Bucket Sort When You Know The Distribution.

David Ponarovsky

April 2, 2023

Abstract

None

The problem. Let $f : [0,1] \to [0,1]$ a fixed distribution function. Write an algorithm that sorts n draws $x_1 ... x_n$ at linear expectation time.

Solution. We will define a partition of the input into a series of n buckets $\mathcal{B} = \{B_k = [t_k, t_{k+1}] : k \in [n]\}$ such that $\mathbf{Pr}[x \in B_i] = \frac{1}{n}$ for any bucket. Assume that we succeed in computing the buckets efficiently. Let the X_{ij} be the indicator of the event that x_j falls to B_i. Then we have:

$$\begin{aligned}\mathbf{Pr}\left[\sum_i |B_i|^2 \geq t\right] &= \mathbf{Pr}\left[\sum_i \left(\sum_j X_{ij}\right)^2 \geq t\right] \\ = \mathbf{Pr}\left[\sum_{i,j,j'} X_{i,j}X_{i,j'} \geq t\right] &= \mathbf{Pr}\left[\sum_{i,j\neq j'} X_{i,j}X_{i,j'} \geq t-n\right] \\ \leq \frac{\sum_{i,j\neq j'} \mathbf{E}\left[X_{ij}X_{ij'}\right]}{t-n} &= \frac{n}{(t-n)\,n^2} 2\binom{n}{2} \leq \frac{n}{t-n}\end{aligned}$$

It follows that for any function $t : \mathbb{N} \to \mathbb{R}$, such that $n = o(t)$, sorting quadric each bucket at turn would last almost surely less than $t(n)$. It shows that knowing the distribution enables one to compute the buckets efficiently. Ensuring the uniform partitioned property leads to the following recursive relation:

$$\begin{aligned}\frac{1}{n} &= \mathbf{Pr}\left[x \in B_k\right] = f\left(t_{k+1}\right) - f\left(t_k\right) \\ &\Rightarrow t_{k+1} \leftarrow f^{-1}\left(\frac{1}{n} + f\left(t_k\right)\right)\end{aligned}$$

Hence, if f can be computed in sublinear time, then we obtained an expected linear time algorithm for sorting $\square$ The result above demonstrates a case when knowing how the input is distributed turns the problem equivalent to facing a uniform distributed input.