# MODULE FOUR PROJECT

## A PRESCRIPTIVE MODEL FOR STRATEGIC DECISION-MAKING : AN INVENTORY MODEL

Submitted By : **HARSHIT GAUR**
MASTER OF PROFESSIONAL STUDIES IN ANALYTICS
ALY 6050 : INTRODUCTION TO ENTERPRISE ANALYTICS
FEBRUARY 7, 2022

Submitted To : **PROF. RICHARD HE**

# ABSTRACT

The importance of making good decisions is well recognized in business, and most people view decision making as an act of rationality. Many decision making situations have become more complicated in recent years, and assistance of some kind is often needed.

In essence, a Decision Model is a template for thinking about, organizing, and managing the business logic that underlies a business decision. The business logic is the way in which business analysts evaluate facts in order to reach conclusions that have both meaning and value to the business. Although there are several decision models available, the process and outcome of decision making must be considered along with certain important requirements or a different kind of mindset.

There are 3 modes of decision making :

- Descriptive : What people do or have done.
- Prescriptive : What people can do.
- Perspective/Normative : What people should do.

**Prescriptive models** are ones that a real decision maker can and should use, and that are tailored to both the specific situation and needs of the decision maker. Both normative theory and descriptive theory provide the theoretical foundation for prescriptive models

# INTRODUCTION

Using this project, we will be performing *Prescriptive Modelling for a Strategic Decision Model - An Inventory*.

## PROBLEM STATEMENT

A company's inventories represent a significant investment; therefore, it is vital that they are managed well. Excess inventories may be a sign of poor financial and operational management.

It is essential for managers to decide how much inventory they should order and when that inventory should be ordered to minimize the total inventory cost, which includes the cost of holding inventory and the cost of ordering it from the supplier.

The below table provides the information related to the inventory.

| | |
|---|---|
| **Annual Demand** | 16000 |
| **Unit Cost** | $77 |
| **Unit Holding Cost / Opportunity Cost** | 17% |
| **Ordering Cost** | $222 |

*Figure 1.1: The information related to the inventory*

As a consultant, I must develop and implement a decision model to aid this client in reaching the best decision.

# Part 1

*I - Information about the inventory of the company.*

- The Annual Demand of the inventory is 16,000 which is assumed to be constant throughout the year.

- A unit costs at $77.

- The holding cost or opportunity cost of a unit is 17% of the unit cost price of that inventory product.

- The cost associated with replenishing the inventory or the ordering cost is $222.

- A policy of the company states that whenever the inventory level reaches a predetermined reorder point which is sufficient to meet the demand until the order gets shipped and received, the new order should be twice the amount of current number of units present in the inventory.

The data and the terminology to define the data is represented as :

- ***Uncontrollable Variables :***
  - Annual Demand

- ***Model Parameters :***
  - Unit Cost
  - Unit Holding Cost (Opportunity Cost)
  - Ordering Cost

- ***Decision Variables (Controllable) :***
  - Predetermined Reorder Point
  - Order Quantity

- ***Objective :***
  - Total Costs

Formula used to compute the "Total Costs" of the inventory of the company -

*Total Costs = Total Holding Cost + Total Ordering Cost*

| | | |
|---|---|---|
| **Annual Demand** | 16000 | Uncontrollable Variable |
| **Unit Cost** | $77 | Model Parameter |
| **Unit Holding Cost / Opportunity Cost** | 17% | Model Parameter |
| **Ordering Cost** | $222 | Model Parameter |
| | | |
| **Predetermined Reorder Point** | 260.46 | Decision Variable |
| **Order Quantity** | 520.91 | Decision Variable |
| | | |
| **Total Holding Cost** | $6,818.77 | |
| **Total Ordering Cost** | $6,819 | |
| | | |
| **Total Costs** | $13,637.55 | Objective |

*Figure 1.2: Variables, Model Parameters, Objective of the inventory problem.*

## *II - Mathematical Functions to compute the Annual Ordering Cost & Annual Holding Cost.*

- ***Annual Ordering Cost :***
  - There are 2 components present to calculate the Ordering Cost.
  - 1st component relates to the number of times orders are placed.
  - 2nd component relates to the cost associated with a single order.

***Annual Ordering Cost*** *= Ordering Cost of 1 time* ***X*** *Number of times orders placed*

- ***Annual Holding Cost :***
  - There are 3 components involved to calculate the Holding Cost.
  - 1st component related to the holding cost of the number of items placed in a single order.

    ***Holding cost of no. of items*** *= Unit Cost* ***X*** *Unit Holding Cost* ***X*** *Order Quantity*

  - 2nd component relates to the number of days the new-order quantity needs to sell all the units (since it is constant throughout the year).

    ***No. of days to sell all new order quantity*** *= (Order Quantity / Annual Demand) / 365 (days)*

  - 3rd component relates to the number of times orders are placed.

    ***No. of times order placed in a year****= (Annual Demand / Order Quantity) / 365 (days)*

  - By the combination of the above components, Annual Holding Cost becomes :

***Annual Holding Cost*** *= Holding cost of no. of items* ***X*** *No. of days to sell all new order quantity* ***X*** *No. of times order placed in a year*

### *III - Using Solver to minimize Total Cost and find the Order Quantity.*

We can minimize the objective by playing around the decision variable using Excel's Solver function.

| | | |
|---|---|---|
| **Predetermined Reorder Point** | 260.46 | Decision Variable |
| **Order Quantity** | 520.91 | Decision Variable |

*Figure 1.3: Values of Decision Variables computed using Excel's Solver function.*

| | | |
|---|---|---|
| **Total Costs** | $13,637.55 | Objective |

*Figure 1.4: Values of Objective computed using Excel's Solver function.*

- The **Predetermined Reorder Point** comes out to be **260.46.**
- It means the when the inventory level reaches around **260** in quantity, a new order can be placed with twice the current quantity in the inventory.

- The **Order Quantity** gets calculated to **520.91** which signifies the approximate number of items to order for the total cost of the inventory to be low.

- In a similar way, we can find the total cost when different order quantity is placed and plot a scatterplot for analysis.

| Order Quantity | Total Cost |
|---|---|
| 520.91 | $13,637.55 |
| 100 | $36,829 |
| 120 | $31,171 |
| 140 | $27,204 |
| 160 | $24,294 |
| 180 | $22,090 |
| 200 | $20,378 |
| 220 | $19,025 |
| 240 | $17,942 |
| 260 | $17,065 |
| 280 | $16,351 |
| 300 | $15,767 |
| 320 | $15,289 |
| 340 | $14,898 |
| 360 | $14,579 |
| 380 | $14,322 |
| 400 | $14,116 |
| 420 | $13,955 |
| 440 | $13,832 |
| 460 | $13,743 |
| 480 | $13,683 |
| 500 | $13,649 |
| 520 | $13,638 |
| 540 | $13,646 |
| 560 | $13,673 |
| 580 | $13,716 |

*Figure 1.5: Computation of Total Cost differing according to the Order Quantity.*


*Figure 1.6: Scatterplot of Total Cost vs Total Quantity.*

1. The scatterplot above depicts that the curve of **'Total Cost' has a Global Minima** in the range of **400 - 600 (Order Quantity).**

2. The Solver function minimized the **Total Cost** value to around **$13,673.55**

*IV - R language was also used to cross-verify our above findings.*

```
> order_qty <- seq(100,8000,by=100)
> total_cost <- (order_qty * ((unit_cost*unit_holding_cost)/no_of_days) * (order_qty/(annual_demand/no_of_days)) *
 (annual_demand/order_qty)) +
+               (ordering_cost * (annual_demand/order_qty))
```

*Figure 1.7: R script to calculate and plot the scatterplot for order quantity's sequence 100 to 8000.*

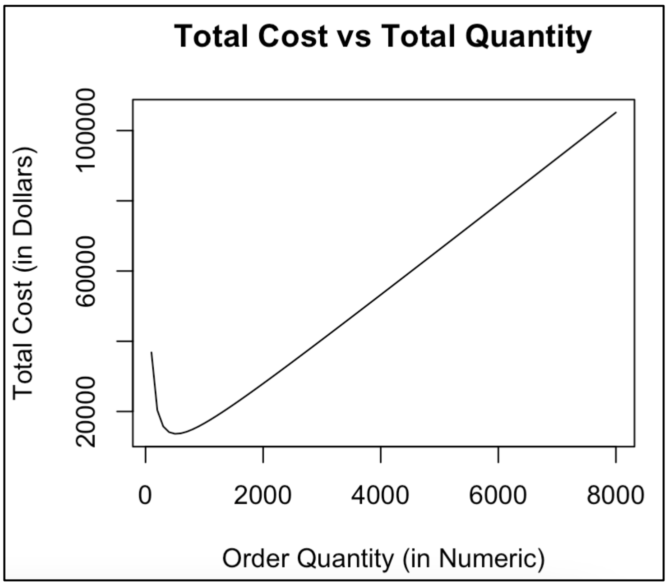


*Figure 1.8: Scatterplot of Total Cost vs Order Quantity for 100 to 8000.*

- This scatterplot depicts that the graph has a minima (most probably a global minima as we have taken in consideration quantities up to 8000 from the available 160000).
- This graph shows that the minima lies in the range of 200 to 700 on x-axis.
- We will now improve our search range.

```
> # Zooming In:
> order_qty <- seq(300,700,by=20)
> total_cost <- (order_qty * ((unit_cost*unit_holding_cost)/no_of_days) * (order_qty/(annual_demand/no_of_days)) *
 (annual_demand/order_qty)) +
+               (ordering_cost * (annual_demand/order_qty))
```

*Figure 1.9: R script to zoom in and improve search range for scatterplot*

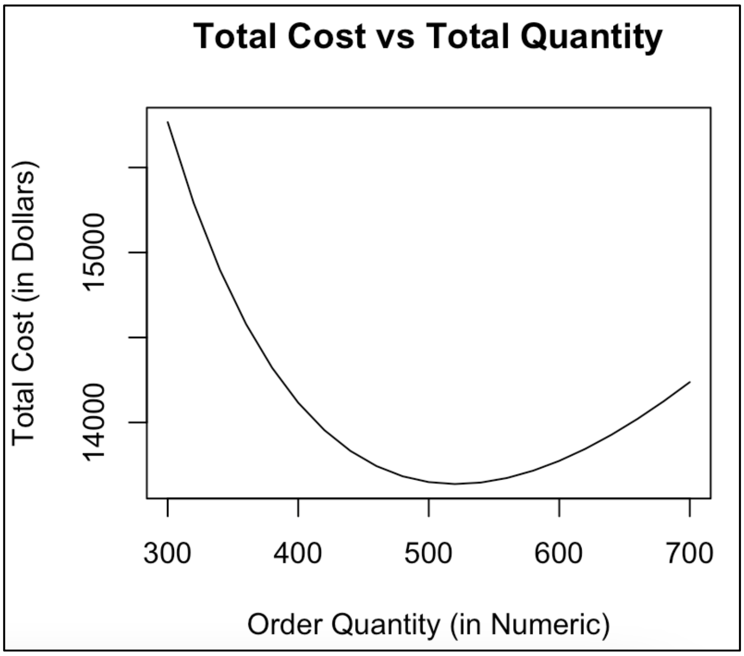


*Figure 1.10: Scatterplot of Total Cost vs Order Quantity for 300 to 700.*

- This scatterplot depicts that the graph has a minima in around 510 - 525 for order qty.
- The minimized Total Cost is around $13,600 - $13,700

## V - What-If Analysis (Sensitivity Analysis).

| | | | Sensitivity Analysis with Model Parameters | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Effect of Unit Cost | | | | | | | |
| | $13,637.55 | $20.00 | $40.00 | $60.00 | $77.00 | $90.00 | $120.00 | $180.00 | $290.00 |
| Effect of Ordering Cost | 100 | 4842.63001 | 6613.74016 | 8384.85031 | 9890.29394 | 11041.5155 | 13698.1808 | 19011.5112 | 28752.6171 |
| | 140 | 6071.23795 | 7842.3481 | 9613.45825 | 11118.9019 | 12270.1235 | 14926.7887 | 20240.1192 | 29981.225 |
| | 180 | 7299.84589 | 9070.95605 | 10842.0662 | 12347.5098 | 13498.7314 | 16155.3967 | 21468.7271 | 31209.8329 |
| | 222 | 8589.88423 | 10360.9944 | 12132.1045 | 13637.5482 | 14788.7698 | 17445.435 | 22758.7654 | 32499.8713 |
| | 240 | 9142.75781 | 10913.868 | 12684.9781 | 14190.4217 | 15341.6433 | 17998.3086 | 23311.639 | 33052.7449 |
| | 280 | 10371.3658 | 12142.4759 | 13913.5861 | 15419.0297 | 16570.2513 | 19226.9165 | 24540.247 | 34281.3528 |
| | 400 | 14057.1896 | 15828.2997 | 17599.4099 | 19104.8535 | 20256.0751 | 22912.7403 | 28226.0708 | 37967.1766 |
| | 600 | 20200.2293 | 21971.3394 | 23742.4496 | 25247.8932 | 26399.1148 | 29055.7801 | 34369.1105 | 44110.2163 |

*Figure 1.11: What-If Analysis (Sensitivity Analysis) using 2 model parameters.*

- The What-If analysis provides some of the below observations -
  - If unit cost is taken as $20 and the ordering cost is taken as $100, the Total Cost would be around $4842.63
  - If unit cost is taken as $60 and the ordering cost is taken as $240, the Total Cost would be around $12132.10
  - If unit cost is taken as $180 and the ordering cost is taken as $222, the Total Cost would be around $22758.76
  - If unit cost is taken as $290 and the ordering cost is taken as $600, the Total Cost would be around $44110.21

- As the unit cost rises up, the Total Cost also rises up.
- In a similar fashion, the Total Cost also rises up as the Ordering Cost also rises up.

To the Vice President of the Operations, I would say that whenever the inventory level reaches around 260, the team would need to place a new order of double this amount (i.e., around 520) to be able to minimize the Annual Total Cost beared by the company which would be around $13,637.

# Part 2

The Annual Demand has a triangular probability distribution between 13,000 and 18,000. It has a mode of around 16,000. Simulations are run consisting of 1000 occurences to calculate the followings.

### *I - Expected Minimum Total Cost*

The lower demand value, upper demand value, and peak demand value have been initialised and 1000 random numbers have been generated for simulation.

For the calculations involved in Triangular Probability Distribution, K-M-N values have been computed using formulas involved. The triangular distribution simulation function has been populated using the formula present in the respective screenshot.

```
lower_demand_val <- 13000
upper_demand_val <- 18000
peak_demand_val <- 16000

# Random Number Generation for simulation
rand_simulation <- runif(1000)

# Triangle Distribution Method Formula
# K = (c - a) / (b - a)
# M = (b - a) * (c - a)
# N = (b - a) * (b - c)

K <- (peak_demand_val - lower_demand_val) / (upper_demand_val - lower_demand_val)
M <- (upper_demand_val - lower_demand_val) * (peak_demand_val - lower_demand_val)
N <- (upper_demand_val - lower_demand_val) * (upper_demand_val - peak_demand_val)
```

*Figure 2.1: Initialisation of values involved in Triangular Probability Distribution.*

```
# If  r <= K, x = a + sqrt(r * M), Else x = b - sqrt((1 - r) * N)

triangular_dist_simulation <- ifelse(rand_simulation <= K,
                              round(lower_demand_val + sqrt(rand_simulation * M), 0),
                              round(upper_demand_val - sqrt((1 - rand_simulation) * N), 0))
```

*Figure 2.2: Calculation of the simulation distribution using formula.*

A for-loop is created which computes the total cost using **R's optimise()** function in order to populate our data frame of simulation results within their respective fields/columns (like, *Demand Value, Order Quantity, Minimum Total Cost, Annual Number of Orders*).

```
> summary(simulation_df)
  Demand Value    Order Quantity   Minimum Total Cost Annual Number of Orders
 Min.   :13080   Min.   :471.0   Min.   :12330      Min.   :27.77
 1st Qu.:15018   1st Qu.:504.7   1st Qu.:13212      1st Qu.:29.76
 Median :15786   Median :517.4   Median :13546      Median :30.51
 Mean   :15729   Mean   :516.2   Mean   :13515      Mean   :30.44
 3rd Qu.:16425   3rd Qu.:527.8   3rd Qu.:13817      3rd Qu.:31.12
 Max.   :17908   Max.   :551.1   Max.   :14428      Max.   :32.50
```

*Figure 2.3: Summary of the simulation data frame with 4 fields/columns.*

- The expected minimum total cost is around **$12,330** , but the mean is around **$13,515**.
- The expected order quantity would be mean of the field 'Order Quantity' which is around **516.2**
- The expected annual number of orders is around **30** according to the observations we've found from the calculations.

### *II - Testing / Checking the Best Fit of the distributions.*

To determine / test / check the best fit of the probability distributions in the above 3 fields, we can use a library named " **fitdistrplus**". This library provides function to plot different type of graphs like -

- Empirical & Theoretical Density Function Plot
- Q-Q Plot
- P-P Plot
- Empirical & Theoretical Cumulative Density Function (CDF) Plot

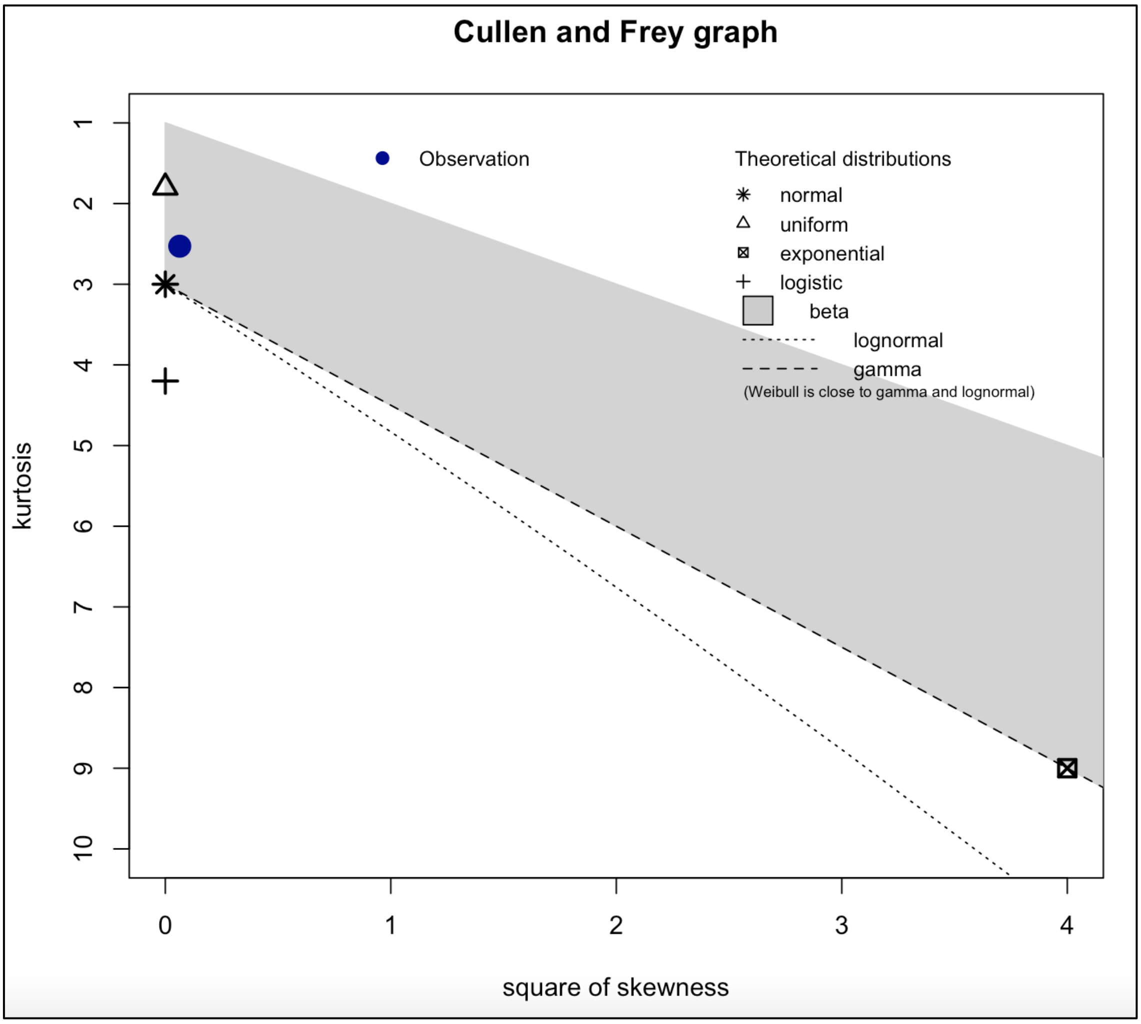


*Figure 2.4: Cullen and Grey Graph to showcase the possibilities of types the distribution can be.*

- The kurtosis & skewness is plotted and it seems like the distribution can be either **Normal, Weibull, or Gamma distribution.**
- Let's check further to eliminate the possibilities.

```
# Fit Test for 'Order Quantity'
fit.weibull <- fitdist(simulation_df$`Order Quantity`, "weibull")
fit.gamma <- fitdist(simulation_df$`Order Quantity`, "gamma")
fit.norm <- fitdist(simulation_df$`Order Quantity`, "norm")
fit.unif <- fitdist(simulation_df$`Order Quantity`, "unif")
```

*Figure 2.5: The distribution of 'Order Quantity' is fitted according to types of distribution*

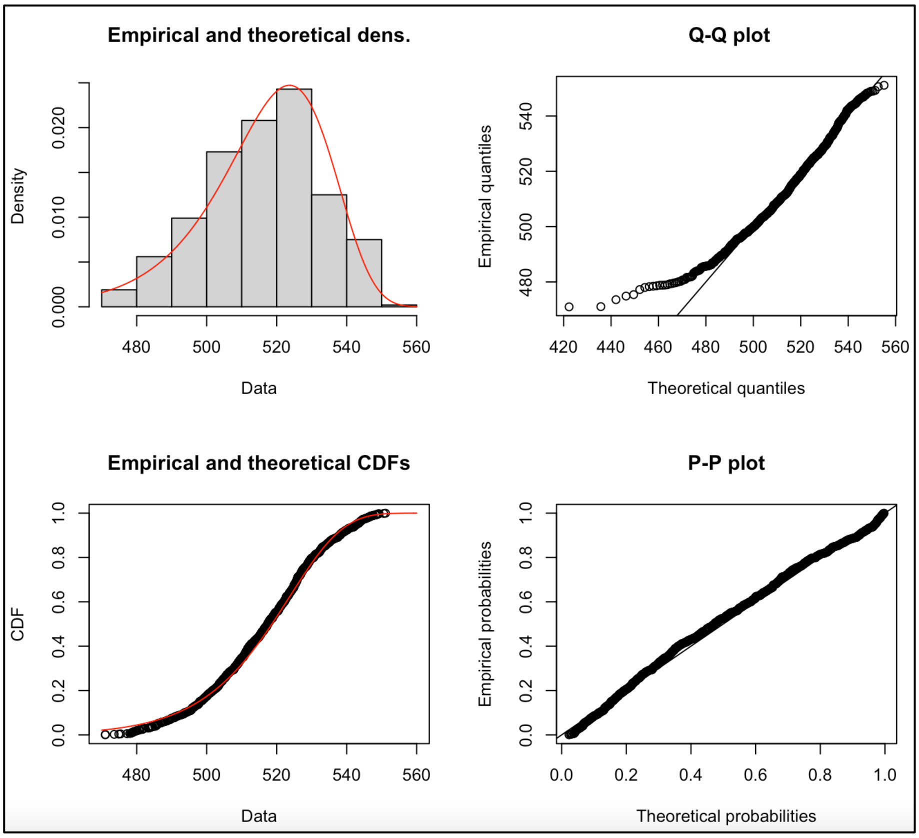


*Figure 2.6: **WEIBULL** distribution of 'Order Quantity'*

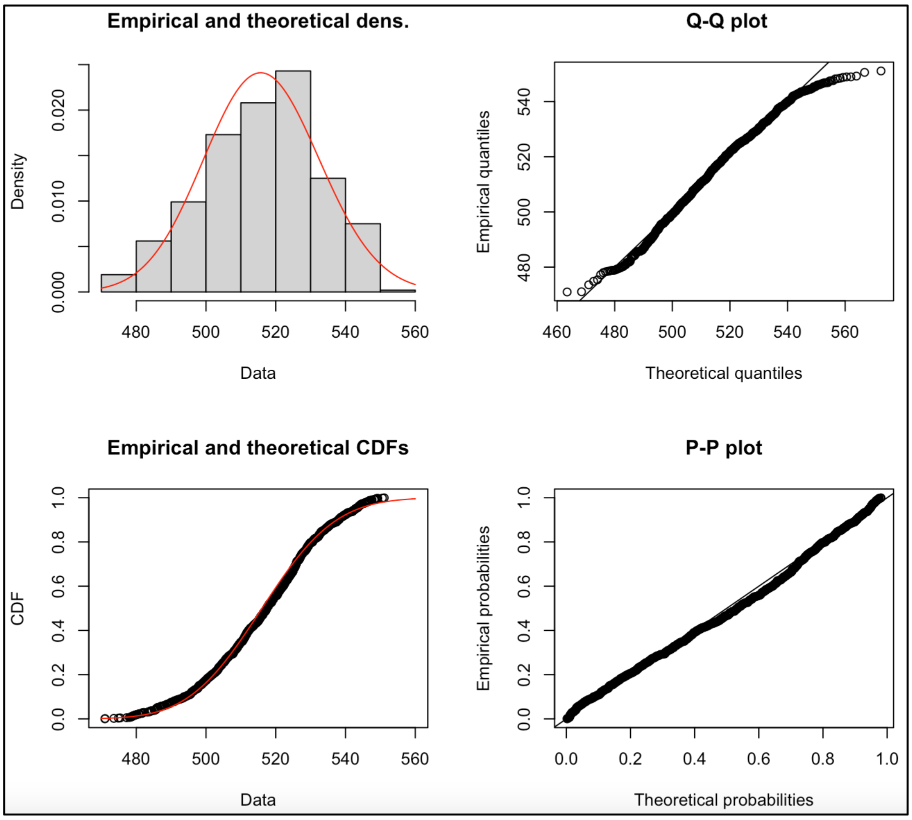


*Figure 2.7: **GAMMA** distribution of 'Order Quantity'*

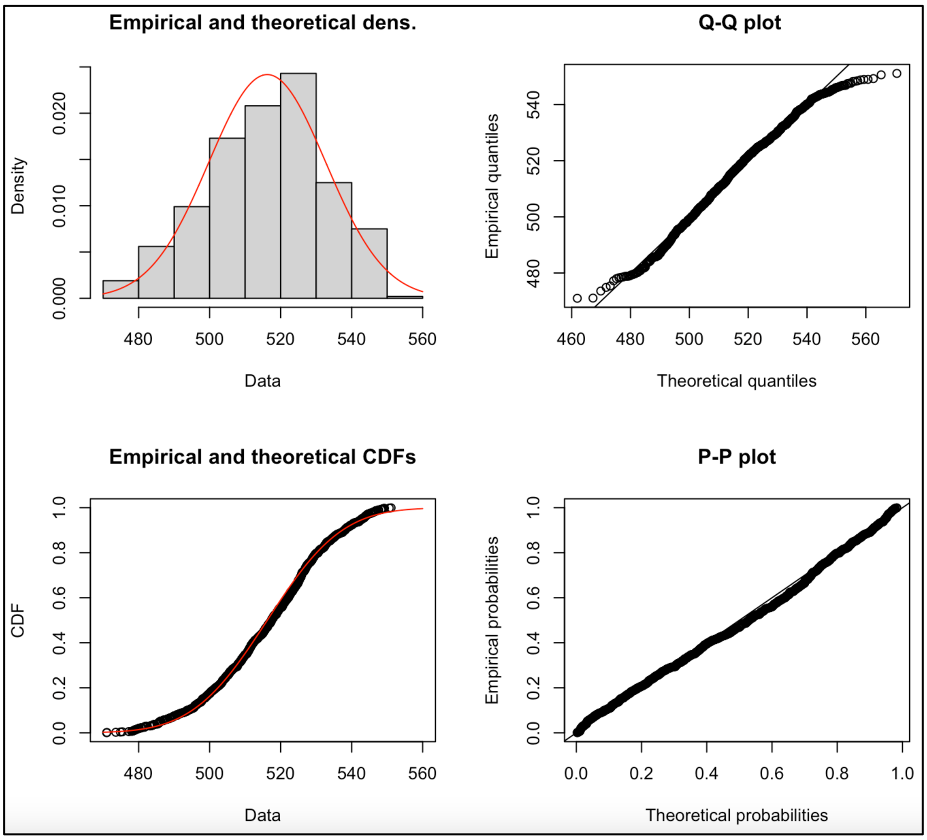


*Figure 2.8: **NORMAL** distribution of 'Order Quantity'*

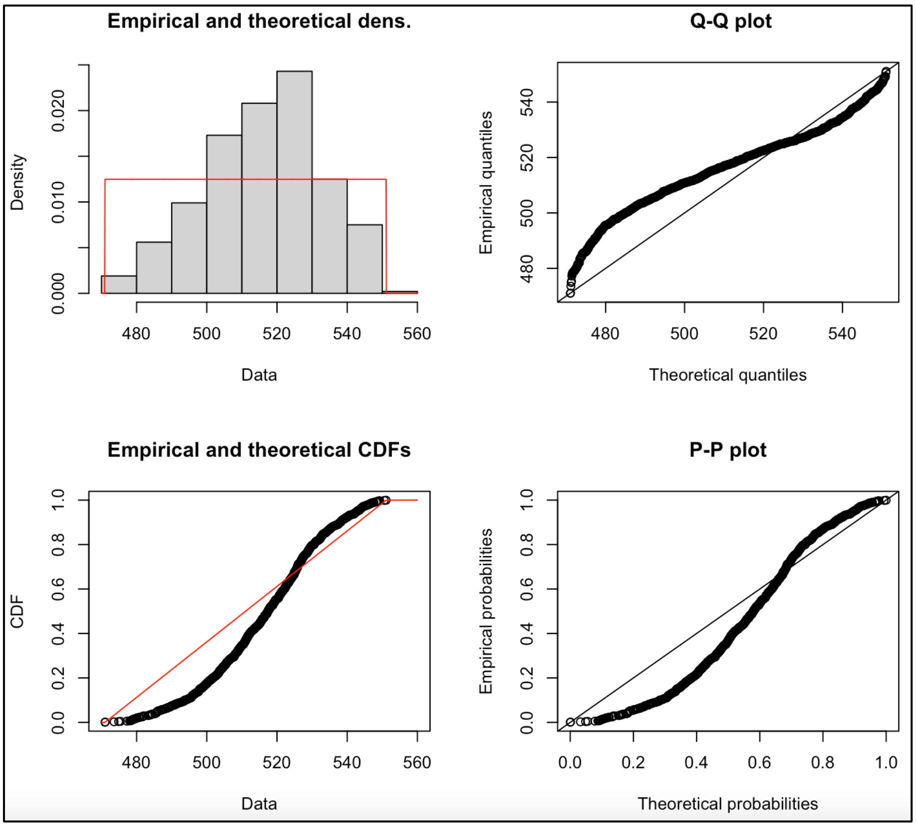


*Figure 2.9: **UNIFORM** distribution of 'Order Quantity'*

- The plots belonging to 4 types of distribution shows that the "Order Quantity" distribution is likely to be a **Normal Distribution**. The reasons being either the QQ-plot is deviating or density function curve is not following up properly with the type.

```
> shapiro.test(simulation_df$`Order Quantity`)

	Shapiro-Wilk normality test

data:  simulation_df$`Order Quantity`
W = 0.98826, p-value = 0.0000003565
```

*Figure 2.10: **Shapiro-Wilk's Test** for the distribution of 'Order Quantity'*

- The Shapiro-Wilk's test for the distribution of 'Order Quantity' also proves that the normality of the distribution is present. In other words, the distribution is normal in nature.

The same can be proven for the other 2 fields -

- Minimum Total Cost
- Annual Number of Orders

```
# Fit Test for 'Minimum Total Cost'
fit.weibull <- fitdist(simulation_df$`Minimum Total Cost`, "weibull")
fit.gamma <- fitdist(simulation_df$`Minimum Total Cost`, "gamma")
fit.norm <- fitdist(simulation_df$`Minimum Total Cost`, "norm")
fit.unif <- fitdist(simulation_df$`Minimum Total Cost`, "unif")

plot(fit.weibull)
plot(fit.gamma)
plot(fit.norm)
plot(fit.unif)

# Fit Test for 'Annual Number of Orders'
fit.weibull <- fitdist(simulation_df$`Annual Number of Orders`, "weibull")
fit.gamma <- fitdist(simulation_df$`Annual Number of Orders`, "gamma")
fit.norm <- fitdist(simulation_df$`Annual Number of Orders`, "norm")
fit.unif <- fitdist(simulation_df$`Annual Number of Orders`, "unif")

plot(fit.weibull)
plot(fit.gamma)
plot(fit.norm)
plot(fit.unif)
```

*Figure 2.11: **Fit-Test** for the other 2 distributions.*

We can convey to the Vice President of the Operations that the expected values for *Order Quantity, Minimum Total Cost, and Annual Number of Orders* have been estimated and the type of all the 3 distributions comes out to be Normal Distribution.

# CONCLUSION

We have used the prescriptive model of the data decision modelling techniques to evaluate and analyse the strategic problem - An Inventory.

- We analysed using the mathematical formulas which calculated the Total Ordering Costs and Total Holding Costs and found out the Total Costs formula in order to apply optimization on it.
- We minimized *the Total Cost (Objective Function*) by using the *Pre-Determined Reorder Point Quantity (Decision Variable).*
- The **Reorder Point** comes out to be around **260**.
- The **Order Quantity** needs to be double its value. Therefore, it comes out to be around **520**.
- The **minimized Total Cost** comes out to be around **$13,673**.
- We performed What-If (Sensitivity Analysis) using the model parameters (*Unit Cost, Ordering Cost*) to have the chart ready for the Vice President of the Operations.
- The expected values of -
    - Order Quantity - 516.2
    - Minimum Total Cost - **$12,330** with mean at around **$13,515**.
    - Annual Number of Orders - **30**
- The determination of best fit of the probability distributions came out to be **Normal Distributions** for all the **3 distributions**.
- We can convey to the Vice President of the Operations that to minimize the total cost to be beared by the company for orders,
    - **Pre-Determined Reorder Point** should be at around **260**.

# BIBLIOGRAPHY

1. Microsoft. (2021). *Excel functions (alphabetical)*. https://support.microsoft.com/en-us/office/excel-functions-alphabetical-b3944572-255d-4efb-bb96-c6d90033e188

2. *Triangular Distribution*. (2021, June 29). Real Statistics Using Excel. https://www.real-statistics.com/other-key-distributions/uniform-distribution/triangular-distribution/

3. ALY 6050 - Prof Richard He - *Lesson 4-2 -- Prescriptive Models* (2022, January), https://northeastern.instructure.com/courses/97784/pages/lesson-4-2-prescriptive-models?module_item_id=6650782

4. *How to determine which distribution fits my data best?* (2015, January 8). Cross Validated. https://stats.stackexchange.com/questions/132652/how-to-determine-which-distribution-fits-my-data-best

5. Stuart M. Dillon. Descriptive Decision Making: Comparing Theory with Practice. https://orsnz.org.nz/conf33/papers/p61.pdf

# APPENDIX

```
#-------- ALY6050_Module4Project_GaurH --------#

############################### PART 1 ###############################
# Annual Demand
annual_demand <- 16000

# Unit Cost
unit_cost <- 77

# Unit Holding Cost (Percentage converted to Fraction)
unit_holding_cost <- (17/100)

# Ordering Cost
ordering_cost <- 222

# No. of days in a year (considered a non-leap year)
no_of_days <- 365

# We wish to select the Order-Quantity that minimizes the Total Cost (Holding + Ordering).

# Let's first investigate how the Total-Cost changes with Order-Quantity
order_qty <- seq(100,8000,by=100)
total_cost <- (order_qty * ((unit_cost*unit_holding_cost)/no_of_days) *
(order_qty/(annual_demand/no_of_days)) * (annual_demand/order_qty)) +
        (ordering_cost * (annual_demand/order_qty))
plot(order_qty, total_cost, 'l', main = "Total Cost vs Total Quantity", xlab = "Order Quantity (in
Numeric)", ylab = "Total Cost (in Dollars)")

# From the plot, it looks like the minimum total cost is attained
# when order-quantity is approximately around 300 - 700

# Zooming In:
order_qty <- seq(300,700,by=20)
total_cost <- (order_qty * ((unit_cost*unit_holding_cost)/no_of_days) *
(order_qty/(annual_demand/no_of_days)) * (annual_demand/order_qty)) +
        (ordering_cost * (annual_demand/order_qty))
plot(order_qty, total_cost, 'l', main = "Total Cost vs Total Quantity", xlab = "Order Quantity (in
Numeric)", ylab = "Total Cost (in Dollars)")

# Minimum Total-Cost Value:
total_cost[which.min(total_cost)]

# Minimum Order-Quantity Value:
order_qty[which.min(total_cost)]

## Using the optimise() function to solve this minimization problem.
order_qty_interval <- c(450,600)

## Function related to total_cost~order_qty to use for optimise() function
total_cost_fn <- function(ord_qty) {
  return ((ord_qty * ((77*(17/100))/365) * (ord_qty/(16000/365)) * (16000/ord_qty)) + (222 *
(16000/ord_qty)))
}

## Use of optimise() function for the minimization problem.
optimise_cost_qty <- optimise(f = total_cost_fn, interval = order_qty_interval,
```

```
                            lower = min(order_qty_interval),
                            upper = max(order_qty_interval),
                            maximum = FALSE)

# Output the result of optimization (Minimum & Objective).
paste('Minimum Order Quantity : ', optimise_cost_qty$minimum)
paste('Objective Total Cost (Minimized) : ', optimise_cost_qty$objective)


############################### PART 2 ###############################

lower_demand_val <- 13000
upper_demand_val <- 18000
peak_demand_val <- 16000

# Random Number Generation for simulation
rand_simulation <- runif(1000)

# Triangle Distribution Method Formula
# K = (c - a) / (b - a)
# M        = (b - a) * (c - a)
# N = (b - a) * (b - c)

K <- (peak_demand_val - lower_demand_val) / (upper_demand_val - lower_demand_val)
M <- (upper_demand_val - lower_demand_val) * (peak_demand_val - lower_demand_val)
N <- (upper_demand_val - lower_demand_val) * (upper_demand_val - peak_demand_val)

# If  r <= K, x = a + sqrt(r * M), Else x = b - sqrt((1 - r) * N)

triangular_dist_simulation <- ifelse(rand_simulation <= K,
                              round(lower_demand_val + sqrt(rand_simulation * M), 0),
                              round(upper_demand_val - sqrt((1 - rand_simulation) * N), 0))

simulation_df <-data.frame(matrix(ncol = 4, nrow = 0))
col_names_simulation_df <- c("Demand Value", "Order Quantity", "Minimum Total Cost",
"Annual Number of Orders")
colnames(simulation_df) <- col_names_simulation_df

# Running the Simulation to find the Expected Values.
len_triangular_dist_sim <- length(triangular_dist_simulation)

for (j in 1:len_triangular_dist_sim) {
  optimise_cost_fn = function(ord_qty) {
    annual_demand <- triangular_dist_simulation[j]
    unit_cost <- 77
    ordering_cost <- 222
    unit_holding_cost <- 0.17

    total_cost <- (ordering_cost * (annual_demand/ord_qty)) +
             (ord_qty * ((unit_cost*unit_holding_cost)/no_of_days) *
(ord_qty/(annual_demand/no_of_days)) * (annual_demand/ord_qty))
  }

  order_qty_threshold <- c(250, 7500)

  cost_optimization <- optimise(f = optimise_cost_fn, interval = order_qty_threshold,
                              lower = min(order_qty_threshold),
                              upper = max(order_qty_threshold),
                              maximum = FALSE)
```

```
  simulation_df[j, "Demand Value"] = triangular_dist_simulation[j]
  simulation_df[j, "Order Quantity"] = cost_optimization$minimum
  simulation_df[j, "Minimum Total Cost"] = cost_optimization$objective
  simulation_df[j, "Annual Number of Orders"] =
triangular_dist_simulation[j]/cost_optimization$minimum
}

# Summary
summary(simulation_df)

# Testing the Best Fit of the distributions.
install.packages("fitdistrplus")
install.packages("logspline")
library(fitdistrplus)
library(logspline)

descdist(simulation_df$`Order Quantity`, discrete = FALSE)
descdist(simulation_df$`Minimum Total Cost`, discrete = FALSE)
descdist(simulation_df$`Annual Number of Orders`, discrete = FALSE)

# Fit Test for 'Order Quantity'
fit.weibull <- fitdist(simulation_df$`Order Quantity`, "weibull")
fit.gamma <- fitdist(simulation_df$`Order Quantity`, "gamma")
fit.norm <- fitdist(simulation_df$`Order Quantity`, "norm")
fit.unif <- fitdist(simulation_df$`Order Quantity`, "unif")

plot(fit.weibull)
plot(fit.gamma)
plot(fit.norm)
plot(fit.unif)

# Fit Test for 'Minimum Total Cost'
fit.weibull <- fitdist(simulation_df$`Minimum Total Cost`, "weibull")
fit.gamma <- fitdist(simulation_df$`Minimum Total Cost`, "gamma")
fit.norm <- fitdist(simulation_df$`Minimum Total Cost`, "norm")
fit.unif <- fitdist(simulation_df$`Minimum Total Cost`, "unif")

plot(fit.weibull)
plot(fit.gamma)
plot(fit.norm)
plot(fit.unif)

# Fit Test for 'Annual Number of Orders'
fit.weibull <- fitdist(simulation_df$`Annual Number of Orders`, "weibull")
fit.gamma <- fitdist(simulation_df$`Annual Number of Orders`, "gamma")
fit.norm <- fitdist(simulation_df$`Annual Number of Orders`, "norm")
fit.unif <- fitdist(simulation_df$`Annual Number of Orders`, "unif")

plot(fit.weibull)
plot(fit.gamma)
plot(fit.norm)
plot(fit.unif)

shapiro.test(simulation_df$`Order Quantity`)
shapiro.test(simulation_df$`Minimum Total Cost`)
shapiro.test(simulation_df$`Annual Number of Orders`)

#-------- END --------#
```