Semicon Simulation: Analysis Portion

Group 15

2023-04-22

PRELIMINARY DATA WRANGLING IN MICROSOFT EXCEL

Step 1: The output data from our JaamSim Semicon Simulation containing 1.4k observations was converted from .txt file format to .xlsx file format using Microsoft Excel.

Step 2: The following Excel formulas were applied to all three Excel files containing data of three variations of the simulation.

- ***Case 1:*** All 5 regions are involved in the system process.
- ***Case 2:*** China is cut off from the system process.
- ***Case 3:*** USA is cut off from the system process.

Our DES was designed to take the ratio of time with respect to 1 hour. In order to get the monthly profits, we divide the values in Column 1 by (30/3600) or 0.0083333333333333.

Only multiples of 0.0083333333333333 are perfectly divisible by the number. Hence, we filter out integers to extract monthly profits for all five major regions involved in the Global Semiconductor Industry.

=[@Months]/A8

=INT([@[Filter Condition '#1]])=[@[Filter Condition '#1]]

Months	US	EU	CN	EA	SEA	Filter Condition #1	Filter Condition #2
8.61E-04	-2000	0	-1700	-2000	300	0.103333333	FALSE
0.001972222	300	-2300	-1700	-2000	300	0.236666667	FALSE
0.002964942	-1700	-2300	-1700	-2000	300	0.355793	FALSE
0.008333333	-1700	-2300	-1700	-2000	300	1	TRUE
0.010883441	-1700	-2300	-3500	-2000	600	1.306012867	FALSE
0.012972281	-3700	-2300	-3500	-2000	600	1.556673667	FALSE
0.014059077	-3700	-2300	-5300	-2000	900	1.6870892	FALSE
0.016066653	-3700	-2300	-7200	-2000	900	1.927998333	FALSE
0.016666667	-3700	-2300	-7200	-2000	900	2	TRUE
0.017906051	-3700	-2300	-9100	-2000	900	2.148726133	FALSE
0.018038875	-3700	-4300	-7100	-2000	900	2.164665	FALSE
0.018767162	-3700	-4300	-5100	-4000	900	2.252059467	FALSE
0.02140442	-3700	-4300	-5100	-4000	-600	2.5685304	FALSE
0.021418309	-3700	-4300	-5100	-5800	1200	2.570197067	FALSE
0.0226906	-3700	-4300	-7000	-5800	1200	2.722872033	FALSE
0.023849378	-3700	-4300	-8800	-5800	1500	2.861925333	FALSE
0.023939223	-3700	-4300	-8800	-5800	0	2.872706733	FALSE
0.023953112	-3700	-4300	-8800	-7600	1800	2.8743734	FALSE
0.024662822	-3700	-6300	-6800	-7600	1800	2.9595387	FALSE

Step 3: All three Excel files are exported as .csv files for further data manipulation and statistical analysis in R.

Loading the data sets:

```
original <- read.csv('original.csv')
cut_china <- read.csv('cut_china.csv')
cut_us <- read.csv('cut_us.csv')
```

DATA CLEANING & MANIPULATION

Filtering data frames by “TRUE” condition:

```
#install.packages("dplyr") -> for data manipulation
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
original <- filter(original, Filter.Condition..2 == 'TRUE')
cut_china <- filter(cut_china, Filter.Condition..2 == 'TRUE')
cut_us <- filter(cut_us, Filter.Condition..2 == 'TRUE')
```

Deleting the “Months” & “Filter.Condition..2” columns:

```
#install.packages("magrittr") -> to use the pipe operator (%>%)
library(magrittr)

original <- original %>% dplyr::select(-c(Months, Filter.Condition..2))
cut_china <- cut_china %>% dplyr::select(-c(Months, Filter.Condition..2))
cut_us <- cut_us %>% dplyr::select(-c(Months, Filter.Condition..2))
```

Renaming “Filter.Condition..2” column name to “Months”:

```
names(original)[6] = "Months"
names(cut_china)[6] = "Months"
names(cut_us)[6] = "Months"
```

Finding the average profit of the overall system for 3 unique cases:

```
original$AVG_Original <- rowMeans(original[1:5], na.rm=TRUE)
cut_china$AVG_Cut_China <- rowMeans(cut_china[1:5], na.rm=TRUE)
cut_us$AVG_Cut_USA <- rowMeans(cut_us[1:5], na.rm=TRUE)
```

Creating a new data frame with the essential data:

```
df <- data.frame(
  Months = c(1:121),
  original$AVG_Original,
  cut_china$AVG_Cut_China,
  cut_us$AVG_Cut_USA)
```

Renaming the columns:

```
names(df)[2] = "AVG_Original"
names(df)[3] = "AVG_Cut_China"
names(df)[4] = "AVG_Cut_USA"
```

Preserving top 120 data points:

We will perform the paired t-test on a data sample consisting of the first 120 months (10 years).

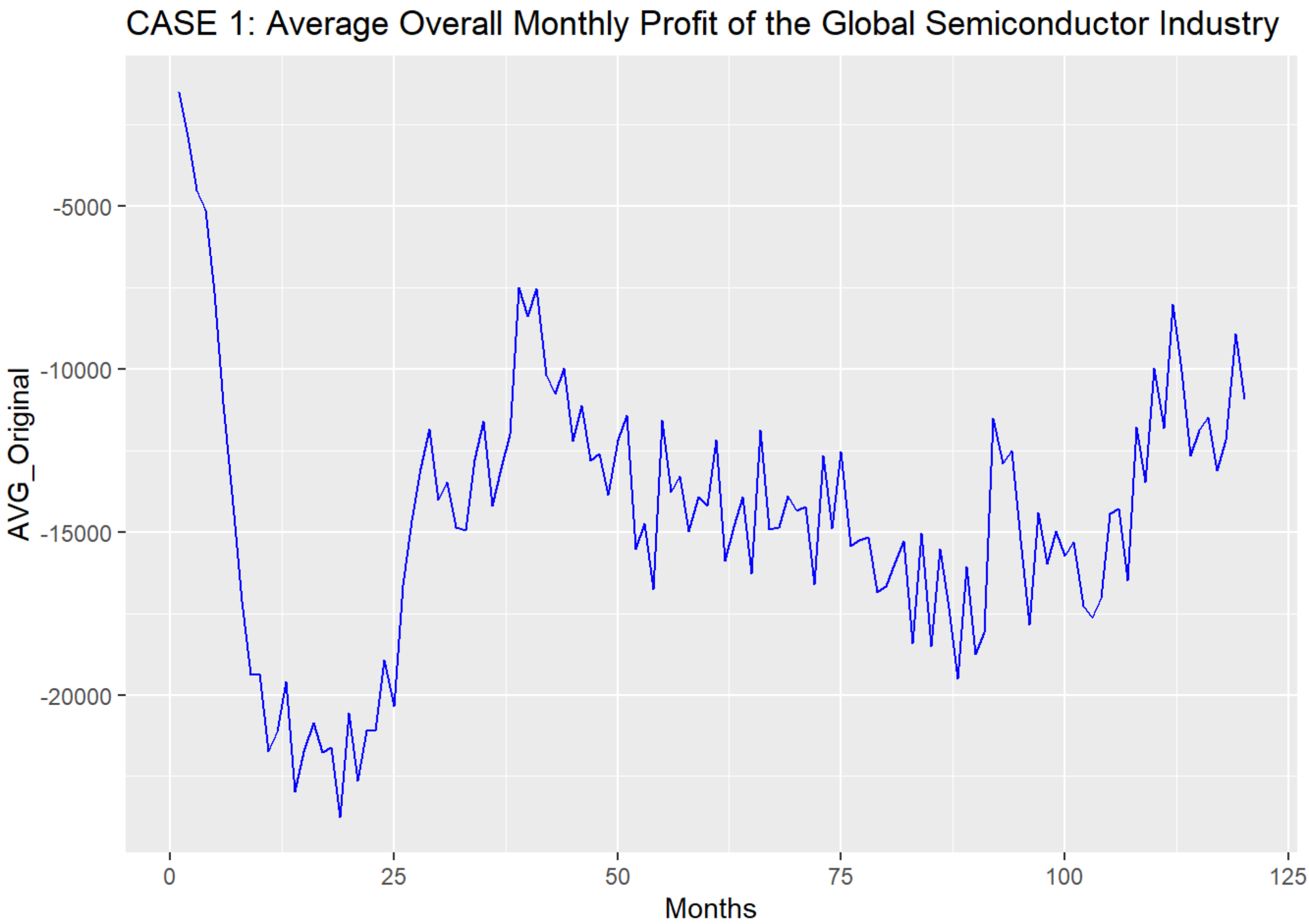
```
df <- head(df, -1)
dim(df)
```

```
## [1] 120   4
```

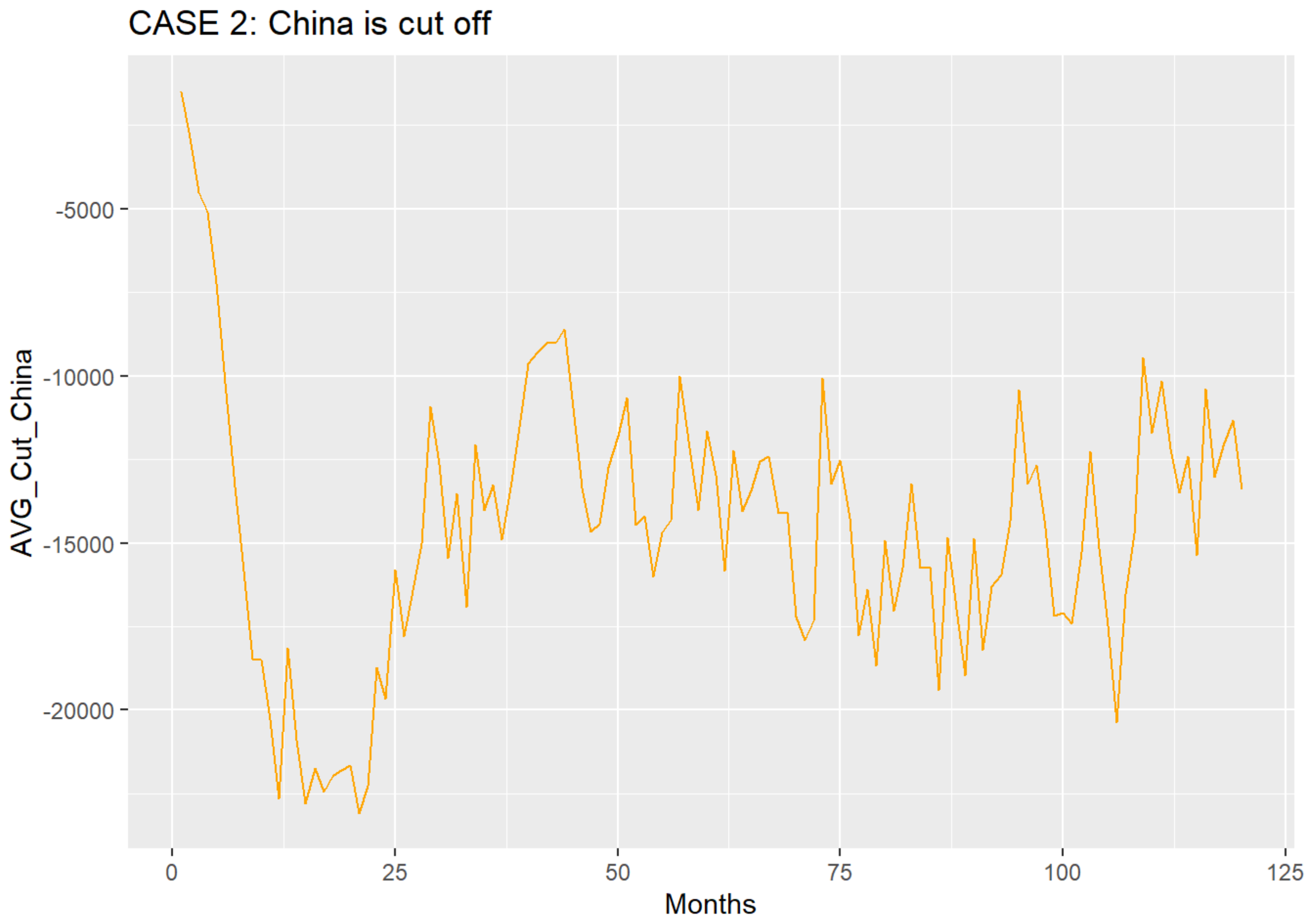
Since the data used for analysis is generated from a terminating simulation with **“empty-and-idle”** initial conditions, it is important to ensure that the simulation is properly initialized to avoid making any incorrect statistical inferences.

The following line graphs perfectly illustrate the initial instabilities in the system processes:

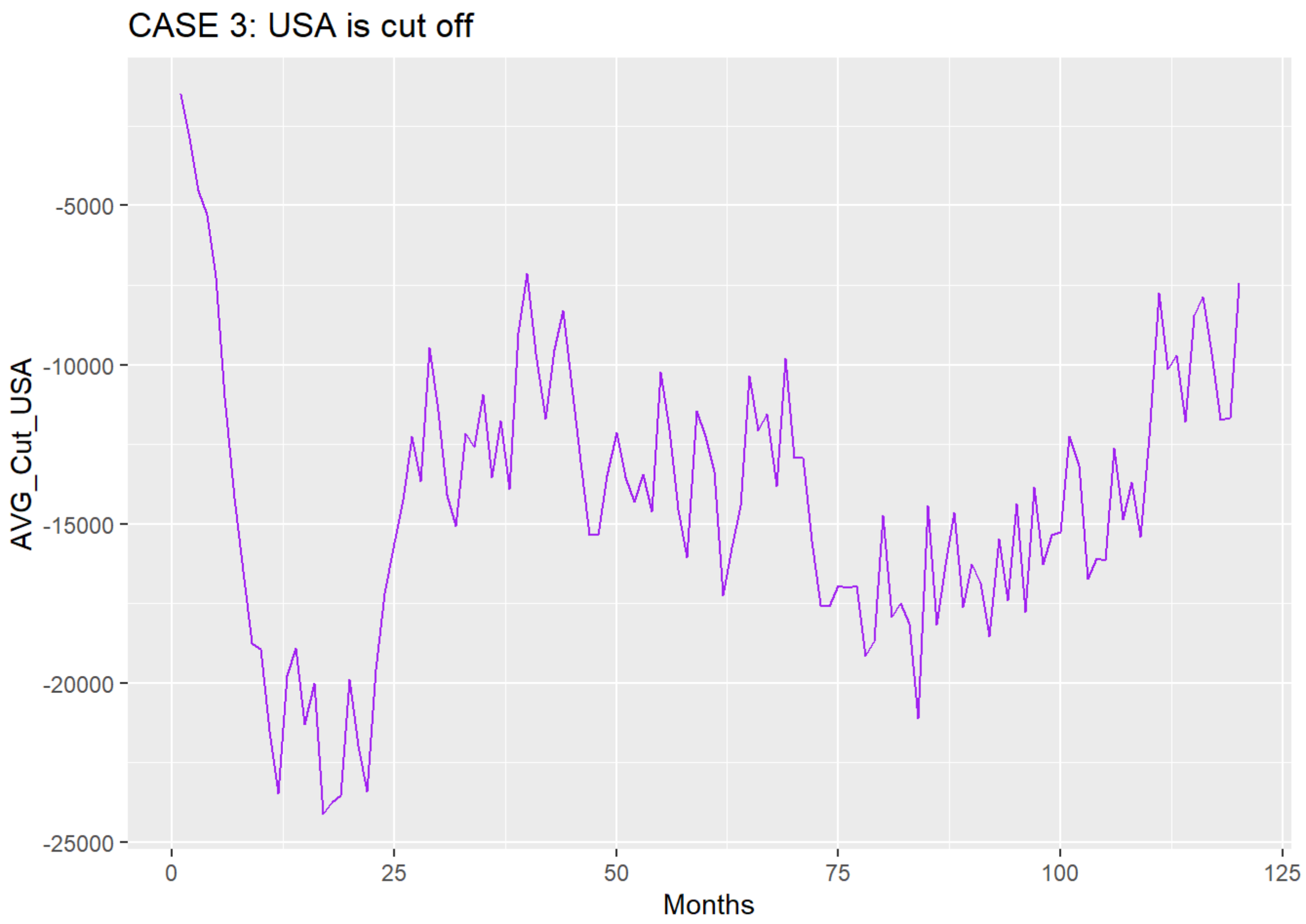
```
#install.packages("ggplot2") -> for data visualizations
library(ggplot2)

ggplot(df, aes(x = Months, y = AVG_Original)) + geom_line(color="blue") + ggtitle("CASE 1: Av
erage Overall Monthly Profit of the Global Semiconductor Industry")
```




```
ggplot(df, aes(x = Months, y = AVG_Cut_China)) + geom_line(color="orange") + ggtitle("CASE 2:
China is cut off")
```




```
ggplot(df, aes(x = Months, y = AVG_Cut_USA)) + geom_line(color="purple") + ggtitle("CASE 3: U
SA is cut off")
```



However, since the data generated is not from a steady-state simulation, we cannot truncate data points using the **Welch's Method**.

The first few observations containing NULL profit values were removed in Microsoft Excel as an attempt to solve biases related to the "empty-and-idle" initial data values.

```
this.SimTime/1[h] -[USA_E].NumberProcessed*[C_USA_E].Value  -[USA_P].NumberProcessed*[C_USA_P].Value  -[USA_M].NumberProcessed*[C_USA_M].Value  -
[USA_A].NumberProcessed*[C_USA_A].Value  +[USA_D].NumberProcessed*[C_USA_D].Value  +[USA_E_CHN].NumberProcessed*[EX_USA_E].Value    -1
*[EUR_P].NumberProcessed*[C_EUR_P].Value -1*[EUR_M].NumberProcessed*[C_EUR_M].Value +1*[EUR_D].NumberProcessed*[C_EUR_D].Value
+[EUR_M_CHN].NumberProcessed*[EX_EUR_M].Value +[EUR_M_USA].NumberProcessed*[EX_EUR_M].Value  -[USA_E_EUR].NumberProcessed*[EX -1
*[CHN_E].NumberProcessed*[C_CHN_E].Value -1*[CHN_P].NumberProcessed*[C_CHN_P].Value -1*[CHN_M].NumberProcessed*[C_CHN_M].Value -1
*[CHN_A].NumberProcessed*[C_CHN_A].Value +[CHN_D].NumberProcessed*[C_CHN_D].Value  +[CHN_A_EUR].NumberProcessed*[EX_CHN_A].V  -1
*[EAS_P].NumberProcessed*[C_EAS_P].Value -1*[EAS_M].NumberProcessed*[C_EAS_M].Value +[EAS_D].NumberProcessed*[C_EAS_D].Value
+[EAS_M_SEA].NumberProcessed*[EX_EAS_M].Value -[CHN_E_EAS].NumberProcessed*[EX_CHN_E].Value -[SEA_E_EAS].NumberProcessed*[EX_SE     -1
*[SEA_E].NumberProcessed*[C_SEA_E].Value -1*[SEA_A].NumberProcessed*[C_SEA_A].Value +[SEA_D].NumberProcessed*[C_SEA_D].Value
+[SEA_E_CHN].NumberProcessed*[EX_SEA_E].Value +[SEA_E_EAS].NumberProcessed*[EX_SEA_E].Value  +[SEA_A_CHN].NumberProcessed*[EX_S
0      0      0      0      0      0
0      0      0      0      0      0
0      -2000  0      -3700  0      300
8.61E-04     -2000 0      -1700 -2000 300
0.001972222 300    -2300 -1700 -2000 300
0.002964942 -1700 -2300 -1700 -2000 300
0.008333333 -1700 -2300 -1700 -2000 300
0.010883441 -1700 -2300 -3500 -2000 600
```

"empty-&-idle" data values
found in .txt file containing raw unstructured data
generated from JaamSim Semicon Simulation

Here is the final data frame!

```
head(df)
```

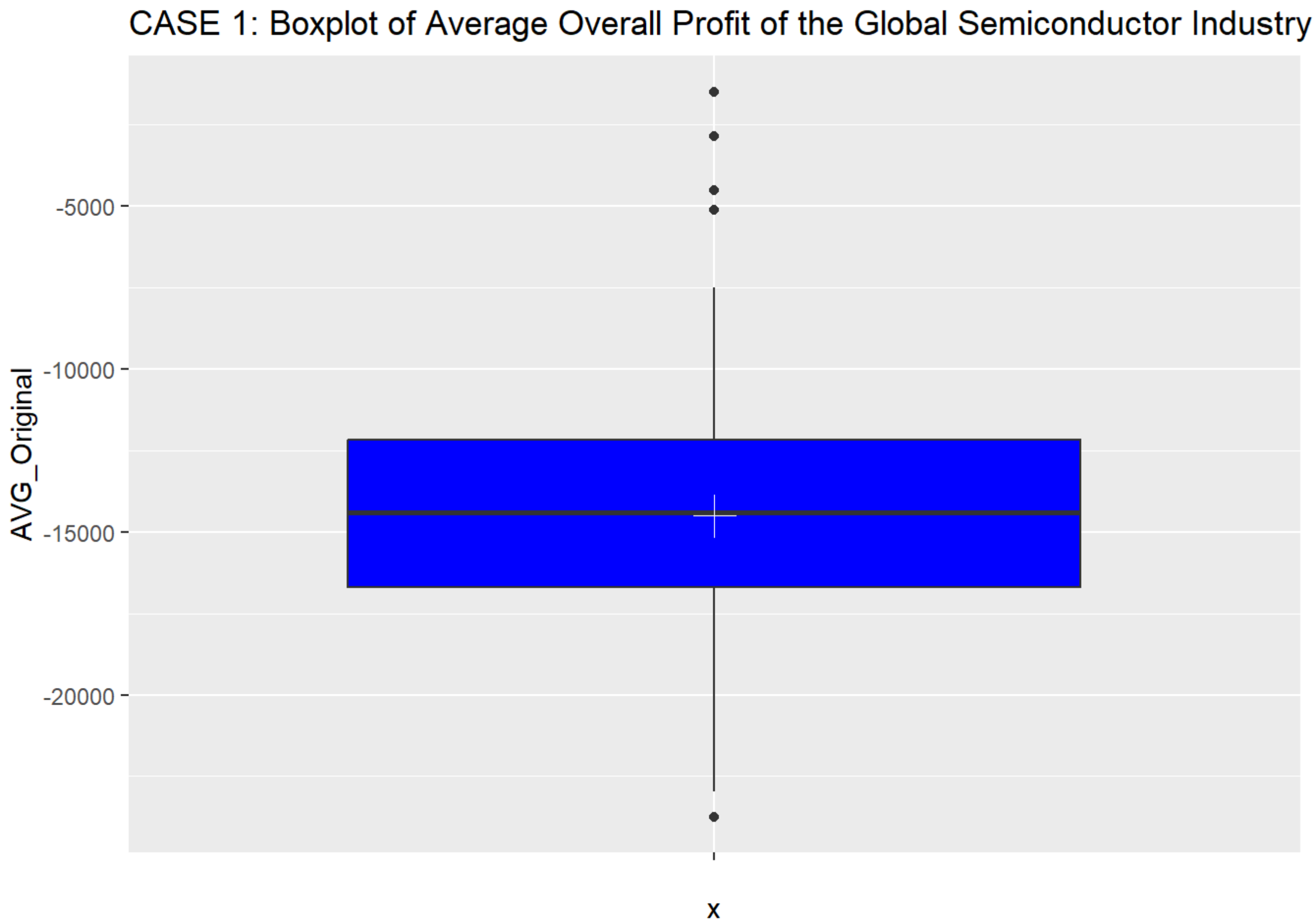
```
##   Months AVG_Original AVG_Cut_China AVG_Cut_USA
## 1      1        -1480         -1480       -1480
## 2      2        -2860         -2860       -2860
## 3      3        -4520         -4520       -4520
## 4      4        -5100         -5100       -5300
## 5      5        -7640         -7240       -7240
## 6      6       -11060        -10220      -11040
```

DATA EXPLORATION

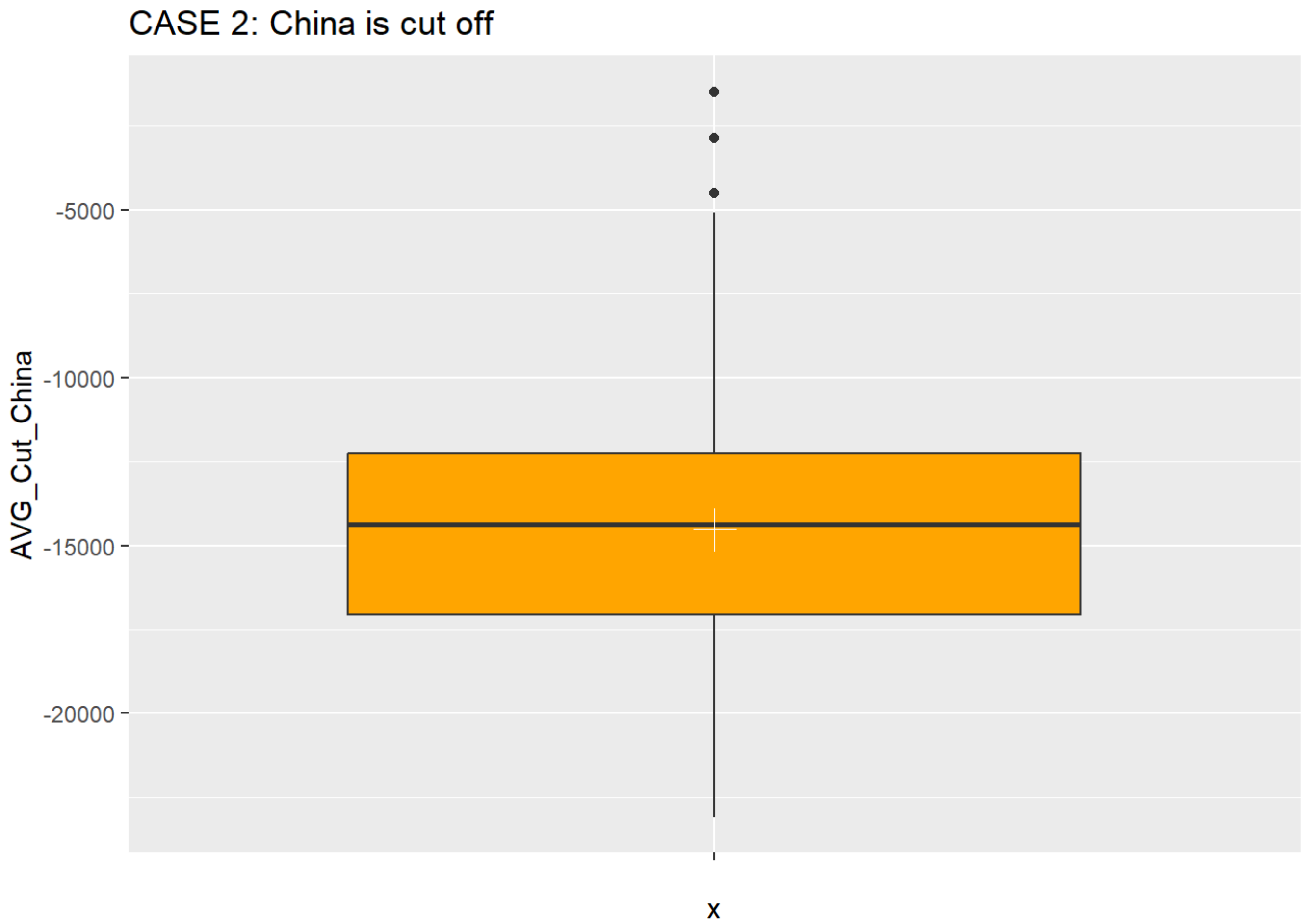
The following box plots are used to visualize the spread of the Average Overall Profits for the three cases over the course of 10 years:

```
ggplot(df) + aes(x = "", y = AVG_Original) + geom_boxplot(fill = "blue") + ggtitle("CASE 1: Boxplot of Average Overall Profit of the Global Semiconductor Industry") + stat_summary(fun.y = "mean", geom = "point", size = 5, color = "white", shape = 3)
```

```
## Warning: The `fun.y` argument of `stat_summary()` is deprecated as of ggplot2 3.3.0.
## ℹ Please use the `fun` argument instead.
```

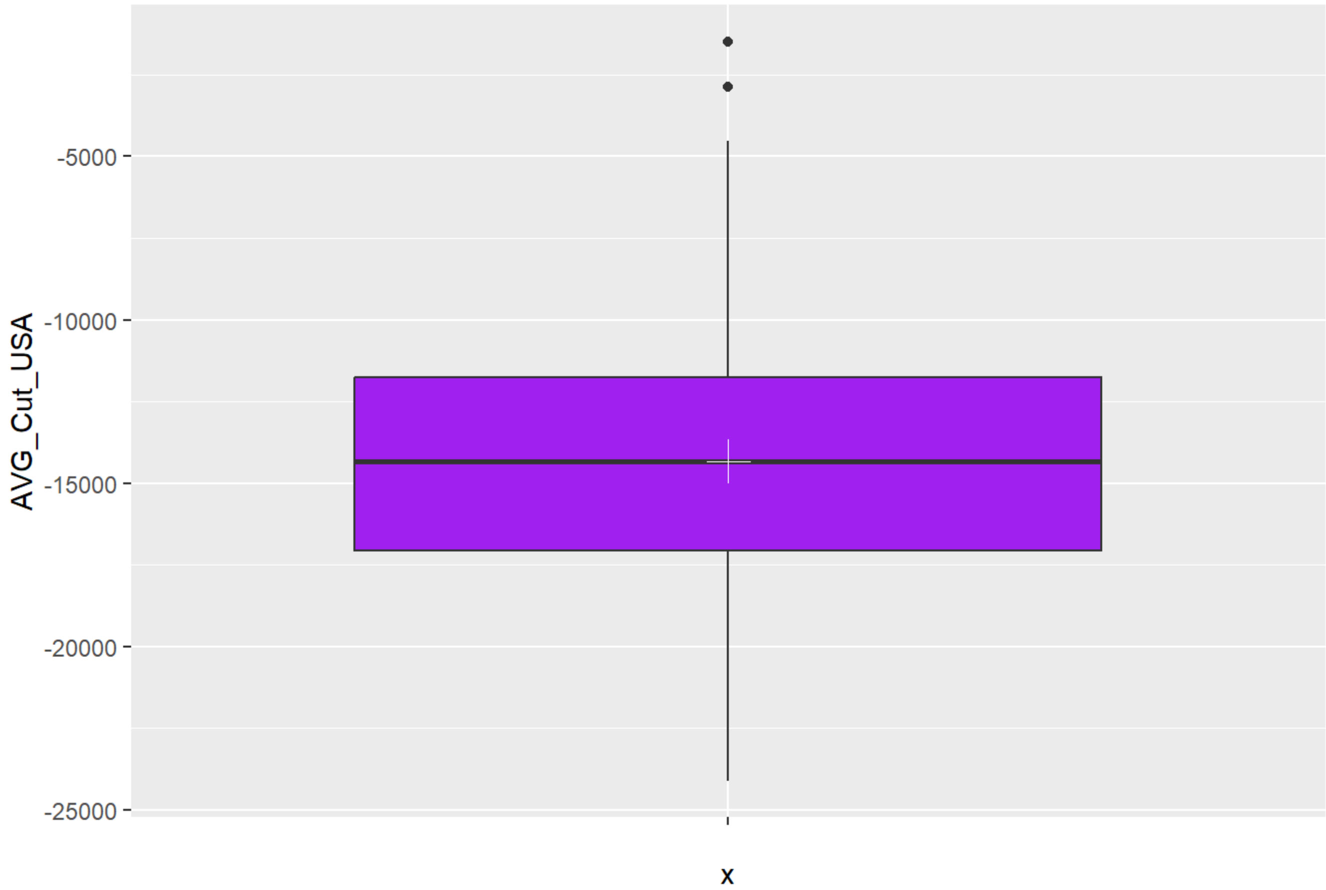



```
ggplot(df) + aes(x = "", y = AVG_Cut_China) + geom_boxplot(fill = "orange") + ggtitle("CASE
2: China is cut off") + stat_summary(fun.y = "mean", geom = "point", size = 5, color = "whit
e", shape = 3)
```




```
ggplot(df) + aes(x = "", y = AVG_Cut_USA) + geom_boxplot(fill = "purple") + ggtitle("CASE 3:
USA is cut off") + stat_summary(fun.y = "mean", geom = "point", size = 5, color = "white", sh
ape = 3)
```

CASE 3: USA is cut off



The box plots has revealed **10 outliers** which must be removed from the data set.

```
boxplot(df$AVG_Original, plot = FALSE)$out
```

```
## [1]  -1480  -2860  -4520  -5100 -23740
```

```
boxplot(df$AVG_Cut_China, plot = FALSE)$out
```

```
## [1] -1480 -2860 -4520
```

```
boxplot(df$AVG_Cut_USA, plot = FALSE)$out
```

```
## [1] -1480 -2860
```

Procedure for removing outliers & replacing those data entries with NULL values:

```
for (x in c('AVG_Original','AVG_Cut_China', 'AVG_Cut_USA'))
{
  value = df[,x][df[,x] %in% boxplot.stats(df[,x])$out]
  df[,x][df[,x] %in% value] = NA
}
```

Mean of the Average Overall Profits for all three cases:

```
colMeans(df[,2:4], na.rm = TRUE)
```

```
##  AVG_Original AVG_Cut_China   AVG_Cut_USA
##     -14803.30     -14824.44     -14527.46
```

STATISTICAL TESTS

In order to evaluate if cutting off China or USA has any impact on the overall economy of the global semiconductor industry, we perform the **paired t-test**. This statistical procedure determines if the mean differences of the paired cases are statistically significant.

The average monthly profit of all five regions is used as the **performance measure** to compare the systems.

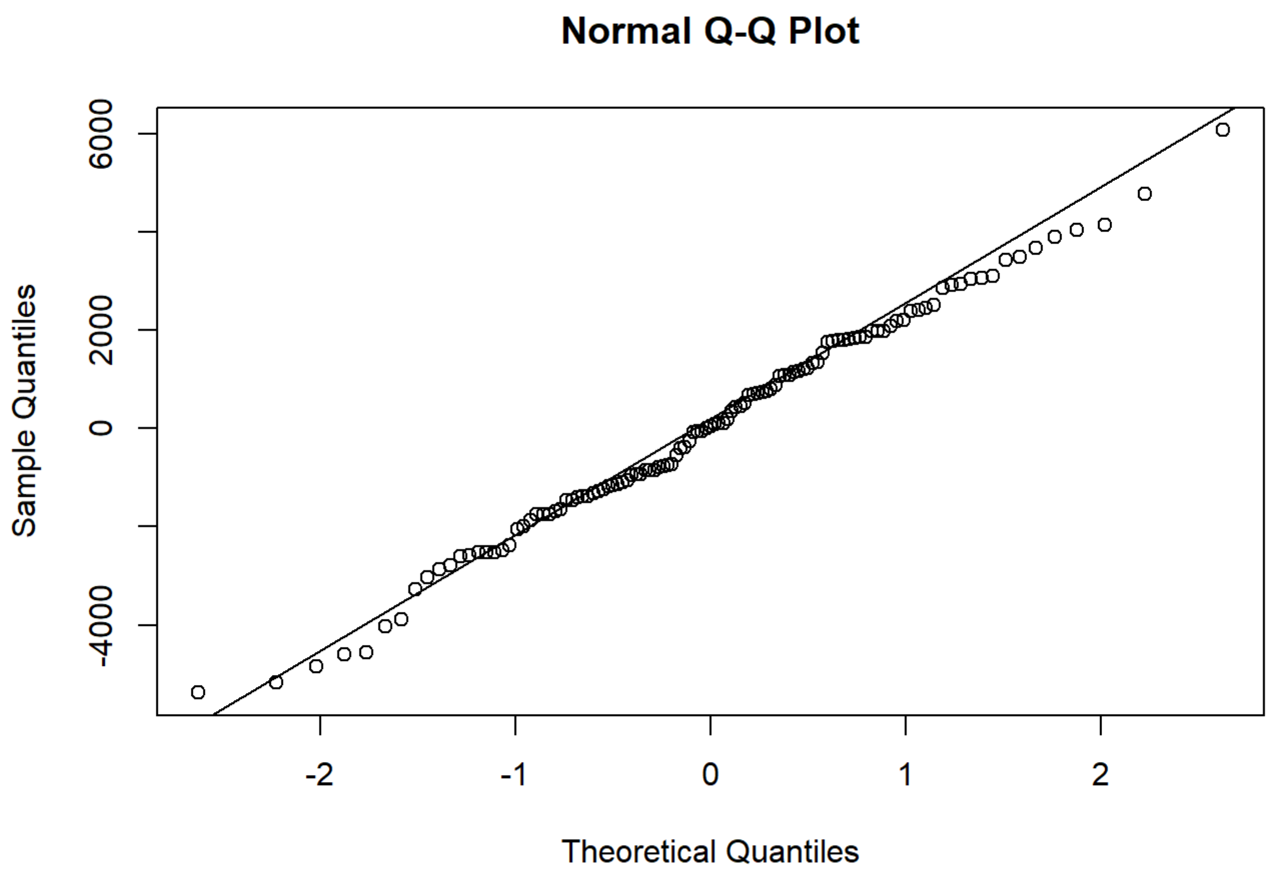
Paired t-test Assumptions:

1. The sample size is adequate with 120 data points per case.
2. There are no outliers in the data.

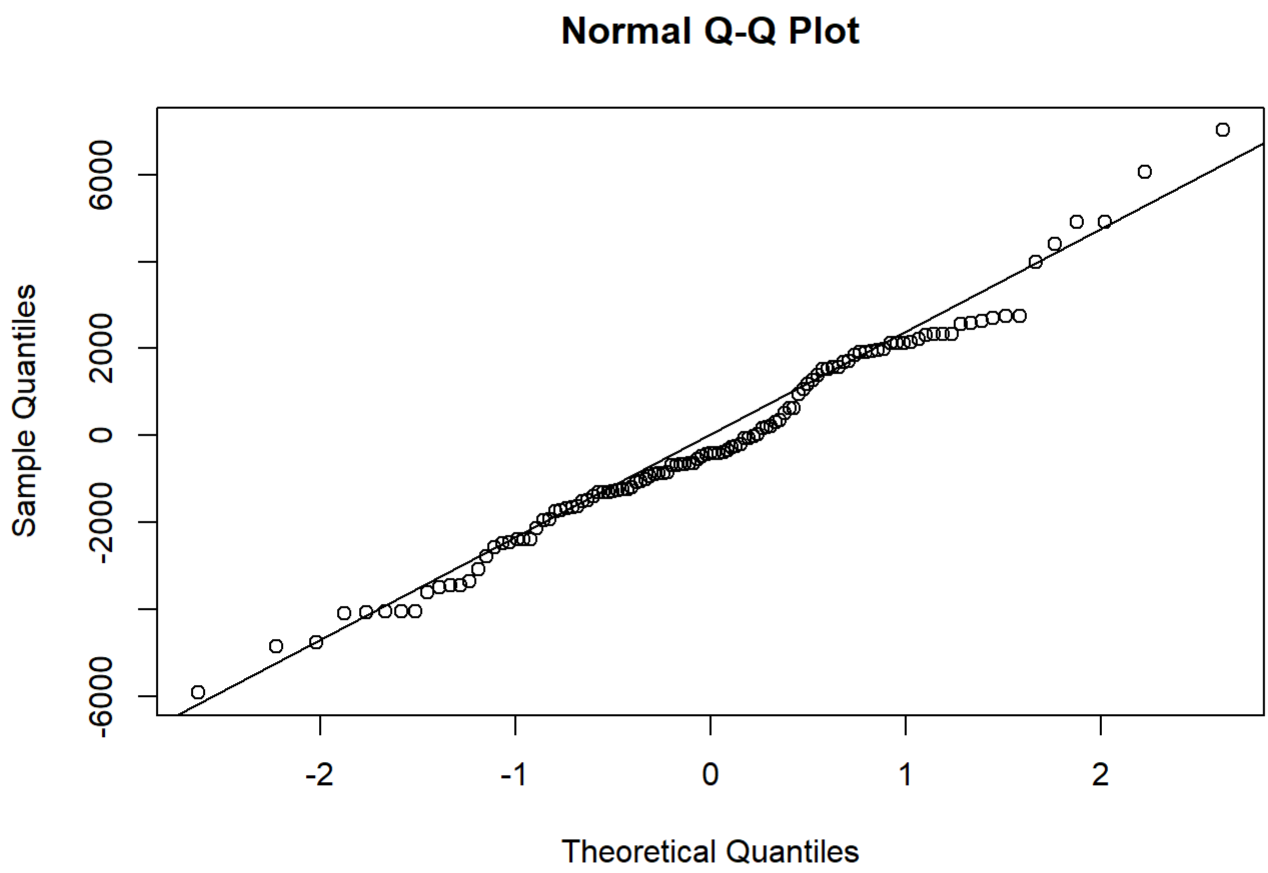
Before proceeding with the paired t-test, we must also check if the differences between the pairs are approximately normally distributed.

Checking for the Assumption of Normality using the Q-Q Plot:

```
#install.packages("stats") -> to perform statistical calculations and tests
library(stats)

difference_OGxCN <- df$AVG_Original - df$AVG_Cut_China
qqnorm(difference_OGxCN)
qqline(difference_OGxCN, datax = FALSE, distribution = qnorm, probs = c(0.25, 0.75))
```




```
difference_OGxUS <- df$AVG_Original - df$AVG_Cut_USA
qqnorm(difference_OGxUS)
qqline(difference_OGxUS, datax = FALSE, distribution = qnorm, probs = c(0.25, 0.75))
```



Since the data points lie closely to the line for both pairs, the assumption of normality has been met!

HYPOTHESIS TESTING: Paired t-test for Comparing Systems

```
t_test_for_case_1 <- t.test(df$AVG_Original, df$AVG_Cut_China, paired = TRUE, data = df)
t_test_for_case_1
```

```
##
##  Paired t-test
##
## data:  df$AVG_Original and df$AVG_Cut_China
## t = 0.21341, df = 114, p-value = 0.8314
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -373.0796  463.1666
## sample estimates:
## mean difference
##        45.04348
```

```
t_test_for_case_2 <- t.test(df$AVG_Original, df$AVG_Cut_USA, paired = TRUE, data = df)
t_test_for_case_2
```

```
##
##  Paired t-test
##
## data:  df$AVG_Original and df$AVG_Cut_USA
## t = -0.85011, df = 114, p-value = 0.397
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -622.0403  248.4751
## sample estimates:
## mean difference
##       -186.7826
```

Since our p-values for both cases are greater than the significance level of 0.05, we fail to reject the null hypothesis which states that the true mean difference is equal to 0 for both cases.

In other words, neither cutting off China nor cutting off USA has any statistically significant impact on the overall profit of the global semiconductor industry.

Therefore, we can infer that the overall economy of the global semiconductor industry is resilient over a decade despite cutting off China & USA from the system.

[NEW] Causal Impact Analysis on Time-Series data:

Though the above statistical test is consistent with the content taught in the SMA module, we realized that the conclusions of the paired t-test will not hold true for our case since the output data generated by our Semicon Simulation is a time-series data.

Hence, our team decided to try a different approach called the **Causal Impact Analysis** to compare the systems for our case.

We install a new R package (CausalImpact) which would aid us to perform this analysis.

MORE DATA MANIPULATION:

The data frame must be manipulated in a desired format of a time-series data to feed into the CausalImpact() function.

```
#install.packages("CausalImpact") <- for time-series causal impact analysis
library(CausalImpact)
```

```
## Warning: package 'CausalImpact' was built under R version 4.2.3
```

```
## Loading required package: bsts
```

```
## Warning: package 'bsts' was built under R version 4.2.3
```

```
## Loading required package: BoomSpikeSlab
```

```
## Warning: package 'BoomSpikeSlab' was built under R version 4.2.3
```

```
## Loading required package: Boom
```

```
## Warning: package 'Boom' was built under R version 4.2.3
```

```
##
## Attaching package: 'Boom'
```

```
## The following object is masked from 'package:stats':
##
##     rWishart
```

```
##
## Attaching package: 'BoomSpikeSlab'
```

```
## The following object is masked from 'package:stats':
##
##     knots
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: xts
```

```
##
## ################################### WARNING ###################################
## # We noticed you have dplyr installed. The dplyr lag() function breaks how    #
## # base R's lag() function is supposed to work, which breaks lag(my_xts).      #
## #                                                                             #
## # Calls to lag(my_xts) that you enter or source() into this session won't     #
## # work correctly.                                                             #
## #                                                                             #
## # All package code is unaffected because it is protected by the R namespace   #
## # mechanism.                                                                  #
## #                                                                             #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                             #
## # You can use stats::lag() to make sure you're not using dplyr::lag(), or you #
## # can add conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop   #
## # dplyr from breaking base R's lag() function.                                #
## ################################### WARNING ###################################
```

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
##
## Attaching package: 'bsts'
```

```
## The following object is masked from 'package:BoomSpikeSlab':
##
##     SuggestBurn
```

```
df <- na.omit(df) # CausalImpact function cannot be used to compare missing values. So, we ke
ep 115 data samples.

dim(df)
```

```
## [1] 115   4
```

```
df$Months <- c(1:115)
head(df)
```

```
##    Months AVG_Original AVG_Cut_China AVG_Cut_USA
## 5       1        -7640         -7240       -7240
## 6       2       -11060        -10220      -11040
## 7       3       -14160        -13320      -14180
## 8       4       -17120        -15840      -16480
## 9       5       -19380        -18460      -18740
## 10      6       -19340        -18500      -18920
```

Creating separate data frames to compare two systems:

```
case2_df <- df[ ,c('AVG_Original','AVG_Cut_China')]
case3_df <- df[ ,c('AVG_Original','AVG_Cut_USA')]

head(case2_df)
```

```
##    AVG_Original AVG_Cut_China
## 5         -7640         -7240
## 6        -11060        -10220
## 7        -14160        -13320
## 8        -17120        -15840
## 9        -19380        -18460
## 10       -19340        -18500
```

```
head(case3_df)
```

```
##    AVG_Original AVG_Cut_USA
## 5         -7640       -7240
## 6        -11060      -11040
## 7        -14160      -14180
## 8        -17120      -16480
## 9        -19380      -18740
## 10       -19340      -18920
```

Since we want to find how the average overall monthly profits of the original global semiconductor industry differs from the average overall monthly profits of the global semiconductor industry after intervention (Cases: cut off China & cut off USA), we append AVG_Cut_China & AVG_Cut_USA columns beneath the AVG_Original column

```
case2_df <- data.frame(AVG_Original = unlist(case2_df, use.names = FALSE))
colnames(case2_df)[1] = "Case_2_Monthly_Profits"
head(case2_df)
```

```
##   Case_2_Monthly_Profits
## 1                  -7640
## 2                 -11060
## 3                 -14160
## 4                 -17120
## 5                 -19380
## 6                 -19340
```

```
case3_df <- data.frame(AVG_Original = unlist(case3_df, use.names = FALSE))
colnames(case3_df)[1] = "Case_3_Monthly_Profits"
head(case3_df)
```

```
##   Case_3_Monthly_Profits
## 1                  -7640
## 2                 -11060
## 3                 -14160
## 4                 -17120
## 5                 -19380
## 6                 -19340
```

We have a total of 230 data points now.

```
# Reality Check!
dim(case2_df)
```

```
## [1] 230   1
```

```
dim(case3_df)
```

```
## [1] 230   1
```

Adding Arbitrary Dates to get the data in the desired time-series data format to perform Causal Impact Analysis:

```
case2_df$time.points <- seq.Date(as.Date("2023-01-01"), by = 1, length.out = 230)
head(case2_df)
```

```
##   Case_2_Monthly_Profits time.points
## 1                  -7640  2023-01-01
## 2                 -11060  2023-01-02
## 3                 -14160  2023-01-03
## 4                 -17120  2023-01-04
## 5                 -19380  2023-01-05
## 6                 -19340  2023-01-06
```

```
case3_df$time.points <- seq.Date(as.Date("2023-01-01"), by = 1, length.out = 230)
head(case3_df)
```

```
##   Case_3_Monthly_Profits time.points
## 1                  -7640  2023-01-01
## 2                 -11060  2023-01-02
## 3                 -14160  2023-01-03
## 4                 -17120  2023-01-04
## 5                 -19380  2023-01-05
## 6                 -19340  2023-01-06
```

Specifying pre-intervention and post-intervention periods:

```
# Getting the indexes to extract date values
case2_df[150, ]
```

```
##     Case_2_Monthly_Profits time.points
## 150                  -9620  2023-05-30
```

```
case2_df[151, ]
```

```
##     Case_2_Monthly_Profits time.points
## 151                  -9320  2023-05-31
```

```
case2_df[230, ]
```

```
##     Case_2_Monthly_Profits time.points
## 230                 -13400  2023-08-18
```

```
# It is the same for case 3 data frame so we don't have to repeat the steps

pre.period <- seq.Date(as.Date("2023-01-01"), as.Date("2023-05-30"), "years")
post.period <- seq.Date(as.Date("2023-05-31"), as.Date("2023-08-18"), "years")
```

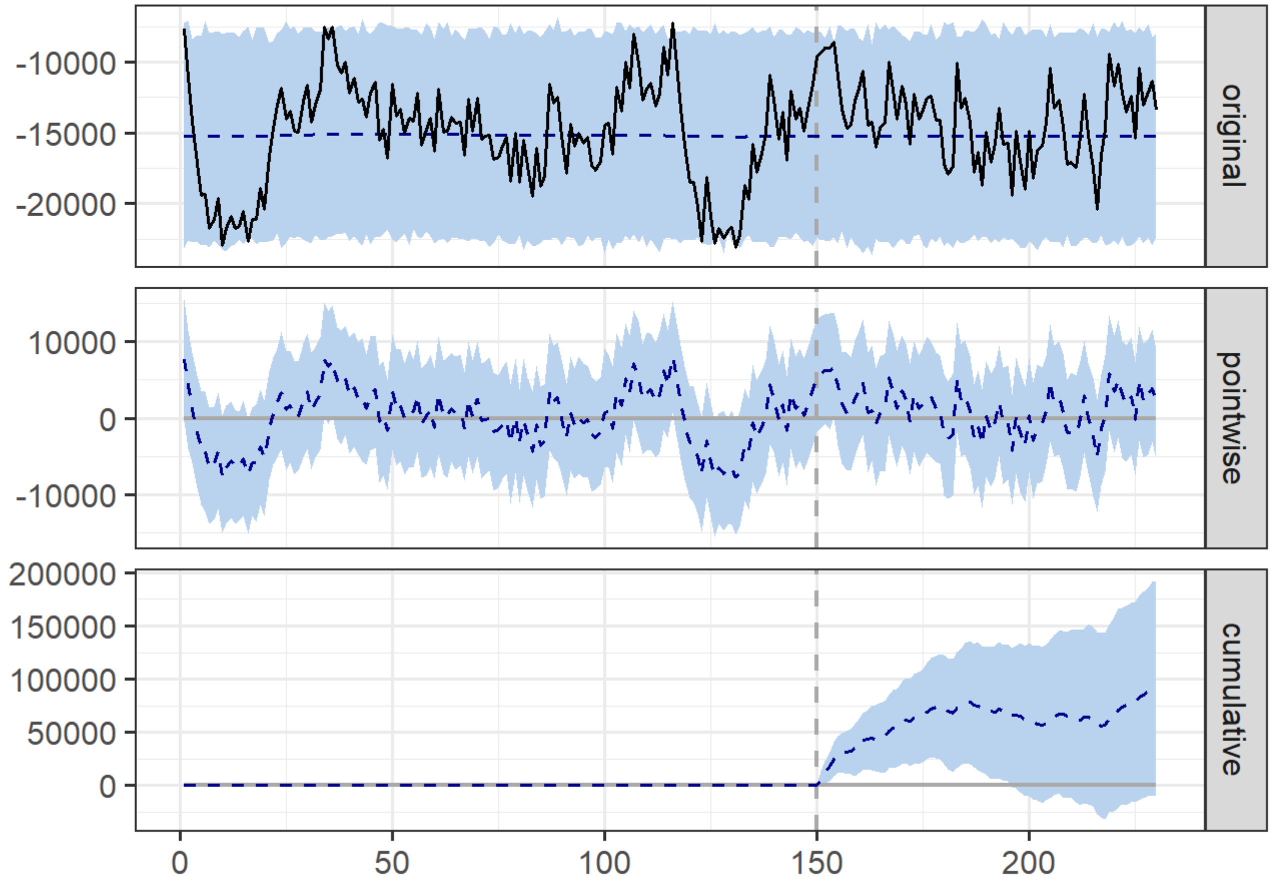
```
head(case2_df)
```

```
##   Case_2_Monthly_Profits time.points
## 1                  -7640  2023-01-01
## 2                 -11060  2023-01-02
## 3                 -14160  2023-01-03
## 4                 -17120  2023-01-04
## 5                 -19380  2023-01-05
## 6                 -19340  2023-01-06
```

```
head(case3_df)
```

```
##   Case_3_Monthly_Profits time.points
## 1                  -7640  2023-01-01
## 2                 -11060  2023-01-02
## 3                 -14160  2023-01-03
## 4                 -17120  2023-01-04
## 5                 -19380  2023-01-05
## 6                 -19340  2023-01-06
```

```
pre.period = c(1,150)
post.period = c(151, 230)

case2_impact <- CausalImpact(data = case2_df$Case_2_Monthly_Profits, pre.period, post.period)
plot(case2_impact)
```



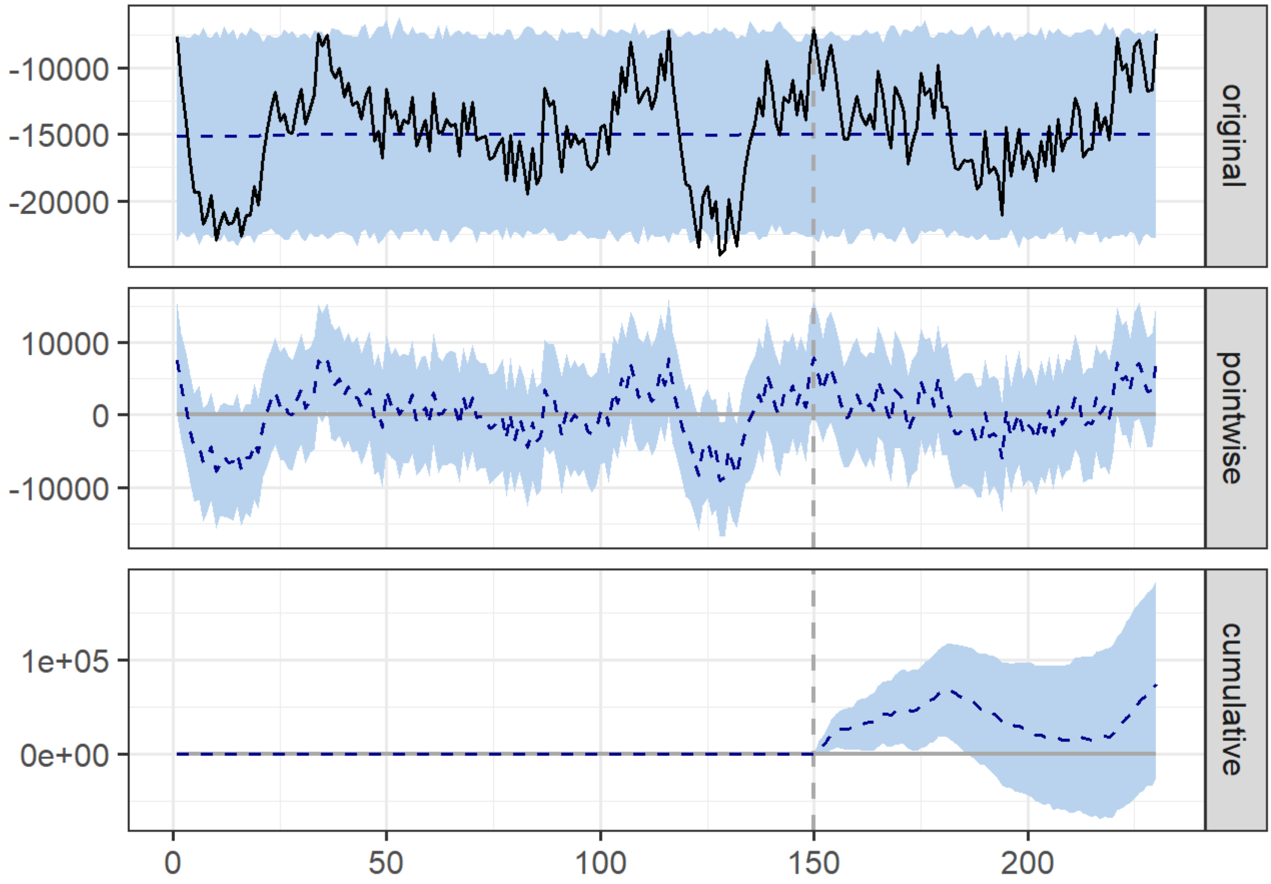

```r
summary(case2_impact)
```

```
## Posterior inference {CausalImpact}
##
##                          Average            Cumulative
## Actual                   -14055             -1124420
## Prediction (s.d.)        -15233 (649)       -1218615 (51923)
## 95% CI                   [-16457, -13929]   [-1316581, -1114298]
##
## Absolute effect (s.d.)   1177 (649)         94195 (51923)
## 95% CI                   [-127, 2402]       [-10122, 192161]
##
## Relative effect (s.d.)   -7.7% (3.9%)       -7.7% (3.9%)
## 95% CI                   [-15%, 0.91%]      [-15%, 0.91%]
##
## Posterior tail-area probability p:   0.03982
## Posterior prob. of a causal effect:  96.018%
##
## For more details, type: summary(impact, "report")
```

```r
summary(case2_impact,"report")
```

```
## Analysis report {CausalImpact}
## 
## 
## During the post-intervention period, the response variable had an average value of approx.
-14.06K. In the absence of an intervention, we would have expected an average response of -1
5.23K. The 95% interval of this counterfactual prediction is [-16.46K, -13.93K]. Subtracting
this prediction from the observed response yields an estimate of the causal effect the interv
ention had on the response variable. This effect is 1.18K with a 95% interval of [-0.13K, 2.4
0K]. For a discussion of the significance of this effect, see below.
## 
## Summing up the individual data points during the post-intervention period (which can only
sometimes be meaningfully interpreted), the response variable had an overall value of -1.12M.
Had the intervention not taken place, we would have expected a sum of -1.22M. The 95% interva
l of this prediction is [-1.32M, -1.11M].
## 
## The above results are given in terms of absolute numbers. In relative terms, the response
variable showed a decrease of -8%. The 95% interval of this percentage is [-15%, +1%].
## 
## This means that, although it may look as though the intervention has exerted a negative ef
fect on the response variable when considering the intervention period as a whole, this effec
t is not statistically significant, and so cannot be meaningfully interpreted. The apparent e
ffect could be the result of random fluctuations that are unrelated to the intervention. This
is often the case when the intervention period is very long and includes much of the time whe
n the effect has already worn off. It can also be the case when the intervention period is to
o short to distinguish the signal from the noise. Finally, failing to find a significant effe
ct can happen when there are not enough control variables or when these variables do not corr
elate well with the response variable during the learning period.
## 
## The probability of obtaining this effect by chance is very small (Bayesian one-sided tail-
area probability p = 0.04). This means the causal effect can be considered statistically sign
ificant.
```

```
case3_impact <- CausalImpact(data = case3_df$Case_3_Monthly_Profits, pre.period, post.period)
plot(case3_impact)
```




```
summary(case3_impact)
```

```
## Posterior inference {CausalImpact}
##
##                          Average            Cumulative
## Actual                   -14034             -1122680
## Prediction (s.d.)        -14960 (669)       -1196822 (53481)
## 95% CI                   [-16340, -13708]   [-1307211, -1096666]
##
## Absolute effect (s.d.)   927 (669)          74142 (53481)
## 95% CI                   [-325, 2307]       [-26014, 184531]
##
## Relative effect (s.d.)   -6.1% (4.2%)       -6.1% (4.2%)
## 95% CI                   [-14%, 2.4%]       [-14%, 2.4%]
##
## Posterior tail-area probability p:   0.07281
## Posterior prob. of a causal effect:  93%
##
## For more details, type: summary(impact, "report")
```

```
summary(case3_impact,"report")
```

```
## Analysis report {CausalImpact}
##
##
## During the post-intervention period, the response variable had an average value of approx.
-14.03K. In the absence of an intervention, we would have expected an average response of -1
4.96K. The 95% interval of this counterfactual prediction is [-16.34K, -13.71K]. Subtracting
this prediction from the observed response yields an estimate of the causal effect the interv
ention had on the response variable. This effect is 0.93K with a 95% interval of [-0.33K, 2.3
1K]. For a discussion of the significance of this effect, see below.
##
## Summing up the individual data points during the post-intervention period (which can only
sometimes be meaningfully interpreted), the response variable had an overall value of -1.12M.
Had the intervention not taken place, we would have expected a sum of -1.20M. The 95% interva
l of this prediction is [-1.31M, -1.10M].
##
## The above results are given in terms of absolute numbers. In relative terms, the response
variable showed a decrease of -6%. The 95% interval of this percentage is [-14%, +2%].
##
## This means that, although it may look as though the intervention has exerted a negative ef
fect on the response variable when considering the intervention period as a whole, this effec
t is not statistically significant, and so cannot be meaningfully interpreted. The apparent e
ffect could be the result of random fluctuations that are unrelated to the intervention. This
is often the case when the intervention period is very long and includes much of the time whe
n the effect has already worn off. It can also be the case when the intervention period is to
o short to distinguish the signal from the noise. Finally, failing to find a significant effe
ct can happen when there are not enough control variables or when these variables do not corr
elate well with the response variable during the learning period.
##
## The probability of obtaining this effect by chance is p = 0.073. This means the effect may
be spurious and would generally not be considered statistically significant.
```

Limitations:

- More research on ways to properly initialize data generated from terminating simulations could have been performed for its implementation in R.
- Averaging the monthly profits might not be the most robust performance measure to assess the overall performance of the global semiconductor industry.
- Indicator variables that directly affect the response variable (AVG_Profits) should have been recognized for its inclusion in the analysis in order to produce highly accurate statistical results.

Appendix:

Another interesting use case of the Semicon Simulation would be to look to the impact of the removal of China & USA on the monthly profits of individual countries. This can be visualized using box plots!