Quality Control-2

The ASTA team

Contents

0.1 OC curves

Usual setup:

- m samples with sample size n.
- Process mean μ and standard deviation σ.
- Sample means have standard deviation $\frac{\sigma}{\sqrt{n}}$.
- By default we use the 3*sigma rule.

Assume that the mean is shifted by $c \times \sigma$.

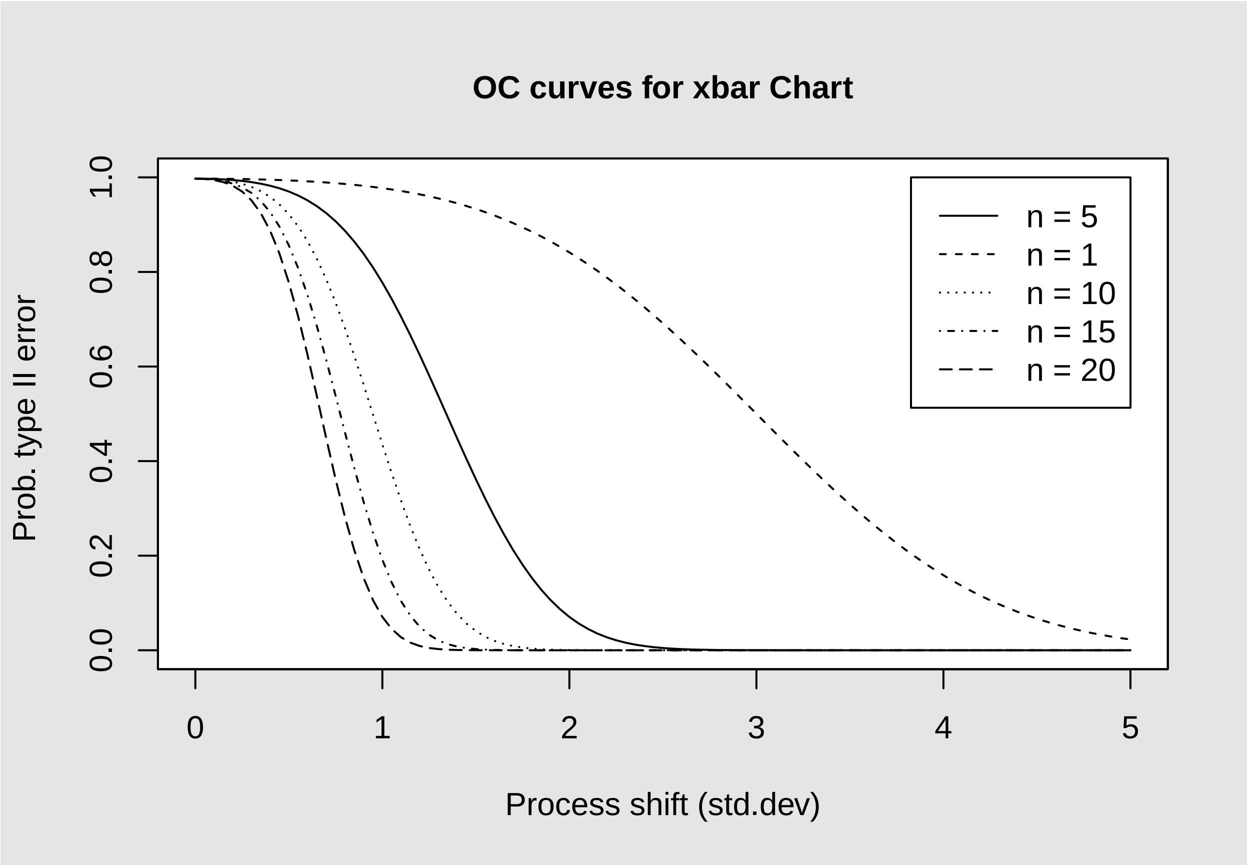
What is the probability of NOT getting an immediate alarm?

This is also called a **type II error**.

```
library(qcc)
data(pistonrings)
diam    <- pistonrings$diameter
phaseI  <- matrix(diam[1:125],25,byrow=TRUE)
phaseII <- matrix(diam[126:200],15,byrow=TRUE)
xbcc    <- qcc(phaseI, std.dev = "UWAVE-SD", type = "xbar", plot = FALSE,
               newdata = phaseII, title = "qq chart: pistonrings")
```

0.2 OC curves - example

```
oc.curves(xbcc)
```



For the actual sample size (n=5) and in case of a process shift c=2, the error probability is around 7%.

If we increase sample size to n=10 then we are almost sure to immediate detection of a shift by 2σ.

0.3 CUSUM chart

The CUSUM chart is generally better than the xbar chart for detecting small shifts in the mean of a process.

Consider the standardized residuals

$$z_i = \frac{\sqrt{n}(x_i - \hat{\mu}_0)}{\hat{\sigma}}$$

where

- $\hat{\mu}_0$ is an estimate of the process mean.
- $\hat{\sigma}$ is an estimate of the process standard deviation.

For an in-control proces the cumulative sum (CUSUM) of residuals should vary around *zero*.

The CUSUM chart is developed to see, if there is a drift away from zero. To that end we define 2 processes controlling for downward(D) respectively upward(U) drift:

$$D(i) = \max\{0, D(i-1) - k - z_i\}$$
$$U(i) = \max\{0, U(i-1) + z_i - k\}$$

where k is a positive number.

0.4 Interpretation of CUSUM chart

Interpretation of

$$D(i) = \max\{0, D(i-1) - k - z_i\}$$
$$U(i) = \max\{0, U(i-1) + z_i - k\}$$

- If $z_i < -k$, i.e. z_i is more that k below zero, then D is increased. If this happens a number of times in a row, then D grows "big".
- If $z_i > k$, i.e. z_i is more that k above zero, then U is increased. If this happens a number of times in a row, then U grows "big".

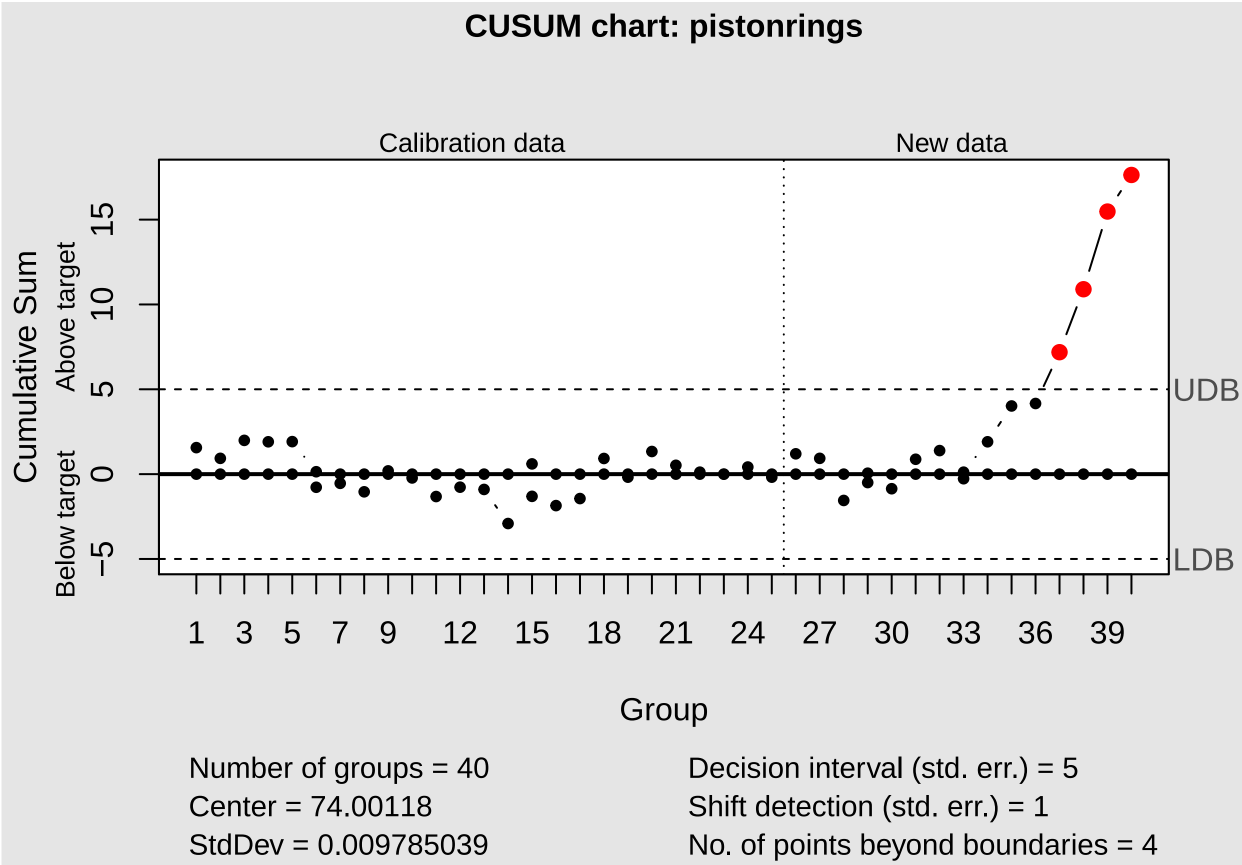
The process is considered out of control if D or U exceeds a limit h.

In `qcc` k and h is specified by the arguments:

- `se.shift` is k, which has a default value of 1.
- `decision.interval` is h, which has a default value of 5.

0.5 CUSUM chart example

```
h <- cusum(phaseI, newdata = phaseII, title = "CUSUM chart: pistonrings")
```



The chart includes a plot of

- U controlling for positive drift, which is clearly happening in phase II.
- D controlling for negative drift.

0.6 EWMA chart

The Exponentially Weighted Moving Average (EWMA) is a statistic for monitoring the process, which averages the data in a way that gives most weight to recent data.

The EWMA is formally defined by

$$M_t = \lambda x_t + (1-\lambda)M_{t-1}, \quad t = 1, 2, \ldots, T$$

where

- M_0 is the mean of some historical data.
- x_t is the measurement at time t.
- T is the length of the sampling period.
- $0 < \lambda \leq 1$ is a smoothing parameter, where $\lambda = 1$ corresponds to "no memory".

The influence of x_t on M_{t+s} is of the order $(1-\lambda)^s$, i.e. exponentially decreasing, which explains the term "Exponentially Weighted".

0.7 EWMA chart

Estimated variance for the EWMA process

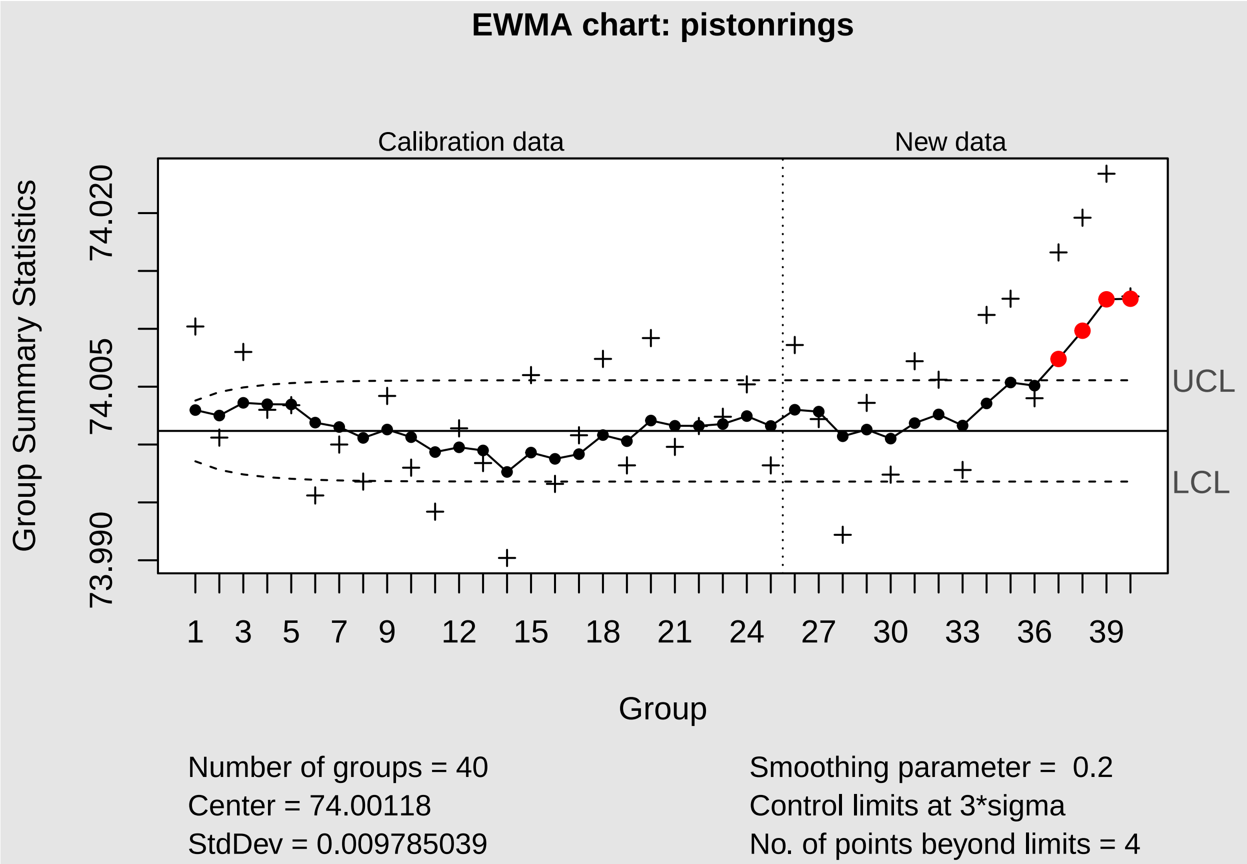
- $s_M^2 = \frac{\lambda}{2-\lambda} s^2$
- s is the standard deviation from historical data.

$$\begin{aligned} \text{UCL:} \quad & M_0 + k s_M \\ \text{CL :} \quad & M_0 \\ \text{LCL:} \quad & M_0 - k s_M \end{aligned}$$

Conventional choise of k is 3.

0.8 EWMA chart example

```
h <- ewma(phaseI, newdata = phaseII, title = "EWMA chart: pistonrings")
```



- Observed values are "plus" and predicted values are "dots".
- Default value of smoothing is 0.2. May be set by the argument `lambda`.
- Default value of k is 3. May be set by the argument `nsigmas`.

0.9 Multivariate charts

- In some situations , it is important to watch the covariation between two or more variables.
- We shall limit our considerations to two variables y and x.
- We assume a linear regression equation for y depending on x.
- We have data to estimate the line and the expected deviation from the line.
- We have estimates of mean and standard deviation for x.

An outlier would deviate from the line and/or the x-mean, so we calculate

- Zy_x denoting the standardized residual from the regression line.
- Zx denoting the standardized residual from the expected mean of x.

We expect that both Zy_x and Zx should be within the limits ± 2. In order to get an overall teststatistic, we calculate

$$T^2 = \frac{m-1}{m-2} Zy_x^2 + Zx^2$$

where m is sample size.

0.10 Multivariate charts

Alternatively, one might interchange the role of x and y. But that does not matter. Actually it holds that

$$T^2 = \frac{m-1}{m-2} Zy_x^2 + Zx^2 = \frac{m-1}{m-2} Zx_y^2 + Zy^2$$

- If the process is in control, then T^2 has a chi-square distribution with 2 degrees of freedom.
- It is critical to the process if T^2 is large, i.e. we only need an upper limit.

```
load(url("https://asta.math.aau.dk/datasets?file=T2Example.RData"))
```

```
head(X$X1, 3)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   72   84   79   49
## [2,]   56   87   33   42
## [3,]   55   73   22   60
```

0.11 Multivariate chart example

```
head(X$X2, 3)
```

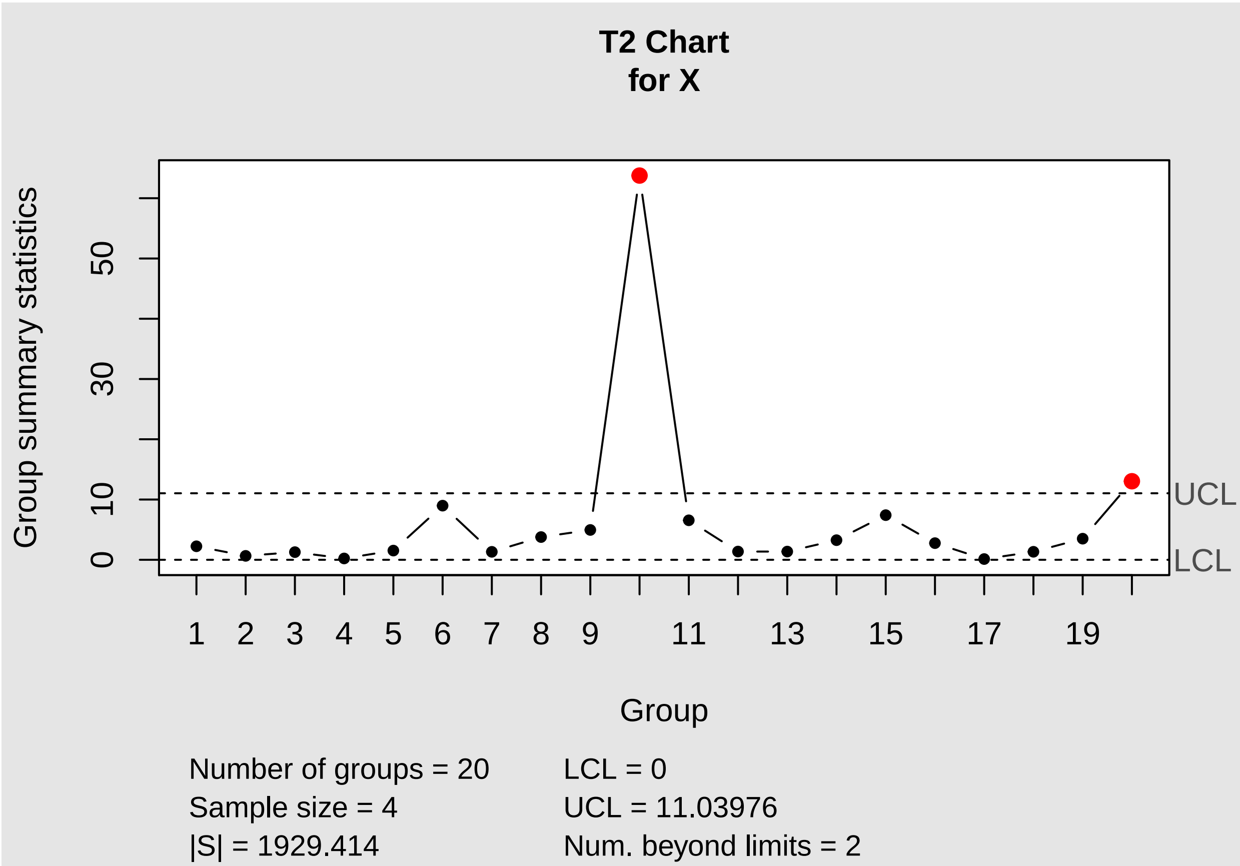
```
##      [,1] [,2] [,3] [,4]
## [1,]   23   30   28   10
## [2,]   14   31    8    9
## [3,]   13   22    6   16
```

- `X` is a list
- `X$X1` is a matrix, where the rows are samples of variable `X1`
- Actual sample size is 4 and the number of samples is 20.

Similarly `X$X2` has samples for variable `X2`

0.12 Multivariate chart example

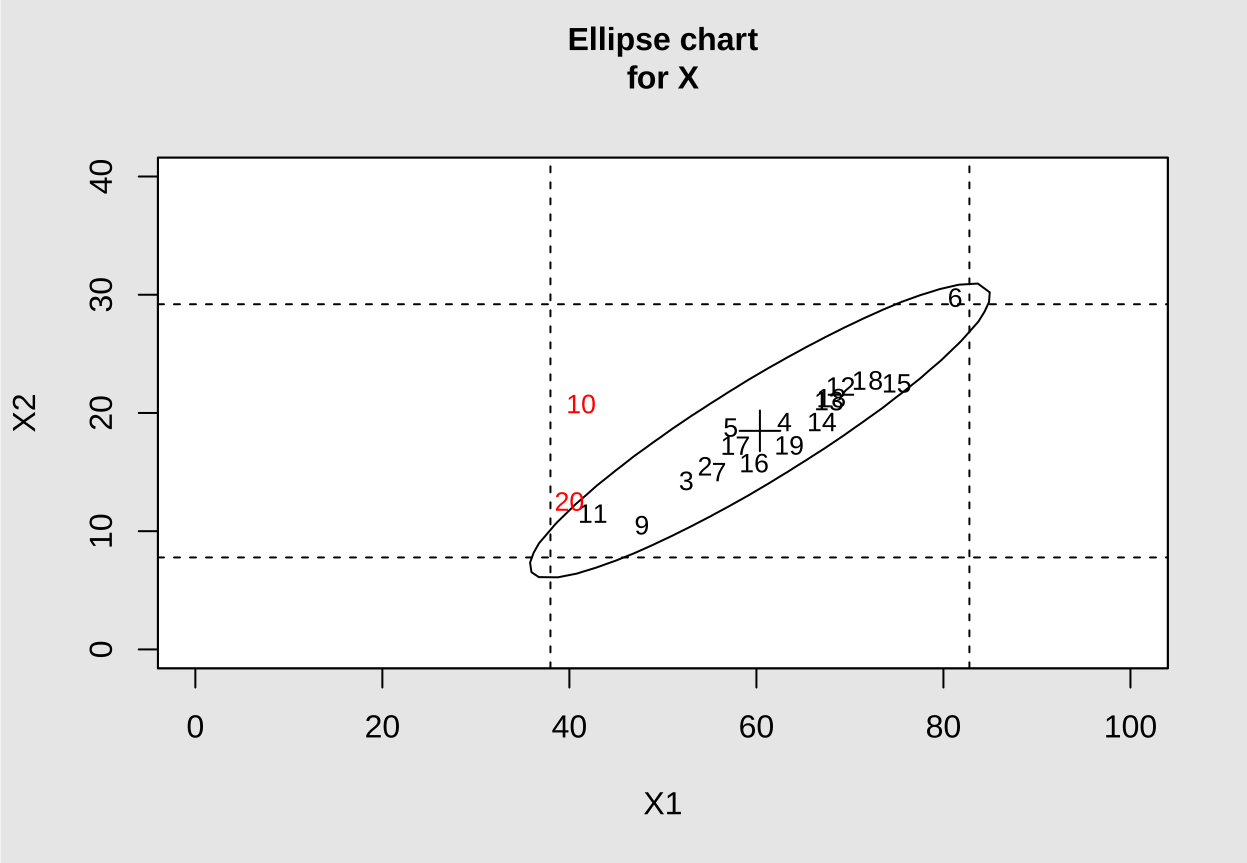
```
h <- mqcc(X, type = "T2")
```



When the parameters are estimated from phase I samples, then the reference distribution is not chi-square, but rather a scaled F-distribution.

0.13 Multivariate chart example

```
ellipseChart(h, show.id = TRUE)
```



Observation 10 deviates a lot from the regression line.

The acceptance area is an ellipse.

0.14 Acceptance sampling

Set-up:

- Production proces where we produce lots of size N.
- From the lot we take a sample of size n.
- If the number of defective items exceeds the number c, then the lot is rejected.

We term this a (N, n, c) sampling plan. Possible reasons for acceptance sampling:

- Testing is destructive
- The cost of 100% inspection is very high
- 100% inspection takes too long

0.15 Sampling distributions

- Let X denote the number of defective items in the sample.
- Let p denote the fraction of defective items in the lot.
- The correct sampling distribution of X is the so called **hypergeometric** distribution with parameters (N, n, p).

- If $N >> n$, then the sampling distribution of X is well approximated by the simpler **binomial** distribution with parameters (n, p).
- If $N >> n$ and p is smal, then the sampling distribution of X is well approximated by the much simpler **poisson** distribution with parameter np.

0.16 OC curve of a sampling plan

For a given sampling plan (N, n, c) the probability of accepting the lot depends on

- the fraction p of defective items in the lot
- the assumed sampling distribution

We can use the function `OC2c` in the package `AcceptanceSampling` to determine these probabilities.

Sampling plan: $(1000, 100, 2)$

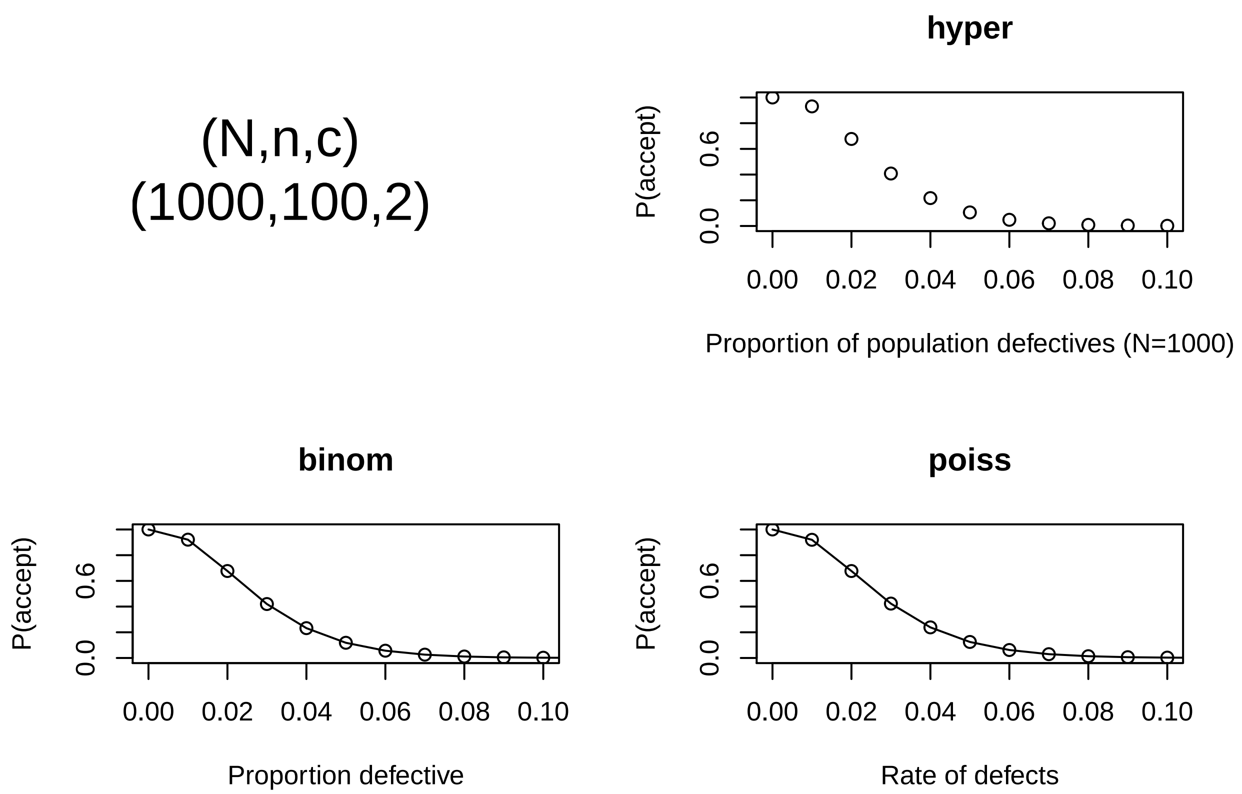
```
library(AcceptanceSampling)
```

```
## Loading required package: methods
```

```
OCbin <- OC2c(100, 2) #default binomial
OCpoi <- OC2c(100, 2, type = "poisson")
OChyp <- OC2c(100, 2, type = "hypergeom", N = 1000)
```

0.17 OC curve of a sampling plan

```
par(mfrow=c(2,2)) #division of plot window
plot(1:10, type = "n", axes = FALSE, xlab = "", ylab = "")
text(4, 5, "(N,n,c)\n(1000,100,2)", cex = 2)
xl <- c(0, 0.1)
plot(OChyp, xlim = xl, main = "hyper")
plot(OCbin, xlim = xl, main = "binom")
plot(OCpoi, xlim = xl, main = "poiss")
```



0.18 Find sampling plan

Suppose that N is fixed. If we specify 2 points on the OC curne, then this determines (n, c).

- PRP: Producer Risk Point with coordinates (p1,q1): p1 is a fraction of defectives, that the producer finds acceptable, e.g. p1=0.01. The corresponding probability q1 of accept should then be high, e.g. q1=0.95.
- CRP: Consumer Risk Point with coordinates (p2,q2): p2>p1 is a fraction of defectives, that the consumer finds unacceptable, e.g. p2=0.05. The corresponding probability q2 of accept should then be low,e.g. q2=0.01.

```
find.plan(PRP = c(.01,.95), CRP = c(.05,.01))[1:2]
```

```
## $n
## [1] 259
##
## $c
## [1] 5
```

Default assumption is binomial sampling.

0.19 Find sampling plan

```
plan <- find.plan(c(.01,.95), c(.05,.01), type = "hyp", N = 200)[1:2]
plan
```

```
## $n
## [1] 121
##
## $c
## [1] 2
```

```
OChyp <- OC2c(plan$n, plan$c, type = "hyp", N = 200, pd = c(.01,.05))
attr(OChyp, "paccept")
```

```
## [1] 1.000000000 0.009609746
```

We cannot have an exact match of the required values $(0.95, 0.01)$ since n and c must be integers.

0.20 Double sampling

Let $0 \leq c_1 < r_1$ be integers.

Furthermore, c_2 is an integer such that $c_1 < c_2$.

- x_1: number of defectives in an initial sample of size n_1.
- If $x_1 \leq c_1$ accept the lot.
- If $r_1 \leq x_1$ reject the lot.
- If $c_1 < x_1 < r_1$: Take a second sample of size n_2 and let x_2 be the number of defectives.
- If $x_1 + x_2 \leq c_2$ accept the lot. Otherwise reject.

This is known as a **double sampling** plan.

0.21 OC curve of a double sampling plan

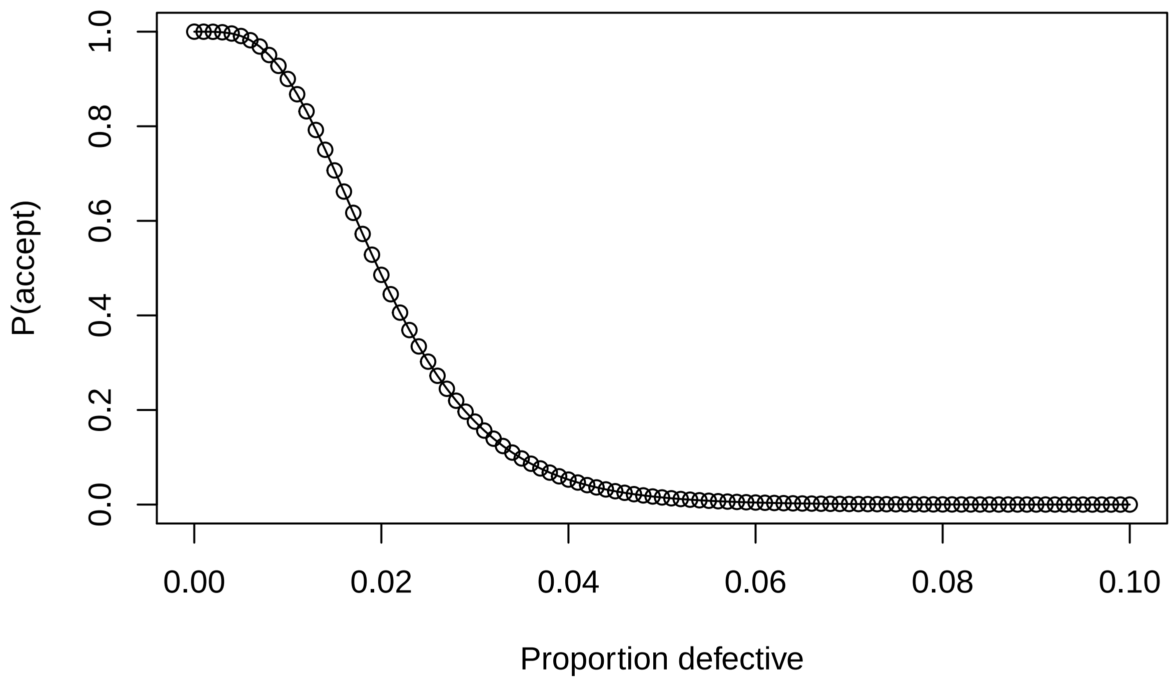
Determining the OC curve of a double sampling plan requires input of $n = c(n_1, n_2)$, $c = c(c_1, c_2)$ and $r = c(r_1, r_2)$, where $r_2 = c_2 + 1$.

```
x <- OC2c(c(125,125), c(1,4), c(4,5), pd = seq(0,0.1,0.001))
x
```

```
## Acceptance Sampling Plan (binomial)
##
##                Sample 1 Sample 2
## Sample size(s)      125      125
## Acc. Number(s)        1        4
## Rej. Number(s)        4        5
```

0.22 OC curve of a double sampling plan

```
plot(x)
```


P(accept)
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.02
0.04
0.06
0.08
0.10
Proportion defective