Recitation 8 — Probability

Parallel and Sequential Data Structures and Algorithms, 15-210 (Fall 2012)

October 17, 2012

1 Announcements

- Assignment 4 is back.
- Assignment 6 is out. It's long, and you have two weekends to complete it. Start this weekend!
- Today's recitation is about probability.
- Questions about homework, class, life, universe?

1.1 Probability Basics

Many of you have seen probability before. We'll quickly go through the basics and move on to more interesting things.

1.1.1 Conditional Probability

$P(A|B) = P(A \cap B)/P(B)$. This identity can be intuitively demonstrated on a Venn diagram.

1.1.2 Independence

Say I throw a fair six-sided dice three times. The probability of rolling an even number, then a four, then an even number again is $\frac{1}{24}$. This is because rolling an even number on a six-sided dice occurs with $\frac{1}{2}$ probability, and rolling a four occurs with $\frac{1}{6}$ probability, and $(\frac{1}{2})^2 \cdot \frac{1}{6} = \frac{1}{24}$. Now, say I draw three cards from a shuffled 52-card deck. What is the probability of drawing any card in the hearts suite, then the queen of hearts, then any hearts card again? This question is harder because the events here are not independent.

Formally, event A is independent from event B iff $P(A|B) = P(A)$; i.e. the probability of A remains the same whether or not B has occurred. From the identity $P(A|B) = P(A \cap B)/P(B)$, it follows that

$$P(A|B) = P(A) \iff P(A \cap B)/P(B) = P(A) \iff P(A \cap B)/P(A) = P(B) \iff P(B|A) = P(B)$$

so we see that event A being independent from event B also implies that B is independent from A.

Using the identity $P(A|B) = P(A \cap B)/P(B)$, the probability that events A, B, and C occur, i.e. $P(A \cap B \cap C)$, can be expressed as $P(A) \cdot P(B|A) \cdot P(C|A \cap B)$. If A, B, and C are mutually independent, then this is just $P(A) \cdot P(B) \cdot P(C)$, which is what made the first dice example so easy. (Note that mutual independence is a stronger condition than pairwise independence.) The second example with cards is hard because we're stuck with trying to figure out the probability of drawing the queen of hearts given that we've drawn some hearts card already, and then the probability of drawing another hearts given that we've drawn some hearts card and then the queen of hearts.

Keep in mind that events can be dependent even when they occur at the same time with the same probability. For example, consider a shell game, where a pea is hidden underneath one of three shells (with uniform probability). While each shell has a $\frac{1}{3}$ probability of containing the pea, the event that shell A contains the pea is not independent from the event that shell B contains the pea. At most one can contain the pea; there isn't a $\frac{1}{27}$ probability that all three shells contain the pea! In this example, the events of A and B containing the pea are mutually exclusive—$P(\text{A contains the pea}|\text{B contains the pea}) = 0$, or equivalently $P(A \cap B) = 0$.

1.1.3 Inclusion-Exclusion

$P(A \cup B) = P(A) + P(B) - P(A \cap B)$. This principle can be intuitively demonstrated on a Venn diagram. Note that $P(A \cap B) = 0$ if and only if A and B are mutually exclusive, so be careful when directly adding probabilities.

1.1.4 Expected value

A random variable (usually a 1 dimensional number in the reals) is a variable representing the outcome of a random process. While ordinary variables are considered to have a single unknown value, random variables are considered to have all possible values, each with a particular probability. This isn't quite true, but it's good enough for 210, where we only deal with discrete probability.

The expected value of a random variable is the weighted mean of its possible values, where each value x is weighted by the probability that x is the outcome. For example, a random variable representing the outcomes of a fair six-sided dice has an expected value 3.5, which is the average of 1, 2, 3, 4, 5, and 6 when each is given equal weight. What would the expected value be for a dice that is unfair, such that even numbers are rolled with probability $\frac{2}{9}$ and odd numbers are rolled with probability $\frac{1}{9}$? You could do it the long way, "$1 \cdot \frac{1}{9} + 2 \cdot \frac{2}{9} + 3 \cdot \frac{1}{9} + \ldots$"; or you could just take the mean of $\langle 1, 3, 5, 2, 2, 4, 4, 6, 6 \rangle$.

The expected value function $\mathbf{E}$ (also known as "expectation") is a linear function, meaning that for any value c, $\mathbf{E}[c \cdot X] = c \cdot \mathbf{E}[X]$ and $\mathbf{E}[X] + \mathbf{E}[Y] = \mathbf{E}[X + Y]$. Furthermore, $\mathbf{E}[X \cdot Y] = \mathbf{E}[X]\mathbf{E}[Y]$ holds when X and Y are independent. For example, in the random process where we have n red dice and m blue dice, and will roll them and multiply the sum of the red values with the sum of the blue values, the expected value is simply $(3.5 \cdot n) \cdot (3.5 \cdot m) = 12.25 \cdot nm$. For more detail about the linearity of expectation, see the lecture slides.

1.2 Max Cut

Whenever the vertices of a graph G are partitioned into two sets A and B, zero or more edges of G run between a vertex in A and a vertex in B. The MAX-CUT problem is determining, for a given graph G, the maximum possible number of edges between any such A and B. As it turns out, MAX-CUT is NP-complete, so there's probably not a fast algorithm for solving it.

1.2.1 Decision Problems

Sometimes, we state MAX-CUT as: Given a graph G and an integer k, is there a cut of size at least k? This is a *decision problem*—the output is "yes" or "no". The first statement of the MAX-CUT problem is an *optimization problem*—the output is an integer.

These two problems are "almost the same". If you know the maximum cut value, you can just compare it to any k, so the optimization problem is at least as hard as the decision problem. Furthermore, if you can answer whether there is a cut of at at least k, then you can just binary search for the maximum k for which this is true. Thus, the optimization problem is at most $O(\log n)$ times harder than the decision problem. Factors of $\log n$ aren't that bad in a polynomial time algorithm, much less an exponential one.

This kind of reduction is quite general; it applies to any problem where it makes sense to ask "is the answer at least k?". It's also often useful, as the decision problem can be easier to think about than the optimization problem.

1.2.2 A Random Algorithm

While we can't solve MAX-CUT quickly, it's quite easy to come within an expected factor of two of the right answer.

Algorithm: randomly assign each vertex to A or B. We define an indicator variable X_e for each edge e to be 1 if e goes between A and B and 0 otherwise. We see that $P(X_e = 1) = 0.5$—fix one endpoint; the other is in the other side with probability 0.5. By linearity of expectation, we expect half the edges to be cut.

Without actually running this random algorithm, we can already tell that there exists some cut of at least half the edges: in any non-empty set of numbers, one of them is at least the mean (or weighted mean). More formally:

Theorem 1.1. $P(X \geq \mathbf{E}[X]) > 0$

Proof. Assume the contrary. Then,

$$\mathbf{E}[X] \;=\; \sum_{i=1}^{n} x_i P(X = x_i) \quad < \quad \sum_{i=1}^{n} \mathbf{E}[X] P(X = x_i) \;=\; \mathbf{E}[X] \sum P(X = x_i) \;=\; \mathbf{E}[X]$$

which is a contradiction. □

It's important to note that this does not guarantee that the random algorithm will actually return a good cut; just that it must be possible. With what probability will the random algorithm return a good cut?

Let m be the number of edges in the graph.
Let X be the random variable for the number of edges cut by the random algorithm.
Let $a \in [0, 0.5]$
Let $p = P(X \leq m(1/2 - a))$.

$$\begin{aligned}
&\mathbf{E}[X] \leq pm(1/2 - a) + (1 - p)m \\
&\Rightarrow m/2 \leq pm(1/2 - a) + (1 - p)m \\
&\Rightarrow 1/2 \leq p/2 - pa + 1 - p \\
&\Rightarrow -1/2 \leq -p/2 - pa \\
&\Rightarrow 1 \geq p + 2pa \\
&\Rightarrow p \leq \frac{1}{1 + 2a}
\end{aligned}$$

So $P(X > m(1/2 - a)) > 1 - \frac{1}{1+2a}$.

1.2.3 A Deterministic Algorithm

There's also a good deterministic algorithm for MAX-CUT that cuts at least half the edges: iterate through the vertices in any arbitrary order, placing them one-by-one into either group A or group B. Place the first vertex in group A, and the second vertex in group B. For each subsequent vertex v, place v in A iff v has more edges to vertices already in B than it does to vertices already in A. (It actually doesn't matter where we place v if it has an equal number of edges to each group; the important thing is, whenever possible, v is placed in the group it has less edges to).

This is guaranteed to cut at least half the edges; to prove this, consider the decision we make for each edge. For every edge e, its second vertex is added to A or B exactly once. At this point, e is either certainly cut in the final partition, or it is certainly uncut. Placing each vertex makes this decision for potentially multiple edges, and each vertex is placed such that at least as many edges are certainly cut as certainly uncut. Therefore, at least as many edges are cut as uncut not only in in the final partition, but also throughout the entire placement process.

1.3 Weak Law of Large Numbers

Theorem 1.2. *(Markov's Inequality)* $P(|X| \geq a) \leq \dfrac{\mathbf{E}[|X|]}{a}$

Proof. Let Y be a random indicator variable that is 1 when $|X| \geq a$ and 0 otherwise.
Note that $\mathbf{E}[Y] = P(|X| \geq a)$.

Trivially, $aY \leq |X|$, because when $|X| < a$ then $Y = 0$ and the LHS is zero; and when $|X| \geq a$, then $Y = 1$ and the LHS is a.

Thus, $\mathbf{E}[aY] \leq \mathbf{E}[|X|] \implies a\mathbf{E}[Y] \leq \mathbf{E}[|X|] \implies P(|X| \geq a) \leq \dfrac{\mathbf{E}[|X|]}{a}$ □

The quantity $\mathbf{E}[(X - \mathbf{E}[X])^2]$ is called the *variance* of X, written $\text{Var}(X)$.

Theorem 1.3. *(Chebyshev's Inequality)* $P(|X - \mathbf{E}[X]| \geq d) \leq \dfrac{\text{Var}(X)}{d^2}$

Proof. Apply Markov's inequality to $P((X - \mathbf{E}[X])^2 \geq d^2)$, resulting in $P((X - \mathbf{E}[X])^2 \geq d^2) \leq \dfrac{\text{Var}(X)}{d^2}$
$(X - \mathbf{E}[X])^2 \geq d^2$ if and only if $|X - \mathbf{E}[X]| \geq d$, so $P((X - \mathbf{E}[X])^2 \geq d^2) = P(|X - \mathbf{E}[X]| \geq d)$
Therefore, Chebyshev's Inequality holds. □

Note that $\text{Var}(aX) = \mathbf{E}[(aX - \mathbf{E}[ax])^2] = \mathbf{E}[(aX - a\mathbf{E}[X])^2] = a^2\mathbf{E}[(X - \mathbf{E}[X])^2] = a^2\text{Var}(X)$.

We will use without proof that if X and Y are independent, then $\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y)$ (the proof is similar to the proof of linearity of expectation, but messier).

Suppose you're testing the speed of your 210 homework code, and want to measure how long in seconds it takes to run. Easy, that's what the UNIX `time` utility is for. But `time` gives different results on each trial; what you're actually interested in is the average time. So, you run several trials and average them together. Why does running more trials give you a "better" average?

Theorem 1.4. *(Weak Law of Large Numbers) Let $X_1, X_2, \ldots$ be an infinite sequence of independent and identically distributed randomly variables. Defines $\overline{X}_n = \frac{1}{n}\sum_{i=1}^n X_i$ (the sample mean of the first n observations). Then $\forall \varepsilon > 0 : \lim_{n\to\infty} P(\overline{X}_n - \mathbf{E}[X] > \varepsilon) = 0$.*

Proof. $\text{Var}(\overline{X}_n) = \dfrac{1}{n^2}\text{Var}(\displaystyle\sum_{i=1}^n X_i) = \dfrac{1}{n^2} n\text{Var}(X_i) = \dfrac{\text{Var}(X_i)}{n}$
So $\lim_{n\to\infty} \text{Var}(\overline{X}_n) = 0$
Apply Chebyshev's Inequality to show that $\lim_{n\to\infty} P(|\overline{X}_n - \mathbf{E}[X]| > \varepsilon) = 0$. □

When sampling systems in real life, it's impossible to take infinitely many measurements. If only n measurements are taken, about how close is their average to the real average? This is a somewhat complicated question that requires more detailed information about the distribution, like the variance. How can you estimate the variance? From your measurements! How good is *that* estimate? *Confidence intervals* answer this kind of question in science and statistics; take a stats class to learn more!

1.4 Optional Exercise: Simulating a Fair Coin

Suppose you have a biased coin which lands heads with an unknown p.

Q: In a sequence of n flips, what is the expected number of times "HH" appears? ("HHH" counts as two appearances)
A: $p^2(n-1)$ times. The first $n-1$ flips independently have a $p \cdot p$ probability of being a heads and being followed by a heads.

Q: How can we generate a fair sequence of random bits using this coin?
A: Flip the coin twice. If you see "HH" or "TT", try again. If you see "HT", output 1. If you see "TH", output 0. This works because each of "HT" and "TH" occur with probability "p(1-p)".

Q: How many flips are lost, on average?
A: We loose $p^2 + (1-p)^2$ of the flips. Can you do better? How much better?

Q: How about the other way: given a fair coin, can you return bits biased with probability p?
A: Only if $p = x/2^k$ for some k. This is because after k flips, all sequences have probability $\frac{1}{2^k}$, so any combination of these probabilities will have denominator $\frac{1}{2^k}$. Of course, every real number in $(0,1)$ can be approximated arbitrarily closely as $x/2^k$. The binary number system wouldn't work without this!