

title: "Exercícios" output: pdf_document: default date: '20230830'

Desempenho do gráfico de controle

Quando o processo está sob controle é:

$ARL_0 = \text{round}(1/\alpha, 0)$

Quando o processo fora de controle é:

$ARL_1 = \text{round}(1/1\beta, 0)$

Regras para os Gráficos de Controle Shewhart

1. Um ou mais pontos fora dos limites de controle.
2. Dois, de três pontos consecutivos, fora dos limites de alerta (2 sigma).
3. Quatro, de cinco pontos consecutivos, além dos limites 1 sigma.
4. Uma sequência de oito pontos consecutivos de um mesmo lado da linha central.
5. Seis pontos de uma sequência em tendência crescente ou decrescente.
6. Quinze pontos em sequência na faixa entre 1 sigma.
7. Quatorze pontos em sequência alternadamente para cima e para baixo.
8. Oito pontos em sequência de ambos os lados da linha central, com nenhum deles acima de 3 sigma.
9. Um padrão não usual ou não aleatório nos dados.
10. Um ou mais pontos perto dos limites de alerta ou de controle.

Exercício

Em uma fábrica de anéis de pistão para motores de automóveis, uma característica crítica da qualidade é o diâmetro interno do anel. O processo pode ser controlado em um diâmetro interno médio de 74 mm, e sabe-se que o desvio padrão do diâmetro do anel é de 0,01 mm. Considere que o diâmetro do anel é normalmente distribuído, e que o tamanho da amostra é $n=5$.

a) Determine os limites de controle dois sigma.

```
mu_0 = 74 # média
sigma_0 = 0.01 #desvio padrão
n = 5 # tamanho da amostra

L = 2 # dois sigma
# supondo uma normalidade
alpha = 2*pnorm(L, lower.tail = F) # a probabilidade de um ponto cair fora dos limites

#precisamos do resultado disso:
LIC = mu_0 - L*sigma_0/sqrt(n) #limite de alerta inferior
LSC = mu_0 + L*sigma_0/sqrt(n) #limite de alerta superior

cat("o limite inferior é:", LIC)

## o limite inferior é: 73.99106
```

```
cat("o limite superior é:", LSC)
```

```
## o limite superior é: 74.00894
```

```
alpha; LIC; LSC
```

```
## [1] 0.04550026
```

```
## [1] 73.99106
```

```
## [1] 74.00894
```

- b) Suponha que tenha sido sugerido o limite de dois sigma, em vez dos limites típicos de três sigma. Que efeito o uso dos limites de dois sigma teria sobre o ARL sob controle do gráfico?

```
ARL_0 = round(1/alpha, 0)
```

Exemplo Resolvido Aula 3

– Anéis de pistão para motores de automóveis são fabricados por um processo. Deseja-se estabelecer um controle estatístico para o diâmetro interno dos anéis. – 25 amostras (m), cada uma de tamanho 5 (n), foram extraídas quando o mesmo estava sob controle. – As medidas são exibidas a seguir

Para xbarra R

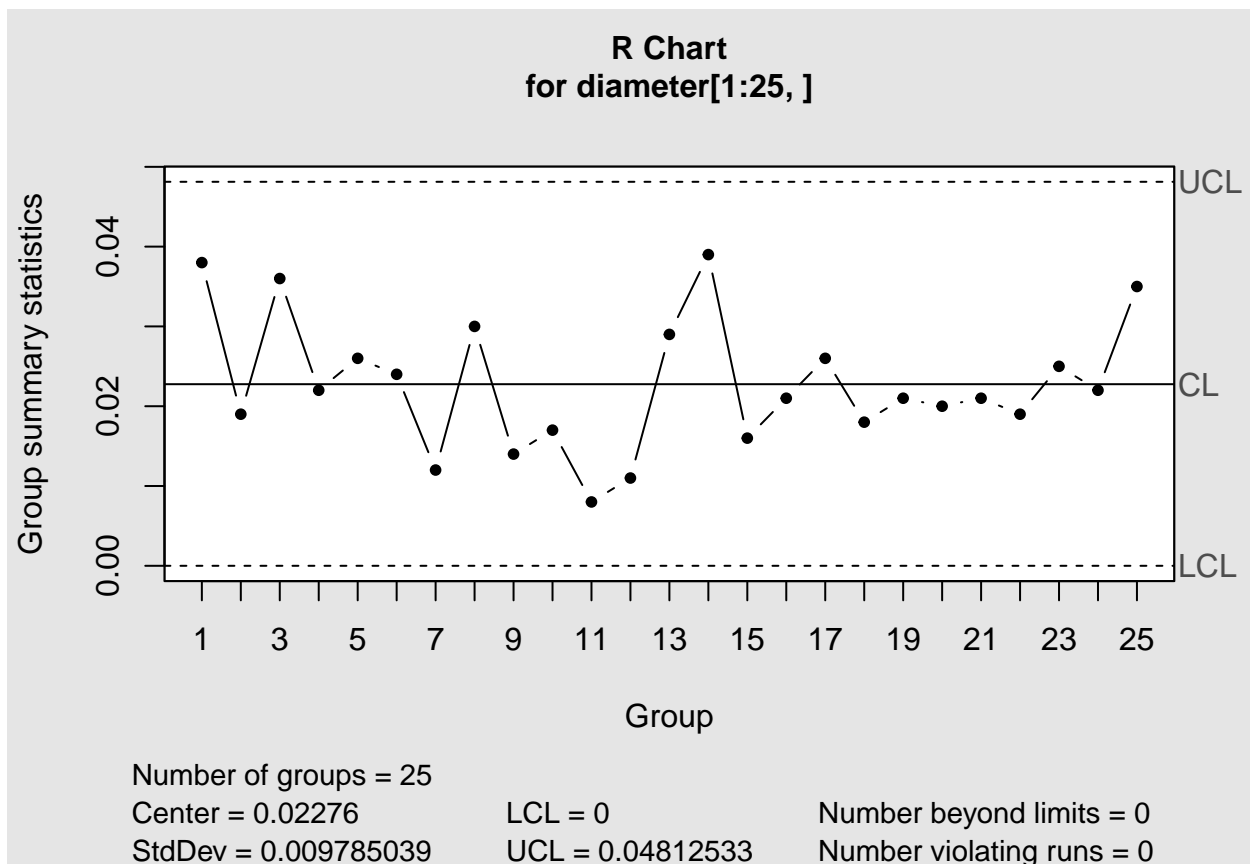
```
library(qcc)
```

```
## Warning: package 'qcc' was built under R version 4.1.3
```

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

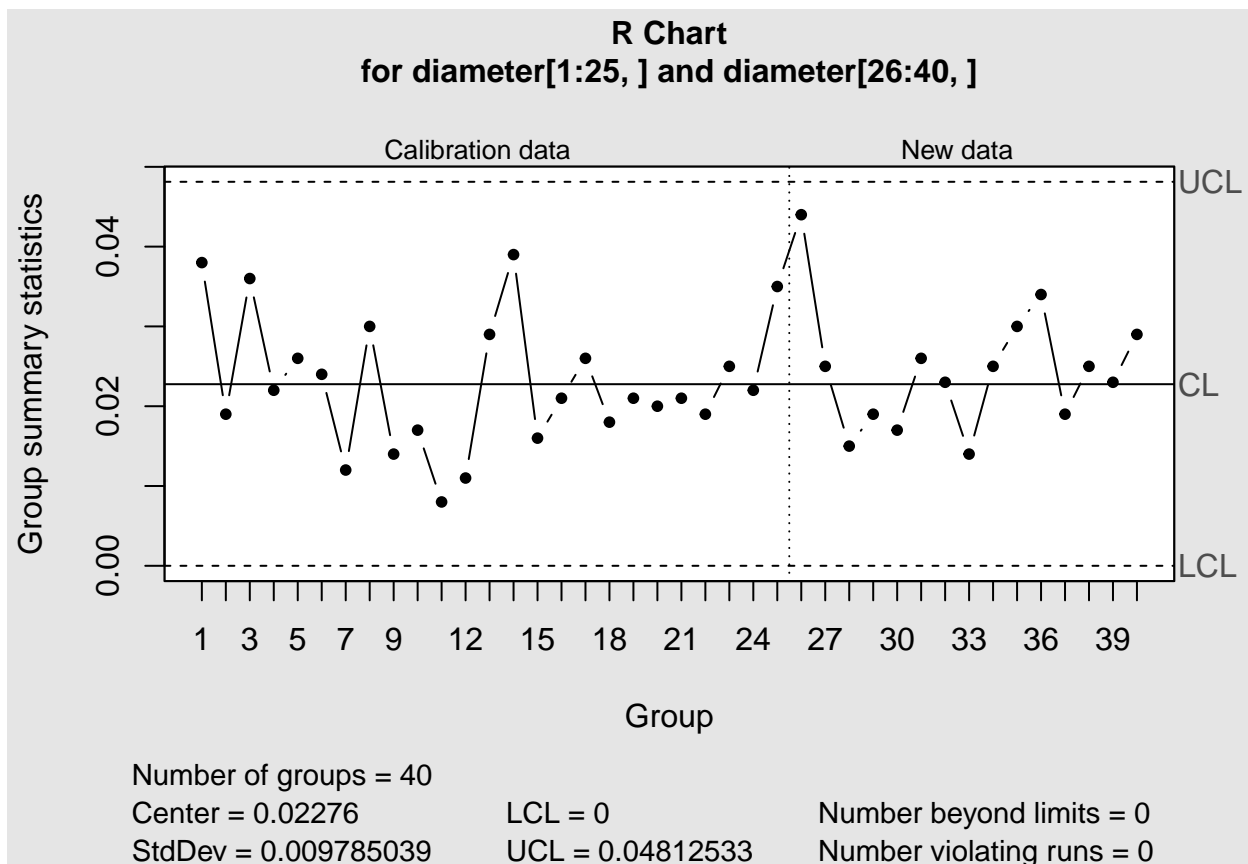
```
## Type 'citation("qcc")' for citing this R package in publications.
```

```
data(pistonrings)
attach(pistonrings)
diameter <- qcc.groups(diameter, sample)
# Gráfico para a amplitude (R)
qcc(diameter[1:25,], type="R")
```



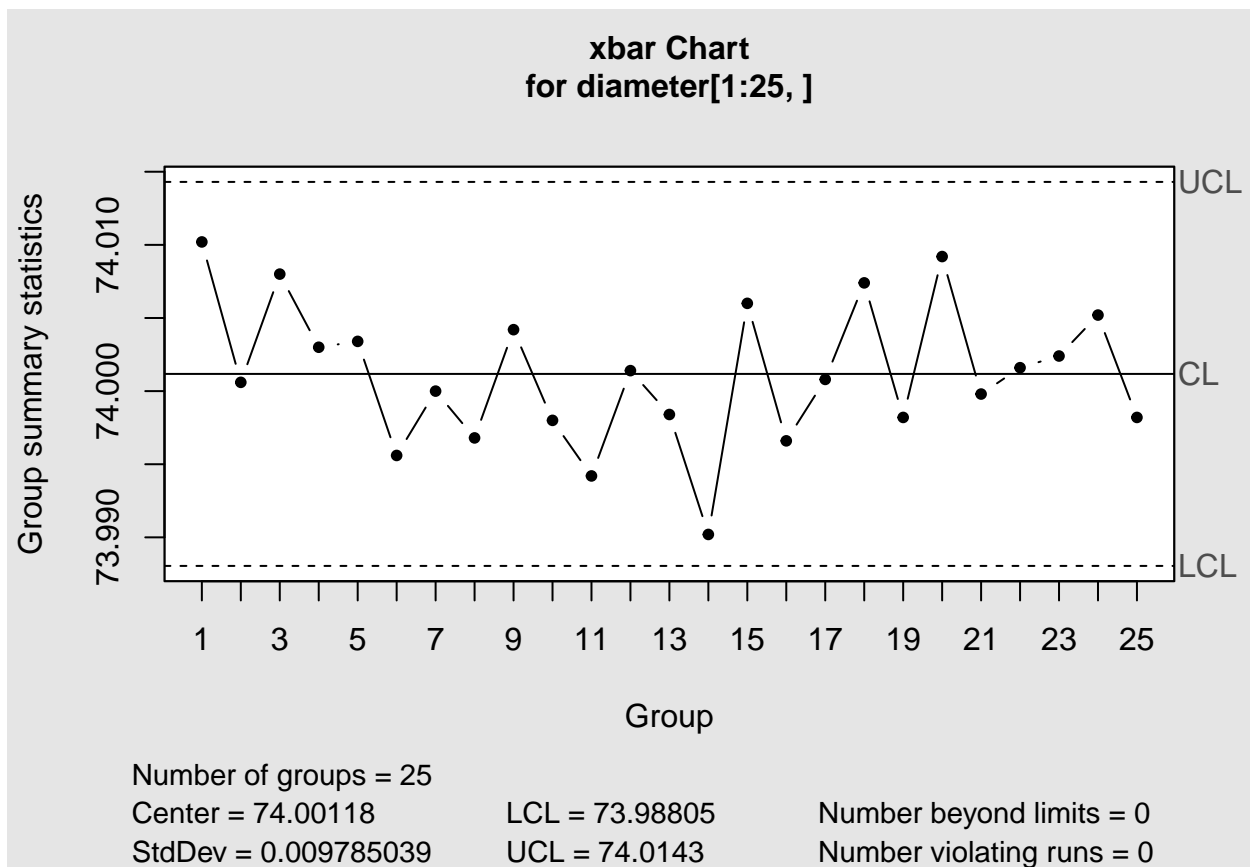
```
## List of 11
## $ call      : language qcc(data = diameter[1:25, ], type = "R")
## $ type      : chr "R"
## $ data.name : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics: Named num [1:25] 0.038 0.019 0.036 0.022 0.026 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes     : Named int [1:25] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center    : num 0.0228
## $ std.dev   : num 0.00979
## $ nsigmas   : num 3
## $ limits    : num [1, 1:2] 0 0.0481
## .. attr(*, "dimnames")=List of 2
## $ violations:List of 2
## - attr(*, "class")= chr "qcc"
```

```
qcc(diameter[1:25,], type="R", newdata=diameter[26:40,])
```



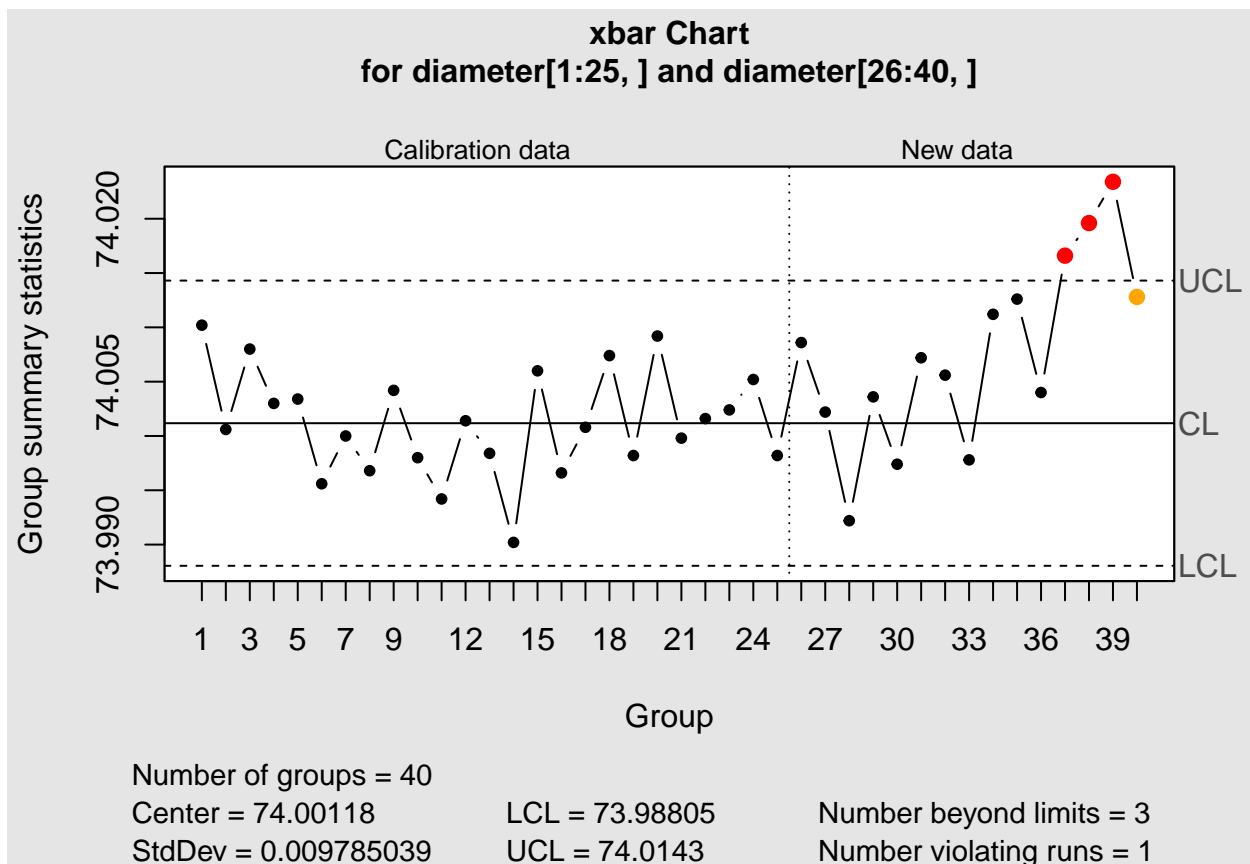
```
## List of 15
## $ call      : language qcc(data = diameter[1:25, ], type = "R", newdata = diameter[26:40, ])
## $ type      : chr "R"
## $ data.name  : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics : Named num [1:25] 0.038 0.019 0.036 0.022 0.026 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center     : num 0.0228
## $ std.dev    : num 0.00979
## $ newstats   : Named num [1:15] 0.044 0.025 0.015 0.019 0.017 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata.name: chr "diameter[26:40, ]"
## $ nsigmas    : num 3
## $ limits     : num [1, 1:2] 0 0.0481
## .. attr(*, "dimnames")=List of 2
## $ violations :List of 2
## - attr(*, "class")= chr "qcc"
```

```
# Construindo a carta para a média
qcc(diameter[1:25,], type="xbar")
```



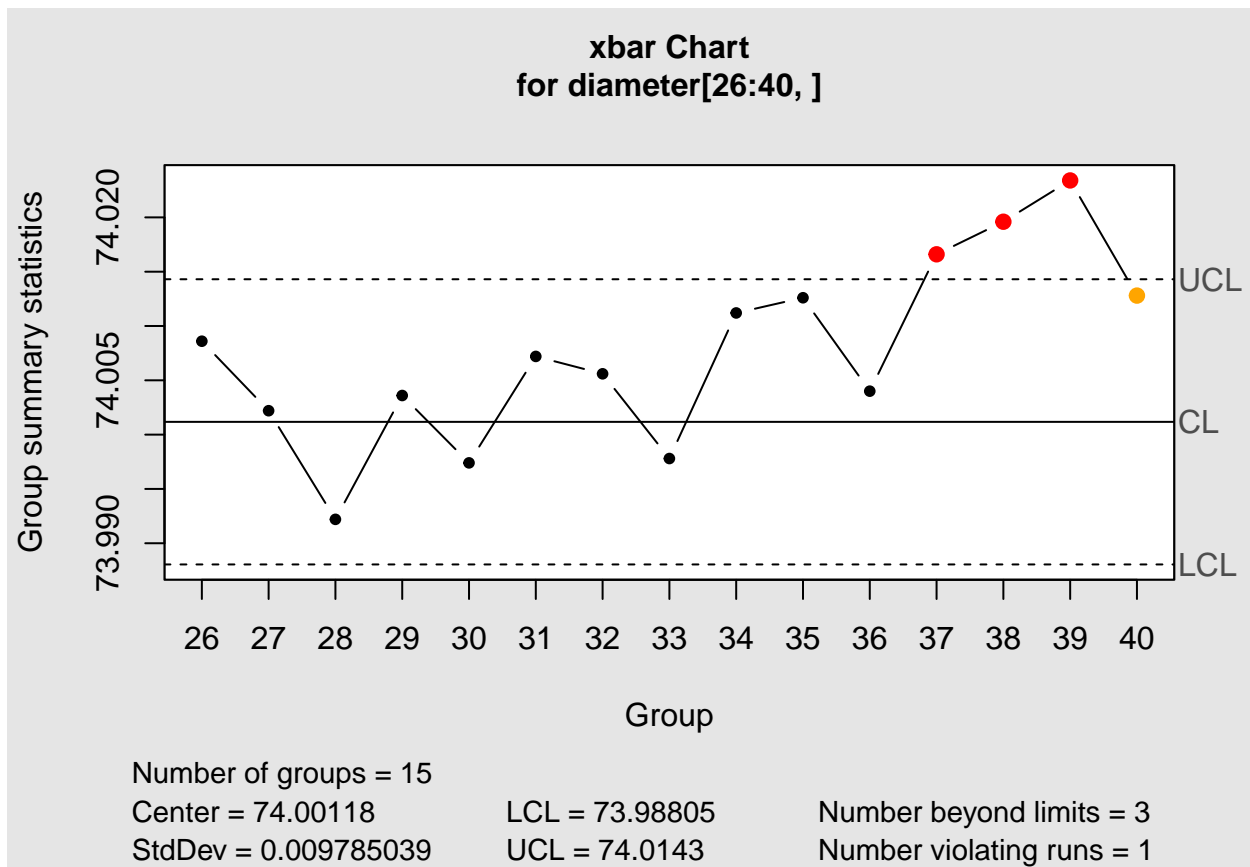
```
## List of 11
## $ call      : language qcc(data = diameter[1:25, ], type = "xbar")
## $ type      : chr "xbar"
## $ data.name : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## ..- attr(*, "dimnames")=List of 2
## $ statistics: Named num [1:25] 74 74 74 74 74 ...
## ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes     : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
## ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center    : num 74
## $ std.dev   : num 0.00979
## $ nsigmas   : num 3
## $ limits    : num [1, 1:2] 74 74
## ..- attr(*, "dimnames")=List of 2
## $ violations:List of 2
## - attr(*, "class")= chr "qcc"
```

```
qcc(diameter[1:25,], type="xbar", newdata=diameter[26:40,])
```

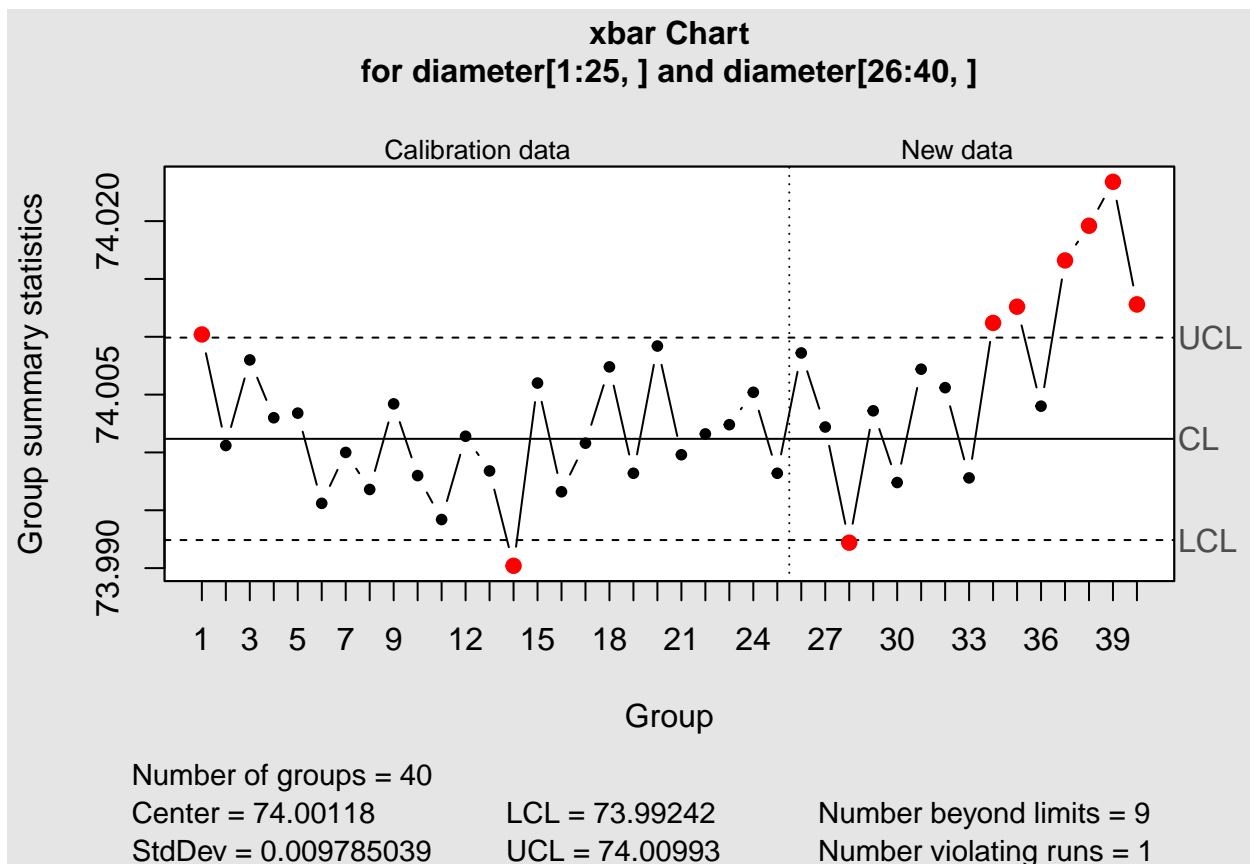


```
## List of 15
## $ call      : language qcc(data = diameter[1:25, ], type = "xbar", newdata = diameter[26:40, ])
## $ type      : chr "xbar"
## $ data.name : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics : Named num [1:25] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center     : num 74
## $ std.dev    : num 0.00979
## $ newstats   : Named num [1:15] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata.name: chr "diameter[26:40, ]"
## $ nsigmas    : num 3
## $ limits     : num [1, 1:2] 74 74
## .. attr(*, "dimnames")=List of 2
## $ violations  :List of 2
## - attr(*, "class")= chr "qcc"
```

```
q <- qcc(diameter[1:25,], type="xbar", newdata=diameter[26:40,], plot=FALSE)
plot(q, chart.all=FALSE)
```



```
qcc(diameter[1:25,], type="xbar", newdata=diameter[26:40,], nsigmas=2) # limites 2-sigma
```



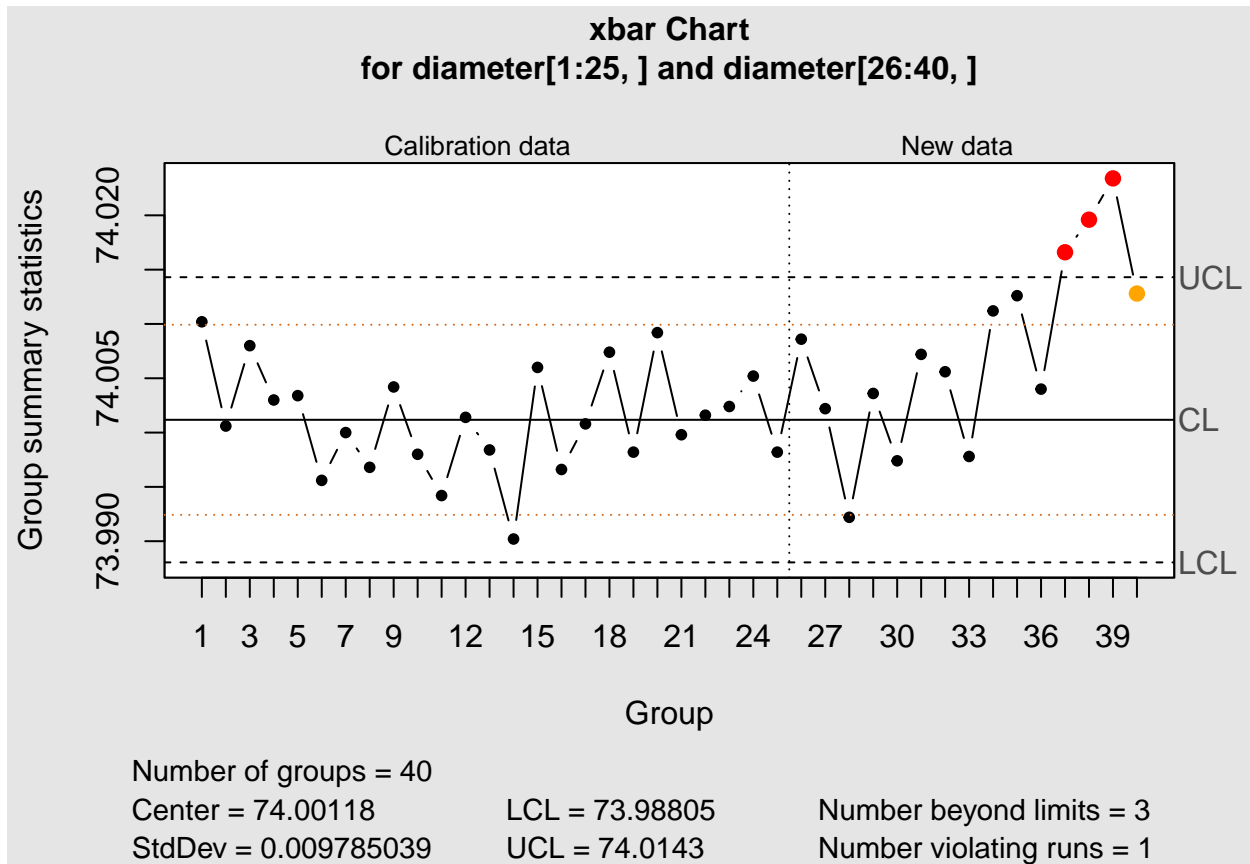
```
## List of 15
## $ call      : language qcc(data = diameter[1:25, ], type = "xbar", newdata = diameter[26:40, ],
## $ type      : chr "xbar"
## $ data.name  : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics : Named num [1:25] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center     : num 74
## $ std.dev    : num 0.00979
## $ newstats   : Named num [1:15] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata.name: chr "diameter[26:40, ]"
## $ nsigmas    : num 2
## $ limits     : num [1, 1:2] 74 74
## .. attr(*, "dimnames")=List of 2
## $ violations  :List of 2
## - attr(*, "class")= chr "qcc"
```



```
q <- qcc(diameter[1:25,], type="xbar", newdata=diameter[26:40,], plot=FALSE)
plot(q, restore.par = FALSE)
(warn.limits <- limits.xbar(q$center, q$std.dev, q$sizes, 2))
```

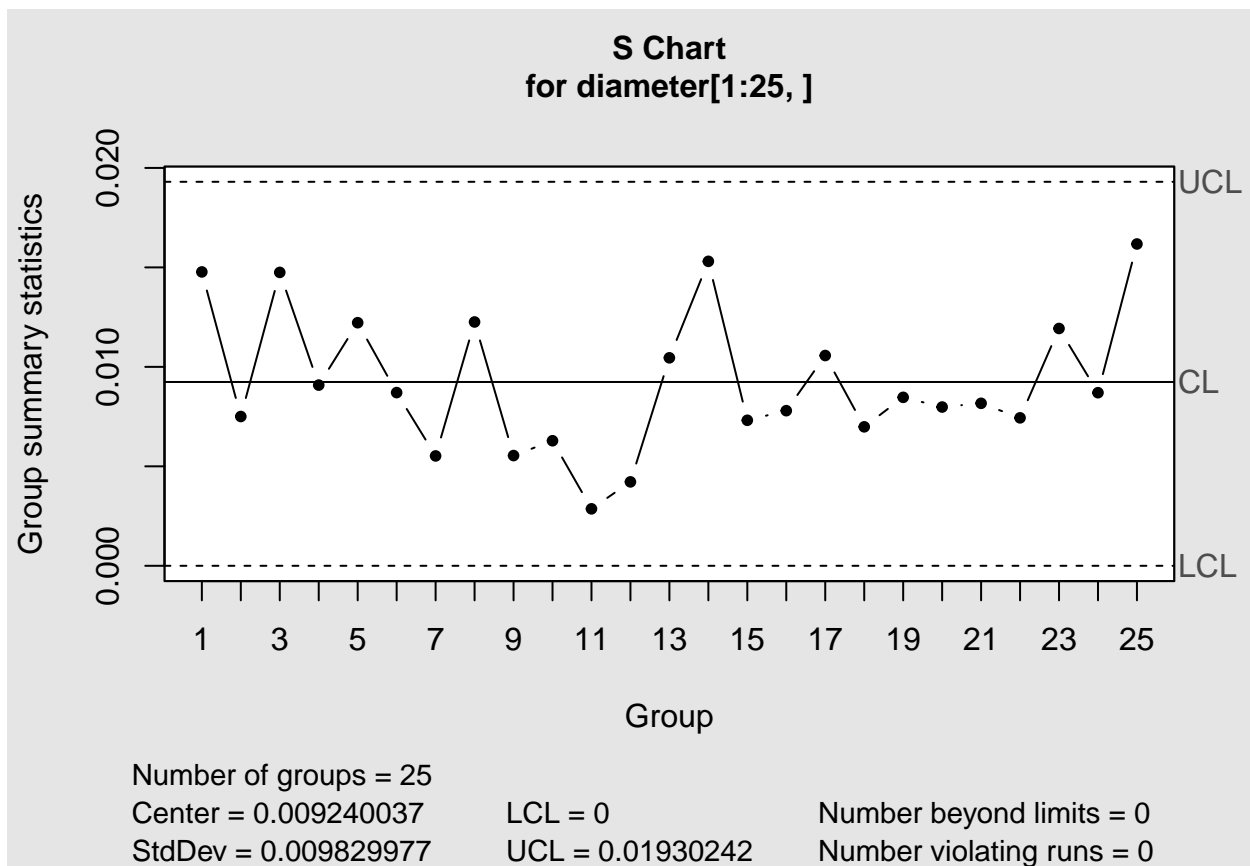
```
##          LCL          UCL
## 73.99242 74.00993
```

```
abline(h = warn.limits, lty = 3, col = "chocolate")
```



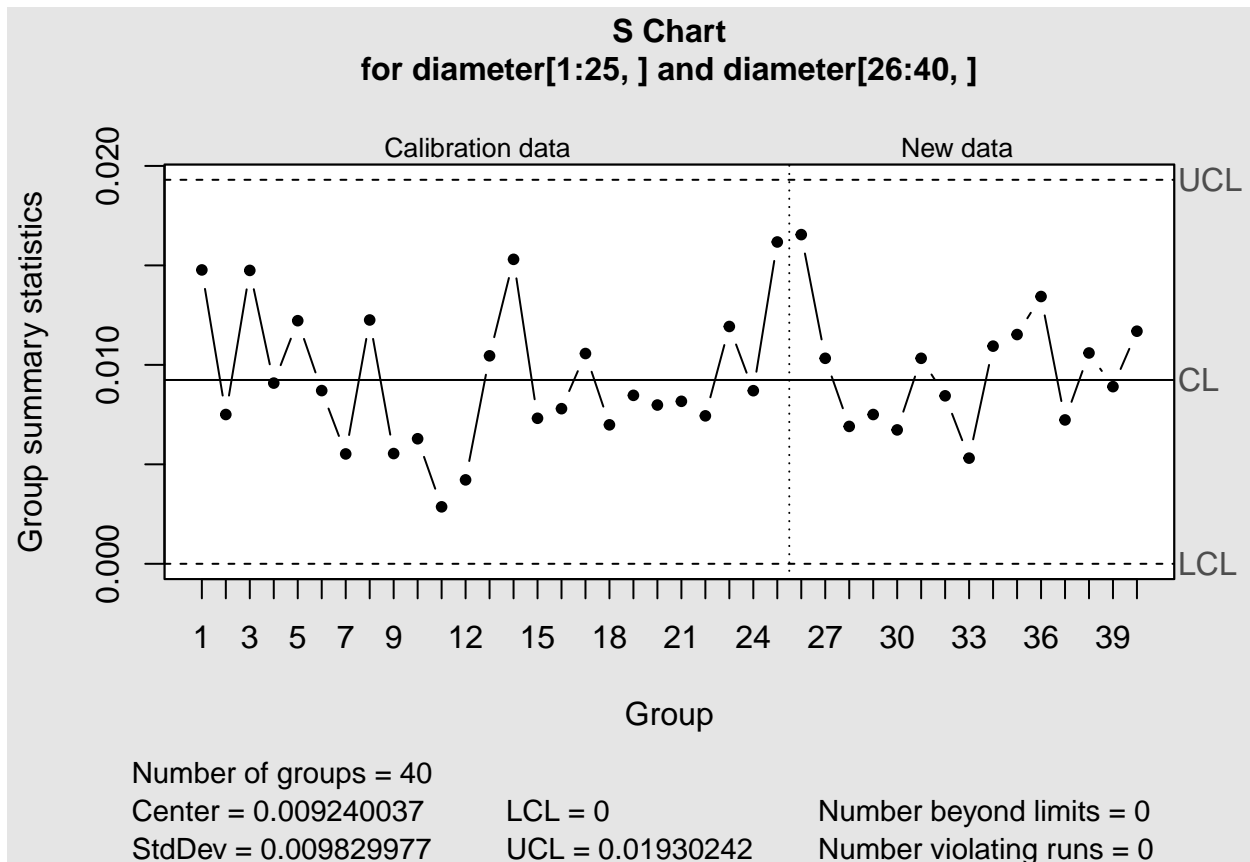
Para xbarra s

```
# Gráfico para o desvio padrão (S)
qcc(diameter[1:25,], type="S")
```



```
## List of 11
## $ call      : language qcc(data = diameter[1:25, ], type = "S")
## $ type      : chr "S"
## $ data.name : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## ..- attr(*, "dimnames")=List of 2
## $ statistics: Named num [1:25] 0.01477 0.0075 0.01475 0.00908 0.01222 ...
## ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes     : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
## ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center    : num 0.00924
## $ std.dev   : num 0.00983
## $ nsigmas   : num 3
## $ limits    : num [1, 1:2] 0 0.0193
## ..- attr(*, "dimnames")=List of 2
## $ violations:List of 2
## - attr(*, "class")= chr "qcc"
```

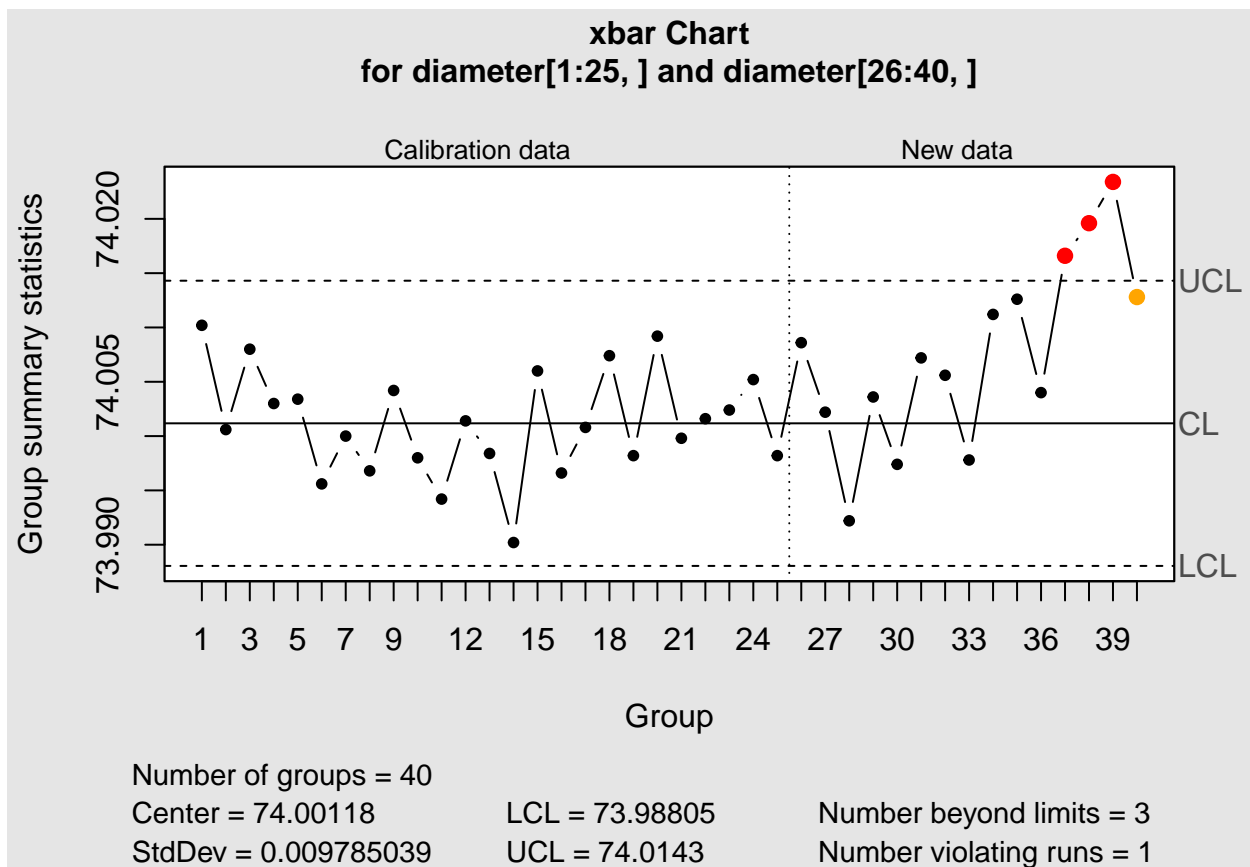
```
qcc(diameter[1:25,], type="S", newdata=diameter[26:40,])
```



```
## List of 15
## $ call      : language qcc(data = diameter[1:25, ], type = "S", newdata = diameter[26:40, ])
## $ type      : chr "S"
## $ data.name  : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics : Named num [1:25] 0.01477 0.0075 0.01475 0.00908 0.01222 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center     : num 0.00924
## $ std.dev    : num 0.00983
## $ newstats   : Named num [1:15] 0.01655 0.01033 0.00691 0.0075 0.00673 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata.name: chr "diameter[26:40, ]"
## $ nsigmas    : num 3
## $ limits     : num [1, 1:2] 0 0.0193
## .. attr(*, "dimnames")=List of 2
## $ violations :List of 2
## - attr(*, "class")= chr "qcc"
```

```
# Gráfico para a média
```

```
qcc(diameter[1:25,], type="xbar", newdata=diameter[26:40,])
```



```
## List of 15
## $ call      : language qcc(data = diameter[1:25, ], type = "xbar", newdata = diameter[26:40, ])
## $ type      : chr "xbar"
## $ data.name  : chr "diameter[1:25, ]"
## $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ statistics : Named num [1:25] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
## $ center     : num 74
## $ std.dev    : num 0.00979
## $ newstats   : Named num [1:15] 74 74 74 74 74 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
## .. attr(*, "dimnames")=List of 2
## $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 ...
## .. attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
## $ newdata.name: chr "diameter[26:40, ]"
## $ nsigmas    : num 3
## $ limits     : num [1, 1:2] 74 74
## .. attr(*, "dimnames")=List of 2
```

```
## $ violations :List of 2
## - attr(*, "class")= chr "qcc"
```