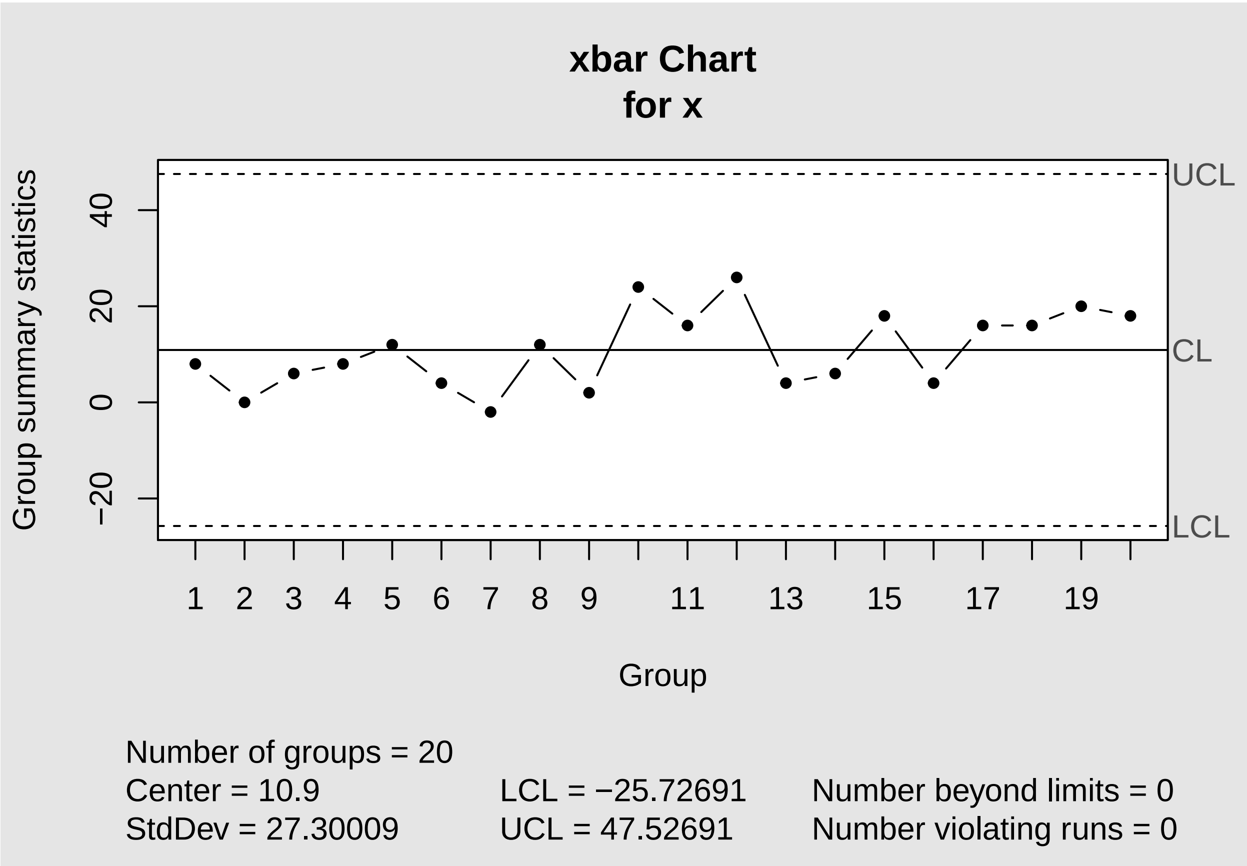
Chapter 6 Quiz

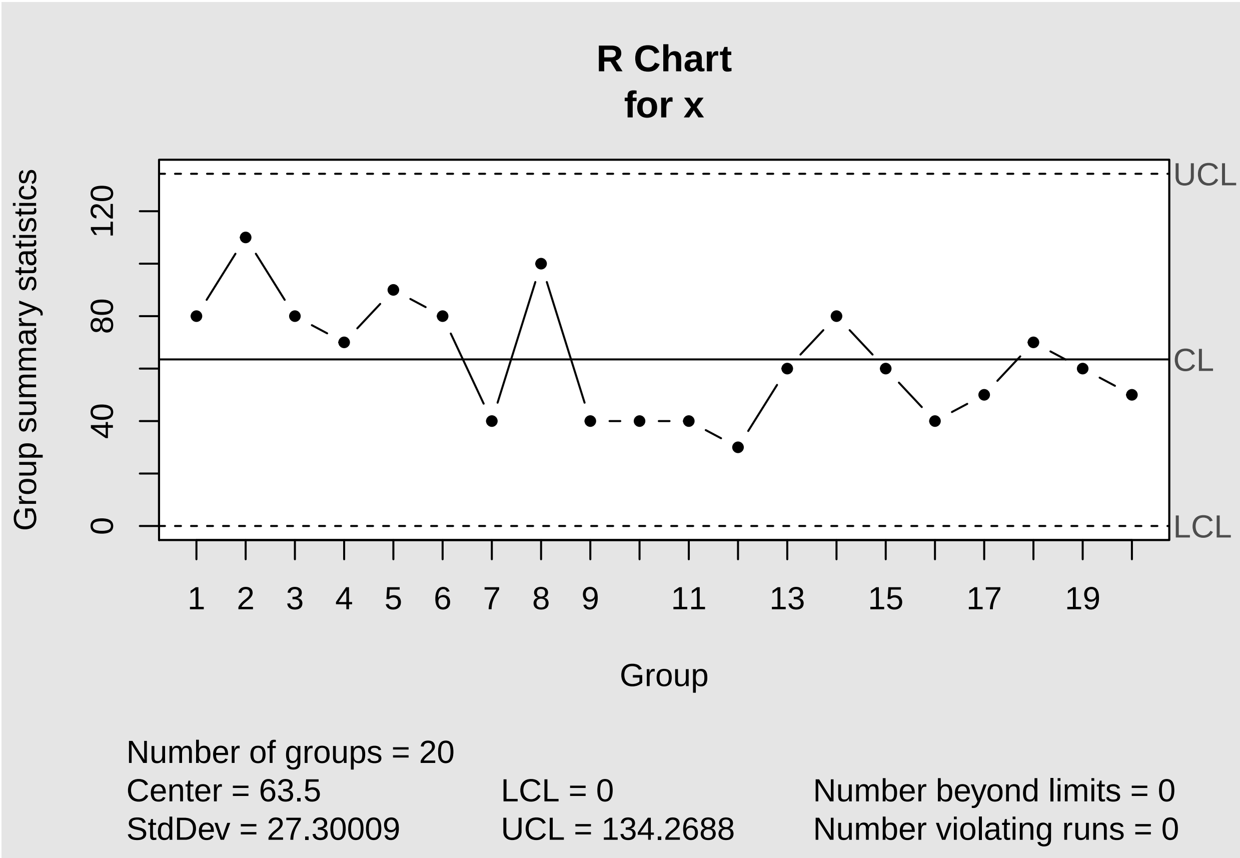
Cody Frisby

10/19/2016

6.3

A





It appears the process is in statistical control. There are no points outside the control limits on either the xbar or the R charts.

B

The process standard deviation can be estimated using the range method

$$\hat{\sigma} = \frac{\bar{R}}{d_2} = \frac{63.5}{2.326} = 27.300086$$

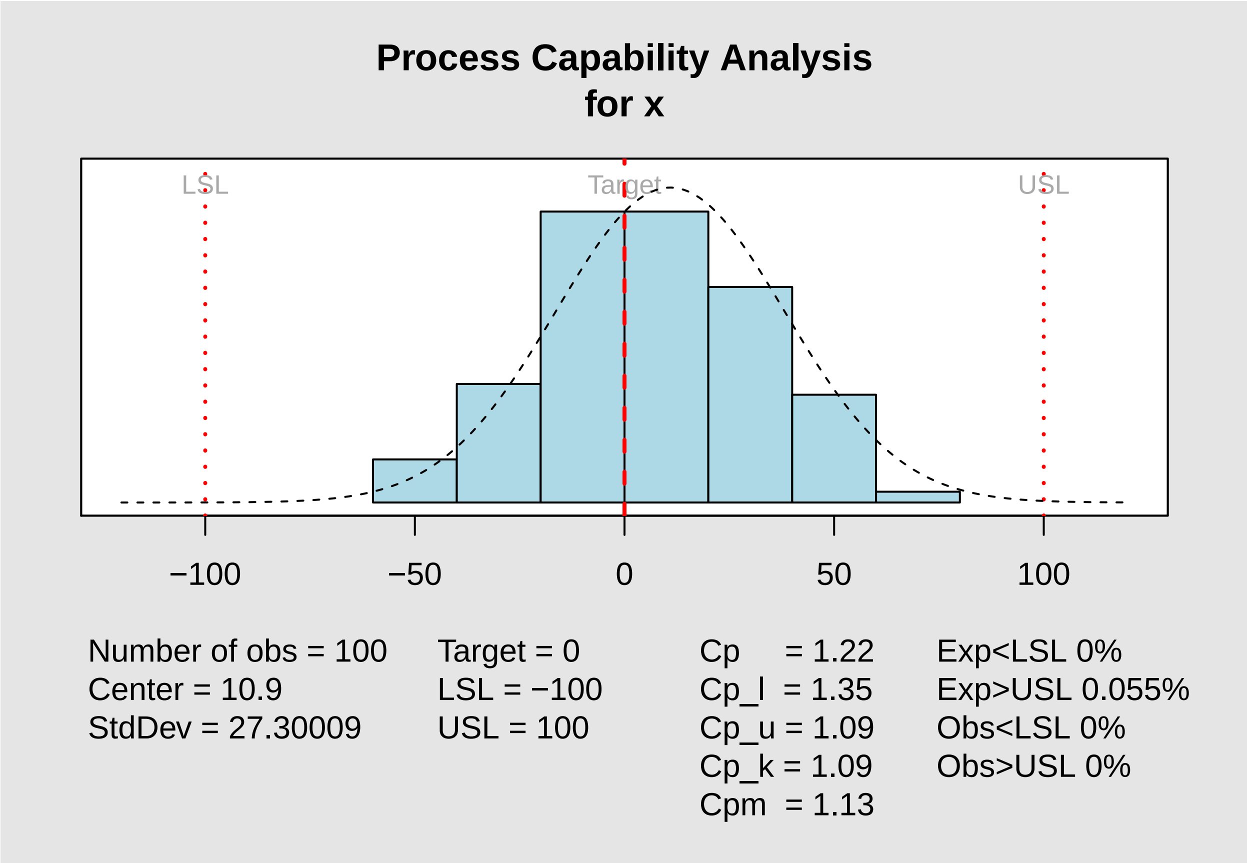
C

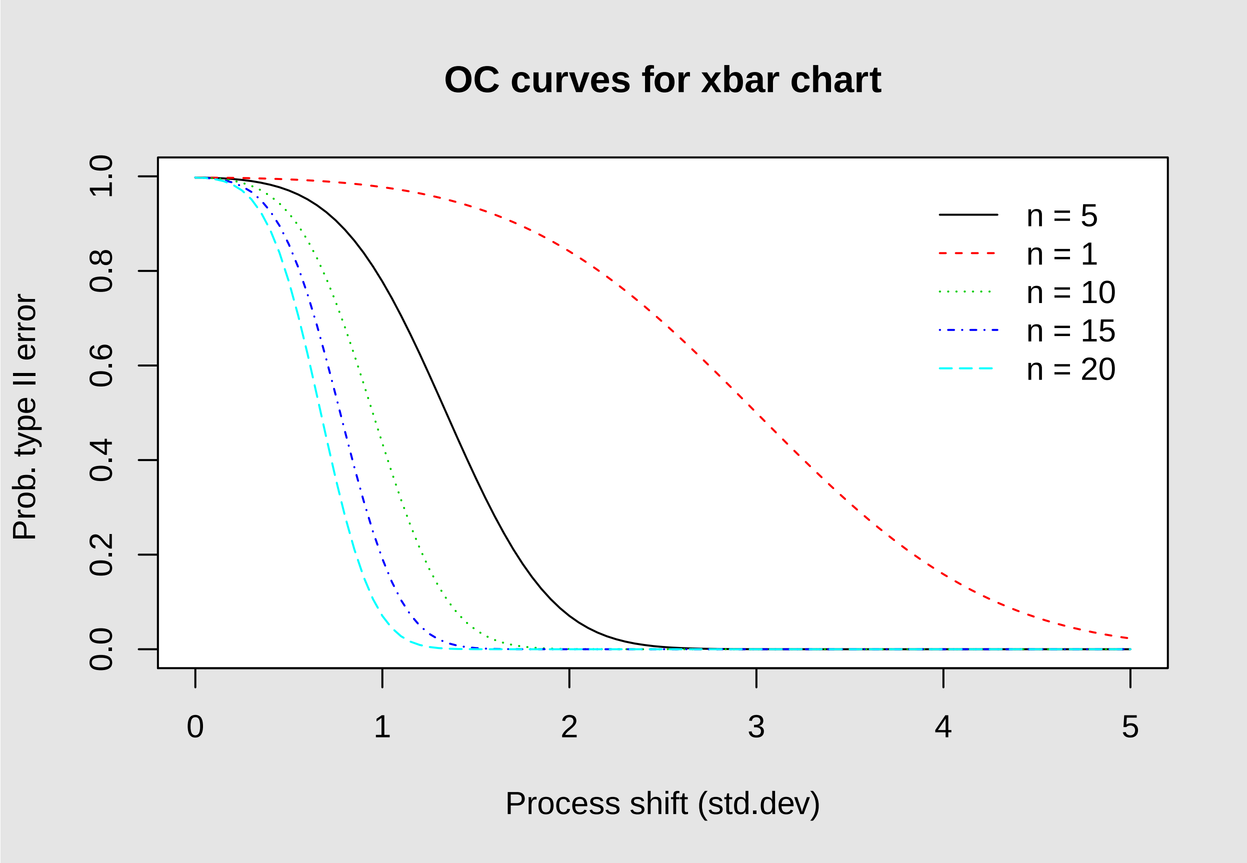
Calculating the PCR estimate of C_p is

$$\hat{C}_p = \frac{USL - LSL}{6\sigma} = \frac{100 - (-100)}{163.8005159} = 1.2209974$$

and it appears that the process is capable of producting a small number of non-conforming units. We can also understand this capability by computing $\frac{1}{\hat{C}_p}100$. This value is understood to be the percentage of the specification that the process will use up.

$$\frac{1}{\hat{C}_p}100 = \frac{1}{1.2209974}100 = 81.900258$$


Process Capability Analysis
for x
LSL
Target
USL
−100
−50
0
50
100
Number of obs = 100
Center = 10.9
StdDev = 27.30009
Target = 0
LSL = −100
USL = 100
Cp = 1.22
Cp_l = 1.35
Cp_u = 1.09
Cp_k = 1.09
Cpm = 1.13
Exp<LSL 0%
Exp>USL 0.055%
Obs<LSL 0%
Obs>USL 0%


OC curves for xbar chart
Prob. type II error
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
Process shift (std.dev)
n = 5
n = 1
n = 10
n = 15
n = 20

R Code:

```
obs <- c(-30,0,-50,-10,20,0,0,70,0,10,40,30,30,30,10,0,20,10,50,50,
         50,50,10,-10,-40,0,0,-30,0,20,0,20,-30,-10,-10,0,20,-20,-10,0,
         -20,-60,20,30,50,40,20,30,20,30,20,30,0,50,50,30,30,50,40,0,
         10,-20,30,-20,20,-40,-20,-10,-20,10,0,10,10,-10,40,-10,30,30,20,30,
         30,30,20,50,10,20,-10,0,10,50,20,40,10,-30,0,0,-20,10,0,10)
xdat <- matrix(obs, ncol=5)
x <- apply(xdat,1,mean) # sample means.
R <- apply(xdat, 1, function(x){max(x)-min(x)}) # ranges
tv <- .577 #table value A2
ucl <- mean(x) + tv*mean(R)
d2 <- 2.326 # table value D4
sig <- mean(R)/d2 # estimate of standard deviation
library(qcc) # a cool quality control package, easy control charts.
x <- xdat
xbar <- qcc(x, "xbar")
rchart <- qcc(x, "R")
cp <- (100 - (-100))/(6 * sig)
cpp <- 1/cp * 100
# adding more...
cap <- process.capability(xbar, c(-100, 100), print = F)
oc.curves(xbar)
```