

# Code documentation for the simulation study: Comparison of confidence intervals summarizing the uncertainty of the combined estimate of a meta-analysis

Felix Hofmann

## Contents

1	Overview	2
2	utils.R	2
3	study_simulation.R	3
4	studies2cis.R	3
5	cis2measures.R	4
6	measures2summary.R	4
7	simulation.R	4

# 1 Overview

This document gives an overview over the code that implements the simulation study *Comparison of confidence intervals summarizing the uncertainty of the combined estimate of a meta-analysis*. The code is split into six different files:

- `utils.R`
- `study_simulation.R`
- `studies2cis.R`
- `cis2measures.R`
- `measures2summary.R`
- `simulation.R`

The following sections will describe the contents of each of these scripts.

## 2 `utils.R`

The script *utils.R* contains short utility functions that are used multiple times throughout the simulation.

- `rep2`
- `error_function`
- `hMeanChiSqMu_f`
- `hMeanChiSqMu_chisq`

Currently, we use only the functions `rep2()` and `error_function()`.

`rep2()` is an extension of the `rep()` function in the *base* package. It works exactly as `rep()` but it can take an input vector for the `each` argument. Its use should become clear in the example below:

```
rep2(x = c("a", "b", "c"), each = 1:3)
## [1] "a" "b" "b" "c" "c" "c"
```

`error_function()` on the other hand is mainly used for logging. Thus, it is only used in `tryCatch`-blocks. The intended use is that in case of errors, the function stores the last object together with the current parameters to a file on disk (files *error.rds* and *pars.rds*). It also writes the error message as well as the iteration number to a text file (*error.txt*).

### 3 study\_simulation.R

The script *study\_simulation.R* contains all functions that are related to the simulation of the individual studies. The file contains the following functions.

- `simRE`
- `pAccept`
- `simREbias`
- `sim_effects`

These functions implement the simulation of the individual studies of a meta-analysis. In the simulation, we only use `sim_effects()`, the main function, which calls the other functions internally. This function has two arguments, *pars* and *i* which corresponds to the current iteration.  
The mechanism, how this

describe  
argu-  
ments

### 4 studies2cis.R

- `get_classic_obj_hc()`
- `get_classic_obj_reml()`
- `get_classic_obj_hk()`
- `tau_prior_bm()`
- `get_classic_obj_bm()`
- `get_classic_obj()`
- `get_classic_ci_reml()`
- `get_classic_ci_hk()`
- `get_classic_ci_hc()`
- `get_classic_ci_bm()`
- `get_classic_pi_reml()`
- `get_classic_pi_hk()`
- `get_classic_interval()`
- `get_classic_intervals()`
- `get_p_value_functions()`
- `get_p_value_args()`
- `get_new_ci_gamma()`

- `get_new_intervals()`
- `get_tau2()`
- `get_ints_reml()`
- `sim2CIs()`
- `calc_ci()`

The main function here is `calc_ci()`, which calls all the other functions internally.

**5    `cis2measures.R`**

**6    `measures2summary.R`**

**7    `simulation.R`**

Describe  
argu-  
ments,  
ex-  
plain  
struc-  
ture