Exploring Sketches for Probability Estimation with Sublinear Memory

Anthony Kleerekoper, Mikel Luján and Gavin Brown
School of Computer Science
The University of Manchester, UK
Email: {kleereka, mikel.lujan, gbrown}@cs.man.ac.uk

***Abstract*—As data sets become ever larger it becomes increasingly complex to apply traditional machine learning techniques to them. Feature selection can greatly reduce the computational requirements of machine learning but it too can be memory intensive. In this paper we explore the use of succinct data structures called sketches for probability estimation as a component of information theoretic feature selection. These data structures are sublinear in the number of items but were designed only for estimating the frequency of the most frequent items. To the best of our knowledge this is the first time they have been examined for estimating the frequency of all items and we find that often some information theoretic measures can be estimated to within a few percent of the correct values.**

***Keywords*—big data, information theoretic feature selection, machine learning, memory efficiency, sketch data structures**

I. Introduction

Information is being gathered and stored in larger and larger amounts, making it harder to apply traditional machine learning algorithms to the new very large data sets. Information theoretic feature selection can greatly reduce the complexity and memory requirements of machine learning by identifying only those items which are most significant [3], [1], [2]. In order to apply information theoretic feature selection, however, the probability of the items in the data set must be known. In the traditional approach a counter is maintained for each item but this requires very large amounts of memory. The problem is exacerbated for more sophisticated methods which require the joint probabilities of two or three items which raises the memory cost even further.

In this paper we consider the use of *sketches* for estimating the underlying probabilities of large data sets. A sketch is a compact data structure capable of approximately summarising the frequencies of data. In particular we explore the two most widely discussed sketches: Count Sketch [4] and CountMin Sketch [5]. These sketches were designed for identifying the heavy-hitters, the most frequent items, within a data set but our aim is to examine their appropriateness for estimating the probabilities of all items.

The rest of this paper is organised as follows. In Section II we describe the related work and discuss how the two sketches we are using work. In Section III we describe why using sketches for probability estimation is not a straightforward choice. In Section IV we describe the experiments we conducted and present the results in Section V. Finally, we discuss our results in Section VI and draw some conclusions in Section VII.

II. Related Work

Sketches are designed as a form of synoptic data structure. The idea is that the sketch can provide a view of the data in a succinct manner thereby saving significant memory and improving processing times. Since sketches are only synopses, the responses they provide are only approximate and much of the work in this field is involved in the analysis of the guarantees that can be provided for the accuracy of the response. Another consequence of sketches being only synopses is that they are designed to answer only a small subset of possible queries and cannot be used to answer any given query on the data.

Sketches are normally accompanied with analysis proving bounds on their error and the probability of that error bound being exceeded. This is usually in the form of (ϵ, δ) pairs where ϵ is the error bound and δ is the probability that the error will be exceeded. The desired values of ϵ and δ determine the size of the sketch.

The first sketch was proposed in the foundational paper by Alon, Mattias and Szegedy [6]. It was this work that created the interest in sketch data structures and started research in this area. The original paper proposed a sketch (since referred to as the AMS sketch) designed to estimate the F_2 norm. The basic idea was to create a random variable whose expected value was equal to the desired quantity (the F_2 norm in this case) and by considering a number of independent variables a good approximation can be found.

This concept is the basis for all sketches and was utilised in the Count Sketch (CS) structure which was designed to approximately find the most frequent items in a data set [4]. The sketch is constructed from a set of counters with each counter tracking the count of a number of items. The counters are arranged into rows and each item is mapped to one counter per row so that whenever an items is seen, one counter per row is updated. Two hash functions are needed per row; the first to spread the items evenly across the counters and a second to determine whether an item increments or decrements the counter.

The first hash function, $h_j(i)$ assigns item i in row j to one of the counter $1 \ldots w$. The second hash function, $g_j(i)$ assigns item i in row j to $\{+1, -1\}$, indicating whether an items increments or decrements the counter. The value of a given counter k in row j is:

$$C[j,k] = \sum_{i=0}^{N} g_j(i) f(i) \mid h_j(i) = k \qquad (1)$$

where $f(i)$ is the true frequency of item i and N is the number of unique items. When recovering the approximate frequency the appropriate counter in each row is found and its value is multiplied by $g_j(i)$. The median of all counters is returned as the estimated frequency $\hat{f}(i)$. In some cases $\hat{f}(i)$ can be negative in which case the estimated frequency is 1.

It is provable that, $E[\hat{f}(i)] = f(i)\ \forall i$. Intuitively it is convenient to think of each counter as containing the true count of the item of interest together with some "noise". By sometimes incrementing and sometimes decrementing the counter and by spreading the items evenly across the counters, the expected value of the noise is zero because frequencies of items cancel each other. Of course the estimate has some variance and to provide (ϵ, δ) guarantees it has been proven that CS must have $O(1/\epsilon^2)$ counters per row and $O\log(1/\delta)$ rows [4]. A CS sketch of this size guarantees that with probability $1-\delta$ the estimate will be no more than $\epsilon\sqrt{F_2}$ larger or smaller than its true value. It should be noted that its error guarantees are not in terms of the true frequency but in terms of the L2 norm (the sum of squares of all items).

Cormode and Muthukrishnan adapted the CS sketch to greatly reduce the size of the sketch at the cost (in theory at least) of accuracy [5]. Their sketch, called the Count Min Sketch (CM), uses only one hash per row and always increments counters as illustrated in Fig. 1. The result is that the counters are biased estimators and do not have an expected value equal to the true frequency. In fact, the value of each counter is always equal to or greater than the true frequency. As its name suggests, CM provides frequency estimates by finding the smallest counter from all the rows which is guaranteed to be the one with the smallest overcount.

Analysis shows that CM can provide an (ϵ, δ) guarantee with only $O(1/\epsilon)$ counters per row and $O\log(1/\delta)$ rows [5]. The error, however, is in terms of the L1 norm (the sum of all items) which is usually greater than the root of the L2 norm. CM uses less space but provides a weaker guarantee of accuracy than CS.

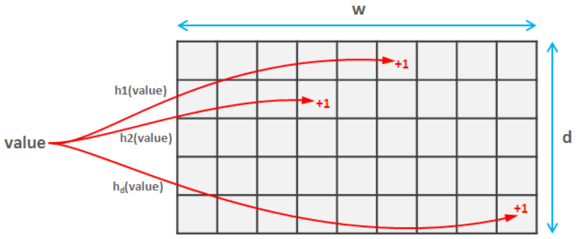


Fig. 1: The Count Min sketch combines the counts of multiple items and returns the counter containing the smallest overcount. Image taken from [7].

In practice the sketches usually perform better than the mathematical analysis suggests, see [8] for a thorough empirical examination of sketch performance. There has also been significant research into other forms of sketches and Cormode has provided an excellent survey recently [9].

III. Sketches for Probability Estimation

The CS and CM sketches were designed to identify the so-called "heavy-hitters", the most frequent items in a data set or stream. Although they can provide estimations for the frequency of all items, and indeed these sketches have been identified as "solving the frequency estimation problem" [8], it seems to have been implicitly understood that the estimations would be more accurate for the most frequent items.

This is intuitively understood using the "noise" concept. For both sketches each counter can be thought of as containing the true count of each item with some added noise. The most frequent items have, by definition, the highest true counts and therefore are least affected by the noise from the other items. So long as frequent items can be kept to separate counters then the estimates of their frequency will probably be good ones. By spreading the items evenly across the counters in each row and using a different spread in each row the probability of two frequent items always colliding to the same counter is greatly reduced leading to good estimates for these items.

In this paper, however, we consider the appropriateness of the sketches for estimating the underlying probability densities of all the data items. There are many applications where estimating the probability of an event or item is important both for its own sake in data processing and as a component in other calculations. Our work is primarily motivated by the use of probability estimation in information theoretic feature selection [1], [2] in which the probability of every item, and frequently of pairs or even triples of items, is needed.

The traditional method for calculating the probability of different items is to accurately count the frequency of each item and divide by the total frequency. In very large data sets the number of counters needed can be very large and this is made worse when the joint probability of two or three items are needed as is required for some feature selection algorithms.

Sketches that can provide estimates of the frequency of any item with fewer than one counter per item are attractive for this problem. It is not clear, however, how accurate they can be at estimating the frequency of lower frequency items and what impact errors of low frequency items have on information theoretic measures.

The aim of this paper is to examine the performance of sketches in estimating the probability densities of large data sets and attempt some evaluation of their appropriateness for this task. To the best of our knowledge this is the first examination of sketches for this purpose.

IV. Experimental Setup

In order to investigate the ability of sketches to estimate probability densities we created synthetic data sets, of varying sizes, with known distributions. We considered uniform, Gaussian and Zipfian distributions with various standard deviations and skews in the cases of Gaussian and Zipfian distributions respectively. For each distribution type we generated data sets of between 100 and 500,000 items.

Our aim is to consider the use of sketches to estimate probabilities using less memory space. Therefore we defined the size of the sketches not in terms of ϵ and δ but in relative terms to the traditional method of probability estimation. That is, we consider sketches that use approximately 100%, 75% and 50% as many counters as there are items. To do this we calculate the required number of counters in the sketches, $|C|$, and then find the value of ϵ and δ needed to create sketches with that number of counters. We assume that $\epsilon = \delta = x$ to provide a single unknown variable and then we can construct an equation, for each sketch, of the sketch size in terms of x. These are given by equations (2) and (3) for CS and CM respectively.

$$|C| = \log_2\left(\frac{1}{x} \times \frac{3}{x^2}\right) \tag{2}$$

$$|C| = \log_2\left(\frac{1}{x} \times \frac{2}{x}\right) \tag{3}$$

Equations (2) and (3) can be rearranged to find x, the value of ϵ and δ used in the sketches. Equations (4) and (5) give the results for CS and CM where $W(\cdot)$ is the Lambert W-function. This result is only approximate because the number of counters per row and the number of rows must be integer values and so x is too low. Therefore the calculated values of x are used as a starting point and is slowly incremented in 1% steps until the difference between the resultant sketch size and the required size is minimised.

$$x = exp\left(-0.5W\left(\frac{2|C|}{3}\right)\right) \tag{4}$$

$$x = exp\left(-W\left(\frac{|C|}{2}\right)\right) \tag{5}$$

To provide some context to the estimates from the sketches we also considered a naive sampling estimator which uses half as many counters as the traditional method. This estimator, which we call half-count (HC), records the accurate count of every second item only. When the count of an item is sought it returns the accurate count if it has recorded it and the count of the preceding item if it has not (it uses the next item if there is no preceding item).

For evaluating the estimates we compare each method (CS, CM and HC) to the traditional method which accurately records the count of every item. We utilise two measures for the difference in the probability estimations: the mean square error (MSE), equation (6), and the mean relative error (MRE), equation (7).

$$MSE = \frac{1}{N}\sum_{i=0}^{N}\left(f(i) - \hat{f}(i)\right)^2 \tag{6}$$

$$MRE = \frac{1}{N}\sum_{i=0}^{N}\frac{\left|f(i) - \hat{f}(i)\right|}{f(i)} \tag{7}$$

We would also like to consider how these differences combine and so we use the probability estimates to calculate the Shannon entropy, equation (8), (which relies on the probability of all items) and find the difference between the estimated entropy and the "true" entropy as calculated using the traditional method.

$$H(I) = -\sum_{i=0}^{N} p(i)\log(p(i)) \tag{8}$$

We also examined the error in entropy estimation when the memory space was reduced by one and two orders of magnitude(relative size of 10% and 1%). For these sizes we do not report MSE or MRE since the errors become either extremely small (for MSE) or extremely large (for MRE). For these results we also tune HC to the same relative size by counting every tenth or hundredth item only.

V. Results

A. Uniform

Fig. 2(a) shows the mean square error (MSE) for the sketches and HC in a uniform distribution. Since the distribution was perfectly uniform the HC method resulted in no errors at all because all items had the same frequency. The MSE for the sketches are also small and fall as the number of items increases. This is to be expected because MSE is an absolute measure and the more items there are the smaller the probability of all items. It is clear, also, that CM outperforms CS. With a relative size of 50%, the MSE of CM is between 97.43% and 99.04% lower than CS and the difference increases linearly with the logarithm of the number of items ($r = 0.802$, $p = 0.017$).

The results for mean relative error (MRE), Fig. 2(b), again show that HC has no error and that CM outperforms CS. MRE is statistically independent of the number of items for CS ($r \geq 0.099$) but falls with the number of items for CM ($r = -0.82$, $p = 0.013$ when the relative size is 50%). This results in a slightly increasing gap between the MRE of CM and CS ($r = 0.813$, $p = 0.014$) with CM having an error between 86.20% and 93.34% lower than CS. With a relative size of 50%, the average MRE for CS is 91.0% while for CM it is still significant at 8.21%.

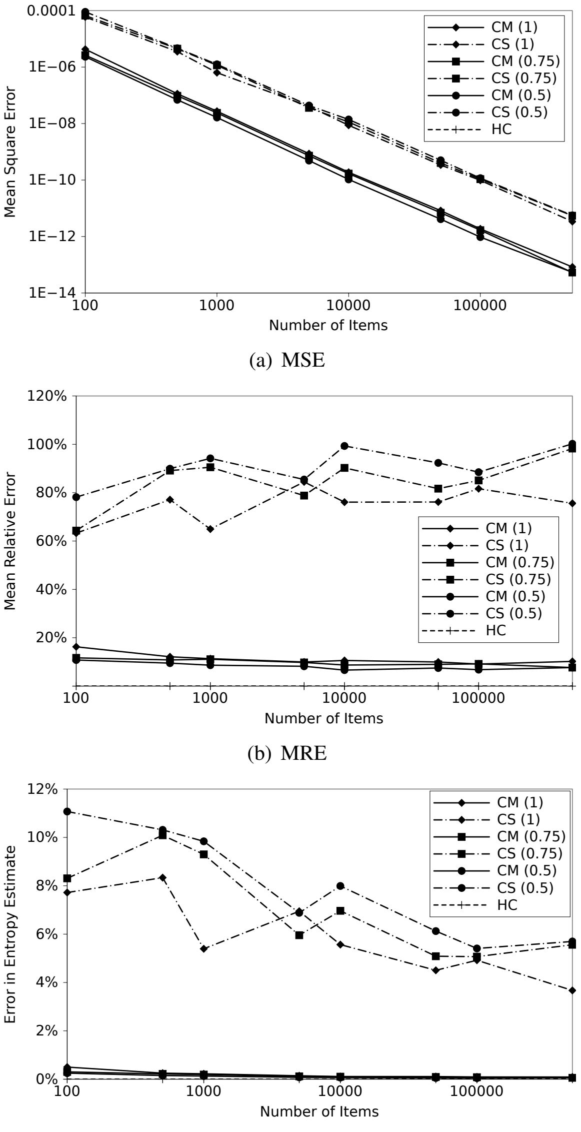
The results for entropy, Fig. 2(c), give an idea of the way the errors in individual frequencies combine. The results show that despite relatively large MRE, when all the items are combined the errors are much smaller. Once again HC has no errors and CM outperforms CS. The error in entropy estimation also falls with an increasing number of items because the individual probabilities become smaller and smaller meaning that errors in individual frequency estimates have an increasingly smaller impact on the entropy estimation. The average error for CS with a relative size of 50% is 7.92% and for CM it is only 0.1%.

B. Gaussian

For the Gaussian distribution there is a second factor aside from the impact of the sketch size, namely the standard deviation. Initially we fix the standard deviation at $\sigma = 2$ and vary the size and then fix the size at 50% and vary the standard deviation.

Fig. 3(a) shows that MSE falls as the number of items increases for the same reason as with the uniform distribution. Here, though, HC does not have zero error though it does have a lower error than CM or CS. CM again outperforms CS. When the relative size is 50% CM outperforms CS by an average of 98.73% and the difference between CM and HC is on average 97.0%. In both cases there is no statistically significant relationship between the differences and the number of items ($p \geq 0.067$).

A similar pattern emerges with MRE, Fig. 3(b). HC again has the lowest error, averaging just 1.14%, and CM outperforms CS. The error for CM falls as the number of items increases ($r = -0.786$, $p = 0.021$ for a relative size of 50%) but for CS there is no statistically significant dependence ($p \geq 0.059$). The difference between CM and CS therefore increases as the number of items increases ($r = 0.854$, $p = 0.007$ when relative size is 50%) while the difference between CM and HC falls ($r = -0.806$, $p = 0.016$ for the same size) although the smallest MRE under CM (with relative size of 50%) is still a significant 6.81%.



(a) MSE

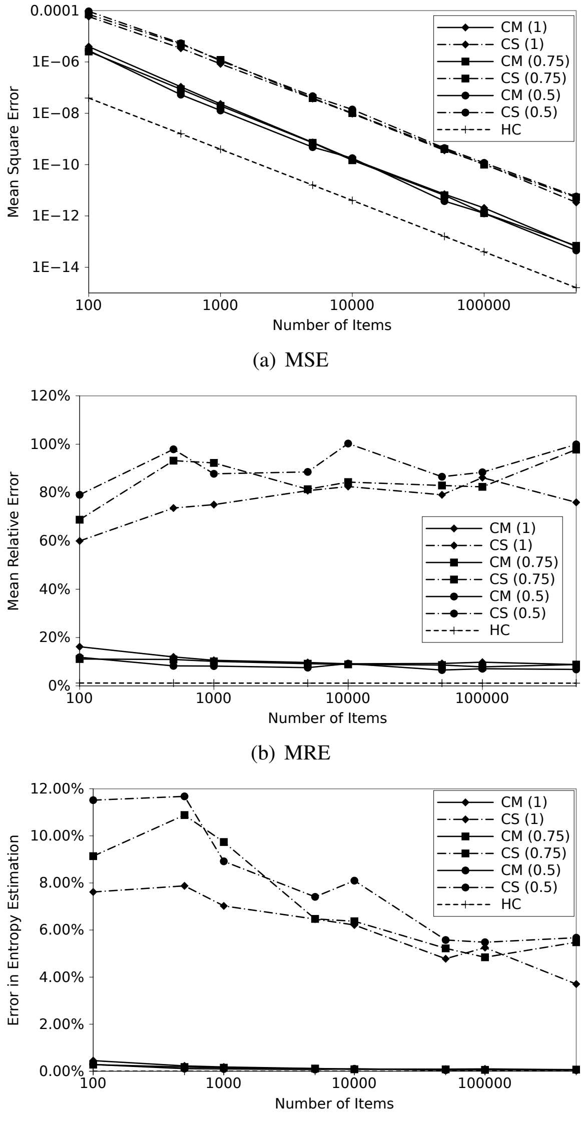
(b) MRE

(c) Entropy

Fig. 2: Performance of CM, CS and HC for a synthetic, perfect uniform distribution (relative size shown in brackets).

As with the uniform distribution, the error in entropy estimation is smaller than MRE for all methods, as shown in Fig. 3(c), and falls with the number of items. In fact, for HC the error in entropy estimation is correct up to at least 5 decimal places. For CM the average error is just 0.1% for relative size of 50% whereas it is 8.04% for CS for the same size.

Fixing the size at 50% and varying the standard deviation shows (Fig. 4(a)) that the standard deviation has little impact on MSE for the sketches but is significant for HC. As before, MSE falls with more items because it is an absolute measure but for HC the value is also significantly lower when the standard deviation is smaller. In the worst case for HC (when $\sigma = 3$), HC still outperforms CM by an average of 93.72% while CM outperforms CS by 98.6%. The gap between CM and HC is questionably dependent on the number of items ($r = -0.704$, $p = 0.051$) while



(a) MSE

(b) MRE

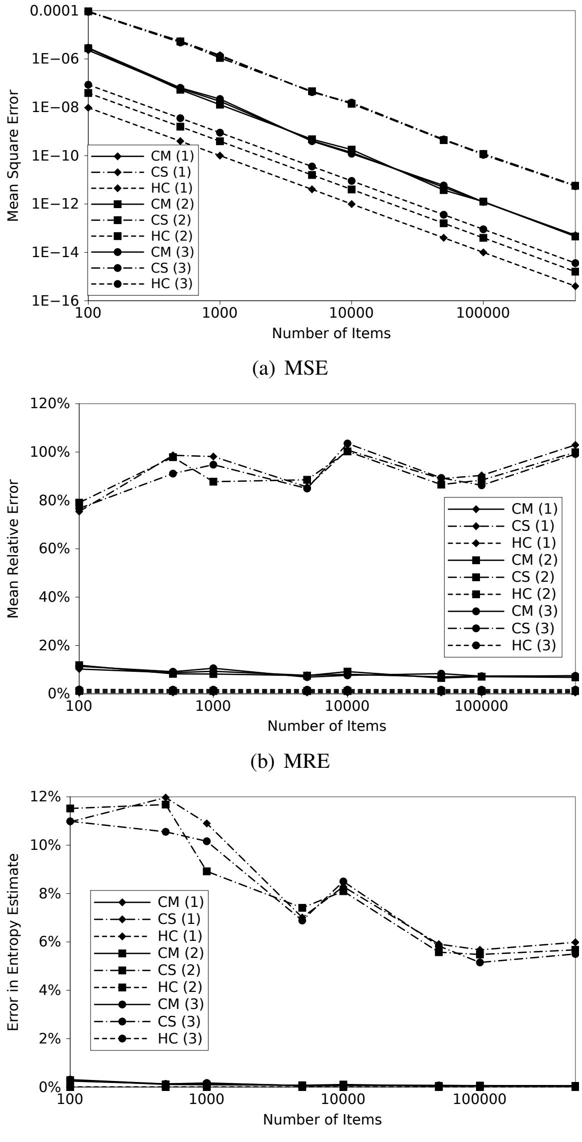
(c) Entropy

Fig. 3: Performance of CM, CS and HC for a synthetic Gaussian distribution with $\sigma = 2$ (relative size shown in brackets).

the gap between CM and CS statistically significantly increases with the number of items ($r = 0.716$, $p = 0.046$).

The results for MRE, Fig. 4(b), are similar to MSE. HC has the smallest error and this is smallest when σ is smallest. In the worst case ($\sigma = 3$) MRE for HC averages just 1.72% and this falls very slowly with more items. For CS, in the best case, the average MRE is 90.72% while for CM it is 8.19%. For CS there is no statistically significant dependence on the number of items ($p \geq 0.253$) while for CM there is a slight fall as the number of items increases ($r \geq -0.786$, $p \leq 0.023$).

When it comes to entropy estimation, Fig. 4(c), the results follow the same pattern as seen before. The errors are much smaller than MRE and fall as the



(a) MSE

(b) MRE

(c) Entropy

Fig. 4: Performance of CM, CS and HC for a synthetic Gaussian distribution with fixed relative size of 50% (σ is shown in brackets).

number of items increase. The estimate using HC is correct up to at least four decimal places while for CM the average error is 0.1% for all values of σ. For CS the error is between 7.95% and 8.34% but further experiments with more values of σ are required to determine whether there is a relationship between the error and standard deviation.

C. Zipfian

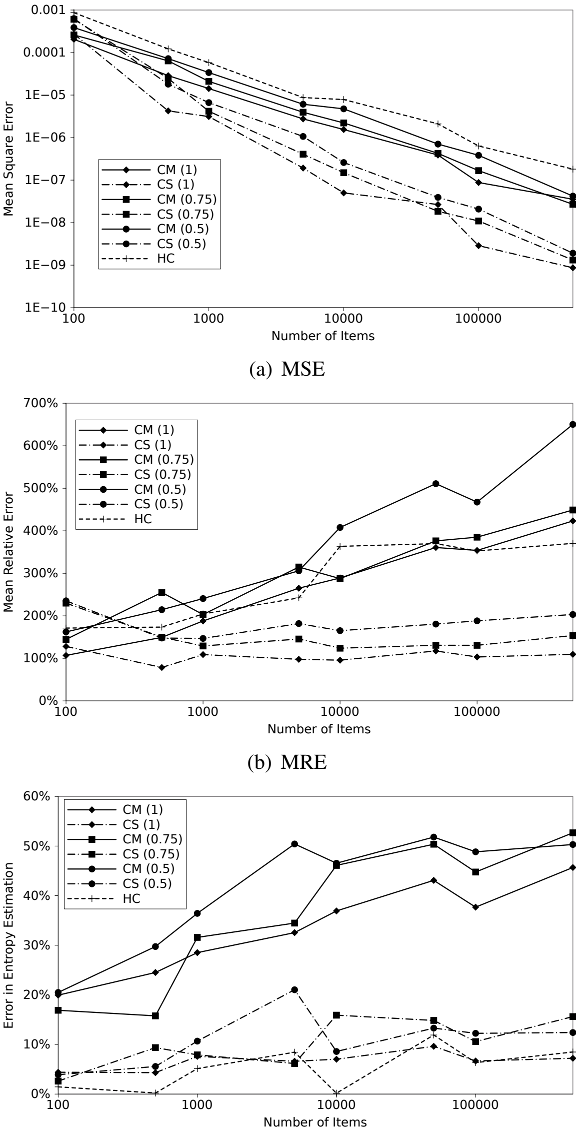
The results for the Zipfian distribution are in many ways very different to the previous results. Fig. 5(a) shows the results for MSE which, like with the other distributions, always decreases as the number of items increases. With the Zipfian distribution, however, HC has the highest error and CS the lowest. Furthermore, the difference between the errors with different size sketches is far more pronounced. MSE is, on average, 48.9% higher with HC than CM with relative size of 50% and the difference between CM and CS is on average 69.26% for the same size. In both cases any relationship between the difference and the number of items is doubtful ($r \geq 0.5$).

For MRE CS again has the lowest error but this time HC is similar to CM with a relative size of 100% and outperforms it when the size is 50%. The errors are extremely high when compared to the Gaussian distribution with the smallest error being 78.55% and the largest 650.23%. MRE is invariant with the number of items for CS ($p \geq 0.16$) but is larger for smaller sizes. For CM the error increases with the number of items ($r \geq 0.968$, $p \leq 2.98e-5$), the same relationship appears for HC ($r = 0.913$, $p = 0.0016$).

When it comes to the entropy estimation, however, while CS continues to outperform CM, HC has comparable error to CS. In fact, HC averages a 28.5% smaller error than the best case of CS. Unlike with the other distributions, the error in entropy estimation actually increases with the number of items for CM but for CS the relationship is doubtful with statistical significant for one size (75%) but not the others.

With a fixed relative size of 50% and varying skew the picture is slightly complex. For MSE, Fig.6(a), the skew has a large impact on the error with larger skews producing much smaller errors. In all cases HC performs similarly to CM with CS having the lowest error. For MRE, Fig. 6(b), the dependence on skew is still very evident but the order of performance changes. With low skew CS has the smallest error followed by HC then CM but as the skew increases CS performs increasingly badly relative to CM and HC. When the skew is 2.5, CM and CS have similar performance and HC is the best performing. When the skew increases to 3, CS now has the largest error followed by CM and then HC.

The story for the entropy estimation, Fig. 6(c), is different again. It remains the case that as the skew increases the errors decrease but this time HC always has the smallest error followed by CS and then CM. When the skew is 2 the errors for the sketches increase with the number of items ($p \leq 0.0038$) but when the skew is 3 the errors fall as there are more items ($p = 0.0008$) for CM but there is no statistical significance to the relationship for CS ($p = 0.659$). At low skew the average error for CM is 42.92%, 11.81% for CS and 6.07% for HC. When the skew is high the averages become 3.25%, 0.27% and 0.23% for CM, CS and HC respectively.



(a) MSE

(b) MRE

(c) Entropy

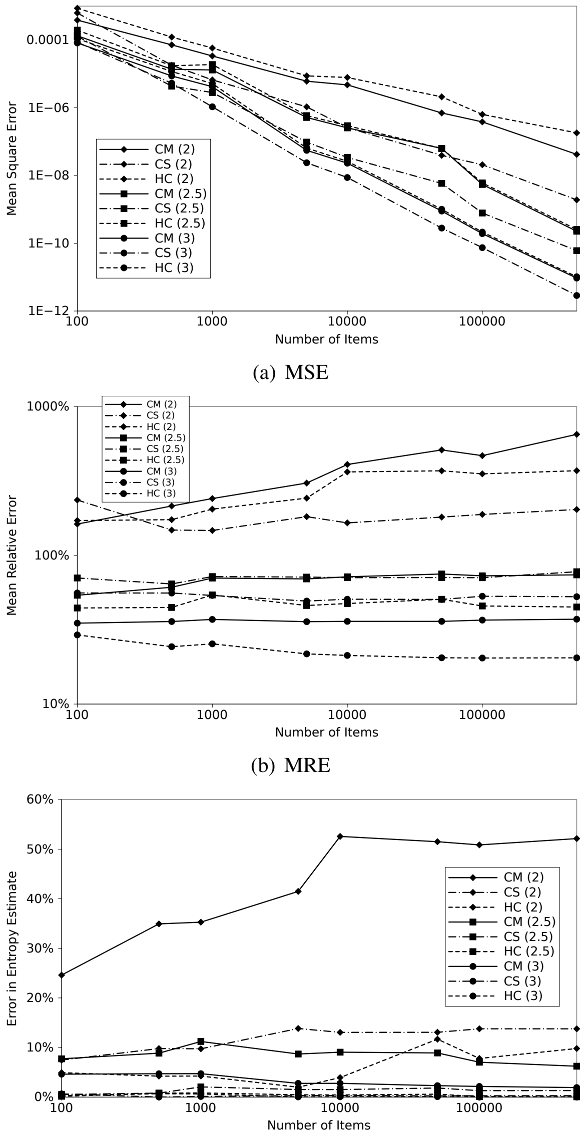
Fig. 5: Performance of CM, CS and HC for a synthetic Zipfian distribution with skew=2 (relative size shown in brackets).

D. Orders of Magnitude

The results for the uniform distribution, Fig. 7(a), show that CS has the largest errors followed by CM with HC having no errors at all. The errors for CM are virtually zero for all magnitudes, with the average across all data set sizes being just 0.22% with a relative size of 1%. In comparison, the average error for CS is 10.15% for the same size and is a significant 5.88% when no memory savings are being made.

A similar pattern is seen for the Gaussian distribution in Fig. 7(b). Again HC has no error and the error for CM is an average of 0.22% even with just 1% relative size. CS has slightly higher errors with the Gaussian distribution compared to a uniform one at 10.87% for 1% relative size and 6.12% for 100% relative size.

Things change dramatically in the Zipfian distribution shown in Fig. 7(c). Here CM has the largest error, averaging between 7.94% with 100% size and 8.90%



(a) MSE

(b) MRE

(c) Entropy

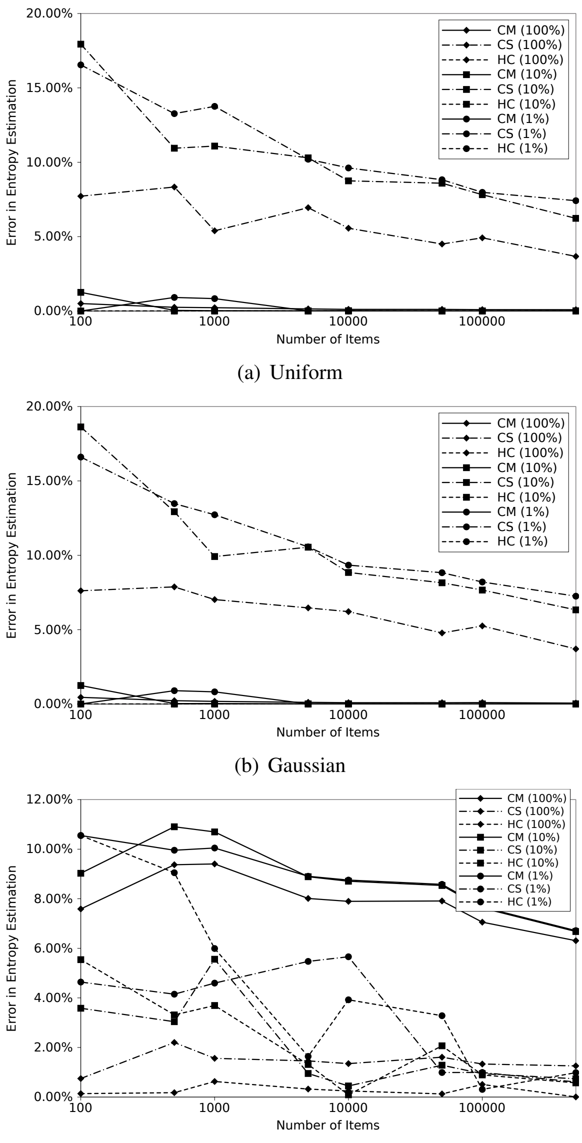
Fig. 6: Performance of CM, CS and HC for a synthetic Zipfian distribution with fixed relative size of 50% (skew shown in brackets).

with 1% size. HC and CS have lower errors with CS outperforming HC only at smaller sizes. For 100% size CS has an average error of 1.44% and HC has 0.27% but the averages rise to 3.39% and 4.47% at 1% relative size.

VI. Discussion

The results presented in the previous section provide the first examination of the performance of sketches for probability estimation. The results show that there is a significant performance difference between uniform and Gaussian distributions on the one hand and Zipfian distributions on the other.

In a perfect uniform distribution CM performs extremely well with virtually no errors whereas CS has significant errors. The frequencies estimated by CM are the sum of a number of counters and therefore when all counters have the same frequency CM will perform very well so long as the items are evenly spread among the columns by the hash functions. The more items there are the better the spread is likely to be and small fluctuations in the number of items hashed into each column have less impact on the total counts. Thus CM will overestimate the frequencies but will correctly find the correct probabilities.



(a) Uniform

(b) Gaussian

(c) Zipfian

Fig. 7: Performance of CM, CS and HC for different synthetic data sets with relative sizes decreasing by two orders of magnitude (relative size shown in brackets).

In contrast, for CS the expected value of all counters would be zero but small fluctuations caused by an uneven spread has a larger impact. It can happen that the frequency estimated by CS is negative in which case there is no information available about the true frequency except the lower-bound that the item was seen. If the true frequency is high then the discrepancy can become very large. This cannot happen with CM.

The results for the Gaussian distribution are very similar to those of the uniform distribution but the errors are a little larger. This is because the true frequencies in the Gaussian distribution do not vary by very large amounts. Increasing the standard deviation lowers the errors because the frequencies become even more similar.

With the Zipfian distribution, however, the frequencies are scale-free. That is, there are very large differences in the true frequencies with a small number of items appearing a very large number of times and most items appearing only once. For these distributions CM has the largest error because the heavy hitters can have a very large impact on a large number of items. This is particularly true as the sketches become smaller and it is harder to avoid collisions with the most frequent items. On the other hand CS can detect when an item has collided with a heavy hitter because its frequency is estimated as a negative number. When this happens the frequency used is just one which is likely to be close to the true count for a non-heavy-hitter. CS therefore only needs to avoid heavy-hitters colliding with other heavy-hitters too often which is easier to do.

Although some measures have been used in this paper to examine the performance of sketches for probability estimation they are not objective. The best that can be hoped for is to compare one or more estimators and consider their performance relative to each other but there can be no objective method of declaring an estimator to be "good enough". The question of whether an estimator is good enough depends entirely on what it will be used for. We have therefore provided a straightforward estimator (HC) for comparison which we hope places the performance of the sketches into some context but we remain cautious about drawing strong conclusions from these results.

VII. Conclusion

In this paper we have considered the use of sketches for estimating the probability of all items in very large data sets. The sketches are memory-efficient data structures designed for identifying and estimating the frequency of so-called heavy-hitters. They are, however, in theory capable of estimating the frequency of all items. To the best of our knowledge this is the first attempt to examine their applicability to this problem.

In order to gain some understanding of the performance of the sketches we considered three metrics: mean square error, mean relative error and the error in entropy estimation. Our results show that for uniform and Gaussian distributions the CM sketch performs very well even with very small sketch sizes whereas CS performs less well, estimating the entropy incorrectly by an average of over 10%. For Zipfian distributions, however, it is CS that performs best with increasingly better performance as the data set becomes more skewed.

The measures we have used are not objective and, inherently, it is impossible to declare an estimator "good enough" without reference to a specific task. Thus the results in this paper are illustrative and suggestive only. Additionally we considered only pure distributions which may not be reflected in real data. In particular data may be mixtures of different distributions.

We do conclude, however, that the sketches produce estimates with smaller errors than might seem likely from their construction which suggests that, for some applications at least, sketches need not be restricted only to the problem of finding the most frequent items.

In the future we would like to examine how the sketches perform with real data sets and whether even small errors (as seen in the entropy estimations) might affect the ranking of items for feature selection algorithms.

Acknowledgment

Anthony Kleerekoper gratefully acknowledges funding from the EPSRC APT Platform Grant (EP/G013500/1). Dr. Luján is supported by a Royal Society University Research Fellowship.

References

[1] Gavin Brown. A new perspective for information theoretic feature selection. In *International Conference on Artificial Intelligence and Statistics*, pages 49–56, 2009.

[2] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[3] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003.

[4] Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *Automata, Languages and Programming*, pages 693–703. Springer, 2002.

[5] Graham Cormode and S Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

[6] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 20–29. ACM, 1996.

[7] Ilya Katsov. Probabilistic data structures for web analytics and data mining, 2012.

[8] Graham Cormode and Marios Hadjieleftheriou. Methods for finding frequent items in data streams. *The VLDB Journal*, 19(1):3–20, 2010.

[9] Graham Cormode. Sketch techniques for approximate query processing. *Foundations and Trends in Databases. NOW publishers*, 2011.