Orchard Code

Sumitra Badrinathan

10/15/2018

The code below shows you how to simulate 100,000 games of Orchard. Over a large number of games, the probability of winning when the crow is allowed to move 4 steps and when your basket strategy is to pick the maximum fruit is $\approx 46\%$

```
numgames <- 100000
numwon <- 0
for (i in 1:numgames) {

  apples <- 4 # define all fruits globally in for loop
  lemons <- 4 # doing this will reset the values to 4 every time you start a new game
  pears <- 4
  plums <- 4
  crow <- 4
  outcome <- 0

  while(outcome == 0) { # while loops are used as stopping rules

    temp <- runif(1, min = 0, max = 1)
    # Numbers between 0 and 1/6 signify that you rolled a apple
    if (temp >= 0 & temp <= 1/6) {
      apples <- apples - 1
    }
    # Numbers between 1/6 and 2/6 signify that you rolled a lemon
    if (temp > 1/6 & temp <= 2/6) {
      lemons <- lemons - 1
    }
    # Numbers between 2/6 and 3/6 signify that you rolled a pear
    if (temp > 2/6 & temp <= 3/6) {
      pears <- pears - 1
    }
    # Numbers between 3/6 and 4/6 signify that you rolled a plum
    if (temp > 3/6 & temp <= 4/6) {
      plums <- plums - 1
    }
    # Numbers between 4/6 and 5/6 signify that you rolled a crow
    if (temp > 4/6 & temp <= 5/6) {
      crow <- crow - 1
    }
    # Numbers between 5/6 and 1 signify that you rolled a basket
    if (temp > 5/6 & temp <= 1) {

      # when you roll a basket, you pick out the fruit that has the maximum
      temp2 <- which.max(c(apples, lemons, pears, plums))
      if (temp2 == 1) {
        apples <- apples - 1
      }
```

```
      if (temp2 == 2) {
        lemons <- lemons - 1
      }
      if (temp2 == 3) {
        pears <- pears - 1
      }
      if (temp2 == 4) {
        plums <- plums - 1
      }
    }
    # two ways of the game ending as below:
    mostfruit <- max(apples, lemons, pears, plums)
    if (mostfruit == 0) {
      outcome <- 1 # you win
    }
    if (crow == 0) {
      outcome <- 2 # you lose
    }
  }
  numwon <- numwon + (outcome == 1) # count # of wins

}
output <- paste("We won", numwon, "out of the", numgames, " games that we simulated" )
print(output)
```

```
## [1] "We won 46102 out of the 1e+05  games that we simulated"
```

This is not the only way to simulate this game! Below I will show you a way to do it with fewer lines of code. The main difference is that I am using the `sample()` function to approximate a die roll instead of `runif`. You can change the basket strategy to pick the minimum fruit (using `which.min()`) or to pick a random fruit (using `sample()`).

```
numgames <- 100000
outcome <- numeric(numgames)
crow_win <- c()
for (i in 1:numgames){
  no_of_fruits <- c(4,4,4,4,4)
  count = 0 #add
  while (outcome[i] == 0){
    count = count + 1
    side_select = sample(1:6,1)
    if (side_select<6){
      no_of_fruits[side_select] = no_of_fruits[side_select] - 1
    } else {
      idx =  which.max(no_of_fruits[1:4])
      #idx = sample(1:4,1)
      #idx = which.max(no_of_fruits[1:4])
      no_of_fruits[idx] = no_of_fruits[idx] - 1
    }
    mostfruit <- max(no_of_fruits[1:4])
    if (mostfruit == 0) {
      outcome[i] <- 1
    }
```

```
    if (no_of_fruits[5] == 0) {
      outcome[i] <- -1
    }
  }
}
prop.table(table(outcome))
```

```
## outcome
##      -1       1
## 0.54105 0.45895
```