# Assignment 4

*Sri Seshadri*

*8/19/2018*

## Problem 1

### Linear Model

In this problem a linear model $y = ax + b$ is fit, where $y$ is Average Revenue and $x$ is Hours of operation. The strategy was to set $a$ and $b$ to be *decision variables* and the *minimize the mean absolute deviation* between the fitted model and the actual average revenue.

The *LSRG* solver in the educational license of ASPE was not available and hence a default non-linear optimization engine was chosen with multiple start points. The model set up is shown in figure 1.

The final model is $y = 4890 + 39.33x$, where where $y$ is Average Revenue and $x$ is Hours of operation.

*For 120 hours of operation the average revenue based on the linear model is $9610*

### Non -linear model

Here the model $y = ax^b$ is fit to the same data, the strategy here was similar to the above, where we minimize the mean absolute deviation between fitted and actuals.

The value of $a = 1113.03$ and $b = 0.46$; thus

$y = 1113.03x^{0.46}$ where $y$ is Average Revenue and $x$ is Hours of operation.

*For 120 hours of operation the average revenue based on the non linear model is $9897.29*

*Comparing the Mean absolute deviation of either models; 678.5 for linear and 571.65 for non linear model, the non linear model prediction seems ideal. However one needs to be mindful of overfitting.*

| Hours of Operation | Average Revenue | Fitted | Abs Deviation |
|---|---|---|---|
| 40 | 5958 | 6463.333333 | 505.3333333 |
| 44 | 6662 | 6620.666667 | 41.33333333 |
| 48 | 6004 | 6778 | 774 |
| 48 | 6011 | 6778 | 767 |
| 60 | 7250 | 7250 | 0 |
| 70 | 8632 | 7643.333333 | 988.6666667 |
| 72 | 6964 | 7722 | 758 |
| 90 | 11097 | 8430 | 2667 |
| 100 | 9107 | 8823.333333 | 283.6666667 |
| 168 | 11498 | 11498 | 0 |
| | | Mean Abs Deviation | 678.5 |

| | |
|---|---|
| Intercept | 4890.00 |
| Slope | 39.33 |

**Decision Variables:** Intercept and Slope

**Objective Funtion :** Minimize Mean absolute deviation

| | |
|---|---|
| Revenue prediction for 120 hrs of operation | 9610 |

Figure 1: Linear Model

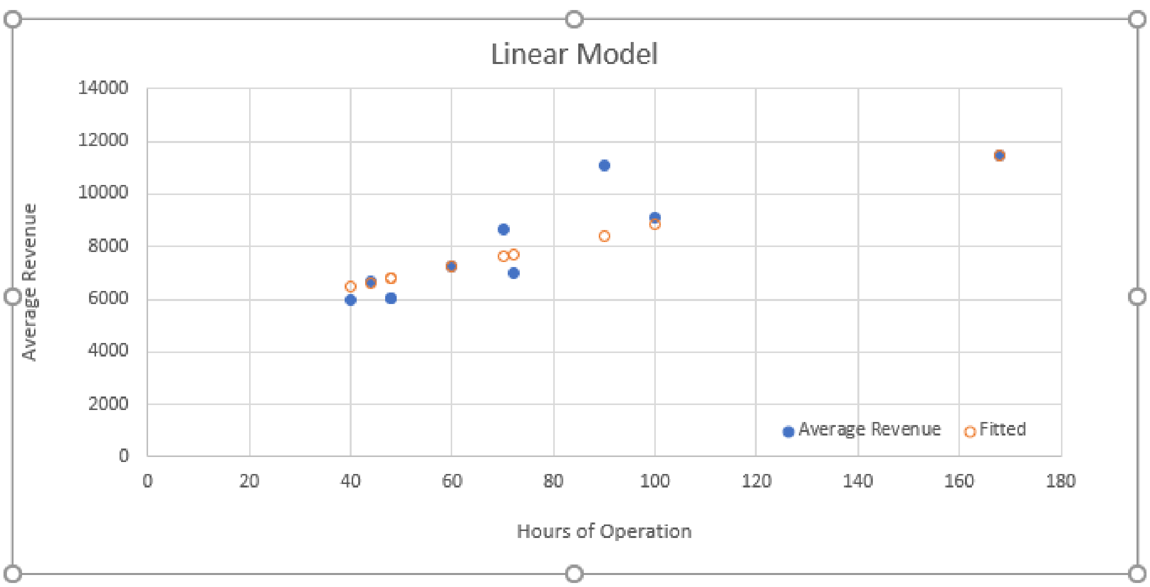



Figure 2: Linear Model plot

| Jours of Operation | Average Revenue | Fitted | Abs Deviation |
|---|---|---|---|
| 40 | 5958 | 5994.34 | 36.339 |
| 44 | 6662 | 6260.87 | 401.135 |
| 48 | 6004 | 6514.52 | 510.519 |
| 48 | 6011 | 6514.52 | 503.519 |
| 60 | 7250 | 7212.99 | 37.007 |
| 70 | 8632 | 7738.78 | 893.222 |
| 72 | 6964 | 7838.93 | 874.927 |
| 90 | 11097 | 8679.40 | 2417.600 |
| 100 | 9107 | 9106.99 | 0.006 |
| 168 | 11498 | 11540.22 | 42.220 |
| | Minimize MAE | | 571.65 |

| | |
|---|---|
| a | 1113.03 |
| b | 0.46 |

Decision variables for y = aX^b

Objective function - Minimize mean absolute deviation

| | |
|---|---|
| Revenue prediction for 120 hrs of operation | 9897.29 |

Figure 3: Non Linear Model

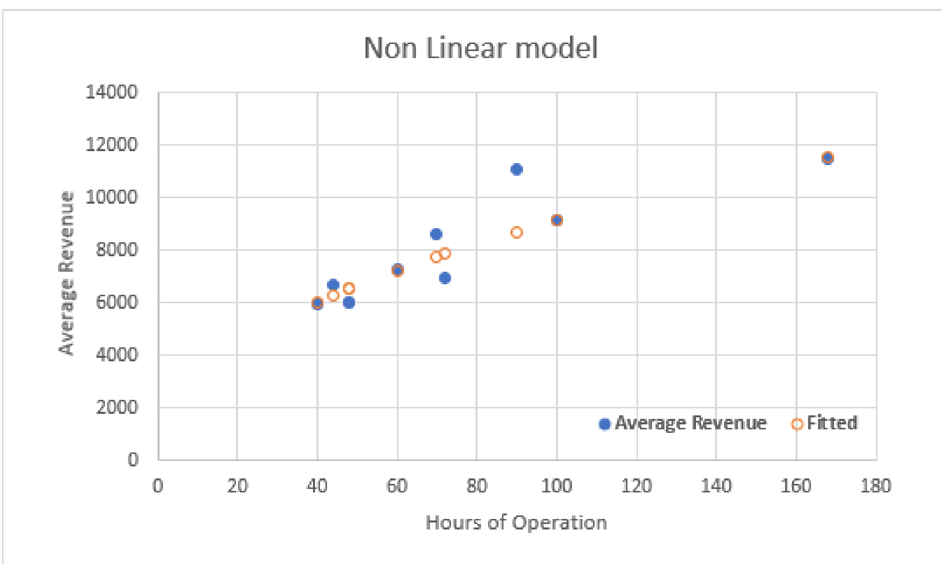
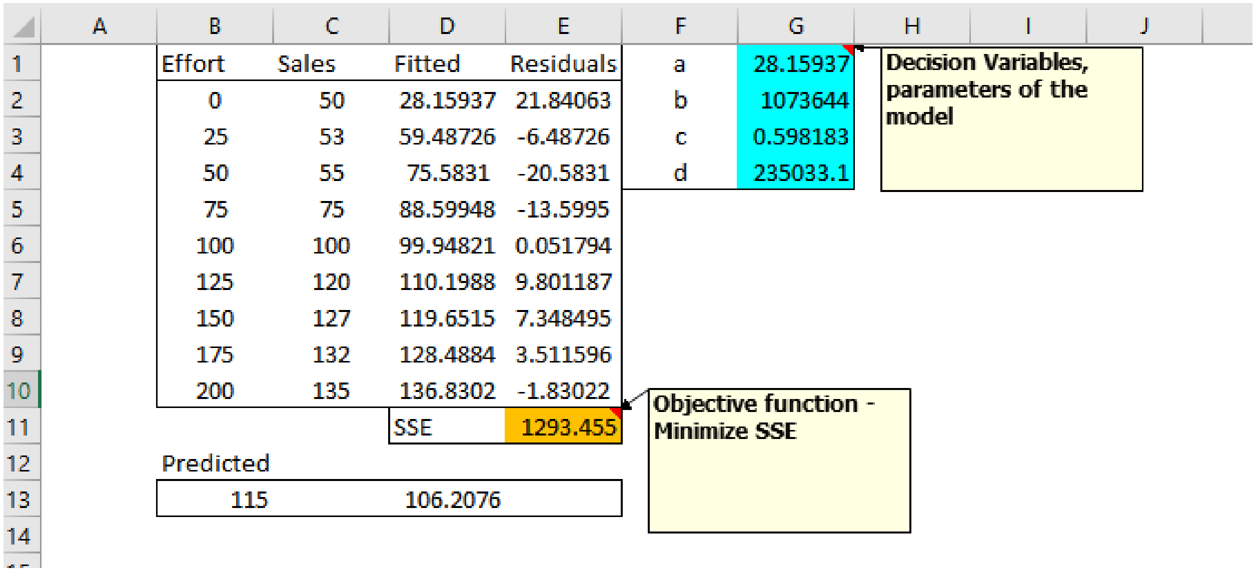



Figure 4: Non-Linear Model plot



| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | | Effort | Sales | Fitted | Residuals | a | 28.15937 |
| 2 | | 0 | 50 | 28.15937 | 21.84063 | b | 1073644 |
| 3 | | 25 | 53 | 59.48726 | -6.48726 | c | 0.598183 |
| 4 | | 50 | 55 | 75.5831 | -20.5831 | d | 235033.1 |
| 5 | | 75 | 75 | 88.59948 | -13.5995 | | |
| 6 | | 100 | 100 | 99.94821 | 0.051794 | | |
| 7 | | 125 | 120 | 110.1988 | 9.801187 | | |
| 8 | | 150 | 127 | 119.6515 | 7.348495 | | |
| 9 | | 175 | 132 | 128.4884 | 3.511596 | | |
| 10 | | 200 | 135 | 136.8302 | -1.83022 | | |
| 11 | | | | SSE | 1293.455 | | |
| 12 | | Predicted | | | | | |
| 13 | | 115 | | 106.2076 | | | |
| 14 | | | | | | | |



Figure 5: S curve model

## Problem 2

In this problem, an S curve model is fit for $S = a + \frac{(b-a)E^c}{(d+E^c)}$; where $(a, b, c, d)$ are constants; $E$ is effort in % and $S$ being the Actual Sales (% of Current)

The optimization model is set such that $a, b, c$ and $d$ are *decision variables* and the objective function is sum of squared errors between the model fit and the actuals (residuals). The objective function is *minimized*.

*a) Hence the model is :*

$S = 28.16 + \frac{(1073644-28.16)E^{0.598}}{(235033.1+E^{0.598})}$

*b) The predicted value for 115% Effort based on this model is 106.2*

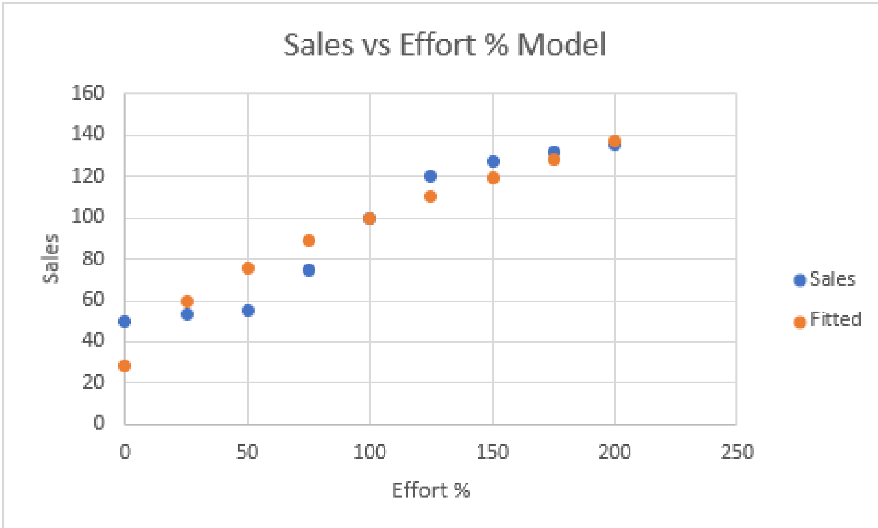



Figure 6: S curve model plot

## Problem 3

This is problem is a modification of the example problem in Chapter 12. The *decision variables* are the Reorder point and Order Quantity. The goal is to find the optimal quantities for the decision variables that maximizes the average monthly profit.

### Assumption made:

The Holding cost and ordering cost are NOT sunk in the profit margin ($45 per unit)

### Limitation of Education version of ASPE

Due to limitation of educational license of ASPE, the holding cost, ordering cost and opportunity cost are not shown in separate columns. They are collapsed into a single column where

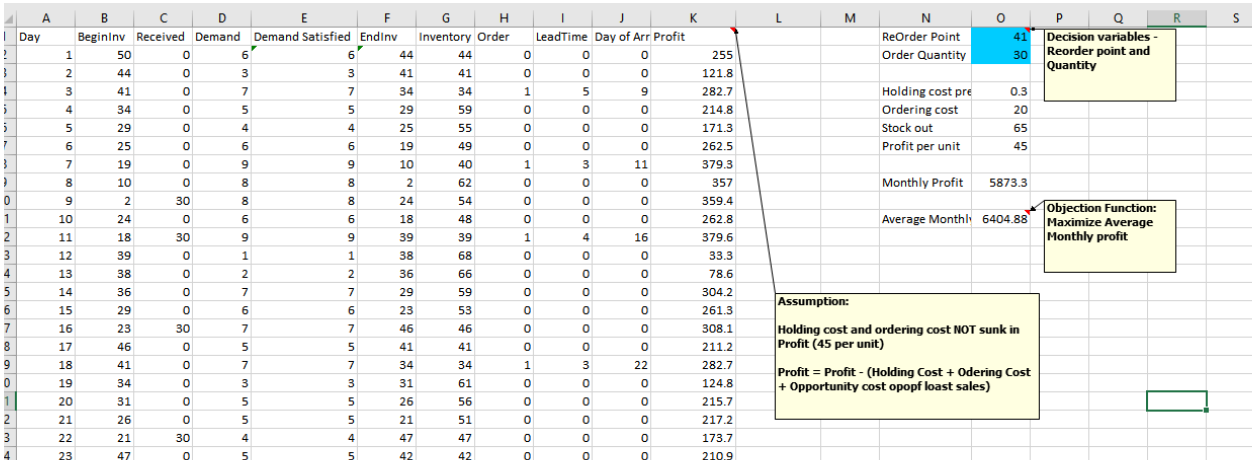
The formula in K2 is `=E2*$O$7-(B2*$O$4+H2*$O$5+(MAX(D2-E2,0)*$O$6))`, which copied all the way down for column K. As shown in Figure 7.

### Constraint

The Re-order point and Order quanity were constrainted to take values >= 1 and constrained to be integers.

### Uncertain inputs

The Demand and Lead time for delivery for orders were stocastic and is shown in Figures 8 and 9.



| Day | BeginInv | Received | Demand | Demand Satisfied | EndInv | Inventory | Order | LeadTime | Day of Arr | Profit |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 50 | 0 | 6 | 6 | 44 | 44 | 0 | 0 | 0 | 255 |
| 2 | 44 | 0 | 3 | 3 | 41 | 41 | 0 | 0 | 0 | 121.8 |
| 3 | 41 | 0 | 7 | 7 | 34 | 34 | 1 | 5 | 9 | 282.7 |
| 4 | 34 | 0 | 5 | 5 | 29 | 59 | 0 | 0 | 0 | 214.8 |
| 5 | 29 | 0 | 4 | 4 | 25 | 55 | 0 | 0 | 0 | 171.3 |
| 6 | 25 | 0 | 6 | 6 | 19 | 49 | 0 | 0 | 0 | 262.5 |
| 7 | 19 | 0 | 9 | 9 | 10 | 40 | 1 | 3 | 11 | 379.3 |
| 8 | 10 | 0 | 8 | 8 | 2 | 62 | 0 | 0 | 0 | 357 |
| 9 | 2 | 30 | 8 | 8 | 24 | 54 | 0 | 0 | 0 | 359.4 |
| 10 | 24 | 0 | 6 | 6 | 18 | 48 | 0 | 0 | 0 | 262.8 |
| 11 | 18 | 30 | 9 | 9 | 39 | 39 | 1 | 4 | 16 | 379.6 |
| 12 | 39 | 0 | 1 | 1 | 38 | 68 | 0 | 0 | 0 | 33.3 |
| 13 | 38 | 0 | 2 | 2 | 36 | 66 | 0 | 0 | 0 | 78.6 |
| 14 | 36 | 0 | 7 | 7 | 29 | 59 | 0 | 0 | 0 | 304.2 |
| 15 | 29 | 0 | 6 | 6 | 23 | 53 | 0 | 0 | 0 | 261.3 |
| 16 | 23 | 30 | 7 | 7 | 46 | 46 | 0 | 0 | 0 | 308.1 |
| 17 | 46 | 0 | 5 | 5 | 41 | 41 | 0 | 0 | 0 | 211.2 |
| 18 | 41 | 0 | 7 | 7 | 34 | 34 | 1 | 3 | 22 | 282.7 |
| 19 | 34 | 0 | 3 | 3 | 31 | 61 | 0 | 0 | 0 | 124.8 |
| 20 | 31 | 0 | 5 | 5 | 26 | 56 | 0 | 0 | 0 | 215.7 |
| 21 | 26 | 0 | 5 | 5 | 21 | 51 | 0 | 0 | 0 | 217.2 |
| 22 | 21 | 30 | 4 | 4 | 47 | 47 | 0 | 0 | 0 | 173.7 |
| 23 | 47 | 0 | 5 | 5 | 42 | 42 | 0 | 0 | 0 | 210.9 |

| | |
|---|---|
| ReOrder Point | 41 |
| Order Quantity | 30 |
| Holding cost pre | 0.3 |
| Ordering cost | 20 |
| Stock out | 65 |
| Profit per unit | 45 |
| Monthly Profit | 5873.3 |
| Average Monthl | 6404.88 |

**Decision variables - Reorder point and Quantity**

**Objection Function: Maximize Average Monthly profit**

**Assumption:**

**Holding cost and ordering cost NOT sunk in Profit (45 per unit)**

**Profit = Profit - (Holding Cost + Odering Cost + Opportunity cost opopf loast sales)**

Figure 7: Inventory model

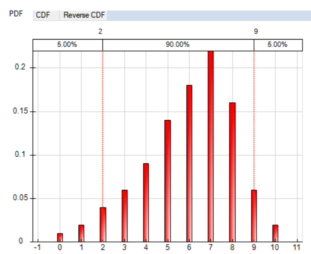



Figure 8: Inventory model - Demand

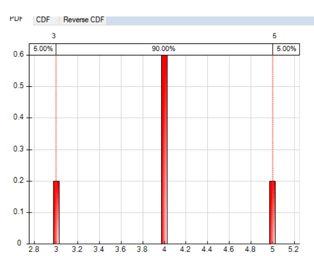



Figure 9: Inventory model - Lead Time

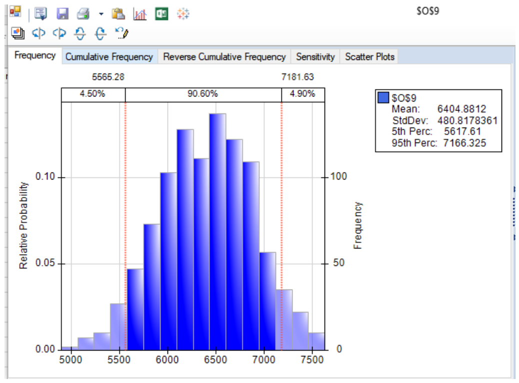



Figure 10: Inventory model - Lead Time

### Objective function variability

The objective function variability is shown in Figure 10.

### Optimal Values

**The Reorder point of 41 and Order Quantity of 30 proved to optimal**. (The values vary subtly based on seed of the random number generator, the seed used here is 123)

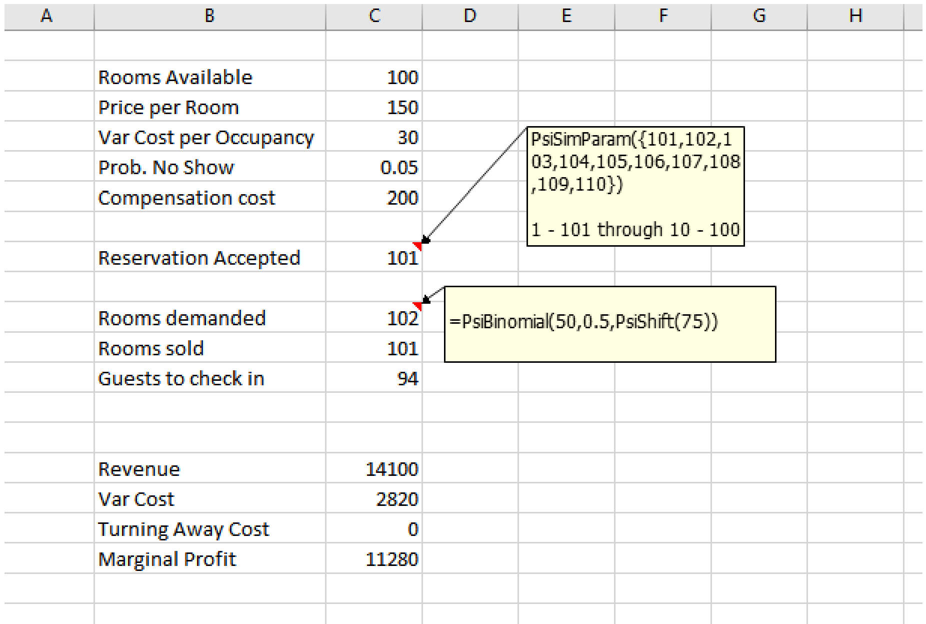



Figure 11: Hotel reservation model

## Problem 4

In this problem, the goal is to find maximum number of reservations that can be accepted for hotel rooms. The hotel has a 100 rooms; there is a 5% chance of no show. When more guests show up than available rooms, there is a compensation of 200 dollars paid to the customer. It is assumed that the guest does not pay for the room that they are checked into. All guests pay ($150) at check in if they have a room. There is variable cost of $30 per room occupied.

### Simulation set up

The simulation set up is shown in figure 11. There are 10 simulations run for different scenarios of accepted reservations ranging from 101 through 110. The Marginal profit (Revenue - variable cost - cost of turning away customers) is monitored for each of the simulation runs.

### Assumptions

*The demand for rooms is assumed to have a binomial distribution with a shift of 75.* See figure 12.

### Simulation out put

**The results of the simulation is shown in figure 13. It is seen that the 5th simulation where maximum reservation accepted is 105 gives us the maximum marginal profit of $11351.20.**

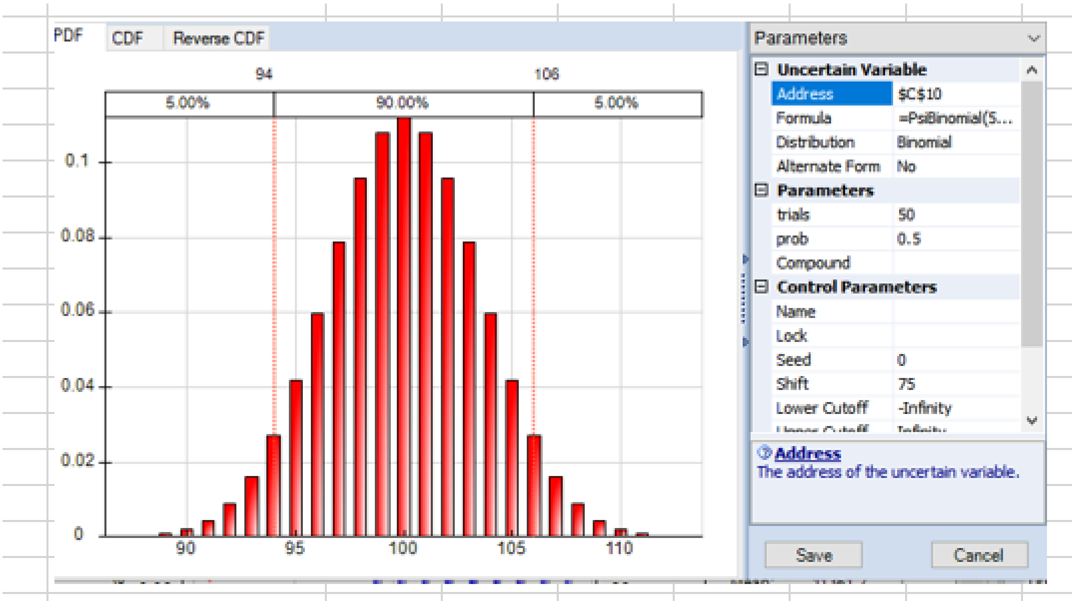



Figure 12: Hotel demand model

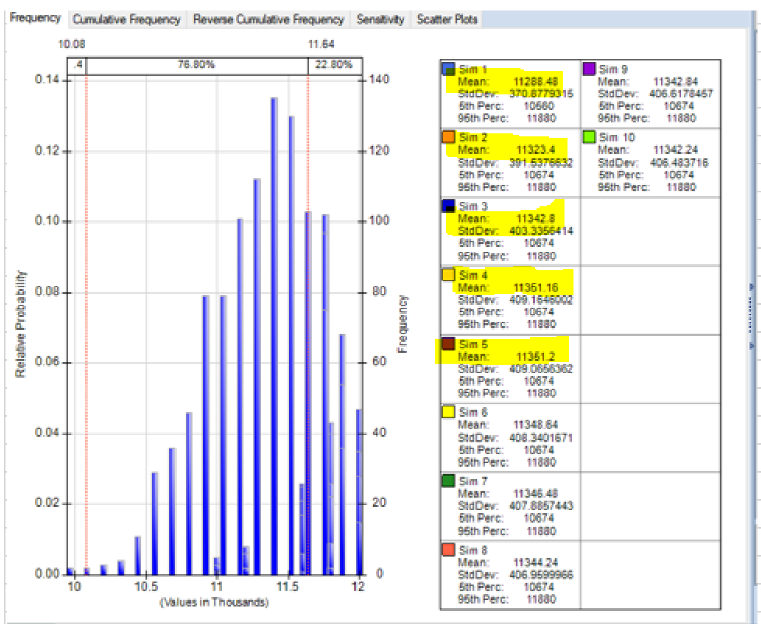



Figure 13: Simulation results

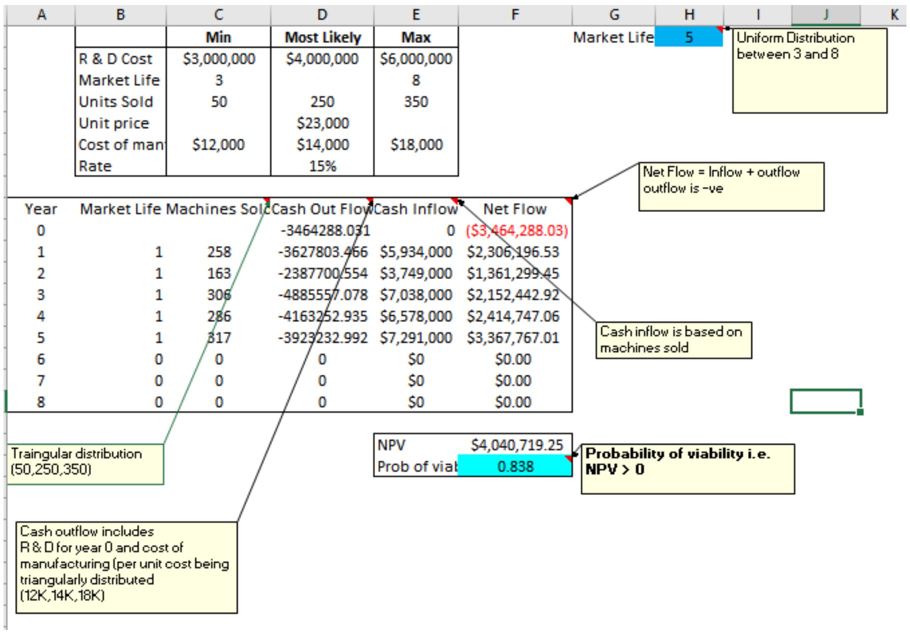



Figure 14: Product viability simulation model

### Problem 5

In this problem the goal is see what is the probability of viability of introducing a prodcut to the market that has uncertain market life, manufacturing cost,unit to be sold per year. However the R & D investment, sale price and cost of capital are assumed to be certain. The net present value of the invenstment over the span of market life is simulated.

### Assumption made.

The problem does not state clearly if the cost of manufacturing per machine randomly varies for each year of market life. The solution to this problem assumes it is so. Each year the manufacturing cost is assumed to have a triangular distribution of min = 12K, most likely = 14K and max = 18K.

The model set is shown in figure 14.

### Simulation output

The Net Present Value (NPV) of the 1000 iterations of the simulation models is studied and the probability of NPV > 0 is evaluated.

**It is seen that the probability of a viable product is 83.3%.**

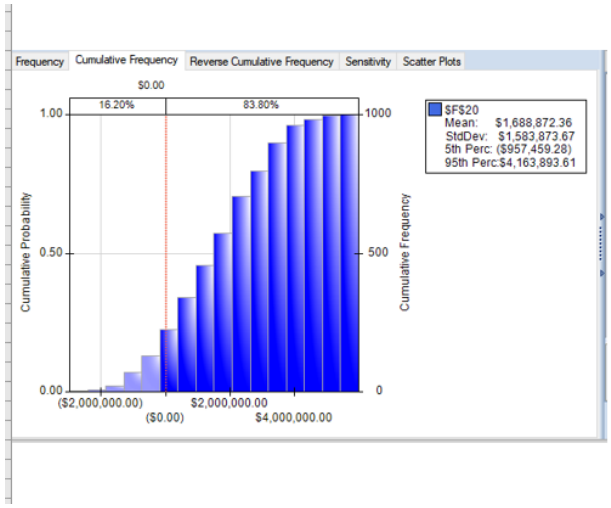



Figure 15: Product viability simulation result

**Extra Credit**

The below function **estimatepi**, takes in an argument for N and outputs a data frame that contain the estimate of pi, standard error of the estimate and the 95% confidence interval around the estimate. The function generates N pairs of uniformly distributed random numbers between 0 and 1 for $x$ and $y$ coordinates. A subfunction **insidecircle** checks if the pair of points lie within a unit circle; by checking if $x^2 + y^2 <= 1$. insidecircle returns a 1 if the pair of point is within the radius of the unit circle, else returns a 0.

Details of the code can be seen in the code chunk below.

```r
#' This function estimates the value of pi using a unit circle.
#'
#' @param N, integer value denoting the number of random points to generate
#'
#' @return data frame with estimate of pi, standard error of estimate and the 95% confidence interval
#' @export
#'
#' @examples estimatepi(1000)
estimatepi <- function(N) {
    library(magrittr)
    library(dplyr)

    # Function to flag points that are inside unit circle.
    insidecircle <- function(x, y) {
        ifelse(x^2 + y^2 <= 1, 1, 0)
    }

    df <- data.frame(x = abs(runif(N)), y = abs(runif(N)))
    df %<>% dplyr::mutate(selected = insidecircle(x, y))

    # Get estimate of points that fall with a unit quarter-circle
    pi_estimate <- df %>% summarise(estimate = sum(selected)/n() *
        4) %>% as.numeric(.)
    # Store the proporation for ease of computation below
    prop <- sum(df$selected)/nrow(df)
    # CI <- prop.test(sum(df$selected),nrow(df))$conf.int

    StdErrorProp <- sqrt(prop * (1 - prop)/nrow(df))
    E <- qnorm(0.975) * StdErrorProp

    CI <- c(prop - E, prop + E)

    pi_se <- StdErrorProp * 4
    pi_CI <- CI * 4

    # Return estimate of pi, the standard error and the 95%
    # confidence interval

    data.frame(pi_estimate = pi_estimate, pi_StdError = pi_se,
        LowerConfLimit = pi_CI[1], UpperConfLimit = pi_CI[2])

}
```

**The function estimatepi is run to estimate the value of pi for N varying from 1000 to 10000 in steps of 500. The code chunk below shows the execution.**

```r
library(purrr)
library(ggplot2)
N <- seq(1000, 10000, 500)
set.seed(111)
results <- map_df(.x = N, .f = estimatepi) %>% mutate(N = N)
```

**Figure 16 shows the confidence interval around the estimate of pi against each value of $N$. The solid red line shows the true value of pi and the dashed red lines show $\pm 0.05$ (total band width of 0.1) from pi.**

**It can be seen that from N >= 9000, the estimate safely falls within $\pm 0.05$ of true value of pi.**

```
# Plot results

results %>% ggplot(mapping = aes(x = N, y = pi_estimate, ymin = LowerConfLimit,
    ymax = UpperConfLimit)) + geom_pointrange(alpha = 0.5) +
    geom_hline(yintercept = pi, col = "red") + geom_hline(yintercept = pi -
    0.05, lty = 2, col = "red") + geom_hline(yintercept = pi +
    0.05, lty = 2, col = "red") + geom_vline(xintercept = 8750,
    col = "blue") + theme_bw()
```

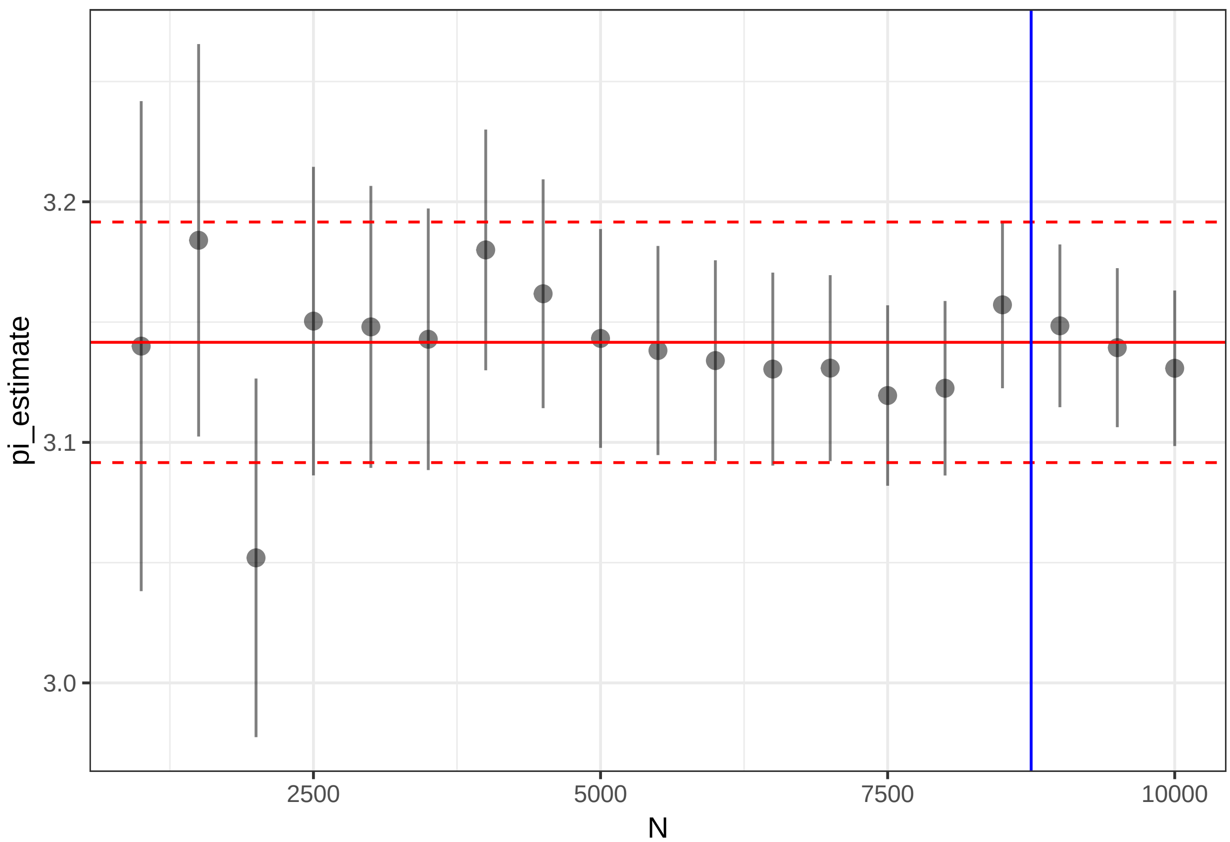


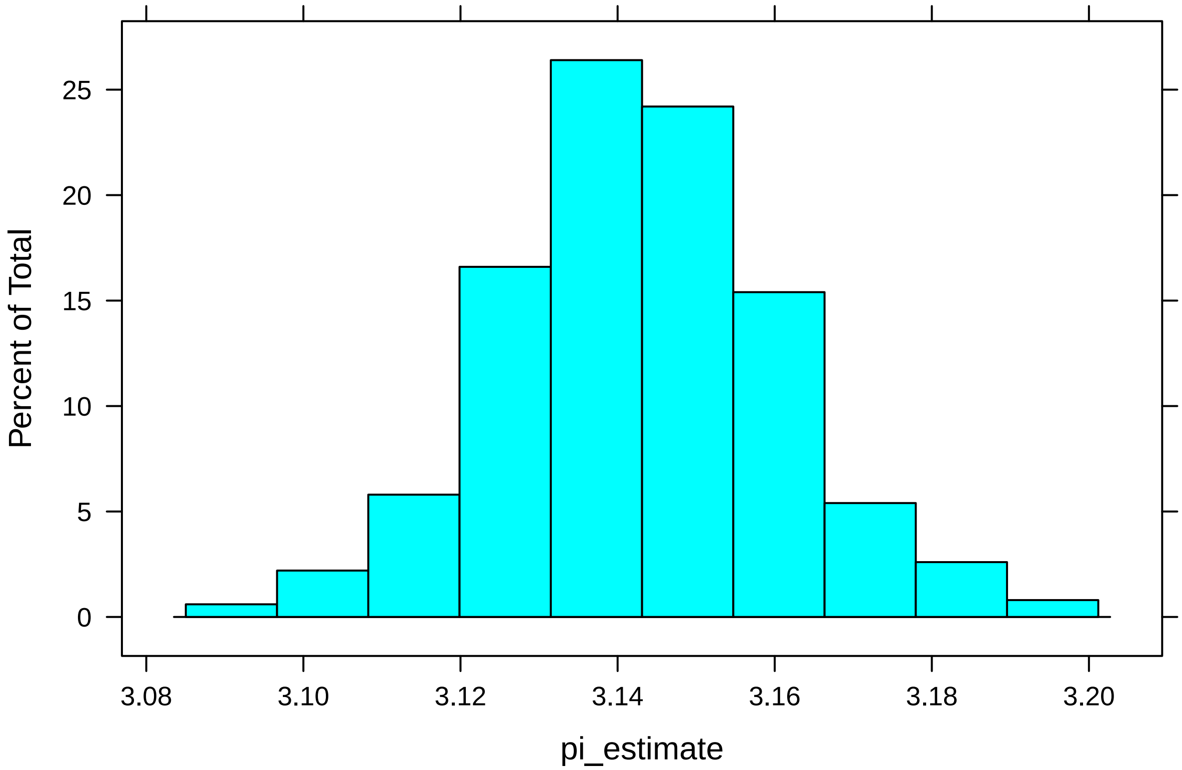

Figure 16: Estimate of pi by N

### Repeated simulation for N = 9000

The estimate of pi is simulated 500 times using a value of N = 9000. The histogram of the extimate is shown in figure 17. The distribution is normal looking as expected - due to central limit theorm.

```
N_optimal = rep(9000, 500)

# 500 simulation iterations
results_rep <- map_df(.x = N_optimal, .f = estimatepi)
lattice::histogram(x = ~pi_estimate, data = results_rep)
```




### Validation

Table 1 shows the Average estimate of pi and the standard deviation of the estimate of pi. The standard error computed with N = 9000 previously is very close to that of the standard deviation of the estimate for the 500 simulations. 100% of the estimate falls within the confidence interval estimated previously.

```
knitr::kable(data.frame(Average_Pi_estimate = mean(results_rep$pi_estimate),
    Pi_std.Dev = sd(results_rep$pi_estimate), pi_stdError = results$pi_StdError[results$N ==
        9000], PercentWithinCI = length(between(results_rep$pi_estimate,
        results$LowerConfLimit[results$N == 9000], results$UpperConfLimit[results$N ==
            9000]))/500 * 100))
```

| Average_Pi_estimate | Pi_std.Dev | pi_stdError | PercentWithinCI |
|---:|---:|---:|---:|
| 3.14287 | 0.0177704 | 0.0172597 | 100 |