随机模拟方法与应用导论作业八

陈稼霖 *45875852*

2019-11-9

11.5(Collecting state quarters).

In 1999, the United States launched the 50 State Quarters program where each of the 50 states was honored with a special quarter. Suppose you purchase 100 "state" quarters where each quarter is equally likely to feature one of the 50 states.

a. Write a function using the sample function to simulate the purchase of 100 quarters and record the number of unique quarters that are purchased.
b. Using the `replicate` function, repeat this process for 1000 purchases. Construct a table of the number of unique quarters you obtain in these 1000 simulations. Use this table to estimate the probability that you obtain at least 45 unique quarters.
c. Use the output from part (b) to find the expected number of unique quarters.
d. Suppose you are able to complete your quarter set by purchasing state quarters from a coin shop for $2 for each quarter. Revise your function to compute the total (random) cost of completing the quarter set. Using the `replicate` function, repeat the quarter-purchasing process 1000 times and compute the expected cost of completing your set.

a. 定义函数`purchase`来模拟买100个25美分硬币，返回买到的非重复的硬币的数量，并用其进行一次模拟。

```
purchase = function(n = 100){
  length(unique(sample(1:50,size = n,replace = TRUE)))
}
```

b. 用函数`replicate`重复模拟1000次，并将结果储存在变量`N`中。

```
N = replicate(1000,purchase())
```

用`N`中的数据估算获得至少45枚不重复硬币的概率。

```
sum(N >= 45) / length(N)
```

```
## [1] 0.299
```

故获得至少45枚不重复硬币的概率约为0.3。

c. 用`N`中的数据估算获得不重复硬币数量的期望值

```
mean(N)
```

```
## [1] 43.427
```

故获得不重复硬币数量的期望值约为43。

d. 定义函数purchase2，计算获得包含所有硬币的集合所需的支出。其中，N1是通过随机购买得到的非重复硬币数，N2是随机购买后尚未获得的硬币数。

```
purchase2 = function(n = 100){
  N1 = length(unique(sample(1:50,size = n,replace = TRUE)))
  N2 = 50 - N1
  0.25 * n + 2 * N2
}
```

若每次随机购买都买100个硬币：定义函数expected.cost，调用函数replicate重复模拟1000次，计算获得包含所有硬币的集合所需的支出的期望值，并进行一次模拟。

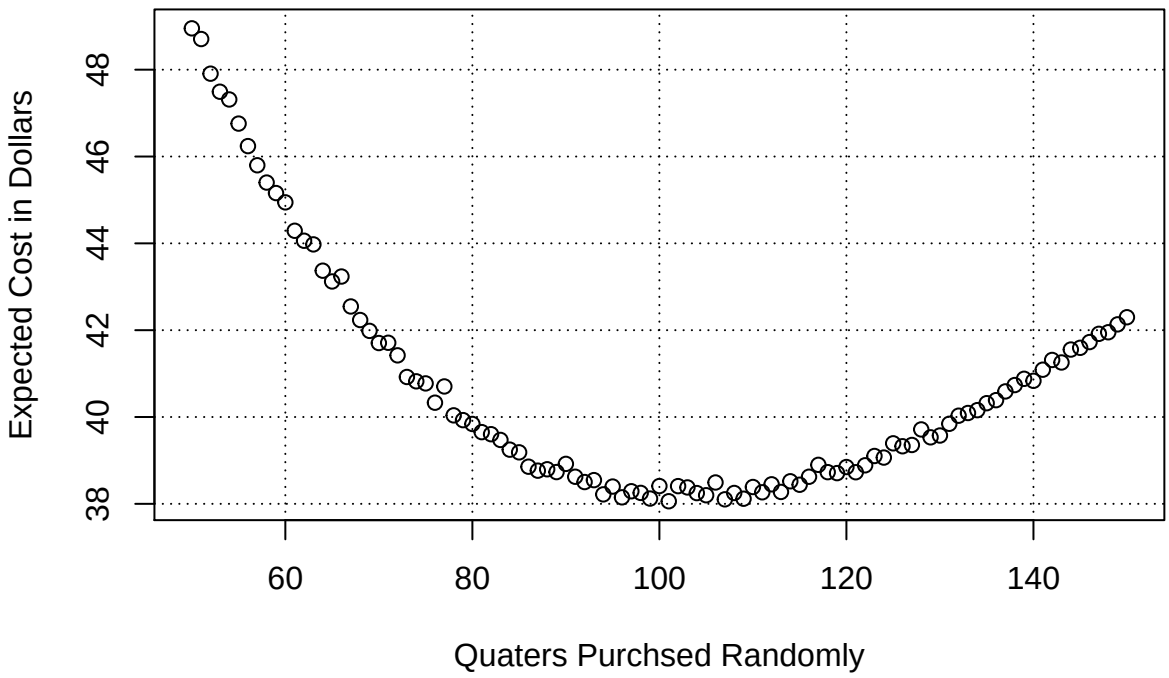
```
expected.cost = function(n = 100){
  mean(replicate(1000,purchase2(n)))
}
expected.cost()
```

```
## [1] 38.21
```

故若每次随机购买100个硬币，则获得包含所有硬币的集合所需的支出的期望值约为38$。

若改变每次随机购买的硬币数量：用函数sapply计算对应的获得包含所有硬币的集合所需的支出，以每次随机购买的硬币数量为横坐标，以期望支出为纵坐标，绘制散点图，从而推断最佳的购买策略和对应的最小期望支出。

```
n = 50:150
costs = sapply(n,expected.cost)
plot(n,costs,xlab = 'Quaters Purchsed Randomly',ylab = 'Expected Cost in Dollars')
grid(col = 'black')
```



如图所示，期望支出关于随机购买的硬币数先单调递减后单调递增，并在随机购买的硬币数为100前后达到极小值，因此每次先随机购买100枚硬币后，再花高价选择性地购买缺少的硬币，就是最优策略，最小的期望支出就是38$左右。