Two small examples worked out in full

1. The "flush" vs. "flush given ace" problem.

Suppose that instead of a five-card hand out of a 52-card deck we have a two-card hand out of a nine-card deck: A 2 3 A 2 3 A 2 3. A flush in this case is two cards of matching color.

In the full deck, there are 9 flushes: A/2, A/3, 2/3 in each of the three colors. There are $\binom{9}{2} = 36$ hands total, so the probability of getting a flush is 9/36, or 1/4.

If we restrict only to hands containing at least one ace, we drop down to 6 flushes: A/2, A/3 in each of the three colors. However, there are now a lot fewer total hands: 3 hands where both cards are aces, and $3 \cdot 6 = 18$ where one is an ace and one is not. The probability of a flush in this case is $6/21 = 28.6\%$, which is larger than 1/4. We've kept the A/2 and A/3 flushes and the A/2, A/3, and A/A non-flushes. We've lost only the 2/3 flushes but all of the 2/3, 2/2, and 3/3 non-flushes.

The confusion comes in when you assign *causality* to the probability difference. In a given game of poker, a player either has a flush or doesn't, and your knowledge of whether or not that player has an ace does not change things. However, taking hands as a whole (alternatively, thinking about trends over many games of poker), there is a higher percentage of flushes among the ace-containing hands than among the hands as a whole.

2. An "at least one" problem.

Why can you not typically do an "at least one" problem as a single computation? Consider the alphabet $\{a, b, c\}$ and the problem of counting the 3-letter sequences (repeats allowed) with at least one a. There are 27 such sequences, so we can list them out. Let's categorize them by how many a's they contain.

0 a's		1 a			2 a's		3 a's
bbb	cbb	abb	bab	bba	aab	baa	aaa
bbc	cbc	abc	bac	bca	aac	caa	
bcb	ccb	acb	cab	cba	aba		
bcc	ccc	acc	cac	cca	aca		

Of the 27 total sequences, 8 contain no a's and the remaining 19 contain at least one. What goes wrong if we try to count the "at least one" sequences all in one product?

The sensible approach might be to say "we have to have an a somewhere, so that's 3 options; then the rest can be anything, so $3 \cdot 3$, for a total of $3 \cdot 3 \cdot 3$ sequences." We've counted everything! Except that in our intended counting, it's not that we're counting everything, it's that we're counting some of our intended sequences multiple times.

The sequences with one a each get counted once. The sequences with 2 a's, however, get counted twice, and most egregiously, the sequence aaa is counted three times: "put the a in the first slot and fill the other two in with a's," "put the a in the second slot and fill the other two in with a's," and "put the a in the third slot and fill the other two in with a's." Whatever slots did not get the "special" a can get a's *anyway* when we fill in the rest.

Can later tasks accomplish the same thing as earlier ones? If so, you are almost certainly overcounting. Take care!