

4.4 A mathematical model for card shuffling

Card shuffling provides almost the only instance of real-world mixing where mathematical probability provides a prediction for how much mixing is required. The most common method of shuffling is the *riffle shuffle*. The deck is divided into two roughly equal halves, held in each hand with thumbs inward; moving thumbs upward releases in rapid succession the bottom cards of each half-deck, positioned so they slightly interleave and can finally be pushed together to reassemble the shuffled deck.

When first learning this shuffle, a beginner will tend to drop a large packet of cards at once from a hand. An expert will shuffle smoothly, so that typically one card is dropped at a time, and the drops alternate between left and right hands. Intuition says the latter should be more “efficient” at mixing the deck. But consider a “perfect shuffle” in which single drops alternate precisely between hands; in this case there’s no randomness at all! And indeed it turns out that after 52 or 8 (depending on which half-deck starts dropping first) perfect shuffles, the deck would back in its original order (though only a few stage magicians can actually carry this out).

The simplest mathematical model (for a single *random* riffle shuffle) corresponds roughly to an intermediate skill level. When dividing the deck, suppose a Binomial(52, 1/2) number of cards go to the left hand. And suppose that at each stage, the chance that the next drop is from the left or right hand is proportional to the number of cards remaining in that hand. This is called the GSR model (xxx see Notes).

xxx picture of one shuffle (abcde and 00000 and $\alpha\beta\dots$)

Let’s now think what exactly is it about card shuffling that we want to understand. Probability calculations involving dealing cards (e.g. the chances of being dealt particular poker or bridge hands) assume the deck is uniformly random – all 52! orderings are equally likely. But suppose we start with a deck in known order, e.g. a brand new deck, and then do k random shuffles (from the GSR model of riffle shuffles, or some other model of some physically different shuffle). Then after the k shuffles, the deck won’t have exactly uniform distribution, but we can hope its distribution is “approximately uniform” in some sense. It turns out that classical theory (of convergence of finite-state Markov chains – see e.g. [43]) tells us that as $k \rightarrow \infty$ the distribution gets closer and closer to the uniform distribution (for any reasonable method of shuffling). But for a particular method, it doesn’t tell us whether 3 or 3 million shuffles are needed to get close. Answering that question requires a different calculation in each model. For the GSR model, the calculation was done in [3] where the following table is given.

number of shuffles	k	3	4	5	6	7	8	9
non-uniformity	d(k)	1.000	1.000	0.924	0.614	0.334	0.167	0.085

Table xxx. Variation distance to uniformity in the GSR model of riffle shuffling [3].

To explain the numbers, consider an event A associated with a deck of cards. The event has some probability $P(A)$ if the deck is in uniform random order, and some probability $P_k(A)$ if the deck was obtained via k random (GSR model) riffle shuffles of a new deck. So treating the deck as “completely random” is an error of magnitude $|P_k(A) - P(A)|$. The *variation distance* is defined as the “worst case” (over events A) of this error:

$$d(k) = \max_A |P_k(A) - P(A)|.$$

Note that there are $52!$ possible orderings of a deck, and since an event A is a subset of the set of orderings, the number of events A being considered equals $2^{52!}$. So [3] need some math theory (rather than brute force computation) to get a fairly simple formula for $d(k)$.

This result has entered the popular science literature under phrases like

It takes 7 shuffles to mix a deck of cards

but this is an oversimplification in several ways:

- The model isn’t completely realistic.
- The number 7 arises as the smallest k making $d(k) < 1/2$. Because $d(7)$ is (about) $1/3$, there are events that for a truly random deck have chance $2/3$ but for a 7-times-shuffled new deck have chance $1/3$, so this doesn’t quite correspond to our intuitive notion of “well-mixed”.
- Conversely, this error applies only to very special events. For more typical events A relevant to real card games, one might reasonably expect 3 or 4 shuffles to be enough (xxx data?).

Despite these defects with the answer “7”, no-one has come up with any more satisfactory answer to the (vague) question “how many shuffles to mix a deck of cards?”

xxx rising sequences, magic trick – later

4.4.1 A little math analysis

Rather than attempting to explain the exact formula leading to the Table xxx numbers, let me outline a simpler to explain an *inequality*

$$d(k) \leq P(\text{52 random } k\text{-bit numbers are not all different}). \quad (4.5)$$

Here a 7-bit number is like 0110011, with uniform random choices of 0 or 1, and we are asking whether amongst 52 such numbers there are any two which are exactly equal.

xxx tie up with birthday problem; prob all different $\approx \exp(-m^2/(2N))$. Here $m = 52$ and $N = 2^k$ so the bound in (4.5) becomes approximately $\exp(-2^{10.4-k})$ which starts getting small at $k = 12$.

xxx so

The argument is interesting (to mathematicians, anyway) because it makes a connection with an apparently quite different topic, *sorting*. In Figure xxx, writing a 0 on each card in the left pile and a 1 on each card in the right pile, the shuffled deck has a xxx 01xxxxxx which records the particular xxx. It turns out that in the GSR model, every possible sequence of 0s and 1s is equally likely. This allows us to consider (for purposes of doing the math analysis) the notion of a *reversed shuffle*. In Figure xxx, start with cards $\alpha\beta\dots$ xxx, randomly attach a label 0 or 1 to each card, collect the “0” cards into a left pile and the “1” cards into a right pile, preserving relative order, and finally out the left pile on top of the right pile. This constitutes one reversed shuffle.

xxx draw figure

Figure xxx shows a sequence of four reversed shuffles. Instead of generating random bits on the fly we can imagine the four random bits on each card are generated at the start; the first reversed shuffle uses the rightmost bit, the second uses the second-rightmost bit, and so on.

Now regard the four bits as the binary representation of a integer. In the left deck of Figure xxx the integers are in haphazard order (from top to bottom: 9, 4, 11, 2, 13). But in the right deck they have been sorted into increasing order (2, 4, 9, 11, 13). (xxx radix sorting).

xxx continue; reversal is riffle shuffle; if distinct then random.