Quality Assurance 344

ECSA Final Report

27 October 2025

Marzuq Sirkhoth

27487644

Contents

Abstract

This project shows how sophisticated statistics and data analyses techniques can be used effectively for process assessment, monitoring, and optimization in the industrial and service sectors. R programming was used to apply description statistics, Statistical Process Control (SPC), as well as Process Capability Indices (Cp/Cpk), to assess the product's performance and reliability of delivery. Type I error probability calculations were also employed as an error measure of first kind in process monitoring. Data corrections or transformation of data was also tackled in this project through consistency gains in the product data sets. Profit optimization models concerning barista employment, service dependability, and unit costs of operation were built in the context of a coffee shop business. The same was done in the context of a car rental agency by balancing employment schedules with reliability of sales. Product performance was analysed relative to time-periods and sources of data employing multivariate data analyses (MANOVA or ANOVA tests).

Question 1: Summary Statistics

This section of the report provides a data analysis of all the given CSV files. It provides a quick summary of each dataset, and then provides a more in-depth analysis with the aid of visualizations.

Products – Branch

As is evident in Figure 1, 60 products have complete data in the branch list. Selling prices are highly skewed: the mean is R4,493 but the median is only R794, which means that a few expensive premium items (as high as R19,725) are pulling the mean upwards, but the majority of items are sold at lower prices. Markups are uniform at an average of 20.5% and between 10% and 30%, reflecting stable setting of margins. The portfolio features a two-tier composition: price-sensitive volume in lower-priced products, and lopsided revenue-contributing products, reflecting that inventory and promotional strategies should be evenly balanced.

Figure 1:

```
Rows: 60 Cols: 5
  ProductID           Category      Description SellingPrice Markup
1    SOF001           Software       coral matt       511.53  25.05
2    SOF002 Cloud Subscription        cyan silk       505.26  10.43
3    SOF003             Laptop burlywood marble       493.69  16.18
Missing per column:
   ProductID     Category  Description SellingPrice       Markup
           0            0            0            0            0
Numeric describe (if any):
             vars  n    mean      sd median trimmed    mad    min      max
SellingPrice    1 60 4493.59 6503.77 794.18 3189.25 525.72 350.45 19725.18
Markup          2 60   20.46    6.07  20.34   20.51   7.31  10.13    29.84
                range  skew kurtosis     se
SellingPrice 19374.73  1.43     0.43 839.63
Markup          19.71 -0.04    -1.24   0.78
```

Products – Head Office

In Figure 2, it is evident that the head office catalogue consists of 360 products, with complete details once more. Pricing is a reflection of branch organization: mean R4,411 and median R797, with values up to R22,420 pushing the mean upward. Markups are all 20.4%. The huge head office catalogue lends credence to the notion that branch products are a selection, and this implies that branch alternatives must be co-ordinated to high-margin merchandise and top-selling low-margin items if sales and profitability are to be optimized.

Figure 2:

```
Rows: 360 Cols: 5
  ProductID Category      Description SellingPrice
1    SOF001 Software       coral silk       521.72
2    SOF002 Software       black silk       466.95
3    SOF003 Software burlywood marble       496.43
  Markup
1  15.65
2  28.42
3  20.07
Missing per column:
   ProductID     Category  Description SellingPrice
           0            0            0            0
      Markup
           0
Numeric describe (if any):
             vars   n    mean      sd median trimmed
SellingPrice    1 360 4410.96 6463.82 797.22 3054.23
Markup          2 360   20.39    5.67  20.58   20.43
                mad    min      max    range  skew
SellingPrice 515.75 290.52 22420.14 22129.62  1.53
Markup         6.66  10.06    30.00    19.94 -0.05
             kurtosis     se
SellingPrice     0.78 340.67
Markup          -1.07   0.30
```

Customers

Figure 3 indicates that the sample consists of 5,000 customers with no missing data. The age range is 16–105, with a mean of 52, showing a broad demographic spread. Income averages R80,797, median of R85,000, and varies from R5,000–R140,000, showing economic diversity. Customers are spread across major American cities like New York, Houston, and Chicago, facilitating segmentation by age, income, and geography for product promotion and targeting.

Figure 3:

```
Rows: 5000 Cols: 5
  CustomerID Gender Age Income     City
1    CUST001   Male  16  65000 New York
2    CUST002 Female  31  20000  Houston
3    CUST003   Male  29  10000  Chicago
Missing per column:
CustomerID     Gender        Age     Income       City
         0          0          0          0          0
Numeric describe (if any):
       vars    n     mean       sd median  trimmed      mad  min    max  range  skew kurtosis     se
Age       1 5000    51.55    21.22     51    50.88    26.69   16    105     89  0.20    -0.99   0.30
Income    2 5000 80797.00 33150.11  85000 81665.00 37065.00 5000 140000 135000 -0.21    -0.75 468.81
```

Sales

As shown in Figure 4, there are 100,000 full transactions in sales records. Orders are characterized by mean of 13.5 units and median of 6 with combination of bulk and small orders. Orders happen evenly day and night, twelve months of the year, during 2022–2023. Operating statistics are indicative of mean picking of 14.7 hours and mean delivery of 17.48 hours, with variation suggesting some standard logistical congestion. This data set is revealing of customer demand, purchasing behaviour, and operating efficiency, guiding inventory planning, warehouse personnel, and delivery optimization.

Figure 4:

```
Rows: 100000 Cols: 12
  CustomerID ProductID Quantity orderTime orderDay orderMonth orderYear pickingHours deliveryHours OrderDate_parsed OrderDate_made  OrderDate
1   CUST1791    CLO011       16        13       11         11      2022     17.72167        24.544               NA     2022-11-11 2022-11-11
2   CUST3172    LAP026       17        17       14          7      2023     38.39083        31.546               NA     2023-07-14 2023-07-14
3   CUST1022    KEY046       11        16       23          5      2022     14.72167        21.544               NA     2022-05-23 2022-05-23
Missing per column:
      CustomerID        ProductID         Quantity        orderTime         orderDay       orderMonth        orderYear     pickingHours
               0                0                0                0                0                0                0                0
   deliveryHours OrderDate_parsed   OrderDate_made        OrderDate
               0           100000              560              560
Numeric describe (if any):
              vars     n    mean    sd  median trimmed   mad     min     max range  skew kurtosis   se
Quantity         1 1e+05   13.50 13.76    6.00   11.46  5.93    1.00   50.00 49.00  1.04    -0.22 0.04
orderTime        2 1e+05   12.93  5.50   13.00   13.12  5.93    1.00   23.00 22.00 -0.23    -0.71 0.02
orderDay         3 1e+05   15.50  8.65   15.00   15.50 10.38    1.00   30.00 29.00  0.00    -1.20 0.03
orderMonth       4 1e+05    6.45  3.28    6.00    6.45  4.45    1.00   12.00 11.00  0.01    -1.18 0.01
orderYear        5 1e+05 2022.46  0.50 2022.00 2022.45  0.00 2022.00 2023.00  1.00  0.15    -1.98 0.00
pickingHours     6 1e+05   14.70 10.39   14.05   13.54  6.92    0.43   45.06 44.63  0.74     0.41 0.03
deliveryHours    7 1e+05   17.48 10.00   19.55   17.78  8.90    0.28   38.05 37.77 -0.47    -0.87 0.03
```

Potential Errors - (Mismatches between Products data and Products Head Office data)

Figure 5 illustrates product data between branch and head office systems for the same ProductIDs, highlighting differences in category, description, selling price, and markup. Huge groups of products are categorized differently in the two systems, e.g., SOF002 as a "Cloud Subscription" at branch but "Software" at head office. Description will also differ, i.e., SOF001 is "coral matt" in the branch but "coral silk" in head office. Sales price and markup rate also differ, e.g., in the case of SOF002, whose branch markup is 10.43% to 28.42% at head office. Generally, nearly all duplicating ProductIDs have at least one difference, i.e., significant data inconsistency. Figure 5 can subsequently be used as a guide for their identification, inspection, and remediation to align the head office and branch product records.

Figure 5:

ProductID	Branch (Category & Description)	Head Office (Category & Description)	Branch Price / Markup	Head Price / Markup
SOF001	Software, coral matt	Software, coral silk	511.53 / 25.05	521.72 / 15.65
SOF002	Cloud Subscription, cyan silk	Software, black silk	505.26 / 10.43	466.95 / 28.42
SOF003	Laptop, burlywood marble	Software, burlywood marble	493.69 / 16.18	496.43 / 20.07
SOF004	Monitor, blue silk	Software, black marble	542.56 / 17.19	389.33 / 17.25
SOF005	Keyboard, aliceblue wood	Software, chartreuse sandpaper	516.15 / 11.01	482.64 / 17.6
SOF006	Mouse, black silk	Software, cornflowerblue marble	478.93 / 16.99	539.33 / 25.57
SOF007	Software, black bright	Software, blue marble	527.56 / 16.79	495.13 / 10.23
SOF008	Cloud Subscription, burlywood silk	Software, cornflowerblue marble	549.02 / 11.95	465.73 / 21.89
SOF009	Laptop, azure sandpaper	Software, black bright	540.41 / 11.34	452.4 / 19.64
SOF010	Monitor, chocolate sandpaper	Software, cornflowerblue matt	396.72 / 23.47	399.43 / 17.08

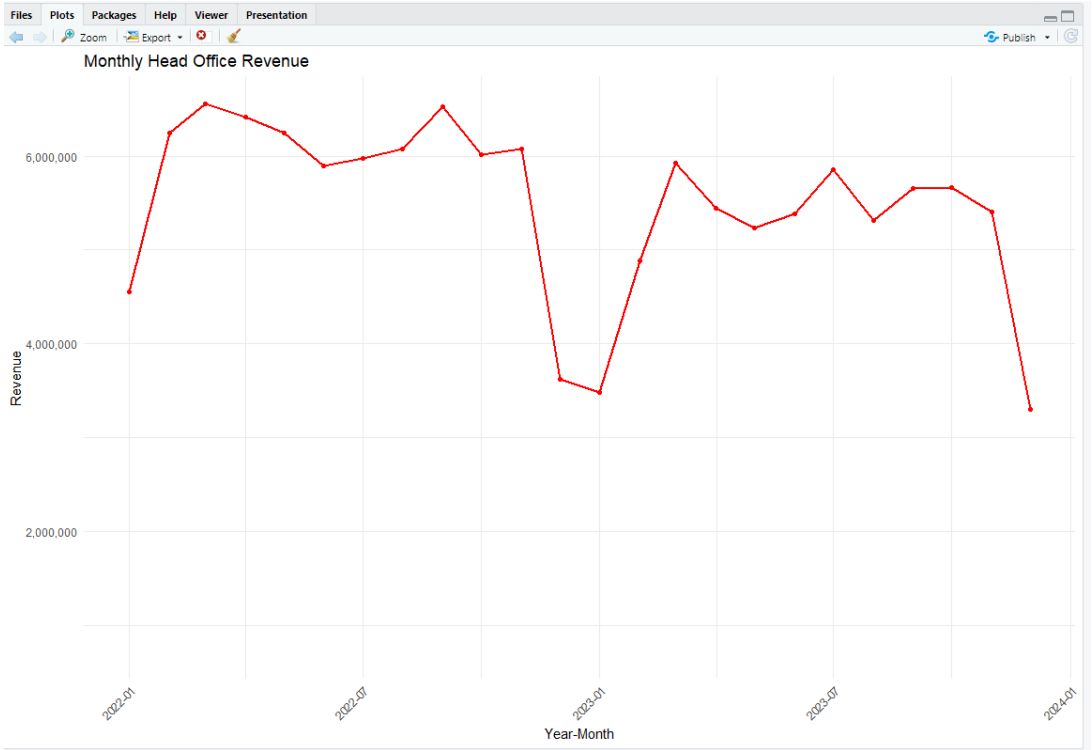
Deeper Analysis + Visualizations

Monthly Products (Head Office Revenue):

Figure 5.1 illustrates that the head office revenue is somewhere between 3 million and 6.5 million monthly, which is much less compared to branch revenue. This means fewer direct sales or lower transactions occur at the head office. In consideration of trends, there are a few spikes in mid-2022, followed by a sharp decline towards the start of 2023, then an even recovery. The drops may be caused by low-demand months, holidays, or operational problems like supply chain problems, while the surges may be caused by promotions, bulk orders, or seasonal demand like end-of-year or back-to-school seasons.

There is head office revenue volatility with up to 50% monthly drops from peak to trough, demonstrating vulnerability to fluctuations in orders or capacity. From a commercial perspective, what this means is that head office only receives a portion of the overall company revenues, and growth strategies must factor for this. To cut down on fluctuations, the company will be looking to enter stable pipelines of orders or multiple bases of customers for sales at head office. Also, massive revenues decline indicate a possible reliance on some significant clients or seasonality sales risk that can be monitored more easily.

Figure 5.1:

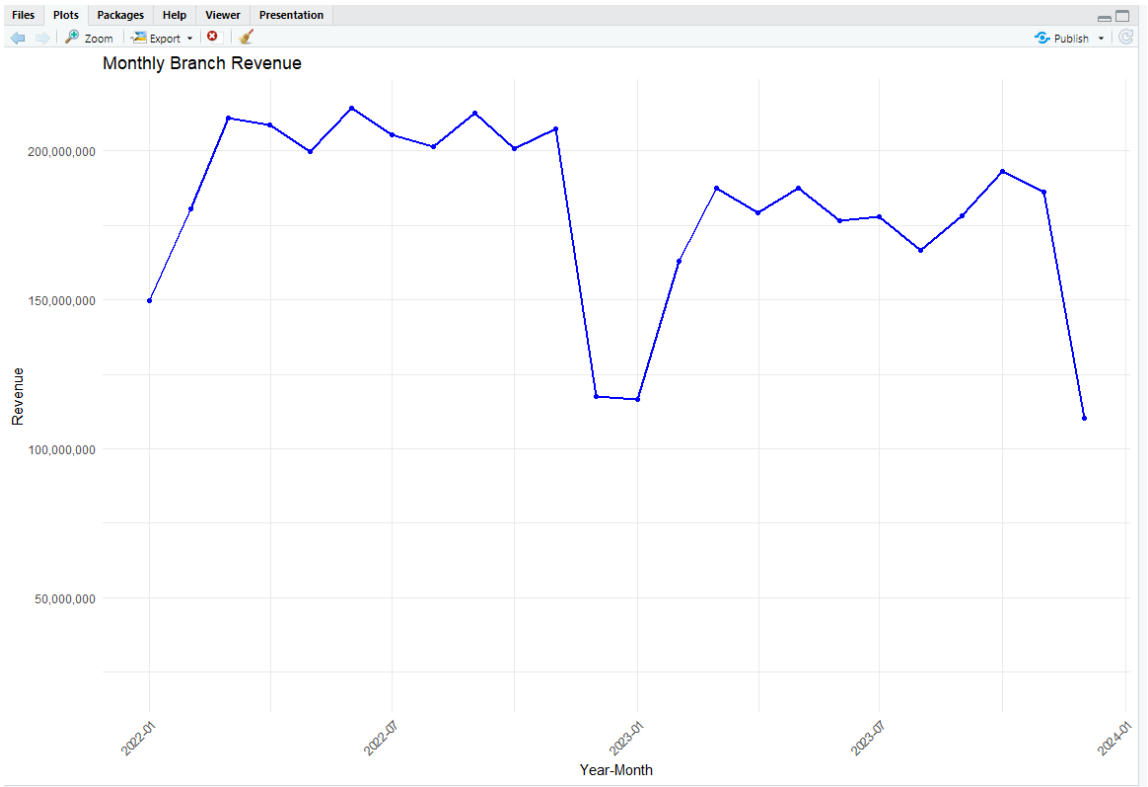


Monthly Products (Branch) revenue:

Figure 5.2 illustrates how branch revenues are around 110 million to 220 million per month, much higher than revenues at head office and therefore the main revenue earners. The revenue pattern has similar peaks in mid-2022 and steep declines in early 2023, similar to that in the head office. That they are similar implies seasonality or overall market forces—such as supply chain stress or macroeconomic patterns—alike in the head office and the branches.

Even higher absolute volatility, branch revenue is relatively stable in relation to mean revenue. Stability comes from higher customer volume, repeat orders, and long-term client relationships. From a business standpoint, this highlights the company's dependence on branch sales in the sense that any disruption in operations within branches, e.g., stockout or labour shortages, would have a disproportionate effect on total revenue. As a strategy to maximize growth, the company would invest in marketing, employees, and inventory in branches and would also consider other options such as cross-selling, promotional activities, or new high-volume branch openings.

Figure 5.2:



Top 10 Categories by Revenue:

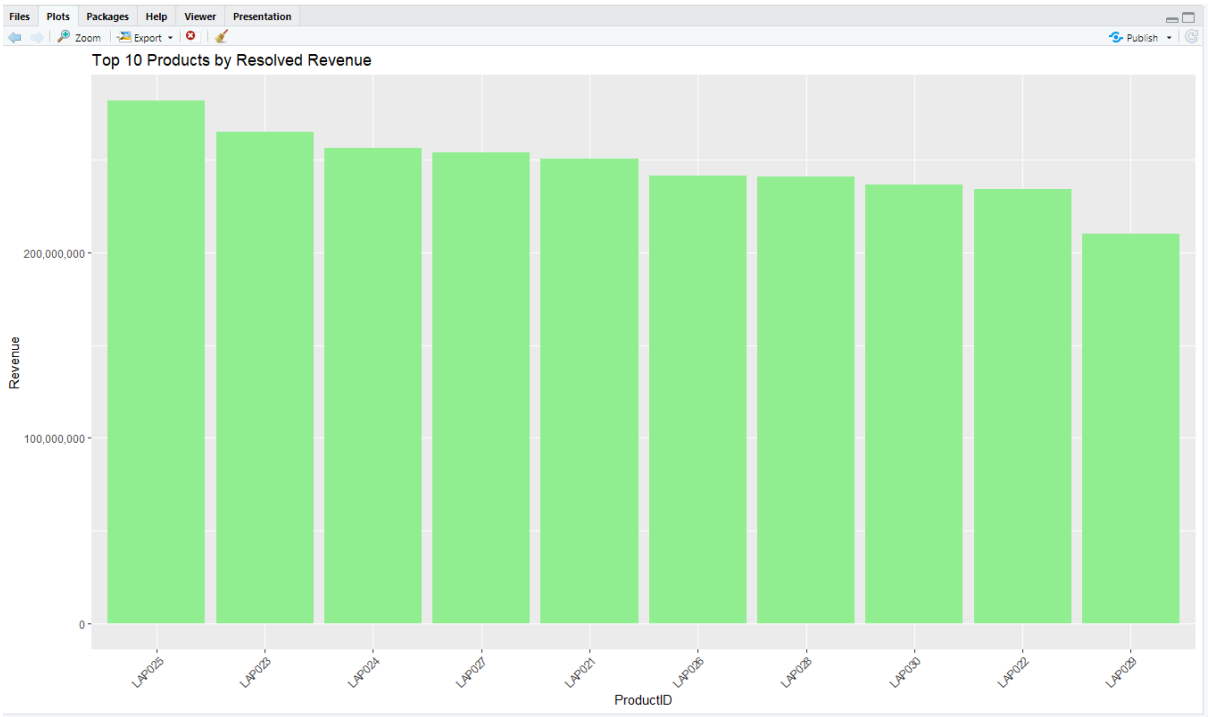
Figure 5.3 showcases the top 10 categories by revenue. Monitors and laptops are ranking number one and two, both of which generated over 800 million in paid revenue. This informs us that the majority of sales are being driven by hardware categories, especially high-end items such as laptops. Accessories such as mice and keyboards also contribute significantly, suggesting a large number of customers are purchasing add-ons or related items in addition to significant hardware. Software and cloud is lower, suggesting digital and service-based revenues are still good but are not the main driver of sales. From a business perspective, this serves to emphasize the importance of maintaining competitive stock and price levels in the laptop and monitor sections and packaging accessories and services (like software and subscription services) in a manner that will create margins.

Figure 5.3:

Top 10 Products by Revenue:

Figure 5.4 illustrates that the top 10 are all laptop models (such as LAP025, LAP023, LAP024), and LAP025 is the revenue leader. Based on the plot, it can be deduced that the firm's strategy toward its laptops as a product category is multi-tailed — customers are buying across the models. From a business standpoint, this reduces risk (because dependence on one model is minimal) but also highlights the need for sound inventory management with lots of laptop SKUs. It may also be helpful to examine customer affinity (city and segment) per laptop model to drive future product releases and promotions.

Figure 5.4:



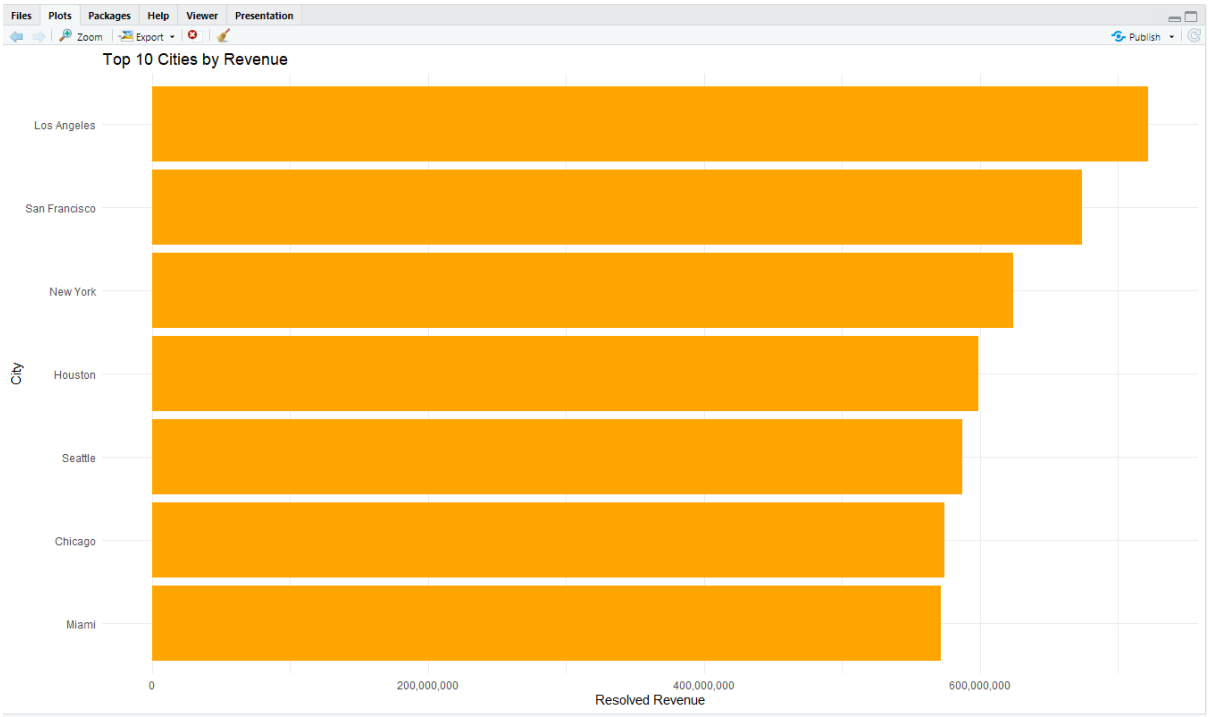
Top 10 Cities by Revenue:

Figure 5.5 illustrates a spatial distribution plot of the sales. Los Angeles is at the top with the most revenue, followed by San Francisco and New York. They are big, affluent markets that are heavily demanding high-tech products, evident in monitor and laptop computer sales leadership. Houston, Seattle, Chicago, and Miami make up the rest of the top 10, and demand would appear not to be relegated to the customary high-tech hubs. On a corporate level, this means regional sales strategies knowledge:

- Tech-hub cities (San Francisco, Los Angeles, New York) will likely outshine to first-class product launches and pre-access initiatives.
- Growth cities (Houston, Miami, Seattle) can potentially provide sites for targeted promotions, local marketing, or logistics streamlining. And high revenues in most cities indicate that the distribution system is solid.

It is hence evident that hardware (laptops/monitors) is responsible for most of the company's revenue. Product range in laptops spreads risk, with potential to segment marketing and logistics by city and potential to expand in upselling software and subscriptions alongside hardware in a bid to drive recurring revenue.

Figure 5.5:



Question 3: Quality Control Charts

3.1 A script was developed in R, to plot s-charts and X-charts for the delivery times. To calculate the respective LCL, UCL, and CL for each plot, formulas 4 and 5 were used from Figure 6. The corresponding calculations are displayed in Figure 7.1 and Figure 7.2 respectively, and the X-chart and s-chart outputs are depicted in Figure 8 and Figure 9 respectively.

Figure 6:

Type of Chart	LCL	CL	UCL
$\bar{x}$ (with R)	$\bar{\bar{x}} - A_2\bar{R}$	$\bar{\bar{x}}$	$\bar{\bar{x}} + A_2\bar{R}$
R	$D_3\bar{R}$	$\bar{R}$	$D_4\bar{R}$
p	$\bar{p} - 3\sqrt{\bar{p}(1-\bar{p})/n}$	$\bar{p}$	$\bar{p} + 3\sqrt{\bar{p}(1-\bar{p})/n}$
$\bar{x}$ (with s)	$\bar{\bar{x}} - A_3\bar{s}$	$\bar{\bar{x}}$	$\bar{\bar{x}} + A_3\bar{s}$
s	$B_3\bar{s}$	$\bar{s}$	$B_4\bar{s}$
x	$\bar{x} - 3\bar{R}/d_2$	$\bar{x}$	$\bar{x} + 3\bar{R}/d_2$
np	$n\bar{p} - 3\sqrt{n\bar{p}(1-\bar{p})}$	$n\bar{p}$	$n\bar{p} + 3\sqrt{n\bar{p}(1-\bar{p})}$
c	$\bar{c} - 3\sqrt{\bar{c}}$	$\bar{c}$	$\bar{c} + 3\sqrt{\bar{c}}$
u	$\bar{u} - 3\sqrt{\bar{u}/n_i}$	$\bar{u}$	$\bar{u} + 3\sqrt{\bar{u}/n_i}$

Figure 7.1 : Constants:

- C4 : (Www.sfu.ca, 2025)
- A3, B3,B4 : (https://www.facebook.com/kenith.grey.1, 2019)

```
c4 <- function(n) {
  return(sqrt(2/(n-1)) * gamma(n/2) / gamma((n-1)/2))
}

control_constants <- function(n) {
  c4_val <- c4(n)

  # A3, B3, B4 formulas
  A3 <- 3 / (c4_val * sqrt(n))
  B3 <- 1 - 3 * sqrt(1 - c4_val^2)
  B4 <- 1 + 3 * sqrt(1 - c4_val^2)

  return(list(A3 = A3, B3 = B3, B4 = B4, c4 = c4_val))
}

n <- 24
constants <- control_constants(n)
constants
```

Figure 7.2 : Control Limits

```
# X-bar chart
xbar_cl <- xbar_bar
xbar_ucl <- xbar_bar + A3*s_bar
xbar_lcl <- xbar_bar - A3*s_bar
xbar_1sigma_upper <- xbar_cl + (xbar_ucl - xbar_cl)/3
xbar_1sigma_lower <- xbar_cl - (xbar_cl - xbar_lcl)/3
xbar_2sigma_upper <- xbar_cl + 2*(xbar_ucl - xbar_cl)/3
xbar_2sigma_lower <- xbar_cl - 2*(xbar_cl - xbar_lcl)/3

# s-chart
s_cl <- s_bar
s_ucl <- B4*s_bar
s_lcl <- B3*s_bar
s_1sigma_upper <- s_cl + (s_ucl - s_cl)/3
s_1sigma_lower <- s_cl - (s_cl - s_lcl)/3
s_2sigma_upper <- s_cl + 2*(s_ucl - s_cl)/3
s_2sigma_lower <- s_cl - 2*(s_cl - s_lcl)/3
```

Figure 8 : X-chart

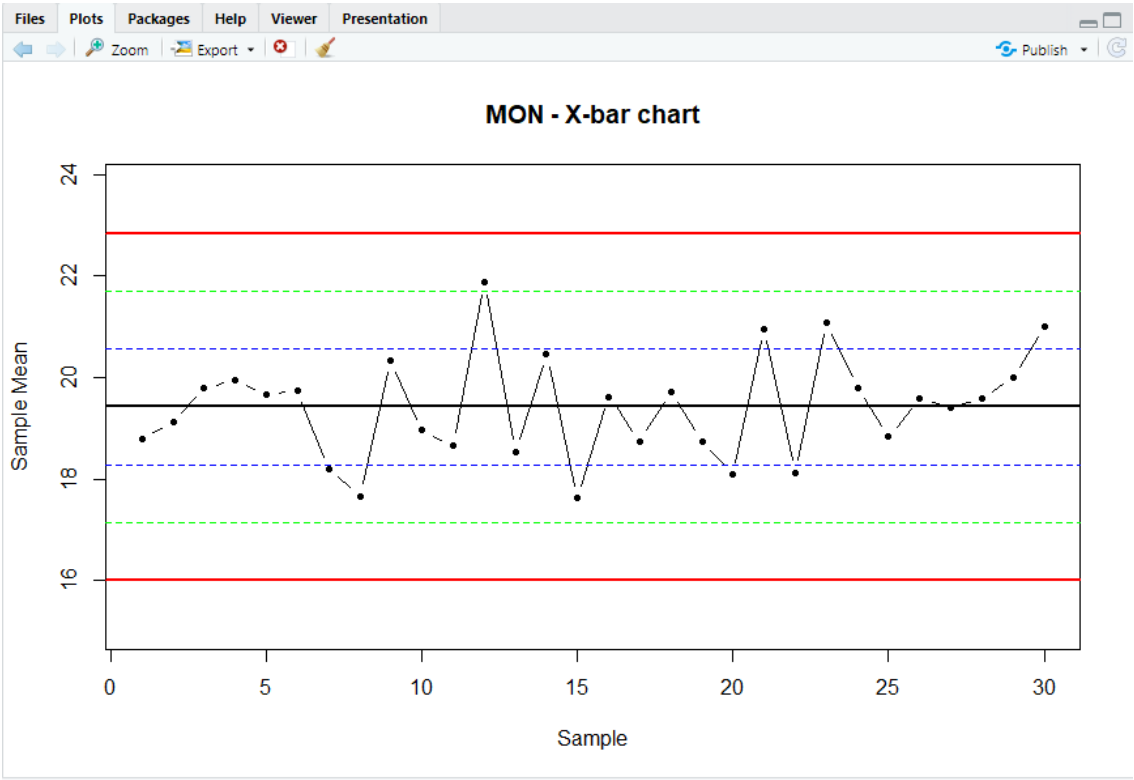
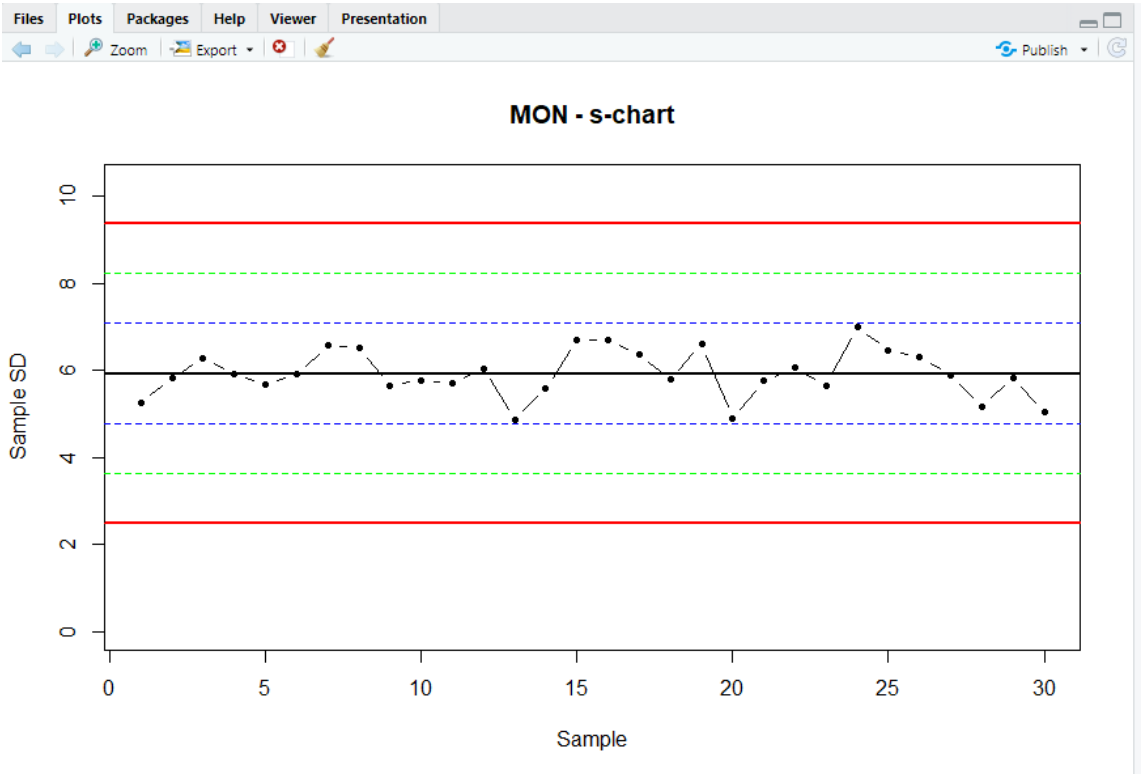


Figure 9: s-chart



The values calculated from the R script are as follows:

- For the X-chart, the CL was calculated to be 19.43, the UCL was calculated to be 22.84 and the LCL was calculated to be 16.01.
- Similarly for the s-chart, its corresponding CL was calculated to be 5.92, its UCL = 9.36 and its LCL = 2.48.

3.2 Each of the following samples (numbered 31, 32, ...) was tested in turn. First, the sample standard deviation was checked against the limits of the s-chart, because an out-of-control s disqualifies the sample mean for interpretation. The sample mean was then compared with the limits of the $\bar{X}$ chart.

In real-life implementation, delivery process data would be received one by one as they are acquired in sales. During continuous process monitoring, 24 samples of delivery are examined as they are acquired and the s-chart is initially verified for proper interpretation of $\bar{X}$ chart.
From the Phase-2 results (sample 31 and onward), product managers would note $\bar{X}$ out-of-control samples, as no s-chart violations were reported. Figure 10 below shows the number of Phase-2 samples beyond control limits for each product type:

Figure 10:

ProductType	Total_Phase2_Samples	Out_of_Control_Xbar	Out_of_Control_S	In_Control
MOU	830	309	0	521
KEY	716	277	0	439
SOF	834	326	0	508
CLO	619	237	0	382
LAP	395	126	0	269
MON	589	199	0	390

3.3 In order to determine which process times are contained in the VOC, the four values in Figure 11 were calculated. Cp is whether or not the process spread is within limits; bigger is better, Cpu is whether or not the process is getting near the upper limit, Cpl is whether or not the process is getting near the lower limit and Cpk combines Cpu and Cpl and states the actual capability of the process. If Cpk is high enough, the process is within the VOC (customer delivery expectation of 0–32 hours). (1factory, 2024)

The code took the first 1000 deliveries into account, and is given in Figure 12, with sd_pt being the standard deviation of the first 1000 delivery times. The results are seen in Figure 13. The capability indices of the delivery times for each product type were computed using the first 1000 deliveries for each product, with a lower specification limit (LSL) = 0 hours and an upper specification limit (USL) = 32 hours. The values of Cp, which indicate the process capability if it were centered, range from 0.89 to 18.14, while the values of Cpk, which are both related to spread as well as centring relative to the specification limits, range from 0.70 to 1.08. A value of 1.0 or greater for the Cpk is often considered just capable of consistently meeting the Voice of the Customer (VOC). (1factory, 2024)

Only the SOF product type, with a Cpk of 1.08, is capable of reaching the VOC. The MOU, KEY, CLO, LAP, and MON product types, all have Cpk readings below 1.0, indicating their processes are not yet at a stage where they can reliably deliver as expected.

Figure 11:

$Cp = (USL-LSL)/(6\sigma)$	$Cpu = (USL - \mu)/(3\sigma)$
$Cpl = (\mu - LSL)/(3\sigma)$	$Cpk = \min(Cpl, Cpu)$

Figure 12:

```
data_pt <- head(sales$deliveryHours[sales$ProductType == pt], 1000)

mean_pt <- mean(data_pt)
sd_pt <- sd(data_pt)

Cp <- (USL - LSL) / (6 * sd_pt)
Cpu <- (USL - mean_pt) / (3 * sd_pt)
Cpl <- (mean_pt - LSL) / (3 * sd_pt)
Cpk <- min(Cpu, Cpl)
```

Figure 13:

ProductType	Cp	Cpu	Cpl	Cpk
MOU	0.92	0.73	1.10	0.73
KEY	0.92	0.73	1.10	0.73
SOF	18.14	35.19	1.08	1.08
CLO	0.90	0.72	1.08	0.72
LAP	0.90	0.70	1.10	0.70
MON	0.89	0.70	1.08	0.70

3.4 In order to calculate the sigma control limits, the UCL and LCL were used from previous section. The output produced in R is depicted in Figure 14.

Figure 14:

```
Product Type: MOU
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 32
C. 4+ consecutive X̄ above +2σ: first3=194, 195, 196  last3=858, 859, 860  total=360

Product Type: KEY
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 32
C. 4+ consecutive X̄ above +2σ: first3=99, 100, 101  last3=744, 745, 746  total=321

Product Type: SOF
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 26
C. 4+ consecutive X̄ above +2σ: first3=133, 134, 135  last3=862, 863, 864  total=359

Product Type: CLO
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 36
C. 4+ consecutive X̄ above +2σ: first3=122, 123, 124  last3=647, 648, 649  total=280

Product Type: LAP
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 24
C. 4+ consecutive X̄ above +2σ: first3=119, 120, 121  last3=423, 424, 425  total=172

Product Type: MON
A. s outside ±3σ: first3=  last3=  total=0
B. Longest run within ±1σ (s): 47
C. 4+ consecutive X̄ above +2σ: first3=134, 135, 136  last3=617, 618, 619  total=259
```

a) For part A, considering the standard deviation charts of all product categories, there were no samples where the s value exceeded the ±3 sigma-control limits. This indicates that the deliveries in question, the variation in the samples was well within the expected control limits. Therefore, there were no such extreme deviations in the spread of the delivery times that would represent any immediate process instability, and there were no first and last three samples to report since the total number of out-of-control s samples was zero for all product types.

b) Part B involves finding the longest run of consecutive samples for which the standard deviation was inside the ±1 sigma-control limits. The analysis shows that these runs varied by product type, with both the MOU and KEY product types each recording a longest run of 32 samples, SOF a run of 26 samples, CLO a run of 36 samples, LAP 24 samples, and MON the longest run observed at 47 samples. These results show that the MON product line experienced the most stable variation within the ±1 sigma limits, with LAP recording comparatively shorter durations of highly consistent sample deviation.

c) For part C, which examines four or more consecutive X-bar samples exceeding the upper second-control limits (+2 sigma), all product types exhibited a number of such

occurrences. The MOU product type accounted for 360 samples under this condition, with the first three at samples 194, 195, and 196, and the last three at 858, 859, and 860. KEY accounted for 321 samples, with the first three at 99, 100, and 101, and the last three at 744, 745, and 746. SOF accounted for 359 samples with the first three at 133, 134, and 135, and the last three at 862, 863, and 864. CLO accounted for 280 samples, LAP 172 samples, and MON 259 samples meeting the same condition, with first and last three samples specified for each. This indicates that there were substantial runs of high means for all product types, suggesting repeated periods where the mean delivery times were consecutively higher than the upper control limits, indicating potential shifts in the process that could be worthy of investigation.

Question 4: Risk, Data correction

4.1 A script was developed in R to estimate the likelihood of making a Type I (Manufacturer's) Error for A, B and C from the previous week's work. The following implemented calculations are provided in Figure 15a. The results can be seen in Figure 15b.

Figure 15a:

```
#A
alpha_A <- 2 * (1 - pnorm(3))

#B
p_within1 <- pnorm(1) - pnorm(-1)
m <- 7
alpha_B <- p_within1^m

#C
p_gt2 <- 1 - pnorm(2)
alpha_C <- p_gt2^4

type1_errors <- data.frame(
  Rule = c("A: point outside ±3σ",
           "B: 7 consecutive within ±1σ",
           "C: 4 consecutive >+2σ"),
  Alpha = c(alpha_A, alpha_B, alpha_C)
)
```

Figure 15b:

```
                      Rule        Alpha
      A: point outside ±3σ 2.699796e-03
B: 7 consecutive within ±1σ 6.911344e-02
      C: 4 consecutive >+2σ 2.678772e-07
```

4.2 As given by the question, Cl = 25.05, UCL = 25.089, LCL = 25.011, μ = 25.028 and σx = 0.017. Due to standard SPC, it is a given that normal distribution can be used. Thus, the following formula that can be used can be seen in Figure 16, and the calculations are shown in Figure 17. The calculations in Figure 17 result in a type II error = 0.841178

Figure 16:

$$P\left(\frac{LCL-\mu}{\sigma_X} \leq Z \leq \frac{UCL-\mu}{\sigma_X}\right)$$

Figure 17:

```
CL  <- 25.05
UCL <- 25.089
LCL <- 25.011

u <- 25.028
sigma_xbar <- 0.017

z_upper <- (UCL - u) / sigma_xbar
z_lower <- (LCL - u) / sigma_xbar

beta  <- pnorm(z_upper) - pnorm(z_lower)
p <- 1 - beta

cat("z_lower =", round(z_lower,6), "\n")
cat("z_upper =", round(z_upper,6), "\n")
cat("Type II error (beta) =", round(beta,6), "\n")
cat("p (1 - beta) =", round(p,6), "\n")
```

4.3 The R code in Figure 18 systematically corrects the data quality issues in both files by implementing the specified business rules. For the head office data, the code first identifies and rectifies the incorrect "NA" prefixes in ProductIDs for items 11-60 by mapping each product's category to its appropriate prefix. The pricing errors are then addressed by extracting the correct 10-price patterns from the local products_data file and applying them in a repeating sequence to items 11-60 using modulo arithmetic, ensuring items 1, 11, 21 share the same price as item 1 of their type. For the local products_data file, the code enforces consistency between ProductID prefixes and their corresponding categories by deriving the proper category from each ID's prefix. The solution outputs two corrected CSV files while including verification checks to validate the corrections and providing a framework for subsequent sales analysis using the rectified data.

There are several differences between version 1 and version 2, both in the product data and in the head-office data, as can be seen by comparing Figure 19 and Figure 20.

For the branch (product) data, there is only one change: in Figure 19, the *Category* column contained different product types such as “Cloud Subscription,” “Laptop,” “Monitor,” and others, but in Figure 20, all entries under *Category* were updated to “Software.” The other columns — *ProductID*, *Description*, *SellingPrice*, *Markup*, and the overall numeric summaries — remain identical between the two versions. This indicates that the only correction applied to the branch data was aligning the *Category* values with the *ProductID* prefix “SOF,” as requested in the instructions.

For the head-office data, there are both structural and statistical differences. The first three rows in both versions (Figure 19 and Figure 20) appear identical, but the numerical summaries reveal a substantial change in the *SellingPrice* distribution. In Figure 19, the mean *SellingPrice* was 4410.96 with a standard deviation of 6463.82, a median of

797.22, and a minimum and maximum of 290.52 and 22420.14, respectively. In Figure 20, the mean dropped to 3827.28, the standard deviation decreased to 6081.49, and the median fell to 617.66, with a slightly higher minimum (350.45) and a lower maximum (19725.18). These differences show that version 2's (Figure 20's) updated head-office dataset contained generally lower and less variable selling prices. The *Markup* column shows only minor statistical differences (mean 20.39 → 20.36, sd 5.67 → 5.95).

In summary, version 2 in Figure 20 corrected the *Category* field in the branch data and adjusted the *SellingPrice* values in the head-office data, resulting in lower average prices, reduced variation, and slightly altered distribution metrics compared with version 1 in Figure 19, while all other aspects — the number of rows, columns, and data completeness — remained unchanged.

Figure 18:

```
library(dplyr)
library(readr)
library(stringr)

products_headoffice <- read.csv("H:/14. Quality Assurance/Project/Part 1/products_Headoffice.csv", stringsAsFactors = FALSE)
products_data <- read.csv("H:/14. Quality Assurance/Project/Part 1/products_data.csv", stringsAsFactors = FALSE)

cat("Original head office data dimensions:", dim(products_headoffice), "\n")
cat("Original local data dimensions:", dim(products_data), "\n")

products_headoffice2025 <- products_headoffice

get_category_prefix <- function(product_id) {
  prefix <- str_extract(product_id, "^[A-Za-z]+")
  return(prefix)
}

get_product_number <- function(product_id) {
  number <- as.numeric(str_extract(product_id, "\\d+"))
  return(number)
}

for(i in 11:60) {
  if(i <= nrow(products_headoffice2025)) {
    current_category <- products_headoffice2025$Category[i]

    category_prefix <- case_when(
      current_category == "Software" ~ "SOF",
      current_category == "Cloud Subscription" ~ "CLO",
      current_category == "Laptop" ~ "LAP",
      current_category == "Monitor" ~ "MON",
      current_category == "Keyboard" ~ "KEY",
      current_category == "Mouse" ~ "MOU",
      TRUE ~ "UNK" # Unknown category fallback
    )
    product_number <- str_pad(i, 3, pad = "0")
    products_headoffice2025$ProductID[i] <- paste0(category_prefix, product_number)
  }
}

correct_prices <- products_data %>%
  group_by(Category) %>%
  mutate(position = row_number()) %>%
  select(Category, position, Correct_SellingPrice = SellingPrice, Correct_Markup = Markup)

get_position <- function(product_number) {
  # Handle the modulo logic: 10 -> position 10, not position 0
  pos <- (product_number - 1) %% 10 + 1
  return(pos)
}
```

```
products_headoffice2025 <- products_headoffice2025 %>%
  mutate(product_number = get_product_number(ProductID),
         position = get_position(product_number)) %>%
  left_join(correct_prices, by = c("Category", "position")) %>%
  mutate(
    SellingPrice = ifelse(product_number >= 11, Correct_SellingPrice, SellingPrice),
    Markup = ifelse(product_number >= 11, Correct_Markup, Markup)
  ) %>%
  select(-Correct_SellingPrice, -Correct_Markup, -product_number, -position)

write.csv(products_headoffice2025, "H:/14. Quality Assurance/Project/Part 3/products_Headoffice2025.csv", row.names = FALSE)

products_data2025 <- products_data

category_mapping <- function(product_id) {
  prefix <- get_category_prefix(product_id)
  category <- case_when(
    prefix == "SOF" ~ "Software",
    prefix == "CLO" ~ "Cloud Subscription",
    prefix == "LAP" ~ "Laptop",
    prefix == "MON" ~ "Monitor",
    prefix == "KEY" ~ "Keyboard",
    prefix == "MOU" ~ "Mouse",
    TRUE ~ "Unknown"
  )
  return(category)
}
products_data2025 <- products_data2025 %>%
  mutate(Category = category_mapping(ProductID))

write.csv(products_data2025, "H:/14. Quality Assurance/Project/Part 3/products_data2025.csv", row.names = FALSE)
```

Figure 19: Previous Results from Week 1

Products – Branch

```
> summary_stats(products, "Products (branch)")
----- Products (branch) -----
Rows: 60 Cols: 5
  ProductID           Category      Description SellingPrice Markup
1    SOF001           Software       coral matt       511.53  25.05
2    SOF002 Cloud Subscription        cyan silk       505.26  10.43
3    SOF003             Laptop burlywood marble       493.69  16.18
Missing per column:
   ProductID     Category  Description SellingPrice       Markup
           0            0            0            0            0
Numeric describe (if any):
             vars  n    mean      sd median trimmed    mad    min      max    range  skew kurtosis     se
SellingPrice    1 60 4493.59 6503.77 794.18 3189.25 525.72 350.45 19725.18 19374.73  1.43     0.43 839.63
Markup          2 60   20.46    6.07  20.34   20.51   7.31  10.13    29.84    19.71 -0.04    -1.24   0.78
```

Products – Head Office

```
> summary_stats(ho_products, "Products (head office)")
----- Products (head office) -----
Rows: 360 Cols: 5
  ProductID Category      Description SellingPrice Markup
1    SOF001 Software       coral silk       521.72  15.65
2    SOF002 Software       black silk       466.95  28.42
3    SOF003 Software burlywood marble       496.43  20.07
Missing per column:
   ProductID     Category  Description SellingPrice       Markup
           0            0            0            0            0
Numeric describe (if any):
             vars   n    mean      sd median trimmed    mad    min      max    range  skew kurtosis     se
SellingPrice    1 360 4410.96 6463.82 797.22 3054.23 515.75 290.52 22420.14 22129.62  1.53     0.78 340.67
Markup          2 360   20.39    5.67  20.58   20.43   6.66  10.06    30.00    19.94 -0.05    -1.07   0.30
```

Figure 20: New Results with corrected File Data

Products – Branch

```
> summary_stats(products, "Products (branch)")
----- Products (branch) -----
Rows: 60 Cols: 5
  ProductID Category      Description SellingPrice Markup
1    SOF001 Software       coral matt       511.53  25.05
2    SOF002 Software        cyan silk       505.26  10.43
3    SOF003 Software burlywood marble       493.69  16.18
Missing per column:
   ProductID     Category  Description SellingPrice       Markup
           0            0            0            0            0
Numeric describe (if any):
             vars  n    mean      sd median trimmed    mad    min      max    range  skew kurtosis     se
SellingPrice    1 60 4493.59 6503.77 794.18 3189.25 525.72 350.45 19725.18 19374.73  1.43     0.43 839.63
Markup          2 60   20.46    6.07  20.34   20.51   7.31  10.13    29.84    19.71 -0.04    -1.24   0.78
```

Products – Head Office

```
> summary_stats(ho_products, "Products (head office)")
----- Products (head office) -----
Rows: 360 Cols: 5
  ProductID Category      Description SellingPrice Markup
1    SOF001 Software       coral silk       521.72  15.65
2    SOF002 Software       black silk       466.95  28.42
3    SOF003 Software burlywood marble       496.43  20.07
Missing per column:
   ProductID     Category  Description SellingPrice       Markup
           0            0            0            0            0
Numeric describe (if any):
             vars   n    mean      sd median trimmed    mad    min      max    range  skew kurtosis     se
SellingPrice    1 360 3827.28 6081.49 617.66 2395.56 327.57 350.45 19725.18 19374.73  1.74     1.46 320.52
Markup          2 360   20.36    5.95  20.23   20.39   7.15  10.09    29.94    19.85 -0.01    -1.15   0.31
```

In addition, more analysis can take place for the data such as providing a summary of the total sales value of 2023 per type as well as providing a graph of total sales value of 2023 per type. Figures 21–24 collectively show the variance of the original dataset vs the modified data. From Figure 21 and as graphically illustrated in Figure 22, the distribution of revenue across product categories is quite even. Every category—Monitor, Laptop, Mouse, Keyboard, Software, and Cloud Subscription—is contributing proportionally to the overall revenue (from approximately R282 million to R384 million), with percentages between 13.9% to 18.9%. This degree of evenness is a sign of possessing a diversified product mix, where both the total amounts (around 100,000 units each) and unit revenue averages experience mediocre volatility.

However, Figure 23 and Figure 24 indicate a much denser pattern. Laptops dominate total revenue at 57.3% (≈R1.16 billion), while Monitors account for 28.5% (≈R578 million) next. The remaining four categories—Cloud Subscription, Keyboard, Software, and Mouse—cumulatively only account for about 14% of total revenue. The shift is also highlighted by the dramatic difference of average revenue per unit.

Overall, Figures 21–22 is an even and balanced product performance, as would be expected from stable pricing and volume sales in all categories. Figures 23–24 is, by contrast, characterized by intense revenue concentration in the sale of more valuable fewer Laptops and Monitors, with poorer sales of peripherals, software, and cloud products. This is indicative of a strategic or market move—perhaps with high-value corporate contracts or premium hardware sales—rather than growth or decline in the market overall.

Figure 21: Summary of Total Sales Value of 2023 Per Type (Unchanged Prices)

Category	Total_Revenue_2023	Total_Qty_2023
<chr>	<dbl>	<int>
Monitor	383655216.	107366
Laptop	375703806.	101279
Mouse	339828477.	104439
Keyboard	339498320.	104565
Software	311112488.	105072
Cloud Subscription	282379354.	105485

Figure 22: Graph of Total Sales Value of 2023 Per Type (Unchanged Prices)

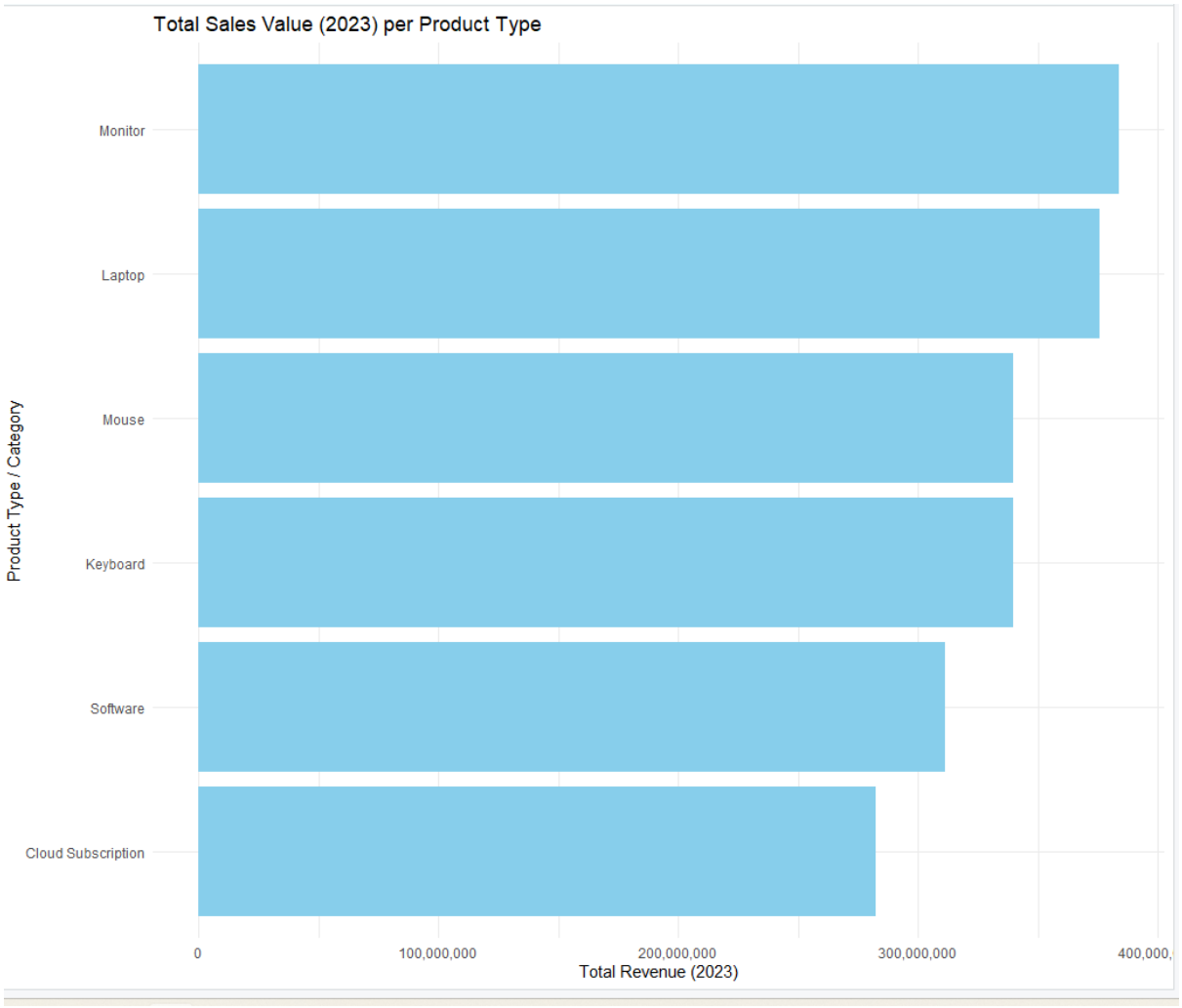
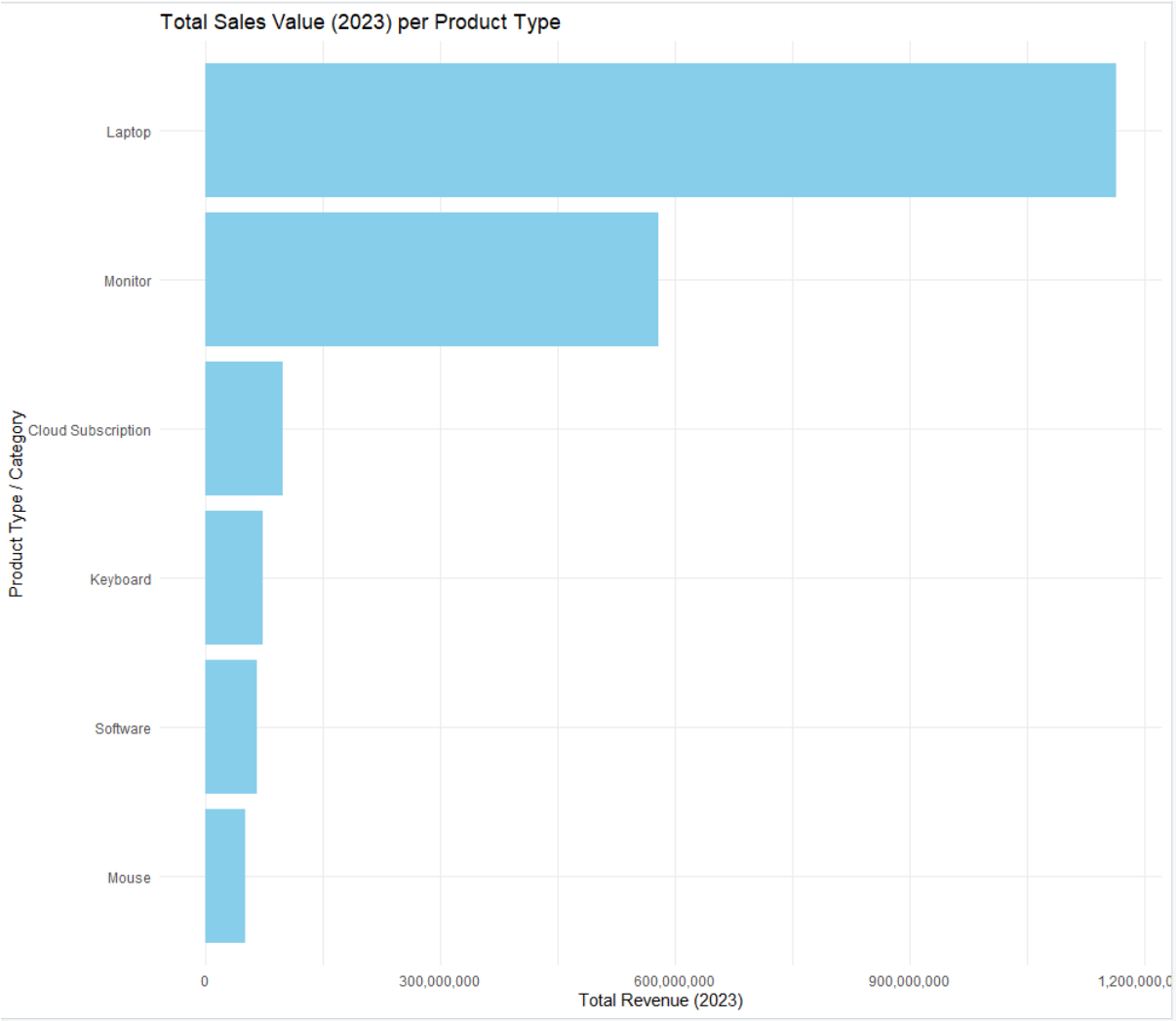


Figure 23: Summary of Total Sales Value of 2023 Per Type (Updated Prices)

Category	Total_Revenue_2023	Total_Qty_2023
<chr>	<dbl>	<int>
Laptop	1163889479.	64414
Monitor	578385570.	91782
Cloud Subscription	98715482.	96691
Keyboard	73499067.	114357
Software	66468485.	131349
Mouse	51219577.	129613

Figure 24: Graph of Total Sales Value of 2023 Per Type (Updated Prices)



Question 5: Optimizing Profit for timeToServe.csv dataset

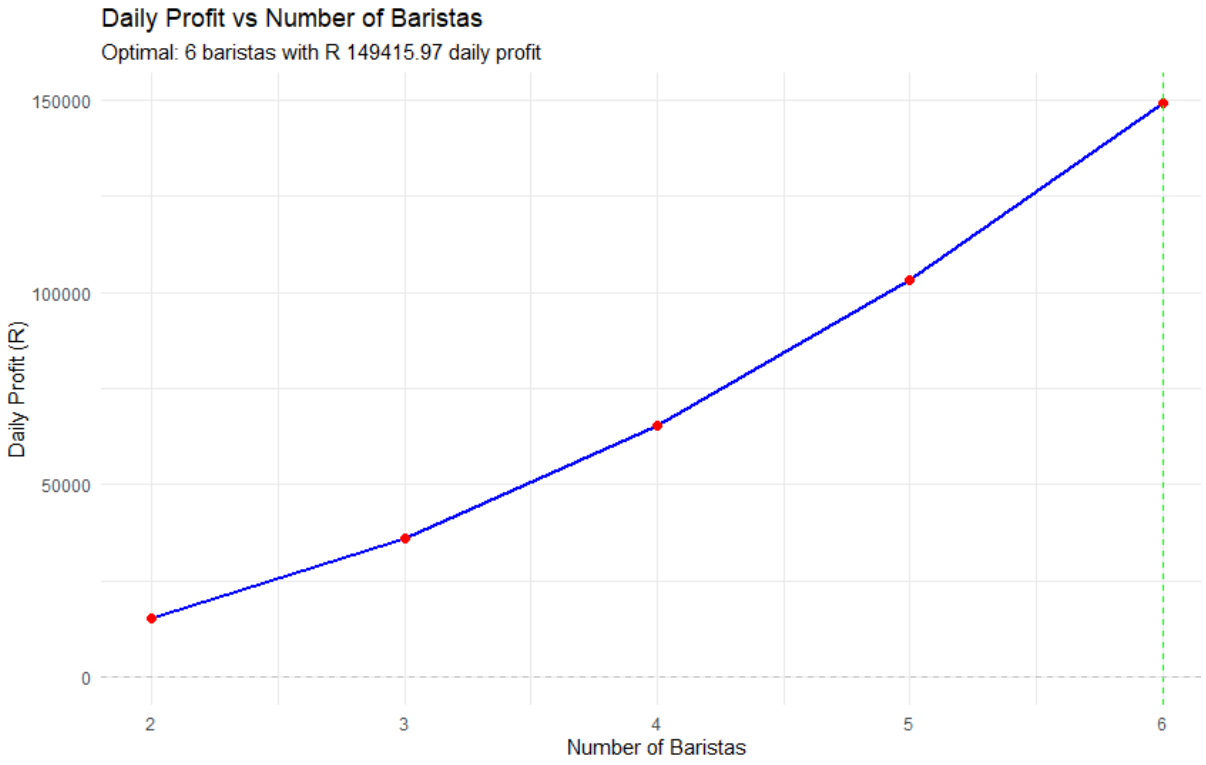
5. The following profit model was built, where Profit = Revenue - Personnel cost, Revenue = 30 * number of customers served, Personnel cost = 1000 * number of baristas. In addition, in order to estimate what percentage of clients should expect reliable service, a threshold value was needed to determine the longest service time that would be tolerated by customers. Through research, a value of 354 seconds was selected (coffeepreneur, 2017). This value could then be used to calculate the percentage of reliable service. Every instance recorded has a service time < 354, which implies a 100% service reliability. A snippet of the R script used to create the profit model is provided in Figure 25, and the output can be seen in Figure 26. Based on the output in Figure 26, there is a general trend between number of Baristas and the profit. It is evident that the optimal solution, i.e., the number of baristas that will maximize the daily profit is 6, contributing to a maximum profit of R 149,416.

Figure 25:

```
library(dplyr)
library(ggplot2)

data <- read.csv("H:/14. Quality Assurance/Project/Part 3/timeToServe.csv")

head(data)
summary(data)

stats_by_baristas <- data %>%
  group_by(V1) %>%
  summarise(
    count = n(),
    mean_service_time = mean(V2),
    median_service_time = median(V2),
    sd_service_time = sd(V2),
    min_service_time = min(V2),
    max_service_time = max(V2)
  )

print(stats_by_baristas)

profit_per_customer <- 30
daily_personnel_cost <- 1000
min_baristas <- 2
work_hours <- 8

customers_per_day <- stats_by_baristas %>%
  mutate(
    seconds_per_customer = mean_service_time,
    customers_per_hour_per_barista = 3600 / seconds_per_customer,
    total_customers_per_hour = customers_per_hour_per_barista * V1,
    total_customers_per_day = total_customers_per_hour * work_hours
  )

profit_analysis <- customers_per_day %>%
  filter(V1 >= min_baristas) %>%  # min barista constraint
  mutate(
    revenue = profit_per_customer * total_customers_per_day,
    personnel_cost = daily_personnel_cost * V1,
    daily_profit = revenue - personnel_cost
  )

print(profit_analysis[, c("V1", "total_customers_per_day", "revenue", "personnel_cost", "daily_profit")])

# Find optimal number of baristas
optimal_baristas <- profit_analysis %>%
  filter(daily_profit == max(daily_profit))
```

Figure 26:

V1	total_customers_per_day	revenue	personnel_cost	daily_profit
<int>	<dbl>	<dbl>	<dbl>	<dbl>
2	575.	17251.	2000	15251.
3	1297.	38912.	3000	35912.
4	2305.	69147.	4000	65147.
5	3603.	108103.	5000	103103.
6	5181.	155416.	6000	149416.



Question 6: DOE and MANOVA or ANOVA.

6.1 - 6.2 The selected datasets for this section are (1) products_data2025 and (2) products_Headoffice2025. The hypotheses were selected as follows:

Null hypothesis – No significant difference in mean Selling Price and Markup between dataset 1 and dataset 2.

Alternative hypothesis – Significant difference in at least one of the dependent variables (Selling Price or Markup) between the two datasets.

A MANOVA was set up and produced in R using built-in functions. The code can be seen in Figure 27, and the MANOVA results can be seen in Figure 28, with graphical outputs displayed in Figure 29. In order to interpret the results, a thorough understanding of p values was required, which was studied in the following source: *Mcleod, 2023*.

The MANOVA results in Figure 28 indicate that the impact of the data source (Local vs Head Office) on the combined dependent variables, Selling Price and Markup, varies across product categories. For Software, the test is significant ($p = 0.0023$), suggesting a multivariate difference between sources; however, follow-up ANOVAs show that this difference is primarily driven by Markup ($p = 0.0404$), while Selling Price does not differ significantly ($p = 0.1213$). In the Cloud Subscription category, neither MANOVA ($p = 0.3619$) nor the individual ANOVAs indicate significant differences, implying that Local and Head Office data are comparable for both Selling Price and Markup. For Laptops, MANOVA is highly significant ($p < 0.001$), with the follow-up ANOVA revealing that the difference is almost entirely due to Selling Price ($p < 0.001$), whereas Markup shows no significant variation ($p = 0.273$). The Monitor category shows no significant multivariate difference ($p = 0.1414$), though Markup approaches marginal significance ($p = 0.079$), suggesting a potential trend. For Keyboards, MANOVA is marginally significant ($p = 0.0598$), and follow-up ANOVAs indicate that Markup differs significantly ($p = 0.0423$), whereas Selling Price does not ($p = 0.1105$). Finally, in the Mouse category, neither MANOVA ($p = 0.222$) nor the individual ANOVAs show significant differences, indicating consistent data between sources.

In conclusion, the differences between Local Office and Head Office information are more evident with respect to Laptops and certain points related to Software and Keyboard offerings, specifically with reference to Selling Price of Laptops and Markup of Software and Keyboard offerings. Differences between the sources of information are found to be statistically insignificant for other categories of products including Cloud Subscription, Monitors, and Mice. This points to differences that may exist with respect to certain categories of products due to pricing differences.

When observing the two boxplot graphs in Figure 29, the following conclusions are relevant. The MANOVA result and box plots provide insight into variations in selling price

and markup between HeadOffice and Local sources across various product types. While comparing subscriptions on the cloud, it can be seen that there is an almost negligible difference in both selling prices and overall markup for both sources, thus supporting the non-significant MANOVA result. For keyboards, it is clear from both box plots and MANOVA data that the Local store holds a slightly higher markup than its HeadOffice counterpart, despite both sources selling the item at the same price, thus supporting the significant markup ANOVA data. Laptops show marked variations in terms of selling price across sources, with HeadOffice charging more than Local sources. However, there seems to be a similar markup on both sources, thus suggesting that this could be the result of differing policies. For monitors, it can be seen from box plots that the Local sources show a more enhanced markup than HeadOffice sources, though it was not significant. Mice show complete overlap in both selling price and overall markup across both sources, thus supporting an insignificant MANOVA result. Lastly, software shows that there is a marked variation in overall markup on both sources despite equal selling prices, thus suggesting that it could be a centralized decision on HeadOffice sources. Also, it shows a significant variation, thus suggesting that HeadOffice sources use more markup on software than Local sources.

Figure 27:

```r
# 1. Load data
products_data2025 <- read.csv("H:/14. Quality Assurance/Project/Part 3/products_data2025.csv")
products_Headoffice2025 <- read.csv("H:/14. Quality Assurance/Project/Part 3/products_Headoffice2025.csv")

# 2. Add a Source column to each dataset
products_data2025$Source <- "Local"
products_Headoffice2025$Source <- "HeadOffice"

# 3. Combine both datasets
all_data <- rbind(products_data2025, products_Headoffice2025)

# 4. Ensure relevant columns exist
# (Optional check)
if(!all(c("Category", "SellingPrice", "Markup", "Source") %in% names(all_data))) {
  stop("Required columns (Category, SellingPrice, Markup, Source) are missing.")
}

# 5. Get all unique categories
categories <- unique(all_data$Category)

# 6. Run MANOVA for each product category
manova_results <- list()

for(cat in categories) {
  cat_data <- subset(all_data, Category == cat)

  # Only run if both sources exist in this category
  if(length(unique(cat_data$Source)) == 2) {
    model <- manova(cbind(SellingPrice, Markup) ~ Source, data = cat_data)
    result <- summary(model, test = "Wilks")

    cat("\n----------------------------------------\n")
    cat("MANOVA Results for Category:", cat, "\n")
    print(result)

    cat("\nFollow-up ANOVAs for each dependent variable:\n")
    print(summary.aov(model))

    # Save results in a list
    manova_results[[cat]] <- list(
      MANOVA = result,
      ANOVA = summary.aov(model)
    )
  } else {
    cat("\nSkipping category:", cat, "(Only one data source present)\n")
  }
}
```

Figure 28:

Software:

```
MANOVA Results for Category: Software
          Df  Wilks approx F num Df den Df   Pr(>F)
Source     1 0.8342   6.6584      2     67 0.002304 **
Residuals 68
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq  Mean Sq F value Pr(>F)
Source       1   64864894 64864894  2.4619 0.1213
Residuals   68 1791640252 26347651

 Response Markup :
            Df  Sum Sq Mean Sq F value Pr(>F)
Source       1  129.15 129.149  4.3664 0.0404 *
Residuals   68 2011.28  29.578
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Cloud Subscription:

```
MANOVA Results for Category: Cloud Subscription
          Df   Wilks approx F num Df den Df Pr(>F)
Source     1 0.97011    1.032      2     67 0.3619
Residuals 68

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq  Mean Sq F value Pr(>F)
Source       1   39344723 39344723  1.6673  0.201
Residuals   68 1604696963 23598485

 Response Markup :
            Df  Sum Sq Mean Sq F value Pr(>F)
Source       1    3.68   3.681  0.0996 0.7533
Residuals   68 2512.99  36.956
```

Laptop:

```
MANOVA Results for Category: Laptop
          Df   Wilks approx F num Df den Df    Pr(>F)
Source     1 0.59972    22.36      2     67 3.641e-08 ***
Residuals 68
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq    Mean Sq F value    Pr(>F)
Source       1 1598841616 1598841616  41.633 1.355e-08 ***
Residuals   68 2611426470   38403330
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

 Response Markup :
            Df  Sum Sq Mean Sq F value Pr(>F)
Source       1   46.69  46.687  1.2235 0.2726
Residuals   68 2594.87  38.160
```

Monitor:

```
MANOVA Results for Category: Monitor
          Df   Wilks approx F num Df den Df Pr(>F)
Source     1 0.94327   2.0147      2     67 0.1414
Residuals 68

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq  Mean Sq F value Pr(>F)
Source       1   35937018 35937018  1.0326 0.3131
Residuals   68 2366474480 34801095

 Response Markup :
            Df  Sum Sq Mean Sq F value  Pr(>F)
Source       1  116.12 116.120  3.1811 0.07896 .
Residuals   68 2482.24  36.503
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

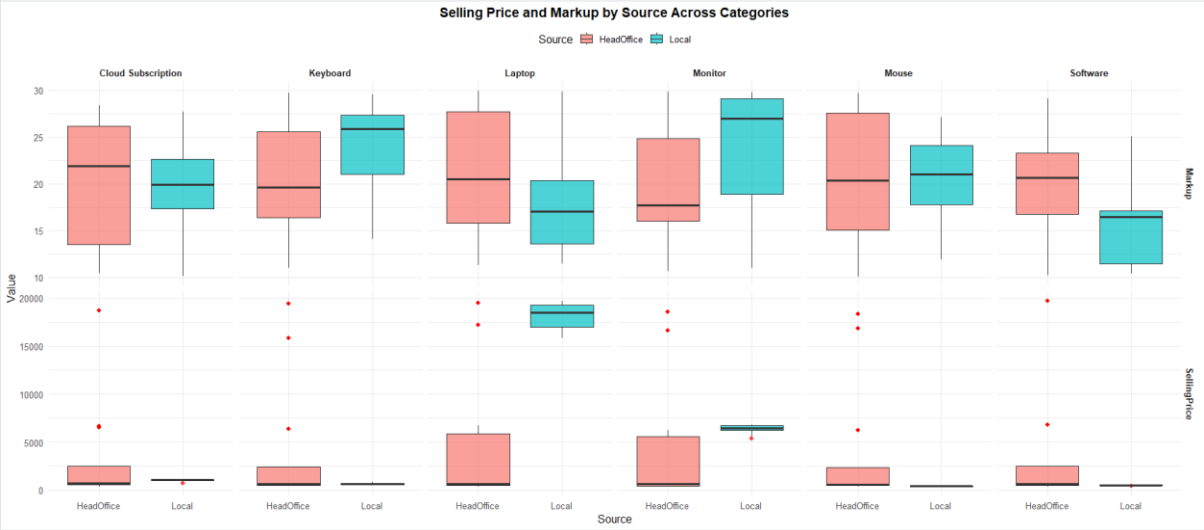
Keyboard:

```
MANOVA Results for Category: Keyboard
          Df   Wilks approx F num Df den Df  Pr(>F)
Source     1 0.91933   2.9395      2     67 0.05975 .
Residuals 68
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq  Mean Sq F value Pr(>F)
Source       1   93690595 93690595  2.6143 0.1105
Residuals   68 2436939495 35837346

 Response Markup :
            Df  Sum Sq Mean Sq F value Pr(>F)
Source       1  121.17  121.17  4.2831 0.0423 *
Residuals   68 1923.73   28.29
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Mouse:

```
MANOVA Results for Category: Mouse
          Df   Wilks approx F num Df den Df Pr(>F)
Source     1 0.95608   1.5389      2     67 0.2221
Residuals 68

Follow-up ANOVAs for each dependent variable:
 Response SellingPrice :
            Df     Sum Sq   Mean Sq F value  Pr(>F)
Source       1  105471759 105471759  2.9822 0.08872 .
Residuals   68 2404934699  35366687
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

 Response Markup :
            Df  Sum Sq Mean Sq F value Pr(>F)
Source       1    0.01   0.009   2e-04 0.9881
Residuals   68 2865.69  42.143
```

Figure 29:


Selling Price and Markup by Source Across Categories
Source HeadOffice Local
Cloud Subscription
Keyboard
Laptop
Monitor
Mouse
Software
Value
Markup
SellingPrice
Source
HeadOffice
Local

Question 7: Reliability of Service

7.1 To estimate how many days per year to expect reliable service, the following formula was used. The script in Figure 30 was used, which added the number of reliable days (days on which number of workers was greater than 15). This correlated to 96 and 270, which gives reliable days equal to 366, resulting in a probability of 366/397 and a reliable number of days of 366/397 * 365 = approximately 336 days.

7.2 The code used to solve this question is provided in Figure 31. Essentially, this code applies simple maths of probability and cost, whereby it calculates the number of staff members that can be most profitably allocated. The code begins by assuming the probability of a given member attending, denoted by (p), which is arrived at by dividing the average number of members actually attending by the total number of members scheduled. The code then applies the binomial distribution formula to determine how likely it is that less than 15 members will show up on a given day, which is a problem day. The 'expected yearly loss' is this probability multiplied by 365 days and R20,000. It then adds the 'cost of extra personnel' given by R25,000 multiplied by 12 months multiplied by the number of additional personnel. The code does all this by testing different values of 'total employees from 16–20 employees' by summing the two costs. It then returns the number of employees which yields the lowest yearly cost, which is most profitable. The optimal number of scheduled staff was calculated to be 17 with an expected annual cost of R 366219.8

Figure 30

```
# --- Data from the histogram (number of workers present) ---
workers <- 12:16
days <- c(1, 5, 25, 96, 270)
total_days <- sum(days)

# --- 7.1: Estimate how many days per year are reliable (>= 14 workers) ---
reliable_days <- sum(days[workers >= 15])
reliable_prob <- reliable_days / total_days
reliable_year <- 365 * reliable_prob
```

Figure 31

```r
# Parameters
problem_threshold <- 15      # less than 15 = problem
loss_per_day <- 20000        # R
hire_cost_month <- 25000     # R per month per person
hire_cost_year <- hire_cost_month * 12

# Estimate attendance probability (p) from data
n_current <- max(workers)                 # assume 16 scheduled
mean_present <- sum(workers * days) / total_days
p_est <- mean_present / n_current

cat("Estimated attendance probability p:", round(p_est, 4), "\n")

# Function to calculate expected annual cost for n scheduled staff
expected_cost <- function(n, p, threshold, loss_per_day, hire_cost_year, base_n = 16) {
  # Probability of a problem day (less than threshold workers present)
  q_prob <- pbinom(threshold - 1, n, p)

  # Expected annual loss due to problem days
  annual_loss <- 365 * q_prob * loss_per_day

  # Annual hiring cost for extra staff (beyond base level)
  extra_staff <- max(0, n - base_n)
  hiring_cost <- extra_staff * hire_cost_year

  total_cost <- annual_loss + hiring_cost

  return(data.frame(
    n = n,
    q_prob = q_prob,
    annual_loss = annual_loss,
    hiring_cost = hiring_cost,
    total_cost = total_cost
  ))
}

# Evaluate for a range of scheduled staff (16 to 20)
staff_range <- 16:20
results <- do.call(rbind, lapply(staff_range, expected_cost,
                                 p = p_est,
                                 threshold = problem_threshold,
                                 loss_per_day = loss_per_day,
                                 hire_cost_year = hire_cost_year))

# Print results
print(results)

# --- Find optimal number of staff (min total cost) ---
optimal <- results[which.min(results$total_cost), ]
cat("\nOptimal number of scheduled staff:", optimal$n,
    "with expected annual cost of R", round(optimal$total_cost, 2), "\n")
```

References

AI used for code: Deepseek; ChatGPT

- https://www.facebook.com/kenith.grey.1 (2019). *Control Chart Constants | Tables and Brief Explanation | R-BAR*. [online] R-BAR. Available at: https://r-bar.net/control-chart-constants-tables-explanations/.
- Www.sfu.ca. (2025). *C4 Function*. [online] Available at: https://www.sfu.ca/sasdoc/sashtml/qc/chapc/sect7.htm? [Accessed 7 Oct. 2025].
- 1factory (2024). *A Guide to Process Capability (Cp, Cpk) and Process Performance (Pp, Ppk) | 1factory*. [online] 1factory.com. Available at: https://www.1factory.com/quality-academy/guide-process-capability.html.
- coffeepreneur (2017). *Why the wait - why a queue in your coffee shop can be a bad thing*. [online] cafesuccesshub.com. Available at: https://www.cafesuccesshub.com/why-the-wait-queue/ [Accessed 23 Oct. 2025]
- Mcleod, S. (2023). *P-value and statistical significance: What it is & why it matters*. [online] Simply Psychology. Available at: https://www.simplypsychology.org/p-value.html.