of false positives for a given set of data.

This notion has been generalized to a wider range of summaries. Hsu *et al.* considered sketches for frequencies [30]. Similar to the Bloom filter case, the aim is to choose a hash function that gives better results for a data distribution than choosing a random hash function. The authors show that it is indeed possible to "learn" a good hash function, and analyze the resulting error under some assumptions on this distribution. Jiang *et al.* [35] expanded the applicability of this approach to a range of other summary types, such as distinct counting and frequency moments. In more detail, the approach is to assume the existence of a "frequency oracle" for the distribution, so that given an item the oracle accurately predicts the frequency of this item in the full distribution. By handling items differently based on their predicted frequency, it is possible to obtain bounds on the size of summaries better than those in the general case without such an oracle.

This paradigm has sparked work in other directions, notably for linear algebra involving large matrices. Indyk *et al.* [33] consider learning a low-rank approximation of a matrix, aiming to minimize the Frobenius norm of the difference between the original and approximate matrix. The approach is to learn a sketch projection matrix through which to generate the approximation. It is observed that the error can be reduced by up to an order of magnitude compared to a randomly chosen sketch. Li *et al.* [44] similarly consider sketches for the Hessian of matrices, and apply these to ML problems such as regularized regression (LASSO) and matrix regression. ML techniques have even been applied to learn how to multiply matrices (Blalock and Guttag [7]): here, the aim is to learn functions that can be applied to matrices A and B so as to allow a fast construction of a matrix C that is close to AB under the Frobenius norm.

It will be interesting to think more generally about summaries augmented with an oracle that (accurately or perfectly) captures some part of the problem being studied, to understand the impact of the hardness of the task. This can be viewed as a different kind of assumption compared to promises on the arrival order of data items (arbitrary, random or worst-case) or on the statistical distribution of data values that have been made in prior work (e.g., [27, 17]). Graphs and matrices are natural candidates: how well can we summarize the structures if we have, for example, a shortest path oracle, or access to the eigenvalues?

4. SUMMARIES IN PRIVACY

The objective of privacy enhancing technologies is to limit the amount of information revealed to an

observer, while the objective of data summarization is to support answering a particular query while limiting the amount of information retained. There is sufficient alignment from these two objectives that it is feasible to use data summaries as part of a privacy solution to assist with the information limitation. This has led to a number of advances in privacy technology. The large scale deployments of private data collection by Google [21] and Apple [2], which both relied on the use of summaries, meant that these were some of the most high-profile applications for data summaries. Specifically, the Rappor system from Google was built on Bloom Filters [8], while the Apple implementation made use of sketches to bound the dimensionality of the data gathered [19]. These two examples were both primarily concerned with gathering frequency statistics from high dimensional distributions, to find the heavy hitters from the input via so-called "frequency oracles" in the local model of differential privacy. Bassily *et al.* formalized this approach in their analysis [4].

More generally, there has been a growth in interest in the area of Federated Analytics (FA), which seeks to gather information from multiple distributed clients in order to provide statistics on the union of their inputs. Unsurprisingly, data summaries can be employed in the construction of federated analytics protocols. The demands of FA go beyond those for summaries that can be constructed independently and merged centrally. Typically, we seek some additional guarantee of privacy. A clear example is given by the TrieHH protocol proposed by researchers at Google [62]. Here, the aim is to find the set of heavy hitter items from a large collection. The general approach is to gather information from distributed clients in order to search for heavy hitters in a hierarchical fashion, similar to approaches performed in the data streaming setting. However, the set of candidate items is identified by a sampling step, with a novel proof that those items whose frequency in the sample exceeds a threshold achieve a (centralized) differential privacy guarantee, without the need for explicit noise addition.

Recently, there has also been interest in studying the inherent privacy offered by data summaries. The intuition is clear: when summaries store very compact information about their input, it is natural to imagine that the information retained about any given input item should be quite small, and hence private. Formalizing this intuition, and ensuing that it is not possible to "invert" the summarization process to recover the input items, requires considerable care and effort. Recent results on approximate counting have shown that the Flajolet-Martin summary achieves a level of differential privacy – provided that the observer does not know which hash functions were used to create the summary (which is assumed to be a uniform random permutation), and the cardinality of items being summarized is not too small [53, 13]. This refines the work of Desfontaines *et al.* [20], which showed that applied directly, many distinct count sketches do *not* provide a privacy guarantee. Most recently, Pagh and Stausholm give a sketch for this problem with privacy guarantees where the hash function can be known to the adversary, and privacy is achieved by perturbing the stored information, i.e., by applying randomized response to the stored bits [49]. This enables private sketches to be shared between multiple parties in order to approximate the cardinality of unions of sets.

Two other foundational summarization tasks are sampling and counting. Work by Cohen *et al.* [14] looks at private sampling from weighted inputs, where the weights can be thought of as the number of individuals who hold a particular item. The aim is to produce a compact collection of items and noisy weights, so that the collection functions as a good sample of the input (representing the weight distribution), while protecting the privacy of individuals who contributed the data. This means that particular care has to be taken to ensure that low weights do not reveal information about the data of the participants. The essence of the approach is to define inclusion probabilities for elements based on weights which achieve both sampling accuracy and differential privacy. In particular, a sampling scheme is defined such that sampling probabilities for weights that differ by one meet the (approximate) differential privacy definition. The approach inherits many of the benefits of (non-private) sampling, such as accurate estimators for linear statistics, and gives solution for many private tasks, such as quantiles and histograms.

Gathering accurate (private) statistics in the distributed setting while minimizing communication naturally benefits from data summarization techniques. This gives the multiparty differential privacy model, which generalizes both the local model (where each of k users holds a single item) and the central model (where multiple items are held by a single entity). Recent work makes use of the Count Sketch, whose sparsity means that it has low sensitivity under differential privacy [31]. Instead of merging the sketches as in a standard linear sketch by using the same set of parameters (sketch size and hash functions), the construction uses different parameters for each user based on the size of their input, and combines the estimates from each sketch with an additional error bound. This approach saves a $\sqrt{k}$ factor in the multiparty model, and achieves an optimal error-communication tradeoff.

It is natural to ask what other problems with a privacy requirement can be helped by the use of summaries, or other ideas inspired by summarization. A particular challenge in privacy is handling

longitudinal data, i.e., situations where a user participates in the data collection multiple times as time goes on, but we wish to give an overall guarantee on the privacy despite a potentially unbounded influence on the data. There have certainly been efforts to address this concern, but the approaches deployed in practice are not entirely satisfying, relying either on "resetting" the privacy budget on a daily basis, or using a somewhat heuristic memoization of random values [2, 21]. The basic idea of keeping a tree-structure over continually observed to reduce the noise to logarithmic [11] has been widely used for similar purposes, most recently in the context of federated learning [36].

5. NEW MODELS: ROBUST STREAMING

One of the core areas that motivates the development of new summary structures is the area of data stream processing. Here, the aim is to summarize a large input arriving as a stream of inputs, in order to answer a basic query, such as estimating the frequency moments of the data distribution. Traditionally, summaries have been analyzed assuming that the stream may be arbitrary, but is fixed independent of the random choices of the summarization algorithm. This allows effective randomized algorithms to be proposed with strong space-accuracy tradeoffs. However, there are cases where this may seem overly optimistic: when the data structure is queried during the arrival of the stream, knowledge of the approximate answer could be used to influence the subsequent items in the input, and elicit an erroneous answer. To ensure the highest level of reliability, we might ask whether it is possible to design summary techniques that are robust to inputs that are chosen adversarially, in reaction to the actions of the algorithm. A starting point is deterministic algorithms: any approach which gives a guarantee that holds over all possible inputs is necessarily robust to adversarial inputs. However, for many fundamental problems in streaming, it is known that there is a large gap between deterministic and randomized bounds, where often no deterministic algorithm can do better than storing the whole input.

A recent line of work has considered this question, and shown that it is possible to construct summaries that are indeed robust in this fashion, with a moderate overhead compared to their non-robust alternatives. Ben-Eliezer and Yogev [6] first considered the adversarial robustness of sampling. It is perhaps not very surprising that drawing a random sample of a stream of data is fairly robust to an adversary choosing the input items, since the sampling is performed without close inspection of any item. However, one could envision an adversary who observes the current state of the sample, and chooses input items in order to try to exaggerate any ways in which the sample is already misrepresentative. The results of Ben-Eliezer and Yogev prove that nevertheless, to evade any such adversary, the sample only needs to be a small factor larger than in the non-adversarial case.

A subsequent work of Ben-Eliezer *et al.* [5] considers a broader range of problems, such as frequency moments, distinct counting and frequency estimation, in the adversarial setting. This work was recognized as the best paper of PODS 2020. The central result is a generic framework which introduces the parameter of the *flip number*. This counts how often the answer of the algorithm must change over the course of observing its input. Since we typically consider approximate algorithms, it is often the case that the summary can give the same output for an extended period while still meeting the required approximation bounds. Consider, for example, the (trivial) streaming algorithm to count the number of items observed so far, n. We can observe that to give a 2-factor approximation, the flip number can be bounded to $O(\log n)$ (we only have to change the output after the input size has doubled). More sophisticated arguments serve to bound the flip number for more challenging functions. The paper then argues that it suffices to run multiple copies of a (non-adversarially robust) streaming summary in parallel. We can report the output of one summary while it is an accurate enough approximation of the true answer, then switch to a 'fresh' instance when this changes. The number of summaries to maintain is then linear in the flip number of the problem considered.

Subsequent work has built on this foundation. Hassidim *et al.* make an intriguing connection between robustness and privacy, by employing differential privacy to thwart the adversary [29]. Specifically, the technique also runs multiple copies of non-adversarial streaming algorithms for the problem, but then aggregates their output in a way that provides a differential privacy guarantee. The intent is that the adversary, observing the changing output of the algorithm, is nevertheless unable to draw strong inferences about the inner state of the various summaries due to the privacy noise. Significantly, the cost of the approach also depends on the flip number, but is now proportional to the square root of the flip number. Another surprising connection work that draws a link between adversarial sampling and the theory of online learning [1]. It shows that the concepts for which there exist effective adversarially robust sampling mechanisms are those that meet a definition of online learnability. Braverman *et al.* have demonstrated that the commonly used technique of "merge and reduce" to build summaries over distributed data brings with it a guarantee of adversarial robustness, providing strong guarantees for various clustering problems such as

k-means, k-median, k-center and more [10]. Meanwhile, Woodruff and Zhou showed tighter bounds for various problems in the sliding window streaming model [59]. A strong separation was shown between the adversarial and non-adversarial model by Kaplan *et al.* [38], by considering the "adaptive data analysis" problem, which can be shown to require exponentially more space in the adversarial setting.

There are many open directions in the area of robust streaming, as evidenced by a recent workshop day dedicated to the topic[1]. Some immediate directions are to understand the true dependence on the flip number in the space bounds. Is it too much to hope for a polylogarithmic bound by keeping this many instances of independent summaries, and selecting random subsets of these to provide an estimate? More generally, could the notion of using differential privacy as a tool to fool adversaries have wider applicability?

6. PROGRESS IN APPROXIMATE COUNTING

Counting is one of the most basic computational tasks, so it is hard to imagine that there would be new progress on it. Nevertheless, in the last few years there have been some intriguing new steps made for counting, specifically on various notions of approximate counting. Approximate counting via the Morris counter is often used as an example in a randomized algorithms class [47]. The algorithm keeps a counter with a small bit depth, and processes increment updates. The internal counter is incremented with probability that decreases exponentially with its value. This can be used to estimate quantities with value up to n using bit depth of only $O(\log \log n)$. Recently, Nelson and Yu [48] revisited this problem, and showed tighter bounds on the accuracy of such counters. In particular, they showed a new algorithm with a simple proof that uses space $O(\log 1/\epsilon + \log \log 1/\delta + \log \log n)$ in order to approximate a quantity up to n with $1 \pm \epsilon$ accuracy with probability $1 - \delta$. They go on to show via a more involved proof that the same bound holds for a lightly modified version of the original Morris algorithm. This improves the dependency on δ exponentially. Offering accurate approximate counters in small space is of value to data science applications which maintain a large number of counters for many different events in parallel.

In a different setting, recent work has tried to reduce the size of counters down to a single bit. Specifically, we have a number of participants who each hold a real value x, scaled to the range $[0, 1]$, and our aim is to gather information from them in order to estimate the mean of their (scaled) values. A simple randomized rounding approach is to round x to 1 with probability x, and 0 otherwise: the expectation of this rounding is x. Ben Basat *et al.* [3] consider a variety of related approaches, and show that variance of $\frac{1}{4}$ of the simple rounding approach can be improved in situations when shared randomness is available, or a biased estimator can be adopted. Note that limiting to a single random bit alone may not make a big difference to communication cost: the overheads in packet-switched networks are such that the difference between sending 1 bit vs. 64 bits is small compared to the cost of packet headers etc. However, this approach offers clearer benefits when sending larger volumes of data (say, a vector of values), or when we want to apply privacy to the transmitted bits, and can randomly noise the bit that is sent.

The counting problem becomes more challenging when we have to address the problem of distinct counting: given an unsorted collection of items (with some repeated), we seek to estimate the cardinality of the support set. This problem appears in many applications where summaries are desirable, and many effective algorithms have been proposed. Perhaps the most famous of these is the HyperLogLog summary presented by Flajolet et al. [23]. A recent advance on this problem is due to Pettie and Wang, who seek to understand tight bounds for the space complexity of this problem – again, this is a pressing concern when maintaining approximate (distinct) counters for a large number of different objects [50]. In particular, they show a new approach to analyzing the space complexity by fusing the Fisher information with the Shannon entropy of the summary. This enables them to revisit the exact constants of an algorithm due to Flajolet and Martin [24], when implemented in a compressed form. Under some restrictions, they show that this sketch is optimal (including the constant factor), which settles a long line of work seeking increasingly tight bounds for this problem. Rather than being a theoretical observation about an impractical algorithm, the "compressed probabilistic counting" technique was already implemented in the Apache data sketches library[2], and has been used internally within Oath (Yahoo!) for monitoring large volumes of statistics. The analytical study of Pettie and Wang complemented the numerical study of Lang, who implemented and evaluated this algorithm [42]. In subsequent work, Pettie *et al.* went on to study the space complexity of non-mergable summaries for distinct counting, and show that sacrificing mergability can obtain slightly higher space efficiency for summaries [51].

The next step might be to move these advances in approximate counting closer to applications. As noted above, the importance of machine learning, which relies in part on large collections of numeric

[1] `https://rajeshjayaram.com/stoc-2021-robust-streaming-workshop.html`

[2] `http://datasketches.apache.org`

values, is a strong candidate to benefit from approximate counters, either during training, or after training for efficient communication and storage on devices. More generally, the proliferation of data means that it is ever easier to capture and store large volumes of data should provide an important use-case for approximate counting in various forms, particularly to handle counters which vary frequently over time. It would be particularly compelling to see empirical evidence of the benefits of using approximate counting in practice.

7. PROGRESS IN QUANTILES

Given a collection of data items from an ordered domain, the quantiles characterize the cumulative distibution function (CDF) of the empirical distribution. In simpler terms, they capture the median, and more generally the percentiles of the data. Given a fixed data set, finding the quantiles can be done easily if it is feasible to sort the data, and with more effort without sorting by a classical linear time algorithm [9]. However, in the context of summarization, we often seek a compact summary that can be created from a stream of updates, or by merging summaries of subsets of the dataset together, without having random access to the dataset in full. Until recently, the state of the art was generally considered to be the Greenwald-Khanna summary (from 2001) [26], and the KLL summary (from 2016) [39]. Both give an additive guarantee as a function of a parameter ϵ: given a target quantile, they guarantee to return an item whose rank in the sorted order of n items is at most ϵn from the target. The GK summary provides a deterministic guarantee with an $O(\frac{1}{\epsilon}\log \epsilon n)$-sized summary, while the KLL summary gives a randomized guarantee with an $O(\frac{1}{\epsilon})$-sized summary.

A number of recent advances have enhanced our understanding of this problem. From PODS 2020, a new result showed that the GK summary is essentially optimal among algorithms which only perform comparisons between items to determine what summary to retain [18]. The main result in the paper is an intricate construction based on white-box knowledge of the operation of a quantile algorithm, to construct paired inputs that maximize the error of a deterministic summary. It proceeds recursively to obtain the $\log \epsilon n$ factor in the lower bound, improving over both the trivial $\Omega(1/\epsilon)$ lower bound, and a more involved bound of $\Omega(\frac{1}{\epsilon}\log(1/\epsilon))$ that is nevertheless independent of the input size [32]. The deterministic lower bound can also be applied in the very low failure probability regime, to provide a lower bound for randomized algorithms, and so shows that the KLL summary is similarly optimal when the error probability is exponentially small.

Other advances on quantiles have considered variations of the problem and showed new results by adapting the KLL algorithm. Zhao *et al.* [61] propose "KLL ±", which accepts an input consisting of a mixture of insertions and deletions. Handling an arbitrary number of deletions can be hard: consider an input which deletes all but an arbitrary handful of items. To give a quantile guarantee on this input, the algorithm must be able to retrieve exactly the set of items which survive to the end. Instead, it is more feasible to consider the case of *bounded deletions*, where the number of deletions is promised to be at most $1 - 1/\alpha$, for a parameter α. The algorithm applies a variant of the KLL algorithm to the stream of insertions and deletions, and drops tuples when an insertion, deletion pair for the same item are placed together in the data structure. The result is shown to provide the desired additive ϵ guarantee with space $\tilde{O}(\frac{\alpha^{1.5}}{\epsilon})$.

A different goal is to provide a *relative error* guarantee for quantiles. That is, instead of answering a query with an item a fixed distance from the target quantile, we seek an item whose distance is a small fraction of the true rank of the target. This is important for cases where we seek to find accurate answers for items in the tail of the distribution, i.e., the 99^{th}, 99.9^{th} and 99.99^{th} percentiles. The problem is challenging, since if we do not retain accurate enough information on items that need high precision, we cannot hope to remedy this deficit. The "relative error quantiles sketch", which adapts the structure of the KLL algorithm to provide this improved accuracy guarantee was given the best paper award in PODS 2021 [15]. The space bound achieved is $O(1/\epsilon \log^{3/2} \epsilon n)$, which improves on prior bounds of $O(1/\epsilon \log^3 \epsilon n)$, and is close to the trivial lower bound of $\Omega(1/\epsilon \log \epsilon n)$.

There are many natural questions for this line of work. Most obviously would be to understand whether the $\log^{3/2} \epsilon n$ can be reduced closer to $\log \epsilon n$, or whether this unusual exponent is inherent. It would also be desirable to streamline and simplify the construction and its proof. In particular, the argument that instances of the relative error quantiles sketch can be merged together is very intricate. This is not to say that the algorithm itself is impractical: it has been implemented within the Apache DataSketches library[3], and used within Splunk for tracking distributions to monitor for changes. A recent empirical study compared the algorithm to a popular alternative approach, the t-digest, and showed that while the t-digest does well on "typical" inputs, there are adversarially crafted inputs on which the t-digest can be made to give extremely high error, while the relative error quantiles sketch maintains the same level of accuracy throughout [16]. Consequently, it would be highly desirable to build a summary that obtains the best of both worlds: small space and high accuracy on typical inputs,

[3] `https://datasketches.apache.org/`

while retaining space and accuracy guarantees even against worst-case inputs.

8. IMPROVEMENTS WITH EXISTING SUMMARIES: NEW BOUNDS AND NEW APPLICATIONS

One reasons for the popularity of summaries in practice is that they often give accurate results even with only small amounts of space allocated. This is in part because they follow the behavior predicted by their theoretical analysis, and often the analysis is fairly tight. That is, rather than being governed by bounds in big-Oh notation with hidden constants, we often understand their costs in closed form, with quite small explicit constants. Still, there is the strong desire to further close the gap between the good performance seen in practice and the worst-case bounds from analysis, to allow even tighter provisioning of resources for the summaries (i.e., allocate the smallest space possible to achieve the desired level of accuracy).

A good example is the Count-Min sketch, a very simple randomized summary. The original analysis uses elementary tools (such as the Markov inequality) to give a strong accuracy bound on a simple biased estimator with explicit constants. More recently, Ting [55] revisited this structure and proposed new estimators for the same sketch which provide more accurate and unbiased estimators for frequency estimation. The analysis makes use of statistical tools such as the bootstrap to provide a data-dependent error guarantee. In particular, it uses information from the parts of the sketch that do not directly answer the query in order to build an improved estimator.

Other works have sought to apply similar tools from statistics in order to give improved bounds. As described in Section 6, Nelson and Yu give improved bounds for the approximate Morris counter [48]. Ertl [22] analyzed distinct counters for the task of estimating the size of intersections between sets. This is a problem with strong lower bounds, since the intersection size can be small while the sets can be large, and so obtaining relative error is not possible. While presenting a new sketch for this problem, Ertl proposed a more general closed-form estimator that can be applied to existing sketches, such as the popular HyperLogLog summary [23]. Lopes *et al.* similarly consider sketches for matrix computations such as least-squares regression, and use a bootstrap-based approach to provide error estimates for the approximate solution. A key feature is that bootstrap is used here to understand the random variation due to a randomization in the algorithm, rather than variation in the data.

Summary techniques are increasingly finding new applications in other areas to help improve bounds. A very partial sampling of these includes:

- Sketches to help solve linear programs [57], making use of the count-sketch summary [12], taking advantage of its ability to accurately capture the heavy hitters.
- Sketches for approximate pattern matching under string edit distance [40], by summarizing strings with low edit distance to the input string.
- Solving regression problems on data that is represented in a factorized format via sketching [34].
- Using sketches to understand the trade-off between distortion and communication in voting situations [46].

We conclude with some natural (if generic) open questions: for what other summary techniques can we obtain improved bounds by exploiting more advanced analysis techniques? What new applications of summaries can there be, across the important areas of optimization, string processing, and graph and linear algebra computations? A more extensive list of open questions, covering a range of topics in sublinear algorithms, can be found at `sublinear.info`.

Acknowledgements.
Thanks to Justin Thaler, Ke Yi, Daniel Ting, David Woodruff, Jelani Nelson, Vladimir Braverman, and Christian Konrad for their suggestions of papers to highlight in this column. Thanks to Dan Olteanu for suggesting this column and providing helpful feedback.

9. REFERENCES

[1] N. Alon, O. Ben-Eliezer, Y. Dagan, S. Moran, M. Naor, and E. Yogev. Adversarial laws of large numbers and optimal regret in online classification. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 447–455. ACM, 2021.

[2] Apple Differential Privacy Team. Learning with privacy at scale. `https://docs-assets.developer.apple.com/ml-research/papers/learning-with-privacy-at-scale.pdf`, 2017.

[3] R. B. Basat, M. Mitzenmacher, and S. Vargaftik. How to send a real number using a single bit (and some shared randomness). In *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021*, volume 198 of *LIPIcs*, pages 25:1–25:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[4] R. Bassily, K. Nissim, U. Stemmer, and A. Thakurta. Practical locally private heavy

hitters. *J. Mach. Learn. Res.*, 21:16:1–16:42, 2020.

[5] O. Ben-Eliezer, R. Jayaram, D. P. Woodruff, and E. Yogev. A framework for adversarially robust streaming algorithms. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS*, pages 63–80. ACM, 2020.

[6] O. Ben-Eliezer and E. Yogev. The adversarial robustness of sampling. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS*, pages 49–62. ACM, 2020.

[7] D. W. Blalock and J. V. Guttag. Multiplying matrices without multiplying. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 992–1004. PMLR, 2021.

[8] B. H. Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

[9] M. Blum, R. W. Floyd, V. R. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, August 1973.

[10] V. Braverman, A. Hasidim, Y. Matias, M. Schain, S. Silwal, and S. Zhou. Adversarial robustness of streaming algorithms through importance sampling. In *NeurIPS*, 2021.

[11] T. H. Chan, E. Shi, and D. Song. Private and continual release of statistics. *ACM Trans. Inf. Syst. Secur.*, 14(3):26:1–26:24, 2011.

[12] M. Charikar, K. C. Chen, and M. Farach-Colton. Finding frequent items in data streams. *Theor. Comput. Sci.*, 312(1):3–15, 2004.

[13] S. G. Choi, D. Dachman-Soled, M. Kulkarni, and A. Yerukhimovich. Differentially-private multi-party sketching for large-scale statistics. *Proc. Priv. Enhancing Technol.*, 2020(3):153–174, 2020.

[14] E. Cohen, O. Geri, T. Sarlós, and U. Stemmer. Differentially private weighted sampling. In *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 130 of *Proceedings of Machine Learning Research*, pages 2404–2412. PMLR, 2021.

[15] G. Cormode, Z. S. Karnin, E. Liberty, J. Thaler, and P. Veselý. Relative error streaming quantiles. In *PODS'21: Proceedings of the 40th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 96–108. ACM, 2021.

[16] G. Cormode, A. Mishra, J. Ross, and P. Veselý. Theory meets practice at the median: A worst case comparison of relative error quantile algorithms. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2722–2731. ACM, 2021.

[17] G. Cormode and S. Muthukrishnan. Summarizing and mining skewed data streams. In *Proceedings of the 2005 SIAM International Conference on Data Mining, SDM*, pages 44–55. SIAM, 2005.

[18] G. Cormode and P. Veselý. A tight lower bound for comparison-based quantile summaries. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS*, pages 81–93. ACM, 2020.

[19] G. Cormode and K. Yi. *Small summaries for big data*. CUP, 2020.

[20] D. Desfontaines, A. Lochbihler, and D. A. Basin. Cardinality estimators do not preserve privacy. *Proc. Priv. Enhancing Technol.*, 2019(2):26–46, 2019.

[21] Ú. Erlingsson, V. Pihur, and A. Korolova. RAPPOR: randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 1054–1067. ACM, 2014.

[22] O. Ertl. Setsketch: Filling the gap between minhash and hyperloglog. *Proc. VLDB Endow.*, 14(11):2244–2257, 2021.

[23] P. Flajolet, E. Fusy, O. Gandouet, and F. Meunier. Hyperloglog: The analysis of a near-optimal cardinality estimation algorithm. *Discrete Mathematics and Theoretical Computer Science Proceedings*, page 127–146, 2007.

[24] P. Flajolet and G. N. Martin. Probabilistic counting algorithms for data base applications. *J. Comput. Syst. Sci.*, 31(2):182–209, 1985.

[25] M. Fragkoulis, P. Carbone, V. Kalavri, and A. Katsifodimos. A survey on the evolution of stream processing systems. *CoRR*, abs/2008.00842, 2020.

[26] M. Greenwald and S. Khanna. Space-efficient online computation of quantile summaries. In *Proceedings of the 2001 ACM SIGMOD international conference on Management of data*, pages 58–66. ACM, 2001.

[27] S. Guha and A. McGregor. Stream order and order statistics: Quantile estimation in random-order streams. *SIAM J. Comput.*, 38(5):2044–2059, 2009.

[28] F. Haddadpour, B. Karimi, P. Li, and X. Li. Fedsketch: Communication-efficient and private federated learning via sketching. *CoRR*, abs/2008.04975, 2020.

[29] A. Hassidim, H. Kaplan, Y. Mansour, Y. Matias, and U. Stemmer. Adversarially robust streaming algorithms via differential privacy. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*, 2020.

[30] C. Hsu, P. Indyk, D. Katabi, and A. Vakilian. Learning-based frequency estimation algorithms. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net, 2019.

[31] Z. Huang, Y. Qiu, K. Yi, and G. Cormode. Frequency estimation under multiparty differential privacy: One-shot and streaming. *CoRR*, abs/2104.01808, 2021.

[32] R. Y. S. Hung and H. Ting. An $\omega(\frac{1}{\epsilon}\log\frac{1}{\epsilon})$ space lower bound for finding ϵ-approximate quantiles in a data stream. In *Frontiers in Algorithmics, 4th International Workshop, FAW 2010,*, volume 6213 of *Lecture Notes in Computer Science*, pages 89–100. Springer, 2010.

[33] P. Indyk, A. Vakilian, and Y. Yuan. Learning-based low-rank approximations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, pages 7400–7410, 2019.

[34] R. Jayaram, A. Samadian, D. P. Woodruff, and P. Ye. In-database regression in input sparsity time. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 4797–4806. PMLR, 2021.

[35] T. Jiang, Y. Li, H. Lin, Y. Ruan, and D. P. Woodruff. Learning-augmented data stream algorithms. In *8th International Conference on Learning Representations, ICLR 2020*. OpenReview.net, 2020.

[36] P. Kairouz, B. McMahan, S. Song, O. Thakkar, A. Thakurta, and Z. Xu. Practical and private (deep) learning without sampling or shuffling. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 5213–5225. PMLR, 2021.

[37] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. A. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konečný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, H. Qi, D. Ramage, R. Raskar, M. Raykova, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao. Advances and open problems in federated learning. *Found. Trends Mach. Learn.*, 14(1-2):1–210, 2021.

[38] H. Kaplan, Y. Mansour, K. Nissim, and U. Stemmer. Separating adaptive streaming from oblivious streaming using the bounded storage model. In *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference*, volume 12827 of *Lecture Notes in Computer Science*, pages 94–121. Springer, 2021.

[39] Z. S. Karnin, K. J. Lang, and E. Liberty. Optimal quantile approximation in streams. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS*, pages 71–78. IEEE Computer Society, 2016.

[40] T. Kociumaka, E. Porat, and T. Starikovskaya. Small space and streaming pattern matching with k edits. *CoRR*, abs/2106.06037, 2021.

[41] T. Kraska, A. Beutel, E. H. Chi, J. Dean, and N. Polyzotis. The case for learned index structures. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018*, pages 489–504. ACM, 2018.

[42] K. J. Lang. Back to the future: an even more nearly optimal cardinality estimation algorithm. *CoRR*, abs/1708.06839, 2017.

[43] K. Li and G. Li. Approximate query processing: What is new and where to go? - A survey on approximate query processing. *Data Sci. Eng.*, 3(4):379–397, 2018.

[44] Y. Li, H. Lin, and D. P. Woodruff. Learning-augmented sketches for hessians. *CoRR*, abs/2102.12317, 2021.

[45] E. Liberty and J. Nelson. Streaming data mining. In *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*, 2012.

[46] D. Mandal, N. Shah, and D. P. Woodruff. Optimal communication-distortion tradeoff in voting. In *EC '20: The 21st ACM Conference on Economics and Computation*, pages 795–813. ACM, 2020.

[47] R. H. Morris Sr. Counting large numbers of events in small registers. *Commun. ACM*, 21(10):840–842, 1978.

[48] J. Nelson and H. Yu. Optimal bounds for approximate counting. *CoRR*, abs/2010.02116, 2020.

[49] R. Pagh and N. M. Stausholm. Efficient differentially private F0 linear sketching. In *24th International Conference on Database Theory, ICDT*, volume 186 of *LIPIcs*, pages 18:1–18:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[50] S. Pettie and D. Wang. Information theoretic limits of cardinality estimation: Fisher meets shannon. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 556–569. ACM, 2021.

[51] S. Pettie, D. Wang, and L. Yin. Non-mergeable sketching for cardinality estimation. In *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021*, volume 198 of *LIPIcs*, pages 104:1–104:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[52] D. Rothchild, A. Panda, E. Ullah, N. Ivkin, I. Stoica, V. Braverman, J. Gonzalez, and R. Arora. Fetchsgd: Communication-efficient federated learning with sketching. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 8253–8265. PMLR, 2020.

[53] A. D. Smith, S. Song, and A. Thakurta. The flajolet-martin sketch itself preserves differential privacy: Private counting with minimal space. In *Annual Conference on Neural Information Processing Systems*, 2020.

[54] K. S. Tai, V. Sharan, P. Bailis, and G. Valiant. Sketching linear classifiers over data streams. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018*, pages 757–772. ACM, 2018.

[55] D. Ting. Count-min: Optimal estimation and tight error bounds using empirical error distributions. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018*, pages 2319–2328. ACM, 2018.

[56] D. Ting, J. Malkin, and L. Rhodes. Data sketching for real time analytics: Theory and practice. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3567–3568. ACM, 2020.

[57] J. van den Brand, Y. T. Lee, A. Sidford, and Z. Song. Solving tall dense linear programs in nearly linear time. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 775–788. ACM, 2020.

[58] S. Vargaftik, R. B. Basat, A. Portnoy, G. Mendelson, Y. Ben-Itzhak, and M. Mitzenmacher. DRIVE: one-bit distributed mean estimation. *CoRR*, abs/2105.08339, 2021.

[59] D. P. Woodruff and S. Zhou. Tight bounds for adversarially robust streams and sliding windows via difference estimators. *CoRR*, abs/2011.07471, 2020.

[60] K. Yi. Random sampling on big data: Techniques and applications. In *Proceedings of the ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017*, 2017.

[61] F. Zhao, S. Maiyya, R. Weiner, D. Agrawal, and A. E. Abbadi. KLL±: Approximate quantile sketches over dynamic datasets. *Proc. VLDB Endow.*, 14(7):1215–1227, 2021.

[62] W. Zhu, P. Kairouz, B. McMahan, H. Sun, and W. Li. Federated heavy hitters discovery with differential privacy. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 108, pages 3837–3847. PMLR, 2020.