580_HW2

Shijie Gao, USC ID:6037-6293-25

2023-01-19

```
#1
#By observation we can see that the possible order must between [20, 40]

original_cost = 20000
original_price = 25000

reorder_cost = 22000
disposed_price = 17000

set.seed(1)
n = 10000 #10000 trials
Q = seq(20, 40, 1) #20 kinds of decisions
mprofit = NULL
sd = NULL

for (q in Q)
{
    profit = rep(0, n)
    demand = sample(c(20, 25, 30, 35, 40), c(0.3, 0.15, 0.15, 0.2, 0.2), size = n, replace = T)
    for (i in 1:n)
    {
        if (demand[i] <= q)
        {
            profit[i] = -q*original_cost + demand[i]*original_price + (q - demand[i])*disposed_price
        }
        else
        {
            profit[i] = -q*original_cost - (demand[i] - q)*reorder_cost + demand[i]*original_price
        }
    }
    sd = c(sd, sd(profit))
    mprofit = c(mprofit, mean(profit))
}
df = data.frame(Q = Q, mprofit = mprofit, sd = sd)
df
```

```
##     Q  mprofit       sd
## 1  20 127538.5 22950.52
## 2  21 128133.5 24741.44
## 3  22 128696.0 26833.74
## 4  23 129413.5 28581.41
## 5  24 129985.5 30447.05
## 6  25 129602.0 32654.61
```

```
## 7  26 129966.5 34877.41
## 8  27 129377.5 37172.34
## 9  28 129120.0 39615.18
## 10 29 129168.0 41894.88
## 11 30 128790.5 44258.06
## 12 31 128671.5 46054.00
## 13 32 127058.5 48115.69
## 14 33 126764.5 50339.81
## 15 34 124795.0 52193.73
## 16 35 123362.5 54492.05
## 17 36 121200.0 55887.68
## 18 37 120354.0 56944.02
## 19 38 117858.0 57704.60
## 20 39 115612.0 59326.97
## 21 40 114368.0 61431.08
```

```
#(a)
opt_sol = 19 + which.max(df$mprofit) #as we start from 20, so 19 + index is the true opt_sol
opt_sol
```

```
## [1] 24
```

```
#24 cars should be ordered in December

#(b)
avg_profit = df[which.max(df$mprofit), "mprofit"]
sd = df[which.max(df$mprofit), "sd"]
lb = avg_profit - 2.34*sd/sqrt(n)
ub = avg_profit + 2.34*sd/sqrt(n)
c(lb, ub)
```

```
## [1] 129273 130698
```

```
#2
set.seed(1)
n = 1000000

annual_fixed_cost = 30000
cost = sample(c(6, 7.5, 9), c(0.25, 0.5, 0.25), size = n, replace = T)

sales = sample(c(60000, 100000), c(0.5, 0.5), size = n, replace =  T)
Gross_profit = rep(0, n)
for (i in 1:n)
{
    if (sales[i] == 60000)
    {
        Gross_profit[i] = sales[i] * 10
    }
    else
    {
        Gross_profit[i] = sales[i] * 8 #sales == 100000
    }
}
profit = Gross_profit - sales*cost - annual_fixed_cost

#(a)
```

```r
expect_annual_profit = mean(profit)
expect_annual_profit
```

```
## [1] 69841.66
```

```r
#(b)
sd = sd(profit)
lb = expect_annual_profit - 1.96*sd/sqrt(n)
ub = expect_annual_profit + 1.96*sd/sqrt(n)
c(lb, ub)
```

```
## [1] 69644.01 70039.31
```

```r
#(c)
fixed_sales = 0.5*60000 + 0.5*100000
fixed_cost =  0.25*6 + 0.5*7.5 + 0.25*9
fixed_price = 0.5*8 + 0.5*10

fixed_expect_profit = fixed_sales*(fixed_price - fixed_cost) - annual_fixed_cost
fixed_expect_profit
```

```
## [1] 90000
```

```r
#Result:expected profit from a simulation is not equal to the profit from
#the scenario where each input assumes its expected value
#The reason is that when we consider the expected value of unit price, we can
#not simply avg ($8+$10)/2 to $9, because the sales for two situations are not the same
#by considering the sales, the truly expected unit price should be:
#fixed_price = (0.5*60000*$10+0.5*100000*$8)/(0.5*60000+0.5*100000) = $8.75
#then we can calculate the truly expected profit: fixed_expect_profit = 70000,
#which is equal to the simulation
```