Dynamic Programming and Reinforcement Learning
Assignment 1: Inventory Control

Tianzheng Hu (2760270), Ivo van Miert (2634569)

November 11, 2022

1 Problem

This is a problem about finding the maximum sales revenue in a limited time. We need to take into account sales turnover, purchasing costs, logistics costs and calculate the dynamic maximum profit of these revenues and expenses influenced by customer buying demand in a finite time period. And customer buying demand grows linearly over time, and purchasing costs change over time in stages. Some of the symbols and meanings used are shown in Table 1 in appendix.

2 Question a

2.1 Basic ideas

To solve this management problem, backward recursion is used. In backward recursion, the problem is solved from the last stage backward to the first stage. The following function is used within the backward recursion:

$$V_t(x) = max_a\{-cost + reward + \sum_y P_{x,y}V_{t+1}(y)\}$$

with $cost = hx + 1_{\{a>0\}}K + price * a$, reward $= P_t * 20 + (1 - P_t) * 0$ and $P_{x,y}$ the probability of going from x inventory items, to y inventory items. The summation part of the formula can be rewritten as:

$$P_tV_{t+1}(x - d + a) + (1 - P_t)V_{t+1}(x + a)$$

2.2 Initial start

Since on time limited in 1000, the very last probability of customer demand is necessarily 1, the revenue income always exists. Meanwhile, the only cost is holding cost. By iterating over the values of x, the entries in the last column of the V matrix can be established by: $V[x, T] = 20 - h * x$, x from 0 to 1000.

2.3 Some bounds

After initializing the V matrix, the backward recursion can be started. When considering the range of purchase quantity a, it is decided to purchase at least 1 item when inventory x is 0. When inventory x is not 0, allowing no items purchased, so the minimum value of $a > max(1 - x, 0)$. Then, even if the demand is 1 for all the remaining time, at most $T - t$ items will only be needed. So at moment t, the maximum demand for the sum of the purchase $a$and the current inventory x is $T - t$, which means that $x + a \leq T - t$ so,

$$max(1 - x, 0) \leq a \leq T - t - x$$

Since the state transition is $x - d + a$ and the range of a has been determined, the range of x is also determined.

$$0 \leq x \leq T - t$$

For every iteration of t, the recursion loops through the inventory level x from 0 to $(T - t)$ to calculate the profit of action a in the state $[x, t]$.

2.4 Expected profit and optimal policy

When time $t > 900$, the matrix V is updated for every t by iterating through x from 0 to $(T - t)$ and a is always 0, demand d is t/T. Since there are no backorders, reward only exist in stock $x > 0$, calculating the following formula:

$$Q[a] = 20*(t/T)*1_{x>0} - x*h - (t/T)*V[max(x-1+a, 0), t+1] - (1-t/T)*V[max(x-0+a, 0), t+1]$$

The profit $Q[a]$ is stored in $V[x, t]$, and action a in list alpha.
When time t in range 0 to 900, the only difference is a. $Price(t)$ is a defined function to return the current price depending on time t. When time $0 < t \leq 500$, it returns 10, when $500 < t \leq 900$ it

returns 15. For every possible action a, the following formula calculates the profit:

$$Q[a+1] = 20*(t/T)*1_{x>0} - x*h + K*1_{a>0} - price(t)*a - (t/T)*V[x-1+a, t+1] - (1-t/T)*V[x-0+a, t+1]$$

Then for every possible a, find the largest $Q[a+1]$ and assign its value to $V[x,t]$, assign the a's value to $alpha[x,t]$. The result of expected profit with 0 inventory at time 0 is approximately 3760.54.

3 Question b

On the common-sense assumption that at the beginning of time, the probability of demand existing is very low, supposed neither the volume of purchases nor the inventory needs to be high to meet the customer's needs. Before time reaches 500, a large number of goods may have to be purchased due to an increase in the purchase price afterwards. Before time reaching 900, there should be another purchase of a large number of goods to avoid customer demand for subsequent events as purchases cannot be made after 900. This conjecture was largely confirmed in the subsequent simulations (Figure 1). The policy visualisation has been enlarged in Figure 2 for easier viewing in order to better see the action in detail.

4 Question c

In this section, stimulating demands d depends on the probablity t/T. Generate d lists depends on $P(d)$ and using $d(t)$ to get the specific d in time t. Calculate the profit for each time t up to 1000, based on the optimal policy alpha matrix already obtained in the previous section. The range of x and a is the same as previous section.

$$TotalProfit = \sum_{t} profit(t)$$

At time t and stock-level x, a is obrtained from the optimal policy determined before, which means a is alpha[x,t]. The at every specific time and stock profit is: $profit = 20 * d(t) * 1_{x>0} - x * h + K * 1_{a>0} - price(t) * a$

Record the result inventory state and policy for simulation, and the first simulation records as shown in Figure 3. All inventory and lost sale records of 10 times simulation are shown in Figure 5 .The Figure 4 shows a jagged growth and slow decline in inventory, which is due to a certain holding cost (h). It is realistic to stage purchases to keep stocks adequate as time advances. Before time 500 there is a peak in stocks, increasing to around 350, which is due to the rise in purchase prices after time 500. It is also consistent with the reality rule to stock up on goods before the price increase to cope with customer buying demand afterwards. Lost sales are basically only seen at the end of the time, when stocks are depleted but purchases are not allowed. When lost sales happen is also shown in the Table 2.

5 Conclusion

The dynamic model was successfully built, the optimal policy form was calculated and the desired profit was found. Heat map visualisation of the optimal policy verified the conjecture of a large number of purchases at $t = 500$ and $t = 900$. Ten simulations were successfully generated, but there were some discrepancies between the simulated profits and the calculated desired profits. The lost sale was found to occur only towards the end of time.

6 Appendix

Description	Symbol	Value
Time	T	1000
Current time	t	-
Purchase action	a	-
Purchase price(per item)	price	10 or 15
Holding cost(per item)	h	0.01
Order cost	K	10
Sale price(per item)	-	20
Demand	d	0 or 1
Profit at time t, stock x	V[x,t]	-
Profit in action a	Q[a]	-
Optimal policy at time t, stock x	alpha[x,t]	-

Table 1: Basic Symbol

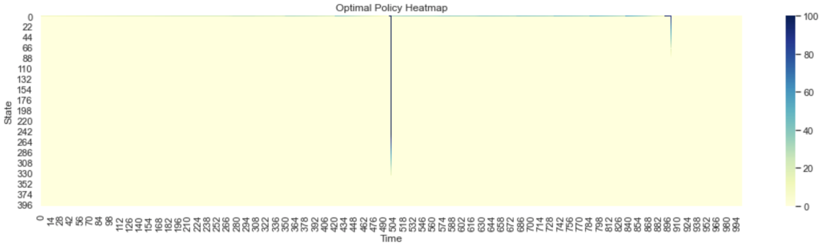


Figure 1: optimal policy

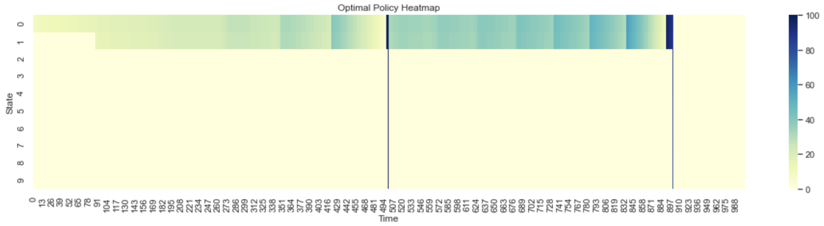
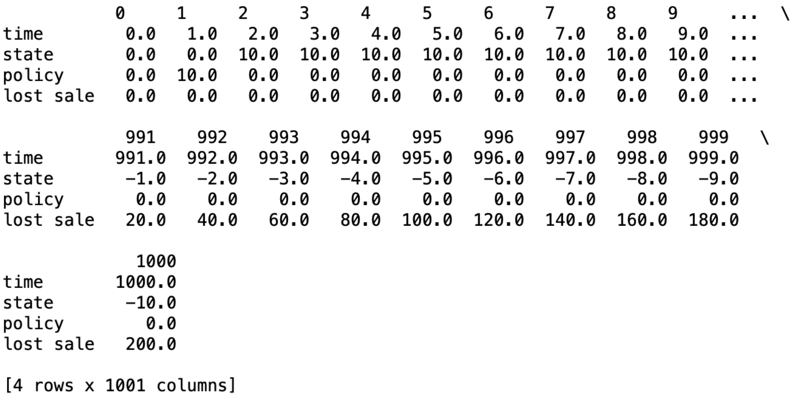


Figure 2: optimal policy in x in 1-10



```
             0     1     2     3     4     5     6     7     8     9     ...  \
time        0.0   1.0   2.0   3.0   4.0   5.0   6.0   7.0   8.0   9.0  ...
state       0.0   0.0  10.0  10.0  10.0  10.0  10.0  10.0  10.0  10.0  ...
policy      0.0  10.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  ...
lost sale   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0  ...

              991    992    993    994    995    996    997    998    999   \
time        991.0  992.0  993.0  994.0  995.0  996.0  997.0  998.0  999.0
state        -1.0   -2.0   -3.0   -4.0   -5.0   -6.0   -7.0   -8.0   -9.0
policy        0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
lost sale    20.0   40.0   60.0   80.0  100.0  120.0  140.0  160.0  180.0

              1000
time        1000.0
state        -10.0
policy         0.0
lost sale    200.0

[4 rows x 1001 columns]
```

Figure 3: simulation state policy

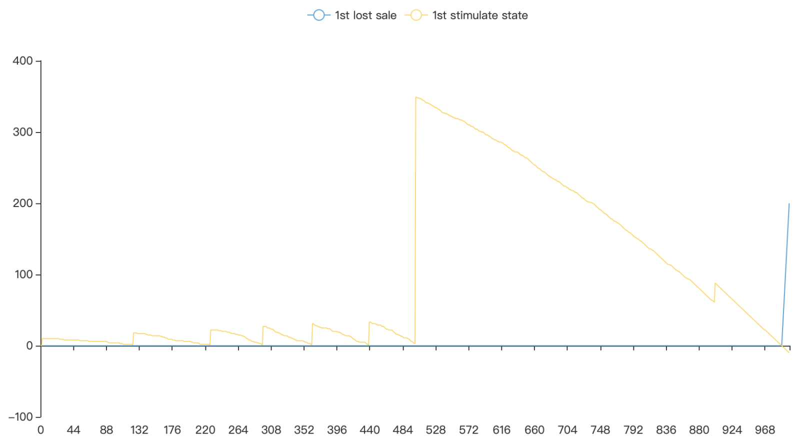


Figure 4: the first simulation inventory and lost sale

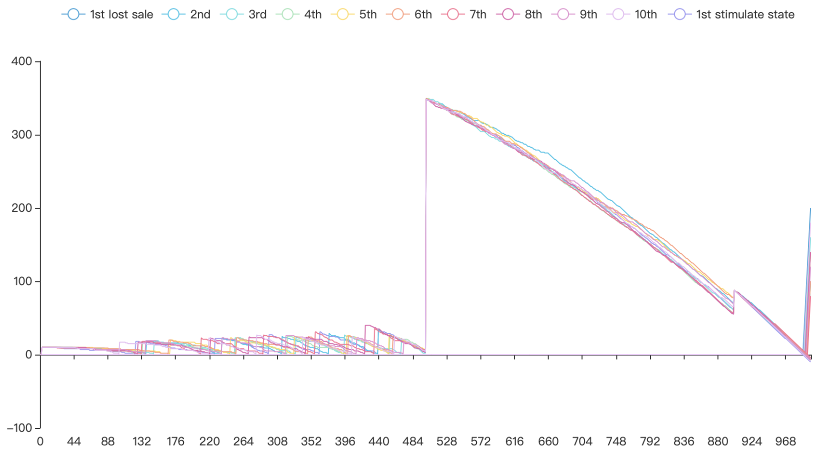


Figure 5: 10 times simulation

simulation	total profit	total loss	lost sale happen (t)
1	4095.32	200	991-1000
2	3962.18	100	996-1000
3	3927.84	160	993-1000
4	3958.21	140	994-1000
5	3830.99	140	994-1000
6	3839.58	100	996-1000
7	4112.96	80	997-1000
8	4088.06	140	994-1000
9	3783.20	120	994-1000
10	4136.24	180	992-1000

Table 2: lost sale happen