Streaming Algorithms for Chi-Square and Kolmogorov-Smirnov Tests using Quantile Sketch

Shantam Maheshwari, Dr. Debajyoti Bera

August 2019

1 Summary

This report discusses streaming algorithms for performing Pearson's Chi-Square, and the Kolmogorov-Smirnov goodness-of-fit tests on streaming data with the aid of an ϵ-approximate quantile-summary, a data structure that can answer quantile queries on a sequence of N elements within a precision of ϵN.

2 Introduction

The Chi-Square test and Kolmogorov-Smirnov test (hereinafter referred to as KS test) are two of the most widely used tests in statistics. Both have two variants - a one-sample and a two-sample test - wherein the one-sample variant checks whether a given set of data belongs to a fixed known distribution, while the two-sample variant checks whether two sets of data belong to a same fixed known distribution. Since only the one-sample variants find use in our current area of research that is randomness testing, the discussion has been limited to the same.

As real-time data is most often present in the form of data streams, storing or batch-processing it is no longer a sustainable option. Moreover, batch sampling of data often allows local anomalous behaviour to go unnoticed. The streaming models of computation thus emergent from this need for real-time updation and space-efficient processing have been extensively studied. However, most of these models lack adoption due to the time investment required for learning, implementing and parameter-tuning.

Fortunately these tests are both non-parametric, i.e., they do not assume the size of the input nor the distribution from which the data streams have been taken. A state-of-the-art quantile summary data structure proposed by Greenwald and Khanna[1], "Quantile Sketch", has therefore been used by Ashwin Lall[2] for computaton of KS statistic in $\Theta(\sqrt{N}\log N)$ space complexity and by Farrow et al.[3] for computation of Chi-Square statistic in $O(K^2\sqrt{N}\log N)$ space complexity, where N denotes the size of the data stream and K denotes the number of bins.

While reviewing the current literature of tests on randomness, we have further noted the fact that the Chi-Square test and the KS test form the basis for majority of the tests found in popular standards and randomness test suites like DieHarder[4], NIST[5], ENT[6], TestU01[7] etc. which are used to determine suitability of random number generators(RNGs) for cryptographic and even military purposes. Hence a discussion of these streaming algorithms deserves importance.

3 Related Work

Wang et al.[8] did an experimental comparative study on various streaming algorithms for computation of quantiles over data streams. The candidates algorithms included Greenwald and Khanna's quantile sketch[1] ($O(\frac{1}{\epsilon}\log(\epsilon N))$ space), Shrivastava et al.'s q-digest [9]($O(\frac{1}{\epsilon}\log u$ space), Manku et al.'s algorithm [10] ($O(\frac{1}{\epsilon}\log^2(\frac{1}{\epsilon})$ space) and *Random*, their own simplified version of Manku et al.'s algorithm($O(\frac{1}{\epsilon}\log^{1.5}(\frac{1}{\epsilon})$ space). The latter two were found to perform better than GK sketch, which still proved to be quite competitive. Gan et al. of Stanford InfoLab proposed *Moment Sketch* that further bested the GK sketch.

However, as noted in Greenwald and Khanna's paper, the quantile sketch has been found to not grow with size of the data stream for random input. This suggests that the GK sketch is still the best choice for randomness testing.

Knuth's algorithm [11] for the KS test required priori knowledge of size N of input and required presorting of the data, thus rendering a time complexity of $O(N\log N)$. Sahni et al.[12] came up with a new linear time algorithm. However, no true on-line streaming(no priori knowledge of size of input) algorithm exists other than the one presented here. The same can also be said about the Chi-Square test[3].

4 Problem Definition

The one-sample variants of the Chi-Square and KS tests check whether a given set of data follows a fixed known distribution. Thus, the null and alternative hypotheses are:

H_o: The sample follows the fixed known distribution.

H_a: The sample does not follow the fixed known distribution.

4.1 One-Sample Chi-Square Test

The data is binned into mutually exclusive ranges and the observed frequencies of each bin are then compared with

the expected frequencies. Let N denote the size of the stream, K denote the number of bins (categories). For $1 \leq i \leq K$, let O_i denote the observed frequency and E_i denote the expected frequency for the i^{th} bin.

The Chi Square statistic is then given by:

$$\chi^2 = \sum_{i=1}^{K} \frac{(O_i - E_i)^2}{E_i}$$

This statistic is compared with the critical value from the Chi-Square distribution at a significance level α, where $0 < \alpha < 1$, and $K - 1$ degrees of freedom (since E_K can be computed if $E_1, ..., E_{K-1}$ are known). The null hypothesis is rejected if $\chi^2 > \chi^2_{1-\alpha,K-1}$.

4.2 One-Sample KS Test

The cumulative distribution function (c.d.f.) of a distribution is defined as the function $F(x) = Pr(X \leq x)$ where X is a random variable drawn from the distribution. The empirical distribution function of n independent observations $X_1, ..., X_n$ is given by $F_n(x) = \frac{\{i|X_i \leq x\}}{n}$. Both of these functions are defined over $\mathbb{R}$ and take values in the range $[0, 1]$.

The KS statistic, given by

$$D_n = \sup_x |F_n(x) - F(x)|$$

gives the maximum absolute difference between the empirical distribution and the c.d.f. it is being tested against.

The statistic $\sqrt{n}D_n$ is then compared to the critical value K_α from the KS distribution at a significance level α, where $0 < \alpha < 1$, and the null hypotheses is rejected if $\sqrt{n}D_n > K_\alpha$.

5 Methodology

5.1 Quantile Sketch

The algorithms discussed below make use of an *ϵ-quantile sketch*, a streaming data structure proposed by Greenwald and Khanna[1], that, given an input data stream of size N, say $X_1, ..., X_N (X_1 \leq X_2 \leq ... \leq X_N)$ in any arbitrary order, can be queried to return a value X_i as the approximate r-ranked element $[1 \leq r \leq N]$ such that $i \in [r - \epsilon N, r + \epsilon N]$.

5.2 KS Test Algorithm

The following lemmas are required for the analysis of the algorithm, proofs for which can be found in Lall's paper[2].

Algorithm 1: One-Sample KS Test

Data: ϵ-Quantile sketch $Q(X_1 \leq X_2 \leq ... \leq X_N)$ of a stream of size N, and a c.d.f. F
Result: D', an estimation of KS statistic D

```
D' ← 0
for j ← 1 to k do
    let i'_j be approx. index of X_{i_j} as computed in Lemma 5.2
    E' ← |i_j/N − F(X_{i_j})|
    D' ← max(D', E')
end
```

Lemma 5.1. *It is possible to extract from a ϵ-quantile sketch a subset $\{X_{i_1}, ..., X_{i_k}\} \subseteq \{X_1, ..., X_N)\}$ (where $X_1 \leq X_2 \leq ... \leq X_N$) such that $i_1 \leq i_2 \leq, ..., \leq i_k$ and $\forall j \in [1, k), i_{j+1} - i_j < 2\epsilon N$.*

Lemma 5.2. *Given some value X_i returned by a ϵ-quantile sketch (where X_i is the i^{th} largest element in the input), it is possible to estimate i to within ϵN additive error.*

5.3 Chi-Square Test Algorithm

The choice of the number of bins K can have a significant impact on the quality of our test. A general rule of thumb is to choose K so as to have at least a count of five in each bin[11].

Farrow's paper claims that the following algorithm not only doesn't assume the value of K, it also guarantees an approximate equidistribution among the bins so that each bin has high expected counts. However, as seen in the screenshot of their code [1] attached below, the number of bins has to be added manually by the user.

```
100     int sample_size;
101     cout << "Enter the memory capacity for the sample size: "<<endl;
102     cin >> sample_size;
103
104     int k;
105     cout << "Enter the number of bins you want to use: " <<endl;
106     cin >> k;
107
108     double mean, SD;
109     cout << "Enter the mean of the normal distribution you want to compare again
110     cin >> mean;
```

The following theorem is required for the analysis of the algorithm, proof for which can also be found in Lall's paper[2].

Theorem 5.3. *For any value $x \in \mathbb{R}$, it is possible to compute the fraction of the stream that has value less than x to within 3ϵ error using an ϵ-quantile sketch of the stream.*

[1] https://github.com/ashlall/StreamStats/blob/master/Test/Chi_Square_test.cpp

Algorithm 2: One-Sample Chi-Square Test

Data: ϵ-Quantile sketch Q of a stream of size N; K number of bins; c.d.f. F; significance level α

Result: D', Truth of null hypothesis

1 $\hat{X}^2 \leftarrow 0$
2 $E_i \leftarrow \frac{N}{K}$
3 **for** $i \leftarrow 1$ **to** K **do**
4 let $l \leftarrow F^{-1}(\frac{i-1}{K})$
5 let $u \leftarrow F^{-1}(\frac{i}{K})$
6 let $\hat{i}_l$ be approx. fraction of stream less than l^*
7 let $\hat{i}_u$ be approx. fraction of stream less than u^*
8 $\hat{O}_i \leftarrow N(\hat{i}_u - \hat{i}_l)$
9 $\hat{\lambda}_i \leftarrow |\hat{O}_i - E_i|$
10 **if** $\hat{\lambda}_i > 2\sqrt{N}$ **then**
11 return *true*
12 $\hat{X}^2 \leftarrow \hat{X}^2 + \frac{\hat{\lambda}_i^2}{E_i}$
13 let $X^2_{1-\alpha,K-1}$ be critical value at significance level α
14 **if** $\hat{X}^2 > X^2_{1-\alpha,K-1}$ **then**
15 return *true*
16 **else**
17 return *false*
18 **end**
19 **end**

* : as computed using **Theorem 5.3**

6 Computational Results

Here we have listed some of the computational results of these algorithms that have the most relevance to our study of Randomness Testing. We shall skip the proofs and summarize the results as their detailed analysis can be found in the original papers.

The Greenwald-Khanna ϵ-quantile sketch takes at most $O(\frac{1}{\epsilon}\log(\epsilon N))$ space and time per update, unassuming of the size N of the input stream and independent of the relative order of the elements[1], thus constituting the bulk of the computation of the KS algorithm, the rest of which can be calculated offline and relatively quickly.

In **Algorithm 1**, say if there are $s = O(\frac{1}{\epsilon}\log(\epsilon N))$ unique values stored in the sketch, we iterate s times and perform binary search in $O(\log s)$ (see line 3 of Algorithm 1) time complexity, thus rendering a running time of $O(s \log s)$, much lesser than initial query operations to extract the values X_{i_j} from the quantile sketch ($O(N)$)[2].

In **Algorithm 2**, computation of $\hat{i}_l$ and $\hat{i}_l$ takes the bulk of the time, again requiring binary search taking $O(\log N)$ time over K iterations, thus rendering a running time of $O(K \log N)$.

7 Concluding Remarks

As seen in the previous section, quantile sketch offers significant improvements in the time and space complexities for computation of the Chi-Square and KS test statistics. Moreover, it has been empirically found that its space usage does not grow with the size of the input stream for random input[1]. Since Chi-Square and KS test form the basis for majority of the tests for randomness, these algorithms can improve randomness testing by orders of magnitude, both in terms of space and time complexities.

Furthermore, finding the use of streaming algorithms for computation of popular statistics lacking in our survey of the current literature, quantile sketches offer new-found unexplored avenues for design of algorithms for the same.

References

[1] Michael Greenwald, Sanjeev Khanna, et al. Space-efficient online computation of quantile summaries. *ACM SIGMOD Record*, 30(2):58–66, 2001.

[2] Ashwin Lall. Data streaming algorithms for the kolmogorov-smirnov test. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 95–104. IEEE, 2015.

[3] Emily Farrow, Junbo Li, Farfan Zaki, and Ashwin Lall. Accessible streaming algorithms for the chi-square test. Denison University.

[4] Robert G Brown, Dirk Eddelbuettel, and David Bauer. Dieharder: A random number test suite. *Open Source software library, under development, URL http://www. phy. duke. edu/˜ rgb/General/dieharder. php*, 2013.

[5] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, and Elaine Barker. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, Booz-Allen and Hamilton Inc Mclean Va, 2001.

[6] John Walker. Ent: a pseudorandom number sequence test program. *Software and documentation available at/www. fourmilab. ch/random/S*, 2008.

[7] Pierre L'Ecuyer and Richard Simard. Testu01: Ac library for empirical testing of random number generators. *ACM Transactions on Mathematical Software (TOMS)*, 33(4):22, 2007.

[8] Lu Wang, Ge Luo, Ke Yi, and Graham Cormode. Quantiles over data streams: an experimental study. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 737–748. ACM, 2013.

[9] Nisheeth Shrivastava, Chiranjeeb Buragohain, Divyakant Agrawal, and Subhash Suri. Medians and beyond: new aggregation techniques for sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 239–249. ACM, 2004.

[10] Gurmeet Singh Manku, Sridhar Rajagopalan, and Bruce G Lindsay. Approximate medians and other quantiles in one pass and with limited memory. *ACM SIGMOD Record*, 27(2):426–435, 1998.

[11] Donald E Knuth. *Art of computer programming, volume 2: Seminumerical algorithms*. Addison-Wesley Professional, 1997.

[12] Teofilo F. Gonzalez, Sartaj Sahni, and William R. Franta. An efficient algorithm for the kolmogorov-smirnov and lilliefors tests. *ACM Trans. Math. Softw.*, 3(1):60–64, 1977.