Test 5:

5000 (P,D) memory error

3000 (P,D) :

S2=[0, 0, 10, 5]

P\_regulated = 12

P\_overhedge = 9

This iteration--- 7461.453959703445 seconds --- (2 hours)

This iteration--- 1425.0932168960571 seconds ---

This iteration--- 7744.660451412201 seconds ---

This iteration--- 601.1723203659058 seconds ---

This iteration--- 2811.6289625167847 seconds ---

--- 20044.169486761093 seconds --- (5.5 hours)

‘q2 = [[ 0., 0., 10., 5., 3.]]])

‘q3 =

[[10., 3., 11., 6., 2.], different from 1000 case which = 0

[10., 3., 11., 4., 0.],

[10., 3., 9., 4., 0.],

[10., 3., 9., 6., 2.],

Q2

Out[6]:

array([[ 0. , 0. , 10. , 5. , 0. ,

9.97473527],

[ 0. , 0. , 10. , 5. , 1. ,

9.9817376 ],

[ 0. , 0. , 10. , 5. , 2. ,

9.98873994],

[ 0. , 0. , 10. , 5. , 3. ,

9.99574227],

[ 0. , 0. , 10. , 5. , 4. ,

5.13229596]])

Q3[1:10,:,:]

Out[14]:

[[ 10. , 3. , 11. , 6. ,

0. , 8.95831944],

[ 10. , 3. , 11. , 6. ,

1. , 8.96132044],

[ 10. , 3. , 11. , 6. ,

2. , 8.96432144],

[ 10. , 3. , 11. , 6. ,

3. , -1.09136379]],

[[ 10. , 3. , 9. , 6. ,

0. , 14.94431477],

[ 10. , 3. , 9. , 6. ,

1. , 14.94731577],

[ 10. , 3. , 9. , 6. ,

2. , 14.95031677],

[ 10. , 3. , 9. , 6. ,

3. , 5.90163388]],

Matlab:

N = 3000

‘’q0 = 3

Q0 = 10.0651666666667 10.0683333333333 10.0790000000000 10.0496666666667 5.28008333333333 0.274166666666666 -13.6917500000000 -28.0420000000000 -46.6990000000000

‘q1 = 11 6 3 1 4 (Different from Q-learning) 0.250000000000000 9.06533333333333

11 4 3 0 3 0.250000000000000 6.98933333333333

9 6 3 2 5 0.250000000000000 15.0320000000000

9 4 3 0 3 0.250000000000000 9.11200000000000

Case:

11 6 3 0 3 0.250000000000000 9.00533333333333

11 6 3 1 4 0.250000000000000 9.06533333333333

11 6 3 2 5 0.250000000000000 8.93333333333333

Newsvendor:

A close up of a mans face

Description automatically generated phi ^-1

alpha1 = underhedged penalty

alpha2 = overedged penalty

D~N(0,sigma\_t^2)

+1, -1 w.p ½ and ½

Mean = 0, variance = 1

First when price changed from 10 to 11, since I don’t know alpha1 at t =2, I will use the mean to replace = 11,

T=1, assume T=0, A0=0

X = 1\*phi^1((10+11)/(11+9))= 7 sum (Different)

3+2 by Q-learning

X = 1\*phi^-1((9+9)/(9+9)) = 7

D = 7 w.p. ½ and 5 w.p. ½

X = 1\*CDF^-1(1.05) = 7

X = 1\*CDF^-1(1) =7

T=0, Assume 1 period decision,

X = 1\*phi^1((10+10)/(10+9))= 7

D = 6 w.p. ½ and 4 w.p. ½

X = 1\*CDF^-1(1.05) = 7

In the case of I must lose

MC Method:

Pbar = 8

P0 = 10;

D0 = 5;

DelP = 1; pu = 1/2; pd=1-pu;

DelD = 1; p =1/2;Pover = 9;

N = 5000

‘q0 = 1, Q0 = --9.9461

‘q1 = 11 6 1 1 2 0.250000000000000 -16.9484000000000

‘q1 = 11 4 1 1 2 0.250000000000000 -10.9356000000000

‘q1 = 9 6 1 1 2 0.250000000000000 -6.94960000000000

‘q1 = 9 4 1 1 2 0.250000000000000 -4.95080000000000

Q-learning:

N = 1000

‘q2 = [[ 0., 0., 10., 5., 0.]]])

Q2 = [ 0. , 0. , 10. , 5. ,

0. , -9.66025485],

‘q3=[[[ 0., 0., 11., 6., 0.],

[ 0., 0., 11., 4., 0.],

[ 0., 0., 9., 6., 0.],

[ 0., 0., 9., 4., 0.],

Q3 = [ 0. , 0. , 9. , 4. ,

0. , -4.002002 ],

[ 0. , 0. , 9. , 4. ,

1. , -4.00700701],

[[ 0. , 0. , 11. , 6. ,

1. , -17.98998999],

[[ 0. , 0. , 9. , 6. ,

0. , -5.99199199],

[ 0. , 0. , 9. , 6. ,

1. , -5.996997 ],

[[ 0. , 0. , 11. , 4. ,

0. , -12. ],

[ 0. , 0. , 11. , 4. ,

1. , -12.00500501],

[ 0. , 0. , 11. , 4. ,

2. , -12.01001001],

In the case of P & Q are perfectly positively correlated

‘q2 = [[ 0., 0., 10., 5., 3.]]])

‘q3 = [10., 3., 11., 6., 2.],

[10., 3., 9., 4., 0.],

In the case of P & Q are perfectly negatively correlated

‘q2 = [[ 0., 0., 10., 5., 3.]]])

‘q3 = [10., 3., 11., 4., 0.],

[10., 3., 9., 6., 2.],

In the case of demand more likely to increase