Counting III
Probability I

Margaret M. Fleck

6 November 2009

This lecture covers the rest of section 5.4 of Rosen, including combinatorial proof, plus some material from section 5.5. It then starts the discussion of probability.

1 Announcements

We're aiming to return exams next Wednesday. Late next week if we fail on Wednesday. From browsing exams, it looks like people did ok on the whole.

2 Overview

We'll see a bunch of combinatorial formulas, as well as some extended cases of counting combinations and permutations. For all of them, there is a good picture or construction that can help you remember the formula, or reconstruct it if necessary. These formulas would not be so pleasant to memorize blindly.

3 Combinatorial proofs

There are a large number of useful identities involving binomial coefficients. A very simple example is:

Claim 1 $\binom{n}{k} = \binom{n}{n-k}$

There are two ways to prove this. First, we could convert both sides to expressions involving factorials and show that they are equal.

Proof: $\binom{n}{n} = \frac{n!}{k!(n-k)!}$
Also, $\binom{n}{n-k} = \frac{n!}{(n-k)!(n-(n-k))!} = \frac{n!}{(n-k)!k!}$
So $\binom{n}{k} = \binom{n}{n-k}$

Alternatively, you can prove this by choosing two sets and discussing why the two numbers of combinations must be the same.

Suppose we have a set S containing n elements. We can match up the subsets containing k elements with the subsets containing $n-k$ elements. Specifically, each subset A of S is matched to $S-A$. This is a bijection, so there are the same number of subsets of size k as subsets of size $n-k$. So $C(n,k)$ and $C(n,n-k)$ must be equal. That is, $\binom{n}{k} = \binom{n}{n-k}$ $\square$

The second proof technique is called a "combinatorial proof."

4 Vandermonde's Identity

Here's another useful identity:

Claim 2 *(Vandermonde's Identity)* $\binom{n+m}{r} = \sum_{k=0}^{r} \binom{n}{k}\binom{m}{r-k}$

We can prove this using a combinatorial argument, as follows:

Proof: Suppose that we have a set A with n elements and a set B with m elements, where A and B don't overlap. If we want to choose r elements from $A \cup B$, we will have to pick some number k from A and then the remaining $r - k$ from B.

So, we need to walk through all possible values of k. For each value of k, we pick k elements from A. This can be done in $\binom{n}{k}$ ways. We then pick $r - k$ elements from B, which can be done in $\binom{m}{r-k}$ ways. So, using the sum and product rule, the total number of choices is

$$\sum_{k=0}^{r} \binom{n}{k}\binom{m}{r-k}$$

5 Pascal's identity

Here's another well-known identity:

Claim 3 *(Pascal's identity)* $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$

[Draw Pascal's triangle. See Rosen p. 367 for a picture.]

If we have Pascal's identity, we can give a recursive definition for the binomial coefficients:

Base: For any natural number k, $\binom{k}{0} = 1$ and $\binom{k}{k} = 1$.

Induction: for any positive numbers n and k, $k \leq n$ $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

6 Proof of Pascal's identity

[This section won't be presented in lecture, but is included as an additional example.]

There are (at least) two ways to prove this identity. We can prove it directly from the factorial equations:

Proof:

$$\begin{aligned}
\left(\begin{array}{c} n \\ k-1 \end{array} \right) + \left(\begin{array}{c} n \\ k \end{array} \right) &= \frac{n!}{(k-1)!(n-(k-1))!} \cdot \frac{n!}{k!(n-k)!} \\
&= \frac{n!}{(k-1)!((n+1)-k)!} \cdot \frac{n!}{k!(n-k)!} \\
&= \frac{kn!}{k!((n+1)-k)!} \cdot \frac{(n+1-k)n!}{k!((n+1)-k)!} \\
&= \frac{kn! + (n+1-k)n!}{k!((n+1)-k)!} \\
&= \frac{(n+1)n!}{k!((n+1)-k)!} \\
&= \frac{(n+1)!}{k!((n+1)-k)!} \\
&= \left(\begin{array}{c} n+1 \\ k \end{array} \right)
\end{aligned}$$

Alternatively, we can use a combinatorial proof:

Proof: Suppose that S is a set with $n+1$ elements. $\left(\begin{array}{c} n+1 \\ k \end{array} \right)$ is the number of k-element subsets of S.

Pick some element $a \in S$. We can divide the k-element subsets of S into two groups: those that contain a and those that don't.

The k-element subsets of S that don't contain a are the same as the k-element subsets of $S - \{a\}$, i.e. the k-element subsets of a set with n elements. This is $\left(\begin{array}{c} n \\ k \end{array} \right)$.

The k-element subsets of S that do contain a can be formed by adding a onto each $k-1$-element subsets of $S - \{a\}$. There are $\left(\begin{array}{c} n \\ k-1 \end{array} \right)$ of these.

So there are $\left(\begin{array}{c} n \\ k-1 \end{array} \right) + \left(\begin{array}{c} n \\ k \end{array} \right)$ k-element subsets of S. That is $\left(\begin{array}{c} n+1 \\ k \end{array} \right) = \left(\begin{array}{c} n \\ k-1 \end{array} \right) + \left(\begin{array}{c} n \\ k \end{array} \right)$

Both methods of proof work fine. The combinatorial proofs are worth knowing about because they are sometimes easier to read/write.

7 Permutations with identical objects

Life gets more interesting when our problem involves multiple copies of the same (or indistinguishable) objects. I'm not going to walk through all the possibilities here, because they get ugly fast. I'll just show a couple cases where the analysis is still fairly nice.

First, suppose I have a list of objects such as $L = (c, o, l, l, e, g, e)$ that contains some duplicates. How many ways can I put the elements of L into different orders?

If we ignored the fact that some objects are duplicates, we would calculate 7! permutations of this list. However, this is double-counting some possibilities, because it doesn't matter what order we put the duplicates in. So we need to divide out by the number of ways we can permute the duplicates. In this case, we have 2! ways to permute the l's and 2! ways to permute the e's. So the true number of orderings of L is $\frac{7!}{2!2!}$.

Similarly, the number of reorderings of $J = (a, p, p, l, e, t, r, e, e, s)$ is $\frac{10!}{2!3!}$.

In general, suppose we have n objects, where n_1 are of type 1, n_2 are of type 2, and so forth through n_k are of type k. Then the number of ways to order our list of objects is $\frac{n!}{n_1!n_2!...n_k!}$.

8 Combinations with repetition

Suppose I have a set S and I want to select a group of objects of the types listed in S, but I'm allowed to pick more than one of each type of object. For example, suppose I want to pick 6 plants for my garden and the set of available plants is $S = \{\text{thyme}, \text{oregano}, \text{mint}\}$. The garden store can supply as many as I want of any type of plant. I could pick 3 thyme and 3 mint. Or I could pick 2 thyme, 1 oregano, and 3 mint.

There's a clever way to count the possibilities here. Let's draw a picture of

a selection as follows. We'll group all our thymes together, then our oreganos, then our mints. Between each pair of groups, we'll put a cardboard separator #. So 2 thyme, 1 oregano, and 3 mint looks like

T T # O # M M M

And 3 thyme and 3 mint looks like

T T T ## M M M

But this picture is redundant, since the items before the first separator are always thymes, the ones between the separators are oreganos, and the last group are mints. So we can simplify the diagram by using a star for each object and remembering their types implicitly. Then 2 thyme, 1 oregano, and 3 mint looks like

** # * # ***

And 3 thyme and 3 mint looks like

*** ## ***

To count these pictures, we need to count the number of ways to arrange 6 stars and two #'s. That is, we have 8 positions and need to choose 2 to fill with #'s. In other words, $\binom{8}{2}$.

In general, suppose we are picking a group of k objects (with possible duplicates) from a list of n types. Then our picture will contain k stars and $n-1$ #'s. So we have $k+n-1$ positions in the picture and need to choose $n-1$ positions to contain the #'s. So the number of possible pictures is $\binom{k+n-1}{n-1}$.

Notice that this is equal to $\binom{k+n-1}{k}$ because we have an identity that says so (see last lecture). We could have done our counting by picking a

subset of k positions in the diagram that we would fill with stars (and then the rest of the positions will get the #'s).

If wanted to pick 20 plants and there were five types available, I would have $\left(\begin{array}{c} 24 \\ 4 \end{array} \right) = \left(\begin{array}{c} 24 \\ 20 \end{array} \right)$ options for how to make my selection. $\left(\begin{array}{c} 24 \\ 4 \end{array} \right) = \frac{24 \cdot 23 \cdot 22 \cdot 21}{4 \cdot 3 \cdot 3} = 23 \cdot 22 \cdot 21$.

9 Introduction to probability

There are three basic motivations for using probability in computer science.

- Modelling real-world events which aren't entirely predictable (rolls of dice, presence of worms in apples, ...).
- Modelling the average behavior of an algorithm, e.g. a sorting algorithm, over a long series of inputs coming from some real-world application.
- Deliberate randomization to prevent an algorithm from interacting badly with patterns in the input data.

The first motivation is very old, because people have been trying for centuries (millenia?) to build strategies for winning money at games of chance.

We've seen the second earlier this term.

The last is less obvious, but also predates computer science. Suppose that you are an archeologist looking for a buried structure such as a temple. Manmade structures frequently have regular spacing of key features such as walls. If you dig exploratory holes at regularly-spaced intervals, you have a significant chance of systematically missing what you are looking for. So it's better practice to dig holes at randomized locations.

Similarly, identifiers (i.e. names of variable and functions) in programs are not randomly-generated strings. Programmers tend to base them on words in English (or whatever language they speak) and certain sorts of strings are much more likely as English words. So compiler algorithms which store

identifiers in tables typically run the identifiers through a randomization algorithm ("hashing") to scatter them uniformly over the storage positions in the table.

10 Overview

A probability analysis is based on a "sample space," which is a set of "outcomes." An outcome is something an experiment might produce. For example, if you are rolling two 6-sided dice, an outcome would be an ordered pair showing the two values e.g. $(2, 4)$. The set of outcomes would then contain all 36 possible ordered pairs of numbers between 1 and 6. In this class, the sample space will always be finite.

Each outcome s can have an associated probability $p(s)$. For example, if the pair of dice are fair, each outcome has probability $\frac{1}{36}$. If the dice are weighted, it might be that (say) $p(1, 6) = \frac{2}{32}$ but $p(2, 6) = \frac{0}{32}$ (i.e. that pair can never actually be produced). We'll start by assuming that all outcomes have the same probability.

An "event" is a subset of the sample space. Typically, our event will contain the outcomes matching some high-level description, e.g. the event E might contain all the dice rolls in which the two numbers add up to 8.

Dangling question to finish in lecture Monday: for how many of our 36 dice outcomes do the two numbers add up to 8? and, therefore, what is the probability that you get a sum of 8 when you roll two (fair) dice?