

masscor package

Cristhian Paredes

2021-09-09

The R package `masscor` provides functions, classes and methods to support mass measurements using non automatic balances as described in EURAMET (2015). The new classes are objects that can store the calibration information for balances and mass standards. Those objects can be used to convert balance readings to both conventional mass and mass, and to perform routine balance verification (by using the normalized error function). Air buoyancy correction factors are calculated using local air density that can be calculated using measured environmental conditions and applying one of several models available in the package. The uncertainty of (corrected) mass measurements can also be evaluated allowing to further asses the suitability of a given mass measurement.

This package contains example data sets with real and hypothetical calibration data for some balances, a single mass standard and a mass standards kit. This vignette is intended as a user manual for the `masscor` package.

0.1 Installation and use

The last released version of the package `masscor` can be installed from CRAN by running:

```
install.packages("masscor")
```

The development version of `masscor` can be installed from GitHub using `install_github()` function from `devtools` package (Wickham, Hester, and Chang 2019):

```
devtools::install_github("Crparedes/masscor", build_vignettes = TRUE)
```

To use the functions in `masscor`, the package must be loaded into R environment after the installation by running:

```
library(masscor)
```

0.2 Balance calibration certificates using the `masscor` package

The calibration of a balance involve tests to determine the repeatability and the error of indications in several points spread along the measurement interval of the balance, and the effect of applying the load in eccentric positions of the balance pan (EURAMET 2015).

The repeatability test consist in taking several balance readings for the same object under similar conditions to asses short time variability. This process can be repeated with other objects of different mass to evaluate the repeatability at different balance loads. The errors of the indications are obtained by comparing balance readings to the masses of mass standards that are traceable to internationally recognized references (traceable to the International System of Units). The eccentricity effect is determined by the variation in balance readings for the same object when intentionally placed at different positions of the weighing pan.

The calibration certificate of a balance will typically include information about the balance division scale which is usually also indicated somewhere in the body of the balance (readability), a table of repeated indications for one or more objects (repeatability), a table with indication errors and associated uncertainties for several mass readings in the measurement interval of the balance, and a table with the differences in the readings for the same object located at several positions in the balance pan (eccentricity).

The indication errors can later be used to correct a given balance reading to a conventional mass value with metrological traceability (see `convMass()`). The uncertainties of the indication errors provide information to estimate the uncertainty in the conventional mass due to this correction.

The package `masscor` uses objects of class `calibCert` to store the information of balance calibration certificates. This class of objects are generated by the function `calibCert()`. Mandatory arguments for this function are the balance division scale, the results of the indication error test, the results of repeatability test, and the results of the eccentricity test:

- Balance division scale is provided at the parameter `d`. The corresponding units can be indicated at the parameter `d.units` (default value is `'mg'`).
- The results of the indication error test are provided as a data frame at the parameter `indError`. The data frame must have three columns containing nominal masses, indication errors and associated uncertainties for at least two mass standards. The corresponding units can be indicated at the parameter `indError.units` as a character vector of length three (one element for nominal masses, indication errors and associated uncertainties, respectively). Default values are `c('g', 'mg', 'mg')`. The uncertainties are taken by default as expanded uncertainties (parameter `expanded = TRUE` by default) and a coverage factor can be indicated at the parameter `k` (default `k = 2`).
- The repeatability test results are provided at the parameter `rep` and can either be a vector of length two with the nominal mass (mean) and standard deviation of the balance readings under repeatability conditions for the same object, or in case the repeatability test is performed at more than one point, a data frame with the nominal masses (in the first column) and standard deviations (in the second column) of the balance readings under repeatability conditions for each object. The corresponding units can be indicated at the parameter `rep.units` as a vector of length two (for the nominal mass and for the standard deviation, respectively). Default values are `c('g', 'mg')`.
- The eccentricity test results is provided at the parameter `eccen` as a vector of length two indicating the load and the maximal difference in the balance readings for the same load placed at different positions. The corresponding units can be indicated at the parameter `eccen.units` as a vector of length two (for the load and the maximal difference, respectively). Default values are `c('g', 'mg')`.

The default values for the parameters are those typical for an analytical balance calibrated at the INM. If the units in the calibration certificate being used are different but are not indicated when creating the object of class `calibCert`, significant miscalculations will occur down the pipe when applying corrections and estimating uncertainties.

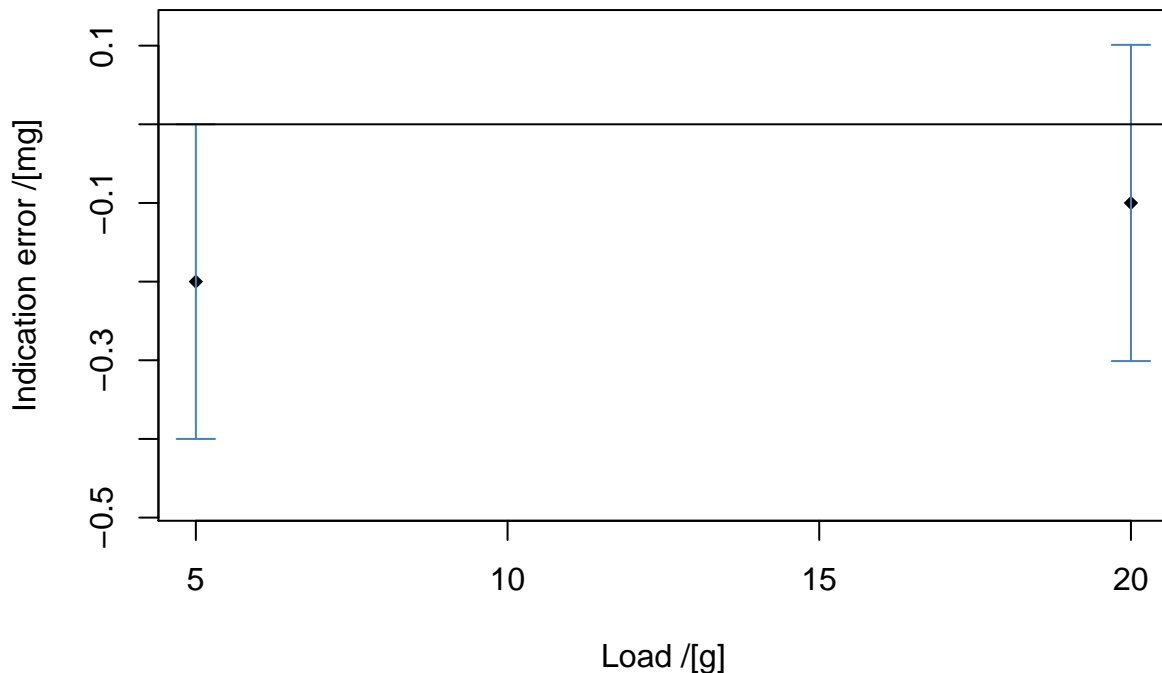
Suppose the simplest case in which an analytic balance (scale division of 0.1 mg) is calibrated using only two (traceable) mass standards of nominal masses 5 and 20 g. The indication errors are -0.2 ± 0.200 and -0.1 ± 0.205 mg, respectively. The standard deviation of six readings under repeatability conditions for the mass standard of 20 g is 0.02 mg, and the maximal difference in the eccentricity test was 0.1 mg for a balance load of 100 g. This information is enough to produce an object of class `calibCert` that can later be used to correct a balance reading in the interval between 5 and 20 g and to later estimate mass measurement uncertainties. A balance identification string can be provided indicating, for example, the brand, type and location of the balance.

```
Bal.1.Lab.317 <- calibCert(balanceID = '(Brand) Analytic balance lab 317',
  d = 0.1, d.units = 'mg',
  indError = data.frame(nominal = c(5, 20),          # grams
                        error   = c(-0.2, -0.1),    # milligrams
                        uncert  = c(0.200, 0.201)), # milligrams
  indError.units = c('g', 'mg', 'mg'),
  rep = c(20, 0.02), rep.units = c('g', 'mg'),
  eccen = c(100, 0.1), eccen.units = c('g', 'mg'))
```

The package provides methods to `print` and `plot` the information contained in objects of class `calibCert`. Plotting a calibration certificate will show the graph of indication error versus balance reading as typically included in balance calibration certificates. Printing the object will show summarized information of the

provided data.

```
print(Bal.1.Lab.317)
#> BALANCE CALIBRATION DATA: (Brand) Analytic balance lab 317
#> Balance default units: [g]
#> Balance division scale: 0.1 [mg]
#>
#> Repeatability results:
#> Standard deviation for a balance load of 20 [g] was 0.02 [mg]
#>
#> Eccentricity:
#> Maximal difference for a balance load of 100 [g] was 0.1 [mg]
#>
#> Error of indication:
#> . Load [g] Error [mg] Uncertainty [mg]
#> 5 -0.2 0.200
#> 20 -0.1 0.201
#> (Uncertainties are expanded with a coverage factor k = 2)
#>
#> Use 'print(x, complete = TRUE)' to show additional information
#> contained in the object of class 'calibCert'
plot(Bal.1.Lab.317)
```



The objects of class `calibCert` shall be created when the balance firstly undergoes calibration, and must be updated each time the balance undergoes recalibration. The best way to share the `calibCert` objects among the people that uses the balance is by saving the `calibCert` object as a `.RData` file using the R function `save()`. When required, the `calibCert` object can be loaded into the R Environment by using the function

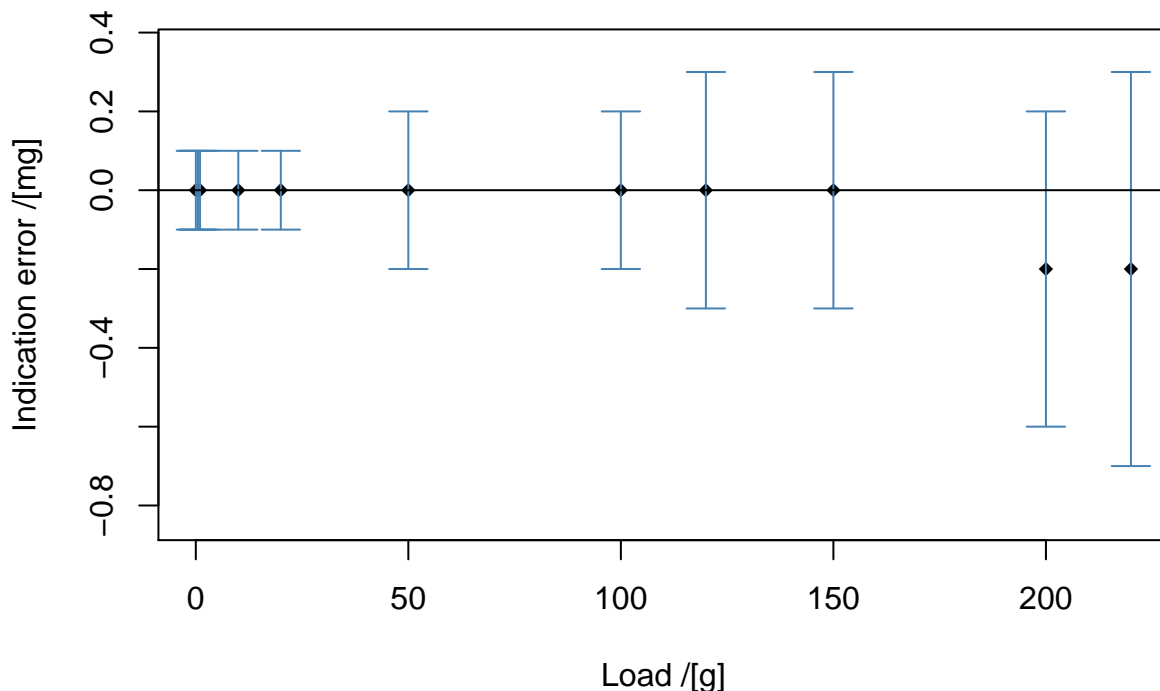
```
load().
```

```
save(Bal.1.Lab.317, file = 'certificateBalance.1_Lab.317.RData')  
load(file = 'certificateBalance.1_Lab.317.RData')
```

The directory to/from which the .RData file is going to be saved/loaded must be kept in mind. See `help(getwd)` for further details.

Additional information of the calibration certificate can be set in the optional parameters of the function `calibCert()` (run `help(calibCert)` for details). Some balance calibration certificates provided as example data sets in the package illustrate the purpose of those optional parameters.

```
data(MT.XPE.204)  
print(MT.XPE.204, complete = TRUE)  
#> BALANCE CALIBRATION DATA: MT XPE 204  
#>   Balance default units: [g]  
#>   Balance division scale: 0.1 [mg]  
#>  
#>   Repeatability results:  
#>     . Load [g] Std Dev [mg]  
#>         0.1      0.00  
#>        100.0     0.04  
#>        220.0     0.03  
#>  
#>   Eccentricity:  
#>     Maximal difference for a balance load of 100 [g] was 0.1 [mg]  
#>  
#>   Error of indication:  
#>     . Load [g] Error [mg] Uncertainty [mg]  
#>         0.01      0.0      0.1  
#>         0.50      0.0      0.1  
#>         1.00      0.0      0.1  
#>        10.00      0.0      0.1  
#>        20.00      0.0      0.1  
#>        50.00      0.0      0.2  
#>       100.00      0.0      0.2  
#>       120.00      0.0      0.3  
#>       150.00      0.0      0.3  
#>       200.00     -0.2      0.4  
#>       220.00     -0.2      0.5  
#>     (Uncertainties are expanded with a coverage factor k = 2)  
#>  
#>   Additional information  
#>     Calibration laboratory: Instituto Nacional de Metrologia de Colombia  
plot(MT.XPE.204)
```



0.3 Conventional mass corrections

We live and perform mass measurements in air. The air causes the objects to float and this buoyancy affects the measurement of their masses. When a balance is calibrated the conventional mass of several mass standards is compared to their balance readings and this information can later be used to correct a given balance reading (R) to the corresponding mass of the object, measured under the conditions of the balance calibration (W^*). The real mass of an object (or simply, its mass) would be that determined in complete vacuum or that corrected by the air buoyancy effect.

Mass air buoyancy corrections are easy to apply if both the densities of the object whose mass is being measured and that of the air causing it to float are known or properly estimated (see Section Mass Air Buoyancy Correction). As long as this correction is usually very small and the complete information to perform it is not always known, a common practice is to report the mass of objects under arbitrarily agreed conditions, leading to the conventional mass of the object.

The conventional mass of an object is defined as the mass m_c of a mass standard that balances this body under conventionally chosen conditions: at a temperature $t_{ref} = 20^\circ\text{C}$, with mass standards of density $\rho_c = 8000 \text{ kg m}^{-3}$, in normal air of density $\rho_0 = 1.2 \text{ kg m}^{-3}$ (Organisation Internationale de M etrologie L egale 2004).

The conventional mass of an object m_c can be estimated as the mass of the object measured under the conditions prevailing at the moment of the calibration of the balance (W^*). W^* can be defined as $W^* = R - E$ where E is the indication error at that point. This indication error can be obtained by interpolating the value into the straight line defined by the two closest points measured in the calibration.

The `masscor` package uses the calibration information contained in an object of class `calibCert` to obtain a conventional mass estimate by using the function `convMass()`.

Suppose we use the balance whose calibration certificate is stored in the example data set MT.XPE.204 provided with the package. For a balance reading of 211.1342 g, the (corrected) mass under the conditions of the calibration is obtained by running:

```
data(MT.XPE.204)
convMass(calibCert = MT.XPE.204, reading = 211.1342, units = 'g')
#> The result corresponds to the mass measurement result at the conditions of the calibration.
#> [1] 211.1344
```

A more accurate value for the conventional mass m_c can be estimated if the air density at the moment of measurement and the density of the object being measured are provided (EURAMET 2015, sec. 9.2)

$$m_c = W^* \left[1 + (\rho_a - \rho_0) \left(\frac{1}{\rho} - \frac{1}{\rho_c} \right) \right]$$

where ρ_a is the local density of air which can be estimated using several models as described in Subsection Air density models and associated uncertainty, ρ_0 is the standard air density (0.0012 g cm^{-3}), ρ is the density of the object and ρ_c is the standard density of the mass standards (8 g cm^{-3}).

```
convMass(calibCert = MT.XPE.204, reading = 211.1342, units = 'g', rho = 7.113,
         rho_air = airDensity(Temp = 21.1, p = 990, h = 46.6,
                             unitsENV = c("deg.C", "hPa", "%")))
#> [1] 211.1343
```

0.3.1 Conventional mass uncertainty

The uncertainty in the conventional mass of an object includes both the uncertainty from the indication error correction and the uncertainty in the balance reading. Each one in turn considers the uncertainty due to balance readability, related to the balance division scale and the uncertainty arising from the lack of repeatability.

Uncertainty from the indication error correction can be obtained by interpolation using the uncertainties in the points used to produce the conventional mass estimate (EURAMET 2015, sec. 7.4.2). This uncertainty source can be obtained using the `uncertErrorCorr()` function:

```
(u_err <- uncertErrorCorr(calibCert = MT.XPE.204, reading = 211.1342, units = 'g'))
#> [1] 0.0002278355
## Result rounded to three significant figures:
signif(u_err, 3)
#> [1] 0.000228
```

Uncertainty in the balance reading combines effects from rounding, repeatability, and eccentricity (EURAMET 2015, sec. 7.4.1.5):

$$u_{read} = \sqrt{2u_d^2 + u_{rep}^2 + u_{ecc}^2}$$

- u_d is the uncertainty arising from the rounding error due to readability, given by the balance division scale: $u_d = \frac{d}{\sqrt{12}}$. This uncertainty is multiplied by two because it affects both the balance reading with the object and the tare balance reading at zero.
- u_{rep} is the standard deviation of repeated measurements. When a single balance reading is taken (as usual), standard deviation from the calibration certificate is used. If more than one point was measured during the repeatability test, the bigger from the standard deviations of the two closest points to the balance reading is used.
- u_{ecc} is the uncertainty due to the eccentricity which is taken from the maximal difference obtained in the eccentricity test ($|\Delta R_{ecc}|_{max}$) at a given balance loading (L_{ecc}), and is assumed to be proportional to the balance reading: $u_{ecc} = R \cdot \frac{|\Delta R_{ecc}|_{max}}{2L_{ecc}\sqrt{3}}$.

The uncertainty from a given balance reading can be calculated by using the `uncertReading()` function:

```
(u_read <- uncertReading(calibCert = MT.XPE.204, reading = 211.1342, units = 'g'))
#> [1] 8.35552e-05
```

The balance reading and the indication error uncertainties are combined as usual ($u_{W^*} = \sqrt{u_{error}^2 + u_{read}^2}$). The calculation can be made directly by using the function `uncertConvMass()`

```
(u_w <- sqrt(u_err ^ 2 + u_read ^ 2))
#> [1] 0.0002426736
(u_w <- uncertConvMass(calibCert = MT.XPE.204, reading = 211.1342, units = 'g'))
#> [1] 0.0002426736
```

0.4 Air buoyancy correction

As mentioned in previous section, when measuring the mass of an object in air the object experiences buoyancy and its mass is affected. This effect is usually very small and can be safely ignored most of the times. When the correction is needed, the mass conventional mass of the object being measured can be corrected by multiplying it by its mass air buoyancy correction (MABC) factor, defined as (NIST 2019, SOP 2, Appendix A, section 8.5.1):

$$MABC = \frac{1 - \frac{\rho_{air}}{\rho_{weights}}}{1 - \frac{\rho_{air}}{\rho}}$$

% where ρ_{air} is the local air density (around 1.2 kg m^{-3}), $\rho_{weights}$ is the density of the mass standards used during calibration (most of the times the mass standards are made of stainless steel with a density of 8000 kg m^{-3}), and ρ is the density of the object.

The function `MABC()` calculates mass air buoyancy correction factors using the above formula. The densities must be provided all in the same units.

```
MABC(rho = 2.115, rho_w = 8, rho_air = 0.001199314) # g/cm^3
#> [1] 1.000417
```

The mass of the object whose mass measurement result is W (after calibration corrections has been applied) is defined as

$$m = W \cdot MABC$$

0.4.1 Air density models and associated uncertainty

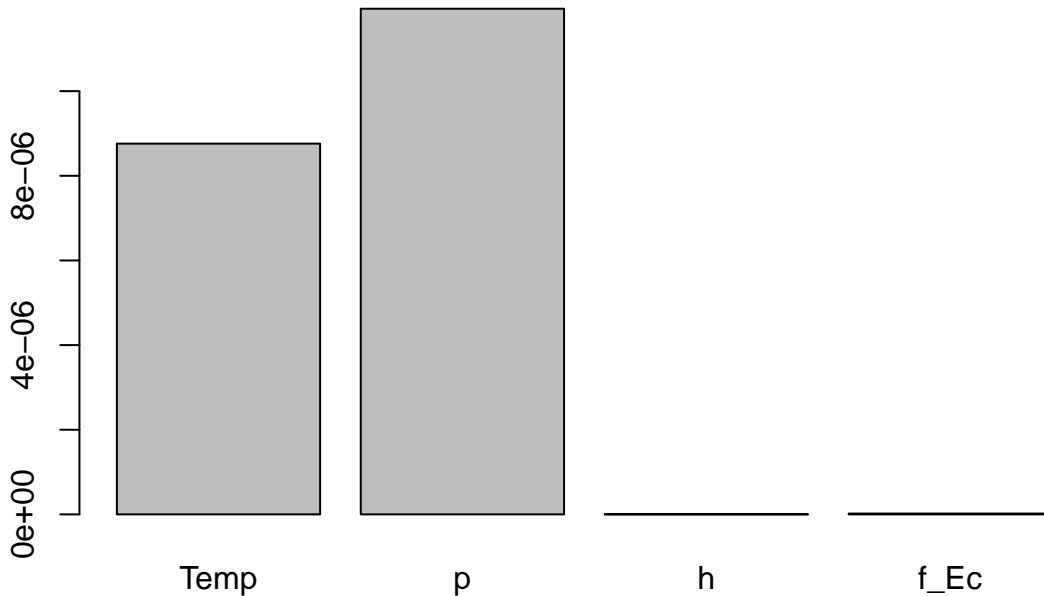
Local air density can be calculated from environmental conditions by applying one of several models. The function `airDensity` implements the CIMP2007 formula (Picard et al. 2008), a CIMP approximated formula [] and the Jones model (Jones 1978). The function `airDensityHASL()` calculates approximated density of local air using a model that relies only on the height above sea level (HASL) in meters. The outputs of those functions are in $[\text{g cm}^{-3}]$.

```
Temp <- 21.5 # Celcius degrees
p <- 751 # Hectopascals
h <- 58.1 # Percentaje
unitsENV = c("deg.C", "hPa", "%")
(rho_a.1 <- airDensity(Temp = Temp, p = p, h = h, unitsENV = unitsENV, model = 'CIMP2007'))
#> [1] 0.0008815079
(rho_a.2 <- airDensity(Temp = Temp, p = p, h = h, unitsENV = unitsENV, model = 'CIMP.approx'))
#> [1] 0.0008815681
(rho_a.3 <- airDensity(Temp = Temp, p = p, h = h, unitsENV = unitsENV, model = 'Jones1978'))
#> [1] 0.0008815098
(rho_a.4 <- airDensityHASL(HASL = 2640)) # Height above sea level for Bogotá-Colombia
#> [1] 0.0008833069
```

The function `uncertAirDensity()` estimates uncertainties for the air densities calculated using the CIMP 2007 formula or the CIMP approx formula. The standard uncertainties from temperature, barometric pressure

and relative humidity must be provided and the function combines them with the uncertainty arising from the model. The uncertainty calculation is made according to the GUM (JCGM 2008) as implemented by the R package `propagate` (Spiess 2018). The function can optionally show the relative contribution of each source in a bar plot.

```
(u_rho_a.1 <- uncertAirDensity(Temp = Temp, p = p, h = h, unitsENV = unitsENV,
                               u_Temp = 2.9, u_p = 10.10, u_h = 11.3, model = 'CIMP2007',
                               plot = TRUE))
```

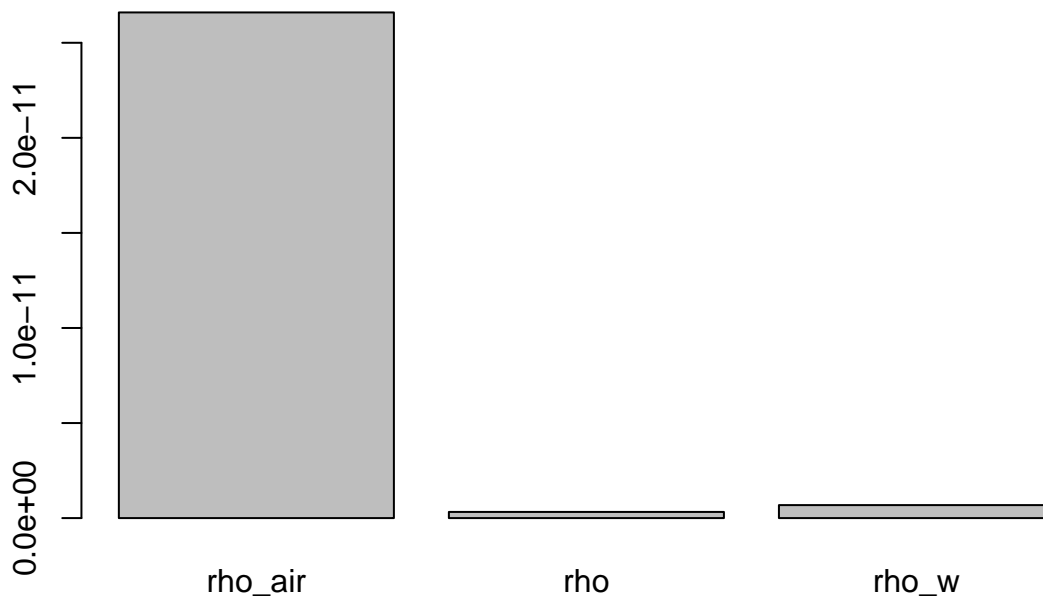


```
#> Relative uncertainty in air density estimation: 1.6681 %
#> [1] 1.481498e-05
```

0.4.2 Uncertainty of air bouyancy correction

The uncertainty calculation of the MABC factor is made by the function `uncertMABC()`, according to the GUM (JCGM 2008) as implemented by the R package `propagate` (Spiess 2018). The function can optionally show the relative contribution of each source in a bar plot.

```
uncertMABC(rho = 2.115, rho_w = 8, rho_air = rho_a.1,
           u_rho = 0.005/sqrt(3), u_rho_w = 0.060, u_rho_air = u_rho_a.1,
           plot = TRUE)
```

```
#> Relative uncertainty in MABC: 5e-04 %
#> [1] 5.253927e-06
```

Bibliography

- EURAMET, Calibration Guide No. 18. 2015. *Guidelines on the Calibration of Non-Automatic Weighing Instruments*. https://www.euramet.org/Media/docs/Publications/calguides/I-CAL-GUI-018_Calibration_Guide_No._18_web.pdf.
- JCGM, Joint Committee for Guides in Metrology. 2008. “JCGM 100: Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement.” JCGM.
- Jones, F. E. 1978. “The Air Density Equation and the Transfer of the Mass Unit.” *Journal of Research of the National Bureau of Standards* 83 (5): 419. <https://doi.org/10.6028/jres.083.028>.
- NIST, National Institute of Standards and Technology. 2019. “Selected Laboratory and Measurement Practices and Procedures to Support Basic Mass Calibrations.” <https://doi.org/10.6028/NIST.IR.6969-2019>.
- Organisation Internationale de Métrologie Légale. 2004. “Conventional Value of the Result of Weighing in Air.” *OIML D 28 Edition 2004 (E)*.
- Picard, A, R S Davis, M Gläser, and K Fujii. 2008. “Revised Formula for the Density of Moist Air (CIPM-2007).” *Metrologia* 45 (2): 149–55. <https://doi.org/10.1088/0026-1394/45/2/004>.
- Spieß, Andrej-Nikolai. 2018. *Propagate: Propagation of Uncertainty*. <https://CRAN.R-project.org/package=propagate>.
- Wickham, Hadley, Jim Hester, and Winston Chang. 2019. *Devtools: Tools to Make Developing r Packages Easier*. <https://CRAN.R-project.org/package=devtools>.