Package `qmrobust` for classical and robust quality management R-functions (Draft)

Werner A. Stahel, ETH Zurich

January 26, 2015

Abstract

The package `qmrobust` presently is a companion to the chapter "Statistical Procedures for Performance-based Specification and Testing" of ...

It currently contains 3 datasets and a function for analyzing interlaboratory studies. We plan to implement some more classical and robust methods for quantile estimation and interlaboratory studies.

1 Introduction

Classical methods for quality management include, besides elementary one- and two-sample tests, control charts and the analysis of interlaboratory studies. They are available through existing packages in R.

This vignette is a companion of the chapter "Statistical Procedures for Performance-based Specification and Testing" of ... which presents procedures for quality management in the concrete producing industry. The package `qmrobust` presently consists of the data sets used in that text and shows how the methods are obtained in R.

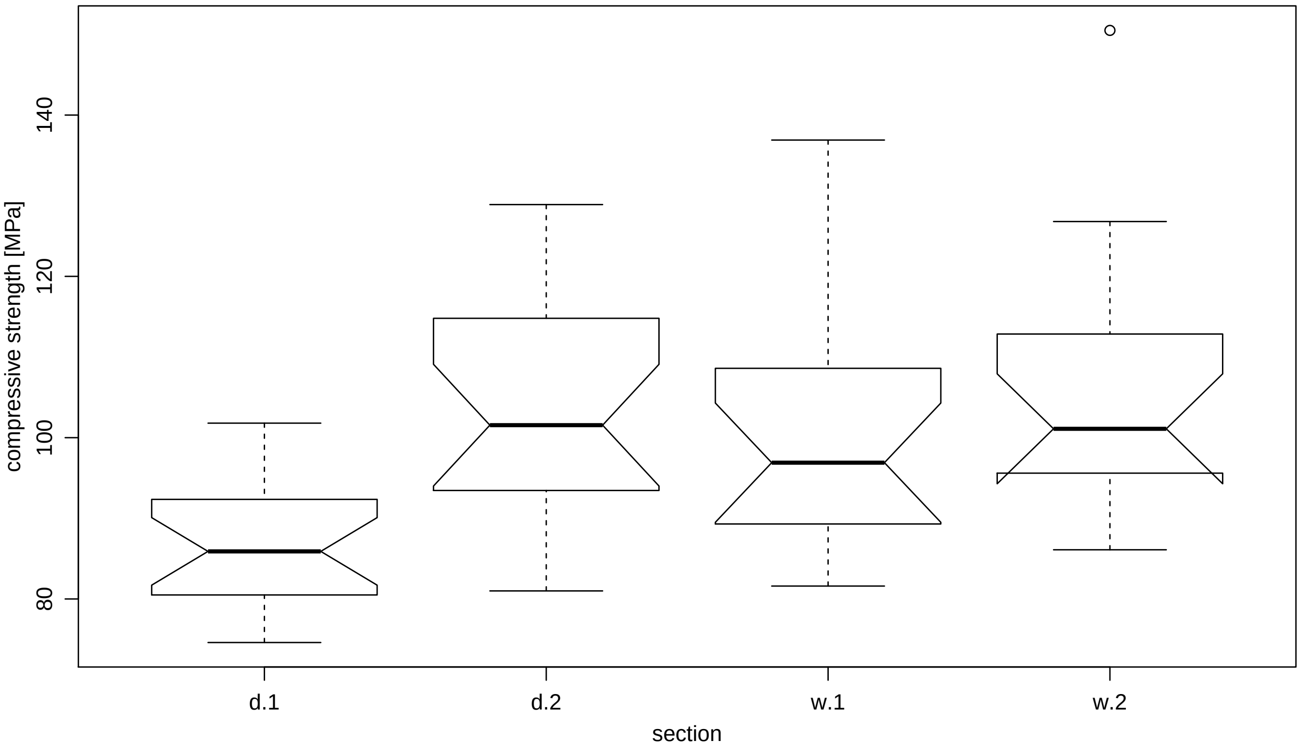
```
> require(qmrobust, lib="../../pkg.Rcheck")
```

2 Datasets

a **Datasets `d.tunnel1` and `d.tunnel2`.** Concrete properties (compressive strength, permeability and porosity) in selected structural components of a new opencast tunnel have been measured and the spatial variability determined Leemann, Hoffmann, Malioka and Faber, 2006. 400 cores were taken from two deck and two wall elements, and different characteristics were measured.

```
> data(d.tunnel1)
> showd(d.tunnel1)

dim:  240 9
    section diameter  n position   layer strength density  perm.O2 cond.CL
1       w.1       68 65        A layer.1    102.2    2428 9.12e-11    1.14
2       w.1       68 65        A layer.2     90.0    2380 8.91e-11    0.98
3       w.1       68 65        A layer.3     92.0    2380 1.17e-10    1.14
...
62      w.2       68 65        A layer.2    107.3    2435 6.35e-11    0.81
122     d.1       68 65        A layer.2     67.3    2350 6.37e-11    1.13
181     d.2       68 65        A layer.1    104.5    2367 4.68e-11    1.00
240     d.2       68 86        V layer.3    115.5    2371 1.74e-11    0.85
```

```
> ## ?d.tunnel1 ## help information, not shown here

> dd <- d.tunnel1[d.tunnel1$layer=="layer.3",]
> plot(strength~section, data=dd, notch=T, ylab="compressive strength [MPa]")
```

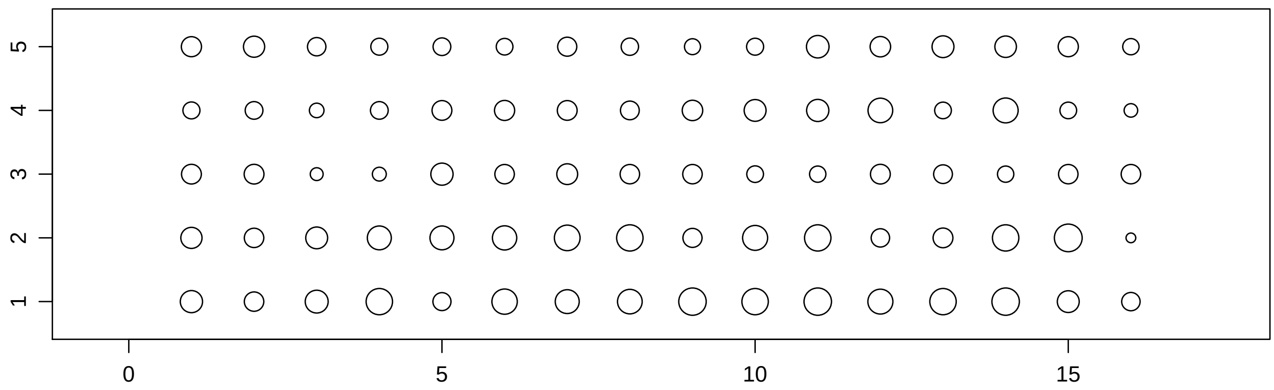


b **Dataset `d.perm`.** The results of a round robin of the air permeability test (SIA Standard 262/1, Annex E) are reported in Jacobs, Leemann, Denarié and Teruzzi, 2009. On two selected elements of a bridge a regular grid of 75 identically sized square areas was delineated. The 75 areas were randomly assigned to 5 participating teams, so each team had to measure permeability in 15 areas each for both elements.

```
> data(d.perm)
> showd(d.perm)

dim:  150 5
    team section rep permeability perm.log
1      S     w.1   0     1.02e-17    -17.0
2      S     w.1   1     1.79e-17    -16.7
3      S     w.1   2     1.76e-17    -16.8
...
40     H     w.1   9     8.20e-18    -17.1
76     L     w.2   0     1.07e-17    -17.0
113    E     w.2   7     1.03e-17    -17.0
150    T     w.2  14     1.85e-17    -16.7
```

c **Dataset `d.coverdepth`.** A third example is taken from Monteiro, Gonçalves and Gulikers, 2014, Fig. 9.4 and features a 5×40 grid of cover depth readings obtained in the top reinforcement layer of the deck slab of a freeway viaduct.

```
> data(d.coverdepth)
> dd <- d.coverdepth[d.coverdepth$section<=2,]
> symbols(dd$column,dd$row, circles=dd$depth, inches=0.7*par("cin")[1],
+         xlab="",ylab="")
```



3 Specification testing

a **Inference on a location parameter.** We assume here that the targetted property of the material, e.g., compressive strength, has a known distribution, usually the normal distribution. Furthermore, the specification is given as a targeted expected value. Thus, we require inference tools for this parameter.

b **Classical one-sample t-test and confidence interval.** The most convenient way to check if a given specification is fulfilled is provided by a confidence interval for the parameter. It is obtained from the function `t.test`.

```
> ( dd <- d.tunnel1[d.tunnel1$section=="w.1","strength"] )

 [1] 102.2  90.0  92.0 101.7 102.7 122.7 109.1  90.3  85.9 121.0 113.5  86.8
[13] 100.5  90.1 101.0  86.4  81.2 115.6 115.4  92.3  94.9  86.3  94.7 104.3
[25] 100.0 103.4  96.9  89.6  86.6    NA 100.2  95.4  89.3 128.1 110.5 108.6
[37]  82.3  95.9  81.6  93.7  98.8  94.8  95.1 103.9  85.6  86.8 116.9    NA
[49] 108.1  94.7    NA 110.8 117.5 131.3 103.8 107.6 106.9  95.3 103.8 136.9

> t.test(dd)

	One Sample t-test

data:  dd
t = 59.5, df = 56, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
```

```
  97.3 104.1
sample estimates:
mean of x
      101
```

The estimated strength is 101 and the confidence interval, $[97.3, 104.1]$. If the specification were 100, it is perfectly fulfilled.

c **Nonparametric test.** The assumption of a normal distribution is not needed in this simple problem, and should be avoided. The corresponding nonparametric test, the signed rank test of Wilcoxon, provides inference on the center of symmetry of the distribution.

```
> dd <- d.tunnel1[d.tunnel1$section=="w.1","strength"]
> wilcox.test(dd, conf.int=T)

        Wilcoxon signed rank test with continuity correction

data:  dd
V = 1653, p-value = 5.28e-11
alternative hypothesis: true location is not equal to 0
95 percent confidence interval:
  96.5 103.5
sample estimates:
(pseudo)median
          99.6
```

The center of symmetry is estimated as 99.6, with a confidence interval from 96.5 to 103.45.

d **Outlier rejection.** The most popular tests for outliers is Grubbs' tests. They are all based on the sorted, standardized observations $X_{[i]}$, $R_i = (X_{[i]} - \overline{X})/S$, where S is the estimated standard deviation.

The first test is intended to detect single outliers at either the low or the high end of the sample. It is based on the test statistic $T_1 = \max\langle -R_1, R_n \rangle$. The second test is designed to detect the case of two outliers on the same side. The test statistic is $T_2 = \max\langle R_1^2 + R_2^2, R_{n-1}^2 + R_n^2 \rangle$. The third test should pinpoint two outliers, one at each end, by calculating $T_3 = R_n - R_1$. The distribution of these test statistics has been derived in the literature.

The package `outliers` contains a function `grubbs.test` which performs three types of this test. A peculiar feature of the function is its default value `FALSE` for the argument `two.sided`. Because the side on which the outliers can occur is rarely known a priori, the argument should be set `TRUE`, except for `type=11`, for which this setting produces nonsense.

```
> require(outliers)
> dd <- d.tunnel1[d.tunnel1$section=="w.1","strength"][1:10]
> grubbs.test(dd, type=10, two.sided=TRUE) ## one outlier on either side?

        Grubbs test for one outlier

data:  dd
G = 1.633, U = 0.671, p-value = 0.8288
alternative hypothesis: highest value 122.7 is an outlier

> grubbs.test(c(dd,60), type=10, two.sided=TRUE)
```

```
        Grubbs test for one outlier

data:  c(dd, 60)
G = 2.168, U = 0.483, p-value = 0.1388
alternative hypothesis: lowest value 60 is an outlier

> grubbs.test(c(dd,60), type=11)

        Grubbs test for two opposite outliers

data:  c(dd, 60)
G = 3.581, U = 0.324, p-value = 0.1683
alternative hypothesis: 60 and 122.7 are outliers

> grubbs.test(dd, type=20, two.sided=TRUE)

        Grubbs test for two outliers

data:  dd
U = 0.317, p-value = 0.2758
alternative hypothesis: highest values 121 , 122.7 are outliers

> grubbs.test(c(dd[1:10],60), type=20, two.sided=TRUE)

        Grubbs test for two outliers

data:  c(dd[1:10], 60)
U = 0.392, p-value = 0.3918
alternative hypothesis: lowest values 60 , 85.9 are outliers
```

4 Conformity testing.

a **Quantile estimation.** Quality measures have a natural variability across the object (brigde, tunnel, ...) to be assessed. Therefore, it will not be sufficient to specify a requirement in terms of the expected value. Rather, a "reasonable minimum" should be prescribed, in the form: "The quality criterion X shall be larger than c with a probability $1-\gamma$, i.e., $P(X < c) < \gamma$. Equivalently, the γ quantile q_γ should be $\geq c$.

Since the decision whether confirmity is warranted or not must be based on a sample, the random nature of the estimators of $P(X < c)$ or q_γ should be taken into account as discussed below.

b **Nonparametric inference.** The most direct way to assess $P(X < c)$ is based on the number K of observations for which $X_i < c$. It has a binomial distribution, $K \sim \mathcal{B}\langle n, p\rangle$. Conformity testing either means testing $p \geq \gamma$ against $p < \gamma$ or determining a one-sided confidence interval for p and checking if the upper bound is $\leq \gamma$. We do both for the cover depth data and a threshold of $c = 45\,\mathrm{mm}$, which is required to be failed with less than $\gamma = 10\%$ probability.

```
> ##- hist(d.coverdepth$depth)  # recommended, but not done here.
> threshold <- 45
> gamma <- 0.1
> x <- sum(d.coverdepth$depth < threshold)
> binom.test( x, nrow(d.coverdepth), p=gamma, alternative="less")

        Exact binomial test

data:  x and nrow(d.coverdepth)
number of successes = 5, number of trials = 200, p-value = 3.871e-05
alternative hypothesis: true probability of success is less than 0.1
95 percent confidence interval:
 0.0000 0.0518
sample estimates:
probability of success
                 0.025
```

Thus, with $n = 200$ observations, the conformity test for $c = 40\,\mathrm{mm}$ was successful, since the p-value is $3.87e-05 < \gamma$, or since the upper bound of the confidence interval, 0.0518, is $< \gamma$. The estimated probability of a lower cover depth was $\widehat{p} = 0.025$.

Such a large sample may be realistic for easy, non-destructive measurements. For other situations, an alternative assessment of conformity is needed.

c **Quantile estimation for normal data with known scale.** If the normal distribution $\mathcal{N}\left\langle \mu, \sigma^2 \right\rangle$ is assumed for the data, then the quantile equals $q_\gamma = \mu + q_\gamma^{(0,1)}\sigma$, where $q_\gamma^{(0,1)}$ is the γ quantile of the standard normal distribution. If the precision of the measurements, expressed by σ, is known – possibly from many similar studies –, then this leads to the estimated quantile $\widehat{q}_\gamma = \bar{x} + q_\gamma^{(0,1)}\sigma$. For the data used before and $\sigma = 15\,\mathrm{MPa}$,

```
> dd <- d.tunnel1[d.tunnel1$section=="w.1","strength"]
> gamma <- 0.02
> sigma <- 15
> mean(dd, na.rm=TRUE) + qnorm(gamma)*sigma

[1] 69.9
```

For drawing inference, the easiest way is to check if the expected value μ is larger than $c + q_\gamma^{(0,1)}\sigma$ as discussed above.

d **Normal data with unknown scale.** ???

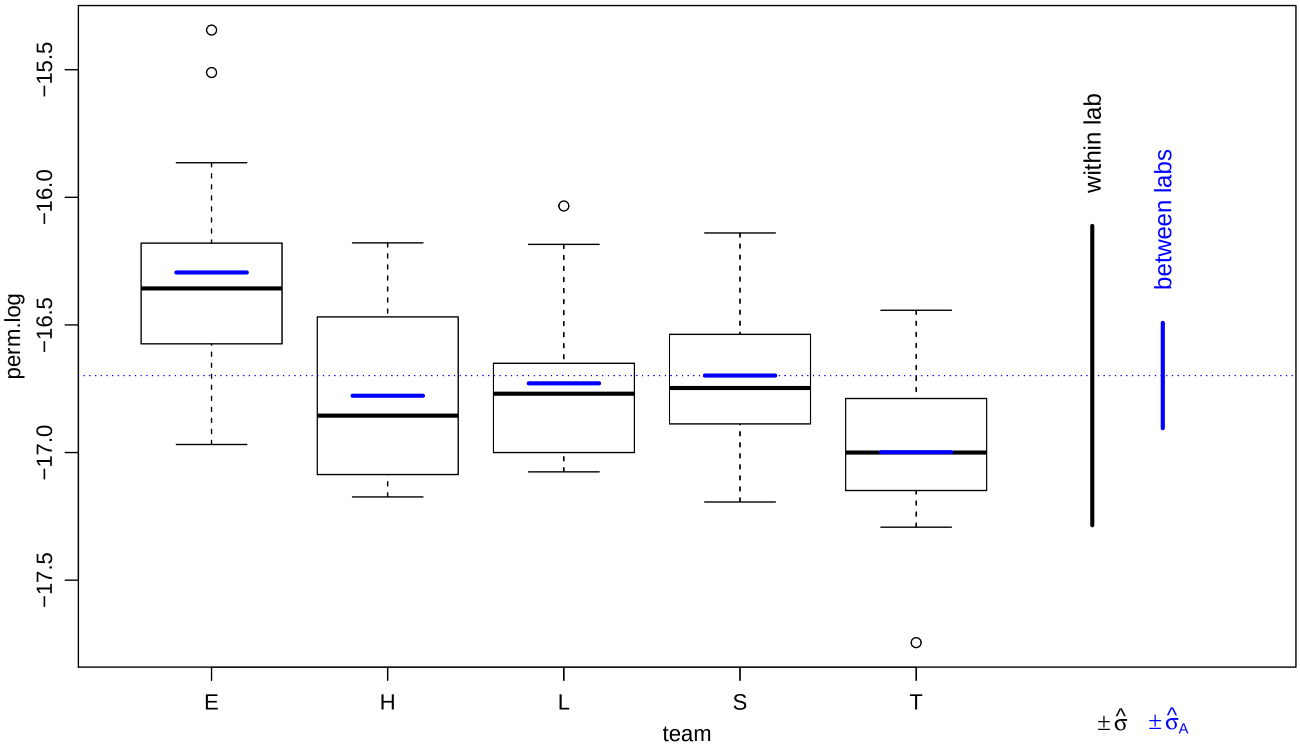
5 Interlaboratory studies

a **Standard procedure.** The classical analysis of interlaboratory studies is based on a one-way analysis of variance. Outlier tests are usually applied to eliminate extreme observations within a lab or the entire set from a lab which deviates extremely from the other labs.

The package provides the function `interlabstats`, which does not (yet) implement any outlier testing or rejection. It comes with `print` and `plot` methods.

For obtaining an instructive example, the data for team E in `d.perm` have been changed, because otherwise, the between groups variance is estimated as zero, which is an atypical case.

```
> dd <- d.perm[d.perm$section=="w.1",]
> dd[dd$team=="E","perm.log"] <- dd[dd$team=="E","perm.log"]+0.5
> ( r.cl <- interlabstats(perm.log~team, data=dd) )

interlabstats.formula(formula = perm.log ~ team, data = dd)
   method =  classical
   overall mean  =  -16.7  ; sigma =  0.586  ; sd.between =  0.207
   repeatability =  1.66  ; reproducibility =  1.76

 groups:
   n  mean    sd
E 15 -16.3 0.669
H 14 -16.8 0.575
L 15 -16.7 0.564
S 15 -16.7 0.549
T 15 -17.0 0.567

> plot(r.cl)
```



b **Mixed model analysis.** A more modern way of analyzing interlaboratory studies relies on maximum likelihood estimation of the parameters of the model. This is achieved by the R-function `lmer` of the package `lme4`. The results can be fed into `interlabstats` to get the desired quantities.

```
> require(lme4)
> r.lme <- lmer(perm.log~(1|team), data=dd, na.action=na.omit)
> summary(r.lme)

Linear mixed model fit by REML ['lmerMod']
Formula: perm.log ~ (1 | team)
   Data: dd

REML criterion at convergence: 65.6

Scaled residuals:
   Min     1Q Median     3Q    Max
-2.251 -0.768 -0.134  0.527  2.874

Random effects:
 Groups   Name        Variance Std.Dev.
 team     (Intercept) 0.0571   0.239
 Residual             0.1211   0.348
Number of obs: 74, groups:  team, 5

Fixed effects:
            Estimate Std. Error t value
(Intercept)  -16.699      0.114    -146
```

```
> ## ---------------------
> interlabstats(r.lme)

lmer(formula = perm.log ~ (1 | team), data = dd, na.action = na.omit)
   method =  lmerMod
   overall mean  =  -16.7  ; sigma =  0.348  ; sd.between =  0.0571
   repeatability =  0.984  ; reproducibility =  0.997
```

c **Robust mixed model analysis.** The classical norms based on the rejection of outlying observations and labs lead to biased results. (Corrections would in principle be possible but are not worked out according to the knowledge of the authors.) An alternative is to use robust estimators, which avoid the rejection of data, but deal with outliers by downweighting their influence on the results adequately. A robust version of the maximum likelihood method is obtained from the package `robustlmm`, function `rlmer`.

```
> require(robustlmm)
> r.lmmrob <- rlmer(perm.log~(1|team), data=dd)
> save(r.lmmrob,file="r.lmmrob.rda")
> #load("r.lmmrob")
> ## summary(r.lmmrob)
> interlabstats(r.lmmrob)

rlmer(formula = perm.log ~ (1 | team), data = dd)
   method =  rlmerMod
   overall mean  =  -16.7  ; sigma =  0.334  ; sd.between =  0.0486
   repeatability =  0.946  ; reproducibility =  0.956
```

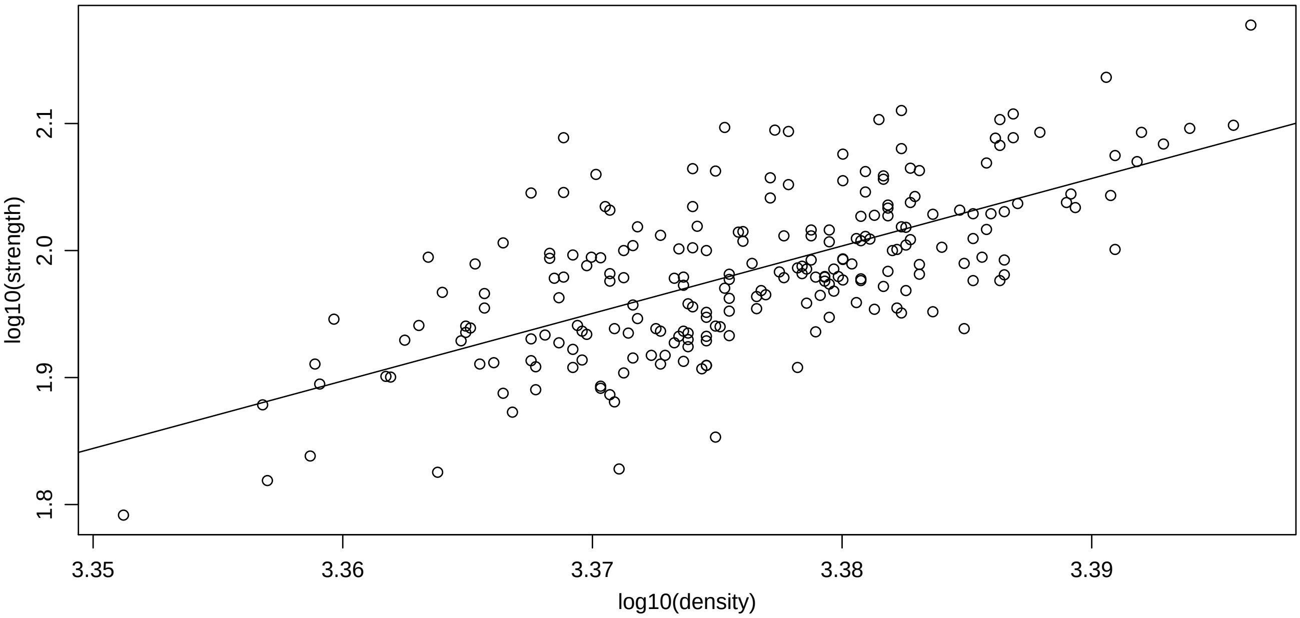
In our example, the robust and nonrobust mixed model estimators are quite similar but different from the classical one:

6 Regression

a **Fitting the model.** Linear regression models are fitted by the R function `lm`.

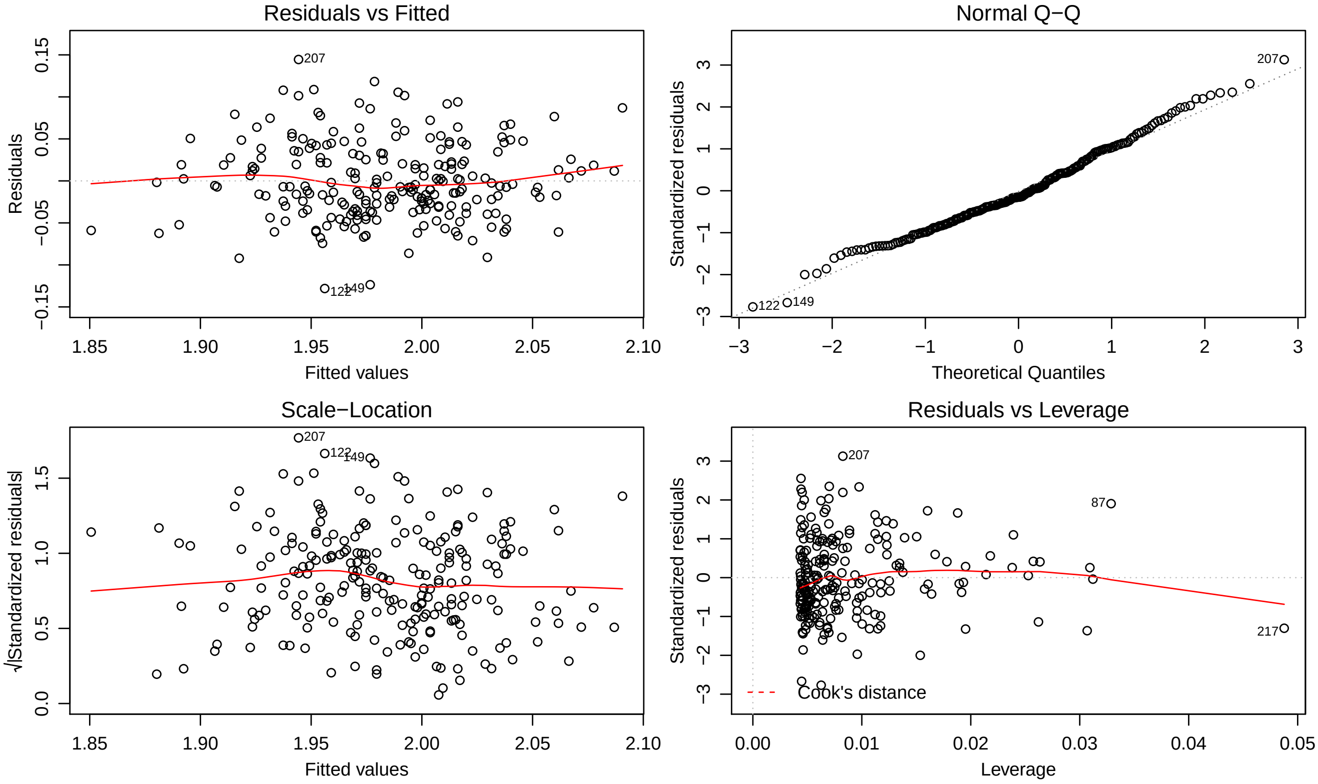
```
> r.lm <- lm(log10(strength) ~ log10(density), data=d.tunnel1)
> options(show.signif.stars=FALSE)
> summary(r.lm) ## results of the fitting

Call:
lm(formula = log10(strength) ~ log10(density), data = d.tunnel1)

Residuals:
     Min       1Q   Median       3Q      Max
-0.12812 -0.03123 -0.00701  0.02970  0.14453

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)      -15.96       1.32   -12.1   <2e-16
log10(density)     5.31       0.39    13.6   <2e-16
```

```
Residual standard error: 0.0464 on 228 degrees of freedom
  (10 observations deleted due to missingness)
Multiple R-squared:  0.449,       Adjusted R-squared:  0.446
F-statistic:  185 on 1 and 228 DF,  p-value: <2e-16

> plot(log10(strength) ~ log10(density), data=d.tunnel1)  ## scater plot
> abline(r.lm)  ## draw the estimated regression line
```



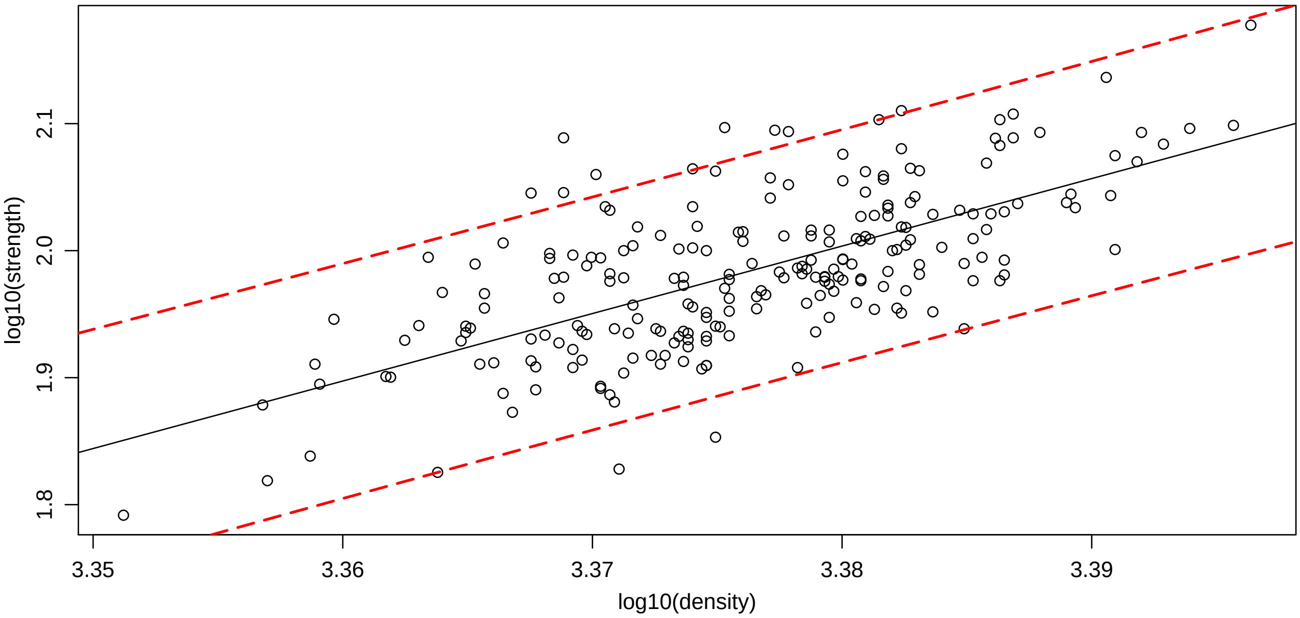
b **Residual analysis.**

```
> par(mfrow=c(2,2), mar=c(3,3,2,1), mgp=c(2,0.8,0))
> plot(r.lm)
```



c **Prediction.**

```
> plot(log10(strength) ~ log10(density), data=d.tunnel1)  ## scater plot
> abline(r.lm)  ## draw the estimated regression line
> lusr <- par("usr")
> x <- seq(lusr[1],lusr[2],length=51) ## equally space values along x
> xdf <- data.frame(density=10^x)
> r.pred <- predict(r.lm, newdata=xdf, interval="prediction")
> matlines(x,r.pred[,2:3], lty=5, col="red", lwd=2)
```



References

Jacobs, F., Leemann, A., Denarié, E. and Teruzzi, T. (2009). Recommendations for the quality control of concrete with air permeability measurements, *Technical Report 641*, Bundesamt für Strassen (ASTRA), Bern, Switzerland.

Leemann, A., Hoffmann, C., Malioka, V. and Faber, M. (2006). Variability of concrete properties in structures, *Technical Report 611*, Bundesamt für Strassen (ASTRA), Bern, Switzerland.

Monteiro, A. V., Gonçalves, A. and Gulikers, J. (2014). Statistical bases for assessing the cover depth in reinforced concrete structures, *Performance-Based Specifications and Control of Concrete Durability*, RILEM TC 230-PSC, RILEM.