Minilab 7b Worksheet

Simulation

One way to get a better understanding of probability is through simulation of processes involving randomness. In this minilab, we have some fun with rolling dice.

1. Rolling Dice by Simulation

We can simulate rolling of a dice using the *discrete uniform distribution* in R. In particular we use the `rdunif()` function (notice the d in rdunif).

(1) Suppose we are given a 12-sided dice in the shape of a dodecahedron (one of the five "Platonic solids"). If it was a fair dice then all outcomes {1,2,3,4,5,6,7,8,9,10,11,12} would be equally likely. If we suspect that the dice was unfair, then we could investigate this by rolling it a lot of times and checking how many times each number comes up on the dice.

```
library(tidyverse)
# Simulation of 12-sided dice rolls
n = 120
rolls = rdunif(n,1,12)
table(rolls)
```

Here `rdunif(n,1,12)` creates a vector with n elements where each element is randomly chosen from {1,2,3,4,5,6,7,8,9,10,11,12} with equal probability. So each element of the vector represent one roll of the 12-sided dice.

The `table` function gives the frequency (count) of each dice face value observed. You should see something similar to the output below (not exactly as some randomness is involved). From 120 rolls of the 12-sided dice, we would expect to see each face turn up 120/12=10 times. Due to random variation, we can see some counts under and some counts over.

```
> table(rolls)
rolls
 1  2  3  4  5  6  7  8  9 10 11 12 
 6  7 12  6 10 14 13  7 15 12 10  8 
```

It is then easy to plot the corresponding barchart, but note that the labels on the horizontal axes are not very helpful.

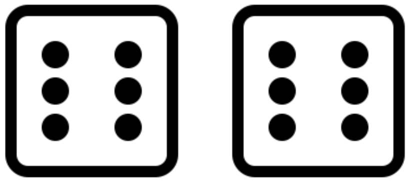
```
mydata = tibble(rolls)
ggplot(mydata,mapping=aes(x=rolls)) +
  geom_bar()
```

Exercise. Consider rolling one 6-sided dice (call it dice A) and then rolling another 6-sided dice (call it dice B) and taking the sum of the values.

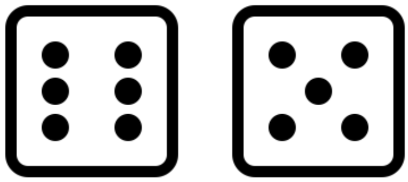
```
n = 36000
rollsA = rdunif(n,1,6)
rollsB = rdunif(n,1,6)
rolls_sum = rollsA + rollsB
table(rolls_sum)
```

Plot a barchart showing the count for each possible value of the sum of the two dice. What counts are you expecting to see (approximately) when n=36000?

Exercise. Consider rolling a fair 6-sided dice until a 6 turns up twice in a row.



Do you think this will take more, the same, or fewer rolls than rolling a fair 6-sided dice until a 6 turns up followed immediately by a 5?



We can use R to simulate these two processes (separately) over 10000 repeated trials and compare the means. Two pieces of R code are provided on Aula:

- expt_double_six.R
- expt_six_five.R

After you have run these, what conclusion do you make?

Exercise. Another application of dice rolling is the classic board game *Snakes and* Ladders. The R code provided on Aula (expt_snakes_ladders.R) implements the particular board below. Run the code to estimate the probability that the player to move first will win. Investigate the head start that the second player would need in order to have a higher probability of winning that the first player (change the initial value of *locationB*).

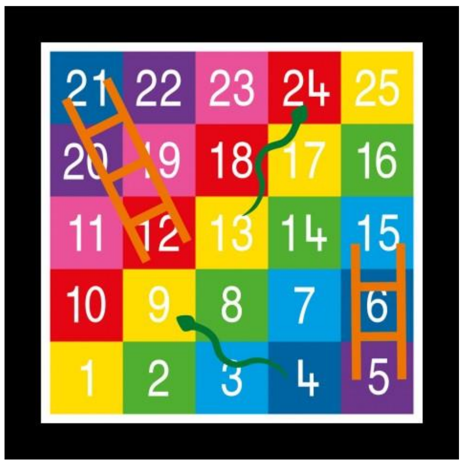


Image from: https://school-playground-markings.co.uk/blox/snakes-and-ladders-1-25/

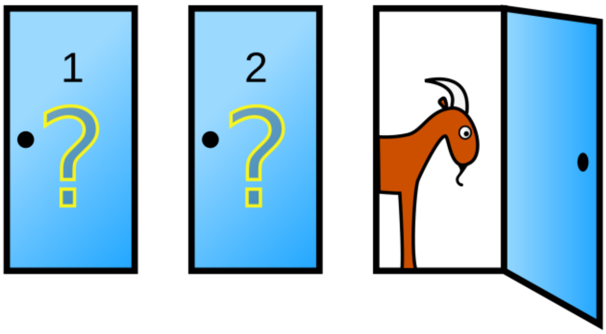
2. Just for Interest (Optional)

If you have a bit more time and are feeling a bit bored …

(1) Checkout the R package tidydice (use install.packages() as usual).

https://cran.r-project.org/web/packages/tidydice/vignettes/tidydice.html

(2) A classic problem in probability is the Monty Hall Problem. Read the description at https://en.wikipedia.org/wiki/Monty_Hall_problem to see what this is all about. There have been many disagreements about how to solve this problem.



Run the R code provided on Aula (expt_montyhall.R). If you follow the "don't switch" strategy, what proportion of wins to you observe (from 10000 trials)? What happens if you follow the "switch" strategy? What conclusion do you make?

A good explanation of the Monty Hall Problem involves a probability tree. Have a look at the YouTube clip: https://www.youtube.com/watch?v=cphYs1bCeDs

Summary

In this minilab, we have looked at some probability questions that can be investigated by rolling dice or equivalently by simulating the rolling of dice within a computer.