Advanced Algorithms Course. Lecture Notes. Part 7

Randomized Algorithms

Basics of Probability Theory

This section is not a full-fledged introduction to probability theory, but only a repetition of the absolute minimum of knowledge needed for a real understanding of randomized algorithms and their analysis.

The mathematical essence of the notion of probability can be described by **Kolmogorov's axioms**, without recurring to the interpretation of proabilities. According to this approach, we first specify a probability space which "lives" on a set Ω. For simplicity we will focus on discrete sets Ω, which is the most relevant case in algorithmic contexts. Subsets of Ω are called **events**. The probability $Pr(A)$ of an event A is a number from the interval $[0, 1]$, and probabilities have to satisfy the following simple properties (and these are Kolmogorov's axioms): $Pr(\emptyset) = 0$; $Pr(\Omega) = 1$; if $A \cap B = \emptyset$ then $Pr(A \cup B) = Pr(A) + Pr(B)$ (additivity).

For a single-element event $A = \{\omega\}$ we may simply write $Pr(\omega)$ instead of $Pr(A) = Pr(\{\omega\})$. A set Ω together with a probability function Pr forms a **probability space**. (In the case of infinite Ω, the function Pr is only defined on a certain family of events, and the additivity axiom is formulated for countably infinite sums.)

From the axioms it follows immediately that $Pr(\Omega \setminus A) = 1 - Pr(A)$, and $Pr(A \cup B) \leq Pr(A) + Pr(B)$ for any events A and B. The latter inequality is so useful that it deserves a name: it is called the **union bound**. One can use it to bound the probablity of a complicated event which is, however, the disjunction of simpler events with easily computable probabilities.

Sometimes we know already that some event B occurs, and we want to know the probability of A under this additional knowledge. This **conditional probability** is given by $Pr(A|B) := Pr(A \cap B)/Pr(B)$. Pronounce

$Pr(A|B)$ as "probability of A given B" or "probability of A conditional on B". We call an event A **independent** of an event B if $Pr(A|B) = Pr(A)$. In that case we obviously get $Pr(A \cap B) = Pr(A)Pr(B)$, hence the independence relation is symmetric, and we can simply say "A and B are independent". It is not always intuitive whether two events are independent; we have to check independence using the definition. Also, do not confuse independent and disjoint events ($A \cap B = \emptyset$) – these are totally different things!

A **random variable** is a function X from a probability space into, e.g., the real numbers. (We only consider the case of real-valued X and discrete Ω.) Formally: $X : \Omega \longrightarrow R$. Every possible value x of X gets a probability in an obvious way: $Pr(X = x) = Pr(X(\omega) = x)$. We may consider $Pr(X = x)$ as a function of x and call it the **distribution** of X. Note that two random variables with equal distributions are not necessarily equal as functions; this distinction is important when we combine several random variables by algebraic operations (see below).

The **expected value** or **expectation** of a random variable X is defined as $E[X] := \sum_{\omega \in \Omega} Pr(\omega)X(\omega)$. Note that $E[X] = \sum_x Pr(X = x) \cdot x$, that is, the expectation depends only on the distribution of X. Intuitively, $E[X]$ is the long-term average of X when we observe the random variable many times independently.

A frequent misunderstanding is that $Pr(X > E[X]) = 1/2$, or similar. This is far from being true in general. For instance, let X be the random variable that describes a win in a lottery (where the stake is not considered in X). The expected win is some (small) positive amount, but the probability of winning anything is very small, certainly not $1/2$. A "probability-free" formulation of this insight is: The average of a set of values is in general distinct from the median!

Random variables X and Y on the same probability space are called **independent** if $Pr(X = x, Y = y) = Pr(X = x)Pr(Y = y)$ for all values x and y. In the same way as for random events we could instead define independence by the property that knowing the value of X has no impact on the distribution of Y, and then this "product rule" comes out.

Random variables, without loss of generality defined on the same probability space, can be combined by arbitrary algebraic operations: We simply apply the operation to their random values. For instance, the sum $X+Y$ of random variables X and Y is given by $(X+Y)(\omega) = X(\omega)+Y(\omega)$. Similarly we can define the product, and so on.

A useful and powerful property is the **linearity of expectation**. It says that E is a linear operator, that means, $E[X+Y] = E[X] + E[Y]$. Note that this holds for arbitrary random variables, not only for independent ones. The proof is a straighforward calculation:

$$E[X+Y] = \sum_{\omega\in\Omega} Pr(\omega)(X+Y)(\omega) = \sum_{\omega\in\Omega} Pr(\omega)(X(\omega)+Y(\omega))$$

$$= \sum_{\omega\in\Omega} Pr(\omega)X(\omega) + \sum_{\omega\in\Omega} Pr(\omega)Y(\omega) = E[X] + E[Y].$$

A similar property for the product does not hold in general. We have $E[XY] = E[X]E[Y]$ in special cases only. The most important sufficient condition is that X and Y are independent. Again, the proof is a straighforward calculation, but this time it is easier to work on the range of values rather than on Ω. Also note carefully in which step independence is used:

$$E[XY] = \sum_z Pr(XY=z)z = \sum_z \sum_{x,y:\, xy=z} Pr(X=x, Y=y)xy$$

$$= \sum_z \sum_{x,y:\, xy=z} Pr(X=x)xPr(Y=y)y = \sum_{x,y} Pr(X=x)xPr(Y=y)y$$

$$= \sum_x Pr(X=x)x + \sum_y Pr(Y=y)y = E[X]E[Y].$$

An important "algorithm" is to repeat a random experiment until success: Suppose that we have a $0,1$-valued random variable that attains value 1 with probability p. We observe this variable many times independently, until result 1 appears for the first time. What is the expected number of iterations needed? Intuitively one would think $1/p$, but intuition is often misleading, therefore we'd better derive this result by calculation. Although this is still a basic exercise, a strict formal treatment would already be a bit tricky: Our probability space is the Cartesian product of infinitely many copies of a probability space with two events. However we may abbreviate somewhat and think in a semi-formal way. Let E_i be the event that the ith iteration is successful. Then $Pr(E_i) = (1-p)^{i-1}p$. Note that the first $i-1$ iterations have failed, and probabilities can be multiplied, because trials are independent. Hence our expected value is $\sum_{i=1}^{\infty} Pr(E_i)\cdot i = \sum_{i=1}^{\infty}(1-p)^{i-1}pi$. Now some standard algebra (that we omit here) confirms the result $1/p$.

Global Minimum Cut Revisited

In a graph $G = (V, E)$ with n nodes and m edges we wish to find a global min-cut (A, B), that is, a partitioning $V = A \cup B$ such that the number of cut edges (those edges between A and B) is minimized. Motivations include the assessment of reliability of networks, finding clusters in graphs, and efficient hierarchical computation of distances in graphs.

We can easily reduce the problem to Minimum Cut, by trying all possible pairs of sources and sinks $s, t \in V$. But since flow and cut algorithms are somewhat sophisticated, you may be pleased to learn an extremely simple randomized algorithm that solves the Global Min-Cut problem as well. However this comes with a price: Success is no longer guaranteed. We will get a correct solution "only" with high probability.

For simplicity we discuss only the basic randomized algorithm for Global Min-Cut, although faster algorithms are known. In the following we have to allow graphs with parallel (multiple) edges. The algorithm works as follows. In every step, choose an edge $e = (u, v)$ at random and contract it. Contraction means: shrink e, identify u, v (merge them into a new vertex), and delete all edges that have been parallel to e (they would be loops at the new vertex). Iterate this step until two nodes remain. This two-node graph represents a cut, in the obvious sense. The whole procedure is repeated a certain number of times from scratch, and finally we output the smallest cut found in this way.

It may seem that this algorithm has nothing to do with the problem. It just repeatedly contracts random edges. However, the intuition is that a small cut has a chance not to be affected by these random contractions, thus being preserved in the end. Still, the analysis which has to confirm this intuition is not so obvious. It uses a clever combination of several elementary tools from probability theory.

Consider any global min-cut (A, B). Let F denote the set of its cut edges, and $k := |F|$. After j steps of the algorithm, clearly the contracted graph has $n - j$ nodes. Moreover, every node has degree at least k, since otherwise the node and its complement set would already form a global min-cut smaller than k, a contradiction. Hence at least $k(n - j)/2$ edges still exist after j steps. Therefore, the probability that unfortunately some of the k edges in F is contracted in the next step is at most $2/(n - j)$. That

means, our specific cut (A, B) is returned with probability at least

$$\prod_{j=0}^{n-3}(1-2/(n-j)) = \prod_{j=0}^{n-3}((n-j-2)/(n-j)) = 2/n(n-1)$$

after the contraction procedure. (However, think carefully: Why is it correct to multiply the probabilities, although the events are certainly not independent?) This is a small probability, but we repeat this $O(m)$-time contraction procedure sufficiently often: Each run fails with probability $1-2/n(n-1)$, but a simple calculation shows that some of $O(n^2)$ runs succeeds, subject to a small constant failure probability. We can make this failure probability arbitrarily small by increasing the hidden constant factor in $O(n^2)$. (Note the superficial similarity to approximation schemes.)