Order statistics and estimating cardinalities of massive data sets

Frédéric Giroire

Order Statistics and Estimating Cardinalities of massive Data Sets

Frédéric Giroire [1]

ALGO project, INRIA Rocquencourt, B. P. 105, 78153 Le Chesnay Cedex, France; and MASCOTTE, joint project CNRS-INRIA-UNSA, 2004 Routes des Lucioles, BP 93, F-06902, France.

Abstract

A new class of algorithms to estimate the cardinality of very large multisets using constant memory and doing only one pass on the data is introduced here. It is based on order statistics rather than on bit patterns in binary representations of numbers. Three families of estimators are analyzed. They attain a standard error of $\frac{1}{\sqrt{M}}$ using M units of storage, which places them in the same class as the best known algorithms so far. The algorithms have a very simple internal loop, which gives them an advantage in term of processing speed. For instance, a memory of only 12kB and only few seconds are sufficient to process a multiset with several million elements and to build an estimate with accuracy of order 2 percents. The algorithms are validated both by mathematical analysis and by experimentations on real internet traffic.

Key words: cardinality, estimates, very large multiset, traffic analysis

1 Introduction

Problem. A multiset is a set where each element can appear several times. The *cardinality* n of the multiset is the number of distinct elements, while the *size* N of the multiset is the total number of elements, counting the repetitions. An important issue in computer science is to estimate the cardinality of a multiset having a very large size. This problem has arisen in the 1980's, motivated by optimisations of classical algorithmic operations on data bases

Email address: frederic.giroire@inria.fr (Frédéric Giroire).
URL: http://algo.inria.fr/giroire/ (Frédéric Giroire).

[1] Partially supported by the European FET project AEOLUS.

(union, intersection, sorting,...). As the data sets to be measured have mostly a very large size N, far beyond the RAM capacities, a natural requirement is to treat the data in one pass using a simple loop, and with a small auxiliary memory (constant or logarithmic in N). More recently, in the past decade, the problem of counting distinct elements has appeared as a crucial algorithmic operation in the context of networking with the development of networks of very large capacity. Typically, the elements are *packets*, each packet belonging to a *flow* (also called connection) identified by a source address and a destination address. Estimating the number of distinct flows in a data stream has many applications in network monitoring and network security, see the detailed survey of Estan, Varghese and Fisk [7]. For instance, one can count the number of distinct flows on a traffic to detect Denial of Service attacks, where abnormally many distinct connections are opened in a short period of time. Other applications include the data mining of language texts [2, 3] or biological data [16, 17]. The crucial point to solve the problem, first developed by Flajolet and Martin [10] in their algorithm PROBABILISTIC COUNTING, is to relax the constraint of giving the exact number of distinct values in the multiset. For most applications, a *probabilistic estimate* of n with good precision is sufficient.

The three families of algorithms. In this article, new estimators, based on *order statistics*, are introduced to solve the problem of estimating the cardinality of very large multiset while using *constant memory* and performing a *single pass* on the data. In addition, *no assumption* is made on the structure of the data.

We assume that a *hash function* h mapping an element to a real value that "looks like" uniformly distributed in the interval $[0, 1]$ is given (see the detailed study of Knuth [19]). Let $S = (e_1, \ldots, e_N)$ be a set of elements and let n be the number of distinct elements in S. Under the assumption on the hashed function h, and without making any assumption on the nature of the repetitions, the set $(h(e_1), \ldots, h(e_N))$ of hashed values can be considered as built from n real values taken independently uniformly at random in $[0, 1]$, and then replicated and permuted in an arbitrary way. Such a set of uniform random values in $[0, 1]$ with arbitrary replications and order of appearance is called an *ideal multiset*. Thus, estimating the number of distinct elements in a real multiset without making any assumptions on the repetitions amounts to estimating the cardinality of an ideal multiset.

The crucial idea is then that the minimum value in an ideal multiset does not depend on the replication structure of the data nor on the ordering, and gives an indication on the number n of distinct values of the multiset (basically, the minimum of n independent uniform values on $[0, 1]$ has more chances of being small if n is large). More precisely, the expectation of the minimum is $\frac{1}{n+1}$. To obtain an estimate of n, the most natural way would be to invert

this minimum, but the inverse happens to have an infinite expectation. To overcome this difficulty, our solution is to build estimates using the inverse of the k-th minimum —instead of the first minimum— composed with a sublinear function, as logarithm or square root. It gives *three families of estimates* of n: the Inverse, Logarithm and Square Root Families. The estimates are then combined with a *stochastic averaging process*, as introduced by Flajolet and Martin in [10]. Stochastic averaging consists in simulating the effect of m experiments on the multiset and then averaging an observable over the m experiments to obtain an estimate with a good precision.

Related work. There has been substantial work on approximate query processing in the database community, see [14, 13, 5]. In [20] Whang, Zanden and Taylor introduced Linear Counting. The principle is to distribute hashed values into buckets and use the number of hit buckets to give an estimate of the number of values. A drawback of this method is that memory is still linear (but with a small constant). To extend it to very large data sets, Estan, Varghese and Fisk proposed in [7] a multiscale version of this principle in their Multiresolution Bitmap algorithm. The algorithm keeps a collection of windows on the previous bitmap. Its estimate has a standard error of $4.4/\sqrt{m}$ while using m words of memory. Another way that has been proposed to estimate cardinality is sampling. The idea is to keep only a fraction of the values that have been read. For instance, in Wegner's Adaptive Sampling this fraction is dynamically chosen in an elegant way. The algorithm has been described and analyzed by Flajolet in [8] and its accuracy is $1.20/\sqrt{m}$. The Probabilistic Counting algorithm of Flajolet and Martin, in [10], uses bit patterns in binary representations of numbers. It has excellent statistical properties with an error close to $0.78/\sqrt{m}$. In [6], the LogLog Counting algorithm of Durand and Flajolet starts from the same idea but uses a different observable. The standard error is $1.30/\sqrt{m}$, but the m words of memory have here a size of order $\log\log n$ and not $\log n$. The same is true for the algorithm HyperLogLog, introduced in [9], based on the harmonic mean rather than the geometric mean, which attains a precision of $1.04/\sqrt{m}$, giving it the best known ratio precision over memory. Finally in [1] the authors present three algorithms to count distinct elements. The first one uses the k-th minimum and corresponds basically to the inverse family estimator. The authors prove that this algorithm $(\epsilon, \delta)-$approximates n using $O(1/\epsilon^2 \log m \log(1/\delta))$ bits of memory and $O(\log(1/\epsilon) \log m \log(1/\delta))$ processing time per elements. In this paper, we generalize this idea by introducing new and more efficient families of estimates and provide a precise analysis. A short version of this work can be found in [15]. Finally, in [4], Chassaing and Gerin propose an other family of estimators and prove its optimality using information and estimation theory.

Results. The three families of estimators are presented in Section 2 and analyzed in Section 4. The main results are presented before the analysis in Section 3. We found *asymptotically unbiased* estimates of n for the three fami-

lies and give their *standard error* in Theorem 1. We then compare the tradeoffs between *accuracy* and *memory* requirement for these families in Theorem 2. We show that, with a fixed amount of memory, M, the precision improves when k increases and that better estimates are obtained when applying sublinear functions. Nevertheless, the three families have an optimal trade-off of $1/\sqrt{M}$. In addition, we propose a *best practical estimate*, MINCOUNT. Using an auxiliary memory of only 12kB, it succeeds in estimating with accuracy of order *2 percents* the cardinality of a multiset with *several million elements*. Note that the algorithms can be adapted to operate on sliding windows [12].

Validation and experimentations. The estimates of the three families are validated using trace files of different kinds (e.g. english texts or router traces) and sizes. The relative error of the estimates is shown to be close to what expected from theory (see Figure 4 in Section 5). We also show that the algorithms are very fast: our implementation takes only few seconds to process files with millions of elements and is only 3 to 4 times slower than the very simple `unix` command `cat -T`, that just replaces the tab characters of a file by ^I (see Figure 6). This is of critical value in the context of in-line analysis of internet traffic where we have only of few tens of machine operations at disposal to process a packet. In Section 5.4, we show how the algorithm MINCOUNT can be used to detect some attacks on a network, e.g. the spreading of the worm Code Red.

2 Three families of estimates

In this section, we present three families of algorithms, the Inverse, Square Root and Logarithm Families, to estimate the cardinality of very large multisets.

Construction of estimators based on the minimum M. Recall from the previous section that we assume at our disposal a hash function h mapping any element to a real value that "looks like" uniformly distributed in the interval $[0, 1]$ —the construction of this hash function may be based on modular arithmetic as discussed by Knuth in [19]. Given any multiset, we can transform it to an ideal multiset (defined previously) by using such a hash function on the elements in the multiset. Thus, the problem of estimating distinct items in a multiset is equivalent to estimating the cardinality of an ideal multiset.

To estimate the number of distinct elements, denoted by n, of an ideal multiset, we consider its minimum, M. An important remark is that the minimum of a sequence of numbers is found with a single pass on the elements and that it is not sensitive to repetitions. The density of the minimum of n uniform random variables over $[0, 1]$ is $\mathbb{P}(M \in [x, x + dx]) = n(1 - x)^{n-1}dx$. So, for $n \geq 1$, its

expectation is

$$\mathbb{E}[M] = \int_0^1 x \cdot n(1-x)^{n-1} dx = \frac{1}{n+1}.$$

M is roughly an estimator of $1/n$. Our hope is now to be allowed to take $1/M$ as an estimate of n. But

$$\mathbb{E}\left[\frac{1}{M}\right] = \int_0^1 \frac{1}{x} \cdot n(1-x)^{n-1} dx = +\infty$$

Unfortunately, the integral does not converge near 0 and is unbounded. In order to obtain an estimate of n we use *indirectly* the minimum M in the following ways.

(1) Instead of using the inverse function alone, we compose it with a sublinear function f, e.g. the (natural) *logarithm* and the *square root*.
(2) Instead of using the first minimum, we take the second, third or more generally the k-th minima.

Thus we obtain three families of estimates namely the *Inverse Family*, the *Square Root Family* and the *Logarithm Family*. We talk about families as we have one estimator per value of k. Their pseudo-code is given in Figure 1.

Simulating m experiments. The precision of the algorithms is given by the *standard error* of their estimate ξ, denoted by $\mathbb{SE}[\xi]$ and defined as follows

$$\mathbb{SE}[\xi] := \frac{\sigma(\xi)}{n},$$

where $\sigma(\xi)$ denotes the standard deviation of ξ. To improve the precision of the algorithms, we would like to average over several similar experiments, as it is well known that the arithmetic mean of m i.i.d. random variables with expectation μ and standard deviation σ has same expectation μ but a standard deviation scaled down by $1/\sqrt{m}$. Doing m experiments involves using m different hashing functions. But hashing all the elements m times is particularly time consuming and building m independent hashing functions is not an easy task. To avoid these difficulties we use a *stochastic averaging process*, as introduced by Flajolet and Martin in [10]. Stochastic averaging consists in simulating the effect of m experiments on the multiset while using a single hash function and then averaging an observable over the m experiments. The principle is to distribute the hashed values among m different *buckets*. That is done by dividing $[0,1]$ into m intervals of size $1/m$. A hashed value x falls in the i-th bucket if $\frac{i-1}{m} \leq x < \frac{i}{m}$. Our algorithms keep the k-th minimum of the i-th bucket, denoted by $M_i^{(k)}$ in the analysis, for i from 1 to m (note that the $M_i^{(k)}$ have to be rescaled by $M_i^{(k)} \leftarrow m \cdot (M_i^{(k)} - \frac{i-1}{m})$ to produce elements in the unit interval). A precise estimate is then built by averaging estimates built from the minimum of each bucket, as seen in Section 3.

Inverse Family Algorithm (F: multiset of hashed values; m)

for $x \in F$ do

if $\frac{i-1}{m} \leq x \leq \frac{i}{m}$ do

actualize the k minima of the bucket i with x

return $\xi_1 := (k-1) \sum_{i=1}^{m} \frac{1}{M_i^{(k)}}$ as cardinality estimate.

Square Root Family Algorithm (F: multiset of hashed values; m)

for $x \in F$ do

if $\frac{i-1}{m} \leq x \leq \frac{i}{m}$ do

actualize the k minima of the bucket i with x

return $\xi_2 := \frac{1}{\left(\frac{1}{(k-1)} + \frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2\right)} \left(\sum_{i=1}^{m} \frac{1}{\sqrt{M_i^{(k)}}}\right)^2$ as cardinality estimate.

Logarithm Family Algorithm (F: multiset of hashed values; m)

for $x \in F$ do

if $\frac{i-1}{m} \leq x \leq \frac{i}{m}$ do

actualize the k minima of the bucket i with x

return $\xi_3 := m \cdot \left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-m} \cdot e^{-\frac{1}{m}\sum_{i=1}^{m} \ln M_i^{(k)}}$ as cardinality estimate.

Fig. 1. Pseudo-code of the three families of estimates.

3 Results of the analysis of the three families of estimates.

The main results of the analysis are presented here. The analysis itself and all the proofs can be found in Section 4. We found *asymptotically unbiased* estimates of n for the three families and give their *standard error* in Theorem 1. Note that all results for the Inverse and the Square Root family are given for $k \geq 3$.

Theorem 1 *Consider the algorithms of the three families built on the k-th minimum ($k \geq 3$) using a stochastic averaging process simulating m experiments and applied to an ideal multiset of unknown cardinality n.*

(1) The estimates returned by the Inverse Family, ξ_1, the Square Root Family, ξ_2, and the Logarithm Family, ξ_3, defined respectively as

$$\xi_1 := (k-1)\sum_{i=1}^{m}\frac{1}{M_i^{(k)}},$$
$$\xi_2 := \frac{1}{\left(\frac{1}{k-1}+\frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2\right)}\left(\sum_{i=1}^{m}\frac{1}{\sqrt{M_i^{(k)}}}\right)^2 \quad \textit{and}$$
$$\xi_3 := m\cdot\left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-m}\cdot e^{-\frac{1}{m}\sum_{i=1}^{m}\ln M_i^{(k)}},$$

are asymptotically unbiased in the sense that, for $i = 1, 2, 3$

$$\mathbb{E}[\xi_i] \underset{n\to\infty}{\sim} n.$$

(2) Their standard error, defined as $\frac{1}{n}\sqrt{\mathbb{V}(\xi_i)}$*, satisfies*

$$\mathbb{SE}[\xi_1] \underset{n\to\infty}{\sim} C_1(m,k) := \frac{1}{\sqrt{k-2}}\cdot\frac{1}{\sqrt{m}},$$
$$\mathbb{SE}[\xi_2] \underset{n\to\infty}{\sim} C_2(m,k) := \left[\frac{1}{m^2}\left(\frac{1}{k-1}+\frac{(m-1)\Gamma(k-\frac{1}{2})^2}{(k-1)!^2}\right)^{-2}\right.$$
$$\cdot\left(\frac{m}{(k-1)(k-2)}+\frac{8\binom{m}{2}\Gamma(k-\frac{3}{2})\Gamma(k-\frac{1}{2})}{(k-1)!^2}\right.$$
$$\left.\left.+\frac{6\binom{m}{2}}{(k-1)^2}+\frac{36\binom{m}{3}\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2}+\frac{24\binom{m}{4}\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right)-1\right]^{1/2},$$
$$\mathbb{SE}[\xi_3] \underset{n\to\infty}{\sim} C_3(m,k) := \sqrt{\left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-2m}\cdot\left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^{m}-1},$$

where Γ *is the Euler Gamma function defined in Section 4.1.*

Note that the equivalents of the standard errors —$C_1(m,k)$, $C_2(m,k)$ and $C_3(m,k)$— do not depend on n. They are studied for m large in Lemma 1 —the proof is given in Section 4.6.

Lemma 1 *When* m *large, the equivalents of the standard errors of the estimates of the three families,* $C_1(m,k)$*,* $C_2(m,k)$ *and* $C_3(m,k)$*, defined in Theorem 1, are equivalent to*

$$C_1(m,k) \underset{m\to\infty}{\sim} \frac{1}{\sqrt{k-2}}\cdot\frac{1}{\sqrt{m}},$$
$$C_2(m,k) \underset{m\to\infty}{\sim} 2\cdot\sqrt{\frac{1}{k-1}\left(\frac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2-1}\cdot\frac{1}{\sqrt{m}},$$
$$C_3(m,k) \underset{m\to\infty}{\sim} \sqrt{\psi'(k)}\cdot\frac{1}{\sqrt{m}},$$

where Γ *and* ψ' *are the Euler Gamma and Trigamma functions defined in Section 4.1.*

We now want to compare the algorithms when m is large (we want precise estimates) with respect to the trade-off between precision and memory. The memory used by the algorithms is that required to store the minimums, i.e., km floating numbers for an estimate built with the k-th minimum. The metric here is the *precision* defined as the relative error of the estimates expressed as a function of the memory, noted M $(= km)$. We have

Theorem 2 (Precision of the algorithms) *The precision of the three families of estimates,* $\mathcal{P}_1(M,k)$*,* $\mathcal{P}_2(M,k)$ *and* $\mathcal{P}_3(M,k)$*, given by*

$$\mathcal{P}_1(M,k) := \sqrt{\frac{k}{k-2}} \cdot \frac{1}{\sqrt{M}},$$
$$\mathcal{P}_2(M,k) := 2 \cdot \sqrt{k\left(\frac{1}{k-1}\left(\frac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2 - 1\right)} \cdot \frac{1}{\sqrt{M}},$$
$$\mathcal{P}_3(M,k) := \sqrt{k \cdot \psi'(k)} \cdot \frac{1}{\sqrt{M}},$$

satisfy, ***for a fixed*** M*,*

(1) $\mathcal{P}_1(M,k)$*,* $\mathcal{P}_2(M,k)$ *and* $\mathcal{P}_3(M,k)$ *are decreasing functions of* k*;*
(2) when k *is large, for* $i = 1,2,3$*,*

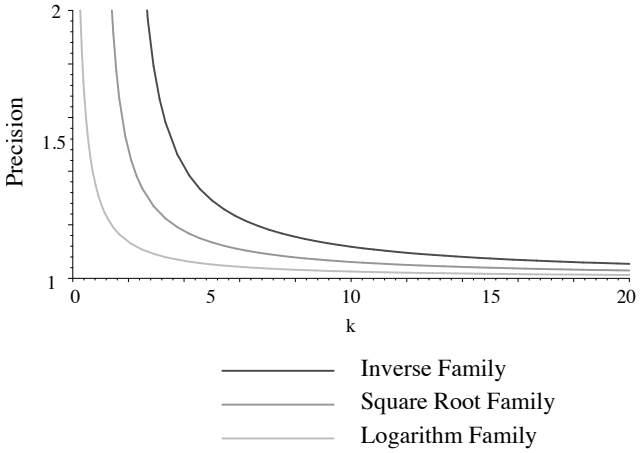
$$\mathcal{P}_i(M,k) \underset{k\to\infty}{\to} \frac{1}{\sqrt{M}}.$$

The proof is given in Section 4.7. The precisions of the three families can be written $\mathcal{P}_i(M,k) = C_i(k) \cdot \frac{1}{\sqrt{M}}$, for $i = 1,2,3$. Note that $C_i(k)$ does *not* depend on M. Figure 2 gives these constants for different values of k. Four main results can be extracted from the theorem.

First result. As expressed in (1), for the three families, the precision improves as k increases. For example, for the logarithm family, we have $\mathcal{P}_3(M,1) = 1.28/\sqrt{M}$, $\mathcal{P}_3(M,2) = 1.14/\sqrt{M}$ and $\mathcal{P}_3(M,3) = 1.09/\sqrt{M}$.

Second result. As expressed in (2), there exists an optimal trade-off between precision and memory for the three families: a precision of $1/\sqrt{M}$ using a memory of M floating numbers.

For practical purposes, we compare the constants of the precision of the three families of estimate and choose a *best practical estimate.*



k	$C_1(k)$	$C_2(k)$	$C_3(k)$
1	–	–	1.28
2	–	–	1.14
3	1.73	1.26	1.09
5	1.29	1.13	1.05
10	1.12	1.06	1.03
100	1.01	1.006	1.003

Fig. 2. Constants of the precision of the three families of estimates for different values of k.

Third result. Though the three families have the same asymptotical precision, we observe that, for all reasonable practical values of k, the precision of the estimate of the Logarithm Family is better than the precision of the estimate of the Square Root Family which is better than the one of the Inverse Family (see Figure 2), that is, for all $3 \leq k \leq 1000$,

$$\mathcal{P}_1(M,k) \geq \mathcal{P}_2(M,k) \geq \mathcal{P}_3(M,k).$$

For example, in the case of the third minimum, we have $\mathcal{P}_1(M,3) = 1.73/\sqrt{M}$, $\mathcal{P}_2(M,3) = 1.26/\sqrt{M}$ and $\mathcal{P}_3(M,3) = 1.09/\sqrt{M}$. More values can be found in Table 1. It means that better estimates are obtained when we apply sublinear functions, such as square root or logarithm.

Fourth result: Best practical estimate. In practice, the optimal efficiency is reached quickly. As a matter of fact, the constant for the estimate of the Logarithm Family using only the third minimum is 1.09. We consider this estimate as the *best practical estimate*. An optimized implementation of an algorithm using this estimate, MINCOUNT, is used in Section 5.3 to study the execution times and in Section 5.4 to analyze Internet traffic traces. Note that, as the best efficiency is attained quickly, the introduction of other families of estimates built with "more" sublinear functions, e.g. $\ln^2$, would not provide enough practical gains in terms of precision.

4 Analysis of the three families of estimates.

The proofs of the results enounced in Section 3 are given here. The main part of this section is the proof of Theorem 1 which shows that the estimates are asymptotically unbiased and gives their standard error. The steps of the analysis are presented in Section 4.2. The proof is then given in Section 4.3 for

the Inverse Family, in Section 4.4 for the Logarithm Family and in Section 4.5 the Square Root Family. The computations strongly rely on special functions; classical definitions and results are first recalled in Section 4.1. The section ends with the proofs of Lemma 1 (analysis of the standard error for large m) and Theorem 2 (comparison of the precision of the estimates) in Sections 4.6 and 4.7.

4.1 Preliminaries: special functions

The computations of the expectations and the standard errors of the estimates intensively use some special functions, in particular the *Euler Gamma function*, defined as

$$\Gamma(z) := \int_0^\infty t^{z-1} e^{-t} dt,$$

and the *Euler Beta function*, defined as

$$\mathcal{B}(\alpha, \beta) := \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt.$$

These two functions and their derivatives are intimately related as

$$\mathcal{B}(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$$

and

$$\frac{d}{d\alpha}\mathcal{B}(\alpha, \beta) = \mathcal{B}(\alpha, \beta)(\psi(\alpha) - \psi(\alpha+\beta)),$$

where ψ is the logarithmic derivative of the Gamma function, also called *Digamma function*. This last function is defined as

$$\psi(z) = \frac{d}{dz} \ln \Gamma(z) = \frac{\Gamma'(z)}{\Gamma(z)}.$$

The Digamma function ψ is related to the *Harmonic Sum* $H_n := \sum_{k=1}^n \frac{1}{k}$ by

$$\psi(n) = H_{n-1} - \gamma,$$

where γ is the *Euler constant* defined as $\lim_{n\to\infty}(H_n - \ln n)$.

The derivative of the function ψ, denoted by ψ', is called *Trigamma function* and is also used in this paper. One of its interesting property is that

$$\psi'(z) = \sum_{k=0}^{\infty} \frac{1}{(z+k)^2}.$$

In particular one may compute $\psi'(1) = \pi^2/6$.

4.2 Method of analysis

Recall that the stochastic averaging process presented in Section 2 distributes the hashed values into m buckets. In the following, N_i denotes the random variables giving the number of values falling into bucket i, for $i = 1, \cdots, m$. A *distribution* of the values is given by $(N_1, \cdots, N_m) = (n_1, \cdots, n_m)$ with $\sum_{i=1}^{m} n_i = m$. We use the notations $\bar{N}$ and $\bar{n}$ for the tuples $(N_1, \cdots, N_m)$ and $(n_1, \cdots, n_m)$.

The analysis of the estimates (consisting of the proof of the main result, Theorem 1) is done in two steps. In the first step, we consider a simplification of the problem and the computations are done under the following hypothesis:

Hypothesis 1 *[Equidistribution hypothesis] Under the* equidistribution hypothesis*, the same number n/m of hashed values falls into each of the m buckets, that is $N_i = n/m$, for $i = 1, \cdots, m$.*

In the second step of the analysis, we prove that the same asymptotic results hold without this hypothesis, using the following *determinization lemma*. Note that, in this case, the hashed values are distributed into the buckets according to a multinomial law, i.e.

$$\mathbb{P}[\bar{N} = \bar{n}] = \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m}.$$

In the following, we note $\binom{n}{\bar{n}}$ for the multinomial $\binom{n}{n_1, \cdots, n_m}$.

Lemma 2 (Determinization of random allocations) *Let f be a function of n and let S_n be defined by*

$$S_n := \sum_{n_1 + \cdots + n_m = n} \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m} f_{n_1} \cdots f_{n_m},$$

with the notation $f_n := f(n)$.
If the function f satisfies

- f is of growth at most polynomial;
- f is of slow variation (in the sense of following Definition 1),

then, for fixed m, there exists a function ϵ such that $\epsilon(n) \underset{n \to \infty}{\to} 0$ and

$$S_n = (f_{n/m})^m (1 + O(\epsilon(n))).$$

Definition 1 (Function of Slow Variation) *A function f of n is* of slow

variation *if there is a function* ϵ *such that* $\epsilon(n) \to 0$ *as* $n \to \infty$ *and if, for all* $j \leq \sqrt{n} \ln n$,

$$f_{n+j} = f_n(1 + O(\epsilon(n))).$$

Proof [Determinization lemma] The idea of the proof is to show that the terms that contribute to the sum S_n are in a central domain defined below.

Definition 2 (Central Domain and Periphery) *A tuple* $\bar{n}$ *is said inside the* Central Domain CD *if, for any* i,

$$\frac{n}{m} - \sqrt{\frac{n}{m}} \ln \frac{n}{m} \leq n_i \leq \frac{n}{m} + \sqrt{\frac{n}{m}} \ln \frac{n}{m}.$$

A tuple outside of CD is said to be in the Periphery P.

We have the following result.

Proposition 2 *When* n *goes to infinity, we have, for a fixed* m,

$$\begin{aligned} \mathbb{P}[\bar{n} \in CD] &= \sum_{\bar{n} \in CD} \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m} \underset{n \to \infty}{\to} 1 \\ \mathbb{P}[\bar{n} \in P] &= \sum_{\bar{n} \in P} \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m} = O(\exp(-C \cdot (\log n)^2)) \underset{n \to \infty}{\to} 0, \end{aligned}$$

where C *is a positive constant.*

Proof It is sufficient to prove the second inequality as an allocation $\bar{n}$ is either in P or in CD. $\bar{n}$ is in the periphery if there is an $i \in \mathbb{N}, 1 \leq i \leq m$, such that

$$n_i \notin \left[\frac{n}{m} - \sqrt{\frac{n}{m}} \ln \frac{n}{m}, \frac{n}{m} + \sqrt{\frac{n}{m}} \ln \frac{n}{m} \right].$$

So,

$$\mathbb{P}[\bar{n} \in P] \leq m \mathbb{P} \left[n_1 \notin \left[\frac{n}{m} - \sqrt{\frac{n}{m}} \ln \frac{n}{m}, \frac{n}{m} + \sqrt{\frac{n}{m}} \ln \frac{n}{m} \right] \right].$$

n_1 is the sum of n random variables following a Bernoulli distribution of parameter $1/m$. The Hoeffding's inequality gives, $\forall t > 0$,

$$\mathbb{P}(|n_1 - \mathbb{E}[n_1]| \geq t) \leq 2 \exp\left(-\frac{2t^2}{n} \right).$$

For $t = \sqrt{\frac{n}{m}} \ln \frac{n}{m}$, we have

$$\begin{aligned} \mathbb{P}\left(|n_1 - \tfrac{n}{m}| \geq \sqrt{\tfrac{n}{m}} \ln \tfrac{n}{m} \right) &\leq 2 \exp\left(-\tfrac{2}{n} \cdot \left(\sqrt{\tfrac{n}{m}} \log\left(\tfrac{n}{m} \right) \right)^2 \right) \\ &= 2 \exp\left(-\tfrac{2}{m} \cdot \left(\log\left(\tfrac{n}{m} \right) \right)^2 \right). \end{aligned}$$

Hence

$$\mathbb{P}[\bar{n} \in P] = O(\exp(-C \cdot (\log n)^2)) \underset{n\to\infty}{\to} 0,$$

with C a positive constant. This ends the proof of Proposition 2. □

We now have to find an equivalent of S_n. Let s_n denote $s_n(\bar{n}) := \frac{1}{m^n}\binom{n}{\bar{n}} f_{n_1} \cdots f_{n_m}$ so that we may write $S_n = \sum_{n_1+\cdots+n_m=n} s_n(\bar{n})$.

Study of S_n in the Central Domain As f has slow variation, there exist $\epsilon(n/m)$ such that $\epsilon(n/m) \underset{n\to\infty}{\to} 0$ and

$$\forall j \leq \sqrt{\frac{n}{m}} \ln \frac{n}{m}, f_{n/m+j} = f_{n/m}(1 + O(\epsilon(n/m))).$$

Hence

$$\begin{aligned}\sum_{\bar{n}\in CD} s_n(\bar{n}) &= \sum_{\sum l_i=0, |l_i|\leq\sqrt{\frac{n}{m}}\ln(\frac{n}{m})} \frac{1}{m^n}\binom{n}{n/m+l_1,\cdots,n/m+l_m}(f_{n/m})^m(1+O(\epsilon(n/m)))^m \\ &= (f_{n/m})^m(1+O(\epsilon(n/m)))^m \sum_{CD} \frac{1}{m^n}\binom{n}{n/m+l_1,\cdots,n/m+l_m}\end{aligned}$$

Proposition 2 gives

$$\sum_{\bar{n}\in CD} s_n(\bar{n}) \underset{n\to\infty}{\sim} (f_{n/m})^m(1+O(\epsilon(n))).$$

Study of S_n in the Periphery. As f has at most a polynomial growth, there exists $a \in \mathbb{N}$ such that $f_n \leq n^a$. Hence

$$\sum_{\bar{n}\in P} s_n(\bar{n}) = \sum_P \frac{1}{m^n}\binom{n}{\bar{n}} f_{n_1} \cdots f_{n_m} \leq (n^a)^m \sum_{\bar{n}\in P} \frac{1}{m^n}\binom{n}{\bar{n}}.$$

Proposition 2 gives

$$\sum_{\bar{n}\in P} s_n(\bar{n}) = n^{am} \cdot O(\exp(-C(\ln n)^2)) \underset{n\to\infty}{\to} 0$$

Hence we have

$$S_n = (f_{n/m})^m(1+O(\epsilon(n))),$$

with $\epsilon(n) \to 0$ as $n \to \infty$. This ends the proof of Lemma 2. □

This lemma is easily extended to the cases where one has a polynomial expression (with polynomial g) of $t \leq m$ different functions $(f^{(1)}_{n_1} \cdots f^{(t)}_{n_t})$ —instead of m identical functions. The most general variant is given in the following lemma —which is used in the analysis of the Square Root Family in Section 4.5.

Lemma 3 (Variant of the determinization lemma) *Let $f^{(1)}, \cdots, f^{(t)}$ be t different functions of slow variation and of growth at most polynomial and let g be a polynomial. Let S_n be defined by*

$$S_n := \sum_{n_1+\cdots+n_m=n} \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m} g(f^{(1)}_{n_1}, \cdots, f^{(t)}_{n_t}),$$

Then, for fixed m, there exists a function ϵ such that $\epsilon(n) \underset{n\to\infty}{\to} 0$ and

$$S_n = g(f^{(1)}_{n/m}, \cdots, f^{(t)}_{n/m})(1 + O(\epsilon(n))).$$

4.3 Analysis of the Inverse Family

The proof of Theorem 1 is given here for the Inverse Family. The proof is done in two steps, as explained in Section 4.2. In the first part of the analysis, the result is proved for a simplification of the problem (equidistribution hypothesis, Hypothesis 1). In the second part, using the determinization lemma, Lemma 2, we show that the same asymptotic results hold for the general problem.

4.3.1 First step of the proof.

Unbiased estimate of n. The goal is to show here that ξ_1, defined as $\xi_1 := (k-1)\sum_{i=1}^{m} \frac{1}{M_i^{(k)}}$, for $k \geq 3$, is an asymptotically unbiased estimate of n. We have

$$\mathbb{E}[\xi_1] = (k-1)\sum_{i=1}^{m} \mathbb{E}[\frac{1}{M_i^{(k)}}].$$

The density of the k-th minimum of n random values chosen in $[0,1]$ according to a uniform distribution is

$$\mathbb{P}(M^{(k)} \in [x, x+dx]) = k\binom{n}{k} x^{k-1}(1-x)^{n-k}dx.$$

Hence we have:

$$\begin{aligned}
\mathbb{E}[\tfrac{1}{M^{(k)}}] &= \textstyle\int_0^1 \frac{1}{x} k\binom{n}{k} x^{k-1}(1-x)^{n-k}dx = k\binom{n}{k}\mathcal{B}(k-1, n-k+1) \\
&= k\binom{n}{k}\tfrac{\Gamma(k-1)\Gamma(n-k+1)}{\Gamma(n)} = k\tfrac{n!}{(n-k)!k!}\tfrac{(k-2)!(n-k)!}{(n-1)!} \\
&= \tfrac{n}{k-1}
\end{aligned}$$

Note that the expectation would be infinite for $k = 1$. Under the equidistribution hypothesis, n/m hashed values fall in each bucket. Hence we have

$$\mathbb{E}[\xi_1] = (k-1)\sum_{i=1}^{m} \frac{1}{k-1}\frac{n}{m} = n,$$

that is ξ_1 is an unbiased estimate of n.

Standard error. What is the standard error of this estimate? We note $\mathcal{M}_1 := \sum_{i=1}^{m} \frac{1}{M_i^{(k)}}$. We have $\xi = (k-1)\mathcal{M}_1$.

$$\mathbb{E}[\mathcal{M}_1^2] = \mathbb{E}[(\sum_{i=1}^{m} \frac{1}{M_i^{(k)}})^2] = \mathbb{E}[\sum_{1\le i\le m} (\frac{1}{M_i^{(k)}})^2 + 2\cdot \sum_{1\le i<j\le m} \frac{1}{M_i^{(k)}}\frac{1}{M_j^{(k)}}].$$

By linearity of the expectation, we have

$$\mathbb{E}[\mathcal{M}_1^2] = \sum_{1\le i\le m} \mathbb{E}[(\frac{1}{M_i^{(k)}})^2] + 2\cdot \sum_{1\le i<j\le m} \mathbb{E}[\frac{1}{M_i^{(k)}}\frac{1}{M_j^{(k)}}].$$

As the $M_i^{(k)}$ are i.i.d., we obtain

$$\mathbb{E}[\mathcal{M}_1^2] = m\cdot \mathbb{E}[(\frac{1}{M_1^{(k)}})^2] + 2\binom{m}{2}\cdot \mathbb{E}[\frac{1}{M_1^{(k)}}]^2.$$

$$\begin{aligned}
\mathbb{E}[\tfrac{1}{(M^{(k)})^2}] &= \textstyle\int_0^1 \frac{1}{x^2} k\binom{n}{k} x^{k-1}(1-x)^{n-k}dx = k\binom{n}{k}\int_0^1 x^{(k-2)-1}(1-x)^{(n-k+1)-1}dx \\
&= k\binom{n}{k}\mathcal{B}(k-2, n-k+1) = k\binom{n}{k}\tfrac{\Gamma(k-2)\Gamma(n-k+1)}{\Gamma(n-1)} \\
&= k\tfrac{n!}{(n-k)!k!}\tfrac{(k-3)!(n-k)!}{(n-2)!} = \tfrac{n(n-1)}{(k-1)(k-2)}.
\end{aligned}$$

Hence, as n/m hashed values fall in each bucket, we get

$$\mathbb{E}[\mathcal{M}_1^2] = m\cdot \frac{\frac{n}{m}(\frac{n}{m}-1)}{(k-1)(k-2)} + 2\binom{m}{2}\cdot \frac{1}{(k-1)^2}(\frac{n}{m})^2.$$

Recall that $\xi = (k-1)\mathcal{M}_1$. So

$$\mathbb{E}[\xi_1^2] = \frac{1}{m}\cdot\frac{k-1}{k-2}(n^2 - mn) + \frac{m(m-1)}{m^2}\cdot n^2.$$

As $\mathbb{V}[\xi_1] := \mathbb{E}[\xi_1^2] - \mathbb{E}[\xi_1]^2$, we have

$$\mathbb{V}[\xi_1] = \frac{1}{m}\cdot\frac{k-1}{k-2}(n^2 - mn) + \frac{m-1}{m}\cdot n^2 - n^2$$

For n large,

$$\mathbb{V}[\xi_1] \underset{n\to\infty}{\sim} \frac{(k-1) + (m-1)(k-2) - m(k-2)}{m(k-2)}\cdot n^2 \underset{n\to\infty}{\sim} \frac{1}{m(k-2)}\cdot n^2.$$

Hence

$$\mathbb{SE}[\xi_1] \underset{n\to\infty}{\sim} \frac{1}{\sqrt{k-2}} \cdot \frac{1}{\sqrt{m}}.$$

Note that the standard error would be infinite for $k = 2$.

4.3.2 Second step of the proof.

We remove now the *equidistribution hypothesis*. The numbers N_i, $i \in 1, \cdots, m$, of hashed values falling into the buckets are no longer identical but they are distributed according to a multinomial law. That is $\mathbb{P}_{(N_1,\cdots,N_m)=(n_1,\cdots,n_m)} = \frac{1}{m^n}\binom{n}{n_1,\cdots,n_m}$. Without the equidistribution hypothesis, the expectation $\mathbb{E}[\xi_1]$ is the sum over all possible allocations $(N_1, \cdots, N_m) = (n_1, \cdots, n_m)$ of the hashed values in the buckets.

$$\begin{aligned}\mathbb{E}[\xi_1] &= \sum_{\bar{n}} \mathbb{P}_{\bar{N}=\bar{n}} \cdot \mathop{\mathbb{E}}_{\bar{N}=\bar{n}} [(k-1)(\tfrac{1}{M_1^{(k)}} + \cdots + \tfrac{1}{M_m^{(k)}})] \\ &= (k-1) \sum_{\bar{n}} \mathbb{P}_{\bar{N}=\bar{n}} \cdot \textstyle\sum_{i=1}^{m} \mathop{\mathbb{E}}_{\bar{N}_i=\bar{n}_i} [\tfrac{1}{M_1^{(k)}}].\end{aligned}$$

The number of hashed values falling into the buckets are distributed according to a multinomial law. So

$$\mathbb{E}[\xi_1] = (k-1) \sum_{n_1+\cdots+n_m=n} \frac{1}{m^n} \binom{n}{n_1, \cdots, n_m} \sum_{i=1}^{m} \mathop{\mathbb{E}}_{\bar{N}_i=\bar{n}_i} [\frac{1}{M_1^{(k)}}].$$

We have previously seen that $\mathbb{E}[\frac{1}{M_i^{(k)}}] = \frac{1}{k-1} \cdot n_i$. Its growth is in $O(n_i)$. It has slow variation in the sense of Definition 1. The variant of the determinization lemma, Lemma 3, applies. Hence

$$\mathbb{E}[\xi] = \mathbb{E}[(k-1) \sum_{i=1}^{m} \mathop{\mathbb{E}}_{\bar{N}_i=\bar{n}_i} [\frac{1}{M_1^{(k)}}](1 + O(\epsilon(n))),$$

with $\epsilon(n) \underset{n\to\infty}{\to} 0$. When n goes to infinity, this formula gives the same equivalent as if exactly n/m values were falling in each bucket, that is the *equidistribution hypothesis*. Hence the estimate ξ_3 is asymptotically unbiased in the general case also. The same method applies for the computation of the standard error, finishing the proof of Theorem 1 for the Inverse Family.

4.4 Analysis of the Logarithm Family

The proof of Theorem 1 is given here for the Logarithm Family. It follows the method of analysis presented in Section 4.2. The computations strongly rely on special functions; classical definitions and results are recalled in Section 4.1.

4.4.1 First step of the proof.

Unbiased estimate of n. We want to show here that ξ_3, defined as

$$\xi := m \cdot \left(\frac{\Gamma(k - \frac{1}{m})}{\Gamma(k)} \right)^{-m} \cdot e^{-\frac{1}{m} \sum_{i=1}^{m} \ln M_i^{(k)}},$$

is an asymptotically unbiased estimate of n. We note $\mathcal{M}_3 := \frac{1}{m} \sum_{i=1}^{m} \ln \frac{1}{M_i^{(k)}}$. Under the equidistribution hypothesis, we have

$$\mathbb{E}[e^{\mathcal{M}_3}] = \mathbb{E}[\exp(\frac{1}{m}(\ln \frac{1}{M_1^{(k)}} + \cdots + \ln \frac{1}{M_m^{(k)}}))] = \mathbb{E}[(M^{(k)})^{-\frac{1}{m}}]^m$$

The density of the k-th minimum is

$$\mathbb{P}(M^{(k)} \in [x, x + dx]) = k \binom{n}{k} x^{k-1} (1 - x)^{n-k} dx.$$

Under the equidistribution hypothesis, n/m hashed values fall in each bucket. Hence, we have

$$\mathbb{E}[e^{\mathcal{M}_3}] = [k \binom{n/m}{k} \int_0^1 x^{-\frac{1}{m}} x^{k-1} (1 - x)^{\frac{n}{m} - k} dx]^m$$

Using the summary on special functions of Section 4.1, we obtain

$$\begin{aligned} \mathbb{E}[e^{\mathcal{M}_3}] &= [k \tbinom{n/m}{k} \mathcal{B}(k - \tfrac{1}{m}, \tfrac{n}{m} - k + 1)]^m \\ &= [\tfrac{(\frac{n}{m})!}{(k-1)!(\frac{n}{m}-k)!} \tfrac{\Gamma(k-\frac{1}{m})\Gamma(\frac{n}{m}-k+1)}{\Gamma(\frac{n}{m}+1-\frac{1}{m})}]^m \\ &= [\tfrac{\Gamma(k-\frac{1}{m})}{\Gamma(k)} \cdot \tfrac{\Gamma(\frac{n}{m}+1)}{\Gamma(\frac{n}{m}+1-\frac{1}{m})}]^m \end{aligned}$$

For n large, we have $\frac{\Gamma(\frac{n}{m}+1)}{\Gamma(\frac{n}{m}+1-\frac{1}{m})} \underset{n \to \infty}{\sim} (\frac{n}{m})^{1/m}$. Hence

$$\mathbb{E}[e^{\mathcal{M}}] \underset{n \to \infty}{\sim} \left(\frac{\Gamma(k - \frac{1}{m})}{\Gamma(k)} \right)^m \cdot \frac{n}{m}.$$

The expression $\frac{1}{m} \left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)} \right)^m$ appears as bias. As ξ_3 is defined as,

$$\xi_3 := m \cdot \left(\frac{\Gamma(k - \frac{1}{m})}{\Gamma(k)} \right)^{-m} \cdot e^{\mathcal{M}_3},$$

we have

$$\mathbb{E}[\xi_3] \underset{n \to \infty}{\sim} n,$$

that is ξ_3 is an asymptotically unbiased estimate of n.

Standard error. What is the standard error of this estimate? Similar computations give us

$$\mathbb{E}[e^{2\mathcal{M}_3}] \underset{n\to\infty}{\sim} \left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^m \cdot \left(\frac{n}{m}\right)^2.$$

As the variance of ξ_3 is defined by $\mathbb{V}[\xi_3] := \mathbb{E}[\xi_3^2] - \mathbb{E}[\xi_3]^2$, we have

$$\mathbb{V}[\xi_3] \underset{n\to\infty}{\sim} m^2 \cdot \left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-2m} \cdot \left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^m \cdot \left(\frac{n}{m}\right)^2 - n^2$$

Giving

$$\mathbb{SE}[\xi_3] \underset{n\to\infty}{\sim} \sqrt{\left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-2m} \cdot \left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^m - 1},$$

finishing the first part of the proof.

4.4.2 Second step of the proof.

We remove now the *equidistribution hypothesis*. The numbers N_i ($i \in 1, \cdots, m$) of hashed values falling into the buckets are no longer identical but they are distributed according to a multinomial law. That is $\mathbb{P}_{(N_1,\cdots,N_m)=(n_1,\cdots,n_m)} = \frac{1}{m^n}\binom{n}{n_1,\cdots,n_m}$. The main factor of the expectation of the estimate $\mathbb{E}[\xi_3]$ is $\mathbb{E}[e^{\mathcal{M}_3}]$. Without the equidistribution hypothesis, this expectation is the sum over all possible allocations $(N_1, \cdots, N_m) = (n_1, \cdots, n_m)$ of the hashed values in the buckets.

$$\begin{aligned}\mathbb{E}[e^{\mathcal{M}_3}] &= \mathbb{E}[\exp(\tfrac{1}{m}(\ln \tfrac{1}{M_1^{(k)}} + \cdots + \ln \tfrac{1}{M_m^{(k)}}))] \\ &= \sum_{\bar{n}} \mathbb{P}_{\bar{N}=\bar{n}} \times \mathop{\mathbb{E}}_{\bar{N}=\bar{n}} [(M_1^{(k)})^{-1/m} \cdots (M_m^{(k)})^{-1/m}].\end{aligned}$$

The number of hashed values falling into the buckets are distributed according to a multinomial law. So

$$\mathbb{E}[e^{\mathcal{M}_3}] = \sum_{n_1+\cdots+n_m=n} \frac{1}{m^n}\binom{n}{n_1,\cdots,n_m} \mathop{\mathbb{E}}_{N_1=n_1}[(M_1^{(k)})^{-1/m}] \cdots \mathop{\mathbb{E}}_{N_m=n_m}[(M_m^{(k)})^{-1/m}]$$

We have previously shown that $\mathop{\mathbb{E}}_{N_i=n_i}[(M_i^{(k)})^{-1/m}] \underset{n_i\to\infty}{\sim} \frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)} \cdot (\frac{n_i}{m})^{1/m}$. Its growth is in $O(n_i^{1/m})$. It has slow variation in the sense of Definition 1. The determinization lemma, Lemma 2, applies. Hence

$$\mathbb{E}[e^{\mathcal{M}_3}] = \left(\mathop{\mathbb{E}}_{N_i=n/m}[(M_i^{(k)})^{-1/m}]\right)^m (1 + O(\epsilon(n))),$$

with $\epsilon(n) \underset{n\to\infty}{\to} 0$. When n goes to infinity, this formula gives the same equivalent as if exactly n/m values were falling in each bucket (the *equidistribution*

hypothesis). Hence the estimate ξ_3 is asymptotically unbiased in the general case also. The same method applies for the computation of the standard error, finishing the proof of Theorem 1 for the Logarithm Family.

4.5 Analysis of the Square Root family

The proof of Theorem 1 is given here for the Square Root Family. It follows the method of analysis presented in Section 4.2 and it can be extended to any integer power.

4.5.1 Preliminary

First, as a small preliminary, we compute $\mathbb{E}[\frac{1}{(M^{(k)})^\alpha}]$. The density of the k-th minimum of n random values chosen in $[0,1]$ according to a uniform distribution is

$$\mathbb{P}(M^{(k)} \in [x, x+dx]) = k\binom{n}{k} x^{k-1}(1-x)^{n-k}dx.$$

Hence we have:

$$\begin{aligned}
\mathbb{E}[\tfrac{1}{(M^{(k)})^\alpha}] = \quad & k\tbinom{n}{k}\textstyle\int_0^1 \frac{1}{x^\alpha}x^{k-1}(1-x)^{n-k}dx = k\tbinom{n}{k}\mathcal{B}(k-\alpha, n-k+1) \\
= \quad & k\tbinom{n}{k}\tfrac{\Gamma(k-\alpha)\Gamma(n-k+1)}{\Gamma(n+1-\alpha)} = \tfrac{\Gamma(k-\alpha)}{(k-1)!}\tfrac{\Gamma(n+1)}{\Gamma(n+1-\alpha)} \\
\underset{n\to\infty}{\sim} \quad & \tfrac{\Gamma(k-\alpha)}{(k-1)!}n^\alpha.
\end{aligned}$$

Note that it gives

$$\mathbb{E}[\tfrac{1}{\sqrt{M^{(k)}}}] \underset{n\to\infty}{\sim} \tfrac{\Gamma(k-\frac{1}{2})}{(k-1)!}\sqrt{n}.$$

4.5.2 First step of the proof (equidistribution hypothesis).

Unbiased estimate of n. The goal is to show here that the estimate of the Square Root Family, ξ_2, defined as

$$\xi_2 := \frac{1}{\left(\frac{1}{k-1} + \frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2\right)}\left(\sum_{i=1}^{m}\frac{1}{\sqrt{M_i^{(k)}}}\right)^2,$$

is an asymptotically unbiased estimate of n. We note $\mathcal{M}_2 := \sum_{i=1}^m \frac{1}{\sqrt{M_i^{(k)}}}$.

$$\begin{aligned}
\mathbb{E}[\mathcal{M}_2^2] &= \mathbb{E}[(\tfrac{1}{\sqrt{M_1^{(k)}}} + \ldots + \tfrac{1}{\sqrt{M_m^{(k)}}})^2] \\
&= \mathbb{E}[\textstyle\sum_{i=1}^m \frac{1}{M_i^{(k)}} + \sum_{1\le i<j\le m}\frac{2}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}]
\end{aligned}$$

By linearity of the expectation and as the $M_i^{(k)}$ are i.i.d, we have

$$\mathbb{E}[\mathcal{M}_2^2] = m \cdot \mathbb{E}\left[\frac{1}{M^{(k)}}\right] + 2\binom{m}{2}\mathbb{E}\left[\frac{1}{\sqrt{M^{(k)}}}\right]^2.$$

Under the equidistribution hypothesis, n/m hashed values fall in each bucket. The preliminary gives us

$$\begin{aligned}\mathbb{E}[\mathcal{M}_2^2] &\underset{n\to\infty}{\sim} m \cdot \tfrac{\Gamma(k-1)}{(k-1)!} \cdot \tfrac{n}{m} + m(m-1)\left(\tfrac{\Gamma(k-\frac{1}{2})}{(k-1)!}\sqrt{\tfrac{n}{m}}\right)^2 \\ &\underset{n\to\infty}{\sim} (\tfrac{1}{k-1} + \tfrac{m-1}{(k-1)!^2}\Gamma(k-\tfrac{1}{2})^2) \cdot n.\end{aligned}$$

The expression $(\frac{1}{k-1} + \frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2)$ appears as bias. By definition of ξ_2, we have

$$\mathbb{E}[\xi_2] \underset{n\to\infty}{\sim} n,$$

that is ξ_2 is an unbiased estimate of n.

Standard error. We now want the standard error of ξ. We first have to compute $\mathbb{E}[\mathcal{M}_2^4]$.

$$\begin{aligned}\mathcal{M}_2^4 = (\tfrac{1}{\sqrt{M_1^{(k)}}} + \ldots + \tfrac{1}{\sqrt{M_m^{(k)}}})^4 &= \textstyle\sum_{i=1}^m \frac{1}{(M_i^{(k)})^2} + \sum_{1\le i<j\le m} \frac{4}{\sqrt{(M_i^{(k)})^3}\sqrt{(M_j^{(k)})}} \\ &+ \textstyle\sum_{1\le i<j\le m} \frac{4}{\sqrt{M_i^{(k)}}\sqrt{(M_j^{(k)})^3}} + \sum_{1\le i<j\le m} \frac{6}{M_i^{(k)}M_j^{(k)}} \\ &+ \textstyle\sum_{1\le i<j<l\le m} \frac{12}{M_i^{(k)}\sqrt{M_j^{(k)}}\sqrt{M_l^{(k)}}} + \sum_{1\le i<j<l\le m} \frac{12}{\sqrt{M_i^{(k)}}M_j^{(k)}\sqrt{M_l^{(k)}}} \\ &+ \textstyle\sum_{1\le i<j<l\le m} \frac{12}{\sqrt{M_i^{(k)}}\sqrt{M_j^{(k)}}M_l^{(k)}} + \sum_{1\le i<j<l<o\le m} \frac{24}{\sqrt{M_i^{(k)}}\sqrt{M_j^{(k)}}\sqrt{M_l^{(k)}}\sqrt{M_o^{(k)}}}\end{aligned}$$

As the $M_i^{(k)}$ are i.i.d., we have by linearity of the expectation

$$\begin{aligned}\mathbb{E}[\mathcal{M}^4] = m\mathbb{E}\left[\tfrac{1}{(M^{(k)})^2}\right] + 8\binom{m}{2}\mathbb{E}\left[\tfrac{1}{\sqrt{(M^{(k)})^3}}\right]\mathbb{E}\left[\tfrac{1}{\sqrt{(M^{(k)})}}\right] + 6\binom{m}{2}\mathbb{E}\left[\tfrac{1}{M^{(k)}}\right]^2 \\ +36\binom{m}{3}\mathbb{E}\left[\tfrac{1}{M^{(k)}}\right]\mathbb{E}\left[\tfrac{1}{\sqrt{(M^{(k)})}}\right]^2 + 24\binom{m}{4}\mathbb{E}\left[\tfrac{1}{\sqrt{(M^{(k)})}}\right]^4\end{aligned}$$

We compute in the preliminary that $\mathbb{E}[\frac{1}{(M^{(k)})^\alpha}] \underset{n\to\infty}{\sim} \frac{\Gamma(k-\alpha)}{(k-1)!}n^\alpha$. As n/m hashed values fall in each bucket, we get

$$\begin{aligned}\mathbb{E}[\mathcal{M}_2^4] \underset{n\to\infty}{\sim} \left(\tfrac{n}{m}\right)^2 \Bigg(m\tfrac{\Gamma(k-2)}{(k-1)!} + 8\binom{m}{2}\tfrac{\Gamma(k-\frac{3}{2})\Gamma(k-\frac{1}{2})}{(k-1)!^2} + 6\binom{m}{2}\tfrac{\Gamma(k-1)^2}{(k-1)!^2} \\ +36\binom{m}{3}\tfrac{\Gamma(k-1)\Gamma(k-\frac{1}{2})^2}{(k-1)!^3} + 24\binom{m}{4}\tfrac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\Bigg).\end{aligned}$$

The standard error of $\xi_2$is defined by $\mathbb{SE}[\xi_2] = \frac{1}{n}\sqrt{\mathbb{E}[\xi_2^2] - \mathbb{E}[\xi_2]^2}$. As $\xi_2^2 := (\frac{1}{k-1} + \frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2)^{-2}\mathcal{M}_2^4$, we obtain

$$\mathbb{SE}[\xi_2] \underset{n\to\infty}{\sim} \left[\frac{1}{m^2}\left(\frac{1}{k-1} + \frac{(m-1)\Gamma(k-\frac{1}{2})^2}{(k-1)!^2}\right)^{-2}\left(\frac{m}{(k-1)(k-2)} + \frac{8\binom{m}{2}\Gamma(k-\frac{3}{2})\Gamma(k-\frac{1}{2})}{(k-1)!^2}\right.\right.$$
$$\left.\left. + \frac{6\binom{m}{2}}{(k-1)^2} + \frac{36\binom{m}{3}\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2} + \frac{24\binom{m}{4}\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right) - 1\right]^{1/2}.$$

4.5.3 Second step of the proof (without the equidistribution hypothesis).

As we have seen, the main factor of the estimate ξ is $\mathbb{E}[\mathcal{M}^2]$. Without the equidistribution hypothesis this expectation is the sum over all possible allocations $N_1, \cdots, N_m = (n_1, \cdots, n_m)$ of the hashed values in the buckets.

$$\begin{aligned}\mathbb{E}[\mathcal{M}^2] &= \mathbb{E}[(\frac{1}{\sqrt{M_1^{(k)}}} + \ldots + \frac{1}{\sqrt{M_m^{(k)}}})^2] \\ &= \textstyle\sum_{i=1}^m \mathbb{E}[\frac{1}{M_i^{(k)}}] + 2\sum_{1\le i<j\le m}\mathbb{E}[\frac{1}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}]\end{aligned}$$

with

$$\mathbb{E}[\frac{1}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}] = \sum_{\bar{n}}\mathbb{P}_{\bar{N}=\bar{n}} \times \mathop{\mathbb{E}}_{\bar{N}=\bar{n}}[\frac{1}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}].$$

For a given allocation, the $M_i^{(k)}$ are independent and

$$\mathbb{E}[\frac{1}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}] = \sum_{n_1+\cdots+n_m=n}\frac{1}{m^n}\binom{n}{n_1,\cdots,n_m}\mathop{\mathbb{E}}_{N_i=n_i}[\frac{1}{\sqrt{(M_i^{(k)})}}]\mathop{\mathbb{E}}_{N_j=n_j}[\frac{1}{\sqrt{(M_j^{(k)})}}].$$

We have seen above that $\mathop{\mathbb{E}}_{N_i=n_i}[\frac{1}{\sqrt{M_i^{(k)}}}] \underset{n_i\to\infty}{\sim} \frac{\Gamma(k-\frac{1}{2})}{(k-1)!}\sqrt{n}_i$. Its growth is in $O(\sqrt{n})$. It has slow variation in the sense of Definition 1. The variant of the determinization lemma (Lemma 3) applies. Hence we have:

$$\begin{aligned}\mathbb{E}[\frac{1}{\sqrt{(M_i^{(k)})}\sqrt{(M_j^{(k)})}}] &= \tfrac{1}{(k-1)!}^2\Gamma(k-\tfrac{1}{2})^2(1+O(\epsilon(n))) \\ \mathbb{E}[\mathcal{M}^2] &= n\left(m\tfrac{1}{(k-1)!}\Gamma(k-1) + 2\binom{m}{2}\tfrac{1}{(k-1)!}^2\Gamma(k-\tfrac{1}{2})^2\right)(1+O(\epsilon(n))).\end{aligned}$$

This formula gives the same asymptotic equivalent as the one with the equidistribution hypothesis. The same method applies for the computation of the standard error. Other variants of the determinization lemma are used, typi-

cally to obtain equivalent of expressions as

$$\begin{aligned}\mathbb{E}[\frac{1}{M_i^{(k)}\sqrt{(M_j^{(k)})}\sqrt{(M_l^{(k)})}}] &= \sum_{n_1+\cdots+n_m=n}\left(\frac{1}{m^n}\binom{n}{n_1,\cdots,n_m}\mathop{\mathbb{E}}_{N_i=n_i}[\frac{1}{(M_i^{(k)})}]\right.\\ &\quad\left.\times\mathop{\mathbb{E}}_{N_j=n_j}[\frac{1}{\sqrt{(M_j^{(k)})}}]\mathop{\mathbb{E}}_{N_l=n_l}[\frac{1}{\sqrt{(M_l^{(k)})}}]\right)\\ &= \mathbb{E}[\frac{1}{(M_i^{(k)})}]\mathbb{E}[\frac{1}{\sqrt{(M_j^{(k)})}}]^2(1+O(\epsilon(n))).\end{aligned}$$

Hence, in the general case, the same estimate may be used and it has asymptotically the same standard error. This ends the proof of Theorem 1. □

4.6 Study of the standard error for large m

This section gives the proof of Lemma 1. It is a study for large m of the equivalents of the standard errors of the three families given in Theorem 1.

Inverse Family. We directly have

$$C_1(m,k) \underset{m\to\infty}{\sim} \frac{1}{\sqrt{k-2}}\cdot\frac{1}{\sqrt{m}},$$

from the definition of $C_1(m,k)$.

Square Root Family. We want to show here that the equivalent of the estimate of the Square Root Family, $C_2(m,k)$, defined as

$$\begin{aligned}C_2(m,k) := \left[\frac{1}{m^2}\left(\frac{1}{k-1}+\frac{(m-1)\Gamma(k-\frac{1}{2})^2}{(k-1)!^2}\right)^{-2}\left(\frac{m}{(k-1)(k-2)}+\frac{8\binom{m}{2}\Gamma(k-\frac{3}{2})\Gamma(k-\frac{1}{2})}{(k-1)!^2}\right.\right.\\ \left.\left.+\frac{6\binom{m}{2}}{(k-1)^2}+\frac{36\binom{m}{3}\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2}+\frac{24\binom{m}{4}\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right)-1\right]^{1/2},\end{aligned}$$

is such that

$$C_2(m,k) \underset{m\to\infty}{\sim} 2\cdot\sqrt{\frac{1}{k-1}\left(\frac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2-1}\cdot\frac{1}{\sqrt{m}},$$

where Γ is the Euler Gamma function defined in Section 4.1. Let us note

$$B_1 := \left(\frac{1}{k-1}+\frac{m-1}{(k-1)!^2}\Gamma(k-\frac{1}{2})^2\right)^2$$

and

$$B_2 := \frac{1}{m^2}\left(\frac{m}{(k-1)(k-2)} + \frac{8\binom{m}{2}\Gamma(k-\frac{3}{2})\Gamma(k-\frac{1}{2})}{(k-1)!^2} + \frac{6\binom{m}{2}}{(k-1)^2} + \frac{36\binom{m}{3}\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2} + \frac{24\binom{m}{4}\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right)$$

in order to have $C_2(m,k) = \sqrt{B_1^{-1}\cdot B_2 - 1}$. When m is large, B_1 can be expressed in the following way

$$\begin{aligned} B_1 &= (m-1)^2\frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4} + 2\frac{m-1}{k-1}\left(\frac{\Gamma(k-\frac{1}{2})}{(k-1)!}\right)^2 + o(m) \\ &= \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\cdot m^2 + 2\left(\frac{\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2} - \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right)\cdot m + o(m) \\ &= \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\cdot m^2\left[1 + 2\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right] \end{aligned}$$

When m is large, B_2 can be expressed in the following way

$$\begin{aligned} B_2 &= \frac{1}{m^2}\Big(m(m-1)(m-2)(m-3)\frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4} \\ &\quad + 6m(m-1)(m-2)\frac{\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2} + o(m^3)\Big) \\ &= \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\cdot m^2 + 6\left(\frac{\Gamma(k-\frac{1}{2})^2}{(k-1)(k-1)!^2} - \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}\right)\cdot m + o(m) \\ &= \frac{\Gamma(k-\frac{1}{2})^4}{(k-1)!^4}m^2\left[1 + 6\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right] \end{aligned}$$

It gives

$$\begin{aligned} B_1^{-1}\cdot B_2 &= \left[1 + 2\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right]^{-1}\cdot\left[1 + 6\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right] \\ &= \left[1 - 2\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right]\cdot\left[1 + 6\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m})\right] \\ &= 1 + 4\left(\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1\right)\cdot\frac{1}{m} + o(\tfrac{1}{m}) \end{aligned}$$

As $C_2(m,k) = \sqrt{B_1^{-1}\cdot B_2 - 1}$, we obtain

$$C_2(m,k) \underset{m\to\infty}{\sim} 2\cdot\sqrt{\frac{(k-1)!^2}{(k-1)\Gamma(k-\frac{1}{2})^2} - 1}\cdot\frac{1}{\sqrt{m}}.$$

Logarithm Family. We want to show here that the equivalent of the estimate of the Logarithm Family, $C_3(m,k)$, defined as

$$C_3(m,k) := \sqrt{\left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-2m}\cdot\left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^{m} - 1},$$

is such that

$$C_3(m,k) \underset{m\to\infty}{\sim} \sqrt{\psi'(k)}\cdot\frac{1}{\sqrt{m}},$$

where Γ and ψ' are the Euler Gamma and Trigamma functions defined in Section 4.1. When m is large, $\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}$ can be expressed as

$$\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)} = 1 - \frac{2}{m}\frac{\Gamma'(k)}{\Gamma(k)} + \frac{2}{m^2}\frac{\Gamma''(k)}{\Gamma(k)} + o(\frac{1}{m^2}).$$

As $\frac{\Gamma'(k)}{\Gamma(k)} = \psi(k)$ and $\frac{\Gamma''(k)}{\Gamma(k)} = \psi(k)^2 + \psi'(k)$, with ψ and ψ' respectively the Euler Digamma and Trigamma functions (see Section 4.1 on special functions), we have

$$\begin{aligned}\ln\left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right) &= -\tfrac{2}{m}\psi(k) + \tfrac{2}{m^2}(\psi'(k)+\psi(k)^2) - \tfrac{1}{2}\tfrac{4}{m^2}\psi(k)^2 + o(\tfrac{1}{m^2}) \\ &= -\tfrac{2}{m}\psi(k) + \tfrac{2}{m^2}\psi'(k) + o(\tfrac{1}{m^2})\end{aligned}$$

Hence

$$\left(\frac{\Gamma(k-\frac{2}{m})}{\Gamma(k)}\right)^m = \exp\left(-2\psi(k) + \frac{2}{m}\psi'(k) + o(\frac{1}{m})\right)$$

Similar computations give

$$\left(\frac{\Gamma(k-\frac{1}{m})}{\Gamma(k)}\right)^{-2m} = \exp\left(2\psi(k) - \frac{1}{m}\psi'(k) + o(\frac{1}{m})\right)$$

Hence

$$\begin{aligned}C_3(m,k) &= \sqrt{\exp(-2\psi(k) + \tfrac{2}{m}\psi'(k) + o(\tfrac{1}{m})) \cdot \exp(2\psi(k) - \tfrac{1}{m}\psi'(k) + o(\tfrac{1}{m})) - 1} \\ &= \sqrt{\exp(\tfrac{\psi'(k)}{m} + o(\tfrac{1}{m})) - 1}\end{aligned}$$

And

$$C_3(m,k) \underset{m\to\infty}{\sim} \sqrt{\psi'(k)} \cdot \frac{1}{\sqrt{m}},$$

finishing the proof.

4.7 Precision and Comparison of the algorithms

This section presents the proof of Theorem 2 (page 8). It is an analysis of the precision of the estimates of the three families, $\mathcal{P}_1(M,k)$, $\mathcal{P}_2(M,k)$ and $\mathcal{P}_3(M,k)$. (1) is direct. We prove here (2), that is, when k is large, we have for $i = 1, 2, 3$,

$$\mathcal{P}_i(M,k) \underset{k\to\infty}{\to} \frac{1}{\sqrt{M}}.$$

Inverse Family. Direct by the definition of the precision of the estimate of

the Inverse Family, $P_1(M,k)$:

$$\mathcal{P}_1(M,k) := \sqrt{\frac{k}{k-2}} \cdot \frac{1}{\sqrt{M}} \underset{k\to\infty}{\sim} \frac{1}{\sqrt{M}}.$$

Square Root Family. The precision of the estimate of the Square Root Family, $\mathcal{P}_2(M,k)$, is defined as

$$\mathcal{P}_2(M,k) := 2 \cdot \sqrt{k\left(\frac{1}{k-1}\left(\frac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2 - 1\right)} \cdot \frac{1}{\sqrt{M}}.$$

The Stirling approximation of the Gamma function says

$$\Gamma(s) = \frac{\sqrt{2\pi}}{\sqrt{s}}\frac{s^s}{e^s}(1 + \frac{1}{12s} + o(\frac{1}{s})).$$

Hence

$$\frac{\Gamma(k-\frac{1}{2})}{\Gamma(k)} = \frac{\sqrt{2\pi}\cdot\sqrt{k}}{\sqrt{2\pi}\cdot\sqrt{k-1/2}} \frac{e^k\cdot(k-1/2)^{k-1/2}}{e^{k-1/2}\cdot k^k} \frac{1+\frac{1}{12(k-1/2)}+o(\frac{1}{k})}{1+\frac{1}{12k}+o(\frac{1}{k})}$$

When k is large, we have the following Taylor expansions

$$\begin{aligned}
\tfrac{1}{\sqrt{k-\frac{1}{2}}} &= \tfrac{1}{\sqrt{k}}(1-\tfrac{1}{2k})^{-1/2} = \tfrac{1}{\sqrt{k}}(1+\tfrac{1}{4k}+o(\tfrac{1}{k})) \\
1+\tfrac{1}{12(k-\frac{1}{2})} &= 1+\tfrac{1}{12k}\tfrac{1}{1-\frac{1}{2k}} = 1+\tfrac{1}{12k}(1+\tfrac{1}{2k}+o(\tfrac{1}{k})) = 1+\tfrac{1}{12k}+o(\tfrac{1}{k})) \\
(k-1/2)^{k-1/2} &= k^{k-1/2}(1-\tfrac{1}{2k})^{k-1/2} = k^{k-1/2}\exp((k-1/2)\ln(1-\tfrac{1}{2k})) \\
&= k^{k-1/2}\exp((k-1/2)(-\tfrac{1}{2k}-\tfrac{1}{8k^2}+o(\tfrac{1}{k^2})) \\
&= k^{k-1/2}\exp(-\tfrac{1}{2}-\tfrac{1}{8k}+\tfrac{1}{4k}+o(\tfrac{1}{k})) \\
&= k^{k-1/2}\exp(-\tfrac{1}{2}+\tfrac{1}{8k}+o(\tfrac{1}{k})) = k^{k-1/2}e^{-1/2}(1+\tfrac{1}{8k}+o(\tfrac{1}{k}))
\end{aligned}$$

With these expansions, we get

$$\begin{aligned}
\tfrac{\Gamma(k-\frac{1}{2})}{\Gamma(k)} &= \tfrac{1}{\sqrt{k}}(1+\tfrac{1}{4k}+o(\tfrac{1}{k}))(1+\tfrac{1}{8k}+o(\tfrac{1}{k})) \\
&= \tfrac{1}{\sqrt{k}}(1+\tfrac{1}{4k}+\tfrac{1}{8k}+o(\tfrac{1}{k})) = \tfrac{1}{\sqrt{k}}(1+\tfrac{3}{8k}+o(\tfrac{1}{k})) \\
\left(\tfrac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2 &= \tfrac{1}{\frac{1}{k}(1+\frac{6}{8k}+o(\frac{1}{k}))} = k\cdot(1-\tfrac{3}{4k}+o(\tfrac{1}{k})).
\end{aligned}$$

As $\mathcal{P}_2(M,k)$ is defined as $\mathcal{P}_2(M,k) := 2 \cdot \sqrt{k\left(\frac{1}{k-1}\left(\frac{\Gamma(k)}{\Gamma(k-\frac{1}{2})}\right)^2 - 1\right)} \cdot \frac{1}{\sqrt{M}}$, we have

$$\begin{aligned}
\mathcal{P}_2(M,k) &= 2\cdot\sqrt{(1+\tfrac{1}{k}+o(\tfrac{1}{k}))(k-\tfrac{3}{4}+o(1))-k}\cdot\tfrac{1}{\sqrt{M}} \\
&= 2\cdot\sqrt{k-\tfrac{3}{4}+1+o(1)-k}\cdot\tfrac{1}{\sqrt{M}}.
\end{aligned}$$

Hence

$$\mathcal{P}_2(M,k) \underset{k\to\infty}{\sim} \frac{1}{\sqrt{M}}.$$

Logarithm Family. The precision of the estimate of the Logarithm Family, $\mathcal{P}_3(M,k)$ is defined as

$$\mathcal{P}_3(M,k) := \sqrt{k\cdot\psi'(k)}\cdot\frac{1}{\sqrt{M}}.$$

When k is large, the Euler Trigamma function $\psi'(k)$ is equivalent to $\frac{1}{k}$, giving

$$\mathcal{P}_3(M,k) \underset{k\to\infty}{\sim} \frac{1}{\sqrt{M}}.$$

It finishes the proof of Theorem 2.

5 Validations and experimentation results

To study the behaviors of the three families of estimates, we use files of different kinds and sizes, presented in Section 5.1. In Section 5.2, the relative error of the estimates is shown to be close to what expected from theory (see Figure 4). They have a very good tradeoff between their precision and the size of the memory: for instance, a memory of only 12kB is sufficient to build an estimate with an accuracy of 2 percents for a multiset with several million elements. In Section 5.3 the execution time of the algorithms is studied using an optimized implementation of the best practical estimate, MINCOUNT [2]. The algorithm only takes a few seconds to process files with millions of elements and is only 3 to 4 times slower than the very simple `unix` command `cat -T`, that just replaces the tab characters of a file with ^I (see Figure 6). Finally, we show in Section 5.4 how MINCOUNT can be used to detect attacks on a network, e.g. the spreading of the Code Red worm.

5.1 File suite

To validate the three families of algorithms we ran simulations using files of different kinds and sizes. Examples of these files are given in Figure 3. The size range is from few tens of thousands to tens of millions. Files are very diverse; english plays, e.g. `Hamlet`; router logs (traces available on the website of the NLANR Measurement and Network Analysis Group `http://moat.nlanr.net`);

[2] See project page, `http://algo.inria.fr/giroire/mincount.html`

Data file	Size (MB)	# elements	# distinct	Type
Hamlet	0.16	32,865	5,044	Text
Random	0.1	10,000	10,000	Random nbs
Tau	9.9	677,801	10,569	Router Logs
Psd-family	11	186,700	19,615	Proteinic seq
Access log	26	829,224	101,398	Access logs
Ind-230t	76	4,322,835	230,292	Router Logs
Psd-all	681	23,073,127	4,052,131	Proteinic seq
Large random	50	5,000,000	4,993,301	Random nbs
Auck-9M	242	13,856,690	9,083,474	Router Logs
100millions	848	10^8	10^8	Structured Text

Fig. 3. Simulation data suite.

access logs collected at the gateway of the INRIA Rocquencourt campus; proteinic sequences (available from the website of the database group of the University of Washington [3]); random number files and consecutive number files. `Random` is a set of 500 files containing 10,000 integers chosen uniformly at random. In the following section, the results for these files are an average over the set. `100millions` is a list of 100 million consecutive integers. This very structured set is used to verify the good alea properties of the algorithm.

5.2 Validation of the algorithms

We estimate here the number of distinct elements of the data suite files of Section 5.1 using estimators of the three families of estimates. Figure 4 shows a summary of typical results for the estimators built with the third minimum. Each of the three horizontal blocks corresponds to results for the estimator of one family. Different columns correspond to simulations with different values of m leading to different expected precisions (recall that m is the number of simulated experiences during the stochastic averaging process). A number in the table is the *experimental relative error*, i.e. the difference in percents between the estimate given by the algorithm and the exact value. For example, the number of different connections in the trace file `Ind-230t` is estimated with a precision of respectively 1.4 %, 4.7 % and 4.0 % by the estimates of the Inverse, Square Root and Logarithm Families. The third set of results,

[3] `http://www.cs.washington.edu/research/xmldatasets/`

	m	4	8	16	32	64	128	256	512	1024
$\frac{1}{M^{(3)}}$	$\%th$	50	35.4	25	17.7	12.5	8.8	6.25	4.4	3.1
	$Random$	53.2	33.5	25.9	17.9	14.3	9.4	6.1	4.3	3.0
	$Ind-230t$	31.0	32.9	10.4	1.9	10.5	9.4	1.4	0.3	0.05
	$Auck-9M$	10.3	4.0	5.1	10.0	2.8	3.9	1.7	1.8	3.0
$\frac{1}{\sqrt{M^{(3)}}}$	$\%th$	36.3	25.7	18.1	12.8	9.1	6.4	4.5	3.2	2.3
	$Random$	38.4	26.5	18.0	13.4	9.0	6.2	4.6	3.1	2.1
	$Ind-230t$	27.9	18.0	5.9	1.9	10.7	11.4	2.3	0.5	0.15
	$Auck-9M$	10.6	4.7	2.1	4.7	5.2	0.08	0.3	2.1	2.9
$\ln \frac{1}{M^{(3)}}$	$\%th$	34.0	23.1	16.0	11.2	7.9	5.6	3.9	2.8	2.0
	$Random$	34.9	22.3	16.0	11.8	8.0	5.5	4.0	2.6	1.9
	$Ind-230t$	25.8	3.5	1.3	4.9	10.6	12.2	3.2	1.2	0.3
	$Auck-9M$	10.7	6.1	0.2	0.6	6.2	2.7	0.6	2.7	2.9

Fig. 4. Relative error (in percents) of the estimates of the three families.

`Random`, corresponding to mean results over simulations on 500 files, validates the precision given in Theorem 1. We point out that the numbers are close to what is predicted by the theory. For example for $m = 32$, we obtain precisions of 17.9, 13.1 and 11.8 for the three families to compare with the expected precisions 17.7, 12.8 and 11.2, indicated in the lines `%th` in Figure 4. It validates the algorithms: the asymptotic regime is quickly reached and the values are well distributed by the hashing function.

The algorithms have a very good tradeoff between the precision and the size of the memory. For instance, for $m = 1024$, the algorithm of the Logarithm Family built on the third minimum (the best practical estimate) stores the three first minimums for m buckets. Using 32-bit floating numbers, this corresponds to a memory of only $12KB$. This is sufficient to build an estimate with accuracy of order 2 percents for a multiset with several million elements.

5.3 Execution time

The algorithms are motivated by the processing of very large multisets of data, in particular in the field of inline analysis of internet traffic. Most backbone networks operated today are Synchronous Optical NETworks (SONET). In these networks, the links are optical fibers classified according to their ca-

```
WHILE  hashed value
    IF  h < M^(2)
        IF  h < M
            M^(2) = M
            M = h
        ELSE M^(2) = h
```

Fig. 5. Internal loop for $k = 2$.

Test files	Mc (s)	Rate (MB/s)	Rate (Me/s)	Mc/ cat -T	Mc/ wc -l	wc -w/ Mc	sort -u | wc/ Mc
Tau	0.20	50.	3.4	4.0	10.	1.4	75.0
psd-family	0.17	65.	3.4	4.0	10.	1.4	58.8
access log	0.43	60.	1.9	3.3	11.	1.7	81.4
psd-all	11.7	58.	2.0	3.0	9.2	1.6	70.2
large random	1.1	45.	4.5	3.5	7.9	1.2	50.5
Auck-9M	4.6	53.	3.0	3.2	8.6	1.5	65.9
100millions	21.7	39.1	4.6	3.6	6.7	1.35	31.5

Fig. 6. MINCOUNT (Mc) execution times on the files of the data suite, corresponding throughput (in millions of bytes per second (MB/s) and millions of elements per second (Me/s)) and timing ratios between MINCOUNT and common `Unix` commands.

pacity from OC-1 (51.84 Mbps) to widely used OC-192 (10 Gbps) and even OC-768 (40 Gbps). It is crucial for carriers to know characteristics of the traffic for network monitoring and network design, see [11] and [18]. In particular, Estan, Varghese and Fisk inventory four major uses of the number of distinct connection statistics in [7]: general monitoring, detection of port scan attacks, detection of denial of service attacks (DoS attacks), and study of the spreading of a worm. At 40 Gbps speed a new packet arrives every 60 nanoseconds, assuming an average packet size of 300 bytes, see [11]. This allows only 150 operations per packet on a 2.5 GHz processor ignoring the significant time taken by the data to enter the processor. Thus execution times of algorithms are crucial in this context. The algorithms of the three families are mainly composed of a *very simple internal loop* that finds the k-th minimum of a multiset. This loop is given for $k = 2$ in Figure 5. Typically, only one comparison is performed in each loop iteration. Hence the algorithms are quite efficient.

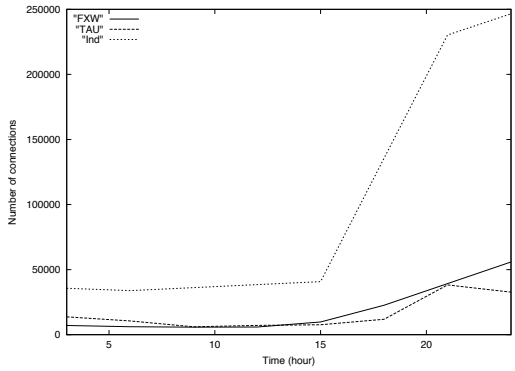


Fig. 7. Connections Peak during the Spreading of the Code Red Worm

MINCOUNT, an optimized implementation of the best practical estimate, is used here. Figure 6 shows MINCOUNT execution times in seconds while processing the files of the data suite. The algorithm takes only few seconds to give an estimate of the cardinality of files with millions of elements. For example, 4.8 seconds are enough to process the trace `Auck-9M` with 13 million elements, including 9 million distinct ones. This corresponds to a rate of 53 MegaBytes per second or 3 million elements per second. The following `Unix` command are used as references for the execution time: `cat -T` that reads the file and displays tab characters as ^I; `wc -l` that displays the number of lines of the file; `wc -w` that displays the number of words of the file; `sort -u | wc` that gives the number of distinct lines of the file. Figure 6 shows the ratios between the execution time of MINCOUNT and the one of the `Unix` commands. The algorithm is only between 3 to 4 times slower than the `cat -T` command that only replace the tab characters in its input by ^I.

5.4 Code Red attacks

We also simulate an analysis of internet traffic to show a typical use of the algorithms, using MINCOUNT. The NLANR Measurement and Network Analysis Group is doing daily network monitoring. We analyze their traces of July 19th 2001 when a Code Red Worm variant started spreading. The Code Red Worm was designed to spread very fast. More than 359,000 computers were infected in less than 14 hours and at the peak of the infection, more than 2,000 new hosts were infected each minute. We considered three sets of traces: one monitored in the Indiana University MegaPOP (Ind), one in the FIX West facility at NASA Ames (FXW) and the last one in Tel Aviv University (TAU). The traces correspond to a window of 1 minute 30 every three hours. We use the algorithm to estimate within 4 % (m=256) the number of active connections using this link during each of these period of times. Results are shown in Figure 7. It is of course a very rough analysis and more data for other links, other days for example would be needed to give precise conclusions about the spread of the worm. But we are able to detect a change of the activity of

the network caused by the infected hosts in the network. We see a very net increase of the number of active connections starting from 3 pm. For the Ind link, the usual load seems to be around 35,000 connections, 33842 at 6 am. At its peak at midnight we estimate a number of 246,558 connections, around 7 times more. Same observation for TAU and FXW: respectively 7,629 and 9,793 connections at 3 pm and 32,670 and 55,877 at midnight. So, by monitoring a link using our algorithm, we are able to see, using constant memory, unusual increase of the traffic, to detect an attack and to give rough indications about its propagation and extent in some parts of the network.

Acknowledgements. I would like to thank my PhD advisor, Philippe Flajolet, for introducing me to the subject of the paper, his help and numerous remarks on this work.

References

[1] Z. Bar-Yossef, T. S. Jayram, R. Kumar, D. Sivakumar, and Luca Trevisan. Counting distinct elements in a data stream. In *RANDOM '02: Proceedings of the 6th International Workshop on Randomization and Approximation Techniques*, pages 1–10. Springer-Verlag, 2002.

[2] A. Broder. On the resemblance and containment of documents. In *SEQUENCES '97: Proceedings of the Compression and Complexity of Sequences 1997*, page 21, Washington, DC, USA, 1997. IEEE Computer Society.

[3] Andrei Z. Broder. Identifying and filtering near-duplicate documents. In *COM '00: Proceedings of the 11th Annual Symposium on Combinatorial Pattern Matching*, pages 1–10, London, UK, 2000. Springer-Verlag.

[4] P. Chassaing and L. Gerin. Efficient estimation of the cardinality of large data sets. In *math.ST/0701347, 2007. Extended abstract in the proceedings of the 4th Colloquium on Mathematics and Computer Science, 2006, pages 419-422*, 2007.

[5] J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In *ICDE '04: Proceedings of the 20th International Conference on Data Engineering*, page 449, Washington, DC, USA, 2004. IEEE Computer Society.

[6] M. Durand and P. Flajolet. Loglog counting of large cardinalities. In G. Di Battista and U. Zwick, editors, *Annual European Symposium on Algorithms (ESA03)*, volume 2832 of *Lecture Notes in Computer Science*, pages 605–617, September 2003.

[7] C. Estan, G. Varghese, and M. Fisk. Bitmap algorithms for counting active flows on high-speed links. *IEEE/ACM Trans. Netw.*, 14(5):925–937, 2006.

[8] P. Flajolet. Adaptive sampling. In M. Hazewinkel, editor, *Encyclopae-*

dia of Mathematics, volume Supplement I, page 28. Kluwer Academic Publishers, Dordrecht, 1997.

[9] P. Flajolet, E. Fusy, O. Gandouet, and F. Meunier. Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm. In Philippe Jacquet, editor, *Analysis of Algorithms 2007 (AofA07)*, Discrete Mathematics and Theoretical Computer Science Proceedings, 2007. In press.

[10] P. Flajolet and P. N. Martin. Probabilistic counting. In *Proceedings of the 24th Annual Symposium on Foundations of Computer Science*, pages 76–82. IEEE Computer Society Press, 1983.

[11] C. Fraleigh, C. Diot, B. Lyles, S. Moon, P. Owezarski, K. Papagiannaki, and F. Tobagi. Design and Deployment of a Passive Monitoring Infrastructure. In *Passive and Active Measurement Workshop*, Amsterdam, April 2001.

[12] É. Fusy and F. Giroire. Estimating the number of active flows in a data stream over a sliding window. In David Appelgate, editor, *Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments and the Fourth Workshop on Analytic Algorithmics and Combinatorics*, pages 223–231. SIAM Press, 2007. Proceedings of the New Orleans Conference.

[13] L. Getoor, B. Taskar, and D. Koller. Selectivity estimation using probabilistic models. In *SIGMOD Conference*, 2001.

[14] P. B. Gibbons. Distinct sampling for highly-accurate answers to distinct values queries and event reports. In *The VLDB Journal*, pages 541–550, 2001.

[15] F. Giroire. Order statistics and estimating cardinalities of massive data sets. In Conrado Martnez, editor, *2005 International Conference on Analysis of Algorithms*, volume AD of *DMTCS Proceedings*, pages 157–166. Discrete Mathematics and Theoretical Computer Science, 2005.

[16] F. Giroire. Directions to use probabilistic algorithms for cardinality for DNA analysis. In *Journées Ouvertes Biologie Informatique Mathématiques (JOBIM 2006)*, pages 3–5, July 2006.

[17] F. Giroire. *Réseaux, algorithmique et analyse combinatoire de grands ensembles*. PhD thesis, Université Paris VI, November 2006.

[18] G. Iannaccone, C. Diot, I. Graham, and N. McKeown. Monitoring very high speed links. In *ACM SIGCOMM Internet Measurement Workshop*, San Francisco, CA, November 2001.

[19] D. E. Knuth. *The Art of Computer Programming*, volume 3: Sorting and Searching. Addison-Wesley, 1973.

[20] K.-Y. Whang, B. T. V. Zanden, and H. M. Taylor. A linear-time probabilistic counting algorithm for database applications. In *TODS 15, 2*, pages 208–229, 1990.