CHAPTER 6 Probability Theory, Part 2: Compound Probability

Introduction

In this chapter we will deal with what are usually called "probability problems" rather than the "statistical inference problems" discussed in later chapters. The difference is that for probability problems we begin with a knowledge of the properties of the universe with which we are working. (See Chapter 4 on the definition of resampling.)

Before we get down to the business of complex probabilistic problems in this and the next two chapters, let's consider a couple of peculiar puzzles which do not fit naturally into any chapter in this book, but which are extremely valuable in showing the power of the Monte Carlo simulation method.

Puzzle Problems

The treasure fleet recovered

This problem is:

> A Spanish treasure fleet of three ships was sunk at sea off Mexico. One ship had a trunk of gold forward and another aft, another ship had a trunk of gold forward and a trunk of silver aft, while a third ship had a trunk of silver forward and another trunk of silver aft. Divers just found one of the ships and a trunk of silver in it. They are now taking bets about whether the other trunk found on the same ship will contain silver or gold. What are fair odds?

(This is a restatement of a problem that Joseph Bertrand posed early in the 19th century.) In the Goldberg variation:

> Three identical boxes each contain two coins. In one box both are pennies, in the second both are nickels, and in the third there is one penny and one nickel.
>
> A man chooses a box at random and takes out a coin. If the coin is a penny, what is the probability that the other coin in the box is also a penny?

These are the logical steps one may distinguish in arriving at a correct answer with deductive logic (portrayed in Figure 6-1):

1. Postulate three ships—Ship I with two gold chests (G-G), ship II with one gold and one silver chest (G-S), and ship III with S-S. (Choosing notation might well be considered one or more additional steps.)

2. Assert equal probabilities of each ship being found.

3. Step 2 implies equal probabilities of being found for each of the six chests.

4. Fact: Diver finds a chest of gold.

5. Step 4 implies that S-S ship III was not found; hence remove it from subsequent analysis.

6. Three possibilities: 6a) Diver found chest I-Ga, 6b) diver found I-Gb, 6c) diver found II-Gc.

From step 2, the cases a, b, and c in step 6 have equal probabilities.

7. If possibility 6a is the case, then the other trunk is I-Gb; the comparable statements for cases 6b and 6c are I-Ga and II-S.

8. From steps 6 and 7: From equal probabilities of the three cases, and no other possible outcome, p (6a) = 1/3, p (6b) = 1/3, p (6c) = 1/3,

9. So p(G) = p(6a) + p(6b) = 1/3 + 1/3 = 2/3.

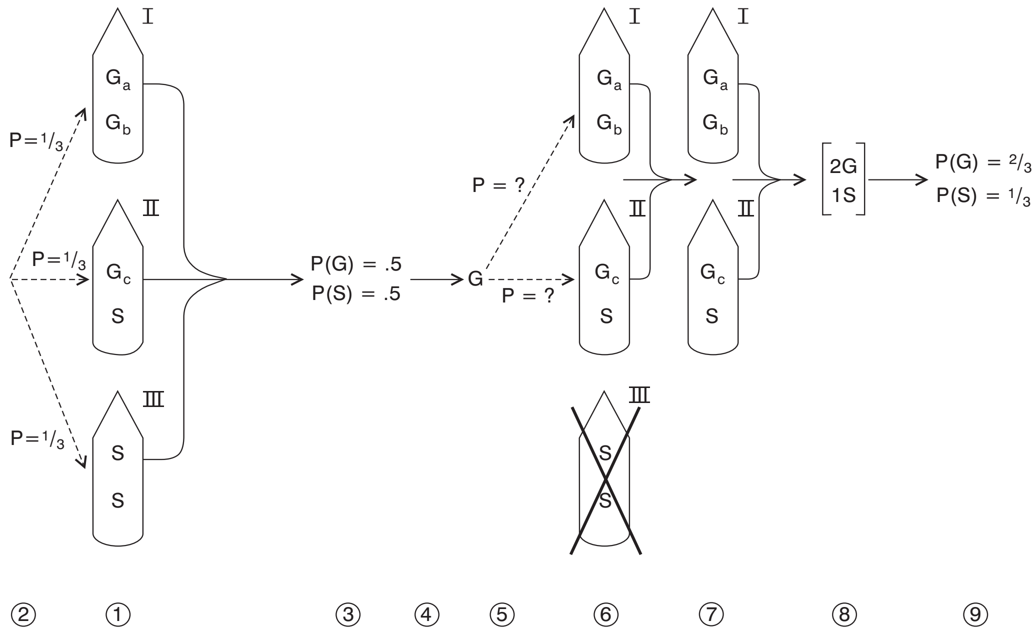

Figure 6-1: Ships with Gold and Silver

The following simulation arrives at the correct answer:

1. Construct three urns containing the numbers "7,7," "7,8," and "8,8" respectively.

2. Choose an urn at random, and shuffle the numbers in it.

3. Choose the first element in the chosen urn's vector (a vector is an array or list of numbers). If "8," stop trial and make no further record. If "7," continue.

4. Record the second element in the chosen urn's vector on the scoreboard.

5. Repeat steps (2 - 5), and calculate the proportion "7's" on a scoreboard. (The answer should be about 2/3.)

Here is a computer simulation with RESAMPLING STATS:

NUMBERS (7 7) gg
The 3 boxes, where "7"=gold, "8"=silver

NUMBERS (7 8) gs

NUMBERS (8 8) ss

REPEAT 1000

> **GENERATE 1 1,3 a**
> Select a box where gg=1, gs=2, ss=3

IF a =1

SCORE 1 z
If a=1, we're in the "gold-gold" box. That means we've picked a gold, and we're guaranteed of getting another gold (7) on our second pick. So we score a "1" for success.

END

IF a=2
If a=2, we're in the gold-silver urn

SAMPLE 1 gs b
Select a coin

IF b =7
If b ="7," we got a gold, so score 0, (for no success) because we can't get a "7" again.

SCORE 0 z

END
Note: if b =8, we got a silver on our first draw and we're not interested in the second draw unless we get a gold first.

END
Note: if a=3, we're not interested either. We can't draw a gold on our first draw.

END

SIZE z k1
How many times did we get an initial gold?

COUNT z =1 k2
Of those times, how often was our second draw a gold?

DIVIDE k2 k1 result
Calculate the latter as a proportion of the former result =0.647

The three-door problem

Consider the famous problem of the three doors: The player faces three closed containers, one containing a prize and two empty. After the player chooses, s/he is shown that one of the other two containers is empty. The player is now given the option of switching from her/his original choice to the other closed container. Should s/he do so?

Answer: Switching doubles the chances of winning.

When this problem was published in the Sunday newspapers across the U.S., the thousands of letters—including a good many from Ph.D.s in mathematics—show that logical mathematical deduction fails badly in this case. Most people—both laypersons and statisticians—arrive at the wrong answer.

Simulation, however—and *hands-on* simulation with physical symbols, rather than computer simulation—is a surefire way of obtaining and displaying the correct solution. Table 6-1 shows such a simple simulation with a random-number table. Column 1 represents the container you choose, column 2 where the prize is. Based on columns 1 and 2, column 3 indicates the container that the "host" would now open and show to be empty. Lastly, column 4 scores whether the "switch" or "remain" strategy would be preferable. A count of the number of winning cases for "switch" and the "remain" gives the result sought.

Not only is the best choice obvious with this simulation method, but you are likely to understand quickly why switching is better. No other mode of explanation or solution brings out this intuition so well. And it is much the same with other problems in probability and statistics. Simulation can provide not only answers but also insight into why the process works as it does. In contrast, formulas frequently produce obfuscation and confusion for most non-mathematicians.

Table 6-1
Simple Simulation With a Random-Number Table

Random Pick: Random Number	Random Pick: Equiv to Door	Actual Location: Random Number	Actual Location: Equiv to Door	Host Opens	Winning Strategy
10	1	01	1	2 or 3	Remain
22	2	25	2	1 or 3	R
24	2	22	2	1 or 3	R
42	1	06	3	2	Change
37	3	81	2	1	C
77	1	11	1	2 or 3	R
99	3	56	2	1	C
96	3	05	2	1	C
89	2	63	3	1	C
85	2	43	1	3	C
28	2	88	2	1 or 3	R
63	3	48	1	2	C
09	3	52	2	1	C
10	1	87	2	3	C
74	1	71	1	2 or 3	R
51	2	51	2	1 or 3	R
02	2	52	2	1 or 3	R
01	1	33	3	2	C
52	2	46	1	3	C
07	1	39	3	2	C
48	1	85	2	3	C

Note: Underlining indicates numbers used. Zeros are omitted; numbers 1, 4, 7 = 1; 2, 5, 8 = 2; 3, 6, 9 = 3

Examples of basic problems in probability

A Poker Problem: One Pair (Two of a Kind)

What is the chance that the first five cards chosen from a deck of 52 (bridge/poker) cards will contain two (and only two) cards of the same denomination (two 3's for example)? (Please forgive the rather sterile unrealistic problems in this and the other chapters on probability. They reflect the literature in the field for 300 years. We'll get more realistic in the statistics chapters.)

We shall estimate the odds the way that gamblers have estimated gambling odds for thousands of years. First, check that the deck is not a pinochle deck and is not missing any cards. (Overlooking such small but crucial matters often leads to errors in science.) Shuffle thoroughly until you are satisfied that the cards are randomly distributed. (It is surprisingly hard to shuffle well.) Then deal five cards, and mark down whether the hand does or does not contain a pair of the same denomination. At this point, we must decide whether three of a kind, four of a kind or two pairs meet our criterion for a pair. Since our criterion is "two and only two," we decide *not* to count them.

Then replace the five cards in the deck, shuffle, and deal again. Again mark down whether the hand contains one pair of the same denomination. Do this many times. Then count the number of hands with one pair, and figure the proportion (as a percentage) of all hands. In one series of 100 experiments, 44 percent of the hands contained one pair, and therefore .44 is our estimate (for the time being) of the probability that one pair will turn up in a poker hand. But we must notice that this estimate is based on only 100 hands, and therefore might well be fairly far off the mark (as we shall soon see).

This experimental "resampling" estimation does not require a deck of cards. For example, one might create a 52-sided die, one side for each card in the deck, and roll it five times to get a "hand." But note one important part of the procedure: No single "card" is allowed to come up twice in the same set of five spins, just as no single card can turn up twice or more in the same hand. If the same "card" did turn up twice or more in a dice experiment, one could pretend that the roll had never taken place; this procedure is necessary to make the dice experiment analogous to the actual card-dealing situation under investigation. Otherwise, the results will be slightly in error. This type of sampling is known as "sampling without replacement," because each card is *not replaced* in the deck prior to dealing the next card (that is, prior to the end of the hand).

Table 6-2

Results of 100 Trials for the Problem "OnePair"

Trial	1	2	3	4	5	6	7	8	9	10	11	12	13
Results	Y	Y	N	N	Y	Y	N	N	Y	N	N	Y	Y
Trial	14	15	16	17	18	19	20	21	22	23	24	25	26
Results	N	Y	Y	Y	Y	Y	N	N	Y	N	Y	N	Y
Trial	27	28	29	30	31	32	33	34	35	36	37	38	39
Results	N	Y	N	Y	Y	N	Y	N	N	N	N	Y	N
Trial	40	41	42	43	44	45	46	47	48	49	50		
Results	N	N	N	N	Y	Y	Y	N	N	Y	N		

Subtotal: 23 Yes, 27 No = 46%

Trial	51	52	53	54	55	56	57	58	59	60	61	62	63
Results	N	Y	N	N	Y	N	Y	Y	N	N	N	Y	Y
Trial	64	65	66	67	68	69	70	71	72	73	74	75	76
Results	Y	N	N	Y	N	N	N	N	Y	N	Y	N	N
Trial	77	78	79	80	81	82	83	84	85	86	87	88	89
Results	N	N	N	N	Y	N	N	N	Y	Y	N	Y	N
Trial	90	91	92	93	94	95	96	97	98	99	100		
Results	Y	Y	N	N	Y	Y	Y	Y	N	Y	N		

Subtotal: 21 Yes, 29 No = 42%
Total: 44 Yes, 56 No = 44%

Still another resampling method uses a *random number table*, such as that which is shown in Table 6-3. Arbitrarily designate the spades as numbers “01-13,” the diamonds as “14-26,” the hearts as “27-39,” and the clubs as “40-52.” Then proceed across a row (or down a column), writing down each successive pair of digits, excluding pairs outside “01-52” and omitting duplication within sets of five numbers. Then translate them back into cards, and see how many “hands” of five “cards” contain one pair each. Table 6-4 shows six such hands, of which hands numbered 2, 3 and 6 contain pairs.

Table 6-3
Table of Random Digits

48	52	78	38	11	90	41	83	43	99	51	55	57	03	83	20
15	11	84	33	09	24	08	52	42	70	37	16	66	73	15	54
25	89	70	11	91	65	41	90	88	04	30	72	15	81	34	46
34	24	66	55	67	79	29	18	36	56	96	95	35	06	05	10
37	27	58	38	23	84	94	39	99	50	74	80	41	85	98	63
12	17	04	68	19	98	53	44	16	32	91	01	71	60	19	12
88	85	44	65	52	01	99	56	72	07	96	39	56	34	86	01
81	92	77	83	10	58	92	33	63	48	62	66	32	61	59	74
08	50	15	18	13	45	65	12	32	92	53	82	07	61	71	80
84	29	90	36	05	95	20	71	17	82	83	38	01	87	74	92
77	76	46	28	47	15	04	21	04	75	51	83	91	37	14	32
01	33	90	94	86	10	03	99	95	98	76	97	97	26	45	62

Table 6-4
Six Simulated Trials for the Problem "OnePair"

	Aces	Deuces	3	4	5	6	7	8	9	10	J	Q	K
	What the Random Numbers Mean												
Spades	01	02	03	04	05	06	07	08	09	10	11	12	13
Diamonds	14	15	16	17	18	19	20	21	22	23	24	25	26
Hearts	27	28	29	30	31	32	33	34	35	36	37	38	39
Clubs	40	41	42	43	44	45	46	47	48	49	50	51	52

Simulation Results

Hand 1:	48	52	38	11	41	no pairs
Hand 2:	15	11	33	09	24	one pair
Hand 3:	25	11	41	04	30	one pair
Hand 4:	34	24	29	18	36	no pairs
Hand 5:	37	27	38	23	39	no pairs
Hand 6:	12	17	04	19	44	one pair

Now let's do the same job using RESAMPLING STATS on the computer. Let's name "One Pair" the file which simulates a deck of playing cards and solves the problem.

Our first task is to simulate a deck of playing cards analogous to the real cards we used previously. We don't need to simulate all the features of a deck, but only the features that matter for the problem at hand. We require a deck with four "1"s, four "2"s, etc., up to four "13"s. The suits don't matter for our present purposes. Therefore, with the URN command we join together in a single array the four sets of thirteen numbers, to represent the 13 denominations.

At this point we have a complete deck in location A. But that "deck" is in the same order as a new deck of cards. If we do not shuffle the deck, the results will be predictable. Therefore, we write SHUFFLE a b and then deal a poker hand by taking the first five cards from the shuffled hand, using the TAKE statement. Now we must find out if there is one (and only one) pair; we do this with the MULTIPLES statement—the "2" in that statement indicates that it is a duplicate, rather than a singleton or triplicate or quadruplicate that we are testing for—and we put the result in location D. Next we SCORE in location z how many pairs there are, the number in each trial being either zero, one, or two. (The reason we cannot put the result of the MULTIPLES operation directly into the scorecard vector z is that only the SCORE command accumulates results from trial to trial rather than over-writing the result of the past trial with the current one.) And with that we end a single trial.

With the REPEAT 1000 statement and the END statement, we command the program to repeat a thousand times the statements in the "loop" between those two lines. When those 1000 repetitions are over, the computer moves on to COUNT the number of "1's" in SCOREkeeping vector z, each "1" indicating a hand with a pair. And we then PRINT to the screen the result which is found in location k. If we want the *proportion* of the trials in which a pair occurs, we simply divide the results of the thousand trials by 1000.

URN 4#1 4#2 4#3 4#4 4#5 4#6 4#7 4#8 4#9 4#10 4#11 4#12 4#13 a
Create an urn (vector) called a with four "1's," four "2's," four "3's," etc., to represent a deck of cards

REPEAT 1000
Repeat the following steps 1000 times

> **SHUFFLE a b**
> Shuffle the deck
>
> **TAKE b 1,5 c**
> Take the first five cards
>
> **MULTIPLES c =2 d**
> How many pairs?
>
> **SCORE d z**
> Keep score of # of pairs

END
End loop, go back and repeat

COUNT z =1 k
How often 1 pair?

DIVIDE k 1000 kk
Convert to proportion

PRINT kk

Note: The file "onepair" on the Resampling Stats software disk contains this set of commands.

In one run of the program, the result in kk was .419, so our estimate would be that the probability of a single pair is .419.

How accurate are these resampling estimates? The accuracy depends on the *number of hands* we deal—the more hands, the greater the accuracy. If we were to examine millions of hands, 42 percent would contain a pair each; that is, the chance of getting a pair in the long run is 42 percent. The estimate of 44 percent based on 100 hands in Table 6-2 is fairly close to the long-run estimate, though whether or not it is close *enough* depends on one's needs of course. If you need great accuracy, deal many more hands.

A note on the "a"s, "b"s, "c"s in the above program, etc.: These "variables" are called "vectors" in RESAMPLING STATS. A *vector* is an array of elements that gets filled with numbers as RESAMPLING STATS conducts its operations. When RESAMPLING STATS completes a trial these vectors are generally wiped clean except for the "SCORE" vector (here labeled "z") which keeps track of the result of each trial.

To help keep things straight (though the program does not require it), we usually use "z" to name the vector that collects all the trial results, and "k" to denote our overall summary results. Or you could call it something like "scrbrd."

How many trials (hands) should be made for the estimate? There is no easy answer[1]. One useful device is to run several (perhaps ten) equal sized sets of trials, and then examine whether the proportion of pairs found in the entire group of trials is very different from the proportions found in the various subgroup sets. If the proportions of pairs in the various subgroups differ greatly from one another or from the overall proportion, then keep running additional larger subgroups of trials until the variation from one subgroup to another is sufficiently small for your purposes. While such a procedure would be impractical using a deck of cards or any other physical means, it requires little effort with the computer and RESAMPLING STATS.

Another Introductory Poker Problem

Which is more likely, a poker hand with two pairs, or a hand with three of a kind? This is a *comparison* problem, rather than a problem in *absolute* estimation as was the previous example.

In a series of 100 "hands" that were "dealt" using random numbers, four hands contained two pairs, and two hands contained three of a kind. Is it safe to say, on the basis of these 100 hands, that hands with two pairs are more frequent than hands with three of a kind? To check, we deal another 300 hands. Among them we see fifteen hands with two pairs (3.75 percent) and eight hands with three of a kind (2 percent), for a total of nineteen to ten. Although the difference is not enormous, it is reasonably clear-cut. Another 400 hands might be advisable, but we shall not bother.

Earlier I obtained forty-four hands with *one* pair each out of 100 hands, which makes it quite plain that *one* pair is more frequent than *either* two pairs or three-of-a-kind. Obviously, we need *more* hands to compare the odds in favor of two pairs with the odds in favor of three-of-a-kind than to compare those for one pair with those for either two pairs or three-of-a-kind. Why? Because the difference in odds between one pair, and either two pairs or three-of-a-kind, is much greater than the difference in odds between two pairs and three-of-a-kind. This observation leads to a general rule: The closer the odds between two events, the *more trials* are needed to determine which has the higher odds.

Again it is interesting to compare the odds with the formulaic mathematical computations, which are 1 in 21 (4.75 percent) for a hand containing two pairs and 1 in 47 (2.1 percent) for a hand containing three-of-a-kind—not too far from the estimates of .0375 and .02 derived from simulation.

To handle the problem with the aid of the computer, we simply need to estimate the proportion of hands having triplicates and the proportion of hands with two pairs, and compare those estimates.

To estimate the hands with three-of-a-kind, we can use a program just like "One Pair" earlier, except instructing the MULTIPLES statement to search for triplicates instead of duplicates. The program, then, is:

URN 4#1 4#2 4#3 4#4 4#5 4#6 4#7 4#8 4#9 4#10 4#11 4#12 4#13 a
Create an urn (vector) called a with four "1"'s, four "2"'s, four "3"'s, etc., to represent a deck of cards

REPEAT 1000
Repeat the following steps 1000 times

> **SHUFFLE a b**
> Shuffle the deck
>
> **TAKE b 1,5 c**
> Take the first five cards

MULTIPLES c =3 d
How many triplicates?

SCORE d z
Keep score of # of triplicates

END
End loop, go back and repeat

COUNT z =1 k
How often 1 triplicate?

DIVIDE k 1000 kk
Convert to proportion

PRINT kk

Note: The file "3kind" on the Resampling Stats software disk contains this set of commands.

To estimate the probability of getting a two-pair hand, we revert to the original program (counting pairs), except that we examine all the results in SCOREkeeping vector z for hands in which we had *two* pairs, instead of *one*.

URN 4#1 4#2 4#3 4#4 4#5 4#6 4#7 4#8 4#9 4#10 4#11 4#12 4#13 a
Join together in an array (vector) called "a" four "1's," four "2's," four "3's," etc., to represent a deck of cards

REPEAT 1000
Repeat the following steps 1000 times

SHUFFLE a b
Shuffle the deck

TAKE b 1,5 c
Take the first five cards

MULTIPLES c =2 d
How many pairs?

SCORE d z
Keep score of # of pairs

END
End loop, go back and repeat

COUNT z =2 k
How often 2 pairs?

DIVIDE k 1000 kk
Convert to proportion

PRINT kk

Note: The file "2pair" on the Resampling Stats software disk contains this set of commands.

For efficiency (though efficiency really is not important here because the computer performs its operations so cheaply) we could develop both estimates in a single program by simply generating 1000 hands, and count the number with three-of-a-kind and the number with two pairs.

Before we leave the poker problems, we note a difficulty with Monte Carlo simulation. The probability of a royal flush is so low (about one in half a million) that it would take much computer time to compute. On the other hand, considerable inaccuracy is of little matter. Should one care whether the probability of a royal flush is 1/100,000 or 1/500,000?

The concepts of replacement and non-replacement

In the poker example above, we *did not replace* the first card we drew. If we were to replace the card, it would leave the probability the same before the second pick as before the first pick. That is, the conditional probability remains the same. *If we replace, conditions do not change.* But if we do not replace the item drawn, the probability changes from one moment to the next. (Perhaps refresh your mind with the examples in the discussion of conditional probability in Chapter 5.)

If we sample with replacement, the sample drawings remain *independent* of each other—a topic addressed in Chapter 5.

In many cases, a key decision in modeling the situation in which we are interested is whether to sample with or without replacement. The choice must depend on the characteristics of the situation.

There is a close connection between the lack of finiteness of the concept of universe in a given situation, and sampling with replacement. That is, when the universe (population) we have in mind is not small, or has no conceptual bounds at all, then the probability of each successive observation remains the same, and this is modeled by sampling with replacement. ("Not finite" is a less expansive term than "infinite," though one might regard them as synonymous.)

Chapter 7 discusses problems whose appropriate concept of a universe is finite, whereas Chapter 8 discusses problems whose appropriate concept of a universe is not finite. This general procedure will be discussed several times, with examples included.

Endnotes

1. One simple rule-of-thumb is to quadruple the original number. The reason for quadrupling is that four times as many iterations (trials) of this resampling procedure give *twice* as much accuracy (as measured by the standard deviation, the most frequent measurement of accuracy). That is, the error decreases with the square root of the number of iterations. If you see that you need *much* more accuracy, then *immediately* increase the number of iterations even more than four times—perhaps ten or a hundred times.