

H9-Lu-Lu

Lu Lu

5/24/2020

```
#A store sells product X. On average, the store gets 1 customer \
    every 3 days. It takes 2 days to get an X after ordering one.

#Strategy #1: Order a new X each time one is sold.
#Pro: no excess stock
#Con: will miss a sale if out of stock

#Strategy #2: Have a standing order of X's so that you receive an X \
    every 3 days.
#Pro: less likely to miss a sale
#Con: stock may build up

probabilityCustomer = 1/3
daysUntilDelivery = 2
stock = 1
deliv = -1
totalSold = 0
totalLost = 0
totalCustomers = 0

numWeeks = 10

AccTotalSold = 0
AccTotalLost = 0
AccNetProfit = 0
numSim = 100
maxStock = 7

def Strategy1(numSim):
    numSim = numSim
    AccTotalSold = 0
    AccTotalLost = 0
    AccNetProfit = 0
    for time in range(0,numSim):
        stock = 1
```

```

    deliv = -1
    totalSold = 0
    totalLost = 0
    totalCustomers = 0
    netProfit = 0
    for week in range(numWeeks):
        for day in range(1,8):
            sold=0
            lost=0
            customers=0
            profit = 0
            if deliv == 0:
                stock +=1
            if deliv >= 0:
                deliv += -1

            if random()<probabilityCustomer:
                customers = 1

            if customers==1:
                if stock > 0:
                    sold += 1
                    profit += 10
                    stock += -1
                    if deliv < 0:
                        deliv = daysUntilDelivery
                else:
                    lost += 1
                    profit -= 2
            totalSold += sold
            totalLost += lost
            totalCustomers += customers
            netProfit += profit
        AccTotalSold += totalSold
        AccTotalLost += totalLost
        AccNetProfit += netProfit
    print 'Average sold = ', n(AccTotalSold/numSim)
    print 'Average customers lost = ', n(AccTotalLost/numSim)
    print 'Average netProfit = ', n(AccNetProfit/numSim)

def Strategy2(numSim):
    numSim = numSim
    AccTotalSold = 0
    AccTotalLost = 0
    AccNetProfit = 0
    for time in range(0,numSim):
        stock = 1

```

```

deliv = -1
totalSold = 0
totalLost = 0
totalCustomers = 0
netProfit = 0
for week in range(numWeeks):
    for day in range(1,8):
        sold=0
        lost=0
        customers=0
        profit = 0
        dayNumber = 7*week+day
        if dayNumber.mod(3) == 0:
            stock += 1

        if random()<probabilityCustomer:
            customers = 1

        if customers==1:
            if stock > 0:
                sold += 1
                profit += 10
                stock += -1
                if deliv < 0:
                    deliv = daysUntilDelivery
            else:
                lost += 1
                profit -= 2

        if stock >= 2:
            profit = 0.5 * stock -2

        totalSold += sold
        totalLost += lost
        totalCustomers += customers
        netProfit += profit
# print 'total sold = ', totalSold
# print 'total customers = ', totalCustomers
# print 'total customers lost = ', totalLost
# print 'final stock = ', stock
# print 'total profit = ', netProfit
AccTotalSold += totalSold
AccTotalLost += totalLost
AccNetProfit += netProfit
print 'Average sold = ', n(AccTotalSold/numSim)
print 'Average customers lost = ', n(AccTotalLost/numSim)
print 'Average netProfit = ', n(AccNetProfit/numSim)

```

```

print 'For Strategy 1, Simulate ', numSim, ' times, the average \
      result: '
Strategy1(numSim)
print
print 'For Strategy 2, Simulate ', numSim, ' times, the average \
      result: '
Strategy2(numSim)

```

For Strategy 1, Simulate 100 times, the average result:

Average sold = 14.260000000000000

Average customers lost = 8.940000000000000

Average netProfit = 124.72000000000000

For Strategy 2, Simulate 100 times, the average result:

Average sold =

21.040000000000000

Average customers lost = 2.340000000000000

Average netProfit = 104.60500000000000

```

probabilityCustomer = 1/3

```

```

daysUntilDelivery = 2

```

```

stock = 1

```

```

deliv = -1

```

```

totalSold = 0

```

```

totalLost = 0

```

```

totalCustomers = 0

```

```

numWeeks = 10

```

```

AccTotalSold = 0

```

```

AccTotalLost = 0

```

```

AccNetProfit = 0

```

```

numSim = 100

```

```

def Strategy3(numSim, maxStock):

```

```

    numSim = numSim

```

```

    maxStock = maxStock

```

```

    AccTotalSold = 0

```

```

    AccTotalLost = 0

```

```

    AccNetProfit = 0

```

```

    for time in range(0,numSim):

```

```

        stock = 1

```

```

        deliv = -1

```

```

        totalSold = 0

```

```

        totalLost = 0

```

```

        totalCustomers = 0

```

```

        netProfit = 0

```

```

    for week in range(numWeeks):
        for day in range(1,8):
            sold=0
            lost=0
            customers=0
            profit = 0
            if deliv == 0:
                stock +=1
            if deliv >= 0:
                deliv += -1
            dayNumber = 7*week+day
            if dayNumber.mod(7) == 0:
                stock += 1

            if random()<probabilityCustomer:
                customers = 1

            if customers==1:
                if stock > 0:
                    sold += 1
                    profit += 10
                    stock += -1
                    if deliv < 0 and stock < maxStock:
                        deliv = daysUntilDelivery
                else:
                    lost += 1
                    profit -= 2

            if stock >= 2:
                profit = 0.5 * stock -2

            totalSold += sold
            totalLost += lost
            totalCustomers += customers
            netProfit += profit
        # print 'total sold = ', totalSold
        # print 'total customers = ', totalCustomers
        # print 'total customers lost = ', totalLost
        # print 'final stock = ', stock
        # print 'total profit = ', netProfit
        AccTotalSold += totalSold
        AccTotalLost += totalLost
        AccNetProfit += netProfit
    print 'Average sold = ', n(AccTotalSold/numSim)
    print 'Average customers lost = ', n(AccTotalLost/numSim)
    print 'Average netProfit = ', n(AccNetProfit/numSim)
    print 'For Strategy 2, Simulate ', numSim, ' times with the max \

```

```
stock of ',maxStock, ' the average result: '
Strategy3(numSim,maxStock)
```

For Strategy 2, Simulate 100 times with the max stock of 7 the average result:

Average sold =

21.36000000000000

Average customers lost = 1.970000000000000

Average netProfit = 109.7550000000000

```
print 'From the previous test, for 100 times, Strategy 1 gets the \
best net profit.'
```

From the previous test, for 100 times, Strategy 1 gets the best net profit.