Analysis of Algorithms, CSOR 4231
Professor Clifford Stein
Columbia University, Fall 2005

Review of Mathematical Concepts

Justin Cranshaw

1 Summations

Here are some basic techniques that can be used to change a given summation into a form that we know how to deal with.

1.1 Changing the Lower Bound

Given the formula $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$, what is $\sum_{i=40}^{100} i$?

Answer:

$$\begin{aligned}
\sum_{i=40}^{100} i &= \sum_{i=1}^{100} i - \sum_{i=1}^{39} i \\
&= \frac{(100)(101)}{2} - \frac{(39)(40)}{2} \\
&= \frac{100 \cdot 101 - 39 \cdot 40}{2} \\
&= \frac{10100 - 1560}{2} \\
&= \frac{8540}{2} = 4270
\end{aligned}$$

Alternative Method:

$$\begin{aligned}
\sum_{i=40}^{100} i &= \sum_{i=1}^{61} (i + 39) \\
&= \sum_{i=1}^{61} i + \sum_{i=1}^{61} 39 \\
&= \frac{(62)(61)}{2} + (61)(39) \\
&= (31)(61) + (61)(39) \\
&= (61)(31 + 39) \\
&= (61)(70) = 4270
\end{aligned}$$

1.2 Working With Different Upper Bounds

Given that $\sum_{i=1}^{n} \frac{1}{i} = \Theta(\log n)$, what is $\sum_{i=1}^{\log n} \frac{1}{i}$?

The trick here is not to get confused by the fact that the upper bound is some function of n. The harmonic series formula tells us that $\sum_{i=1}^{n} = \Theta(\ln(\textit{the upper bound of the sum}))$. So in this case we get

$$\sum_{i=1}^{\log n} \frac{1}{i} = \Theta(\ln(\log n)).$$

1.3 Combining Different Formulas

Given $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$, and $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$, what is $\sum_{i=1}^{n} i(i+1)$?

Answer:

$$\begin{aligned}
\sum_{i=1}^{n} i(i+1) &= \sum_{i=1}^{n} i^2 + i \\
&= \sum_{i=1}^{n} i^2 + \sum_{i=1}^{n} i \\
&= \frac{n(n+1)(2n+1)}{6} + \frac{n(n+1)}{2} \\
&= \frac{n(n+1)(2n+1) + 3n(n+1)}{6} \\
&= \frac{n(n+1)(2n+4)}{6}
\end{aligned}$$

1.4 Reversing the order of a summation

Sometimes a summation can be put into a more familiar form by expanding the terms and rearranging them in a different order. Most often, this new order is just a reversal of how the terms were originally indexed. The summation $\sum_{i=0}^{n-1} \frac{n}{n-i}$ is an example of when such a technique is useful.

$$\begin{aligned}
\sum_{i=0}^{n-1} \frac{n}{n-i} &= n \sum_{i=0}^{n-1} \frac{1}{n-i} \\
&= n \left(\frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \cdots + \frac{1}{n-(n-2)} + \frac{1}{n-(n-1)} \right) \\
&= n \left(\frac{1}{n} + \frac{1}{n-1} + \cdots + \frac{1}{2} + 1 \right) \\
&= n \left(1 + \frac{1}{2} + \cdots + \frac{1}{n-2} + \frac{1}{n-1} + \frac{1}{n} \right) \\
&= n \sum_{k=1}^{n} \frac{1}{k} \\
&= n \left[\Theta(\log n) \right] \\
&= \Theta(n \log n)
\end{aligned}$$

2 Counting

Most of the pitfalls that people run into with counting is when to multiply different counts and when to add different counts. I'll try to make that clear here.

2.1 Definitions

When do you use the sum rule? The sum rule tells us that if we are selecting a single item from two disjoint sets, the total number of ways to select the element is the sum of the cardinalities of the two sets. For example, if task a can be done in n_a ways and task b can be done in n_b ways, but the two tasks can not be done at the same time, then the total number of ways to do *one* of these tasks is $n_a + n_b$.

When do you use the product rule? Generally, any time choices are made in sequence you use the product rule. Formally, the product rule tells us that the number of ways to choose an ordered pair of elements is the number of ways to choose the first component times the number of ways to choose the second component. For example. If there are 10 paths from a to b, and 20 paths from b to c, then there are 200 paths from a to c through b. Notice that the selection of a path from a to c is done in sequence. First you select an a-b path, then you select a b-c path.

A *permutation* of a set S is a fixed ordered **sequence** of the elements of S. If $|S| = n$, then there are $n!$ permutations of the elements of S. This follows directly from the product rule. A *k-permutation* on a set S is an ordered sequence of k elements from S. Again, if $|S| = n$, then the total number of k permutations is $\frac{n!}{(n-k)!}$, which also follows directly from the product rule.

A *k-combination* differers from a permutation in that a combination is a **set** rather than a sequence, and hence the order of the elements selected is of no concern. If we divide the number of k-permutations of S by the total number of ways to mix up k elements, this will give us a count of the total number of combinations. The number of k-combinations is often denoted $\binom{n}{k}$ and read "n choose k." And so we get $\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$.

2.2 Summations and counting

Many people were confused why $\sum_{i=1}^{n}\sum_{j=i+1}^{n} 1 = \binom{n}{2}$. This may be surprising at first since it seems like we are working more with ordered pairs than sets of size 2. However note that for every $k, l \in \{1, 2, \ldots, n\}$, at exactly one iteration either $j = k$ and $i = l$, or $j = l$ and $i = k$. Thus every iteration counts exactly one way to choose two elements from n, and so the total number of iterations is the number of ways to choose 2 from n, or $\binom{n}{2}$.

2.3 Using the sum rule

Suppose we have a 16 bit string, $x \in \{0,1\}^{16}$. Define the weight of binary string x as the number of 1's in x. How may 16 bit binary strings are there with a weight that is divisible by 4?

First let's ask how many 16 bit binary strings there are of a given weight k? Note that any bitstring can be expressed as the set of indices on which the string is 1. For instance, over length 4 bit strings, the set $\{1, 4\}$ represents the string 1001, and the set $\{2\}$ represents the string 0100.

There are of course, 16 such positions in our case, so a given bit string over $\{0,1\}^{16}$ can be expressed as a subset of $\{1, 2, \ldots, 15, 16\}$. Since the order that the indices in this set does not matter, the total number of strings of weight k is the total number of length k subsets of $\{1, 2, \ldots, 16\}$, which is $\binom{16}{k}$. Since we are counting weights that are divisible by 4, the possible string weights are 0,4,8,12, and 16. Thus the total number of strings with weight divisible by 4 is

$$\binom{16}{0} + \binom{16}{4} + \binom{16}{8} + \binom{16}{12} + \binom{16}{16} = 1 + 1820 + 12870 + 1820 + 1$$
$$= 16512$$

Note that we use the sum rule here because a bitstring can only be one weight at a time. The event that the string is weight 4 is disjoint from the event that it is weight 12.

2.4 Using the product rule

We can use the product rule to count the number of functions that go from a set X to a set Y. Recall that a function $f : X \to Y$ assigns exactly one element of Y to every element of X. Suppose that $|X| = n$ and $|Y| = m$. For each element $x \in X$, there are m choices for $f(x)$. Since the choices for each $x \in X$ are made in sequence, to get the total number of functions, for each x we simply multiply by the number of choices we have for $f(x)$. Since $|X| = n$ it follows that the number of functions from X to Y is m^n.

3 Probability

3.1 Getting a Flush in Poker

What is the probability of being dealt a flush (**all the same suit**) in a 5 card poker game?

It is clear from the definition of probability that

$$\Pr[\text{getting a flush}] = \frac{\text{total number of flush hands}}{\text{total number of poker hands}}. \tag{1}$$

Since we are dealt 5 cards out of a deck of 52, and since the order of the cards does not matter, we get that the total number of poker hands is

$$\begin{aligned}\text{total number of poker hands} &= \binom{52}{5} \\ &= \frac{52!}{5!\,(52-5)!} \\ &= \frac{52!}{5!\,(47!)} \\ &= \frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5!} \\ &= \frac{311875200}{120} = 2598960\end{aligned}$$

Note that there are 13 cards from each of the 4 poker suits. Thus, given any particular suit, there are $\binom{13}{5}$ possible flushes within that suit. Since there are 4 suits, the total number of flushes is $4\binom{13}{5}$.

$$\begin{aligned} \text{total number of poker hands} &= 4\binom{13}{5} \\ &= 4\frac{13!}{5!\,(13-5)!} \\ &= 4\frac{13!}{5!\,(8!)} \\ &= 4\frac{13\cdot 12\cdot 11\cdot 10\cdot 9}{5!} \\ &= 4\frac{154440}{120} = 4\cdot 1287 = 5148 \end{aligned}$$

Thus we get, that

$$\begin{aligned} \Pr[\text{getting a flush}] &= \frac{\text{total number of flush hands}}{\text{total number of poker hands}} \\ &= \frac{4\binom{13}{5}}{\binom{52}{5}} \\ &= \frac{5148}{2598960} \approx \frac{1}{500} \end{aligned}$$

3.2 Getting a Full House in Poker

What is the probability of being dealt a full house (**one pair, and a three of a kind**) in a 5 card poker game?

Again from the definition of probability that

$$\Pr[\text{getting a full house}] = \frac{\text{total number of full house hands}}{\text{total number of poker hands}}. \tag{2}$$

We showed above that the total number of hands is $\binom{52}{5}$, we now just need to count the total number of full houses. There are a total of 13 possible card values, $(2, 3, \ldots, 9, 10, J, Q, K, A)$. Thus, there are 13 possible values for the triplet. Once we fix a value, v for the triplet, there are 4 differently suited cards achieving value v, so there are $\binom{4}{3}$ possible triplets with value v. Now fix a value w different from v for the pair. There are 12 possible choices for w. Again, once we fix w, there are 4 differently suited cards with value w. Thus in all there are $\binom{4}{2}$ possible pairs with value w. Thus the total number of full house hands is $13\binom{4}{3}12\binom{4}{2}$. Thus

$$\begin{aligned} \Pr[\text{getting a full house}] &= \frac{\text{total number of full house hands}}{\text{total number of poker hands}} \\ &= \frac{13\binom{4}{3}12\binom{4}{2}}{\binom{52}{5}} \\ &= \frac{13\cdot 4\cdot 12\cdot 6}{2598960} \\ &= \frac{3744}{2598960} \approx \frac{1}{694} \end{aligned}$$

3.3 Conditional Probability

The conditional probability of an event E given that F has occurred is

$$\Pr[E|F] = \frac{\Pr[E \cap F]}{\Pr[F]}.$$

Given that a family with two children has at least 1 boy, what is the probability that they have 2 boys. Then we let E bet the event that both children are boys, and let F be the event that at least one child is a boy. We would like to calculate

$$\Pr[E|F] = \frac{\Pr[E \cap F]}{\Pr[F]}.$$

First, the universe is BB, BG, GB, GG. Thus, since $E \cap F = \{BB\}$, it follows that $\Pr[E \cap F] = \frac{1}{4}$. Furthermore, since $F = \{BB, BG, GB\}$, then $\Pr[F] = \frac{3}{4}$. It follows that

$$\begin{aligned} \Pr[E|F] &= \frac{\Pr[E \cap F]}{\Pr[F]} \\ &= \frac{1/4}{3/4} = \frac{1}{3} \end{aligned}$$

4 Random Variables

Random variables are one of those concepts in math that have an unfortunate name, often causing more confusion on the subject than is justified. This is because they are neither *random*, nor are they even *variables* in any traditional sense. Random variables are actually functions from the event space of a probabilistic experiment, to the real numbers. Put simply, they are just a way to assign a *value* to every outcome in an event space.

A simply concrete example that should clarify this issue is with rolling dice. The event space for rolling a single dice is $S = \{1 \text{ is face up}, 2 \text{ is face up}, \ldots, 6 \text{ is face up}\}$. I'm being overly verbose with the individual events for a reason. I want to highlight the fact that there is no initial numerical value associated with each individual outcome until you assign a random variable to the event space. So let's define the following random variable $X : S \to \mathbb{R}$ such that

$$\begin{aligned} X(1 \text{ is face up}) &= 1 \\ X(2 \text{ is face up}) &= 2 \\ X(3 \text{ is face up}) &= 3 \\ X(4 \text{ is face up}) &= 4 \\ X(5 \text{ is face up}) &= 5 \\ X(6 \text{ is face up}) &= 6 \end{aligned}$$

Note that there is nothing special about the values that I assigned to each outcome. This just happens to be the most often used and natural random variable associated with the event of rolling a dice. I could have for instance, defined a random variable $X_{\pi,\sqrt{2}}$ that assigned all events with an odd number on the face the value π and all events with an even number on the face $\sqrt{2}$.

4.1 Expectation of a random variable

Given that the event space of a probabilistic experiment is S, define the *expected value* of a random variable $X : S \to \mathbb{R}$ as

$$E[X] = \sum_{s \in S} X(s)\Pr[s],$$

where $\Pr[s]$ is the probability that event s occurs.

Let's calculate the expected value of the random variable on dice where $X(i \text{ is face up}) = i$. Let's assume that a fair dice is used and each outcome is equally likely. Then we get

$$\begin{aligned} E[X] &= 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + \cdots + 6 \cdot \frac{1}{6} \\ &= \frac{1}{6} \cdot (1+2+3+4+5+6) \\ &= \frac{1}{6} \cdot (21) \\ &= 3.5 \end{aligned}$$

4.2 Indicator random variables

An *indicator random variable* X over an event space S is just a random variable $X : S \to \{0,1\}$. IRV's are particularly useful when counting the expected number of times a particular event occurs in a probabilistic experiment.

Let's do another experiment with dice, except let's use a loaded dice that rolls a 6 with probability $\frac{1}{2}$ and rolls all other numbers with probability $\frac{1}{10}$. Suppose we roll this dice 100 times. Then let $X_{6,i}$ be an IRV such that $X_{6,i} = 1$ if and only if a 6 is rolled on toss i. Let X be the total number of sixes rolled in the experiment. What is the expected value of X?

$$\begin{aligned} E[X] &= E\left[\sum_{i=1}^{100} X_{6,i}\right] \\ &= \sum_{i=1}^{100} E[X_{6,i}] \\ &= \sum_{i=1}^{100} \Pr[\text{a six is rolled on toss } i] \\ &= \sum_{i=1}^{100} \Pr[\text{a six is rolled on a single toss}] \qquad \text{(Since each toss is independent)} \\ &= \sum_{i=1}^{100} \frac{1}{2} \\ &= 100 \cdot \frac{1}{2} \\ &= 50 \end{aligned}$$

So we expect 50 of the 100 rolls to be six.

5 Recurrences

5.1 $T(n) = T(\sqrt{n}) + n$

Let's unfold this recurrence to find a general expression at step i.

$$\begin{aligned} T(n) &= T(n^{1/2}) + n \\ &= T(n^{1/4}) + 2n \\ &= T(n^{1/8}) + 3n \\ &\quad \dots \\ &= T(n^{1/2^i}) + i \cdot n \end{aligned}$$

Now let's assume a base case of $T(c) = 0$, for $c < 2$, which we can do without loss of generality if we only care about the asymptotic behavior. Then, since the base case returns 0, the only contributing factor in this recurrence is the work done at each level. Thus to solve this recurrence, we just need to find the i such that $n^{1/2^i} < 2$.

$$\begin{aligned} n^{\frac{1}{2^i}} &< 2 \\ n &< 2^{2^i} \\ \lg n &< 2^i \\ \lg\lg n &< i \end{aligned}$$

Thus after $O(\lg\lg n)$ many steps, the recurrence bottoms out, and $i \cdot n = O(n \lg\lg n)$ work is done in total. Thus $T(n) = O(n \lg\lg n)$.

6 Big Theta Proof

To prove that $n \log n = \Theta(\log n!)$, it suffices to show that $n \log n = O(\log n!)$ and $\log n! = O(n \log n)$.

Show $n \log n = O(\log n!)$:

Note that for $i = 0, 1, \dots (n-1)$ we have $(n-i)(i+1) \geq n$. Thus, we can show that

$$\begin{aligned} n^n &= (n)(n)\dots(n)(n) \\ &\leq [(n)(1)][(n-1)(1+1)]\dots[(n-(n-2))((n-2)+1)][(n-(n-1))((n-1)+1)] \\ &= (n)(1)(n-1)(2)\dots(2)(n-1)(1)(n) \\ &= (n!)^2 \end{aligned}$$

Thus, it follows that $\log(n^n) = n \log n \leq \log((n!)^2) = 2\log(n!)$, and hence $n \log n = O(\log(n!))$ with witness $c = 2$ and $k = 1$.

Show $\log n! = O(n \log n)$:

Note that $n! = (1)(2)\dots(n-1)(n) \leq (n)(n)\dots(n)(n) = n^n$. Thus $\log(n!) \leq \log n^n = n \log n$ Thus, we have exhibited, with witnesses $c = 1$ and $k = 1$, that $\log(n!) = O(n \log n)$.

7 Limits

Occasionally in this course you will need to work with limits. I'll briefly review some techniques involved with solving the types of limits we will see in this class.

Usually we will be concerned with the limiting value of a ratio of two functions. For instance, given functions $f(n)$ and $g(n)$, we will be concerned with limits of the form

$$\lim_{n\to\infty} \frac{f(n)}{g(n)}.$$

This limit will fall into one of three scenarios:

1. $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.
2. $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$.
3. $\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$ from some constant c.

Limits can be thought of intuitively as the behavior of the function as n goes off to infinity. So case 1 means, as n approaches infinity, the value of $\frac{f(n)}{g(n)}$ vanishes. Case 2 means, as n goes to infinity, $\frac{f(n)}{g(n)}$ itself also approaches infinity without bound. Finally case 3 means as n approaches infinity, the fraction $\frac{f(n)}{g(n)}$ approaches some constant c.

We can use our notions of asymptotic growth to define when each case occurs.

1. If $f(n) = o\left(g(n)\right)$, then $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.
2. If $f(n) = \omega\left(g(n)\right)$, then $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$.
3. If $f(n) = \Theta\left(g(n)\right)$, then $\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$.

I realize that this is a slightly recursive definition of limits, since the book defines the asymptotic functions o, ω, and Θ with limits. This is intentional, since I wanted to include as little calculus as possible in this review. However in practice, one typically cancels and simplifies the fraction $\frac{f(n)}{g(n)}$, to get a new fraction $\frac{\hat{f}(n)}{\hat{g}(n)}$ such that it is obvious how $\hat{f}$ and $\hat{g}$ relate to each other asymptotically. Aside from algebraic manipulations of the numerator and denominator, there are also calculus based methods to find such functions $\hat{f}$ and $\hat{g}$.

7.1 L'Hopital's Rule

L'Hopital's rule for limits is an iterative process by which a limit can be reduced into a more simple form. It is defined very simply using the derivative. Suppose we wish to find the limit $\lim_{n\to\infty} \frac{f(n)}{g(n)}$. L'Hopital's rule says that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{f'(n)}{g'(n)},$$

where $f'(n)$ and $g'(n)$ are the first derivatives of f and g respectively.

In practice, if one were faced with a limit he/she could not solve, then he/she could simply take the derivatives of the numerator and denominator to get a new limit that is presumably easier to work with. This process iterates. That is, if the new functions are still too difficult to work with, simply differentiate again and again until the limit is in a workable form, or until a pattern is noticed.