Chapter 2

Basics of Probability and Tail Inequalities

Randomized algorithms use random coin tosses to guide the progress of the algorithm. Although the actual performance of the algorithm may depend on the outcomes of these coin tosses, it turns out that one can often show that with reasonable probability, the algorithm has the desired properties. This model can dramatically improve the power of an algorithm. We will give examples where this ability can lead to very simple algorithms, and in fact sometimes randomization turns out to be necessary. In this chapter, we begin with the basics of probability. We relate the notion of a random variable with the analysis of a randomized algorithm – often, the running time of a randomized algorithm will be a random variable. We will then describe techniques for bounding the probibility of a random variable exceeding certain values, thereby bounding the running time.

Note Since randomized techniques have been used extensively used as a basic tool, this chapter lays down some of the foundations of such applications for readers who are not familiar with this methodology. For others, this chapter can be used as reference as required.

2.1 Basics of Probability Theory

In this section, we do a brief review of the axiomatic approach to probability theory. We will deal with the discrete case only. We begin with the notion of a sample space, often denoted by Ω. It can be thought of as the set of outcomes (or elementary events) in an experiment. For example, if we are rolling a dice, then Ω can be defined as the set of 6 possible outcomes. In an abstract setting, we will define Ω to be any set (which will be finite or countably infinite). To see an example where Ω can

be infinite, consider the following experiment: we keep tossing a coin till be see a Heads. Here the set of possible outcomes are infinite – for any integer $i \geq 0$, there is an outcome consisting of i Tails followed by a Heads. Given a sample space Ω, a *probability measure* Pr assigns a non-negative real value p_ω to each elementary event $\omega \in \Omega$. The probability measure Pr should satisfy the following condition:

$$\sum_{\omega \in \Omega} p_\omega = 1. \tag{2.1.1}$$

A *probability space* consists of a sample space Ω with a *probability measure* associated with the elementary events. In other words, a probability space is specified by a pair $(\Omega, \Pr)$ of sample space and probability measure. Observe that the actual probability assigned to each elementary event (or outcome) is part of the axiomatic definition of a probability space. Often one uses prior knowledge about the experiment to come up with such a probability measure. For example, if we assume that a dice is fair, then we could assign equal probability, i.e., 1/6 to all the 6 outcomes. However, if we suspect that the dice is biased, we could assign different probabilities to different outcomes.

Example 2.1 *Suppose we are tossing 2 coins. In this case the sample space is* $\{HH, HT, TH, TT\}$. *If we think all 4 outcomes are equally likely, then we could assign probability 1/4 to each of these 4 outcomes. However, assigning probability* $0.3, 0.5, 0.1, 0.1$ *to these 4 outcomes also results in a probability space.*

We now define the notion of an *event*. An event is a subset of Ω. Probability of an event E is defined as $\sum_{\omega \in E} p_\omega$, i.e., the total sum of probabilities of all the outcomes in E.

Example 2.2 *Consider the experiment of throwing a dice, i.e.,* $\Omega = \{1, 2, 3, 4, 5, 6\}$, *and suppose the probabilities of these outcomes (in this sequence) are 0.1, 0.2, 0.3, 0.2, 0.1, 0.1. Then* $\{2, 4, 6\}$ *is an event (which can be also be defined as the event that the outcome is an even number) whose probability is* $0.2 + 0.2 + 0.1 = 0.5$.

The following properties follow immediately from the definition of the probability of an event (proof deferred to exercises):

1. For all $A \subset \Omega$, $0 \leq \Pr[A] \leq 1$
2. $\Pr[\Omega] = 1$
3. For mutually disjoint events $E_1, E_2 \ldots$, $\Pr[\cup_i E_i] = \sum_i \Pr[E_i]$

The principle of Inclusion-Exclusion also has its counterpart in the probabilistic world, namely

Lemma 2.1

$$\Pr[\cup_i E_i] = \sum_i \Pr[E_i] - \sum_{i<j} \Pr[E_i \cap E_j] + \sum_{i<j<k} \Pr[E_i \cap E_j \cap E_k] \ldots$$

Example 2.3 *Suppose we pick a number uniformly at random from 1 to 1000. We would like to calculate the probability that it is divisible by either 3 or 5. We can use the principle of inclusion-exclusion to calculate this. Let E be the event that it is divisible by either 3 or 5. Let E_1 be the event that it is divisible by 3 and E_2 be the event that it is divisible by 5. By the inclusion-exclusion principle*

$$\Pr[E] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \cap E_2].$$

Clearly E_1 happens if we pick a multiple of 3. The number of multiples of 3 in the range $[1, 1000]$ is $\lfloor 1000/3 \rfloor = 333$, and so, $\Pr[E_1] = \frac{333}{1000}$. Similarly, $\Pr[E_2] = \frac{200}{1000}$. It remains to compute $\Pr[E_1 \cap E_2]$. But note that this is exactly the probability that the number is divisible by 15, and so, it is equal to $\frac{\lfloor 1000/15 \rfloor}{1000} = \frac{66}{1000}$. Thus, the desired probability is $467/1000$.

Definition 2.1 *The* conditional probability *of E_1* given *E_2 is denoted by $\Pr[E_1|E_2]$ and is given by*

$$\frac{\Pr[E_1 \cap E_2]}{\Pr[E_2]}$$

assuming $\Pr[E_2] > 0$.

Definition 2.2 *A collection of events $\{E_i | i \in I\}$ is* independent *if for all subsets $S \subset I$*

$$\Pr[\cap_{i \in S} E_i] = \Pi_{i \in S} \Pr[E_i]$$

Remark E_1 and E_2 are independent if $\Pr[E_1|E_2] = \Pr[E_1]$.

The notion of independence often has an intuitive meaning – if two events depend on experiments which do not share any random bits respectively, then they would be independent. However, the converse may not be true, and so the only way to verify if two events are independent is to check the condition above.

Example 2.4 *Suppose we throw two die. Let E_1 be the event that the sum of the two numbers is an even number. It is easy to check that $E_1 = 1/2$. Let E_2 be the event that the first dice has outcome "1". Clearly, $\Pr[E_2] = 1/6$. It is also clear that $\Pr[E_1 \cap E_2]$ is $1/12$ – indeed, for $E_1 \cap E_2$ to occur, the second dice can have only 3 outcomes. Since $\Pr[E_1 \cap E_2] = \Pr[E_1] \cdot \Pr[E_2]$, these two events are independent.*

We now come to the notion of a *random variable*.

Definition 2.3 *A* random variable *(r.v.) X is a real-valued function over the sample space, $X : \Omega \to \mathbb{R}$.*

In other words, a random variable assigns a real value to each outcome of an experiment.

Example 2.5 *Consider the probability space defined by the throw of a fair dice. Let X be function which is 1 if the outcome is an even number, and 2 if the outcome is an odd number. Then X is a random variable. Now consider the probability space defined by the throw of 2 fair die (where each of the 36 outcomes are equally likely). Let X be a function which is equal to the sum of the values of the two die. Then X is also a random variable which takes values in the range $\{2, \ldots, 12\}$.*

With each random variable X, we can associate several events. For example, given a real x, we can define the event $[X \geq x]$ as the set $\{\omega \in \Omega : X(\omega) \leq x\}$. One can similarly define the events $[X = x], [X < x]$, and in fact $[X \in S]$ for any subset S of real numbers[1]. The probability associated with the event $[X \leq x]$ (respectively, $[X < x]$) are known as *cumulative density function*, cdf (respectively *probability density function* or pdf) and help us to characterize the behavior of the random variable X. As in the case of events, one can also define the notion of independence for random variables. Two random variables X and Y are said to be independent if for all x and y in the range of X and Y respectively

$$\Pr[X = x, Y = y] = \Pr[X = x] \cdot \Pr[Y = y].$$

It is easy to check from the above definition that if X and Y are independent random variables, then

$$\Pr[X = x | Y = y] = \Pr[X = x].$$

As in the case of events, we say that a set of random variables $X_1, \ldots, X_n$ are mutually independent if for all reals $x_1, \ldots, x_n$, where x_i lies in the range of X_i, for all $i = 1, \ldots, n$,

$$\Pr[X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n] = \prod_{i=1}^{n} \Pr[X_i = x_i].$$

The *expectation* of a r.v. X, whose range lies in a (countable) set R, is denoted by $\mathbb{E}[X] = \sum_{x \in R} x \cdot \Pr[X = x]$. The expectation can be thought of as the typical value of X if we conduct the corresponding experiment. One can formalise this intuition – the law of large number states that if we repeat the same experiment many times, then the average value of X is very close to $\mathbb{E}[X]$ (and gets arbitrarily close as the number of experiments goes to infinity).

[1] We are only considering the case when X can be countably many different values.

A very useful property of expectation, called the *linearity property*, can be stated as follows

Lemma 2.2 *If X and Y are random variables, then*

$$\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$$

Remark Note that X and Y do not have to be independent !
Proof: Assuming R be the union of ranges of X and Y – we will assume that R is countable, though the result holds in general as well. We can assume that both X and Y have range R (if $r \in R$ is not in the range of X, we can add it to the range of X with the provision that $\Pr[X = r] = 0$). Then,

$$\mathbb{E}[X+Y] = \sum_{r_1 \in R, r_2 \in R} (r_1 + r_2) \Pr[X = r_1, Y = r_2].$$

If X and Y were independent, we could have just written $\Pr[X = r_1, Y = r_2]$ as $\Pr[X = r_1] \cdot \Pr[Y = r_2]$, and the result would follow trivially.

We proceed as follows:

$$\sum_{r_1 \in R, r_2 \in R} (r_1 + r_2) \Pr[X = r_1, Y = r_2] = \sum_{r_1 \in R, r_2 \in R} r_1 \cdot \Pr[X = r_1, Y = r_2] + \sum_{r_1 \in R, r_2 \in R} r_2 \Pr[X = r_1, Y = r_2] \quad (2.1.2)$$

If X and Y were independent, we could have just written $\Pr[X = r_1, Y = r_2]$ as $\Pr[X = r_1] \cdot \Pr[Y = r_2]$, and the result would follow trivially.
Now observe that $\sum_{r_1 \in R, r_2 \in R} r_1 \cdot \Pr[X = r_1, Y = r_2]$ can be written as $\sum_{r_1 \in R_1} r_1 \cdot \sum_{r_2 \in R_2} \Pr[X = r_1, Y = r_2]$. But now observe that $\sum_{r_2 \in R_2} \Pr[X = r_1, Y = r_2]$ is just $\Pr[X = x_1]$, and so $\sum_{r_1 \in R_1} r_1 \cdot \sum_{r_2 \in R_2} \Pr[X = r_1, Y = r_2]$ is same as $\mathbb{E}[X]$. One can similarly show that the other term in the RHS of (2.1.2) is equal to $\mathbb{E}[Y]$. □

The linearity of expectation property has many surprising applications, and can often be used to simplify many intricate calculations.

Example 2.6 *Suppose we have n letters meant for n different people (with their names written on the respective letters). Suppose we randomly distribute the letters to the n people (more formally, we assign the first letter to a person chosen uniformly at random, the next letter to a uniformly chosen person from the remaining $n-1$ persons, and so on). Let X be the number of persons who receive the letter meant for them. What is the expectation of X? We can use the definition of X to calculate this quantity, but the reader should check that even the expression of $\Pr[X = r]$ is non-trivial, and then, adding up all such expressions (weighted by the corresponding probability) is*

a long calculation. We can instead use linearity of expectation to compute $\mathbb{E}[X]$ *in a very simple manner as follows. For each person* i*, we define a random variable* X_i*, which takes only two values – 0 or 1* [2]*. We set* X_i *to 1 if this person receives the correct letter, otherwise to 0. It is easy to check that* $X = \sum_{i=1}^{n} X_i$*, and so, by linearity of expectation,* $\mathbb{E}[X] = \sum_i \mathbb{E}[X_i]$*. It is easy to compute* $\mathbb{E}[X_i]$*. Indeed it is equal to* $0 \cdot \Pr[X_i = 0] + 1 \cdot \Pr[X_i = 1] = \Pr[X_i = 1]$*. Now* $\Pr[X_i = 1]$ *is* $1/n$ *because this person receives each of the* n *letters with equal probability. Therefore,* $\mathbb{E}[X] = 1$*.*

Lemma 2.3 *For independent random variables* X, Y*,*

$$\mathbb{E}[X \cdot Y] = \mathbb{E}[X] \cdot \mathbb{E}[Y]$$

Proof:

$$\begin{aligned}
\mathbb{E}[XY] &= \sum_i \sum_j x_i \cdot y_j P(x_i, y_j) \text{ where } P \text{ denotes joint distribution ,} \\
&= \sum_i \sum_j x_i \cdot y_j p_X(x_i) \cdot p_Y(y_j) \text{ from independence of } X, Y \\
&= \sum_i x_i p_X(x_i) \sum_j y_j p_Y(y_j) \\
&= \mathbb{E}[X] \cdot \mathbb{E}[Y]
\end{aligned}$$

□

As in the case of events, we can also define conditional expectation of a random variable given the value of another random variable. Let X and Y be two random variables. Then, the *conditional expectation* of X given $[Y = y]$ is defined as

$$\mathbb{E}[X|Y = y] = \sum_x \Pr x \cdot [X = x|Y = y]$$

The **theorem of total expectation** that can be proved easily states that

$$\mathbb{E}[X] = \sum_y E[X|Y = y]$$

[2]These are called indicator random variables and often simplify calculations in many situations.

2.2 Tail inequalities

In many applications, especially in the analysis of randomized algorithms, we would like to bound the running time of our algorithm (or the value taken by some other random variable). Although one can compute the expectation of a random variable, it may not give any useful information about how likely the random variable is going to be close to its expectation. For example, consider a random variable which is uniformly distributed in the interval $[0, n]$, for some large number n. Its expectation is $n/2$, but the probability that it lies in the interval $[n/2(1-\delta), n/2(1+\delta)]$ is only 2δ, where δ is a small constant. We will see examples of other random variables where this probability will be very close to 1. Therefore, to say something more meaningful about a random variable, one needs to look beyond its expectation. The law of large number states that if we take many independent trials of a random variable, then the average value taken by the random variable over these trials converges (almost certainly) to the expectation. However, it does not say anything about how fast this convergence happens, or how likely the random variable is going to be close to its expectation if we perform this experiment only once.

In this section, we give various inequalities which bound the probability that a random variable deviates from its expectation by a large amount. The foremost such inequality is the Markov's inequality, which just uses the expectation of a random variable. As mentioned above, it may not yield very strong bounds, but it is the best one can say when we do not have any other information about the random variable.

As a running example, we will use a modification of the experiment considered in the previous chapter. We are given an array A of size m (which is even). Half of the elements in A are colored red and the rest are colored green. We perform the following experiment n times independently: pick a random element of A, and check its color. Define X as a random variable which counts the number of times we picked a green element. It is easy to show, using linearity of expectation, that $\mathbb{E}[X]$ is $n/2$. We would now be interested in tail inequalities which bound the probability that X deviates from its mean.

Markov's inequality Let X be a non-negative random variable. Then

$$\Pr[X \geq k\mathbb{E}[X]] \leq \frac{1}{k} \tag{2.2.3}$$

This result is really an "averaging" argument (for example, in any class consisting of n students, at most half the students can get twice the average marks). The proof of this result also follows easily. Let R be the range of $X \geq 0$.

$$\begin{aligned}\mathbb{E}[X] = \sum_{r \in R} r \cdot \Pr[X = r] \geq \sum_{r \in R : r \geq k\mathbb{E}[X]} r \cdot \Pr[X = r] &\geq k\mathbb{E}[X] \cdot \sum_{r \in R : r \geq k\mathbb{E}[X]} \Pr[X = r] \\ &= k\mathbb{E}[X] \Pr[X \geq k\mathbb{E}[X]]\end{aligned}$$

Cancelling $\mathbb{E}[X]$ on both sides yields Markov's inequality. Unfortunately there is no symmetric result which bounds the probability of events $[X < k\mathbb{E}[X]]$, where $k < 1$. To see why Markov's inequality cannot yield a two-sided bound, consider the following example.

Example 2.7 *Let X be a random variable which takes two values - 0 with proability $(1 - 1/n)$, and n^2 with probability $1/n$ (think of n as a large number). Then $\mathbb{E}[X]$ is n. However, $\Pr[X < n/2]$ is $1 - 1/n$, which is very close to 1.*

We now apply this inequality on our running example.

Example 2.8 *In the example of array A with elements colored red or green, we know that $\mathbb{E}[X] = n/2$. Therefore, we see that $\Pr[X > 3n/4] \leq 1/4$.*

Note that we get a very weak bound on the probability that $[X \geq 3n/4]$ in the example above. Ideally, one would think that the probability of this event would go down as we increase n (and indeed, this is true). However, Markov's inequality is not strong enough to prove this. The reason for this is that one can easily design random variables X whose expectation is $n/2$ but the probability of going above $3n/4$ is at most $2/3$. The extra information, that X is a sum of several independent random variables, is not exploited by Markov's inequality. Also, notice that we cannot say anything about the probability of the event $[X \leq n/4]$ using Markov's inequality. We now show that there are inequalities which can exploit facts about higher moments of X, and give stronger bounds.

The notion of *expectation of random variable* can be extended to functions $f(X)$ of random variable X in the following natural way (we can think of $Y := f(X)$ as a new random variable)

$$E[f(X)] = \sum_i p_i f(X = i)$$

The variance of a random variable is given by $\mathbb{E}[X^2] - \mathbb{E}[X]^2$. Consider the random variable X in Example 2.7. Its variance is equal to

$$\mathbb{E}[X^2] - \mathbb{E}[X]^2 = n^3 - n^2.$$

Let us now compute the variance of the random variable in our running example. We first show that if X_1 and X_2 are two independent random variables then variance of $X_1 + X_2$ is sum of the variance of the two random variables. The variance of $X_1 + X_2$ is given by

$$\begin{aligned} \mathbb{E}[(X_1 + X_2)^2] - \mathbb{E}[X_1 + X_2]^2 &= \mathbb{E}[X_1^2] + \mathbb{E}[X_2^2] + 2\mathbb{E}[X_1 X_2] - \mathbb{E}[X_1]^2 - \mathbb{E}[X_2]^2 - 2\mathbb{E}[X_1]\mathbb{E}[X_2] \\ &= \mathbb{E}[X_1^2] - \mathbb{E}[X_1]^2 + \mathbb{E}[X_2^2] - \mathbb{E}[X_2]^2, \end{aligned}$$

because $\mathbb{E}[X_1 X_2] = \mathbb{E}[X_1]\mathbb{E}[X_2]$ (we use independence of these two random variables here). The same observation extends by induction to sum of several random variables. Let us apply this observation to our running example. Let X_i be the random variable which is 1 if we pick a green element on the i^{th} trial, 0 otherwise. Variance of X_i is $\mathbb{E}[X_i^2] - \mathbb{E}[X_i]^2$. Since X_i is a 0-1 random variable, $\mathbb{E}[X_i^2] = \mathbb{E}[X_i]$, and so, its variance is $1/2 - 1/4 = 1/4$. Let X denote the total number of green elements seen. So, $X = \sum_{i=1}^{n} X_i$ and its variance is $n/4$.

If we have bounds on the variance of a random variable, then the following gives a stronger tail bound

Chebychev's inequality

$$\Pr[|X - \mathbb{E}[X]| \geq t] \leq \frac{\sigma}{t^2} \tag{2.2.4}$$

where σ is the variance of X. The proof of this inequality follows from applying Markov's inequality on the random variable $Y := (X - \mathbb{E}[X])^2$. Observe that this is a two-sided inequality – not only it bounds the probability that X goes much above its mean, but also the probability of X going much below its mean.

Example 2.9 *We now apply this inequality to our running example. We get*

$$\Pr[X \geq 3n/4] \leq \Pr[|X - \mathbb{E}[X] \geq n/4|] \leq \frac{n/4}{9n^2/16} = \frac{4}{9n}.$$

Thus this probability goes to 0 as n goes to infinity.

We see in the example above that Chebychev inequality gives a much stronger bound than Markov's inequality. In fact, it is possible to get much stronger bounds. Chebychev just uses bounds on the second moment of X. With knowledge of higher moments, we can give tighter bounds on probability that X deviates from its mean by a large amount. If $X = \sum_i^n X_i$ is the sum of n mutually independent random variables where each X_i is Bernoulli random variable (i.e., takes values 0 or 1 only), then

Chernoff bounds gives

$$\Pr[X \geq (1+\delta)\mu] \leq \frac{e^{\delta\mu}}{(1+\delta)^{(1+\delta)\mu}}, \tag{2.2.5}$$

where δ is any positive parameter and μ denotes $\mathbb{E}[X]$. The analogous bound for deviations below the mean is as follows:

$$\Pr[X \leq (1-\delta)\mu] \leq \frac{e^{\delta\mu}}{(1+\delta)^{(1+\delta)\mu}}, \tag{2.2.6}$$

where δ lies between 0 and 1.

Before we get into the proof of these bounds, we state more usable versions which often suffice in practice. It is easy to check that for any $\delta > 0$, $\ln(1+\delta) > \frac{2\delta}{2+\delta}$. Therefore

$$\delta - (1+\delta)\ln(1+\delta) \leq -\frac{\delta^2}{2+\delta}.$$

Taking exponents on both sides, we see that

$$\frac{e^{\delta\mu}}{(1+\delta)^{(1+\delta)\mu}} \leq e^{-\frac{\delta^2\mu}{2+\delta}}.$$

Thus we get the following:

- For $0 \leq \delta \leq 1$,
$$\Pr[X \geq (1+\delta)\mu] \leq e^{-\delta^2\mu/3}, \tag{2.2.7}$$
and
$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/3} \tag{2.2.8}$$
- For $\delta > 2$,
$$\Pr[X \geq (1+\delta)\mu] \leq e^{-\delta\mu/2} \tag{2.2.9}$$
$$Prob(X \geq m) \leq \left(\frac{np}{m}\right)^m e^{m-np} \tag{2.2.10}$$

We now give a proof of the Chernoff bound (2.2.5). The proof for the case (2.2.6) is analogous.

$$\Pr[X \geq (1+\delta)\mu] = \Pr[e^{\lambda X} \geq e^{\lambda(1+\delta)\mu}] \leq \frac{\mathbb{E}[e^{\lambda X}]}{e^{\lambda(1+\delta)\mu}},$$

where λ is a positive parameter that we shall fix later, and the last inequality follows from Markov's inequality. Notice that $\mathbb{E}[e^{\lambda X}] = \mathbb{E}[\prod_{i=1}^n e^{\lambda X_i}] = \prod_{i=1}^n \mathbb{E}[e^{\lambda X_i}]$ because $X_1, \ldots, X_n$ are mutually independent. Let p_i denote the probability with which X_i takes the value 1. Then $\mathbb{E}[e^{\lambda X_i}] = (1-p_i) + p_i \cdot e^{\lambda} = 1 + p_i(e^{\lambda} - 1) \leq e^{p_i(e^{\lambda}-1)}$, because $1 + x \leq e^x$ for any positive x. Since $\mu = \sum_{i=1}^n p_i$, we get we get

$$\Pr[X \geq (1+\delta)\mu] \leq \frac{e^{\mu(e^{\lambda}-1)}}{e^{\lambda(1+\delta)\mu}}.$$

Now we choose $\lambda > 0$ to minimise the right hand side, i.e., to minimise $e^{\lambda} - \lambda(1+\delta)$. It is easy to check that this is minimised at $\lambda = \ln(1+\delta)$. Substituting this value of λ in the RHS of the inequality above gives us the Chernoff bound (2.2.5).

Example 2.10 *We now apply Chernoff bound to our running example. Here* $\mu = n/2$. *Using* $\delta = 1/2$ *in (2.2.7) to get*

$$\Pr[X \geq 3n/4] \leq e^{-n/12}.$$

Note that for large values of n *this is much sharper bound than the one obtained using Chebychev's inequality.*

Example 2.11 *(Balls in Bins) Suppose we throw* n *balls into* n *bins, where each ball is thrown independently and uniformly at random into one of the bins. Let* Y_i *denote the number of balls which fall in bin* i. *We are interested in the random variable* $Y := \max_{i=1}^{n} Y_i$, *i.e., the maximum number of balls which fall in a bin. We will use Chernoff bound to show that* Y *is* $O(\ln n)$ *with high probability. Let us first consider a fixed bin* i *and show that* Y_i *is* $O(\ln n)$ *with high probability. For a ball* j, *let* X_j *be the indicator random variable which is 1 if ball* j *falls in bin* i, *0 otherwise. Clearly,* $\Pr[X_j = 1]$ *is* $1/n$. *Now,* $Y_i = \sum_{j=1}^{n} X_j$, *and so,* $\mathbb{E}[Y_i] = 1$. *Since* $X_1, \ldots, X_n$ *are independent Bernoulli random variables, we can apply (2.2.9) with* $\delta = 4 \ln n$ *to get*

$$\Pr[Y_i \geq 4 \ln n + 1] \leq e^{-2 \ln n} = 1/n^2.$$

Now we use union bound to get

$$\Pr[Y \geq 4 \ln n + 1] \leq \sum_{i=1}^{n} \Pr[Y_i \geq 4 \ln n + 1] \leq 1/n.$$

Thus, with probability at least $1 - 1/n$, *no bin gets more than* $4 \ln n + 1$ *balls.*

It turns out that one can get a sharper bound if we use (2.2.5) directly. It is left as an exercise to show that Y *is* $O(\ln n / \ln \ln n)$ *with high probability.*

Example 2.12 *Suppose we toss a fair coin* n *times independently. What is the absolute value of the difference between the number of Heads and the number of Tails ? Using Chernoff bounds, one can show that this random variable is very likely to be* $O(\sqrt{n})$. *To see this, let* X_i *be the indicator random variable which is 1 if the outcome of the* i^{th} *coin toss is Heads, 0 otherwise. Then the random variable* $X = \sum_{i=1}^{n} X_i$ *counts the number of Heads which are seen during this experiment. Clearly,* $\mu := \mathbb{E}[X] = n/2$. *Using* $\delta = 3/\sqrt{n}$ *in (2.2.7) and in (2.2.8), we see that* $\Pr[|X - n/2| \geq \sqrt{n}]$ *is at most* e^{-3}, *which is about 0.05.*

2.3 Generating Random numbers

The performance of any randomized algorithm is closely dependent on the underlying random number generator (RNG) in terms of efficiency. A common underlying

assumption is availability of a RNG that generates a number uniformly in some range $[0, 1]$ in unit time or alternately $\log N$ independent random *bits* in the discrete case for the interval $[0 \ldots N]$. This primitive is available in all standard programming languages - we will refer to this RNG as $\mathcal{U}$. We will need to adapt this to various scenarios that we describe below.

2.3.1 Generating a random variate for an arbitrary distribution

We consider a discrete distribution $\mathcal{D}$, which is specified by distribution function $f(s)$, $s = 1, \ldots, N$. We would like to generate a random variate according to $\mathcal{D}$. The distribution $\mathcal{D}$ can be thought of generating a random variable X with weight $w_i = f(i)$ where $\sum_i w_i = 1$. A natural way to sample from such a distribution is as follows. We can divide the interval $[0, 1]$ into consecutive subintervals $I_1, I_2 \ldots$ such that I_j has length w_j. Now, using the RNG $\mathcal{U}$, we sample a random point in the interval $[0, 1]$. If it falls in the interval I_j, we output j. It is easy to see that the probability that this random variable takes value j is exactly $f(j)$.

As stated, the above process can take $O(N)$ time because we need to figure out the interval in which the randomly chosen point lies. We can make this more efficient by using binary search. More formally, let $F(j)$ denote $\sum_{i=1}^{j} f(i)$ – it is also called the *cumulative distribution function* (CDF) of $\mathcal{D}$. Clearly, the sequence $F(1), F(2), \ldots, F(N) = 1$ forms a monotonically non-decreasing sequence. Given a number x in the range $[0, 1]$, we can use binary search to find the index j such that x lies between $F(j)$ and $F(j + 1)$. Therefore, we can sample from this distribution in $O(\log N)$ time.

This idea of dividing the unit interval into discrete segments does not work for a continuous distribution (for example, the normal distribution). However, we can still use a simple extension of the previous idea. A continuous distribution is specified by a CDF $F()$, where $F(s)$ is supposed to indicate the probability of taking a value less than or equal to s. We assume that $F()$ is continuous (note that $F(-\infty) = 0$ and $F(+\infty) = 1$). In order to sample from this distribution, we again a sample a value x uniformly from $[0, 1]$ using $\mathcal{U}$. Let s be a value such that $F(s) = x$ (we are assuming we can compute F^{-1}, in the discrete case, we were using a binary search procedure instead). We output the value s. It is again easy to check that this random variable has distribution given by $\mathcal{D}$.

2.3.2 Generating random variables from a sequential file

Suppose a file contains N records from which we would like to sample a subset of n records uniformly at random. There are several approaches to this basic problem:

- *Sampling with replacement* We can use $\mathcal{U}$ to repeatedly sample an element from the file. This could lead to *duplicates*.

- *Sampling without replacement* We can use the previous method to choose the next sample but we will reject duplicates. The result is an uniform sample but the efficiency may suffer. In particular, the expected number of times we need to invoke the RNG for the k-th sample is $\frac{N}{N-k}$ (see exercises).

- *Sampling in a sequential order* Here we want to pick the samples $S_1, S_2 \ldots S_n$ in an increasing order from the file, i.e., $S_i \in [1 \ldots N]$ and $S_i < S_{i+1}$. This has applications to processes where can scan the records exactly once and we cannot retrace.

  Suppose we have selected $S_1, \ldots, S_m$ so far, and scanned the first t elements. Conditioned on these events, we select the next element (as S_{m+1}) with probability $\frac{n-m}{N-t}$. Again we implement this process by choosing a random value x in the range $[0, 1]$ using $\mathcal{U}$ and then checking if x happens to be more or less than $\frac{n-m}{N-t}$.

  In order to show that this random sampling procedure is correct, let us calculate the probability that this process selects elements $s_1, \ldots, s_n$, where $1 \leq s_1 \leq s_2 \leq \ldots \leq s_n \leq N$. Let us condition on the fact that $S_1 = s_1, \ldots, S_m = s_m$. What is the probability that $S_{m+1} = s_{m+1}$. For this to happen we must not select any of the elements in $s_m + 1, \ldots, s_{m+1} - 1$, and then select s_{m+1}. The probability of such an event is exactly

$$\frac{n-m}{N-s_{m+1}} \cdot \prod_{t=s_m+1}^{s_{m+1}-1} \left(1 - \frac{n-m}{N-t}\right).$$

  Taking the product of the above expression for $m = 1, \ldots, n$, we see that the probability of selecting $s_1, \ldots, s_n$ is exactly $\frac{1}{\binom{N}{n}}$.

  Although the above procedure works, it calls $\mathcal{U}$ N times. Here is a more efficient process which calls $\mathcal{U}$ fewer number of times. It is easy to check that the distribution of $S_{i+1} - S_i$ is given by (see exercises)

$$F(s) = 1 - \frac{\binom{(N-t-s)}{(n-m)}}{\binom{(N-t)}{(n-m)}} \; s \in [t+1, N]. \tag{2.3.11}$$

  Thus we can sample random variables from the distribution $S_1, S_2 - S_1, \ldots, S_n - S_{n-1}$, and then select the corresponding elements.

- *Sampling in a sequential order from an arbitrarily large file*: This case is same as above except that we do not know the value of N. This is the typical scenario in a streaming algorithm (see Chapter 16).

  In this case, we always maintain the following invariant:

  Among the i records that we have scanned so far, we have a sample of n elements chosen uniformly at random from these i elements.

  Note that the invaraint makes sense only when $i \geq n$ because the n samples are required to be distinct. Further, when $i = n$, the first n records must be chosen in the sample. Now assume that this invariant holds for some $i \geq n$. Let $S_{n,i}$ denote the random sample of n elements at this point of time. When we scan the next record (which may not happen if the file has ended), we want to restore this invariant for the $i+1$ records. Clearly the $i+1$-th record needs to be in the sample with some probability, say p_{i+1} and if picked, one of the previous sampled records must be replaced.

  Note that $p_{i+1} = \frac{n}{i+1}$. This follows from the fact that there are $\binom{i+1}{n}$ ways of selecting n samples from the first $i+1$ elements, and exactly $\binom{i}{n-1}$ of these contain $i+1$. Therefore,

  $$p_{i+1} = \frac{\binom{i}{n-1}}{\binom{i+1}{n}} = \frac{n}{i+1}.$$

  If the (i+1)-th record is indeed chosen, we drop one of the previously chosen n samples with equal probability. To see this, notice that the invariant guarantees that the set $S_{n,i}$ is a uniformly chosen sample of n elements. We claim that dropping one of the samples uniformly at random gives us $S_{n-1,i}$, i.e., a uniform $n-1$ sample. The probability that a specific subset of $n-1$ elements, say S^* is chosen is the probability that $S^* \cup \{x\}$ was chosen, ($x \notin S^*$) and x was dropped. You can verify that

  $$\frac{1}{n} \cdot (i - n + 1) \cdot \frac{1}{\binom{i}{n}} = \frac{1}{\binom{i}{n-1}}$$

  where the term $(i - n + 1)$ represents the number of choices of x. The RHS is the uniform probability of an $n-1$ sample. Thus the sampling algorithm is as follows: when we consider record $i+1$, we select it in the sample with probability $\frac{n}{i+1}$ – if it gets selected, we drop one of the earlier chosen samples with uniform probability.

Procedure Random Permutation($\{x_1, x_2 \dots x_n\}$)

Input : Objects $\{x_1, x_2 \dots x_n\}$;
Output: A random permutation $\Pi = \{x_{\sigma(1)}, x_{\sigma(2)} \dots x_{\sigma(n)}\}$;
Initialize an array of size $m(> n)$ as *unmarked* ;
for *i = 1 to n* **do**
    while $A[j]$ *is marked* **do**
        Generate a random number $j \in_{\mathcal{U}} [1, m]$;
    $A[j] \leftarrow i$;
    mark $A[j]$;
Compress the marked locations in $A[1, n]$ and Return A where $\sigma(A[j]) = j$;

Figure 2.1: Generating a random permutation of n distinct objects

2.3.3 Generating a random permutation

Many randomized algorithms rely on the properties of random permutation to yield good expected bounds. Some algorithms like Hoare's quicksort or randomized incremental construction actually start from the assumption on an initial random order. However, the input may not have this property, in which case the onus is on the algorithm to generate a random permutation. Broadly speaking, any such algorithm must have access to random numbers and also ensure that all the permuations of the input objects are equally likely outcomes.

We describe the algorithm in Figure 2.1. The algorithm runs in n iterations, in the i^{th} iteration, it assigns x_i to a random location in the permutation. It places the ordered elements (according to the random permutation) in an array A. Note that the size of A is slightly larger than n, and so, some positions in A will remain empty at the end. Still, we can read the permutation from A by scanning it from left to right.

In the array A, the algorithm marks the locations which are occupied. The main loop tries to assign x_i to a random location among the unmarked (unoccupied) locations in the array A. For this, it keeps trying until it finds a free position. We need to prove the following
(i) After termination, all permutations are equally likely.
(ii) The expected number of executions of the loop is not too large - preferably linear in n.
(iii) Returning the n elements in contiguous locations takes m steps.

To balance (ii) and (iii), we have to choose m somewhat carefully. We make some simple observations

Claim 2.1 *If the number of unmarked locations in A is t, then each of the t locations is chosen with equal likelihood.*

This follows from a simple application of conditional probability, conditioned on a location being unmarked. Consider any fixed set N of distinct n locations. Conditioned on assigning the elements $x_1, x_2 \ldots x_n$ to N, all permutations of $x_1, x_2 \ldots x_n$ are equally likely. Again this follows from the observation that, after $x_1.x_2 \ldots x_i$ are assigned, x_{i+1} is uniformly distributed among the unoccupied $n - i$ locations. Since this holds for any choice of N, the unconditional distribution of the permutations is also the same.

The number of iterations depend on the number of unsucessful attempts to find an unassigned location. The probability of finding an unassigned location after i assignments is $\frac{m-i}{m} = 1 - \frac{i}{m}$. Since the locations are chosen independently, the expected number of iterations to find a free location for the x_{i+1} is $\frac{m}{m-i}$ and from the linearity of expectation, the total expected number of iterations is

$$\sum_{i=0}^{n-1} \frac{m}{m-i} = m\left(\frac{1}{m} + \frac{1}{m-1} \ldots \frac{1}{m-n+1}\right) \tag{2.3.12}$$

So, $m = n$, this is $O(n \log n)$ whereas for $m = 2n$, this becomes $O(n)$. Since the probabilities are independent, we can obtain concentration bounds for deviation from the expected bounds using Chernoff-Hoefding bounds as follows.

What is the probability that the number of iterations exceed $3n$ for $m = 2n$? This is equivalent to finding fewer than n assignments in $3n$ iterations. Let $p_i = \frac{2n-i}{2n}$, then for $i \leq n$, $p_i \geq 1/2$ where p_i is the probability of finding a free location for x_i. Let us define 0-1 random variables X_i such that $X_i = 1$ if the i-th iteration is successful, i.e., we find an unmarked location. To terminate, we need n unmarked locations. From our previous observation, $\Pr[X_i = 1] \geq 1/2$. So $\mathbb{E}[\sum_{i=1}^{3n} X_i] \geq 3n/2$. Let $X = \sum_i X_i$ be the number of successes in $3n/2$ iterations. Then X is a sum of independent Bernoulli random variables and a straightforward application of Chernoff bounds (Equation 2.2.8 shows that

$$\Pr[X < n] = \Pr[X < (1 - 1/3)\mathbb{E}[X]] \leq \exp(-\frac{3n}{36})$$

which is inverse exponential.

Claim 2.2 *A random permutation of n distinct objects can be generated in $O(n)$ time and $O(n)$ space with high probability.*

The reader would have noted that as m grows larger, the probability of encountering a marked location decreases. So, it is worth estimating for what value of m, there

will be exactly n iterations with high probability, i.e., no reassignment will be necessary. This could be useful in online applications where we need to generate random permutations. Using Equation 2.2.10, we can bound the probability that the number random asignments in a location exceeds 1 as

$$\left(\frac{n}{2m}\right)^2 e^{2-n/m} \leq O(n^2/m^2)$$

Note that the expected number of assignments in a fixed location $\mu = \frac{n}{m}$. From union bound, the probability that any of the m locations has more than 1 assignment is bound by $O(\frac{n^2}{m})$. So, by choosing $m = \Omega(n^2)$, with probability $1 - O(\frac{n^2}{m})$ the number of iterations is n, i.e., there is no reassignment required.

Further Reading

There are several excellent textbooks on introductory probability theory and randomized algorithms []. Most of the topics covered in this chapter are classical, and are covered in these texts in more detail. Chernoff bounds are among the most powerful tail inequalities when we are dealing with independent random variables. There are similar bounds which sometimes give better results depending on the parameters involved, e.g., Hoeffding's bound. Maintaining a random sample during a streaming algorithm is a common subroutine used in many streaming algorithms (see e.g., Chapter 16). The idea that picking n elements out of an array of size $2n$ or more results in small repetitions is often used in many other applications, for example hashing (see Chapter!**??**).

Exercises

Exercise 2.1 *Consider the experiment of tossing a fair coin till two heads or two tails appear in succession.*
(i) Describe the sample space.
(ii) What is the probability that the experiment ends with an even number of tosses ?
(iii) What is the expected number of tosses ?

Exercise 2.2 *In a temple, thirty persons give their shoes to the caretaker who hands back the shoes at random. What is the expected number of persons who get back their own shoes.*

Exercise 2.3 *A chocolate company is offering a prize for anyone who can collect pictures of n different cricketers, where each wrap has one picture. Assuming that*

each chocolate can have any of the pictures with equal probability, what is the expected number of chocolates one must buy to get all the n different pictures ?

Exercise 2.4 *There are n letters which have corresponding n envelopes. If the letters are put blindly in the envelopes, show that the probability that none of the letters goes into the right envelope tends to $\frac{1}{e}$ as n tends to infinity.*

Exercise 2.5 *Imagine that you are lost in a new city where you come across a cross-road. Only one of them leads you to your destination in 1 hour. The others bring you back to the same point after 2,3 and 4 hours respectively. Assuming that you choose each of the roads with equal probability, what is the expected time to arrive at your destination ?*

Exercise 2.6 *A gambler uses the following strategy. The first time he bets Rs. 100 - if he wins, he quits. Otherwise. he bets Rs. 200 and quits regardless of the result. What is the probability that he goes back a winner assuming that he has probability 1/2 of winning each of the bets. What is the generalization of the above strategy ?*

Exercise 2.7 Gabbar Singh problem *Given that there are 3 consecutive blanks and three consecutive loaded chambers in a pistol, and you start firing the pistol from a random chamber, calculate the following probabilities. (i) The first shot is a blank (ii) The second shot is also a blank given that the first shot was a blank (iii) The third shot is a blank given that the first two were blanks.*

Exercise 2.8 *In the balls in bins example 2.11, show that the maximum number of balls in any bin is $O(\ln n / \ln \ln n)$ with high probability.*

Exercise 2.9 *Suppose we throw m balls independently and uniformly at random in n bibs. Show that if $m \geq n \ln n$, then the maximum number of balls received by any bin is $O(m/n)$ with high probability.*

Exercise 2.10 *Three prisoners are informed by the jailor that one of them will be acquited without divulging the identity. One of the prisoners requests the jailor to divulge the identity of one of the other prisoner who won't be acquited. The jailor reasons that since at least one of the remaining two will not be acquited, reveals the identity. However this makes this prisoner very happy. Can you explain this ?*

Exercise 2.11 *For random variables X, Y, show that*
(i) $\mathbb{E}[X \cdot Y] = \mathbb{E}[Y \times \mathbb{E}[X|Y]]$
(ii) $\mathbb{E}[\mathbb{E}[X|Y]] = \mathbb{E}[X]$
(iii) $\mathbb{E}[\phi_1(X_1) \cdot \phi_2(X_2)] = \mathbb{E}[\phi_1(X_1)] \cdot \mathbb{E}[\phi_2(X_2)]$ *for functions ϕ_1, ϕ_2 of random variables.*

Exercise 2.12 *Give an example to show that even if* $\mathbb{E}[X \cdot Y] = \mathbb{E}[X] \cdot \mathbb{E}[Y]$*, the random variables* X, Y *may not be independent.*
Hint: Consider X *and some appropriate function of* X*.*

Exercise 2.13 *Let* $Y = \sum_{i=1}^{n} X_i$ *where* X_i*s are identically distributed random variables with expectation* μ*. If* n *is a non-negative integral random variable, then* Y *is known as random sum. Show that* $\mathbb{E}[Y] = \mu \cdot \mathbb{E}[n]$*.*

Exercise 2.14 *Let* Y *be a random variable that denotes the number of times a fair dice must be rolled till we obtain a six. Assume that the outcomes are independent of each other. How many times do we have to roll the dice to obtain* k *successes ?*
Let X *be a random variable that denotes this, then*
(i) Compute $\mathbb{E}[X]$
(ii) Show that $\Pr[X \geq 10k] \leq \frac{1}{2^k}$ *using Chernoff bounds.*
The distribution of Y *is known as geometric and* X *is known as negative binomial.*

Exercise 2.15

For a discrete random variable X, e^{Xs} is known as the *moment generating function* and let $M(s) = \mathbb{E}[e^{sX}]$. Show that
$\mathbb{E}[X^k] = \frac{d^k M}{ds^k}|_{s=0}, k = 1, 2, \ldots$. This is a useful formulation for computing the k-th moment of a random variable.
Hint: Write down the series for e^{sX}.

Exercise 2.16 *Let* $G(n, p)$ *be a graph on* n *vertices where we add an edge between every pair of vertices independently with probability* p*. Let* X *denote the number of edges in the graph* $G(n, p)$*. What is the expectation of* X *? What is the variance of* X*?*

Exercise 2.17 *Let* $G(n, p)$ *be as above. A triangle in this graph is a set of three vertices* $\{u, v, w\}$ *(note that it is an unordered triplet) such that we have edges between all the three pairs of vertices. Let* X *denote the number of triangles in* $G(n, p)$*. What are the expectation and the variance of* X*?*

Exercise 2.18 *Consider the algorithm for sampling from a continuous distribution in Section 2.3.1. Prove that the random variable has the desired distribution.*

Exercise 2.19 *Consider the problem of uniformly sampling* n *distinct elements from a file containing* N *elements. Suppose we have already sampled a set* S k *elements. For the next element, we keep on selecting a uniform sample from the file till we get an element which is not in* S*. What is the expected number of times we need to sample from the file?*

Exercise 2.20 *Consider the problem of sampling in a sequential order. Prove that the distribution of* $S_i - S_{i-1}$ *is given by the expression in 2.3.11.*