SPC

Quality and Reliability Modelling

Joseph O'Reilly

03 March, 2025

Contents

Theory **1**

Control limits for the Mean Chart 1

Control limits for the Range Charts 1

Standard deviation 1

Tutorial 1 **2**

Tutorial 2 **6**

Data set 6

Tutorial 3 **14**

Tutorial 4 **17**

Tutorial 5 **18**

Tutorial 5 19

institiúid teicneolaíochta dhún dealgan

DUNDALK

INSTITUTE OF TECHNOLOGY

DECLARATION: I declare that:

- This work is entirely my own, and no part of it has been copied from any other person's words or ideas, except as specifically acknowledged through the use of inverted commas and in-text references;
- No part of this assignment has been written for me by any other person except where such collaboration has been authorised by the lecturer concerned;
- I understand that I am bound by DkIT Academic Integrity Policy. I understand that I may be penalised if I have violated the policy in any way;
- I have not used generative artificial intelligence (AI) (e.g. ChatGPT) unless it has been permitted by the lecturer(s) concerned;
- This assignment has not been submitted for any other module at DkIT or any other institution, unless authorised by the relevant Lecturer(s);
- I have read and abided by all of the requirements set down for this assignment.

TYPED SIGNATURE:Joseph O'Reilly

DATE: 24/02/2025

Theory

Control limits for the Mean Chart

$$centreline = \bar{x}$$

$$UCL_x = \bar{x} + A_2\bar{R}$$

$$LCL_x = \bar{x} - A_2\bar{R}$$

Control limits for the Range Charts

$$Centreline = \bar{R}$$

$$UCL_R = D_4\bar{R}$$

$$LCL_R = d_3\bar{R}$$

Standard deviation

$\sigma = \frac{\bar{R}}{d_2}$

The process capability index is found using:

Tutorial 1

Project Aim: Phase I – Determination of Process Limits

5 machine samples are taken each hour for 10 hours and a key characteristic is measured. If the specification limits are 50± 5, what are your conclusions regarding the ability of the process to produce items conforming to specifications?

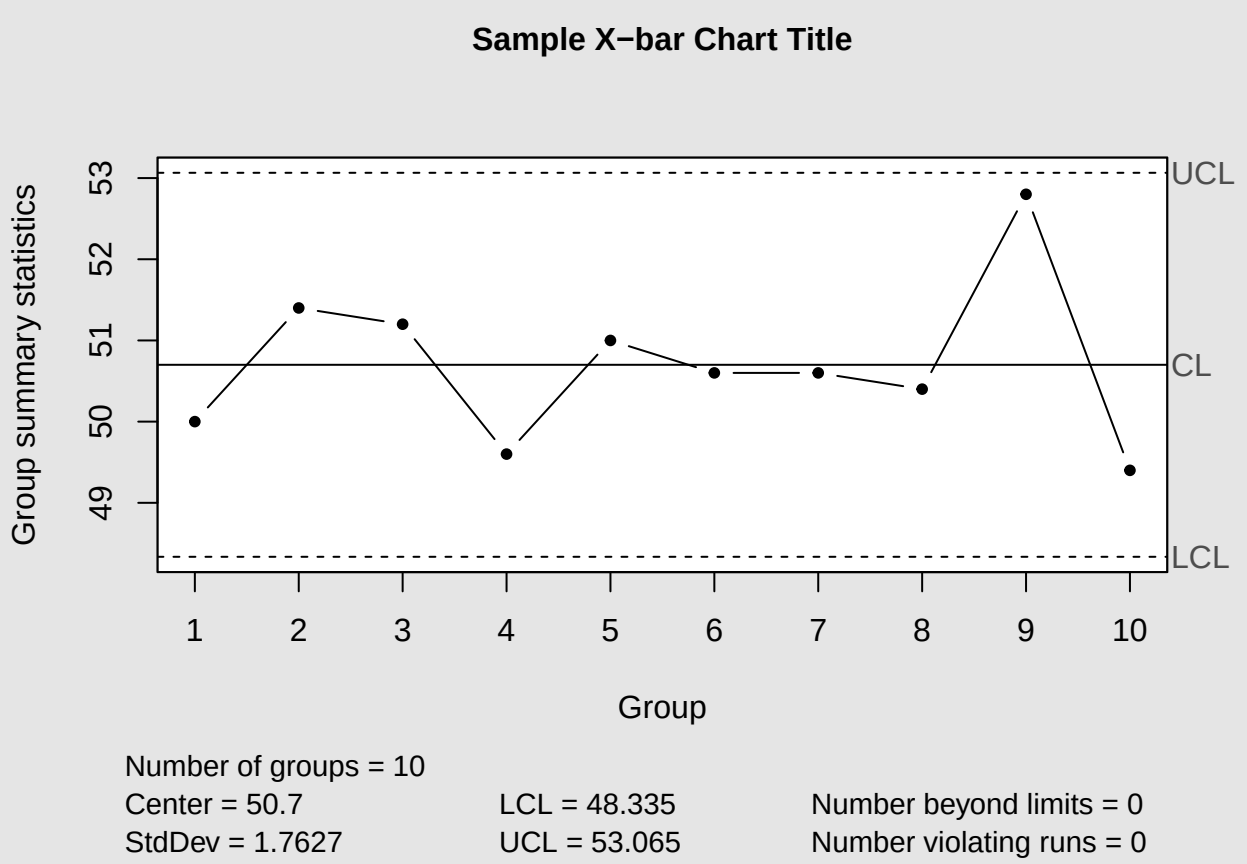
```
library(qcc)
```

```
## Package 'qcc' version 2.7
```

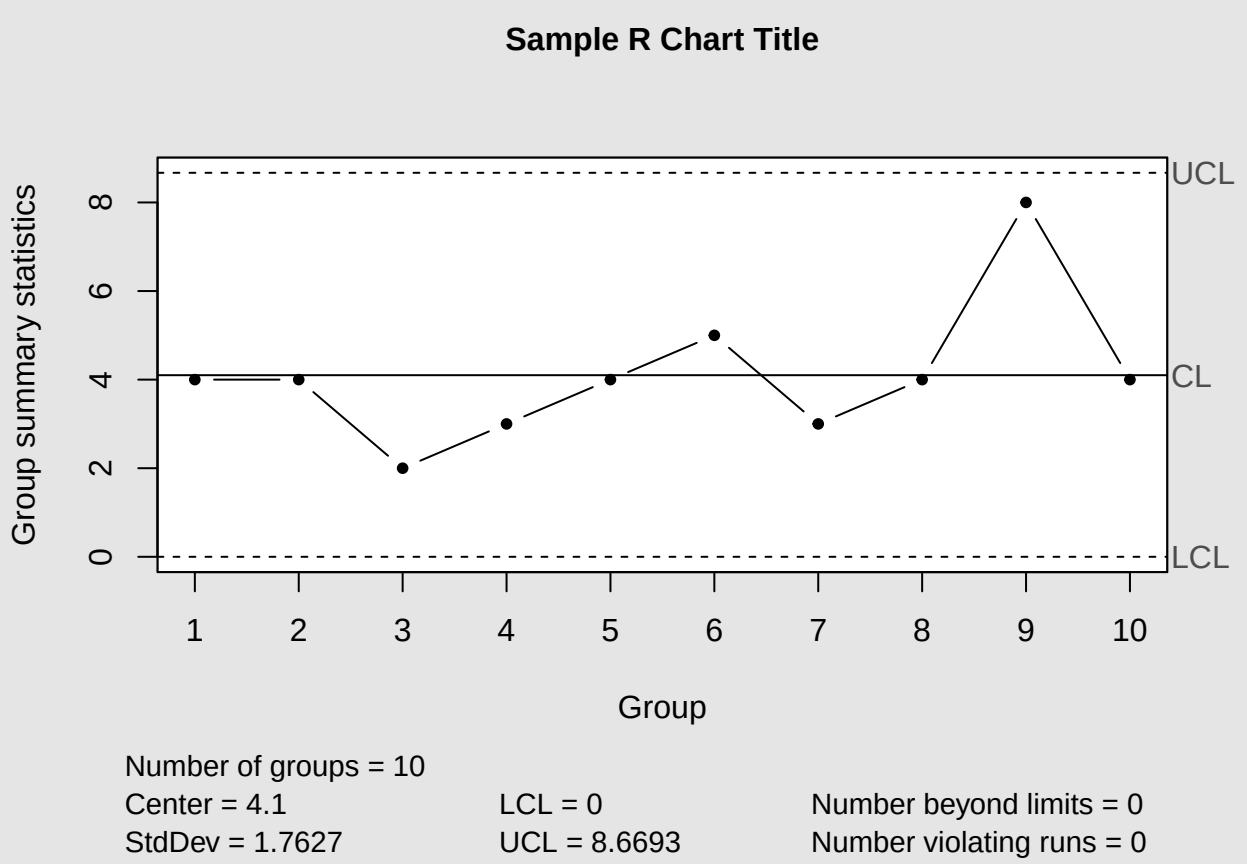
```
## Type 'citation("qcc")' for citing this R package in publications.
```

```
#Chart Data (5 samples (n=5) taken each hour for 10 hours)
chartdata <- read.table(header = FALSE, text = "
49 50 49 49 53
51 53 51 53 49
51 52 52 50 51
48 50 50 51 49
50 51 49 53 52
48 51 51 53 50
51 52 49 51 50
48 50 52 51 51
49 52 53 53 57
48 50 48 52 49")
```

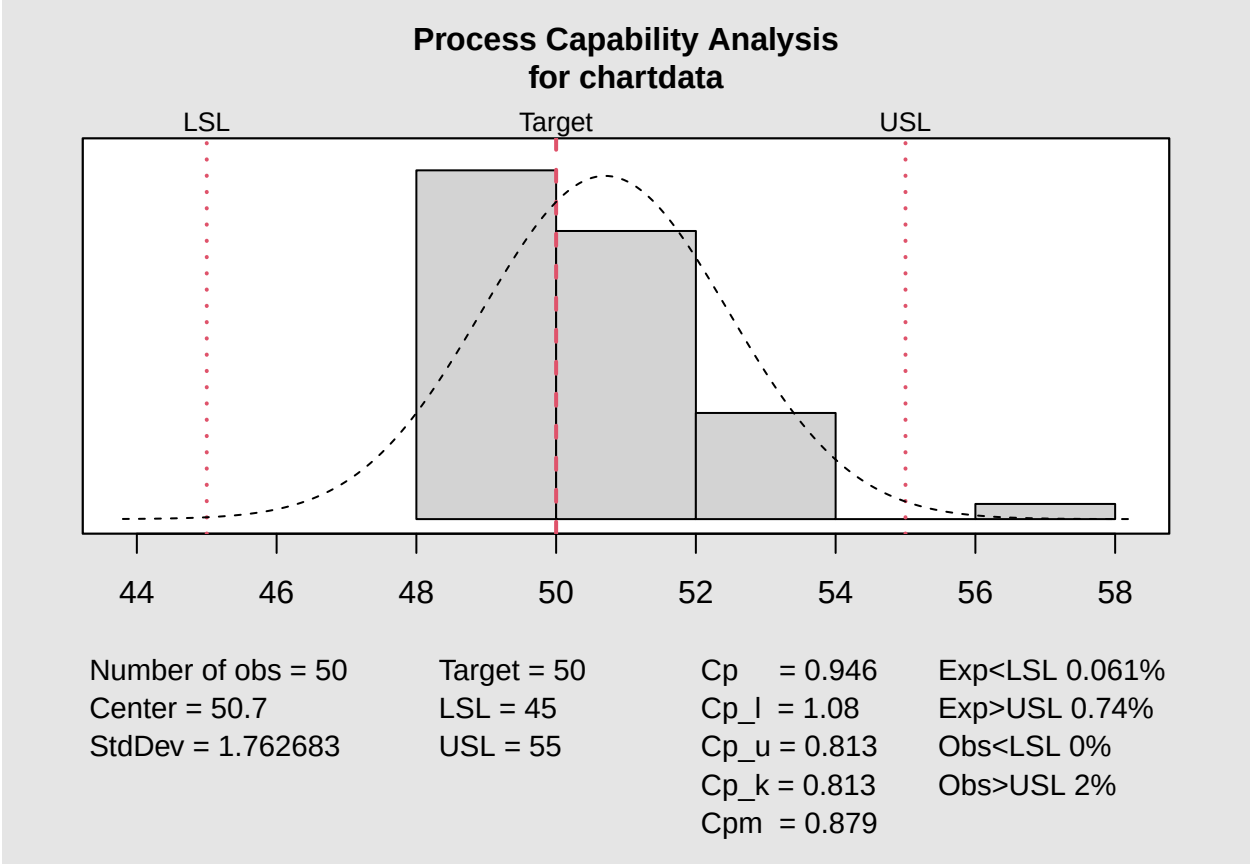
```
# plot xbar chart
xbar <- qcc(data = chartdata,
type = "xbar",
sizes = 5,
title = "Sample X-bar Chart Title",
digits = 5,
plot = TRUE,bg.margin = "white", bg.figure = "white")
```




```r
# R Chart
rbar <- qcc(data = chartdata,
type = "R",
sizes = 5,
title = "Sample R Chart Title",
digits = 5,
plot = TRUE,bg.margin = "white", bg.figure = "white")
```




```
# Determine the Process Capability
process.capability(xbar,spec.limits=c(45,55),target= 50)
```




```
##
## Process Capability Analysis
##
## Call:
## process.capability(object = xbar, spec.limits = c(45, 55), target = 50)
##
## Number of obs = 50           Target = 50
##        Center = 50.7            LSL = 45
##        StdDev = 1.763           USL = 55
##
## Capability indices:
##
##        Value    2.5%   97.5%
## Cp    0.9455  0.7588  1.1319
## Cp_l  1.0779  0.8827  1.2731
## Cp_u  0.8132  0.6574  0.9689
## Cp_k  0.8132  0.6275  0.9988
## Cpm   0.8788  0.6957  1.0615
##
## Exp<LSL 0.061%    Obs<LSL 0%
## Exp>USL 0.74%     Obs>USL 2%
```

Tutorial 2

Phase I and Phase II

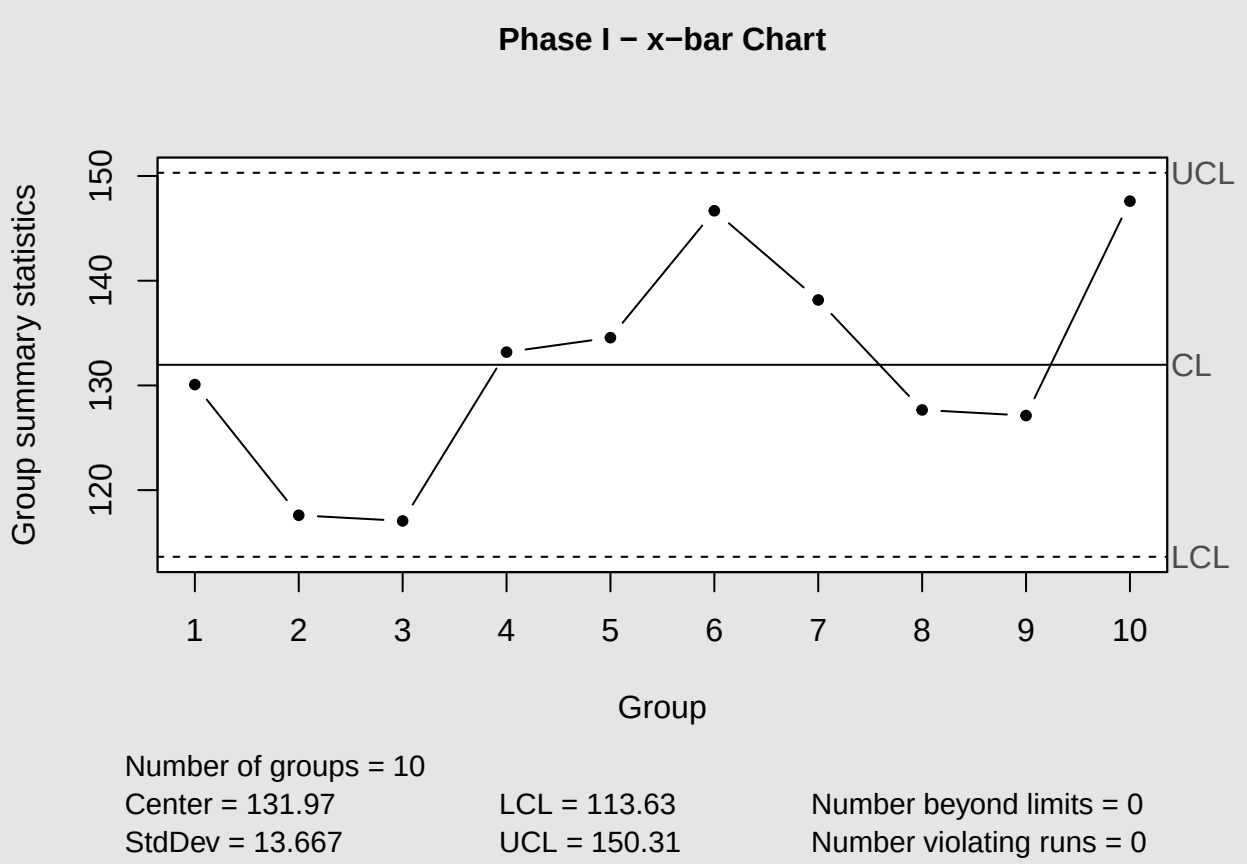
a) Phase I - 5 machine samples are taken each hour for 10 hours and a key characteristic is measured. If the specification limits are 130± 60, what are your conclusions regarding the ability of the process to produce items conforming to specifications?

b) Phase II – begins after we have a "clean" set of process data gathered under stable conditions and representative of in-control process performance. In phase II, we use the control chart to monitor the process by comparing the sample statistic for each successive sample as it is drawn from the process to the control limits.

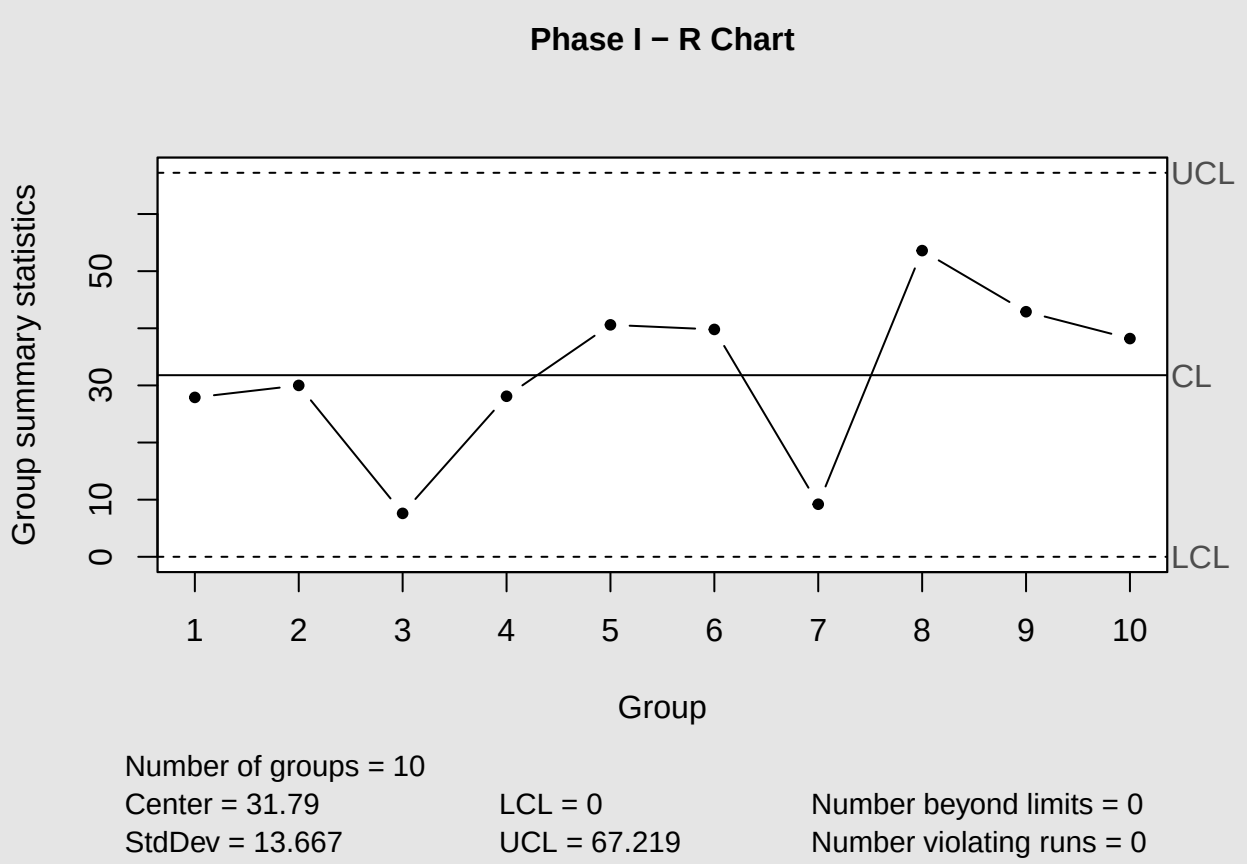
Data set

```
chartdata <- read.table(header = FALSE, text = "
138.1 110.8 138.7 137.4 125.4 #Phase 1 - Calibration
118.5 116.5 130.2 122.6 100.2
120.4 114.3 112.8 118.5 119.3
132.7 151.1 124.0 123.0 135.1
139.6 127.9 151.1 143.7 110.5
125.3 160.2 130.4 152.4 165.1
138.6 139.0 131.9 140.2 141.1
145.2 101.0 154.6 120.2 117.3
125.9 135.3 121.5 147.9 105.0
123.4 150.0 161.6 148.8 154.2
127.5 154.2 139.1 132.2 141.9 # Phase 2
131.5 143.1 118.5 103.2 121.6
111.0 127.3 110.4 121.0 143.9
137.4 148.7 127.4 125.0 127.5
131.0 184.8 182.2 143.3 212.8
181.3 193.2 180.7 169.1 174.3
186.9 180.2 149.2 175.2 185.0
167.8 143.9 157.5 171.8 194.9
178.2 186.7 142.4 159.4 167.6
152.6 143.6 132.8 168.9 177.2")
```

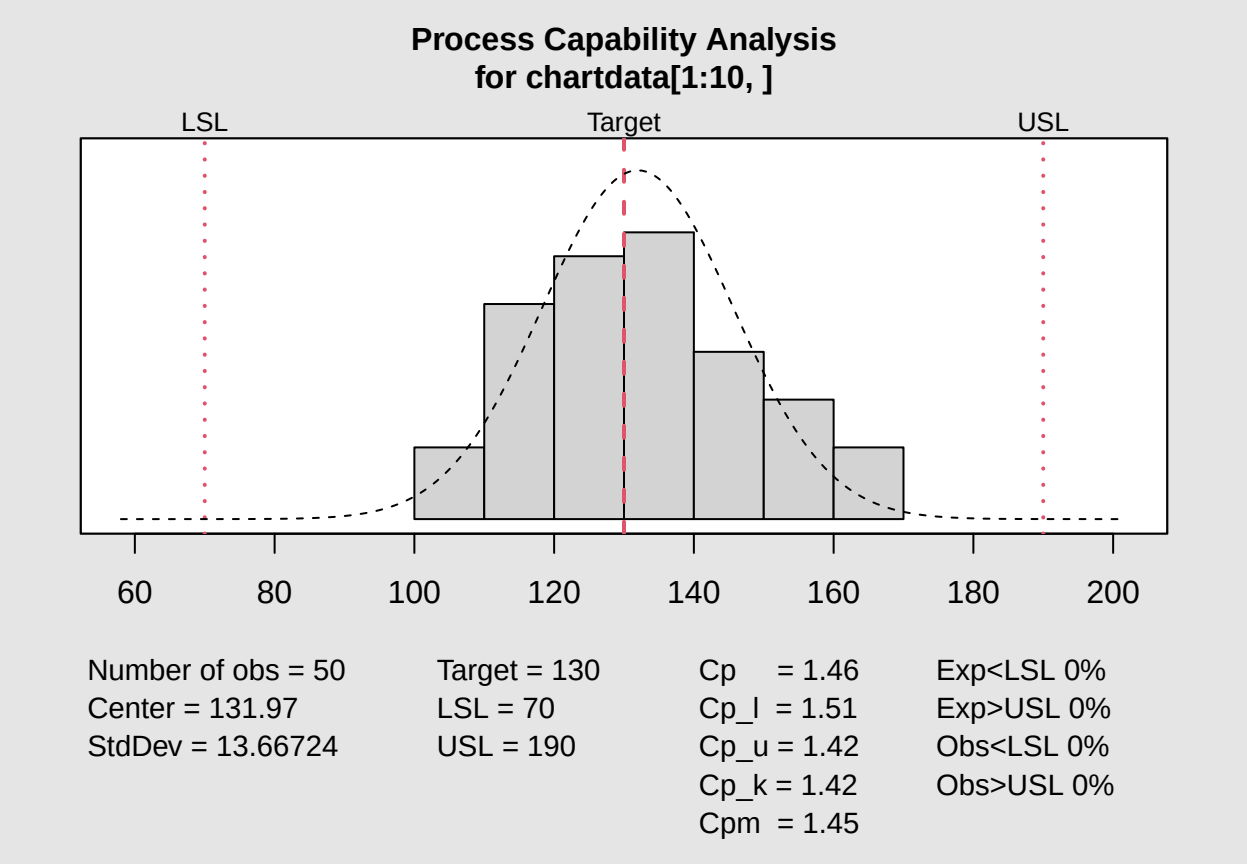
```
library(qcc)
############ Phase I ##########
# Determine Natural Variation of Process
xbar_phase_I<- qcc(data = chartdata[1:10,],
type = "xbar",
sizes = 5,
title = "Phase I - x-bar Chart",
digits = 5,
plot = TRUE,bg.margin = "white", bg.figure = "white")
```




```
rbar_phase_I <- qcc(data = chartdata[1:10,],
type = "R",
sizes = 5,
title = "Phase I - R Chart",
digits = 5,
plot = TRUE, bg.margin = "white", bg.figure = "white")
```



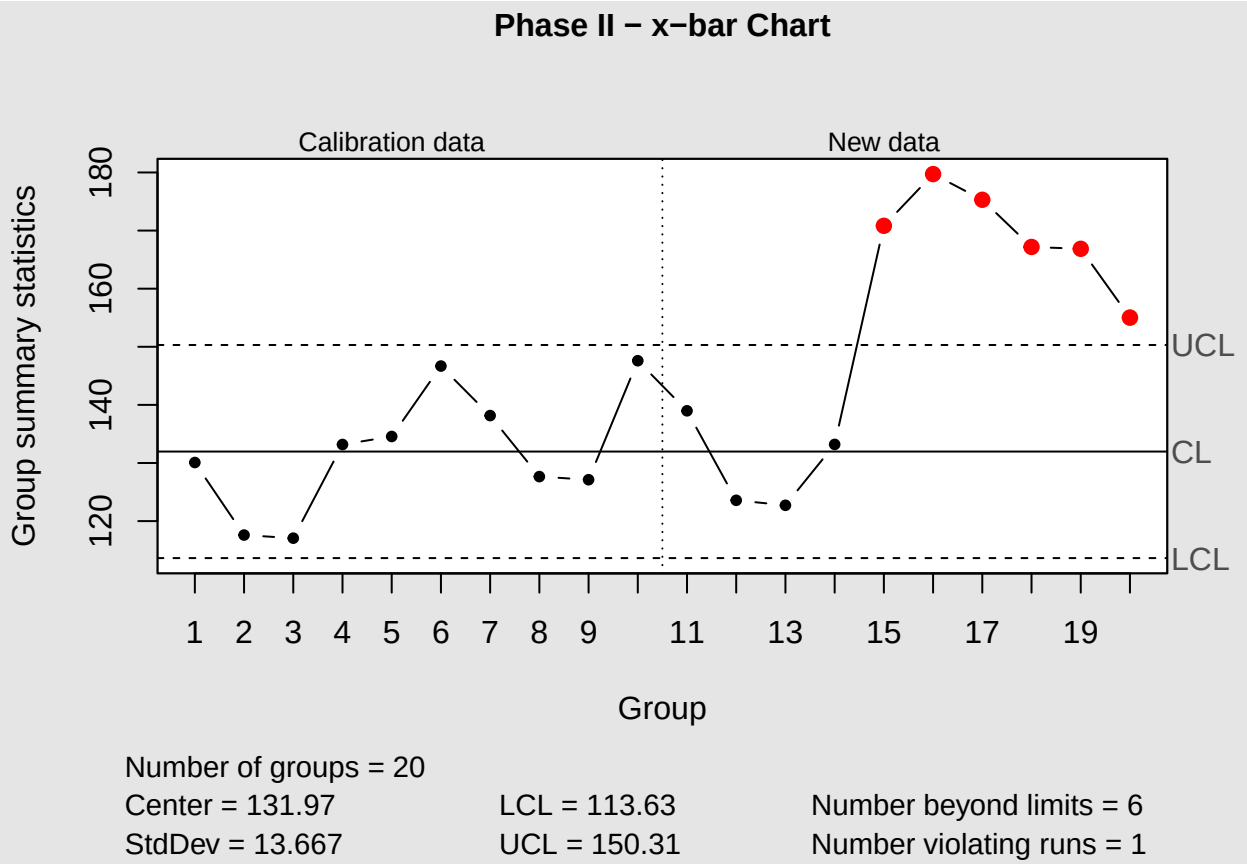

```
process.capability(xbar_phase_I,spec.limits=c(70,190),target= 130)
```




```
##
## Process Capability Analysis
##
## Call:
## process.capability(object = xbar_phase_I, spec.limits = c(70,     190), target = 130)
##
## Number of obs = 50           Target = 130
##        Center = 132             LSL = 70
##        StdDev = 13.67           USL = 190
##
## Capability indices:
##
##       Value   2.5%  97.5%
## Cp    1.463  1.174  1.752
## Cp_l  1.511  1.249  1.774
## Cp_u  1.415  1.168  1.663
## Cp_k  1.415  1.120  1.710
## Cpm   1.448  1.162  1.734
##
## Exp<LSL 0%    Obs<LSL 0%
## Exp>USL 0%    Obs>USL 0%
```

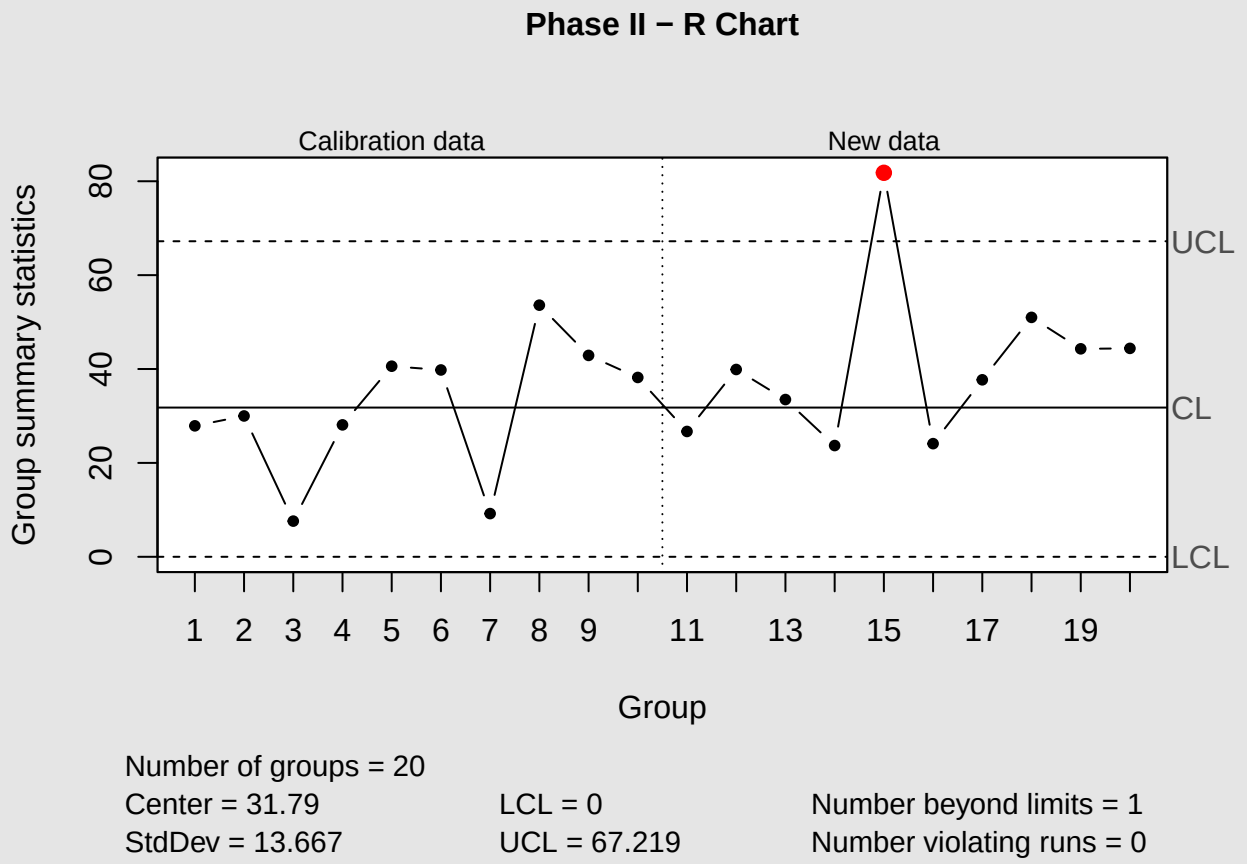
The graph is slightly skewed right, which means it is not a normally distributed dataset.

```
############ Phase II ##########
# Limits are based on calibration data only
xbar_phase_II <- qcc(data = chartdata[1:10,],
type = "xbar",
newdata=chartdata[11:20,],
sizes = 5,
title = "Phase II - x-bar Chart",
digits = 5,
plot = TRUE,bg.margin = "white", bg.figure = "white")
```



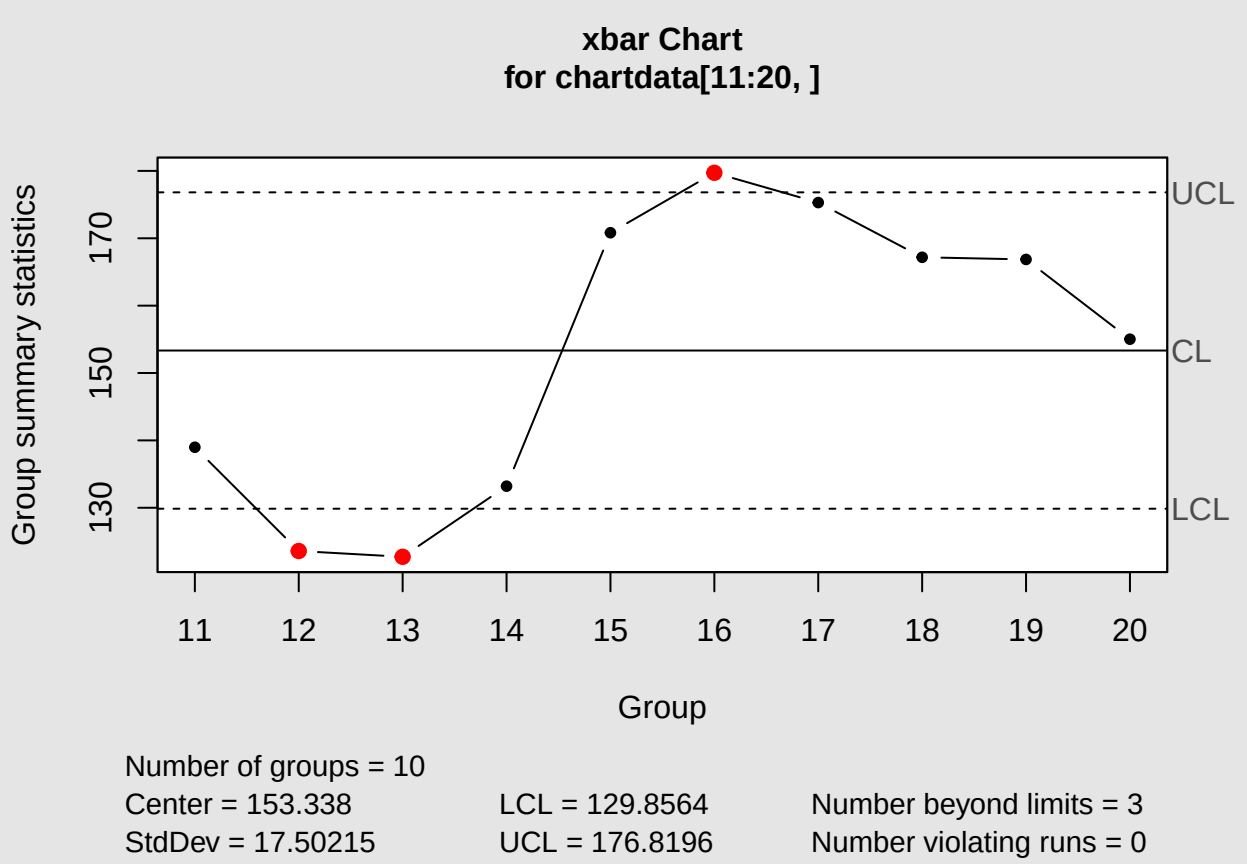
It can be seen that there is a significant number of parts that are beyond the limit.

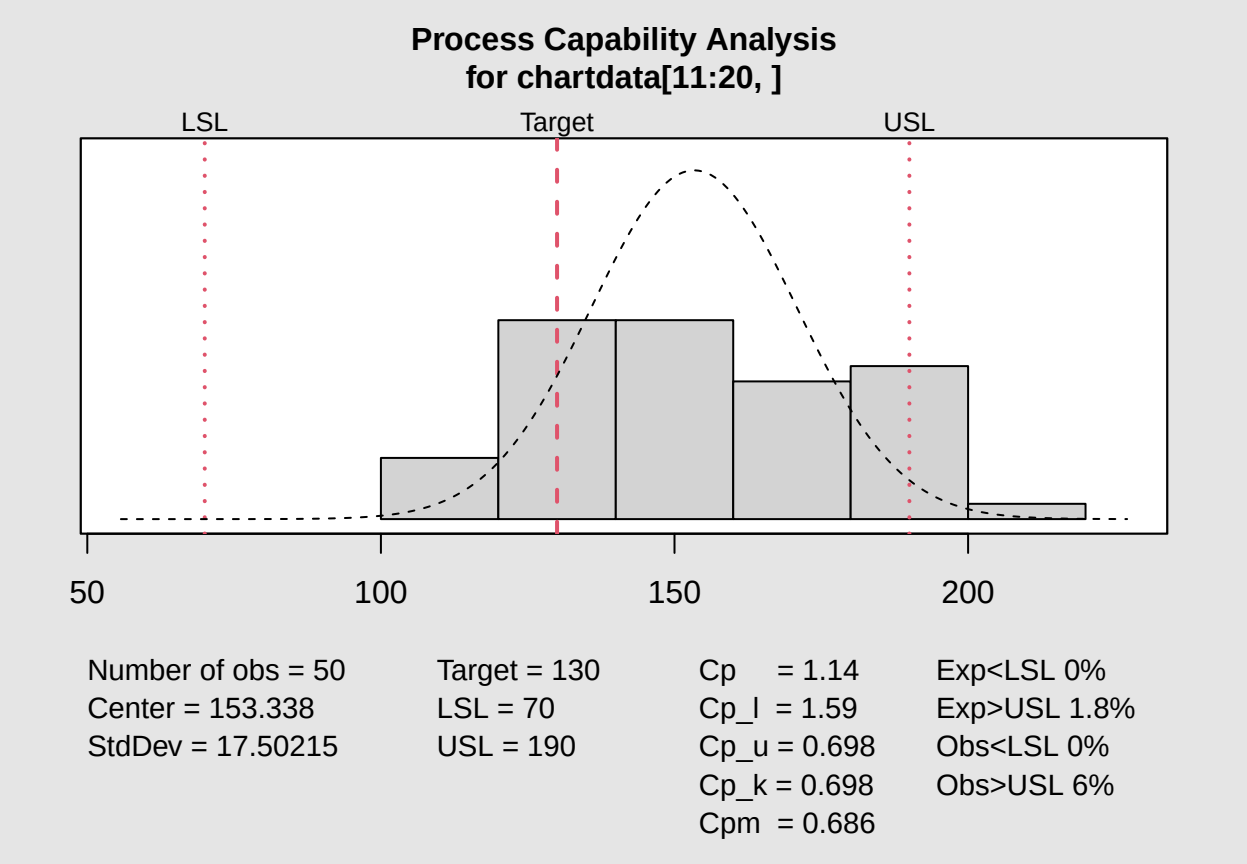
```
rbar_phase_II <- qcc(data = chartdata[1:10,],
type = "R",
newdata=chartdata[11:20,],
sizes = 5,
title = "Phase II - R Chart",
digits = 5,
plot = TRUE,bg.margin = "white", bg.figure = "white")
```



This graph shows there is significant deviation, enough that there is not normality, and the range of parts produced are not falling within the range accepted by the consumer.

```
process.capability(qcc(data = chartdata[11:20,],
type = "xbar"),
spec.limits=c(70,190),
target= 130)
```


xbar Chart
for chartdata[11:20,]
Group summary statistics
130
150
170
UCL
CL
LCL
11
12
13
14
15
16
17
18
19
20
Group
Number of groups = 10
Center = 153.338
StdDev = 17.50215
LCL = 129.8564
UCL = 176.8196
Number beyond limits = 3
Number violating runs = 0




```
##
## Process Capability Analysis
##
## Call:
## process.capability(object = qcc(data = chartdata[11:20, ], type = "xbar"),     spec.limits = c(70, 19
##
## Number of obs = 50           Target = 130
##        Center = 153.3           LSL = 70
##        StdDev = 17.5            USL = 190
##
## Capability indices:
##
##        Value    2.5%   97.5%
## Cp    1.1427  0.9170  1.3680
## Cp_l  1.5872  1.3123  1.8621
## Cp_u  0.6982  0.5587  0.8378
## Cp_k  0.6982  0.5320  0.8645
## Cpm   0.6856  0.5143  0.8566
##
## Exp<LSL 0%    Obs<LSL 0%
## Exp>USL 1.8%  Obs>USL 6%
```

This graph confirms that the results are not within the acceptable limit.

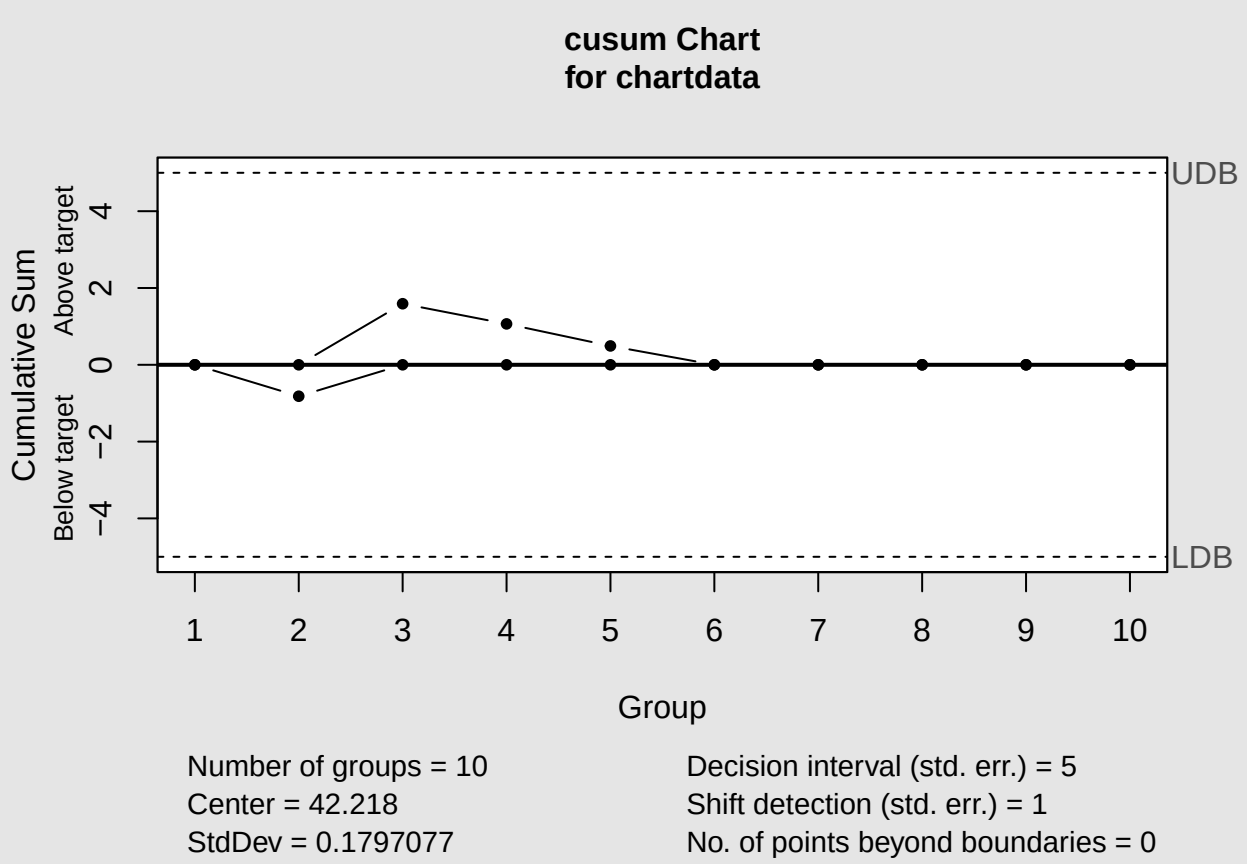
Tutorial 3

A cumulative sum chart (CUSUM) is a type of control chart used to detect the deviation of the individual values or subgroup mean from the adjusted target value, in other words monitor the deviation from the target value. CUSUM chart is an alternative to Shewhart control charts. The basic advantage of CUSUM chart is that it is more sensitive to the small shift of the process mean when compared to the Shewhart charts (Individuals I-MR or Xbar charts)."

The EWMA – Exponentially Weighted Moving Average chart is used in statistical process control to monitor variables (or attributes that act like variables) that make use of the entire history of a given output. This is different from other control charts that tend to treat each data point individually.

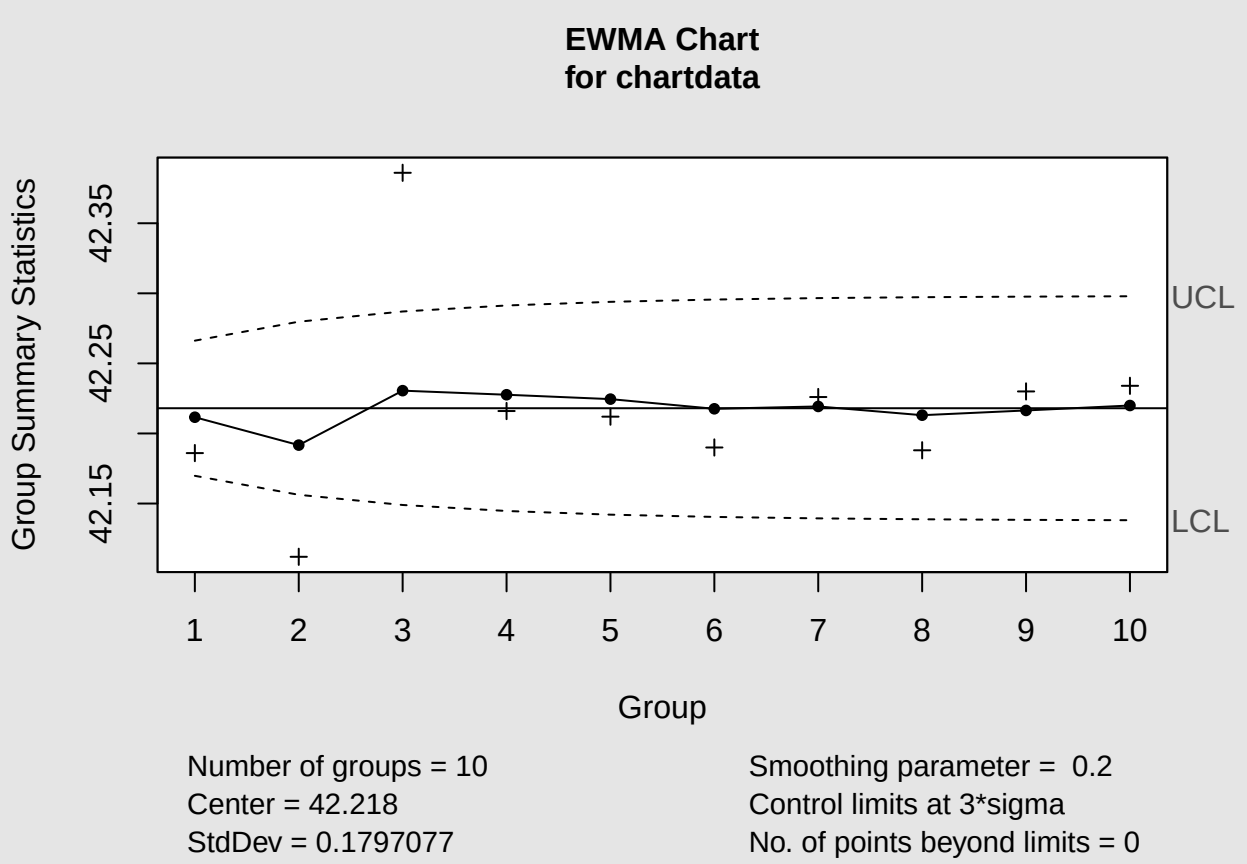
The cumulative sum chart and the exponentially weighted moving average (EWMA) charts are also monitors the mean of the process, but the basic difference is unlike Xbar charts they consider the previous value means at each point. Moreover these charts are considered as a reliable estimate when correct standard deviation exists.

```
library(qcc)
#Chart Data (5 samples (n=5) taken each hour for 10 hours)
chartdata <- read.table(header = FALSE, text = "
42.14 42.27 42.31 42.04 42.17
42.28 42.13 42.14 41.79 42.22
42.13 42.54 42.23 42.34 42.69
42.30 42.19 42.35 42.09 42.15
42.27 42.35 42.45 41.98 42.01
42.22 42.34 42.15 42.19 42.05
41.94 42.05 42.86 42.15 42.13
42.25 42.20 42.02 42.35 42.12
42.18 42.24 42.25 42.34 42.14
42.28 42.32 42.35 42.26 41.96")
cusum(chartdata)
```




```
## List of 14
##  $ call             : language cusum(data = chartdata)
##  $ type             : chr "cusum"
##  $ data.name        : chr "chartdata"
##  $ data             : num [1:10, 1:5] 42.1 42.3 42.1 42.3 42.3 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ statistics       : Named num [1:10] 42.2 42.1 42.4 42.2 42.2 ...
##   ..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...
##  $ sizes            : int [1:10] 5 5 5 5 5 5 5 5 5 5
##  $ center           : num 42.2
##  $ std.dev          : num 0.18
##  $ pos              : num [1:10] 0 0 1.59 1.066 0.491 ...
##  $ neg              : num [1:10] 0 -0.819 0 0 0 ...
##  $ head.start       : num 0
##  $ decision.interval: num 5
##  $ se.shift         : num 1
##  $ violations       :List of 2
##  - attr(*, "class")= chr "cusum.qcc"
```

```
ewma(chartdata)
```




```
## List of 15
##  $ call      : language ewma(data = chartdata)
##  $ type      : chr "ewma"
##  $ data.name : chr "chartdata"
##  $ data      : num [1:10, 1:5] 42.1 42.3 42.1 42.3 42.3 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ statistics: Named num [1:10] 42.2 42.1 42.4 42.2 42.2 ...
##   ..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...
##  $ sizes     : int [1:10] 5 5 5 5 5 5 5 5 5 5
##  $ center    : num 42.2
##  $ std.dev   : num 0.18
##  $ x         : int [1:10] 1 2 3 4 5 6 7 8 9 10
##  $ y         : Named num [1:10] 42.2 42.2 42.2 42.2 42.2 ...
##   ..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...
##  $ sigma     : num [1:10] 0.0161 0.0206 0.023 0.0244 0.0253 ...
##  $ lambda    : num 0.2
##  $ nsigmas   : num 3
##  $ limits    : num [1:10, 1:2] 42.2 42.2 42.1 42.1 42.1 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ violations: Named int(0)
##   ..- attr(*, "names")= chr(0)
##  - attr(*, "class")= chr "ewma.qcc"
```

The graph shows that the measured results are well within an acceptable range, which means the machine is producing parts that are well within the amount accepted by the consumer.

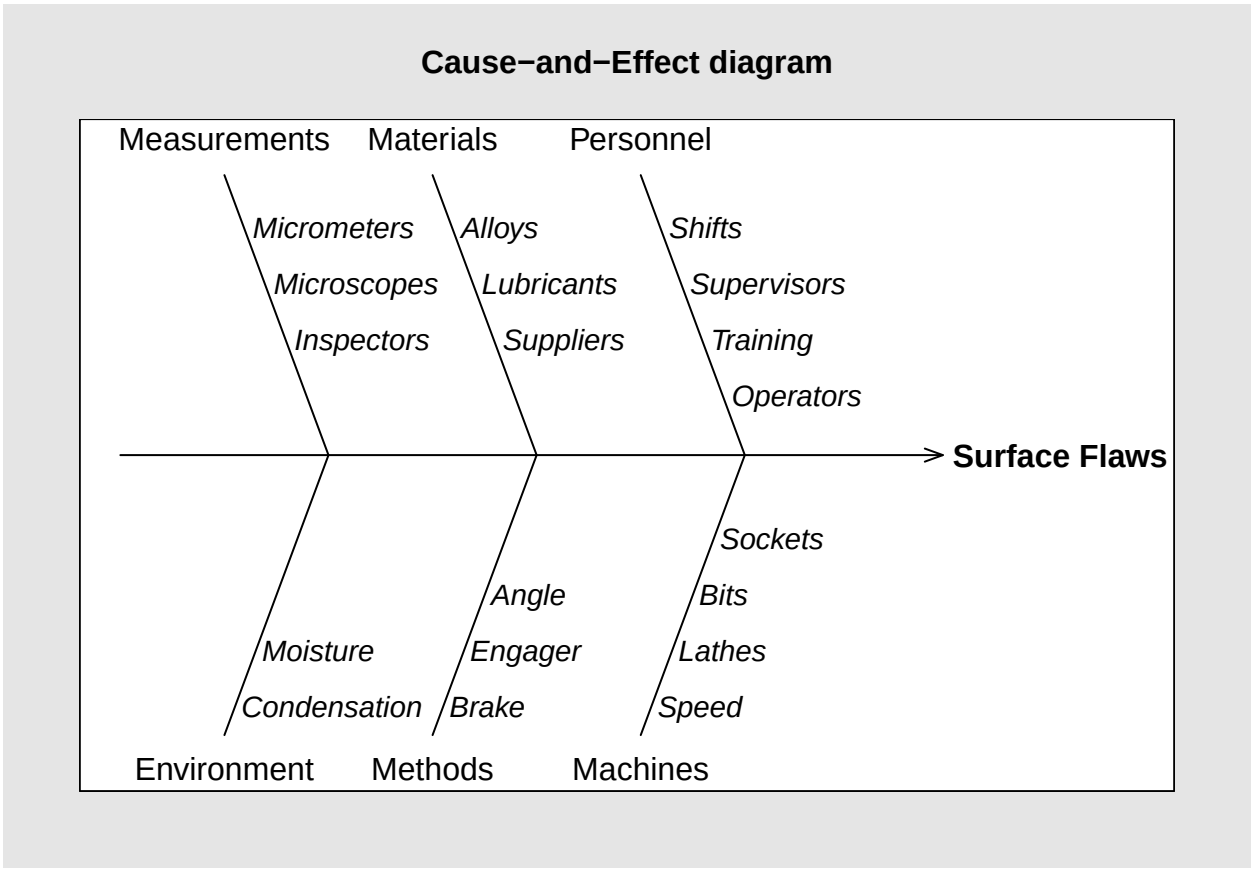
Tutorial 4

A Cause and Effect (a.k.a. Fishbone) identifies many possible causes for an effect or problem. It helps in following ways:

Identify possible causes for an effect or problem.

Effective Brainstorming as it records most of the possible causes of the problem.

```
library(qcc)
cause.and.effect(cause=list(Measurements=c("Micrometers", "Microscopes",
"Inspectors"),
Materials=c("Alloys", "Lubricants", "Suppliers"),
Personnel=c("Shifts", "Supervisors", "Training", "Operators"),
Environment=c("Condensation", "Moisture"),
Methods=c("Brake", "Engager", "Angle"),
Machines=c("Speed", "Lathes", "Bits", "Sockets")),
effect="Surface Flaws")
```

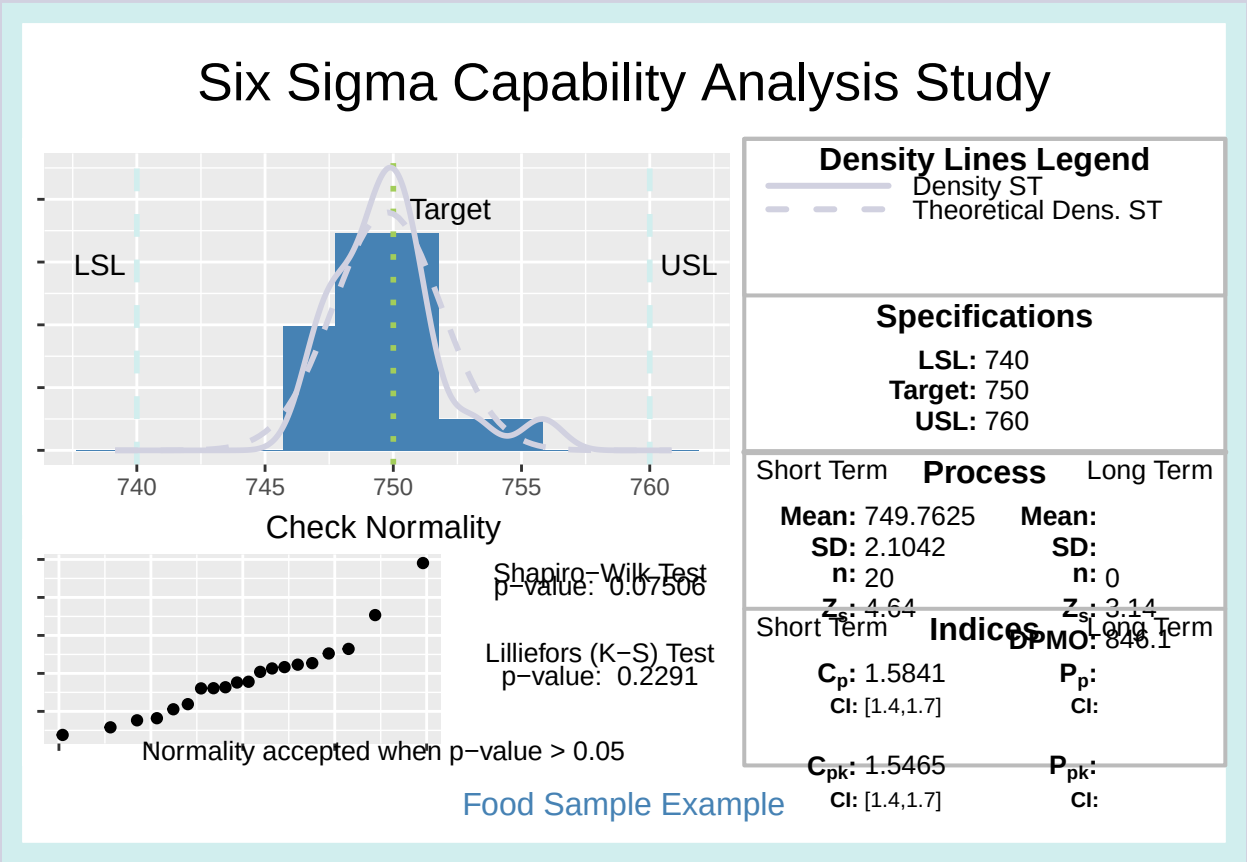


Tutorial 5

```
library(SixSigma)


foodsample<-c(755.81, 750.54, 751.05, 749.52, 749.21, 748.38, 748.11, 753.07, 749.56, 750.08,
747.16, 747.53, 749.22, 746.76, 747.64, 750.46, 749.27, 750.33, 750.26, 751.29)
# perform Capability Study
ss.study.ca(foodsample, LSL = 740, USL = 760, Target = 750, alpha = 0.5, f.su = "Food Sample Example")
```

```
## Warning in ss.study.ca(foodsample, LSL = 740, USL = 760, Target = 750, alpha =
## 0.5, : Normality test/s failed
```



Tutorial 5

```
library(SixSigma)

foodsample<-c(755.81, 750.54, 751.05, 749.52, 749.21, 748.38, 748.11, 753.07, 749.56, 750.08,
747.16, 747.53, 749.22, 746.76, 747.64, 750.46, 749.27, 750.33, 750.26, 751.29)
# perform Capability Study
ss.study.ca(foodsample, LSL = 740, USL = 760, Target = 750, alpha = 0.5, f.su = "Food Sample Example")
```

```
## Warning in ss.study.ca(foodsample, LSL = 740, USL = 760, Target = 750, alpha =
## 0.5, : Normality test/s failed
```