

```

1 # Python code for simple probabilistic inventory by dynamic programming
2
3 import numpy
4
5 hugeNumber = float("inf")
6 unused     = -1000
7
8 stages     = 3
9 startInventory = 1
10 inventoryCapacity = 3
11 productionCapacity = 4
12
13 setupCost   = 3.0
14 variableCost = 2.0
15 holdingCost = 1.0
16 salvageValue = 2.0
17
18 minDemand = 1
19 maxDemand = 2
20
21 # demandProb[t,d] is probability demand will be d in period t
22 demandProb = numpy.array([[unused, unused, unused],
23                             [0.0, 0.5, 0.5],
24                             [0.0, 0.5, 0.5],
25                             [0.0, 0.5, 0.5]])
26
27 f = numpy.zeros([stages + 2, inventoryCapacity+1])
28 x = numpy.zeros([stages + 1, inventoryCapacity+1], dtype=int)
29
30 # Set the value of ending up in each final state (not zero in this case)
31 for i in range(inventoryCapacity + 1) :
32     f[stages+1,i] = -salvageValue*i # Negative "cost" = benefit for each leftover unit
33
34 for t in range(stages,0,-1) :
35
36     for i in range(inventoryCapacity+1) :
37
38         minProduction = max(0, maxDemand - i)
39         maxProduction = min(productionCapacity, inventoryCapacity - i + minDemand)
40
41         value = hugeNumber
42
43         for p in range(minProduction, maxProduction+1) :
44
45             # Compute production cost
46             productionCost = variableCost*p
47             if p > 0 :
48                 productionCost += setupCost
49
50             moveValue = productionCost
51             # Accumulate expected value of this decision
52             for d in range(minDemand,maxDemand+1) :
53                 j = i + p - d
54                 moveValue += demandProb[t,d]*(holdingCost*j + f[t+1,j])
55
56             if moveValue < value :
57                 value = moveValue
58                 bestMove = p
59
60         # End of p Loop
61
62         f[t,i] = value
63         x[t,i] = bestMove
64
65     # End of i Loop
66
67 # End of t Loop
68

```

```
69 print("Optimal expected cost is " + str(f[1,startInventory]))
70 print("Period 1: produce " + str(x[1,startInventory]))
71 for t in range(2,stages+1) :
72     print("Period " + str(t) + ":")
73     for i in range(inventoryCapacity + 1) :
74         print("    If inventory=" + str(i) + ", produce " + str(x[t,i]))
75
```