

# Stochastic Models and Optimization: Problem Set 3

Roger Garriga Calleja, José Fernando Moreno Gutiérrez, David Rosenfeld, Katrina Walker

March 10, 2017

## Q1. Inventory control with forecasts

### (a): Single period

**Primitives**  $D$ : Demand, which is a random variable

$$D = \begin{cases} P_L & \text{with probability } q \\ P_S & \text{with probability } 1 - q \end{cases}$$

$p$ : Sale price per unit

$c$ : Ordering cost per unit

$s$ : Salvage value per unit

$b = p - c$ : Backorder cost per unit

$h = c - s$ : Inventory holding cost per unit

$Q$ : Inventory decision.

The profit for this problem can be written as:

$$\pi(Q) = E[p \cdot \min\{D, Q\} - c \cdot Q + s(Q - D)^+]$$

$$\pi(Q) = pE[D] - (p - c)E[(D - Q)^+] + (c - s)E[(Q - D)^+]$$

$$\pi(Q) = pE[D] - bE[(D - Q)^+] - hE[(Q - D)^+]$$

We then expand the formula to account for the expectation of demand  $D$  which takes value  $P_L$  with probability  $q$ :

$$\pi(Q) = pE[D] - b[q \cdot E_{P_L}[(D - Q)^+] + (1 - q) \cdot E_{P_S}[(D - Q)^+]] - h[qE_{P_L}[(Q - D)^+] + (1 - q)E_{P_S}[(Q - D)^+]]$$

We take the derivative of the profit with respect to quantity in order to minimise, and set it equal to zero (with  $P_L$  and  $P_S$  the probability distributions in the large and small demand cases respectively):

$$\pi'(Q^*) = -b[qP_L(D > Q^*) + (1 - q)P_S(D > Q^*)] + h[qP_L(Q^* > D) + (1 - q)P_S(Q^* > D)] = 0$$

$$-bq(1 - P_L(D \leq Q^*)) - b(1 - q)(1 - P_S(D \leq Q^*)) + hqP_L(D \leq Q^*) + h(1 - q)P_S(D \leq Q^*) = 0$$

We can then obtain that the optimal decision  $Q^*$  satisfies:

$$qP_L(D \leq Q^*) + (1 - q)P_S(D \leq Q^*) = \frac{b}{b + h}$$

### (b) Multi-period problem

**Primitives**

$x_k$ : Inventory level at period  $k$

$u_k$ : Decision variable over the number of units to order.

$w_k$ : Demand at period k.

$y_{k+1} = \xi$  where  $\xi$  takes the value "large demand" with probability q:

$$\xi_k = \begin{cases} L & \text{with probability } q \\ S & \text{with probability } 1 - q \end{cases}$$

$c$ : ordering cost per unit

$p$ : Sale price per unit

$s$ : Salvage value

$b = p - c$ : Backorder cost per unit

$h = c - s$ : Inventory holding cost per unit

### Dynamics:

$$x_{k+1} = x_k + u_k - w_k$$

$$y_{k+1} = \xi_k$$

$$g_N(x_N) = -s(x_N)$$

$$g_k(x_k, u_k, w_k) = cu_k - pw_k + b\max\{w_k - x_k - u_k, 0\} + h\max\{x_k + u_k - w_k, 0\}$$

**DP algorithm:**  $J_N(x_N) = -s(x_N)$

$$J_k(x_k) = \min_{u_k \geq 0} E_{w_k, \xi_k} [g_k(x_k, u_k, w_k) + J_{k+1}(x_k + u_k - w_k)]$$

We can introduce the variable  $z_k = x_k + u_k$ , and rewrite our DP equation as follows:

$$J_k(x_k) = \min_{z_k \geq x_k} [G_k(z_k)] - cx_k$$

$$\text{where } G_k(z_k) = cz_k + bE[\max\{0, w_k - z_k\}] + hE[\max\{0, z_k - w_k\}] + E[J_{k+1}(z_k - w_k)]$$

Once again we can expand this expression to take into account the two different values of  $\xi$  for large and small demand:

$$G_k(z_k) = cz_k + bqE[\max\{0, w_k - z_k\} | \xi = L] + b(1 - q)E[\max\{0, w_k - z_k\} | \xi = S] + hqE[\max\{0, z_k - w_k\} | \xi = L] + h(1 - q)E[\max\{0, z_k - w_k\} | \xi = S] + E[J_{k+1}(z_k - w_k)]$$

$G_k(z_k)$  is a convex function for all k, and there will be a solution  $S_k$  which minimises  $G(z)$  such that the optimal ordering policy becomes:

$$u_k^* = \mu_k^* = S_k - x_k \text{ if } S_k > x_k, \text{ and } 0 \text{ otherwise.}$$

## Q2. Inventory Pooling

### Primitives

D = demands

Q = quantity ordered

P = price

h = inventory costs = c-s

b = backholding costs = p-c

First we will show that  $Q_p^* = \sqrt{n}Q^* + \mu(n - \sqrt{n})$

$$\begin{aligned} G(Q) &= hE[(Q - D)]^+ + bE[(D - Q)]^+ \\ G'(Q^*) &= hP(D \leq Q^*) - b(1 - P[(D \leq Q^*)]) = 0 \\ P(D \leq Q^*) &= \frac{b}{h + b} \end{aligned}$$

Now, for the pooling we would obtain the same:

$$P\left(\sum_{i=1}^n D_i \leq Q_p^*\right) = \frac{b}{b + R}$$

Since this  $\sum_{i=1}^n D_i = \sqrt{n}D_1 + \mu(n - \sqrt{n})$ ,

$$P(\sqrt{n}D_1 + \mu(n - \sqrt{n}) \leq Q_p^*) = \frac{b}{b + R} \iff P(D_1 \leq \frac{1}{\sqrt{n}}(Q_p^* - \mu(n - \sqrt{n}))) = \frac{b}{b + h}$$

which implies that  $\frac{1}{\sqrt{n}}(Q_p^* - \mu(n - \sqrt{n})) = Q^*$ . Thus,  $Q_p^* = \sqrt{n}Q^* + \mu(n - \sqrt{n})$ . Next, we will apply the hint to prove the desired result:

$$\begin{aligned} nG(Q^*) &= n[hE[(Q^* - D)]^+ + bE[(D - Q^*)]^+] \\ G(Q_p^*) &= [hE[(Q_p^* - \sum_{i=1}^n D_i)]^+ + bE[(\sum_{i=1}^n D_i - Q_p^*)]^+] \end{aligned}$$

Since  $\sum_{i=1}^n D_i = \sqrt{n}D_1 + \mu(n - \sqrt{n})$  and  $Q_p^* = \sqrt{n}Q^* + \mu(n - \sqrt{n})$ , we can show that:

$$\begin{aligned} G(Q_p^*) &= [hE[(\sqrt{n}Q^* + \mu(n - \sqrt{n}) - \sqrt{n}D_1 - \mu(n - \sqrt{n}))]^+ \\ &\quad + bE[\sqrt{n}D_1 + \mu(n - \sqrt{n}) - \sqrt{n}Q^* + \mu(n - \sqrt{n})]^+]] \\ &= \sqrt{n}hE[(Q^* - D_1)]^+ + \sqrt{n}bE[(D_1 - Q^*)]^+ = \frac{nG(Q^*)}{\sqrt{n}} \end{aligned}$$

Q.E.D.

## Q4 An Investment Problem

Primitives:

$x_k$ : wealth at the beginning of period  $k$ .

$u_k$ : amount invested.

$w_k$ : outcome (return).

$$w_k = \begin{cases} 2u_k & \text{w.p. } p \\ 0 & \text{w.p. } 1 - p \end{cases}$$

where  $\frac{1}{2} < p < 1$ .

Constrains:

$$u_k \geq 0,$$

$$u_k \leq x_k.$$

Dynamics:

$$x_{k+1} = x_k - u_k + w_k,$$

Cost:

$$g_N(x_N) = x_N$$

$$g_k(x_k, u_k, w_k) = 0.$$

DP algorithm:

To simplify th  $J_N(x_N) = \log(x_N)$ .

$$J_k(x_k) = \max_{u_k \in \mathcal{U}_k} \mathbb{E}_{w_k} [J_{k+1}(x_k - u_k + w_k)].$$

We will first prove that  $J_k(x_k) = A_k + \log(x_k)$  by induction (where  $\log$  is the logarithm in base  $e$ ).

$$J_{N-1}(x_{N-1}) = \max_{u_{N-1}} \mathbb{E}_{w_{N-1}} [\log(x_{N-1} - u_{N-1} + w_{N-1})] = \quad (1)$$

$$= \max_{u_{N-1}} \mathbb{E}_{w_{N-1}} [\log(x_{N-1} - u_{N-1} + w_{N-1})] = \quad (2)$$

$$= \max_{u_{N-1}} [p \log(x_{N-1} + u_{N-1}) + (1-p) \log(x_{N-1} - u_{N-1})]. \quad (3)$$

Now, we derive and apply FOC to find the maximum,

$$\frac{p}{x_{N-1} + u_{N-1}} - \frac{1-p}{x_{N-1} - u_{N-1}} = 0 \quad (4)$$

$$px_{N-1} - pu_{N-1}^* = x_{N-1} + u_{N-1}^* - px_{N-1} - pu_{N-1}^*, \quad (5)$$

solving we obtain  $u_{N-1}^* = (2p-1)x_{N-1}$ , which makes sense, because  $2p-1$  is between 0 and 1 for  $\frac{1}{2} < p < 1$ . Plugging into the expression of  $J_{N-1}(x_{N-1})$

$$J_{N-1}(x_{N-1}) = p \log(x_{N-1} + (2p-1)x_{N-1}) + (1-p) \log(x_{N-1} - (2p-1)x_{N-1}) = \quad (6)$$

$$= p \log(2px_{N-1}) + (1-p) \log(2(1-p)x_{N-1}) = \quad (7)$$

$$= p \log(2p) + p \log(x_{N-1}) + (1-p) \log(2(1-p)) + (1-p) \log(x_{N-1}) = \quad (8)$$

$$= p \log(2p) + (1-p) \log(2(1-p)) + \log(x_{N-1}). \quad (9)$$

So, for  $N-1$  we have  $A_{N-1} = p \log(2p) + (1-p) \log(2(1-p))$ . Now, suppose  $J_{k+1}(x_{k+1}) = A_{k+1} + \log(x_{k+1})$  let's prove it for  $k$ :

$$J_k(x_k) = \max_{u_k} \mathbb{E}_{w_k} [A_{k+1} + \log(x_{k+1})] = \quad (10)$$

$$= \max_{u_k} \mathbb{E}_{w_k} [A_{k+1} + \log(x_k - u_k + w_k)] = \quad (11)$$

$$= \max_{u_k} [A_{k+1} + p \log(x_k + u_k) + (1-p) \log(x_k - u_k)]. \quad (12)$$

Applying first order conditions we would obtain the same equation as before ( $A_{k+1}$  does not depend on  $u_k$ ), so  $u_k = (2p - 1)x_k$ . Plugging in the  $J_k(x_k)$  we obtain the same way as before

$$J_k(x_k) = A_{k+1} + p \log(2p) + (1 - p) \log(2(1 - p)) + \log(x_k). \quad (13)$$

So,  $J_k(x_k) = A_k + \log(x_k)$ , where  $A_k = A_{k+1} + p \log(2p) + (1 - p) \log(2(1 - p))$ . So, the policy will be to invest  $(2p - 1)x_k$  of wealth each period  $k$ .