MGMT-3453-X20: Homework 15-17

Bill Ash (5555WRA)

4/20/2021

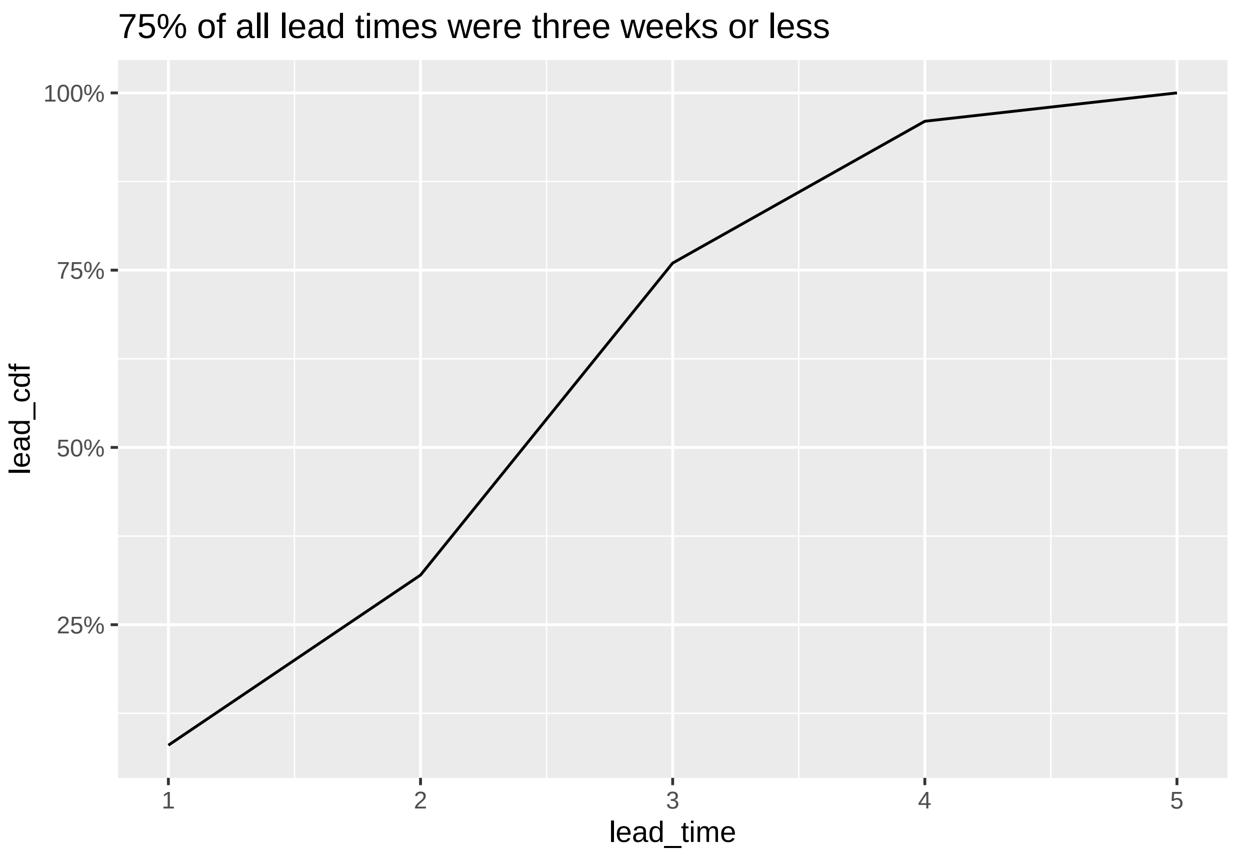
Simulation (LONG PROBLEM)

Order cost == shipping: the extra cost associated with processing the order. - There is no carry cost if there is no inventory - If there is 0 units there is no stock out costs - If there is 0 units and we sell a unit there IS A stock out cost.

```
# Simulation problem: start with no open orders

# $1/ Unit / Week on the ending balance
order_cost = 10

# $3/ Unit/ Week
carry_cost = 1

stock_out_cost = 3

# Current stock Units
balance_on_hand = 12

# Reorder point
order_point = 7

# EOQ
order_quantity = 8
```

Lead time analysis - *the lead time is not constant*

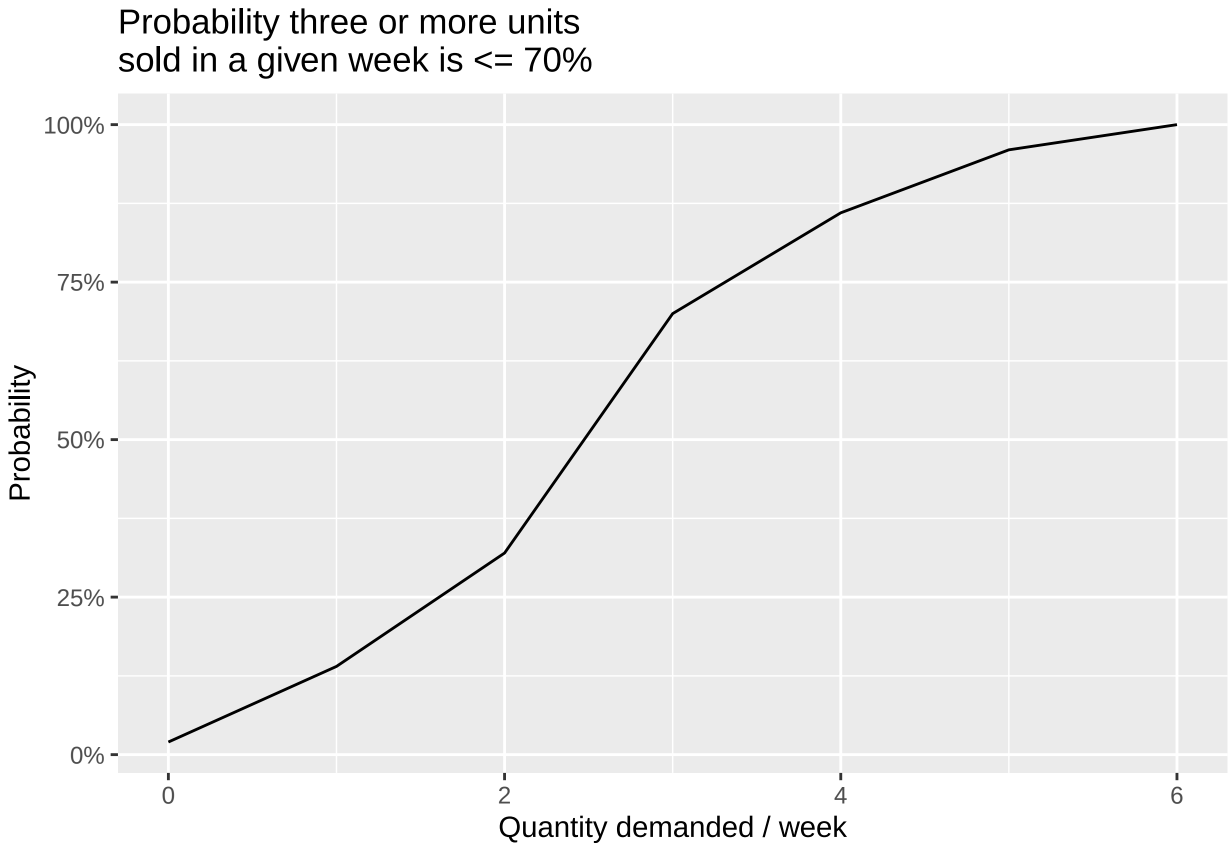
```
# Simulate the various lead times
lead_times <- tibble(
  lead_time = 1:5,
  lead_frequency = c(2, 6, 11, 5, 1),
  # probability the lead time will be 1 week or 5 weeks.
  lead_probability = lead_frequency / sum(lead_frequency),
  lead_cdf = cumsum(lead_probability)
)

lead_times %>%
  ggplot(aes(lead_time, lead_cdf)) +
  geom_line() +
  scale_y_continuous(labels = scales::percent_format()) +
  labs(title = '75% of all lead times were three weeks or less')
```



Demand analysis: *Demand is not always constant*

- Have we ever had a situation in which we sold zero units: once

```
# Historical data
demand_analysis <- tibble(
  # Units sold in a given week
  demand_week = 0:6,
  demand_frequency = c(1, 6, 9, 19, 8, 5, 2),
  demand_probability = demand_frequency / sum(demand_frequency),
  demand_cdf = cumsum(demand_probability)
)

demand_analysis %>%
  ggplot(aes(demand_week, demand_cdf)) +
  geom_line() +
  scale_y_continuous(labels = scales::percent_format()) +
  labs(title = 'Probability three or more units\nsold in a given week is <= 70%',
  x = 'Quantity demanded / week', y = 'Probability')
```



Reorder point question

How many do we have on hand / on order

- `quantity on hand + quantity on order <= Reorder Point`
  - Yes, place an order
  - No, don't place an order

SIMULATION EXPLANATION

```
# Order point is 7: the point at which we need to order new product
# Starting quantity on hand is 12.
# Week one we do not need to order.

tibble(
  week = 1:10,
  # Randomly chosen lead time 1:5 weeks
  # LeadTimeRandom = hash function for choosing the lead time
  # Randomly chosen demand 0:6 units / week
  # DemandNumberRandom = hash function for choosing quantity demanded
  QuantityDemanded = c(1, 3, 2, 4, 2, 1, 3, 3, 0, 4),
  # Starting quantity of 12 - 1 = 11 for week one
  QuantityOnHand = c(11, 8, 6, 2, 0, -1, 4, 1, 9, 5),
  # Order when TotalQuantity <= 7
  QuantityOrdered = c(0, 0, 0, 8, 0, 0, 8, 0, 0, 0),
  # Lead time probability is used to determine how long the order is in transit
    # Could be a value 1:6.
          OnOrder = c(0, 0, 0, 0, 8, 8, 0, 8, 0, 0),
  # Product is received two weeks later
  QuantityReceived = c(0, 0, 0, 0, 0, 0, 8, 0, 8, 0),
  TotalQuantity = QuantityOnHand + QuantityOrdered + OnOrder,
  OrderCost = ifelse(QuantityOrdered > 0, 10, 0),
  # Carry cost is $1/ unit on ending balance
  CarryCost = ifelse(QuantityOnHand > 0, QuantityOnHand * carry_cost, 0),
  StockOutCost = ifelse(QuantityOnHand < 0, 3, 0),
  TotalCost = OrderCost + StockOutCost + CarryCost,
  CumulativeCost = cumsum(TotalCost)
) %>%
  knitr::kable()
```

week	QuantityDemanded	QuantityOnHand	QuantityOrdered	OnOrder	QuantityReceived	TotalQuantity	OrderCost	CarryCost	StockOutCost	TotalCost	CumulativeCost
1	1	11	0	0	0	11	0	11	0	11	11
2	3	8	0	0	0	8	0	8	0	8	19
3	2	6	0	0	0	6	0	6	0	6	25
4	4	2	8	0	0	10	10	2	0	12	37
5	2	0	0	8	0	8	0	0	0	0	37
6	1	-1	0	8	0	7	0	0	3	3	40
7	3	4	8	0	8	12	10	4	0	14	54
8	3	1	0	8	0	9	0	1	0	1	55
9	0	9	0	0	8	9	0	9	0	9	64
10	4	5	0	0	0	5	0	5	0	5	69

- If the Total Quantity is less than or equal to 7, place an order.
  - The order will have a lead time based on the lead time analysis done

Purpose is to balance stock outs, reorder point, and carry costs to reduce the cumulative total cost of managing inventory.

End simulation

Example MRP scheduling problem

What is the projected available balance in week number 1?

```
# Batch Size when placing orders
q = 20
# Lead time
lead_time = 1
# Safety Stock
safety_stock = 0
# Starting balance
starting_balance = 21

# Work problem
tibble(
  product = 'parta',
  week = 1:6,
  # Given part of the problem
  gross_requirements =        c(5, 15, 18, 8, 12, 22),
  # Starting balance (21) - gross requirements
  ProjectedAvailableBalance = c(16, 1, 3, 15, 3, 1),
  # Need to cover the gross requirements for the subsequent week
  PlannedOrder =              c(0, 20, 20, 0, 20, 0),
  # One week lead time stated in problem
  ScheduledReceived =         c(0, 0, 20, 20, 0, 20)
) %>%
  knitr::kable()
```

product	week	gross_requirements	ProjectedAvailableBalance	PlannedOrder	ScheduledReceived
parta	1	5	16	0	0
parta	2	15	1	20	0
parta	3	18	3	20	20
parta	4	8	15	0	20
parta	5	12	3	20	0
parta	6	22	1	0	20

If we want to reduce the number of stock outs, we can increase the quantity ordered.

We calculate the gross requirements for sub-assembly b by looking at the planned orders for product a.

- Sub-assemblies are derived demand from the finished product. We only need to build sub-assemblies when product a is ordered.
  - Planned order in week 2, 3, and 5 will have gross requirements of:

```
b requirements 20 (planned order a) * 2 (each a requires two b sub-assemblies)
```

```
# Batch size 40
q = 40
# Two week lead time
lead_time = 2

tibble(
  product = 'partb',
  week = 1:6,
  # Calculated by looking at the Planned order field.
  gross_requirements =        c(0, 40, 40, 0, 40, 0),
  # Starting balance (20) + scheduled to receive 32 in first week & 0 gross req.
  ProjectedAvailableBalance = c(52, 12, 12, 12, 12, 12),
  # Need to cover the gross requirements for the subsequent week
  PlannedOrder =              c(40, 0, 40, 0, 0, 0),
  # Two week lead time stated in problem
  ScheduledReceived =         c(32, 0, 40, 0, 40, 0)
) %>%
  knitr::kable()
```

product	week	gross_requirements	ProjectedAvailableBalance	PlannedOrder	ScheduledReceived
partb	1	0	52	40	32
partb	2	40	12	0	0
partb	3	40	12	40	40
partb	4	0	12	0	0
partb	5	40	12	0	40
partb	6	0	12	0	0

Part c: to get the gross requirements - multiply the planned order releases for part a and c by 1..

```
# Lot for lot: order what is required
q = 'LotForLot'
# One week lead time
lead_time = 1
# Maintain safety stock of 10: Add 10 to gross_requirements
a = c(0, 20, 20, 0, 20, 0)
b = c(40, 0, 40, 0, 0, 0)
c_gross_requirements  = a + b

tibble(
  product = 'partc',
  week = 1:6,
  # Calculated by looking at the Planned order field.
  gross_requirements =  c_gross_requirements, # 40, 20, 60, 0, 20, 0
  # Starting balance (50) + scheduled to receive 32 in first week & 0 gross req.
  ProjectedAvailableBalance = c(10, 10, 10, 10, 10, 10),
  # Need to cover the gross requirements for the subsequent week
  PlannedOrder =              c(20, 60, 0, 20, 0, 0),
  # Two week lead time stated in problem
  ScheduledReceived =         c(0, 20, 60, 0, 20, 0)
) %>%
  knitr::kable()
```

product	week	gross_requirements	ProjectedAvailableBalance	PlannedOrder	ScheduledReceived
partc	1	40	10	20	0
partc	2	20	10	60	20
partc	3	60	10	0	60
partc	4	0	10	20	0
partc	5	20	10	0	20
partc	6	0	10	0	0