1 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 $Q_p^* = \sqrt{n}Q^* + \mu(n - \sqrt{n})$

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

Now, for the pooling we would obtain the same:

$$P(\sum_{L=1}^{n} D_i \leqslant Q_p^*) = \frac{b}{b+R}$$

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

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

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

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

Q.E.D.

Problem 4 (An Investment Problem): An investor has the opportunity to make N sequential investments: at time k he may invest any amount $u_k \geqslant 0$ that does not exceed

his current wealth x_k (does not exceed his current wealth, x_0, plus his gain or minus his loss thus far). He wins his investment back and as much more with probability p, where $\frac{1}{2} < p < 1$, and he loses his investment with probability $(1-p)$. Find the optimal investment strategy.

<u>Primitives:</u>
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 $\frac{1}{2} < p < 1$.

<u>Constrains:</u>
$u_k \geqslant 0$,
$u_k \leqslant x_k$.

<u>Dynamics:</u>
$x_{k+1} = x_k - u_k + w_k$,

<u>Cost:</u>
$g_N(x_N) = x_N$
$g_k(x_k, u_k, w_k) = 0$.

<u>DP algorithm:</u>
To simplify th $J_N(x_N) = \log(x_N)$.
$J_k(x_k) = \max\limits_{u_k \in \mathcal{U}_k} \mathbb{E}_{w_k}\left[J_{k+1}(x_k - u_k + w_k)\right]$.
We will first prove that $J_k(x_k) = A_k + \ln(x_k)$ by induction.

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

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

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

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

$$\frac{p}{x_{N-1} + u_{N-1}} - \frac{1-p}{x_{N-1} - u_{N-1}} = 0 \tag{4}$$

$$px_{N-1} - pu_{N-1} = x_{N-1} + u_{N-1} - px_{N-1} - pu_{N-1}, \tag{5}$$

solving we obtain $u_{N-1} = (2p-1)x_{N-1}$. Plugging into the expression of $J_{N-1}(x_{N-1})$

$$J_{N-1}(x_{N-1}) = \tag{6}$$