QUALITY ASSURANCE 344
ECSA REPORT

17 October 2022

23539321
KR Kritzinger

Plagiarism

Voorletters en van **Initials and surname**	KR Kritzinger
Studentenommer **Student number**	23539321
Modulenaam **Module name**	Quality assurance 344
Titel van opdrag **Title of assignment**	ECSA project

Afrikaans: Plagiaatverklaring

- Ek het die Stellenbosch Universiteit Plagiaatbeleid gelees en verstaan, sowel as die definisies van plagiaat en selfplagiaat soos beskryf in die Beleid [Plagiaat: Die gebruik van ander se idees of materiaal sonder erkenning, of die hergebruik van ’n persoon se eie voorheen geëvalueerde of gepubliseerde materiaal sonder erkenning of ’n aanduiding daarvan (selfplagiaat of tekshergebruik)].
- Ek verstaan ook dat direkte vertalings plagiaat is.
- Dienooreenkomstig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek verstaan dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.
- Ek verklaar dat die werk in hierdie skryfstuk vervat my eie oorspronklike werk is en dat ek dit nie vantevore (in die geheel of gedeeltelik) ingehandig het vir bepunting in hierdie module/werkstuk of ’n ander module/werkstuk nie.

English: Plagiarism declaration

- I have read and understand the Stellenbosch University Policy on Plagiarism and the definitions of plagiarism and self-plagiarism contained in the Policy [Plagiarism: The use of the ideas or material of others without acknowledgement, or the re-use of one’s own previously evaluated or published material without acknowledgement or indication thereof (self-plagiarism or text-recycling)].
- I also understand that direct translations are plagiarism.

- Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.

- I declare that the work contained in this assignment is my own work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

20/11/2022

Handtekening / Signature

Datum / Date

Table of content

Table of figures

Table of table

Introduction

The report is a descriptive analysis of an online company's sales data over the years 2021 to 2029. The sales data firstly needs to be formatted before it can be used. This entails removing all the rows of the data that has NA values.

The report is split into 6 parts. Part 1 is formatting the data for the other parts. In part 2 the data was analysed and compared to see if there are any correlations or valuable information that can be used. Part 3 is statistic process control where the data is evaluated within the control limits. Part 4 evaluates the out-of-control data and the probabilities of making a Type I or Type II error. In part 5 MANOVAs are created to see what impact certain features has on other features within the data. The probability of delivering products on time is evaluated at the end.

The last part of the report is the R code used to create the graphs and calculations in the Appendix.

Body

Part 1 & 2

In part 1 the dataset (salesTable2022) was split into two different datasets. These datasets where called “valid data”, which included all the data that had values, and “incomplete data”, which included all the values that had missing values.

t	X	ID	AGE	Class	Price	Year	Month	Day	Delivery.ti	Why.Bought
1	1	19966	54	Sweets	246.21	2021	7	3	1.5	Recommended
2	2	34006	36	Household	1708.21	2026	4	1	58.5	Website
3	3	62566	41	Gifts	4050.53	2027	8	10	15.5	Recommended
4	4	70731	48	Technolog	41843.21	2029	10	22	27	Recommended
5	5	92178	76	Household	19215.01	2027	11	26	61.5	Recommended
6	6	50586	78	Gifts	4929.82	2027	4	24	14.5	Random
7	7	73419	35	Luxury	108953.5	2029	11	13	4	Recommended
8	8	32624	58	Sweets	389.62	2025	7	2	2	Recommended
9	9	51401	82	Gifts	3312.11	2025	12	18	12	Recommended
10	10	96430	24	Sweets	176.52	2027	11	4	3	Recommended
11	11	87530	33	Technolog	8515.63	2026	7	15	21	Browsing
12	12	14607	64	Gifts	3538.66	2026	5	13	13.5	Recommended
13	13	24299	52	Technolog	27641.97	2024	5	29	17	Browsing
14	14	77795	92	Food	556.83	2025	6	3	3	Random
15	15	62567	73	Clothing	347.99	2024	3	29	8.5	Website
16	16	14839	47	Technolog	54650.41	2027	12	30	18.5	Recommended
17	17	96208	44	Technolog	14739.09	2028	3	17	13	Recommended
18	18	39674	69	Technolog	22315.17	2026	8	20	20.5	Recommended
19	19	98694	74	Sweets	546.48	2025	5	9	2	Recommended
20	20	99187	54	Luxury	81620.21	2027	9	14	3	Recommended
21	21	59365	72	Gifts	3314.76	2028	4	30	13	Recommended
22	22	37221	24	Sweets	220.91	2021	3	8	3	Recommended
23	23	78120	23	Gifts	2378.31	2023	3	10	12	Recommended
24	24	65860	30	Gifts	2440.41	2021	5	11	9.5	Recommended
25	25	70953	70	Gifts	3962.67	2024	10	6	12.5	Recommended
26	26	58327	45	Luxury	83248.5	2027	1	2	4.5	Recommended
27	27	39049	60	Luxury	26681.03	2029	6	18	2	Recommended

Table 1: Valid data

r	X	ID	AGE	Class	Price	Year	Month	Day	Delivery.time	Why.Bought
1	12345	18973	93	Gifts	NA	2026	6	11	15.5	Website
2	16321	81959	43	Technolog	NA	2029	9	6	22	Recommended
3	19541	71169	42	Technolog	NA	2025	1	19	20.5	Recommended
4	19999	67228	89	Gifts	NA	2026	2	4	15	Recommended
5	23456	88622	71	Food	NA	2027	4	18	2.5	Random
6	34567	18748	48	Clothing	NA	2021	4	9	8	Recommended
7	45678	89095	65	Sweets	NA	2029	11	6	2	Recommended
8	54321	62209	34	Clothing	NA	2021	3	24	9.5	Recommended
9	56789	63849	51	Gifts	NA	2024	5	3	10.5	Website
10	65432	51904	31	Gifts	NA	2027	7	24	14.5	Recommended
11	76543	79732	71	Food	NA	2028	9	24	2.5	Recommended
12	87654	40983	33	Food	NA	2024	8	27	2	Recommended
13	98765	64288	25	Clothing	NA	2021	1	24	8.5	Browsing
14	144444	70761	70	Food	NA	2027	9	28	2.5	Recommended
15	155555	33583	56	Gifts	NA	2022	12	9	10	Recommended
16	166666	60188	37	Technolog	NA	2024	10	9	21.5	Website
17	177777	68698	30	Food	NA	2023	8	14	2.5	Recommended

Table 2: incomplete data

In the graph belowe for the sales price of products it is cleear that the majority of products has a sales price of between -588.8 and 7000. The graph is exponenstial and the previous statement is comfirmed by the mean of 12293.7 which is a lot lower than the max value of 116619. The standard divialion (20888.97) is also high compared to the Q1 (482.3) and Q3 (15270.7) values also hughlighting the fact that most of the data (75%) falls within 13% of the range of the data. This could be because the company sells high folumes of lower priced products and smaller amounts of products with a higher price.

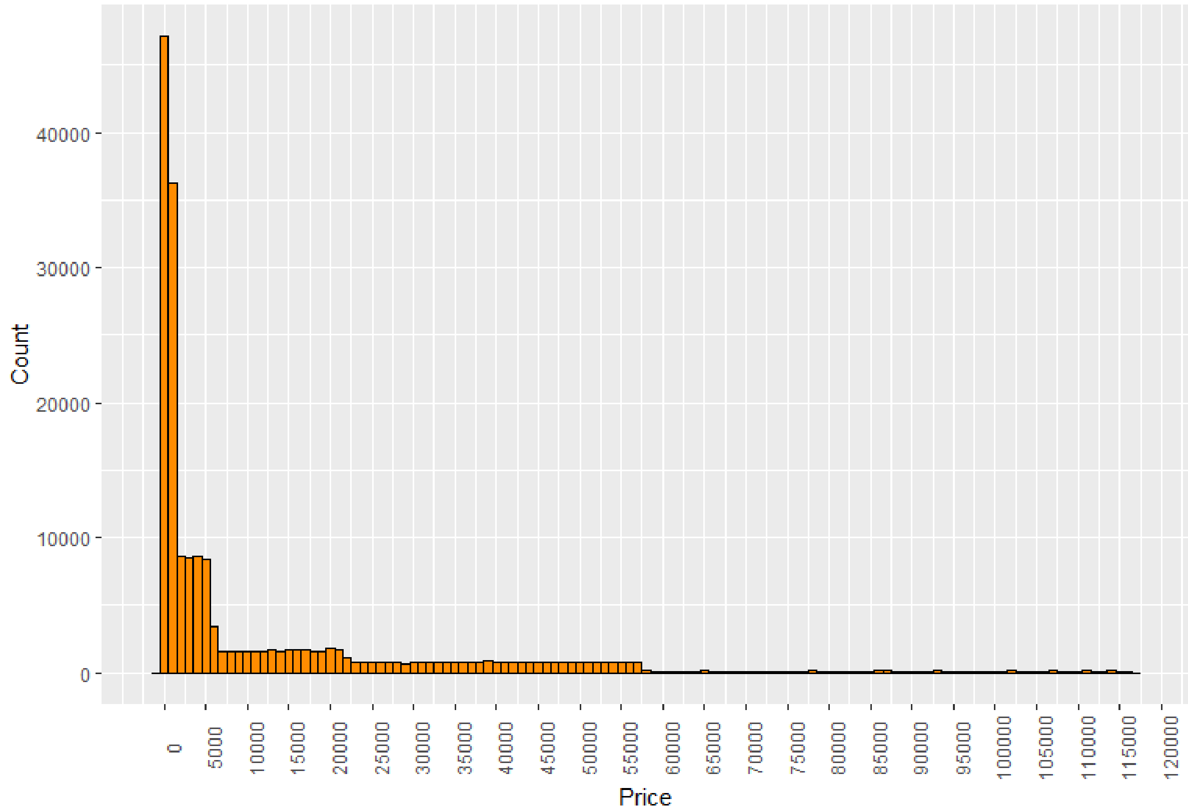


Figure 1: price of products

As seen in the graph of the delivery time of the valid data set the graph is multimodal, meaning that there are two peaks. This can be a result of the company selling more than one product, indicating that the products have different delivery times. Before any calculations where done we assumed USL=24 hours and LSL= 0.

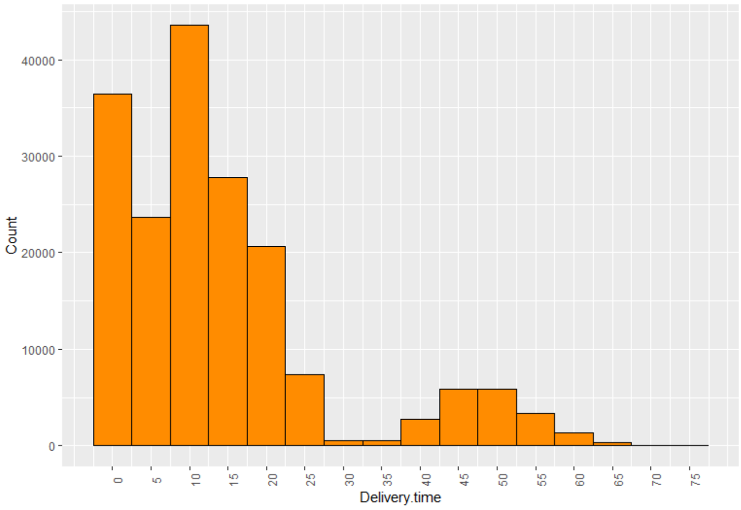


Figure 2: Delivery time of valid data

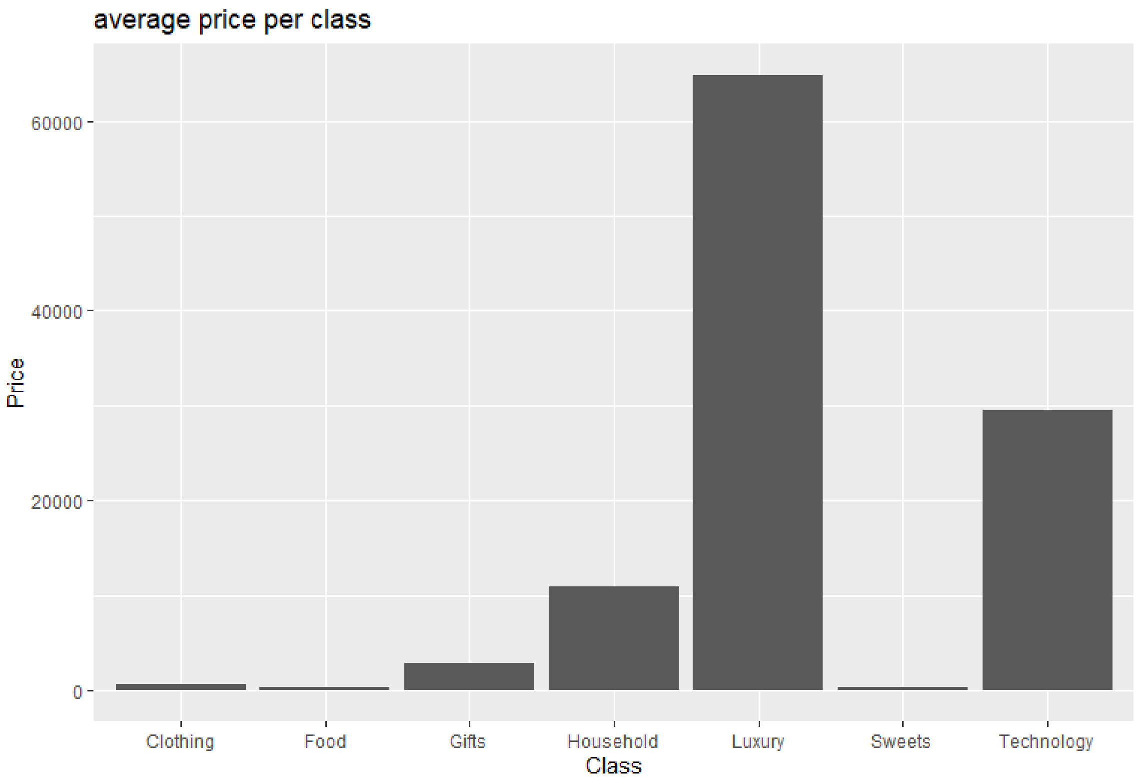


Figure 3: Average price per class

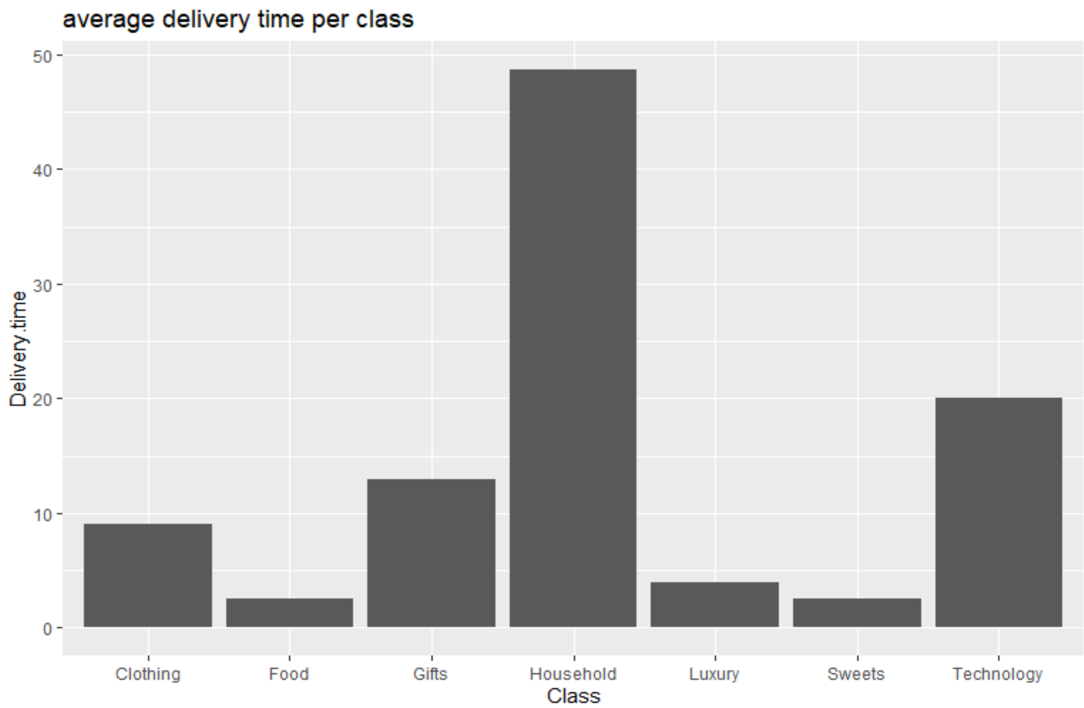


Figure 4: Average delivery time per class

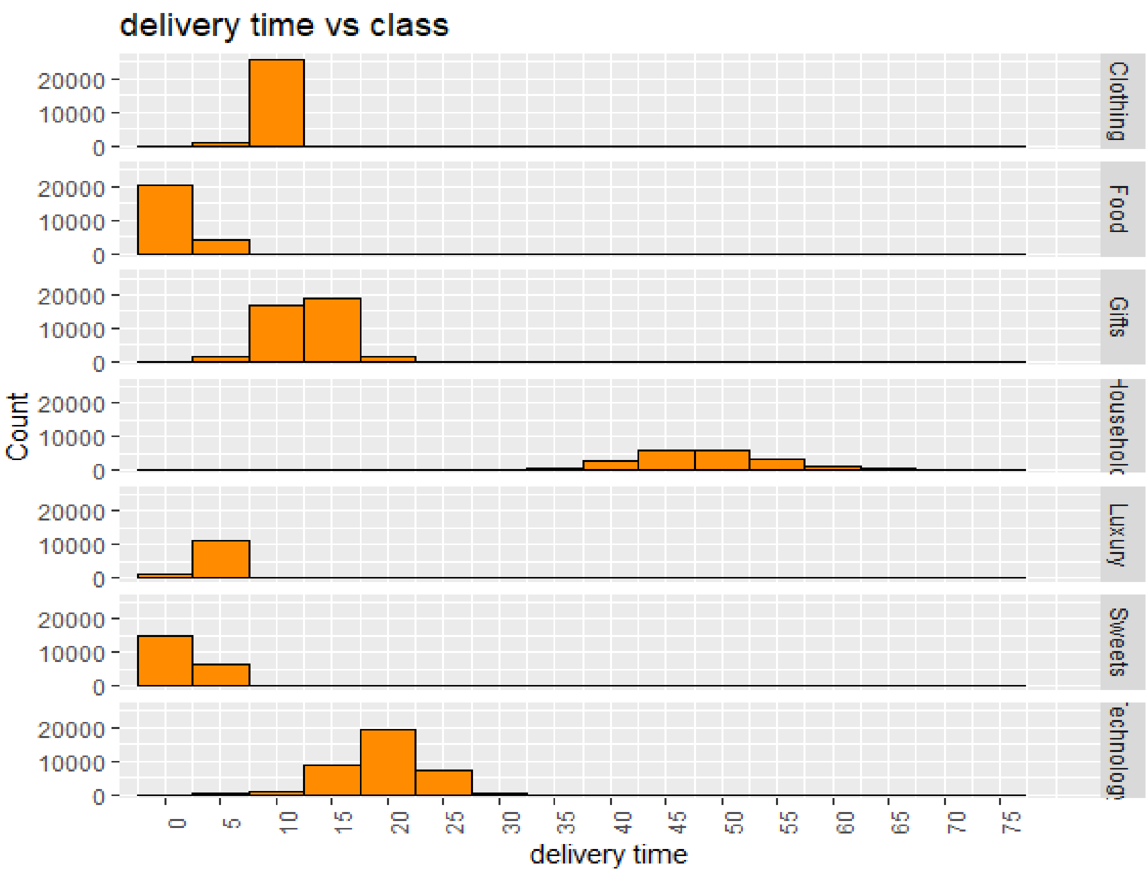


Figure 5: delivery time per class

As seen by the graphs of average price per class and average delivery time per class the statements made about the company selling more than one product is proven. There is a noticeable difference in the average price and average delivery time between the different classes of products. This would impact the graphs of the total price and delivery time. It also explains why the graph of total delivery time was multimodal. The household class has a much larger average delivery time than any other class and thus there is a correlation between the two graphs. Most of the delivery time is because of the technology and household classes. This could be because they are generally large items and thus, they take longer to deliver or that these classes have stock that are only delivered as the products are sold.

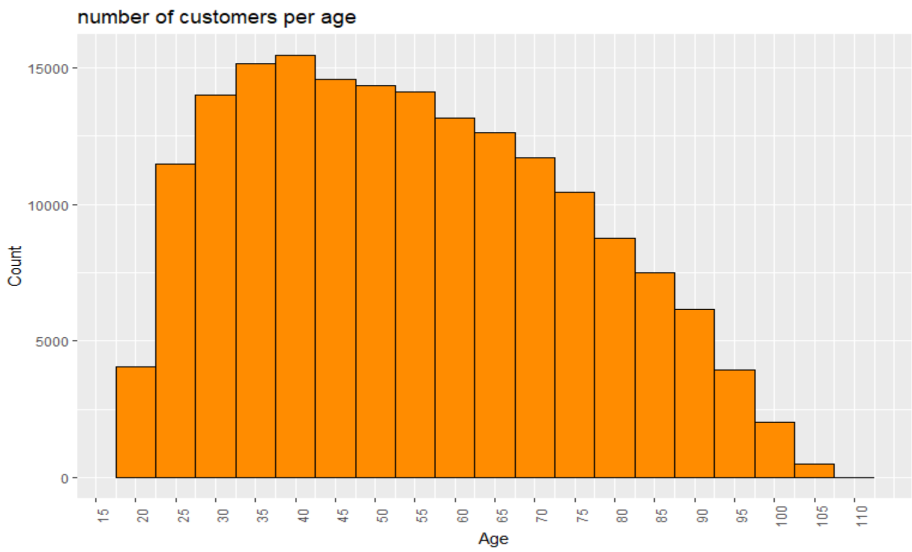


Figure 6: Number of customers per age

The graph of number of customers per age shows that there are customers of all ages that buys products from the company. The graph is also skewed to the right which implies that more customers are over the age of 35 than under. This is good information to use when marketing to a target audience.

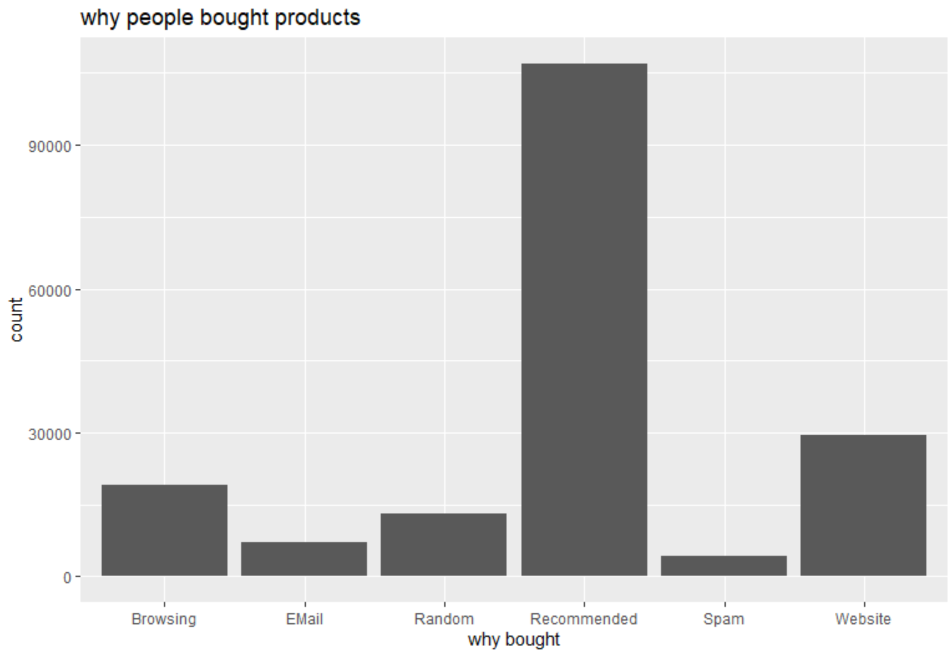


Figure 7: why people bought products

In the graph above people generally buy products because it is recommended to them. This shows that most customers are happy with the quality of products and service they receive. And that more effort can be put into marketing to make more people aware of the products.

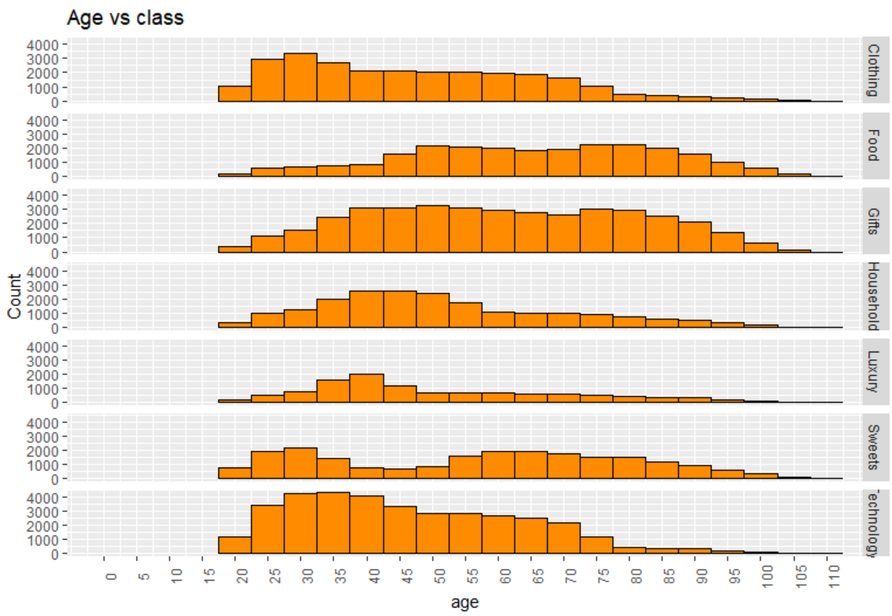


Figure 8: Age vs Class

More people below the age of 60 buy technological product, people between 30 and 45 tend to buy the most luxury items. This could be because that is the age where people start to settle in their lives. All the information in the Age vs Class graph can be used to target a specific age group to advertise specific products too.

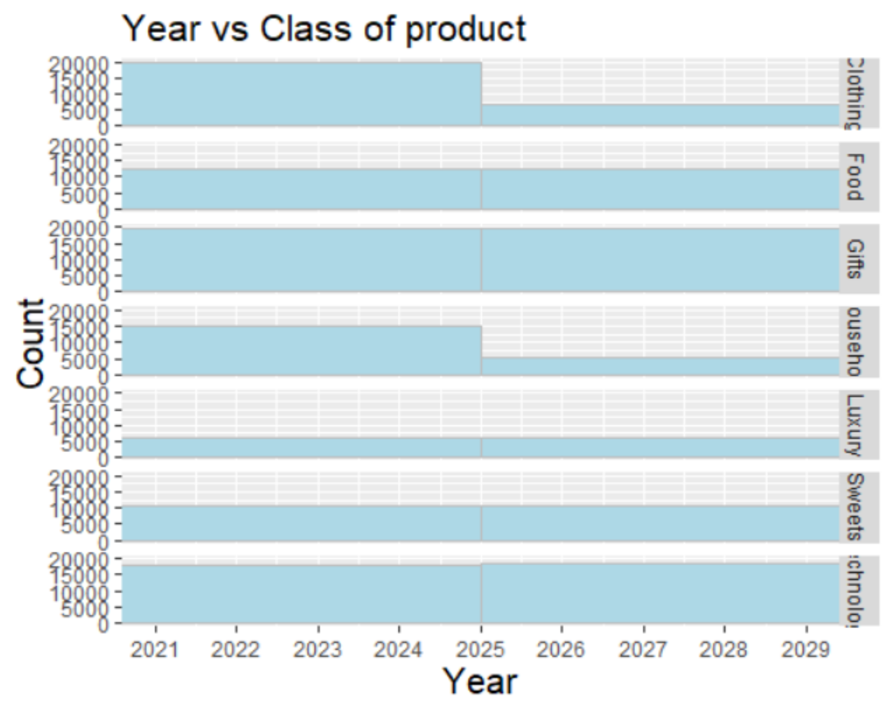


Figure 9: year vs class

Some of the classes was bought more frequently in different time periods. But most of the classes has about the same amount for all the years and thus stays constant.

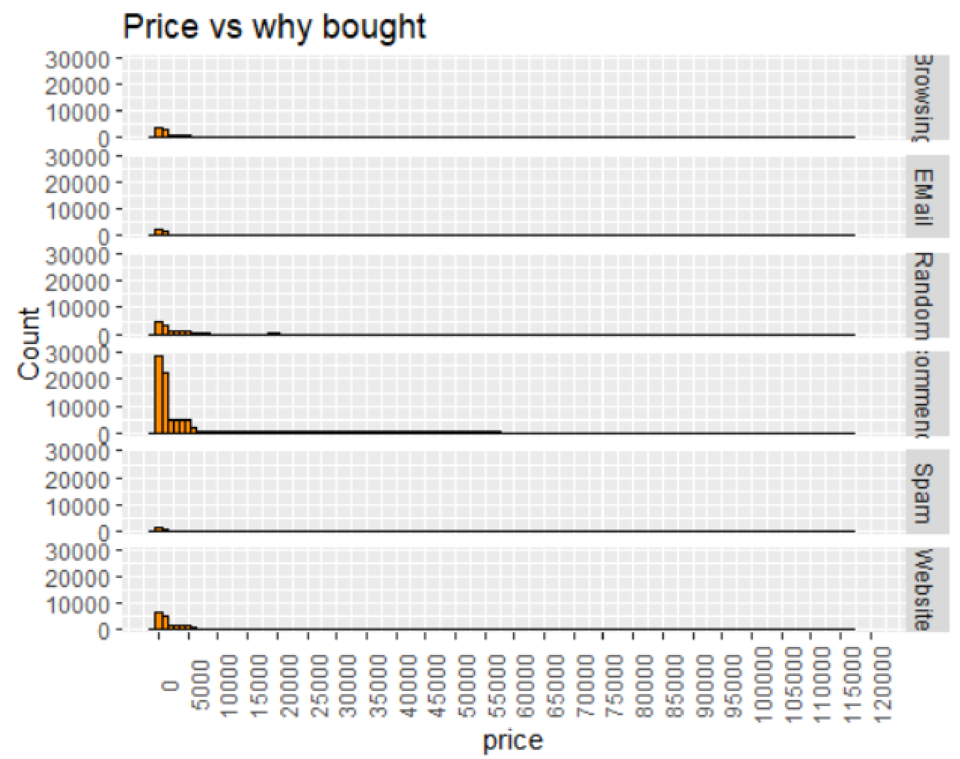


Figure 10: price vs why bought

It is evident that some reasons why people by product leads to the buying higher priced products, but for most part the price range stays relatively low for all the reasons why people bought products.

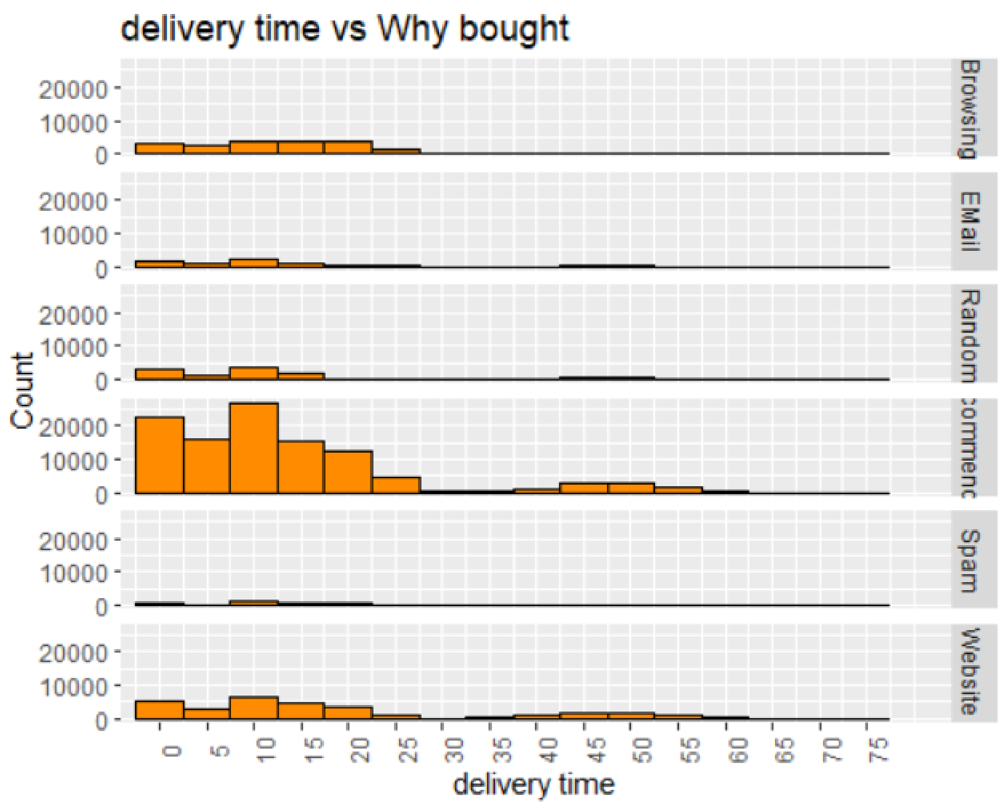


Figure 11: delivery time vs why bought

The delivery time is very different for all the reasons why people bought products. This can be that the two variables are not dependent on each other. If this is not the case the reason must be investigated.

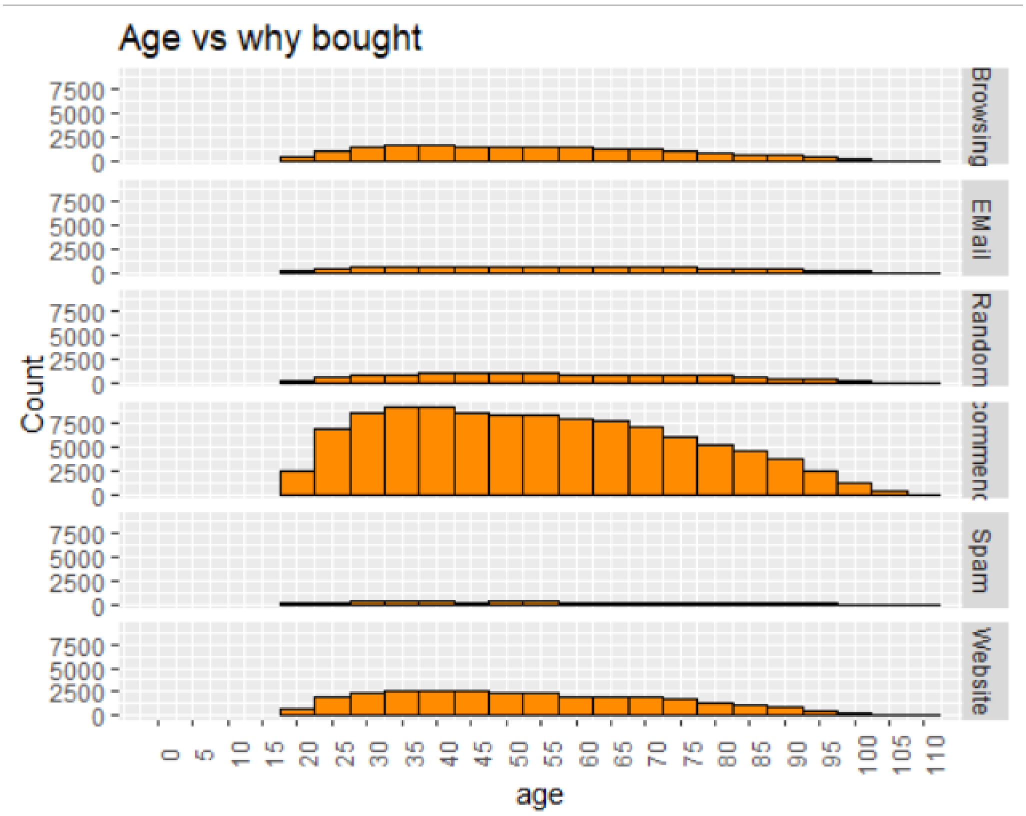


Figure 12: Age vs why bought

The number of ages that buys different type of products for different reasons is very different in this graph. This means that different age groups use different types of communications and has different reasons why the buy certain products. This information is helpful for the marketing department to ensure that they use the best method to get through to the correct are group.

Process capability indices:

```
summary(valid_data$Delivery.time)

min_dt<- 0.5
Q1_dt <- 3.0
med_dt <- 10.0
mean_dt <- 14.5
Q3_dt<- 18.5
max_dt<-75.0
sd_dt<- sd(valid_data$Delivery.time) #13.95608

tech<-valid_data[valid_data$Class=="Technology",]
mean_tech = mean(tech$Delivery.time)
sd_tech<- sd(tech$Delivery.time)
#assuming
USL<- 24 #hours
LSL<- 0

#process capability indices

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

Figure 14: Process capability

Values	
Cp	1.14220682164984
Cpk	0.379693342575295
Cpl	1.90472030072439
Cpu	0.379693342575295

Figure 13: values

Cp	1.142
Cpk	0.380
Cpl	1.905
Cpu	0.380

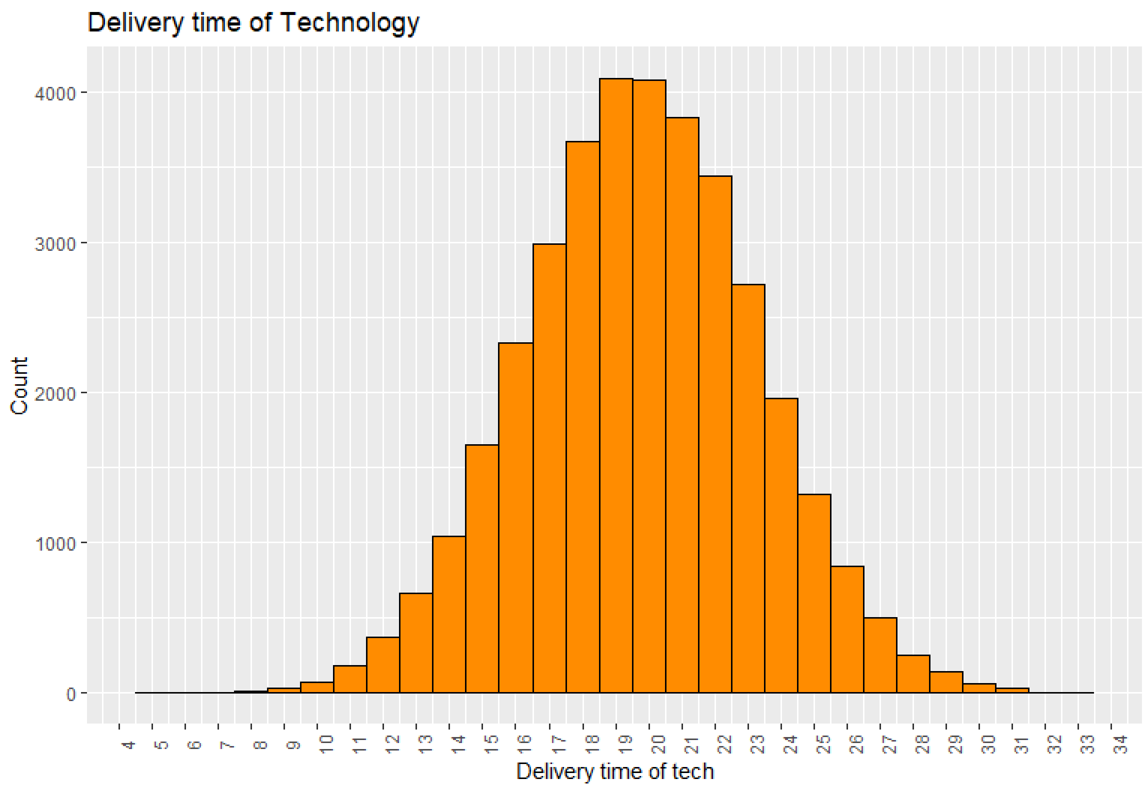


Figure 15: graph of delivery time of technology.

The graph shows a normal distribution of data. LSL of 0 is logical because having a lower specification limit (the lowest limit that a customer would except) as zero would mean that there is no waiting time for delivery. This means that the product is received as it is bought.

The Cp value of 1.142 is an indication that the process is capable, and all the data is inside the USL (Upper Specification Limit) and LSL (Lower Specification Limit). The Cpk value accounts for shifts in the data and we take the minimum value because we account for the most critical value. The value of Cpk = 0.380 indicates that the mean value has shifted but is still well within the specification limits. This all shows that the delivery time of the company is still within the customers specifications but some of the data is starting to fall outside the specifications.

A good result would be that the delivery time of technology falls within the LSL and USL, meaning that the value is within the time frame that a customer would be satisfied with. If the value is larger than the USL it could result in the customer not being satisfied and the business could lose the customer.

Part 3: Statistical process control (SPC)

The X&S charts are constructed by using 30 sample which has 15 sales per sample. The charts are created using ordered data.

X-chart:

Column1	UCL	U2Sigma	U1Sigma	CL	L1Sigma	L2Sigma	LCL
Technology	22.9731	22.10688	21.24066	20.37444	19.50822	18.642	17.77579
Clothing	9.404681	9.259787333	9.114893667	8.97	8.825106333	8.680212667	8.535319
Household	50.24618	49.01819333	47.79020667	46.56222	45.33423333	44.10624667	42.87826
Luxury	5.493524	5.240868	4.988212	4.735556	4.4829	4.230244	3.977587
Food	2.70933	2.63622	2.56311	2.49	2.41689	2.34378	2.27067
Gifts	9.487909	9.112309667	8.736710333	8.361111	7.985511667	7.609912333	7.234213
Sweets	2.896798	2.757124667	2.617451333	2.477778	2.338104667	2.198431333	2.058758

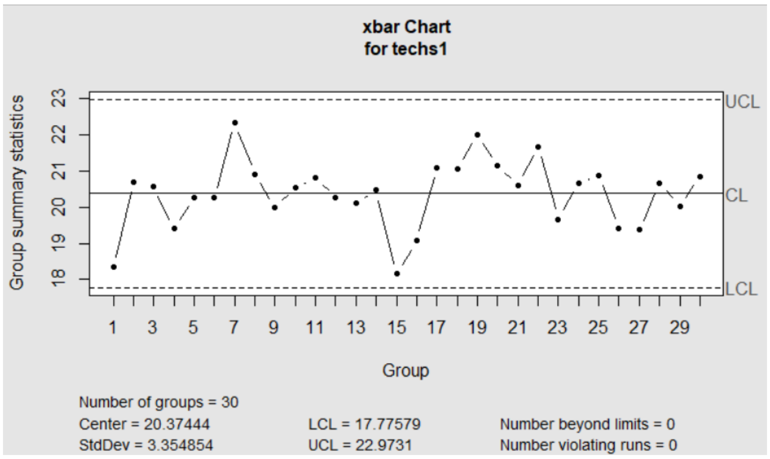
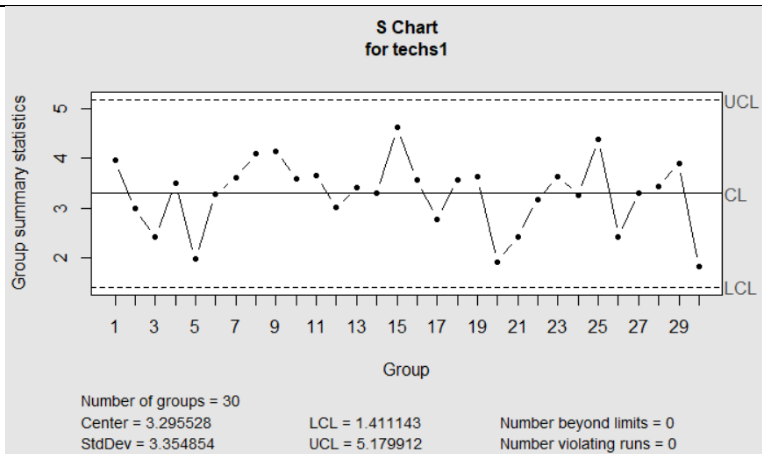
Table 3: X-chart table

S-Chart:

Column1	UCL	U2Sigma	U1Sigma	CL	L1Sigma	L2Sigma	LCL
Technology	5.179912	4.551784	3.923656	3.295528	2.6674	2.039272	1.411143
Clothing	0.8664496	0.7613819	0.6563142	0.5512465	0.4461788	0.3411111	0.236044
Household	7.343248	4.849961867	5.562329333	4.67187	3.781410667	2.890951333	2.000493
Luxury	1.51086	1.327649633	1.144439267	0.9612289	0.778018533	0.594808167	0.411598
Food	0.4371911	0.3841763	0.3311615	0.2781467	0.2251319	0.1721171	0.119102
Gifts	2.246048	1.973687	1.701326	1.428965	1.156604	0.884243	0.611882
Sweets	0.8352331	0.7339508	0.6326685	0.5313862	0.4301039	0.3288216	0.227539

Table 4: S-Chart table

Sample values X & S chart per class for first 30 samples:

X-Chart	S-chart
Technology	
 *Figure 16: x-chart for technology sample*	 *Figure 17: s-chart for technology sample*

The data for technology is controlled as it is between the UCL and LCL.

Clothing

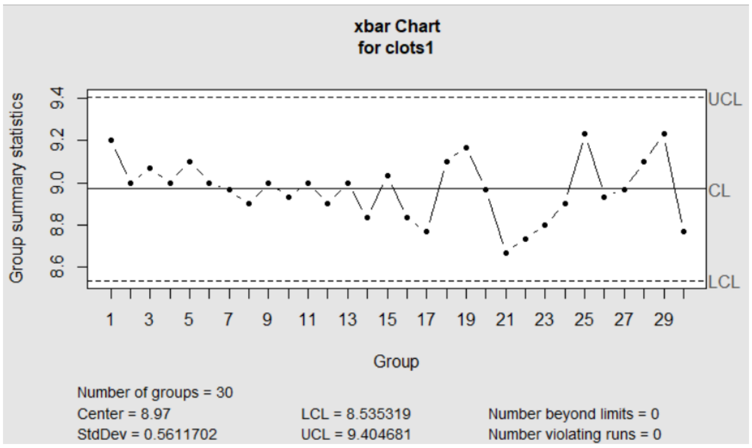


Figure 18: x-chart for clothing sample

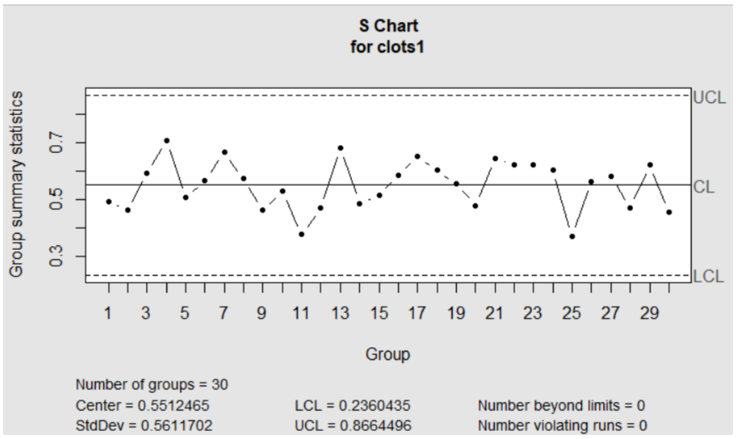


Figure 19: s-chart for clothing sample

The data for clothing is controlled as it is between the UCL and LCL.

Household

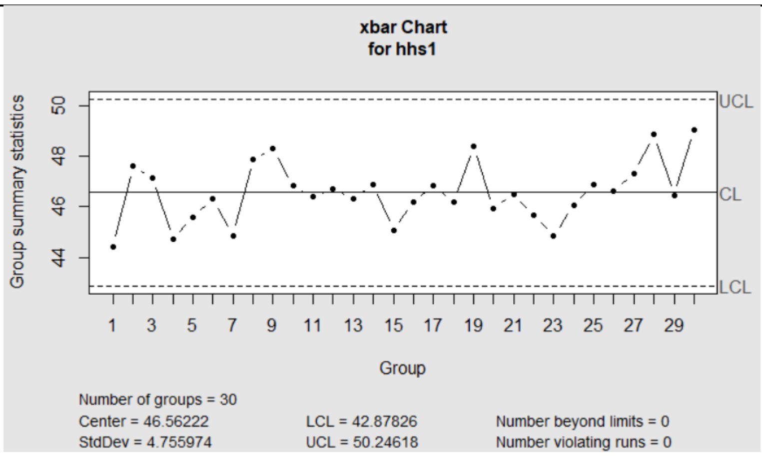


Figure 20: x-chart for household sample

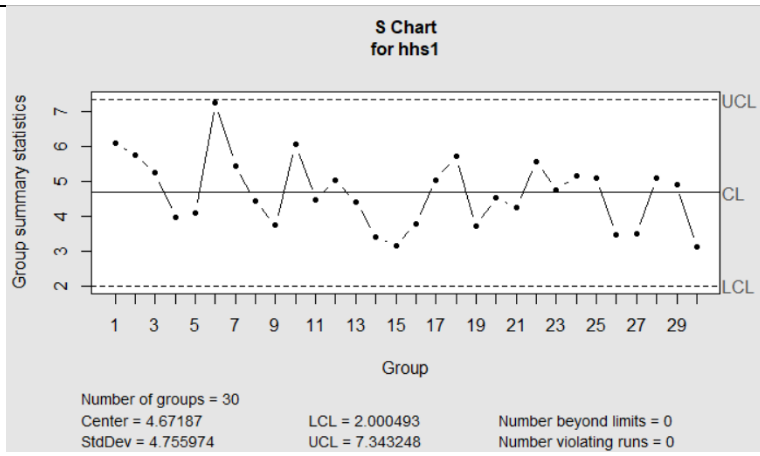


Figure 21: s-chart for clothing sample

The data for household is controlled as it is between the UCL and LCL.

Luxury

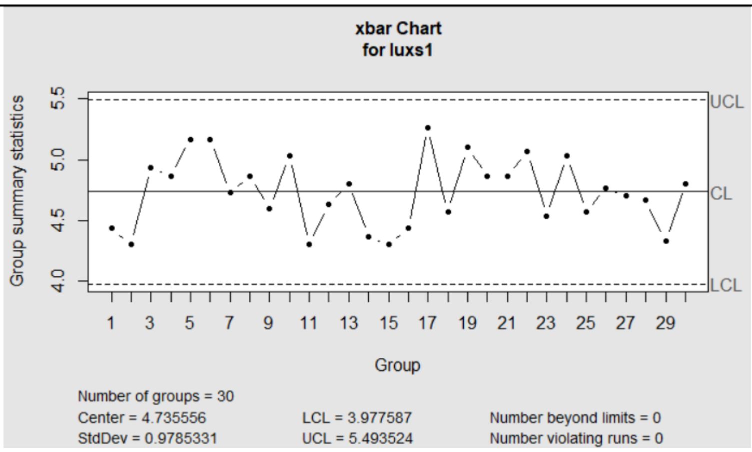


Figure 22: x-chart for luxury sample

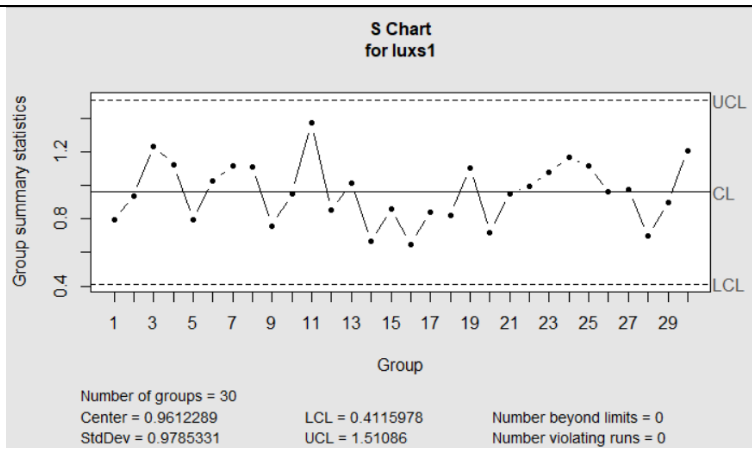


Figure 23: s-chart for luxury sample

The data for luxury is controlled as it is between the UCL and LCL.

Food

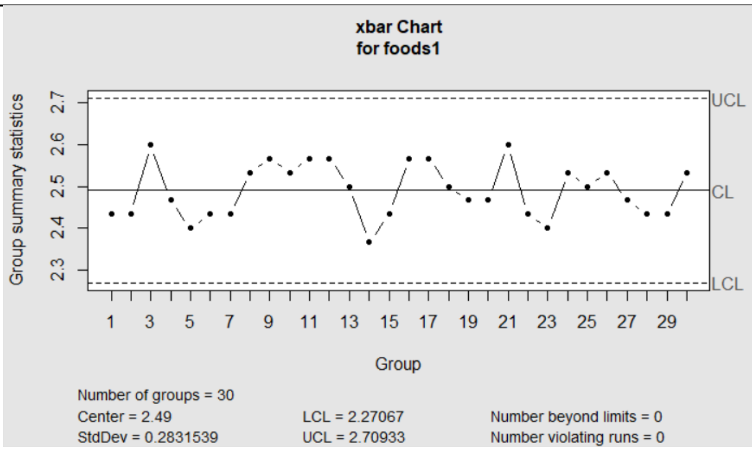


Figure 24: x-chart for food sample

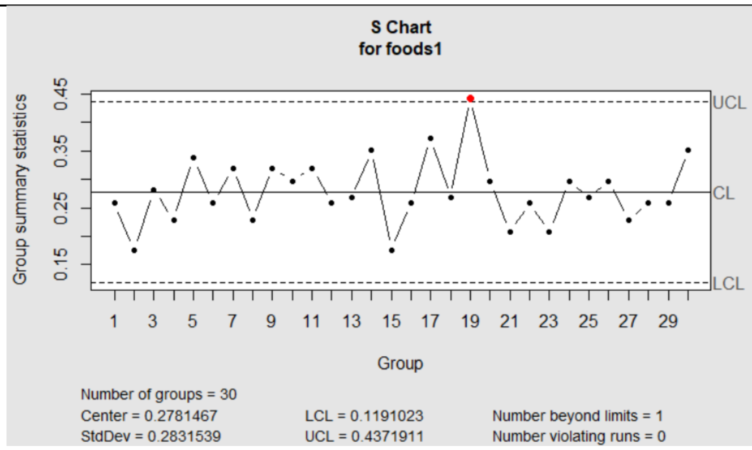


Figure 25: s-chart for food sample

The data for food is controlled as it is between the UCL and LCL.

The one instance of the s-bar that is outside the control limits needs to be removed from the x-bar.

Gifts

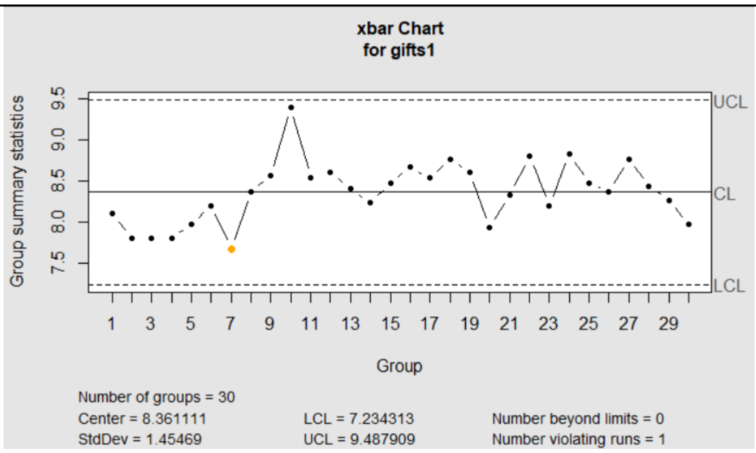


Figure 26: x-chart for gifts sample

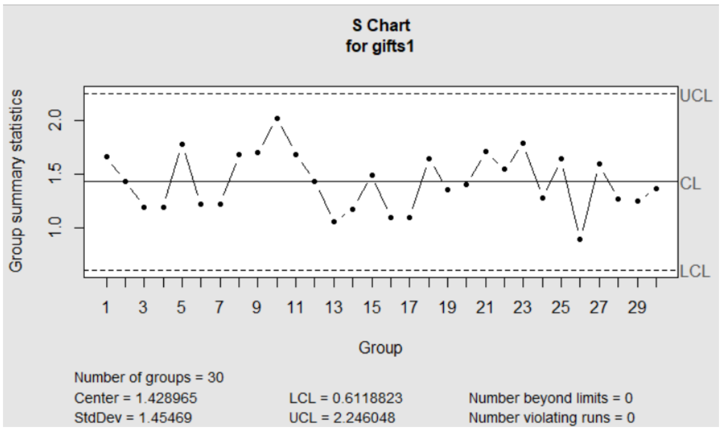


Figure 27: s-chart for gifts sample

The data for food is controlled as it is between the UCL and LCL.

Sweets

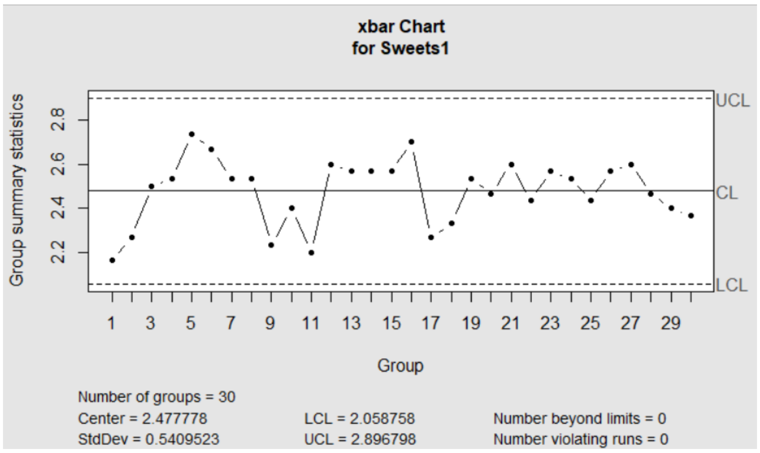


Figure 28: x-chart for sweets sample

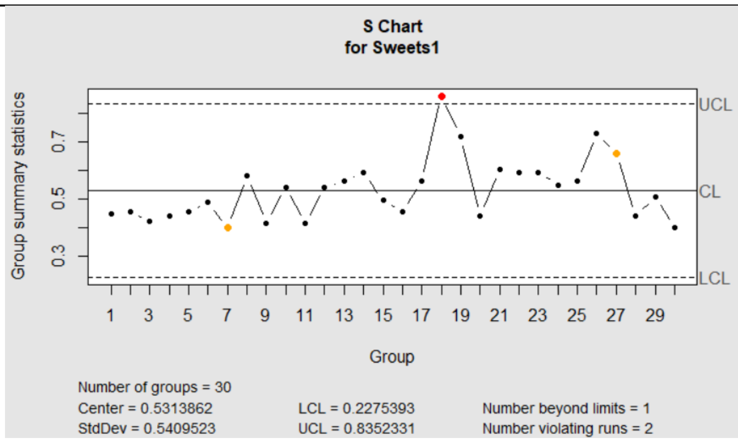


Figure 29: s-chart for sweets sample

The data of sweets is controlled the one value that falls outside of the control limits of the s-bar needs to be removed from the x-bar and then the data will be fine.

X & S charts per class for all data:

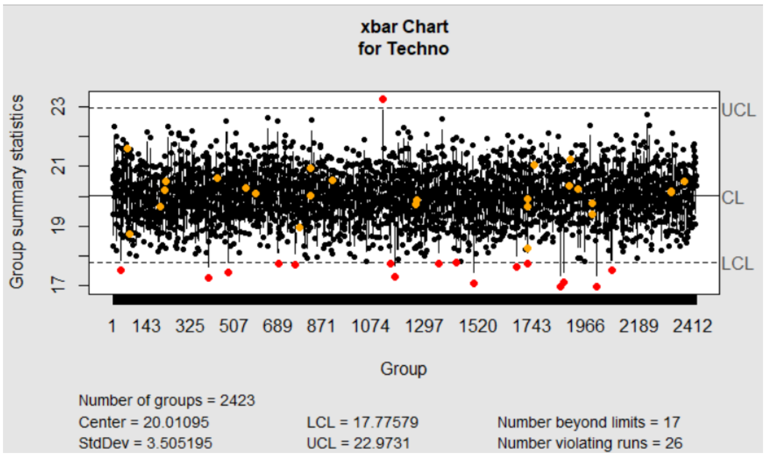
<table>
<tr><th>X-Chart</th><th>S-chart</th></tr>
<tr><td colspan="2">Technology</td></tr>
<tr><td>


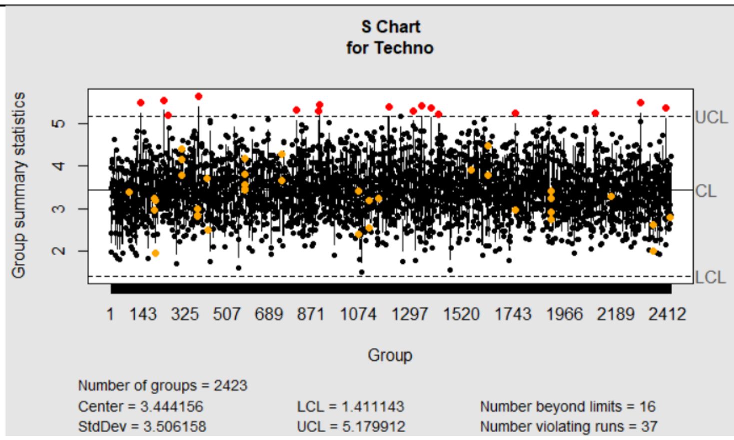
Figure 30: x-chart for technology
</td><td>


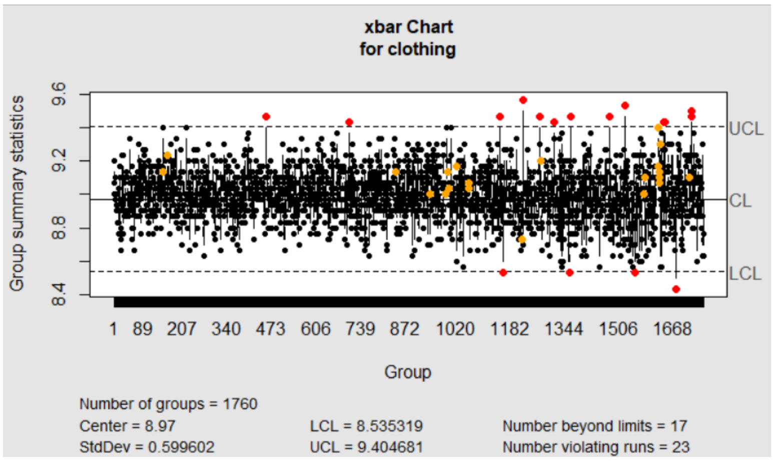
Figure 31:s-chart for technology
</td></tr>
<tr><td colspan="2">Some of the data falls outside the limits, but for the most part the data is controlled. The instances that fall outside the control limits of the s-chart should be removed from the x chart to makes sure the data is all within the control limits.</td></tr>
<tr><td colspan="2">Clothing</td></tr>
<tr><td>


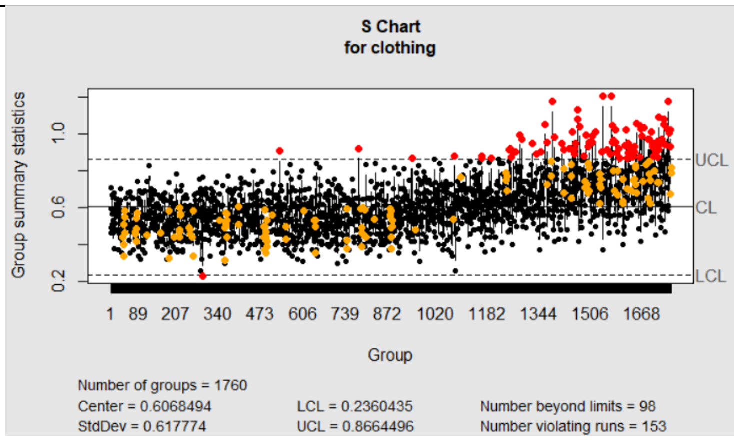
Figure 32:x-chart for clothing
</td><td>


Figure 33:s-chart for clothing
</td></tr>
<tr><td colspan="2">Most of the samples are within the control limits making the data controlled. The s chart starts to move out of the UCL at the end and the reason for this need to be investigated. A possible cause could be seasons that change and because of the instances that are outside the control limits of the s chart needs to be removed from the x chart.</td></tr>
</table>

Household

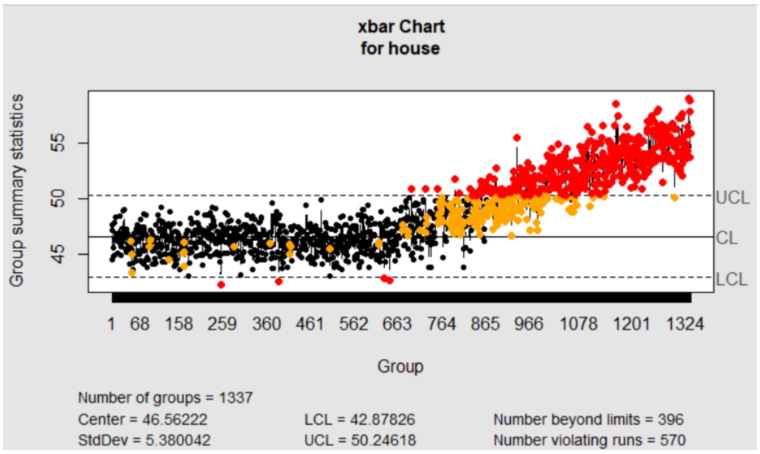


Figure 34:x-chart for household

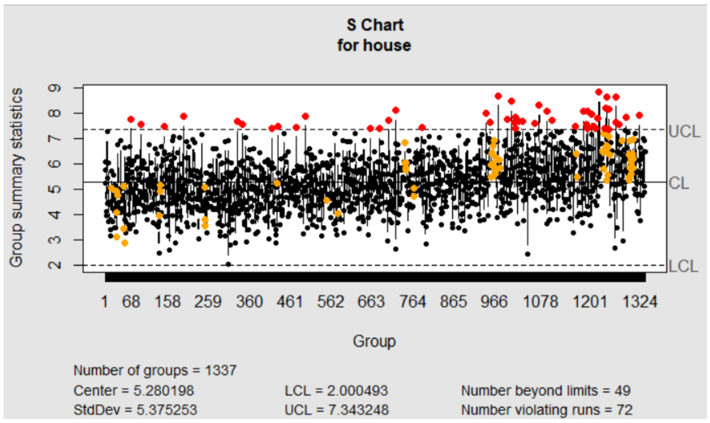


Figure 35: s-chart for household

The data falls outside the controlled limits. There is a rapid increase in the delivery time for the household class and this must be investigated. The data is not sable or controlled.

Luxury

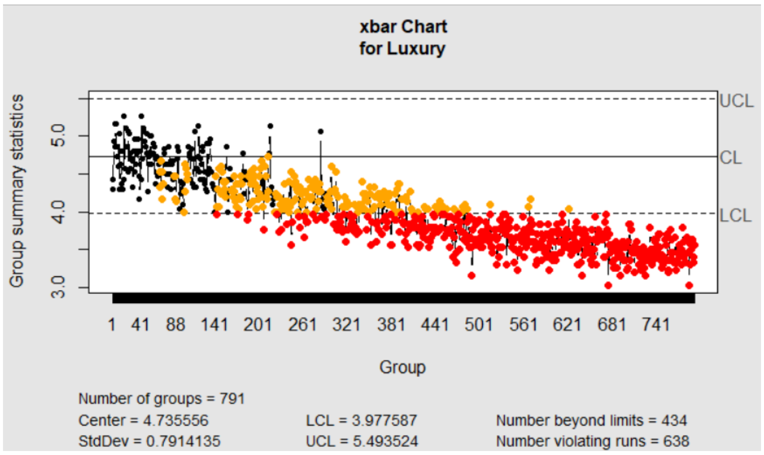


Figure 36: x-chart for luxury

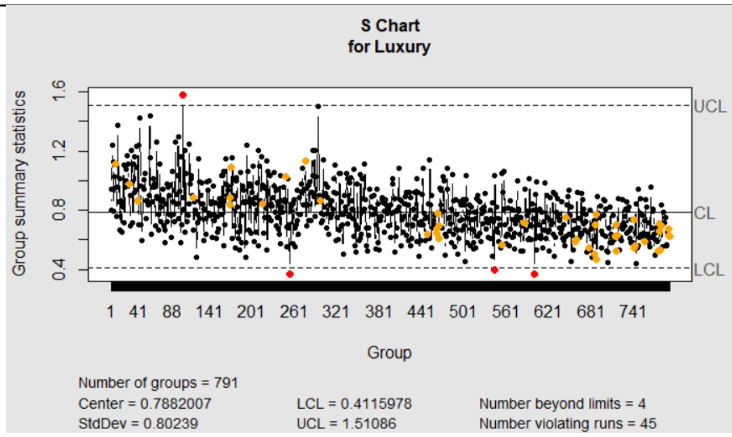


Figure 37: s-chart for luxury

There is a rapid decrease for the luxury class delivery time. This decrease must be investigated. The data is not controlled nor stable. The decrease in delivery time is not a bad thing as this means the products are delivered at a faster pace and thus saving money.

Food

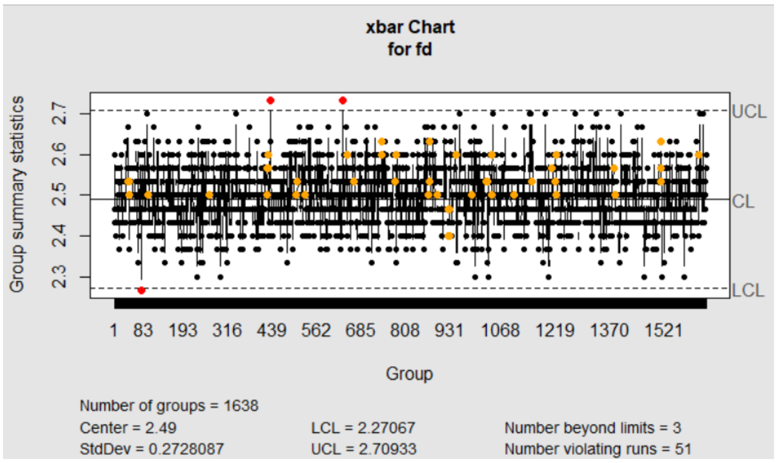


Figure 38: x-chart for food

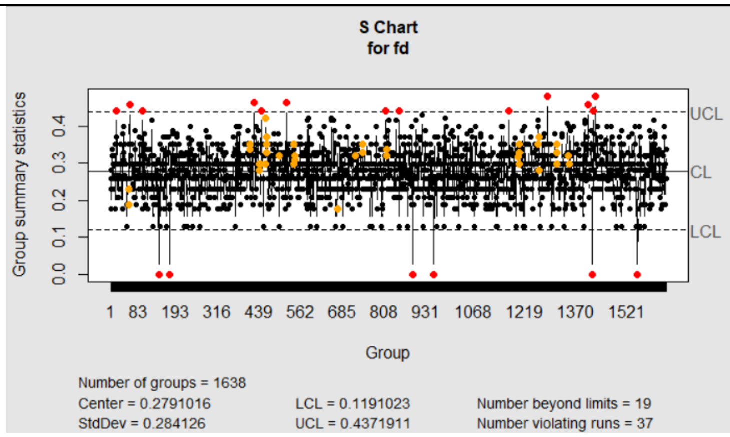


Figure 39: s-chart for food

The data for food is controlled as the majority falls between the limits. The instance that fall outside of the UCL and LCL of the s chart needs to be removed from the x chart and then it will be totally in control.

Gifts

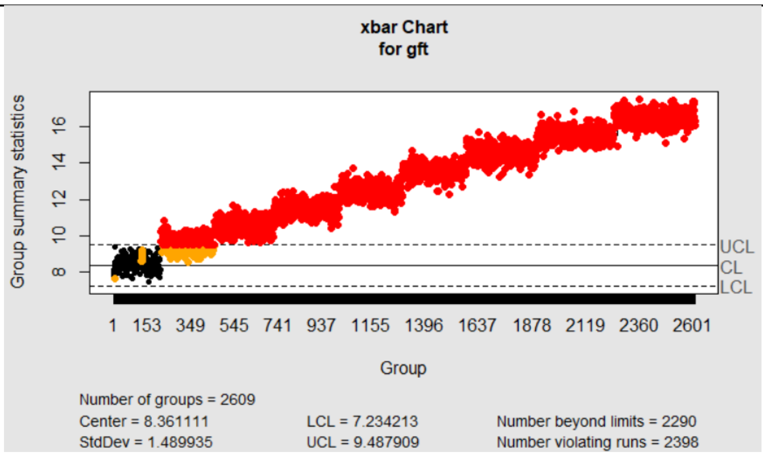


Figure 40: x-chart for gifts

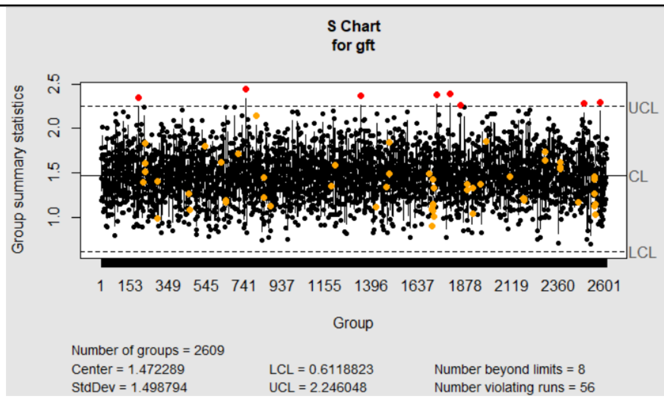


Figure 41: s-chart for gifts

The data is not controlled or stable. The delivery time increases rapidly for the gifts class and this increase must be investigated. This could lead to a rise in expenses and would be bad for business.

Sweets

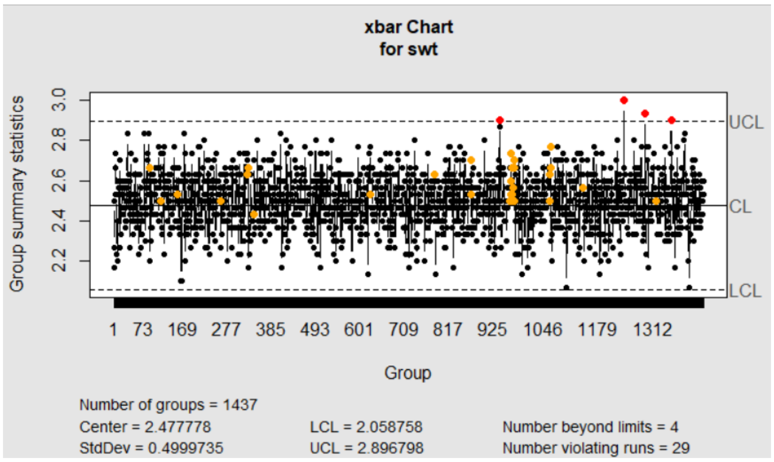


Figure 42: x-chart for sweets

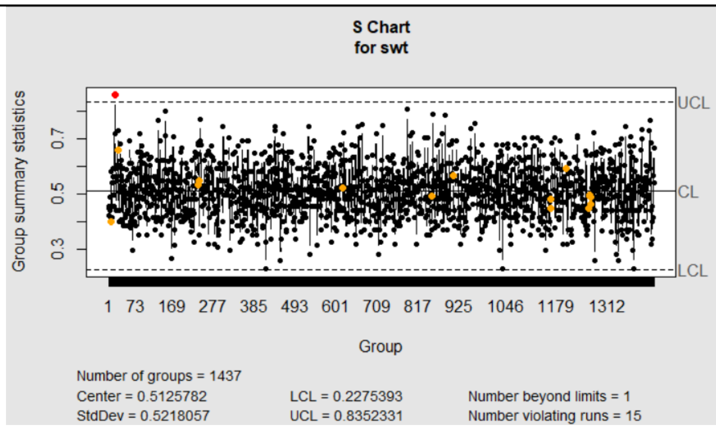


Figure 43: s-chart for sweets

The data of sweets is controlled even if some of the values fall outside the limits the x-chart is appropriate.

Part 4

Control table x values:

Column1	Total	1st sample	2nd sample	3rd sample	3rd last	2nd last	last
Technology	17	37	398	483	1872	2009	2071
Clothing	17	455	702	1152	1677	1723	1724
Household	396	252	387	629	1335	1336	1337
Luxury	434	142	171	184	789	790	791
Food	3	75	432	NA	NA	NA	633
Gifts	2290	213	216	218	2607	2608	2609
Sweets	4	942	1243	NA	NA	1294	1358

Table 5: control table x values

The table shows the number of data points that are outside of the control limits. It also shows the position of the 1st, 2nd, 3rd,3rd last, 2nd last and sample that was outside the control limits. In the table it is clear that the Household, Luxury and Gifts classes were out of control. This is seen by the high number of total data points that are out of control. The reason for this must be investigated by the company.

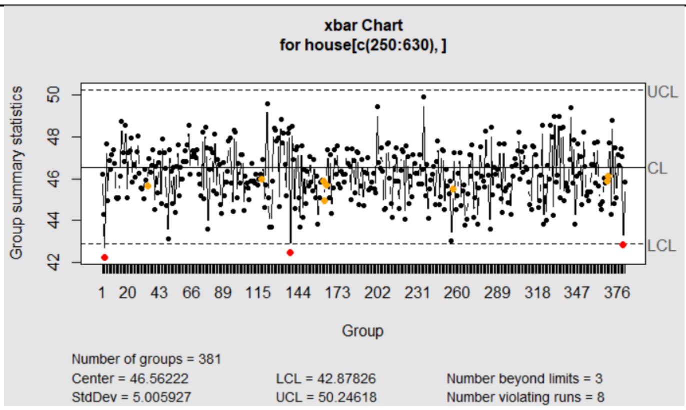
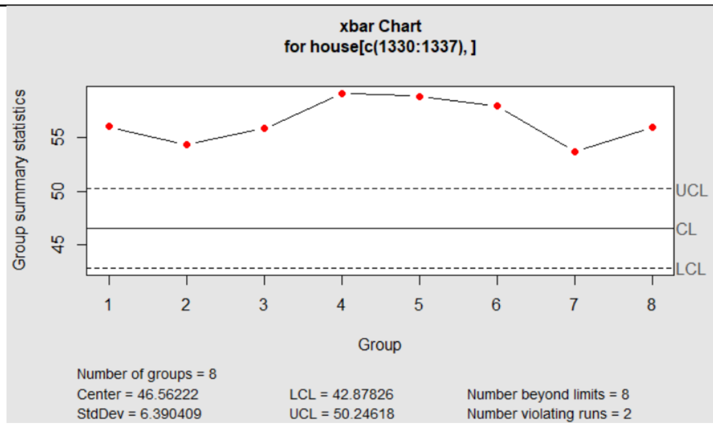
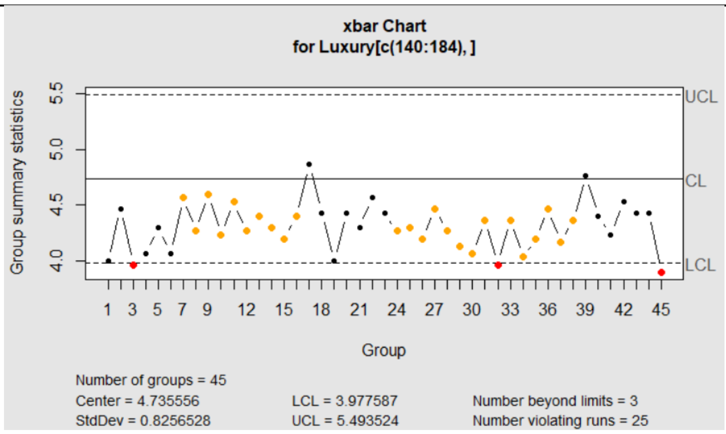
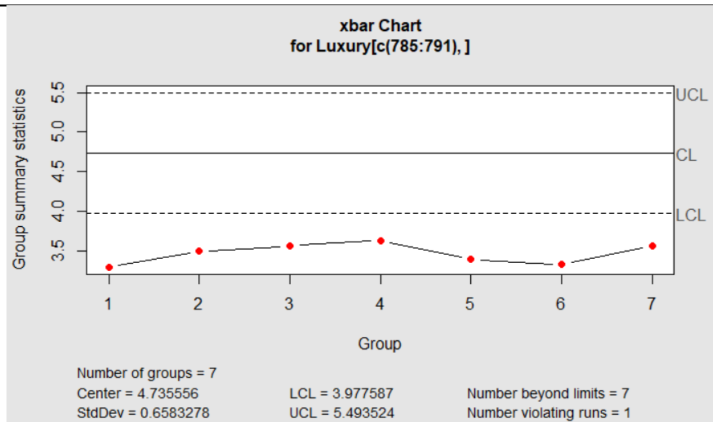
Control table s values:

Column1	Total	1st sample	2nd sample	3rd sample	3rd last	2nd last	last
Technology	16	129	230	251	2095	2290	2400
Clothing	98	289	530	780	1754	1756	1757
Household	49	65	89	147	1271	1290	1323
Luxury	4	103	254	NA	NA	543	600
Food	19	19	57	96	1422	1429	1553
Gifts	8	193	746	1342	1855	2493	2576
Sweets	1	8	NA	NA	NA	NA	NA

Table 6: control table s values

The table above shows the total number of values of the standard deviation that falls outside the UCL and LCL. It also shows the position of the first 3 values and last 3 values that fall outside the control limits. Out of the table it is clear that the Clothing, Household and Food classes have the most values that fall outside the limits and thus will have the most values that need to be removed from the corresponding x data sets. The Clothing class is by far the largest with 98 values.

Plots of the first three and last three points out of control classes:

First 3	Last 3
Household	
 *Figure 44:first 3 out of control for household*	 *Figure 45:last 3 out of control for household*
Luxury	
 *Figure 46:first 3 out of control for Luxury*	 *Figure 47: last 3 out of control for luxury*

Gifts

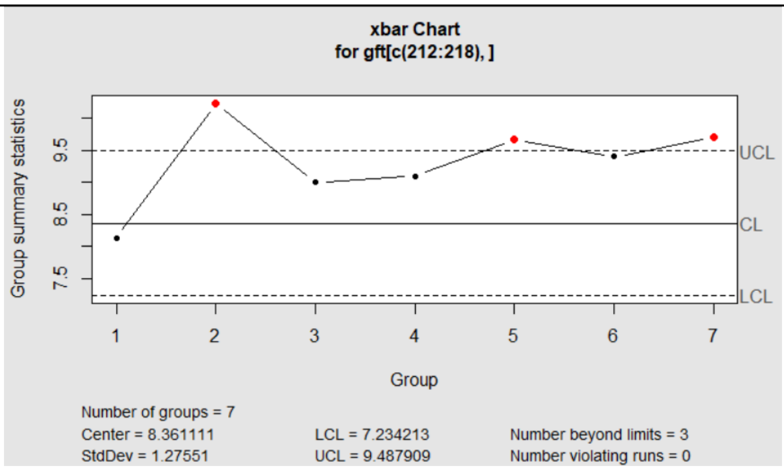


Figure 48: first 3 out of control for gifts

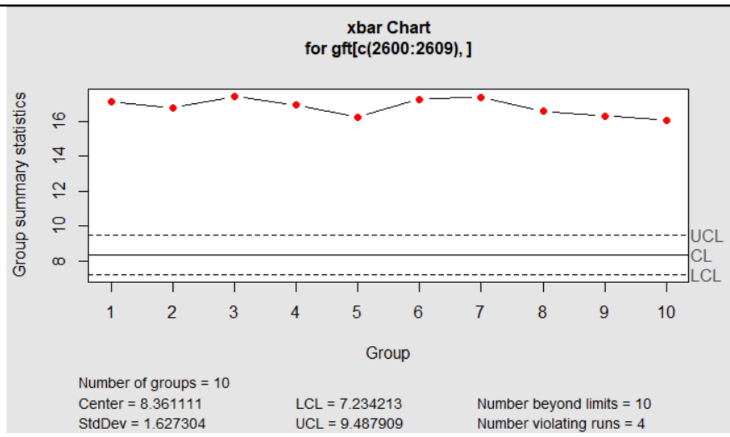


Figure 49: last 3 out of control for gifts

The plot shows the first 3 and last 3 samples that are out of control for the three classes that were identified as the out-of-control classes. In all three graphs the class starts in control and as the years go on the data starts to move more out of control. The reason for this needs to be investigated.

Most consecutive samples of S-bar between -0.3 & 0.4:

Class	Max consecutive	Ending sample index number	Position in sample space	Value
Technology	6	372	97	3.26708
Clothing	4	1013	236	0.5345225
Household	3	843	212	4.719514
Luxury	4	63	19	0.9536896
Food	6	441	138	0.2968084
Gifts	5	1651	457	1.71617
Sweets	4	971	259	0.5163978

Table 7: consecutive s-bar samples

From the table it shows the largest number of consecutive samples between the sigma values of -0.3 and 0.4 is food with 6. This is low, indicating that a large amount of the sample values falls outside this sigma values.

The likelihood of making a Type I error

Assumptions for the Null Hypothesis (Type I) error:

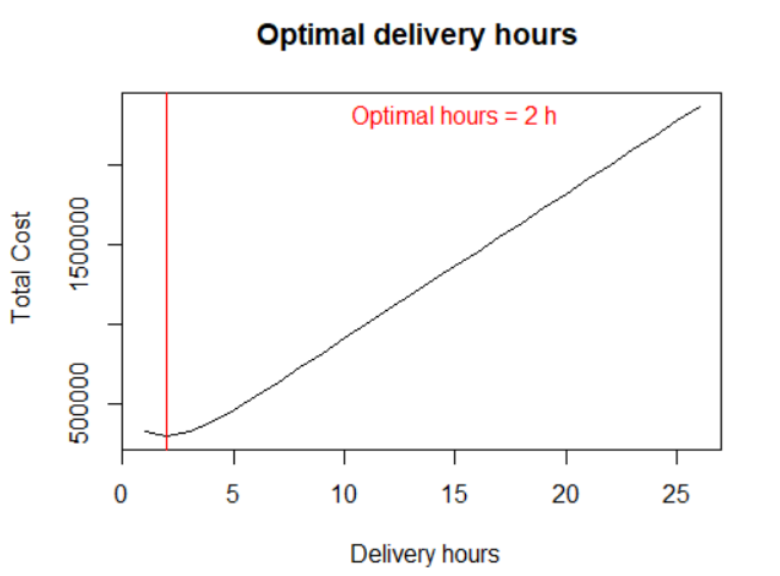
- **Ho =** the processes in control and centred on the centreline
- **Ha =** process is not in control and has moved from the centreline or has increased or decreased in variation.

	Process is fine	Process is not fine
SPC shows process not fine.	Type I (manufacturing) error	Correct the process (fix the process)
SPC shows that the process is fine.	Correct to do nothing	Type II (consumer's) error

Probability of performing a Type I error:

- **A =** The probability is 0.002699767 or 0.27%. This is the probability that the product will actually be delivered on time, but the computer indicates that it has not been delivered on time.
- **B =** The probability is 0.72666 or 72.67% that B will have a Type I error between the values of -0.3 & 0.4.
-

Minimizing the delivery time:



The cost for different hours of the delivery time of the technology class was compared by looping through all the hours. The current mean hours is 20.01095. This is with 1356 sales that are above a 26-hour delivery time. At a rate of R329 lost per hour it leads to a loss of R446 126. It cost R2.5 per hour to reduce the mean delivery time by and hour (shifting the graph to the left). For 26 hours this will cost a maximum of R 636 072.5

The minimum delivery cost needs to be calculated to reduce the loss of sales. This is done by looping through all the hours and finding the optimum value to shift the mean by. The optimum value found was 2 hours. This means the new mean is 20.01095 – 2 = 18.01095 hours.

Looking at the graph it is clear that it is different from the Taguchi loss function as it is not symmetrical.

Probability of making Type II error

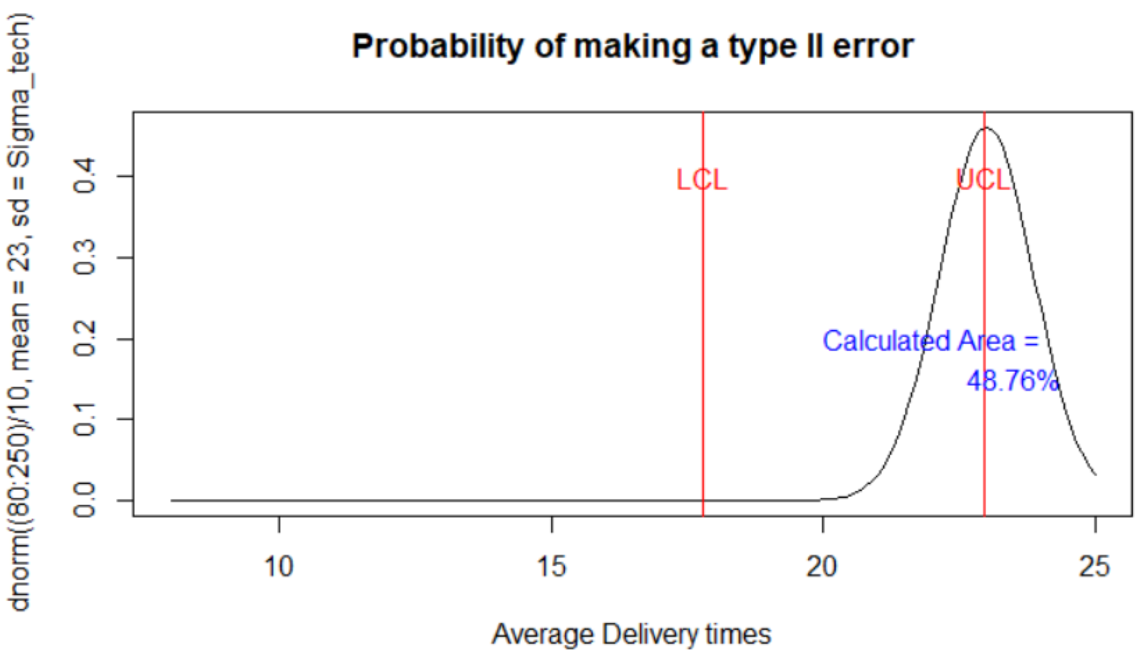


Figure 50: Probability of Type II error

When the average moves to 23 hours the probability of making a Type II error for delivery time of the technology class is 0.487613 or 48.76%. This is relatively high, and the company must beware to assume that the product is on time just because the computer show it is. The control limits are shown between the UCL and LCL which is indicated with red line on the graph.

Part 5: DOE and MANOVA

P=0.05 is used as the p value for the MANOVA test as it is the value that is most universal.

MANOVA number 1:

- **Dependent variables:** Price, Delivery time, Age
- **Independent variable:** Class
- **Ho =** Price, delivery time, Age makes no significant difference to the way people buy the products.
- **H1 =** At least one of Price, delivery time or age will influence the buying pattern observed.

MANOVA test P value:

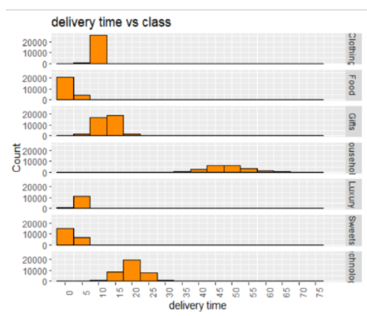
P= 2.2e-16

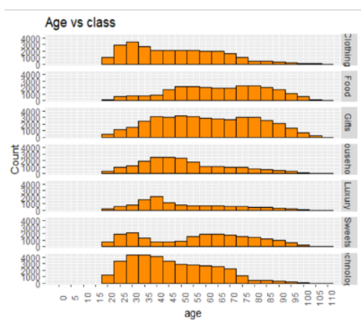
P<0.05 thus there is at least one variable that influences the class.

Reject H0

Dependent variable	P value	Influence or not
Price	2.2e-16	P<0.05 thus it will influence the class and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 3
Delivery time	2.2e-16	P<0.05 thus it will influence the class and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 5
Age	2.2e-16	P<0.05 thus it will influence the class and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 8

Table 8: MANOVA 1





In conclusion:

All the variable influences the buying patterns of the different classes. The delivery time for the household class is much higher than the other classes. But this is not feasible as the X-bar chart shows that Household is out of control. More people below the age of 60 buy technological product, people between 30 and 45 tend to buy the most luxury items. This could be because that is the age where people start to settle in their lives. The luxury class also has a higher price. This is most possible because luxury items are more expensive items.

MANOVA number 2:

- **Dependent variables:** Price, Delivery time, Age
- **Independent variable:** Why bought
- **Ho =** Price, delivery time, Age makes no significant difference to the why people buy the products.
- **H1 =** At least one of Price, delivery time or age will influence why people bought certain products.

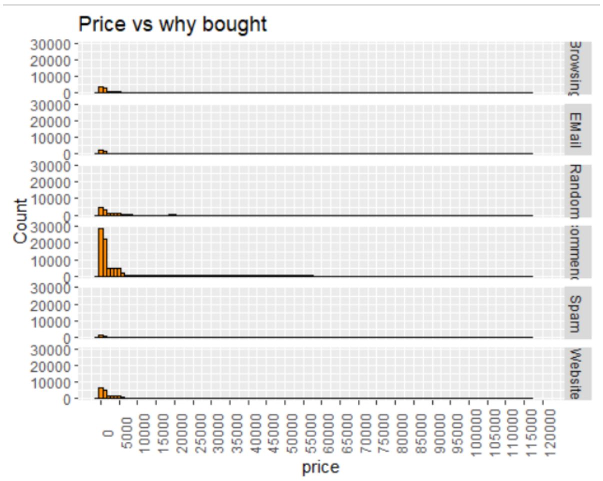
MANOVA test P value:

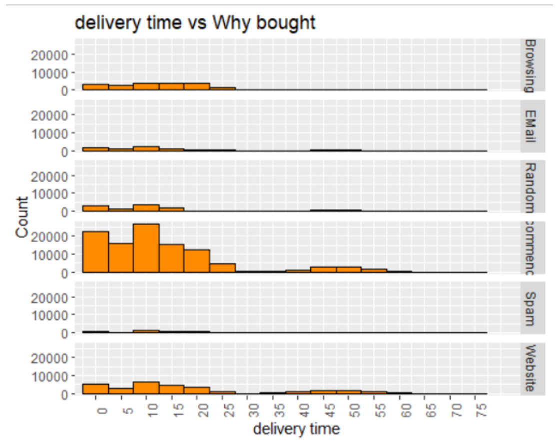
P= 2.2e-16

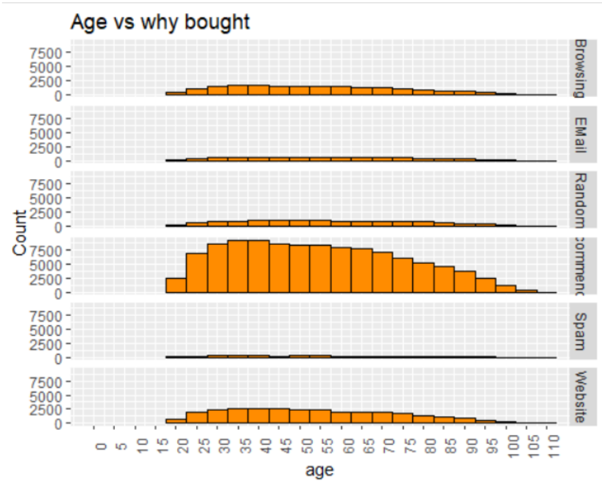
P<0.05 thus there is at least one variable that influences why the products was bought. Reject H0

Table 9: MANOVA 2

Dependent variable	P value	Influence or not
Price	2.2e-16	P<0.05 thus it will influence why people bought the products and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 10
Delivery time	2.2e-16	P<0.05 thus it will influence why people bought the products and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 11
Age	2.2e-16	P<0.05 thus it will influence why people bought the products and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 12







In conclusion:

All the variables will influence why products were bought. The price has the highest count for comments. This can be because we are in a digital age where most people use the internet to communicate and thus trusts the reviews people leave in comments. The will also explain why the delivery time and age features are so highly influential with comments. Spam has a low count for all the features. This could be because of people having spam filters and most people do not read spam messages.

MANOVA number 3:

- **Dependent variables:** Day, Month, Year
- **Independent variable:** Class
- **Ho =** Day, Month, Year makes no significant difference to the way people buy the products.
- **H1 =** At least one of Day, Month, Year will influence the buying patterns.

MANOVA test P value:

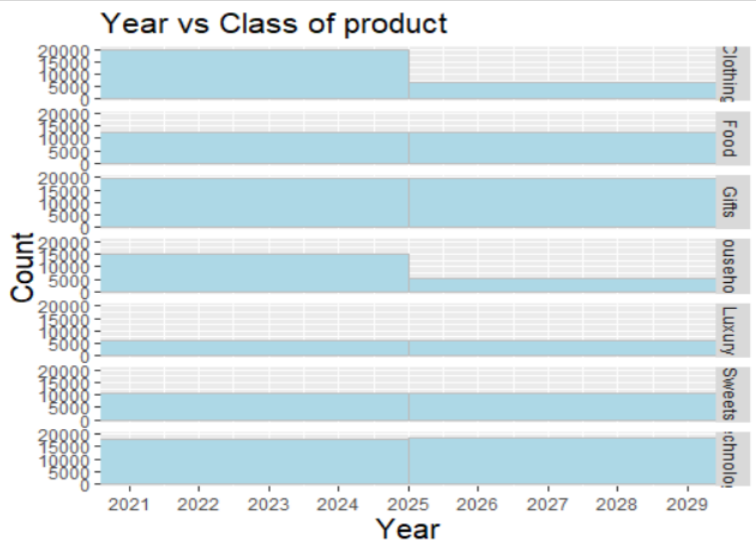
P= 2.2e-16

P<0.05 thus there is at least one variable that influences the class.

Reject H0

Table 10: MANOVA 3

Dependent variable	P value	Influence or not
Day	0.1766	P > 0.05. This indicates that the day does not influence the buying patterns of the different classes.
Month	0.2859	P > 0.05. This indicates that the month does not influence the buying patterns of the different classes.
Year	2.2e-16	P<0.05 thus Year will influence the classes and thus also the buying pattern. It is also different for all sales. As seen in the link: Figure 9



In conclusion:

Only the year will influence the buying pattern of the classes. The food, gifts, luxury, sweets, and technology classes have roughly the same count through all the years. Household and Clothing have different counts throughout the years. The count decrease for both classes from 2026-2029 and the reason for this needs to be investigated.

Part 6: Reliability of the service and products

Problem 6:

Taguchi loss function: $L(x) = k^*(x - T)^2$

- k = 28125
- T = 0.06

L(x) = 28125*(x-0.06)^2

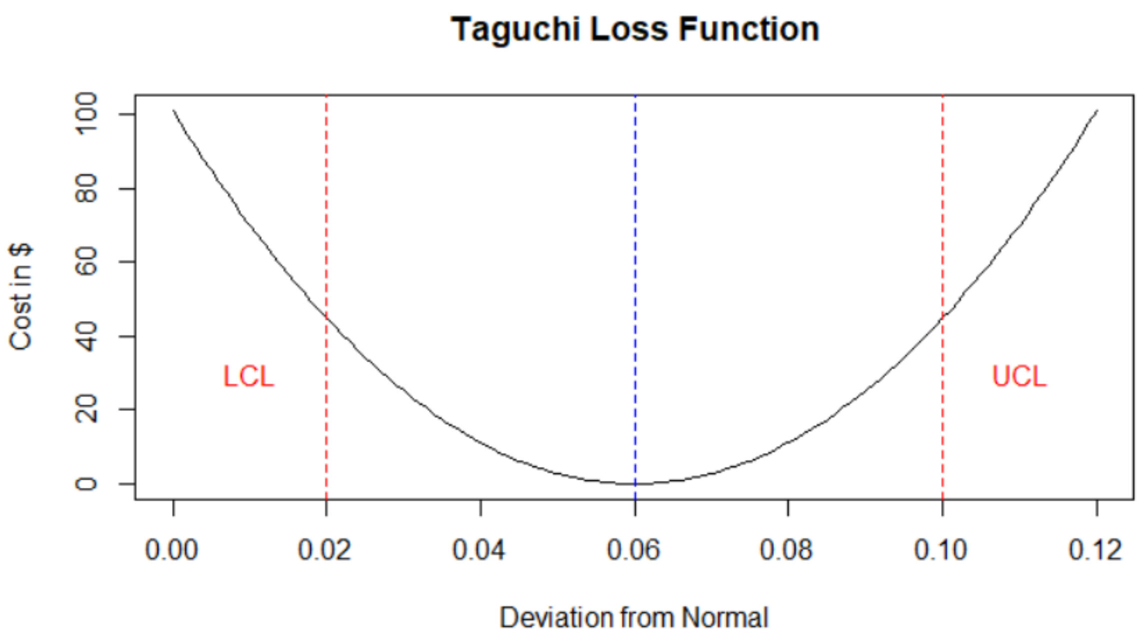


Figure 51: Taguchi loss function $45

The LCL is 0.02 and the UCL is 0.1. The more the product deviates from the normal value of 0.06 the more the cost will be. If the product is outside the control limits the cost will be bigger than $45. Thus, it is best to stay within the UCL and LCL.

Problem 7:

Taguchi loss function: $L(x) = k^*(x - T)^2$

- k = 21875
- T = 0.06

$L(x) = 21875*(x-0.06)^2$

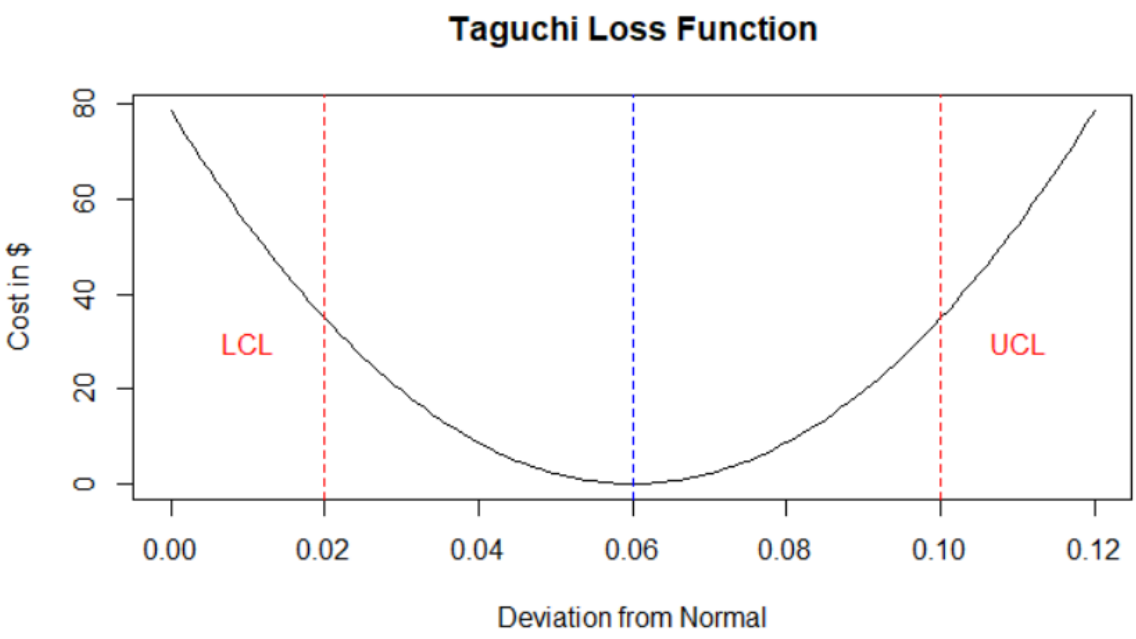


Figure 52: Taguchi loss plot $35

7.B:

- $L(x) = 21875*(x-0.06)^2$
- $L(0.027) = 21875*(0.027)^2$
- $L(0.027) = \$15.946875$

It the deviation is 0.027 cm the company will make a loss of $15.95/product. The deviation of 0.027 cm is still within the UCL and LCL and thus lower than $35 loss.

Problem 27:

a) The probability that one machine at each stage is working:

P(working) = P(A) x P(B) x P(C)

P(working) = 0.85 x 0.92 x 0.9

P(working) = 0.7038

This means that the probability of Magnaplex getting a product through the manufacturing process without failure is 0.7038 or 70.38%.

b) The reliability of having two machines at each stage:

P(working) = (1-P (A not working)) x (1-P (B not working)) x (1-P (C not working))

P(working) = $(1 - (1-0.85)^2) \times (1 - (1 - 0.92)^2) \times (1 - (1 - 0.90)^2)$

P(working) = 0.9615

The probability when from 0.7038 to 0.9615 which is a much larger change of getting through the manufacturing process without failure.

6.3:

For 21 vehicles for 1560 days:

Table 11: Binomial for 21 vehicles

	Number of vehicles available	Number of days
1	21	**1344**
2	20	**190**
3	19	**22**
4	18	**3**
5	17	**1**

Using the binomial formula: $P(x) = \binom{n}{x} p^x q^{n-x} = \frac{n!}{(n-x)!\,x!} p^x q^{n-x}$

N = 21

- P(x<1) = 0.03118647
- P (2) = 0.03318544
- P (3) = 0.02568446
- P (4) = 0.0267331
- P (5) = 0.03549307

The results were obtained by running a function for all the values and then determining at what probability the value was equal to zero. This was then the values entered above.

Each bullet represents a line in the table that was used to calculate the probability.

The weighted mean is: $\frac{x1*1344+x2*190+x3*22+x4*3+x5*1}{1560}$

W = 0.03134654

This value is used to calculate the reliable delivery probability.

The reliability of the delivery vehicles:

P(x<2) = 0.9731416

Calculated by using the weighted mean of the values.

The reliability of the delivery vehicles in days:

355.1967 or at least 355 days

For 21 drivers for 1560 days:

Table 12: x drivers available for x days

	Number of drivers available	Number of days
1	21	**1458**
2	20	**95**
3	19	**6**
4	18	**1**

Using the binomial formula:

N = 21 for 8 hours/day

- P(x<3) = 0.07408802
- P (4) = 0.08077165
- P (5) = 0.05401372
- P (6) = 0.05498461

The results were obtained by running a function for all the values and then determining at what probability the value was equal to zero. This was then the values entered above. Each bullet represents a line in the table that was used to calculate the probability.

The weighted mean is: $\frac{x11*1458+x22*95+x33*6+x44*1}{1560}$

W = 0.07440558

This value is used to calculate the reliable delivery probability.

The reliability of the delivery drivers:

P(x<4) = 0.9830687

Calculated by using the weighted mean od the values.

The reliability of the delivery drivers in days:

358.8201 or at least 358 days.

Total reliability:

Reliability = reliability of vehicle * reliable of drivers

Reliability = 0.9731416 * 0.9830687

Reliability = 0.956665

Total reliable days:

349.1827 or at least 349 days

The number of vehicles is now changed to 22:

We assume that the probability of failure stays the same for vehicles and that we are using weighted average.

By using binomial equation:

- P (22) = 0.4962558
- P (20) = 0.3533047
- P (19) = 0.1200494
- P (18) = 0.0258994
- P (17) = 0.003981113

New reliability:

The sum of all probabilities = 0.9994903

The new reliable delivery days for vehicles is:

364.814 or at least 364 days which means 365 days of the year.

Conclusion

The data was cleared of all NA instances and then stored as “Valid data”. The charts created with the valid sales data was used to gain an understanding of what is happening within the data and what possible correlations are. This was used to create x-bar and s-bar charts that showcase the control limits for all the different classes.

After evaluating the x-graphs and s-graphs of all the classes it was determined that the Household, Luxury, and gifts classes were out of control. Both Household and Gifts have an increase of delivery time over time. A possible reason could be that theses class of products are not bought as frequently and thus are not stock in large quantities, making the delivery time longer. But the reason for the increase must be investigated. The luxury class has a decrease in delivery time which is good for the business but the reason for this must still be determined and investigated. The probability of making a Type I error is smaller than Type II. This could lead to the SPC showing the delivery time is fine but in reality, it is not. The company must thus make sure not to assume products are delivered on time.

Overall the analysis has given very helpful in sites that can help the company improve.

References

Hessing, T. (2021) *Process capability (CP & CPK)*, *Six Sigma Study Guide*. Available at: https://sixsigmastudyguide.com/process-capability-cp-cpk/ (Accessed: October 18, 2022).

Gysbert, T. (2022) "QA344 Statistics." Stellenbosch: Stellenbosch University.

Appendix

```
#23539321 KR Kritzinger

##################part 1##########

salesTable2022 <- read_excel("stellenbosch/3de jaar/werk/semester 2/Quality assurance/ECSA Project/salesTable2022.xlsx")

library(dplyr)
library(ggplot2)
library(tidyverse)
library(qcc)

df<- salesTable2022 #create a variable to store the data imported.

 #this removes all the values from the dataset that has NA values.

  valid_data<-df[which(!is.na(df[,5])),]

  #add the numbers to n
  nrow(  valid_data) #179983
  t <- c(1:179983)
  bind_n <- cbind(t,valid_data)
  head(bind_n)
  new_n_valid_data<-as.data.frame(bind_n)

  write.csv(new_n_valid_data,  file = "stellenbosch/3de jaar/werk/semester 2/Quality assurance/ECSA Project/valid_data.csv",col.names = TRUE, row.names = TRUE) #this takes the variable that contains all the valid data and creates and excel file for the data.
```

```
# Extracts all the data containing NA values and puts it into a variable.

which(is.na(df[,5]))
d_mat <- as.data.frame(df)
d_no <- na.omit(d_mat)
which(is.na(d_no))

incomplete_data<-df[which(is.na(df[,5])),]

nrow(incomplete_data)  #17
r <- c(1:17)
bind_m <- cbind(r,incomplete_data)
new_incomplete_data<-as.data.frame(bind_m)

write.csv(new_incomplete_data,  file = "stellenbosch/3de  jaar/werk/semester  2/Quality
assurance/ECSA Project/incomple_data.csv",col.names  = TRUE, row.names = TRUE) # Creates an
excel file for the variable containing all the NA values.

ordData<-valid_data[base::order(valid_data$Year,  valid_data$Month,  valid_data$Day,
valid_data$X),] #orders the data.

##################Part 2#################

ggplot(valid_data,aes(x=Price))  +
  geom_histogram(color = "black", fill = "DarkOrange",binwidth  = 1000) +
  scale_x_continuous(  breaks = seq(0,  120000, 5000))  +
  theme(axis.text.x = element_text(angle  = 90)) +
```

```
  xlab("Price") + ylab("Count")
```

#the data is skewed exponentially, most of the data is to the left hand side of the graph.meaning the most items sold is not highly priced. And that people will rather buy products that has a lower price.

```
summary(valid_data$Price)

min_p<- -588.8
Q1_p<- 482.3
med_p <- 2259.6
mean_p <- 12293.7
Q3_p <- 15270.7
max_P <- 116619.0
sd_p<- sd(valid_data$Price) #20888.97

ggplot(valid_data,aes(x=Delivery.time, labels= TRUE)) +
  geom_histogram(color = "black", fill = "DarkOrange",binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 75, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
  xlab("Delivery.time") + ylab("Count")

#the graph has two peaks

summary(valid_data$Delivery.time)

min_dt<- 0.5
Q1_dt <- 3.0
med_dt <- 10.0
mean_dt <- 14.5
Q3_dt<- 18.5
max_dt<-75.0
```

```
sd_dt<- sd(valid_data$Delivery.time)  #13.95608

tech<-valid_data[valid_data$Class=="Technology",]
mean_tech = mean(tech$Delivery.time)
sd_tech<- sd(tech$Delivery.time)
#assuming
USL<- 24 #hours
LSL<- 0

#process capability indices

Cp  <- (USL-LSL)/(6*sd_tech)
Cpu <- (USL- mean_tech)/(3*sd_tech)
Cpl <- (mean_tech  - LSL)/(3*sd_tech)
Cpk <- min(Cpl,Cpu)



ggplot(tech,aes(x=Delivery.time,  labels= TRUE)) +
  geom_histogram(color = "black", fill = "DarkOrange",binwidth  = 1) +
  scale_x_continuous(  breaks = seq(0,  35, 1)) +
  theme(axis.text.x  = element_text(angle  = 90)) +
  ggtitle("Delivery  time of Technology ") +
  xlab("Delivery  time of tech") + ylab("Count")



#graph of the average price per class of item.
valid_data %>%
  group_by(Class) %>%
  summarize(Price  = mean(Price))  %>%
  ggplot(aes(Class, Price))  +
  ggtitle("average price per class") +
```

```
  geom_bar(stat = "identity")


#totals of price per year.
valid_data %>%
  group_by(Year) %>%
  summarize(Price = sum(Price)) %>%
  ggplot(aes(Year, Price)) +
  ggtitle("total price per year") +
 geom_bar(stat = "identity")


#the average delivery time per class
valid_data %>%
  group_by(Class) %>%
  summarize(Delivery.time = mean(Delivery.time)) %>%
  ggplot(aes(Class, Delivery.time)) +
  ggtitle("average delivery time per class") +
  geom_bar(stat = "identity")

#how many customers there are per age group.

ggplot(valid_data,aes(x=AGE, labels= TRUE)) +
  geom_histogram(color = "black", fill = "DarkOrange",binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 110, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
  ggtitle("number of customers per age") +
  xlab("Age") + ylab("Count")

#how many customers bought products because x.
ggplot(valid_data,aes(x=Why.Bought))+
```

```
  geom_bar(stat="count")+
  xlab("why bought") + ylab("count") +
  ggtitle("why people bought products")

#what age group buys more of what products
ggplot(valid_data, aes(x = AGE)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 110, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
  coord_cartesian(c(0,110)) +
  facet_grid(Class~.) +
  ggtitle("Age vs class") +
  xlab("age") + ylab("Count")

#price vs class
ggplot(valid_data, aes(x = Price)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 1000) +
  scale_x_continuous( breaks = seq(0, 120000, 5000)) +
  theme(axis.text.x = element_text(angle = 90)) +
  coord_cartesian(c(0,120000)) +
  facet_grid(Class~.) +
  ggtitle("Price vs class") +
  xlab("price") + ylab("Count")

#delivery time vs class
ggplot(valid_data, aes(x = Delivery.time)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 75, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
  coord_cartesian(c(0,80)) +
  facet_grid(Class~.) +
```

```
  ggtitle("delivery time vs class") +
  xlab("delivery time") + ylab("Count")

#year vs class
ggplot(valid_data, aes(x = Year)) +
  geom_histogram(color = "grey", fill = "lightblue", binwidth = 6) +
  scale_x_continuous(breaks = seq(2021, 2029, 1)) +
  coord_cartesian(c(2021,2029)) +
  facet_grid(Class~.) +
  xlab("Year") + ylab("Count") +
  theme(axis.title.x = element_text(size = 15)) +
  theme(axis.title.y = element_text(size = 15)) +
  ggtitle("Year vs Class of product") +
  theme(plot.title = element_text(size = 16))



#price vs why bought
ggplot(valid_data, aes(x = Price)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 1000) +
  scale_x_continuous( breaks = seq(0, 120000, 5000)) +
  theme(axis.text.x = element_text(angle = 90)) +
  coord_cartesian(c(0,120000)) +
  facet_grid(Why.Bought~.) +
  ggtitle("Price vs why bought") +
  xlab("price") + ylab("Count")

#delivery time vs why bought
ggplot(valid_data, aes(x = Delivery.time)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 75, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
```

```
  coord_cartesian(c(0,80)) +
  facet_grid(Why.Bought~.) +
  ggtitle("delivery time vs Why bought") +
  xlab("delivery time") + ylab("Count")



# age vs why bought
ggplot(valid_data, aes(x = AGE)) +
  geom_histogram(color = "black", fill = "DarkOrange", binwidth = 5) +
  scale_x_continuous( breaks = seq(0, 110, 5)) +
  theme(axis.text.x = element_text(angle = 90)) +
  coord_cartesian(c(0,110)) +
  facet_grid(Why.Bought~.) +
  ggtitle("Age vs why bought") +
  xlab("age") + ylab("Count")

###################PART 3


#######part 3######

####3.1#######

  # samples

#########technology
technology <- ordData[ordData$Class == "Technology",]
technology_dt <- as.data.frame(technology$Delivery.time)  #delivery time extracted

tech_s<- technology_dt[1:(30*15),]  #creates a sample
techs1<- matrix(tech_s, 30, 15, TRUE) #sample 1
```

```
qcc(techs1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3)  #plots xbar from sample 1

qcc(techs1, type="S") #plots s graph from sample 1




# x_chart

 UCL_tx = 22.9731

  U2S_tx = CL_tx + (UCL_tx - CL_tx)/(3)*2

  U1s_tx = CL_tx + (UCL_tx - CL_tx)/(3)

  CL_tx = 20.37444

  L1s_tx = CL_tx - (UCL_tx - CL_tx)/(3)

  L2s_tx =CL_tx - (UCL_tx - CL_tx)/(3)*2

   LCL_tx =17.77579


# s_chart


   UCL_ts =5.179912

   U2S_ts =CL_ts + (UCL_ts - CL_ts)/(3)*2

   U1s_ts = CL_ts + (UCL_ts - CL_ts)/(3)

   CL_ts = 3.295528

   L1s_ts = CL_ts - (UCL_ts - CL_ts)/(3)

   L2s_ts =CL_ts - (UCL_ts - CL_ts)/(3)*2

   LCL_ts =1.411143






#########clothing

clot <- ordData[ordData$Class == "Clothing",]
```

```
clot_dt <- as.data.frame(clot$Delivery.time) #delivery time extracted


clot_s<- clot_dt[1:(30*15),] #creates a sample

clots1<- matrix(clot_s, 30, 15, TRUE) #sample 1


qcc(clots1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3) #plots xbar from sample 1

qcc(clots1, type="S") #plots s graph from sample 1


# x_chart

UCL_cx = 9.404681

U2S_cx = CL_cx + (UCL_cx - CL_cx)/(3)*2

U1s_cx = CL_cx + (UCL_cx - CL_cx)/(3)

CL_cx = 8.97

L1s_cx = CL_cx - (UCL_cx - CL_cx)/(3)

L2s_cx =CL_cx - (UCL_cx - CL_cx)/(3)*2

LCL_cx =8.535319


# s_chart


UCL_cs =0.8664496

U2S_cs =CL_cs + (UCL_cs - CL_cs)/(3)*2

U1s_cs = CL_cs + (UCL_cs - CL_cs)/(3)

CL_cs = 0.5512465

L1s_cs = CL_cs - (UCL_cs - CL_cs)/(3)

L2s_cs =CL_cs - (UCL_cs - CL_cs)/(3)*2

LCL_cs =0.2360435



#########household

hh <- ordData[ordData$Class == "Household",]

hh_dt <- as.data.frame(hh$Delivery.time) #delivery time extracted
```

```
hh_s<- hh_dt[1:(30*15),]  #creates a sample

hhs1<- matrix(hh_s, 30, 15, TRUE) #sample 1


qcc(hhs1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3)  #plots xbar from sample 1

qcc(hhs1, type="S") #plots s graph from sample 1


# x_chart

UCL_hx = 50.24618

U2S_hx = CL_hx + (UCL_hx - CL_hx)/(3)*2

U1s_hx = CL_hx + (UCL_hx - CL_hx)/(3)

CL_hx = 46.56222

L1s_hx = CL_hx - (UCL_hx - CL_hx)/(3)

L2s_hx =CL_hx - (UCL_hx - CL_hx)/(3)*2

LCL_hx =42.87826


# s_chart


UCL_hs =7.343248

U2S_hs =CL_hs + (UCL_hs - CL_hs)/(30)*2

U1s_hs = CL_hs + (UCL_hs - CL_hs)/(3)

CL_hs = 4.67187

L1s_hs = CL_hs - (UCL_hs - CL_hs)/(3)

L2s_hs =CL_hs - (UCL_hs - CL_hs)/(3)*2

LCL_hs =2.000493



#########luxury

lux <- ordData[ordData$Class == "Luxury",]

lux_dt <- as.data.frame(lux$Delivery.time)  #delivery time extracted
```

```
lux_s<- lux_dt[1:(30*15),]  #creates a sample
luxs1<- matrix(lux_s, 30, 15, TRUE) #sample 1

qcc(luxs1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3)  #plots xbar from sample 1
qcc(luxs1, type="S") #plots s graph from sample 1

# x_chart
UCL_lx = 5.493524
U2S_lx = CL_lx + (UCL_lx - CL_lx)/(3)*2
U1s_lx = CL_lx + (UCL_lx - CL_lx)/(3)
CL_lx = 4.735556
L1s_lx = CL_lx - (UCL_lx - CL_lx)/(3)
L2s_lx =CL_lx - (UCL_lx - CL_lx)/(3)*2
LCL_lx =3.977587

# s_chart

UCL_ls =1.51086
U2S_ls =CL_ls + (UCL_ls - CL_ls)/(3)*2
U1s_ls = CL_ls + (UCL_ls - CL_ls)/(3)
CL_ls = 0.9612289
L1s_ls = CL_ls - (UCL_ls - CL_ls)/(3)
L2s_ls =CL_ls - (UCL_ls - CL_ls)/(3)*2
LCL_ls =0.4115978


########food
food <- ordData[ordData$Class == "Food",]
food_dt <- as.data.frame(food$Delivery.time)  #delivery time extracted

food_s<- food_dt[1:(30*15),] #creates a sample
```

```
foods1<- matrix(food_s, 30, 15, TRUE) #sample 1

qcc(foods1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3) #plots xbar from sample 1
qcc(foods1, type="S") #plots s graph from sample 1

# x_chart
UCL_fx = 2.70933
U2S_fx = CL_fx + (UCL_fx - CL_fx)/(3)*2
U1s_fx = CL_fx + (UCL_fx - CL_fx)/(3)
CL_fx = 2.49
L1s_fx = CL_fx - (UCL_fx - CL_fx)/(3)
L2s_fx =CL_fx - (UCL_fx - CL_fx)/(3)*2
LCL_fx =2.27067

# s_chart

UCL_fs =0.4371911
U2S_fs =CL_fs + (UCL_fs - CL_fs)/(3)*2
U1s_fs = CL_fs + (UCL_fs - CL_fs)/(3)
CL_fs = 0.2781467
L1s_fs = CL_fs - (UCL_fs - CL_fs)/(3)
L2s_fs =CL_fs - (UCL_fs - CL_fs)/(3)*2
LCL_fs =0.1191023


########gifts
gift <- ordData[ordData$Class == "Gifts",]
gift_dt <- as.data.frame(gift$Delivery.time) #delivery time extracted

gift_s<- gift_dt[1:(30*15),] #creates a sample
gifts1<- matrix(gift_s, 30, 15, TRUE) #sample 1
```

```
qcc(gifts1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3)  #plots xbar from sample 1

qcc(gifts1, type="S")  #plots s graph from sample 1


# x_chart

UCL_gx = 9.487909

U2S_gx = CL_gx + (UCL_gx - CL_gx)/(3)*2

U1s_gx = CL_gx + (UCL_gx - CL_gx)/(3)

CL_gx = 8.361111

L1s_gx = CL_gx - (UCL_gx - CL_gx)/(3)

L2s_gx =CL_gx - (UCL_gx - CL_gx)/(3)*2

LCL_gx =7.234213


# s_chart


UCL_gs =2.246048

U2S_gs =CL_gs + (UCL_gs - CL_gs)/(3)*2

U1s_gs = CL_gs + (UCL_gs - CL_gs)/(3)

CL_gs = 1.428965

L1s_gs = CL_gs - (UCL_gs - CL_gs)/(3)

L2s_gs =CL_gs - (UCL_gs - CL_gs)/(3)*2

LCL_gs =0.6118823



#######sweets

sweets <- ordData[ordData$Class == "Sweets",]

sweets_dt <- as.data.frame(sweets$Delivery.time)  #delivery time extracted


sweets_s<- sweets_dt[1:(30*15),]  #creates a sample

Sweets1<- matrix(sweets_s, 30, 15, TRUE) #sample 1
```

```
qcc(Sweets1, type="xbar", std.dev="UWAVE-SD",ordsigmas=3)  #plots xbar from sample 1
qcc(Sweets1, type="S") #plots s graph from sample 1


# x_chart
UCL_sx = 2.896798
U2S_sx = CL_sx + (UCL_sx - CL_sx)/(3)*2
U1s_sx = CL_sx + (UCL_sx - CL_sx)/(3)
CL_sx = 2.477778
L1s_sx = CL_sx - (UCL_sx - CL_sx)/(3)
L2s_sx =CL_sx - (UCL_sx - CL_sx)/(3)*2
LCL_sx =2.058758


# s_chart


UCL_ss =0.8352331
U2S_ss =CL_ss + (UCL_ss - CL_ss)/(3)*2
U1s_ss = CL_ss + (UCL_ss - CL_ss)/(3)
CL_ss = 0.5313862
L1s_ss = CL_ss - (UCL_ss - CL_ss)/(3)
L2s_ss =CL_ss - (UCL_ss - CL_ss)/(3)*2
LCL_ss =0.2275393



# X-CHART
x_data = c(UCL_tx,U2S_tx,U1s_tx,CL_tx,L1s_tx,L2s_tx,LCL_tx,
        UCL_cx,U2S_cx,U1s_cx,CL_cx,L1s_cx,L2s_cx,LCL_cx,
        UCL_hx,U2S_hx,U1s_hx,CL_hx,L1s_hx,L2s_hx,LCL_hx,
        UCL_lx,U2S_lx,U1s_lx,CL_lx,L1s_lx,L2s_lx,LCL_lx,
        UCL_fx,U2S_fx,U1s_fx,CL_fx,L1s_fx,L2s_fx,LCL_fx,
        UCL_gx,U2S_gx,U1s_gx,CL_gx,L1s_gx,L2s_gx,LCL_gx,
        UCL_sx,U2S_sx,U1s_sx,CL_sx,L1s_sx,L2s_sx,LCL_sx)
```

```
x_chart<- matrix(x_data, nrow = 7, ncol = 7, byrow = TRUE)

rownames(x_chart)<- c("Technology","Clothing","Household","Luxury","Food","Gifts","Sweets")
colnames(x_chart)<- c("UCL","U2Sigma","U1Sigma","CL","L1Sigma","L2Sigma","LCL")


#S-Chart
s_data = c(UCL_ts,U2S_ts,U1s_ts,CL_ts,L1s_ts,L2s_ts,LCL_ts,
        UCL_cs,U2S_cs,U1s_cs,CL_cs,L1s_cs,L2s_cs,LCL_cs,
        UCL_hs,U2S_hs,U1s_hs,CL_hs,L1s_hs,L2s_hs,LCL_hs,
        UCL_ls,U2S_ls,U1s_ls,CL_ls,L1s_ls,L2s_ls,LCL_ls,
        UCL_fs,U2S_fs,U1s_fs,CL_fs,L1s_fs,L2s_fs,LCL_fs,
        UCL_gs,U2S_gs,U1s_gs,CL_gs,L1s_gs,L2s_gs,LCL_gs,
        UCL_ss,U2S_ss,U1s_ss,CL_ss,L1s_ss,L2s_ss,LCL_ss)

s_chart<- matrix(s_data, nrow = 7, ncol = 7, byrow = TRUE)

rownames(s_chart)<- c("Technology","Clothing","Household","Luxury","Food","Gifts","Sweets")
colnames(s_chart)<- c("UCL","U2Sigma","U1Sigma","CL","L1Sigma","L2Sigma","LCL")

write.csv(x_chart, file = "stellenbosch/3de jaar/werk/semester 2/Quality assurance/ECSA Project/x_chart.csv")
        write.csv(s_chart, file = "stellenbosch/3de jaar/werk/semester 2/Quality assurance/ECSA Project/s_chart.csv")


#####3.2#####

UCL_mean<- x_chart[,1]
LCL_mean <- x_chart[,7]
UCL_sd<- s_chart[,1]
```

```
LCL_sd<- s_chart[,7]
CL_mean<- x_chart[,4]
mean_sd<-s_chart[,4]
U2s_sd = s_chart[,2]
U1s_sd = s_chart[,3]
L1s_sd = s_chart[,5]
L2s_sd = s_chart[,6]

  # all the data

##technology
nr_tech<-(nrow(technology_dt)-2)/15
nr_tech

tech_samples<- technology_dt[1:(nr_tech*15),]
Techno<- matrix(tech_samples, nr_tech, 15, TRUE)

qcc(Techno, type="xbar",limits=c(LCL_mean[1],UCL_mean[1]))
qcc(Techno, type="S",limits=c(LCL_sd[1],UCL_sd[1]))

##clothing
nr_clot <- (nrow(clot_dt)-3)/15

clot_samples <- clot_dt[1:(nr_clot*15),]
clothing<- matrix(clot_samples, nr_clot, 15, TRUE)

qcc(clothing, type="xbar",limits=c(LCL_mean[2],UCL_mean[2]),center=CL_mean[2])
qcc(clothing, type="S",limits=c(LCL_sd[2],UCL_sd[2]))

##household
nr_hh <- (nrow(hh_dt)-12)/15
```

```
hh_samples <- hh_dt[1:(nr_hh*15),]
house<- matrix(hh_samples, nr_hh, 15, TRUE)

qcc(house, type="xbar",limits=c(LCL_mean[3],UCL_mean[3]),center=CL_mean[3])
qcc(house, type="S",limits=c(LCL_sd[3],UCL_sd[3]))

##luxury
nr_lux <- (nrow(lux_dt)-4)/15

lux_samples <- lux_dt[1:(nr_lux*15),]
Luxury<- matrix(lux_samples, nr_lux, 15, TRUE)

qcc(Luxury, type="xbar",limits=c(LCL_mean[4],UCL_mean[4]),center=CL_mean[4])
qcc(Luxury, type="S",limits=c(LCL_sd[4],UCL_sd[4]))

##food
nr_fd <- (nrow(food_dt)-13)/15

fd_samples <- food_dt[1:(nr_fd*15),]
fd<- matrix(fd_samples, nr_fd, 15, TRUE)

qcc(fd, type="xbar",limits=c(LCL_mean[5],UCL_mean[5]),center=CL_mean[5])
qcc(fd, type="S",limits=c(LCL_sd[5],UCL_sd[5]))

##gifts
nr_g <- (nrow(gift_dt)-14)/15

g_samples <- gift_dt[1:(nr_g*15),]
gft<- matrix(g_samples, nr_g, 15, TRUE)
```

```
qcc(gft, type="xbar",limits=c(LCL_mean[6],UCL_mean[6]),center=CL_mean[6])
qcc(gft, type="S",limits=c(LCL_sd[6],UCL_sd[6]))

##sweets
nr_s <- (nrow(sweets_dt)-10)/15

swt_samples <- sweets_dt[1:(nr_s*15),]
swt<- matrix(swt_samples,  nr_s, 15, TRUE)

qcc(swt, type="xbar",limits=c(LCL_mean[7],UCL_mean[7]),center=CL_mean[7])
qcc(swt, type="S",limits=c(LCL_sd[7],UCL_sd[7]))



 #################### part 4 ########################


####### 4.1 A ########

#######x values#####

##technology
techmean <- apply(Techno,  1, mean)

which(techmean  >UCL_mean[1] | techmean<LCL_mean[1])
length(which(techmean  >UCL_mean[1] | techmean<LCL_mean[1]))

##clothing
clotmean <- apply(clothing,  1, mean)
```

```
which(clotmean >UCL_mean[2] | clotmean<LCL_mean[2])

length(which(clotmean  >UCL_mean[2] | clotmean<LCL_mean[2]))


##Household

housemean <- apply(house, 1, mean)


which(housemean  >UCL_mean[3] | housemean<LCL_mean[3])

length(which(housemean  >UCL_mean[3] | housemean<LCL_mean[3]))  #its clear household is out of
control


##Luxury

luxmean <- apply(Luxury, 1, mean)


which(luxmean  >UCL_mean[4] | luxmean<LCL_mean[4])

length(which(luxmean  >UCL_mean[4] | luxmean<LCL_mean[4])) #it's clear luxury is out of control


##food


foodmean <- apply(fd, 1, mean)


which(foodmean >UCL_mean[5] | foodmean<LCL_mean[5])

length(which(foodmean  >UCL_mean[5] | foodmean<LCL_mean[5]))


##gifts

giftmean <- apply(gft, 1, mean)


all<-c(which(giftmean >UCL_mean[6] | giftmean<LCL_mean[6]))

all

last_three<-c(all[2288],all[2289],all[2290]) #get the last three values

last_three
```

```
length(which(giftmean  >UCL_mean[6] | giftmean<LCL_mean[6])) #it's clear gifts is out of contol


##sweets
sweetsmean <- apply(swt, 1, mean)


which(sweetsmean  >UCL_mean[7] | sweetsmean<LCL_mean[7])
length(which(sweetsmean  >UCL_mean[7] | sweetsmean<LCL_mean[7]))


 control_data<-c( 17,17,396,434,3,2290,4,
             37,455,252,142,75,213,942,
             398,702,387,171,432,216,1243,
             483,1152,629,184,"NA",218,"NA",
             1872,1677,1335,789,"NA",2607,"NA",
             2009,1723,1336,790,"NA",2608,1294,
             2071,1724,1337,791,633,2609,1358)



control_table<- matrix(control_data, nrow=7, ncol= 7, byrow= FALSE)


rownames(control_table)<-
c("Technology","Clothing","Household","Luxury","Food","Gifts","Sweets")
colnames(control_table)<- c("Total","1st sample","2nd sample","3rd sample","3rd last ","2nd
last","last")


write.csv(control_table, file = "stellenbosch/3de jaar/werk/semester 2/Quality assurance/ECSA
Project/control_table.csv")




##########first 3 and last 3#################
```

#####household

##first 3

```
qcc(house[c(250:630),],  type="xbar",limits=c(LCL_mean[3],UCL_mean[3]),center=CL_mean[3])
```

##last 3

```
qcc(house[c(1330:1337),],  type="xbar",limits=c(LCL_mean[3],UCL_mean[3]),center=CL_mean[3])
```

#####Luxury

##first 3

```
qcc(Luxury[c(140:184),],  type="xbar",limits=c(LCL_mean[4],UCL_mean[4]),center=CL_mean[4])
```

##last 3

```
qcc(Luxury[c(785:791),],  type="xbar",limits=c(LCL_mean[4],UCL_mean[4]),center=CL_mean[4])
```

#####Gifts

##first 3

```
qcc(gft[c(212:218),], type="xbar",limits=c(LCL_mean[6],UCL_mean[6]),center=CL_mean[6])
```

##last 3

```
qcc(gft[c(2600:2609),], type="xbar",limits=c(LCL_mean[6],UCL_mean[6]),center=CL_mean[6])




#######s value#####


##technology
techs <- apply(Techno, 1, sd)

which(techs >UCL_sd[1] | techs<LCL_sd[1])
length(which(techs >UCL_sd[1] | techs<LCL_sd[1]))

##clothing
clots <- apply(clothing, 1, sd)

which(clots >UCL_sd[2] | clots<LCL_sd[2])
length(which(clots >UCL_sd[2] | clots<LCL_sd[2]))

##Household
houses <- apply(house, 1, sd)

which(houses >UCL_sd[3] | houses<LCL_sd[3])
length(which(houses >UCL_sd[3] | houses<LCL_sd[3])) #its clear household is out of control

##Luxury
luxs <- apply(Luxury, 1, sd)

which(luxs >UCL_sd[4] | luxs<LCL_sd[4])
length(which(luxs >UCL_sd[4] | luxs<LCL_sd[4])) #it's clear luxury is out of control
```

```
##food

foodss <- apply(fd,  1, sd)

which(foodss  >UCL_sd[5] | foodss<LCL_sd[5])
length(which(foodss  >UCL_sd[5] | foodss<LCL_sd[5]))

##gifts
giftss <- apply(gft, 1, sd)

alls<-c(which(giftss >UCL_sd[6] | giftss<LCL_sd[6]))
alls

length(alls)  #it's clear gifts is out of contol

##sweets
sweetss <- apply(swt,  1, sd)

which(sweetss  >UCL_sd[7] | sweetss<LCL_sd[7])
length(which(sweetss  >UCL_sd[7] | sweetss<LCL_sd[7]))

control_data_s<-c( 16,98,49,4,19,8,1,
          129,289,65,103,19,193,8,
          230,530,89,254,57,746,"NA",
          251,780,147,"NA",96,1342,"NA",
          2095,1754,1271,"NA",1422,1855,"NA",
          2290,1756,1290,543,1429,2493,"NA",
          2400,1757,1323,600,1553,2576,"NA")


control_table_s<- matrix(control_data_s, nrow=7, ncol= 7, byrow= FALSE)
```

```
rownames(control_table_s)<-
c("Technology","Clothing","Household","Luxury","Food","Gifts","Sweets")

colnames(control_table_s)<-  c("Total","1st sample","2nd  sample","3rd  sample","3rd  last ","2nd
last","last")


write.csv(control_table_s,  file = "stellenbosch/3de  jaar/werk/semester  2/Quality assurance/ECSA
Project/control_table_s.csv")



####### 4.1 B ########



tech_sd<- apply(Techno,1,sd)

clothing_sd<- apply(clothing,1,sd)

household_sd<-  apply(house,1,sd)

luxury_sd<- apply(Luxury,1,sd)

food_sd <- apply(fd,1,sd)

gifts_sd <- apply(gft,1,sd)

sweets_sd <- apply(swt,1,sd)



##technology

up_tech_new <-  mean_sd[1] + ((UCL_sd[1]-mean_sd[1])/3)*0.4

low_tech_new <- mean_sd[1] - ((UCL_sd[1]-mean_sd[1])/3)*(0.3)


tech_index_limit  <- which(tech_sd<up_tech_new  & tech_sd>low_tech_new)

max_tech<- max(diff(which(diff(tech_index_limit)!=1)))

tech_index_limit[diff(which(diff(tech_index_limit)!=1))==max_tech]


##clothing

up_cloth_new  <- mean_sd[2]  + ((UCL_sd[2]-mean_sd[2])/3)*0.4
```

```
low_cloth_new <- mean_sd[2]  - ((UCL_sd[2]-mean_sd[2])/3)*0.3

cloth_index_limit  <- which(clothing_sd<up_cloth_new  & clothing_sd>low_cloth_new)
max_cloth<- max(diff(which(diff(cloth_index_limit)!=1)))
which(diff(which(diff(cloth_index_limit)!=1))==max_cloth)

##house
up_house_new  <- mean_sd[3] + ((UCL_sd[3]-mean_sd[3])/3)*0.4
low_house_new <- mean_sd[3]  - ((UCL_sd[3]-mean_sd[3])/3)*0.3

house_index_limit  <- which(household_sd<up_house_new  & household_sd>low_house_new)
max_house<-  max(diff(which(diff(house_index_limit)!=1)))

##luxury
up_lux_new  <- mean_sd[4]  + ((UCL_sd[4]-mean_sd[4])/3)*0.4
low_lux_new <- mean_sd[4]  - ((UCL_sd[4]-mean_sd[4])/3)*0.3

lux_index_limit  <- which(luxury_sd<up_lux_new  & luxury_sd>low_lux_new)
max_lux<- max(diff(which(diff(lux_index_limit)!=1)))

##food
up_food_new <- mean_sd[5]  + ((UCL_sd[5]-mean_sd[5])/3)*0.4
low_food_new <- mean_sd[5] - ((UCL_sd[5]-mean_sd[5])/3)*0.3

food_index_limit  <- which(food_sd<up_food_new  & food_sd>low_food_new)
max_food<- max(diff(which(diff(food_index_limit)!=1)))

##gifts
up_gifts_new <- mean_sd[6]  + ((UCL_sd[6]-mean_sd[6])/3)*0.4
low_gifts_new <- mean_sd[6]  - ((UCL_sd[6]-mean_sd[6])/3)*0.3
```

```
gifts_index_limit <- which(gifts_sd<up_gifts_new  & gifts_sd>low_gifts_new)
max_gifts<- max(diff(which(diff(gifts_index_limit)!=1)))


##sweets
up_sweets_new <- mean_sd[7] + ((UCL_sd[7]-mean_sd[7])/3)*0.4
low_sweets_new <- mean_sd[7] - ((UCL_sd[7]-mean_sd[7])/3)*0.3

sweets_index_limit <- which(sweets_sd<up_sweets_new  & sweets_sd>low_sweets_new)
max_sweets<- max(diff(which(diff(sweets_index_limit)!=1)))




###### 4.2 #####

#For A: Outside sigma 6 is the error.
A1=(1-pnorm(3))+(pnorm(-3))
A1
#For B: outside of -0.5 and 1.2 sigma
B1=(1-pnorm(0.4))+pnorm(-0.3)
B1




####### 4.3 ####



#need to extract the deliveries thats above 26 hours
lost <- sum(technology$Delivery.time>26)  #currently 1356 lost sales
lost*329 # currently losing R446 124 because of lost sales above 26 hours
mean(technology$Delivery.time)#mean  is currently 20.01095
```

```
x<-(technology[technology$Delivery.time>26,])

max <- max(x$Delivery.time)  #max hours over 26
(max-26)*2.5*nrow(technology)  #to move the whole distribution before 26 will be max R636 072.5

#now need to find the min hours

optimal <- c()
for (i in 1:26){
  newDeliveryTime  <- technology$Delivery.time  -i
  optimal[i] <- sum(newDeliveryTime>26)*329  + length(tech$Delivery.time)*2.5*i
}
plot(optimal, main = "Optimal delivery hours ", xlab = " Delivery hours", ylab = " Total Cost", type = 
"l")
abline(v  = which.min(optimal),  col = "red")
26 - which.min(optimal)
text(x=c(15),y=c(2305000),c("Optimal  hours = 2 h"), col=c("red"))

####### 4.4 ####

Sigma_tech <- (UCL_mean[1]-LCL_mean[1])/6

#probability of sample being inside the limits with mean of 23 and Sigma
pnorm(UCL_mean[1],mean=23,sd=Sigma_tech)-
  pnorm(LCL_mean[1],mean=23,sd=Sigma_tech)
#0.487613
```

```
#rough plot

plot(x=(80:250)/10,  y=dnorm((80:250)/10,mean=23,sd=Sigma_tech),type="l",main=  "Probability of making a type II error",
    xlab= "Average Delivery times")

abline(v=c(UCL_mean[1],LCL_mean[1]),col="red")

text(x=c(LCL_mean[1],UCL_mean[1],22,23.5),y=c(0.4,0.4,0.2,0.15),c("LCL",  "UCL", "Calculated Area =", "48.76%"), col=c("red","red","blue","blue"))



###################Part  5######



#MANOVA TEST PRICE, DELIVERY TIME, AGE OF EACH CLASS

mshapiro.test( )


manova1 <- manova(cbind(Delivery.time,  Price, AGE) ~ Class, data = valid_data)

summary(manova1)

summary.aov(manova1)


# delivery time, price age of why bought


manova2 <- manova(cbind(Delivery.time,  Price, AGE) ~ Why.Bought, data = valid_data)

summary(manova2)

summary.aov(manova2)
```

```
#day, month, year of class
manova3 <- manova(cbind((Day), (Month), (Year)) ~ Class, data = valid_data)
summary(manova3)
summary.aov(manova3)








##########Part 6#######


###### problem 6 ####


k <- 28125
x <- c(0:120)/1000
y <- k*(x-0.06)^2

plot(x = x, y=y, xlab = "Deviation from Normal", ylab = "Cost in $", type = "l",
    main = "Taguchi Loss Function")
abline(v=0.06, col = "blue",lty="dashed")
abline(v=0.02, col = "red",lty="dashed")
text(x=c(0.01),y=c(30),c("LCL"), col=c("Red"))
abline(v=0.1, col = "red",lty="dashed")
text(x=c(0.11), y=c(30),c("UCL"),col="red")

###### problem 7####

k <- 21875
```

```
x <- c(0:120)/1000

y <- k*(x-0.06)^2


plot(x = x, y=y,  xlab = "Deviation from Normal", ylab = "Cost in $", type = "l",

    main = "Taguchi Loss Function")

abline(v=0.06,  col = "blue",lty="dashed")

abline(v=0.02,  col = "red",lty="dashed")

text(x=c(0.01),y=c(30),c("LCL"),  col=c("Red"))

abline(v=0.1,  col = "red",lty="dashed")

text(x=c(0.11),  y=c(30),c("UCL"),col="red")


#### 6.3 #####



##############vehicles#####


#first of all 21 P(x<1)

xx <- (c(0:21)/100)

prob_vehicles1  <- function(x){

  pbinom(1,21,prob  = x,log = FALSE)*1560-(1560  -(190+22+3+1))


}

plot(xx,prob_vehicles1(xx))

vehicles1 <- uniroot(prob_vehicles1,  lower = 0.0001,upper  = 0.2)

x1 <- vehicles1$root



#second 20 cars P(2)


prob_vehicles2  <- function(x){

  dbinom(2,21,prob  = x,log = FALSE)*1560-190
```

```
}
plot(xx,prob_vehicles2(xx))
vehicles2 <- uniroot(prob_vehicles2,  lower = 0.03,upper  = 0.04)
x2 <- vehicles2$root


#third 19 cars P(3)
prob_vehicles3 <- function(x){
  dbinom(3,21,prob  = x,log = FALSE)*1560-22

}
plot(xx,prob_vehicles3(xx))
vehicles3 <- uniroot(prob_vehicles3,  lower = 0.01,upper  = 0.03)
x3 <- vehicles3$root

#fourth 18 cars P(4)
prob_vehicles4 <- function(x){
  dbinom(4,21,prob  = x,log = FALSE)*1560-3

}
plot(xx,prob_vehicles3(xx))
vehicles4 <- uniroot(prob_vehicles4,  lower = 0.01,upper  = 0.03)
x4 <- vehicles4$root

#fifth 17 cars P(5)
prob_vehicles5 <- function(x){
  dbinom(5,21,prob  = x,log = FALSE)*1560-1

}
plot(xx,prob_vehicles5(xx))
```

```
vehicles5 <- uniroot(prob_vehicles5,  lower = 0.0,upper  = 0.04)
x5 <- vehicles5$root



#weighted mean

w1 <- (x1*1344+x2*190+x3*22+x4*3+x5*1)/1560

#expected days of reliable delivery
rel_vehicles <- pbinom(2,21,prob=w1,log=FALSE)
rel_vehicle_days <- rel_vehicles*365

##################drivers##################
######### drivers 21
prob_driver1  <- function(x){
  pbinom(3,21,prob  = x,log = FALSE)*1560-(1560  -(95+6+1))

}
plot(xx,prob_driver1(xx))
driver1 <- uniroot(prob_driver1,  lower = 0.04,upper = 0.55)
x11 <- driver1$root

#second 20
prob_driver2  <- function(x){
  dbinom(4,21,prob  = x,log = FALSE)*1560-95

}

plot(xx,prob_driver2(xx))

driver2 <- uniroot(prob_driver2,  lower = 0.07,upper = 0.09)
```

```
x22 <- driver2$root

#third 19
prob_driver3  <- function(x){
  dbinom(5,21,prob  = x,log = FALSE)*1560-6
}
plot(xx,prob_driver3(xx))
driver3 <- uniroot(prob_driver3,  lower = 0.04,upper = 0.06)
x33 <- driver3$root

#fourth 18
prob_driver4  <- function(x){
  dbinom(6,21,prob  = x,log = FALSE)*1560-1
}
plot(xx,prob_driver4(xx))

driver4 <- uniroot(prob_driver4,  lower = 0.04,upper = 0.06)
x44 <- driver4$root

#weighted mean
w2 <- (1458*x11+x22*95+x33*6+x44*1)/1560

#reliable drivers
reL_drivers  <- pbinom(4,21,prob=w2,log=FALSE)
rel_driver_days  <- reL_drivers*365

#total reliable probability
tot_reliable <- reL_drivers*rel_vehicles

#expected reliable days in year
tot_reliable*365
```

```
#349.1827

###########PART2:############
#INCRREASE AMOUNT OF VEHICLES TO 22

#vehicles

p1 <- dbinom(0,22,prob=w1,log=FALSE)
p2 <- dbinom(1,22,prob=w1,log=FALSE)
p3 <- dbinom(2,22,prob=w1,log=FALSE)
p4 <- dbinom(3,22,prob=w1,log=FALSE)
p5 <- dbinom(4,22,prob=w1,log=FALSE)

p_tot <- p1+p2+p3+p4+p5
0.9994903*365
```