COMPSCI 514: Problem Set 1

Due: 9/23 by 11:59pm in Gradescope.

Instructions:

- You are allowed to, and highly encouraged to, work on this problem set in a group of up to three members.
- Each group should **submit a single solution set**: one member should upload a pdf to Gradescope, marking the other members as part of their group in Gradescope.
- You may not talk with anyone outside your group about the homework expect for the instructor and TAs.
- You must show your work/derive any answers as part of the solutions to receive full credit.

1. Probability Practice (10 points)

1. (2 points) Consider storing n items in a hash table with $m = n$ buckets, using a fully random hash function $\mathbf{h} : [n] \to [n]$ (i.e., each item is assigned independently to a uniform random bucket). What is the expected fraction of buckets that have at least one item in them? What is the limit of this value as $n \to \infty$? What about when $m = 2n$? **Hint:** Use linearity of expectation.

2. (2 points) I store $1,000$ items in a hash table with $100,000$ buckets, using a fully random hash function. What is the probability that there is **at least 1** collision. What if I use $1,000,000$ buckets? What is the probability that there is at least one collision?

3. (2 points) Prove that $\text{Var}(\mathbf{X} + \mathbf{Y}) = \text{Var}(\mathbf{X}) + \text{Var}(\mathbf{Y}) + 2\mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{Y} - \mathbb{E}[\mathbf{Y}])]$.

4. (2 points) Design two random variables $\mathbf{X}$ and $\mathbf{Y}$ that satisfy $\text{Var}(\mathbf{X}) + \text{Var}(\mathbf{Y}) > \text{Var}(\mathbf{X} + \mathbf{Y})$. Design two different random variables that satisfy $\text{Var}(\mathbf{X}) + \text{Var}(\mathbf{Y}) < \text{Var}(\mathbf{X} + \mathbf{Y})$

5. (2 points) Describe a random variable $\mathbf{X}$ with $\mathbb{E}[\mathbf{X}] = 0$, $\mathbb{E}[\mathbf{X}^2] = 1$ and $\mathbb{E}[\mathbf{X}^4] = 1$. Prove that this is the only random variable satisfying these three conditions. **Hint:** In your proof, you may want to use that for any random variable $\mathbf{Y}$, $\text{Var}[\mathbf{Y}] = \mathbb{E}[\mathbf{Y}^2] - \mathbb{E}[\mathbf{Y}]^2$.

2. More Independence, Fewer Collisions (6 points)

As discussed in class, in practice, a *fully random hash function* that maps any input to a uniform and independently chosen output is not efficiently implementable. So, random hash functions that approximate the behavior of a fully random hash function are often used. In class, we talked about

2-universal hash functions, which have collision probability $\leq 1/n$ for any two items. A k-universal hash function $\mathbf{h} : U \to [n]$ is any random hash function that satisfies, for any inputs $x_1, \ldots, x_k \in U$,

$$\Pr[\mathbf{h}(x_1) = \mathbf{h}(x_2) = \ldots = \mathbf{h}(x_k)] \leq \frac{1}{n^{k-1}}.$$

1. (2 points) Suppose you hash n balls into n hash buckets using a 2-universal hash function. Show that for $t = 5n$, the number of pairwise collisions exceeds t with probability at most 1/10.

2. (2 points) Use the above to argue that for $t = 4\sqrt{n}$, that the maximum load on any bin exceeds t with probability at most 1/10, when hashing n balls into n hash buckets. **Hint:** Don't use a union bound.

3. (2 points) Generalize this result to k-universal hash functions for $k > 2$. Show that if $t = cn^{1/k}$ for large enough constant c, that the probability of the maximum load exceeding t at most 1/10. **Hint:** Instead of pairwise collisions, consider the expected number of *k-wise collisions*.

3. Stacking Hash Tables (6 points)

In class we show that if we store n items in a hash table with m buckets, using a fully random hash function $\mathbf{h} : [n] \to [m]$ (i.e., each item is assigned independently to a uniform random bucket), then $m = cn^2$ for some sufficiently large constant c, there are *no collisions* with probability at least 9/10. Thus, the table has worst case $O(1)$ lookup time, but very large space complexity. We proposed 2-level hashing as a way to reduce this space complexity. Here we will analyze an alternative scheme.

1. (1 point) Consider storing n items in two hash tables with m buckets each, using two different fully random hash functions $\mathbf{h}_1 : [n] \to [m]$, $\mathbf{h}_2 : [n] \to [m]$. An item x is stored in bucket $\mathbf{h}_1(x)$ of the first table, unless this bucket already has an item in it. In that case, it is stored in bucket $\mathbf{h}_2(x)$ of the second table. Assuming that there are no collisions (i.e., that all buckets in the second table have at most one item in them), what is the worst case lookup time for this scheme?

2. (1 point) Describe one possible advantage of this scheme over 2-level hashing.

3. (4 points) How large must m be such that, with probability at least 9/10, there are no collisions in this scheme? **Hint:** First bound the number of collisions in the first table. Then use this to bound the number of collisions in the second table.

4. Concentration with k-wise Independence (10 points)

We say a set of random variables $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ is k-wise independent if for any subset S of $[n]$ with at most k elements then

$$\Pr[\bigcap_{i \in S} \{\mathbf{X}_i = j_i\}] = \prod_{i \in S} \Pr[\mathbf{X}_i = j_i] \qquad \forall j_1, j_2, \ldots, j_{|S|} .$$

1. (2 points) Suppose $\mathbf{X}_1$ and $\mathbf{X}_2$ are independent random variables that are each equally likely to be -1 or 1 and that $\mathbf{X}_3 = \mathbf{X}_1/\mathbf{X}_2$. Are the variables $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ 2-wise independent? Are they 3-wise independent? Prove your answers.

2. (2 points) Prove that if $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ are 2-wise independent then $\mathrm{Var}(\sum_{i\in[n]} \mathbf{X}_i) = \sum_{i\in[n]} \mathrm{Var}(\mathbf{X}_i)$.

3. (2 points) Suppose $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ are 2-wise independent and each $\mathbf{X}_i$ is equally likely to be 0 or 2. Let $\mathbf{X} = \frac{1}{n}\sum_{i\in[n]} \mathbf{X}_i$. Prove the best upper bound you can on the probability $\Pr[|\mathbf{X} - 1| \geq 0.1]$ if $n = 1000$.

4. (2 points) Suppose $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ are 4-wise independent and each $\mathbf{X}_i$ is equally likely to be 0 or 2. Let $\mathbf{X} = \frac{1}{n}\sum_{i\in[n]} \mathbf{X}_i$. Prove the best upper bound you can on the probability $\Pr[|\mathbf{X} - 1| \geq 0.1]$ if $n = 1000$. **Hint:** You may use the fact that $\mathbb{E}[(\mathbf{X} - 1)^4] = 3/n^2 - 2/n^3$.

5. (2 points) Consider the CAPTCHA example discussed in class. Let $\mathbf{D} = \sum_{i<j, i,j\in[m]} \mathbf{D}_{ij}$ be the total number of pairwise duplicates when drawing m CAPTCHAS from a database of size n. For $m = 1000$ and $n = 1000000$ prove the best upper bound you can on $\Pr[\mathbf{D} \geq 10]$. How does this compare to the bound proven using Markov's inequality in class? **Hint:** Start by arguing that the $\mathbf{D}_{ij}$ random variables are pairwise independent.