Advanced Agribusiness Management

Rodney Beard

May 9, 2017

Readings

The Dynamic Economic Lot size Model

- Demand and purchase cost can vary over time.
- Agricultural products can have both seasonal demand and supply
- Assume the inventory manager is typically an agribusiness
- Quantity ordered may be constrained at certain time periods due to availability Dynamic lot size model doesn't consider this.
- discrete time setting $t = 0, 1, 2, 3, \ldots, T$
- $d(t)$ demand at time t

Decision variables

- $x(t$ inventory at time t
- $z(t)$ order quantity at time t

Decision sequence

1. Observe inventory $x(t)$ at time t
2. decide how much $z(t)$ to order at time t
3. Continue until step 1 at time T is reached

Wagner-Whitin -Dynamic Economic Lot-sizing Problem

$$\min \sum_{t=0}^{t-1} [k(t)H(z(t)) + c(t)z(t)] + \sum_{t=1}^{T} h(t)x(t)$$

subject to

$$x(t+1) = x(t) + z(t) - d(t), t = 0, \dots, T-1$$

$$x(0) = x_0$$

$$x(t) \geq 0, t = 1, \dots, T$$

$$z(t) \geq 0t = 1, \dots, T-1$$

where

- $k(t)$ - fixed order cost at time t
- $c(t)$ - variable order cost at time t
- $h(t)$ - inventory holding cost at time t.
- $H(z)$ is the Heaviside function,

$$H(z) = \begin{cases} 1 & z > 0 \\ 0 & z \leq 0 \end{cases}$$

This includes purchase costs when purchases are made but otherwise sets them to zero. (Not derivative of the Heaviside function is the Dirac delta function, normally H should be used for Heaviside and δ for Dirac. Source for these notes used δ should be H)

Solution procedures

- linear programming
- mixed integer linear programming
- forward recursion
- backward recursion (dynamic programming)

Example

t	$d(t)$	$k(t)$	$c(t)$	$h(t)$
0	10	40	2	-
1	2	40	2	1
2	12	40	2	1
3	4	40	2	1
4	14	40	2	1
5	-	-	-	1

How much should we order at each time?

Solution via Integer programming

The Heavisde function makes the model discontinuous and non-linear. Transform the problem by introducing binary indicator variables for ordering:

$$v(t) = \begin{cases} 1 & orderattimet \\ 0 & otherwise \end{cases}$$

$$v(t) \in \{0, 1\}$$

and introduce the constraint

$$z(t) \leq D(t, T)v(t)$$

where

- $D(t) = \sum_{s=0}^{T} d(s)$ cumulative demand up until t
- $D(t, u) = D(u-1) - D(t-1), t \leq u$
  so $D(t, T) = D(T-1) - D(t-1)$ or

$$\sum_{s=0}^{T-1} d(s) - \sum_{s=0}^{t-1} d(s) = \sum_{t}^{T-1} d(s)$$

Integer programming version

$$\min_{x,z} \sum_{t=0}^{t-1} [k(t)H(z(t)) + c(t)z(t)] + \sum_{t=1}^{T} h(t)x(t)$$

subject to

$$x(t+1) = x(t) + z(t) - d(t), t = 0, \ldots, T-1$$

$$x(0) = x_0$$

$$x(t) \geq 0, t = 1, \ldots, T$$
$$z(t) \geq 0 t = 1, \ldots, T-1$$

$$z(t) \leq D(t, T)v(t)$$

$$v(t) \in \{0, 1\}, t = 1, \ldots, T-1$$

Exercise 1 (2 pts)

Using the previous data solve the dynamic lot sizing problem as a mixed integer programming problem using SciPy Optimize

Capacity constraint on z - Time varying (seasonal) production

This can happen because the processing plant has a constraint on its ability ot process orders or because of a lack of availability of supply, fruit and vegetables of rexample are seasonal and may not be able in sufficient quantity

$$z(t) \leq z_{max}(t), t = 1, \ldots, T - 1$$

Reformulate the integer constraint in the integer programming problem to be

$$z(t) \leq \max\{D(t, T), z_{max}(t)\} v(t), t = 1, \ldots, T - 1$$

Capacitated Dynamic Lot sizing Problem

- To solve we need to forecast future production of $z_{max}(t)$ adding this constraint will lead to larger invenotires to avoid stockouts.
- Capacitated lot-sizing problem also distorts set-up costs because these are incurred in every period but should perhaps occur only once at the beginning of a production run.

Exercise 2 (1 pt)

Resolve the mixed integer programming problem with a capacity constraint on z, allow the capacity to be time varying to mimic seasonal supply changes. Choose your own values.

Stochastic Demand

- *lambda* demand rate
- L Lead time
- q batch size
- r re-order point

$$Pr\{D(t) = d\} = \frac{(\lambda t)^d e^{-\lambda t}}{d!}$$

- order quantity q, re-order point r policy (r, q) policy.
- re-order point, target level s policy, (r, s) policy.

Poisson demand

For each $t > 0$

$$D(t,u) = D(u) - D(t), u \geq t$$

where $D(t)$ is cumulative demand through time.

$$\mathbf{D} = \{D(t), t \geq 0\}$$

D follows a Poisson process.

Variables

- $I(t)$ inventory
- $B(t)$n backorders (unfulfilled orders, like excess demand)
- $IN(t) = I(t) - B(t)$ net inventory
- $A(t)$ stockout indicator
- $IO(t)$ inventory on order (not the same as a back-order)
- $IP(t)$ inventory position

$$I(t) = \max\{IN(t), 0\}$$

Performance measures

These are what management needs to track

- $\bar{A}$ Average stockout frequency
- $\bar{B}$ average backorders
- $\bar{I}$ average inventory
- $\bar{OF}$ average order frequency.

Example:

$$\bar{I} = \lim_{T \to \infty} \frac{1}{T} \int_0^T I(t)dt$$

(r,q) Policy, q=1

Known as a base-stock policy.
Base-stock level $s = r + 1$
Re-order whenever $IP(t) = s$

$$IN(t) = s - IO(t)$$

where $IO(t) = D(t - L, t)$

Computing the performance measures

Remember this is what management wants to know.

- $\bar{A} = Pr\{IN \leq 0\} = Pr\{D > s\} = 1 - Pr\{D \leq s\} = G^0(j)$
- $\bar{B} = E(IN^- = E(\max(D - s, 0)) = \lambda L - \sum_{0 \leq j \leq} G^0(j)$
- $\bar{I} = E(IN^+) = E(IN + IN^-) = s - \lambda L + \bar{B}$
- $\bar{OF} = \lambda$

These can be easily calculated from demand data. Perfromance constraint percent order filled from stock is 1-stockout rate.

Exercise

- Compute $\bar{A}$ this can be done from the Poisson tables (see next two slides)
- Compute $\bar{B}$
- Compute $\bar{I}$
- Problem: Demand is 3.6 per month, Lead-time is 2 weeks (think about time units here) and 80% of orders should be filled from stock (performance constraint).
- Complete the table

s	$\bar{A}$	$\bar{B}$	$\bar{I}$
0			
1			
2			
3			
4			
5			
6			

What level of s should be chosen to meet the performance constraint?

Poisson Table

THE POISSON DISTRIBUTION

Cumulative Distribution Function

The columns correspond to different values for the mean $(\lambda))$ of a Poisson variable. The entries in the body of the table represent the probabilities that such a random variable does not exceed the integer x at the left of the row. For example, a Poisson variable of mean 0.8 is 2 or less with probability 0.953.

x	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0	0.905	0.819	0.741	0.670	0.607	0.549	0.497	0.449	0.407	0.368
1	0.995	0.982	0.963	0.938	0.910	0.878	0.844	0.809	0.772	0.736
2	1.000	0.999	0.996	0.992	0.986	0.977	0.966	0.953	0.937	0.920
3	1.000	1.000	1.000	0.999	0.998	0.997	0.994	0.991	0.987	0.981
4	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.996
5	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999
6	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

x	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
0	0.333	0.301	0.273	0.247	0.223	0.202	0.183	0.165	0.150	0.135
1	0.699	0.663	0.627	0.592	0.558	0.525	0.493	0.463	0.434	0.406
2	0.900	0.879	0.857	0.833	0.809	0.783	0.757	0.731	0.704	0.677
3	0.974	0.966	0.957	0.946	0.934	0.921	0.907	0.891	0.875	0.857
4	0.995	0.992	0.989	0.986	0.981	0.976	0.970	0.964	0.956	0.947
5	0.999	0.998	0.998	0.997	0.996	0.994	0.992	0.990	0.987	0.983
6	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997	0.997	0.995
7	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
8	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

x	2.2	2.4	2.6	2.8	3.0	3.2	3.4	3.6	3.8	4.0
0	0.111	0.091	0.074	0.061	0.050	0.041	0.033	0.027	0.022	0.018
1	0.355	0.308	0.267	0.231	0.199	0.171	0.147	0.126	0.107	0.092
2	0.623	0.570	0.518	0.469	0.423	0.380	0.340	0.303	0.269	0.238
3	0.819	0.779	0.736	0.692	0.647	0.603	0.558	0.515	0.473	0.433
4	0.928	0.904	0.877	0.848	0.815	0.781	0.744	0.706	0.668	0.629
5	0.975	0.964	0.951	0.935	0.916	0.895	0.871	0.844	0.816	0.785
6	0.993	0.988	0.983	0.976	0.966	0.955	0.942	0.927	0.909	0.889
7	0.998	0.997	0.995	0.992	0.988	0.983	0.977	0.969	0.960	0.949
8	1.000	0.999	0.999	0.998	0.996	0.994	0.992	0.988	0.984	0.979
9	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.996	0.994	0.992
10	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997
11	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999
12	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

x	4.2	4.4	4.6	4.8	5.0	5.2	5.4	5.6	5.8	6.0
0	0.015	0.012	0.010	0.008	0.007	0.006	0.005	0.004	0.003	0.002
1	0.078	0.066	0.056	0.048	0.040	0.034	0.029	0.024	0.021	0.017
2	0.210	0.185	0.163	0.143	0.125	0.109	0.095	0.082	0.072	0.062
3	0.395	0.359	0.326	0.294	0.265	0.238	0.213	0.191	0.170	0.151
4	0.590	0.551	0.513	0.476	0.440	0.406	0.373	0.342	0.313	0.285
5	0.753	0.720	0.686	0.651	0.616	0.581	0.546	0.512	0.478	0.446
6	0.867	0.844	0.818	0.791	0.762	0.732	0.702	0.670	0.638	0.606
7	0.936	0.921	0.905	0.887	0.867	0.845	0.822	0.797	0.771	0.744
8	0.972	0.964	0.955	0.944	0.932	0.918	0.903	0.886	0.867	0.847
9	0.989	0.985	0.980	0.975	0.968	0.960	0.951	0.941	0.929	0.916
10	0.996	0.994	0.992	0.990	0.986	0.982	0.977	0.972	0.965	0.957
11	0.999	0.998	0.997	0.996	0.995	0.993	0.990	0.988	0.984	0.980
12	1.000	0.999	0.999	0.999	0.998	0.997	0.996	0.995	0.993	0.991
13	1.000	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997	0.996
14	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
15	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999
16	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

x	6.5	7.0	7.5	8.0	8.5	9.0	9.5	10.0	10.5	11.0
0	0.002	0.001	0.001	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1	0.011	0.007	0.005	0.003	0.002	0.001	0.001	0.000	0.000	0.000
2	0.043	0.030	0.020	0.014	0.009	0.006	0.004	0.003	0.002	0.001
3	0.112	0.082	0.059	0.042	0.030	0.021	0.015	0.010	0.007	0.005
4	0.224	0.173	0.132	0.100	0.074	0.055	0.040	0.029	0.021	0.015
5	0.369	0.301	0.241	0.191	0.150	0.116	0.089	0.067	0.050	0.038
6	0.527	0.450	0.378	0.313	0.256	0.207	0.165	0.130	0.102	0.079
7	0.673	0.599	0.525	0.453	0.386	0.324	0.269	0.220	0.179	0.143
8	0.792	0.729	0.662	0.593	0.523	0.456	0.392	0.333	0.279	0.232
9	0.877	0.830	0.776	0.717	0.653	0.587	0.522	0.458	0.397	0.341
10	0.933	0.901	0.862	0.816	0.763	0.706	0.645	0.583	0.521	0.460
11	0.966	0.947	0.921	0.888	0.849	0.803	0.752	0.697	0.639	0.579
12	0.984	0.973	0.957	0.936	0.909	0.876	0.836	0.792	0.742	0.689
13	0.993	0.987	0.978	0.966	0.949	0.926	0.898	0.864	0.825	0.781
14	0.997	0.994	0.990	0.983	0.973	0.959	0.940	0.917	0.888	0.854
15	0.999	0.998	0.995	0.992	0.986	0.978	0.967	0.951	0.932	0.907
16	1.000	0.999	0.998	0.996	0.993	0.989	0.982	0.973	0.960	0.944
17	1.000	1.000	0.999	0.998	0.997	0.995	0.991	0.986	0.978	0.968
18	1.000	1.000	1.000	0.999	0.999	0.998	0.996	0.993	0.988	0.982
19	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.994	0.991
20	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.998	0.997	0.995
21	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998
22	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999
23	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Newsvendor Problems

The story is about newspapers which are only valuable on a single day. We can also apply the same argument to the sale of fruit and vegetables that perish in a short period of time or other perishable items.

Newsvendor Model

Assume $k = 0$ and there is a single time period, demand is stochastic and the item is perishable after one period. Demand D is random with mean $\mu = E(D)$ and variance $Var(D)$, c is the unit cost of the item to the vendor and $p > c$ the selling price, $s < c$ is a salvage value which may be zero or even negative if there is a cost of disposing of excess items.
Q is the number of units ordered. $\min\{Q, D\}$ the number of units sold. $\max\{Q - D, 0\}$ units are leftover or salvaged.

$$\pi(Q) = pE\min\{Q, D\} + sE\max\{Q - D, 0\} - cQ$$

How many items Q should be ordered?

Use $\min\{Q, D\} = D - \max\{D - Q, 0\}$

$$\pi(Q) = (p - c)\mu - G(Q)$$

where

$$G(Q) = (c-s)E\max\{Q - D, 0\} + (p-c)E\max\{D - Q, 0\} \geq 0$$

This can be rewritten in terms of probabilities using an indicator function and differentiating (remember the dirac function is the derivative of the Heaviside function and we can approximate max with a Heaviside function, an indicator function is a dirac measure with value 1.)

$$G'(Q) = (c - s)E\mathbb{1}_{Q>D} - (p - c)E\mathbb{1}_{Q>D}$$

Profit maximization now becomes cost minimization, i.e. minimize G.

$$G'(Q) = (c - s)Pr\{Q > D\} - (p - c)Pr\{Q > D\} = 0$$

rearrange to obtain

$$F(Q) = Pr\{D \leq Q\} = \frac{p - c}{p - s}$$

Q can be found by finding $Q = F^{-1}(\frac{p-c}{p-s})$
different distributions can be used normal, Poisson, log-normal etc.

Normal distribution

$$D = \mu + \sigma Z$$

Example on board.

Conclusion

- Many extensions of stochastic inventory problems are possible. this is an active area of interest.
- Many agribusiness inventory management problems involve storage of perishable items
- Next we move on to scheduling