Lecture Notes, CSE 232, Fall 2014 Semester

Dr. Brett Olsen

Week 10 - Probability and Combinatorics

This week we're going to talk about two related topics. First, combinatorics - counting objects of a particular kind or size, like the number of ways in which a particular event could happen, or the number of ways to create a particular kind of string or binary tree, etc. Secondly, probability - how likely is it that a particular kind of event will happen out of all possible events.

Combinatorics Combinatorics is the study of countable discrete structures. There are three major fields in combinatorics: permutations (how many ways can I arrange an ordered array of n discrete objects? $n!$), combinations (how many ways can I select a subset of objects from an unordered set of objects? 2^n), and partitions (how many ways can I divide up an integer into smaller numbers that sum to it?).

But rather than go into great detail into the theory behind combinatorics, let's instead talk about a couple of sequences commonly used in combinatorics that you'll see repeatedly in a wide range of problems.

Fibonacci Sequence The Fibonacci sequences is well known, and comes up often in a variety of contexts. There are several different ways to define and calculate the sequence, and some of these tricks can be used for related sequences.

Fibonacci numbers are defined recursively as the sum of the previous two numbers in the sequence:

$$F_0 = 0 F_1 = 1 F_n = F_{n-2} + F_{n-1} \tag{1}$$

which gives the sequence 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89...

Implementing this recursive solution directly is very slow, as it runs in $O(\phi^n)$ exponential time, where ϕ is the golden mean $\frac{1+\sqrt{5}}{2} = 1.618\ldots$. This time complexity is related to the approximation of ϕ as the ratio of F_n to F_{n-1}.

We can instead calculate the sequence in linear $O(n)$ time with a simple dynamic programming solution by simply iterating from F_0 to F_n, saving the last two values of the sequence to compute the next one.

There are two other interesting, faster, methods for calculating the numbers of a Fibonacci sequence:

First, by $O(log(n))$ solution using exponentiation of a matrix:

$$\begin{bmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^p \tag{2}$$

We can exponentiate the matrix in $log(n)$ time by repeated squaring of the matrix.

There is an even faster $O(1)$ method: F_n is the closest integer to the real number $(\phi^n - (-\phi^{-n}))/\sqrt{5}$. For small values of n, we can do this in constant time, but beware of using it for larger values where rounding errors may cause problems.

Binomial Coefficients Binomial coefficients are defined as the k_{th} coefficients of the expansion of the n_{th} power of a binomial $(x + y)$. For example, $(x + y)^3 = 1x^3 + 3x^2y + 3xy^2 + 1y^3$, so $\{1, 3, 3, 1\}$ are the binomial coefficients for $n = 3$ and $k = \{0, 1, 2, 3\}$. These coefficients can also be interpreted as the number of ways to choose k distinct items from a set of n total items, usually denoted as $\binom{n}{k}$. A closed-form solution for binomial coefficients can be given in terms of factorials:

$$\binom{n}{k} = C(n, k) = \frac{n!}{(n-k)! \times k!} \tag{3}$$

However, this solution is slow to calculate for for large n and k, and likely to cause problems with integer overflows. A much better solution is the recursive formula

$$C(n,0) = C(n,n) = 1C(n,k) = C(n-1,k-1) + C(n-1,k) \tag{4}$$

When large number of binomial coefficients are required, it can be worthwhile to construct the full Pascal's triangle from this recursive solution:

```
n=0 1 n=1 1 1 n=2 1 2 1 n=3 1 3 3 1 n=4 1 4 6 4 1 n=5 1 5 10 10 5 1 n=6 1 6 15 20 15 6 1
```

Catalan Numbers While I expect everyone has heard of Fibonacci number and binomial coefficients before, probably most of you are completely new to what are known as the Catalan numbers. Before I discuss what they're used for, let's define them. Just like the binomial coefficients, they can be represented as a closed-form expression in terms of factorials, which in turn can be broken down into binomial coefficients:

$$Cat(n) = \frac{2n!}{n! \times n! \times (n+1)} Cat(n) = \binom{2n}{n} \times \frac{1}{n+1} Cat(n) = \binom{2n}{n} - \binom{2n}{n+1} \tag{5}$$

We can also define them recursively:

$$Cat(0) = 1 Cat(n) = \frac{2n \times (2n-1)}{(n+1) \times n} \times Cat(n-1) \tag{6}$$

So we can use the same approaches we might take for the binomial coefficients to calculate these. The first few Catalan numbers are 1, 1, 2, 5, 14, 42, 132, 429, 1,430, 4,862, 16,796, 58,786, 208,012, 742,900, 2,674,440, 9,694,845, 35,357,670.

The Catalan numbers have an *enormous* number of possible interpretations combinatorially, most notably as the number of *Dyck words* of length $2n$, where a Dyck word is a string with n Xs and n Ys where no prefix of the string contains more Ys than Xs. Other interpretations: the number of ways to correctly match n pairs of parentheses, the number of ways to triangulate a polygon of $n+2$ sides, the number of distinct binary trees with n nodes, *etc.*, *etc.* There are far too many interpretations for me to cover them all, so in general, if you come across a combinatorial problem you're not sure about - check the first few simple values and see if they are the Catalan numbers!

OEIS When working with an unknown sequence, the Online Encyclopedia of Integer Sequences is a great resource (if the internet is available to you). Generate the first 10 or so of the sequence, which can usually be done by hand, and then run a search on the OEIS. If you're lucky and the sequence is clearly identifiable, it will tell you what the sequence is and give you the general formula on how to construct the sequence efficiently.

Probability OK, so with combinatorics we've been interested in counting the number of ways of some event happening N_{event}. Suppose we instead want to find out how likely it is that that event will happen. For this, we need to also know the *total* number of possibilities that could happen N_{total}; then we can calculate the probability as the ratio of the two:

$p_{event} = \frac{N_{event}}{N_{total}}$

Note that this is always a number between 0 and 1, inclusive. To do this calculation correctly, it is essential to correctly enumerate all the distinct events that could happen. For example, suppose we wished to know how likely it is that we would get a sum of 3 on a throw of two six-sided dice. An incorrect way to calculate this might be to note that there are 11 different possible sums of the dice (2 to 12) and one of them is the number that we want, giving

$p_3 = \frac{1}{11}$(wrong!)

This is the wrong answer, however. To obtain the right answer, we need to notice that there are actually 36 total possibilities, each corresponding to one roll of the first and one of the second dice and 2 different ways of rolling a sum of 3 - a 2 on the first and a 1 on the second dice and vice versa, giving

$p_3 = \frac{2}{36} = \frac{1}{18}$(right!)

Combinatorics will obviously come in quite handy when trying to count these possibilities. Let's look at another example. Suppose we choose four numbers at random from the integers between 1 and 10 without replacement and we wish to know how likely it is that we get both 1 and 2 in our set. We know that there are $2^{10} = 1024$ total subsets, but this includes subsets that are smaller or larger than 4 elements. Instead for the denominator we need the number of subsets of size 4, which is $\binom{10}{4} = 210$. How many of them contain both 1 and 2? Well, this is equivalent to taking the eight elements that are larger than 2 and randomly choosing 2 of them to fill up the remainder of the set, or $\binom{8}{2} = 28$. This gives us a probability of $\frac{28}{210} = \frac{2}{15}$.

Probabilities can be manipulated in several different ways. Most importantly, if we have two different, disjoint events that could happen and we wish to know the probability of either one of them happening, it's simply the sum of the probabilities of either of them happening.

Expected Values Very often we're interested not only probabilities but also in some variable associated with each event. Suppose we have a number of different possibilities, each of which has a score or a value associated with it, and we wish to know what the average score will be if we randomly pick events repeatedly. For example, suppose we roll two dice and win \$10 if the product of the dice is at least 25, but lose \$2 otherwise. How much money are we likely to win on average? Well, first let's calculate the probabilities of each event. As we described above, there are 36 different possible outcomes for the pair of rolls. 4 of these - (6, 6), (5, 6), (6, 5), (5, 5) - are wins for us, while the remaining 32 are losses. The expected value of the roll is then the sum of the products of the probability and score for each event:

$EV = \frac{4}{36} \times 10 + \frac{32}{36} \times -2 = -\frac{2}{3}$.

So on average, we can expect to lose \$0.66 a roll playing this game.

Very often we'll want to compare the expected value of several different options and pick the best one. Usually the difficulty here is not calculating the expected value, but correctly identifying the score of each possibility for our option and using combinatorics to calculate the probability of that possibility occuring. This is what your homework assignment this week will be focused on.

Bayes' Theorem (optional) $P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$.

Lab Section Again, we're going to work on this week's homework problem during lab section:
Google Code Jam Round 1A 2012 - Password Problem