Problem set #2: Monte Carlo

Milica Cudina

2021-12-21

Problem #1: Simple gambling

Consider a game in which a spinner is spun. The spinner is equally likely to land on *yellow*, *red*, or *blue*. If the spinner lands on *yellow*, the player loses \$1. If the spinner lands on *red*, the player wins \$1. If the spinner lands on *blue*, no money changes hands.

Use *Monte Carlo* to estimate the expected winnings based on 10,000 runs. Does the result agree with your theoretical expected value?

Now, complete 1,000 batches of 10,000 runs. What does the histogram of the averages of the 1,000 batches look like? What is their mean? What is their variance? What is their standard deviation?

Solution: Let's encode *yellow* as 1, *red* as -1, and *blue* as 0. Here is the result of a single run:

```
spinner=c(-1,0,1)
one.run=sample(spinner,1)
one.run
```

```
## [1] 0
```

If I want many runs, I will simply increase the number of simulated values above.

```
n.sim=10000
many.runs=sample(spinner,n.sim, replace=TRUE)
```

Finally, I calculate the mean of the vector or runs.

```
mean(many.runs)
```

```
## [1] 0.0078
```

The number is close to zero, so I am happy with how it turned out.

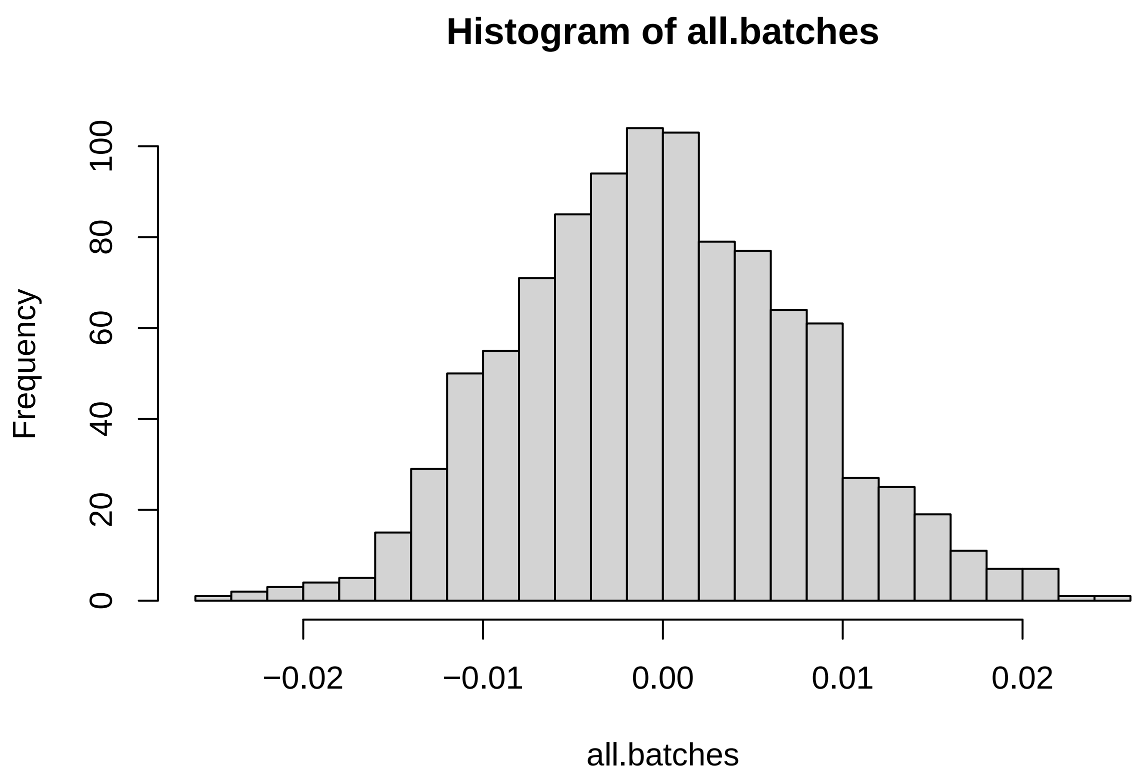
Now, I create a function which does exactly what we did above, i.e., performs $10,000$ runs of the game and calculates average winnings.

```
spin.mc<-function(){
  mean(sample(spinner,n.sim,replace=TRUE))
}
spin.mc()
```

```
## [1] -0.0219
```

Next, I repeat the same function $1,000$ times and record all the mean winnings in a vector.

```r
n.batches=1000
all.batches=replicate(n.batches,spin.mc())
hist(all.batches, breaks=25)
```




```r
mean(all.batches)
```

```
## [1] -7.28e-05
```

```r
var(all.batches)
```

```
## [1] 6.368805e-05
```

```r
sd(all.batches)
```

```
## [1] 0.007980479
```

Problem #2: Gambling

Consider the following game:

- if a fair die lands on an odd number, the player loses one dollar;
- if a fair die lands on an even number which is not a perfect square, no money changes hands;
- if a fair die lands on an even number which is also a perfect square, the player wins nine dollars.

What is the theoretical expectation of the amount won in this game?

Use *Monte Carlo* to estimate the expected winnings based on 10,000 runs. Does the result agree with your theoretical expected value?

Solution: The probability of losing one dollar is 1/2. The probability of winning nine dollars is 1/6. The probability of no money changing hands is 1/3. So, this is the expected amount won:

```
p.loss<-1/2
p.win<-1/6
p.no<-1/3
loss.amt<-(-1)
win.amt<-9
exp.game<-p.loss*loss.amt+p.win*win.amt
exp.game
```

```
## [1] 1
```

Using *Monte Carlo*, we have the following:

```
amounts<-c(loss.amt,0,win.amt)
probs<-c(p.loss, p.no, p.win)
mean(sample(amounts, n.sim, replace=TRUE, prob=probs))
```

```
## [1] 1.005
```

Problem #3: The binomial distribution

Consider a random variable X which has the binomial distribution with the number of trials equal to 50 and the probability of success in every trial equal to 2/5.

What is the expected value of this random variable?

Use *Monte Carlo* to estimate the expected value based on 10,000 runs. Does the result agree with your theoretical expected value?

Solution: Here are the parameters of the random variable X:

```
n.trials=50
prob.s<-2/5
```

So, the expected value of the random variable X equals:

```
exp.X<-n.trials*prob.s
exp.X
```

```
## [1] 20
```

Using *Monte Carlo*, we have

```
runs<-rbinom(n.sim, size=n.trials, prob=prob.s)
mean(runs)
```

```
## [1] 20.0449
```