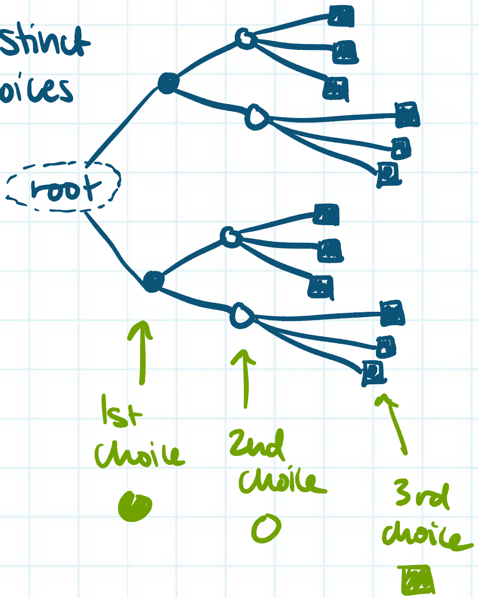
recap

without replacement # of choices reduces each time

with replacement "putting things back before redrawing" → two trials can have the same outcome.

First Rule of Counting arrangement made by succession of distinct choices



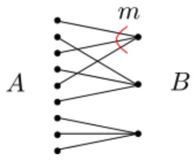
2nd Rule of Counting don't care about "order" of final arrangement

↑ think about using when we're picking out of a set

2nd rule : basically executing 1st rule, then realizing that some of these arrangements are actually the same thing!

↓

$$\binom{n}{k} = \frac{n!}{(n-k)!\,k!}$$



$$|B| = \frac{|A|}{m}$$

recap

Super versatile, sometimes tricky to identify

Balls + Bins Method (aka Stars + Bars)

putting indistinguishable items into distinct "bins"

think about it as counting bitstrings with a set number of ones

0010010 or 00010001 or 1100000

bin 1 bin 2 bin 3

how to count this? k objects, n bins (n-1 dividers)

_ _ _ _ _ _ ... _ _ ← create (n-1) + k slots for objects + dividers

then pick which k slots will be for the objects → $\binom{n+k-1}{k}$

balls and bins™ $\binom{n+k-1}{k}$

aka stars and bars.

n = bins; k = objects (in the above formula)

used when objects (balls) are indistinguishable but bins are distinguishable.*

the bins might sometimes not be bins but "boundaries" !!

- how many 12-length bitstrings are there with exactly 3 ones? think of distributing the nine 0s around the fixed 1s →
  __ 1 __ 1 __ 1 __ (9 balls, 4 bins)
- arrange ABCDEF st C is left of E (doesn't need to be immediate left)? consider C and E to be fixed "posts" and arrange other letters around them →
  ___ C ___ E ___ (4 balls, 3 bins)
  - there's also like a few other ways to do this problem
- addition/distribution problems— finding 3 positive numbers that add up to 5, that kinda stuff

tbh I don't like calling this balls & bins bc <u>there are some scenarios where there are literal balls and bins but you **don't** use this, such as:</u>

- colored balls, numbered balls = they are distinguishable → 1st rule of counting
- each bin can have only 1 ball → 1st rule of counting, number of available bins goes down each time

reproduced from aishani's sp20 csm slides.

first rule of counting

used when making a succession of choices (choices do not have to necessarily be independent). good to use when order matters. *

↳ AND we're picking from same pool of choices each time

if you're choosing items out of disjoint sets (eg one hat, one shoe...) try 1st rule of counting.

eg out of one bag of 10 marbles, picking 3 where RGB ≠ GBR is 10·9·8 choices.

second rule of counting

the concept is used to get rid of duplicate configurations that should count as the same thing.

- how many ways to arrange letters in BANANA? start with 8!, then need to divide by (3!2!) to take care of the fact that rearranging the As and Ns amongst each other still gives the same configuration

* remember <u>not</u> to strictly equate a counting method with a situation (eg sampling w/o replacement, order matters, etc) since it really depends on a case-by-case basis.
better to just get familiar with a good few situations to build intuition™

random strategies

(there are many ways to solve a lot of these problems, this is just what I use!)

how to put people into even groups (ie pairs, triplets, etc)

n = number of total people; x = number of groups;

p = num people in each group.

formula: $$\frac{\binom{n}{p}\binom{n-p}{p}\binom{n-2p}{p}\cdots\binom{p}{p}}{x!}$$

ie first picking p people for 1st group, then picking p out of who is left, and so on. then div by $x!$ bc one can rearrange the order in which groups are picked.

counting "equally likely" events (by symmetry)

if one event can easily be turned into another event, then they are equally likely.

count the number of 8-length bitstrings with more zeroes than ones. so this is equal to the # of len 8 bitstrings with more ones than zeroes, because you can go from one set to the other by switching the 1/0s. so there are equal amounts of each.

- bitstrings can generally be divided into 3 categories:
  - (1) equal number of 1s & 0s (this is empty if odd-length bitstring)
  - (2) more 1s than 0s
  - (3) more 0s than 1s

bijection between these 2 sets

^^ last 2 categories will be of equal size!

so to solve the problem w the 8-length bitstring…

$$\frac{2^8 - \binom{8}{4}}{2}$$ subtract the number of category 1 strings, then divide by 2

reproduced from aishani's sp20 csm slides.

putting things in a circle

basically just count the configurations as if they weren't in a circle, then divide by the number of objects (ie the number of times you can rotate the circle and have it still be technically the same arrangement)

order does(n't) matter probability.

so intuitively, you should keep these in mind:

- if order does matter, there are more possible configurations, smaller probability of each happening
- if order doesn't matter, there are fewer configurations, greater probability of each happening
- if you are counting the event space and sample space separately, you can calculate probability by either considering order to matter or not, *as long as* you are consistent between numerator/denominator

another technique to find probability is to just multiply the probabilities of individual events (be careful of in/dependence) and account for order later on.

not in scope for mt1