Werner Wosniok
wwosniok@math.uni-bremen.de

Truncated minimum chi-square (TMC) estimation

TMC Software Manual

Version 2020-07-27

1 Purpose of the TMC software

The purpose of the TMC programme is the indirect estimation of reference Intervals (RIs) from routine laboratory data. The data used by the approach is assumed to be a mixture of values from patients who do not suffer from a disease that affects the measurand under study and from patients who have values that are affected (decreased or increased) by some disease. The latter are for brevity called “pathological values”.

The first basic assumption of the TMC (and all other indirect methods) is that there is a subinterval in the set of all measurements which contains no pathological values. Only this subinterval can and will be used to estimate RIs. Size and location of this subinterval are a priori unknown and are determined by the TMC approach.

The second basic assumption is that non-pathological values are distributed according to a power normal distribution (PND), which is a generalisation of the normal (Gaussian) distribution. It has three parameters: the shape parameter λ, the location parameter μ and the dispersion parameter σ. Interesting special cases are λ = 1, which defines a Gaussian distribution, and λ = 0, which defines a log-Gaussian (lognormal) distribution.

The TMC software identifies a not affected subinterval in the data, fits a PND distribution to this subinterval and calculates the corresponding RIs. The analysis is done by age and sex, if corresponding data are provided, and, if a sufficient number of age groups is available, sex-specific smooth functions are provided for the relation between RI and age. This allows the determination of RIs for each age in the range covered by the data. Also, diagnostic quantities for the data (suspicious values, diurnal variation, goodness of fit for the result) are provided.

The user has to supply the data (see 2.5), to describe details of the data (see 2.6) and the required analysis (see 3.1 for the analysis of a single data file and 3.2 for the analysis of a file sequence in a simulation).

The present TMC version operates completely in the R environment (1).

2 Technical requirements

2.1 Technical requirements: software

The programme requires the R software, available from https://cran.r-project.org/. It requires the packages modeest and stringr, available from the same source. However, if the TMC programme detects that these packages are missing, it tries installing them. In that case, the user is asked to select an R mirror from a list of available mirrors. Use the mirror that is geographically closest to you.

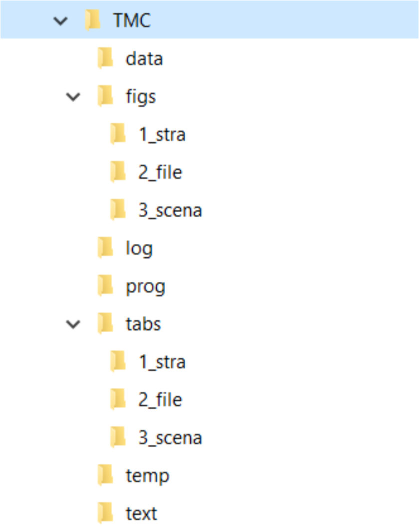
The TMC programme is provided as a zip file, which contains all necessary files and the directory stricter needed. Alternatively, an installation programme is available, which installs the directory structure and copies all necessary files from a web site to their positions in the user's directory structure. If the installation finds an old TMC version, it makes a backup copy to the backup directory.

The TMC programme is expected to run under all R versions >= 3.5.1.

2.2 Installation

2.2.1 Directory structure

TMC uses the following directory structure:



```
TMC
    data
    figs
        1_stra
        2_file
        3_scena
    log
    prog
    tabs
        1_stra
        2_file
        3_scena
    temp
    text
```

This structure is used when the TMC programme executes. It will be installed automatically, if the zip file is unzipped under conservation of the directory structure or if the installation programme is used. Otherwise, the directories must be defined manually.

The `data` directory in the supplied TMC version contains an example data set (`TMC4_Test.csv`) and also an example input data information file (`DataFileInfo.csv)` (see 2.6). The `TMC/data` directory may be used for storing user data, but this location is not mandatory. Data files may be located everywhere, and their location (path and file name) must always be specified in the data information file. Also, the data information file may be stored everywhere. Its location must be specified in `TMC_seg010_User.R`. There, the variable `info.file` contains name and path of the data information file (default position: the `TMC/data` directory).

The `figs` directory contains 3 subdirectories for various figures which are produced during a run. In `1_stra,` stratum-specific graphics files are stored (stratum: each combination of sex and age class defines a stratum). In 2_file, file-specific graphics files are stored (e.g. displays of all existing values, result figures showing RL vs age). The third directory, 3_scen, contains result figures from the evaluation of sequences of data files, typically in the framework of a simulation.

The log directory contains a text file for each run, containing programme versions and parameter settings used in that run. A run is identified by its RunId, which is the date and time of day of the run (example: 2020-07-14_115339.txt is the log file for the run started on July, 14, 2020 at 11:53:39).

The prog directory contains all R programme segments. The only segment that routinely needs modification by the user is the start segment, which controls the analysis of a data file. Details concerning the start segment are given in 3.1 . A start segment example is provided with the installation (TMC_seg000_Start_Test.R) . Immediately after installation, also the file TMC_seg010_User.R may need modification, see 2.3 .

The tabs directory contains tables that are produced during a run. The directory has a subdirectory structure similar to the figs directory.

The temp directory is used by TMC for storing temporary files and of no interest for the user.

The text directory contains this manual.

2.2.2 Installation from the supplied zip file

The installation zip file, named TMC_public_yyyy-mm-dd.zip, where yyyy-mm-dd identifies the programme version, contains

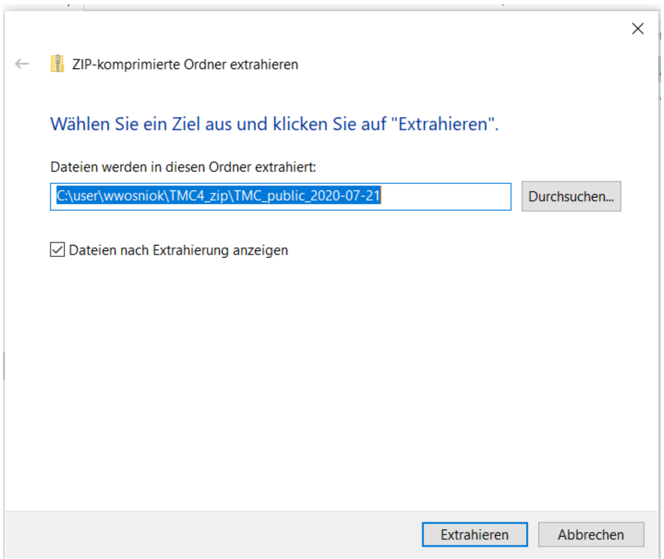
- the directory structure (see 2.2.1),
- all programme segments including a test start segment,
- a test data file,
- a test data information file,
- a test output table
- a test output figure

For the FIRST Installation:

Open the zip file with a right mouse click. This opens a box where you select the directory into which TMC will be installed. In the example below (next page), TMC will be extracted to D:\Programs\TMC. Clicking “Extract all” / “Alles extrahieren” does the extraction.

For the installation of UPDATES to an existing installation:

Rename the present TMC directory to TMC_ yyyy-mm-dd.zip, where yyyy-mm-dd is the date of the new installation. Then proceed as under “**FIRST** Installation”. In this way, all so far used files, particularly the TMC_seg010_User.R file, all data files and all start files are still available (in the renamed directory) and can be copied to the new installation, thus saving installation work.



ZIP-komprimierte Ordner extrahieren

Wählen Sie ein Ziel aus und klicken Sie auf "Extrahieren".

Dateien werden in diesen Ordner extrahiert:

C:\user\wwosniok\TMC4_zip\TMC_public_2020-07-21

☑ Dateien nach Extrahierung anzeigen

Continue with 2.3 .

2.2.3 Installation with the installation programme

- Copy the installation file to the directory in which you want the TMC directory to be installed, the 'installation directory'.
- Open R
- Change the working directory to the installation directory (File / Change dir) or (Datei/ Verzeichnis wechseln)
- Open the installation file
- Run the installation file (Ctrl+a, then Ctrl+s) or (Strg+a, then Strg+s)

This causes the installation of TMC. Details of the installation depend on whether this is a new installation or an update of an existing installation:

If there is not yet a TMC directory in the installation directory:

- the TMC directory and all necessary subdirectories will be created
- all programme files will be copied to TMC/prog
- the manual will be copied to TMC/text
- a test data set 'TMC_Test.csv' will be copied to TMC/data
- a sample data description file 'DataFileInfo.csv' will be copied to TMC/data.

If there is already a TMC directory in the installation directory:

- all files from TMC/prog are moved to TMC/backup/yyyy-mm-dd-HH-MM, where yyyy-mm-dd-HH-MM is the date and time of day of the new TMC installation
- all actual programme files will be copied to TMC/prog
- the actual manual will be copied to TMC/text
- a test data set 'TMC_Test.csv' will be copied to TMC/data

After installation, the installation programme makes TMC/prog the working directory. Continue with 2.3 .

2.3 Setting user options after installation

The programme segment `TMC_seg010_User.R file` holds the user identification and some general settings, which must be set before the first TMC execution.

2.3.1 Set the user name

Purpose:	Controls certain output components and paths
Variable name:	`user`
Default value	“`Usr`”
Comment	Modification is not normally needed, only needed if the user wants to modify the programme code

2.3.2 Set name and location of the file containing the data file information

Purpose	Instructs TMC where to look for the data file information
Variable name:	`info.file`
Default value	`“../Data/DataFileInfo.csv”`
Comment	Modification is not normally needed, only needed if the user wants this file somewhere else

2.3.3 Control amount of printed summary tables

Purpose	Controls printing of only 2 essential summary tables or all 6 summary tables
Variable name:	`print.6tables`
Default value	`FALSE`
Comment	`FALSE` should suffice for most applications

2.3.4 Control details in plots

Purpose	Controls printing of legends, gridlines and more details in plots
Variable name:	`plot.details`
Default value	`FALSE`
Comment	`TRUE` is helpful when exploring data, but produces output that may collide with publication rules

2.3.5 Control printing log messages

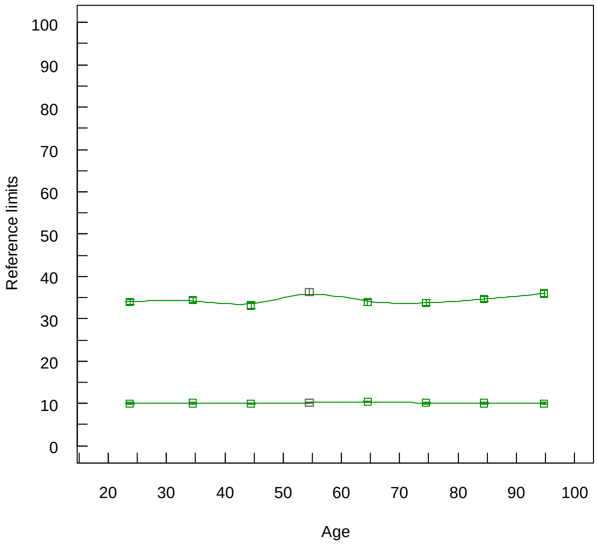
Purpose	Writes messages indicating the position in the programme that is being executed
Variable name:	`print.log.message`
Default value	`FALSE`
Comment	`TRUE` is useful when searching programming errors

2.4 Installation test

The provided test start file `TMC_seg000_Start_Test.R` analyses the provided data set `TMC4_Test.csv,` using the data information file `DataFileInfo.csv`, both located in `TMC/data`. Results are stored in the subdirectories of `TMC/figs` and `TMS/tabs` with file names beginning with `TMC4_test` (the data file name), followed by a description of the stratum (e.g. `F18-29` for females, 18-29 years old) and a code for the type of the output file. This code is explained under 4.

The installation test is done as follows:

- Open the test file `TMC_seg000_Start_Test.R`
- Run this file (Ctrl+a, then Ctrl+s) or (Strg+a, then Strg+s). It does the analysis of the supplied test data file, using the also supplied data information file.
- The last graph file produced by the test start file should be `TMC4/figs/2_file/ TMC4_Test_3-F160.010.emf` which should look like this:



If this plot appears, the programme has regularly terminated. The execution of the analysis on a usual notebook requires roughly 3 minutes.

2.5 Input data

Data for analysis must be in ASCII format, e.g. a csv file as produced by Excel®. Rows in the file correspond to measurements (patients), columns correspond to variables. The first row contains the variable names. They serve as information to the human reader only and are not explicitly used by TMC. They are, however, read by TMC and transformed to names that are in line with the R naming conventions (roughly: may contain letters and numbers, preferably no special characters except '.' and '_' if needed. The TMC programme locates variables via the column numbers which must be specified for each variable in the data file information, see 2.6.

Columns in a data file are separated by ";", the decimal sign is ".". Deviations from this standard must be specified in the data file information.

Columns with following information are required in each data file.

Value	The measured value. No units. A specification of the form < DL is allowed, where DL is the detection or determination limit. Only one DL is allowed in a file.
Age	Patient's age. Any unit (Years, days, ..) is allowed. The same unit must be used in the data and in the definition of age classes (see below)
Sex	Patient's sex. Any coding is allowed (no more than two classes). In the output, males and females are always coded as 'M' and 'F'.

The following optional columns may be present in the data and are used by TMC if the programme is instructed to do so.

DateTime Date and time of the measurement. The required format is dd.mm.yyyy hh:mm:ss, e.g. 21.02.2014 02:36:00. Dots and colons may be replaced by some other character, the second's information may be missing.

OH Information on whether the patient is an outpatient or hospitalized, with the coding left to the user.

Device Name of the device used for the measurement. The coding is left to the user.

2.6 Data information file

This file (a csv file) keeps the information about the structure of data files. By using this file, information on a data file has to be entered only once, even if many analyses are made on the same dataset. The link between a data file and the analysing programme is established by the file number `FileNo`, which is chosen by the user. This number must be unique.

The data information file has the standard name `DataFileInfo.csv`. It uses, as the data files, semicolons to separate fields and decimal points. Both can be changed, if necessary.

Required columns in the data information file are

`FileNo` File number, identifies a file uniquely

`FileName` File name, including the extension, without the path

`Path` Path to the file. May be given as absolute path (starting from the drive name) or as relative path (starting from the TMC/prog directory). Use "/" to separate directories (not "\"). Data files may lie everywhere, they need not lie in the TMC directory as in the provided example.

`Value` the number of the column containing the measured values (In Excel: column A is column 1)

`Label` Label for the measurand. Will appear in output and plots, useful for describing the measurand and providing the measurement unit. All characters are allowed here.

`Rounding` the rounding unit of "Value". 1 means that integer numbers are reported in the data, 0.1 indicates that 1 decimal place is given. Any number of decimal places is possible. TMC rounds the data by this unit.

`DecChar` Decimal sign used in the data (typically . or ,)

`Sex` the number of the column containing the sex code

`SexCodeF` the code for 'female'

`SexCodeM` the code for 'male'

`Age` the number of the column containing the age

`OH` the number of the column containing the outpatient / hospitalized information. Can also be used for other variables that define subgroups. Selection of the subgroup to analyse is done in the Start segment.

`DateTime` gives the number of the column containing date and time of making the measurement. Presently required format is "DD-MM-YYYY hh:mm:ss".

The user may add further variables to the data information file in order to describe the data in more detail. However, the following names are reserved names in the data information file and should not be used by a user for adding new information:

FileNo;FileName;Path;Source;DateReceived;Label;Value;Rounding;Sex;SexCodeM;SexCodeF;
Age;DateTime;OH;Device;DecChar;Group;lambda.gen;xc.RL1.gen;xc.RL2.gen;xc.mode.gen;

yc.mode.gen;yc.sig.gen;xl.RL1.gen;xl.RL2.gen;xl.prev.gen;xr.RL1.gen;xr.RL2.gen;xr.prev.gen;
ntotal

An example data information file is provided with the installation under "`TMC/data`".

3 Running the TMC programme

For doing a TMC analysis, the user has to fill an R script, the starting segment, with all details describing the intended analysis and then to run this script. An example starting segment, `TMC_seg000_Start_Test.R,` is provided with the TMC installation. The user may copy this example file to a new file for a real analysis. For an easy overview, each measurand should have its own starting segment. The starting segments should then be named in an informative way, e.g. by adding the name of the measurand to the file name.

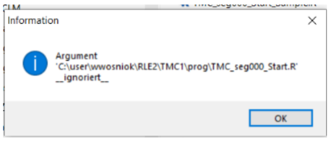
Instructions on how to fill the start segment and how to execute it are given in 3.1.

3.1 Using TMC for analysing a single data file

It is assumed that the R software was linked to files with suffix ".R" (automatically opens *.R files after clicking) during installation. Otherwise this linking should be done with the windows explorer., Alternatively, the user may install a desktop icon which starts R and goes to the `/TMC/prog` directory (not described here).

In the flow below, the name `TMC_seg000_Start_Test.R` is used as name of the starting segment. It can be replaced by any other name starting segment. The flow below assumes that no icon calling TMC was installed.

- Open the Windows explorer
- Go to the TMC home directory
- Open the prog directory
- Click at the file `TMC_seg000_Start_Test.R`. This starts the R programme.
  If a message box

  Argument
  'C:\user\wwosniok\RLE2\TMC1\prog\TMC_seg000_Start.R'
  __ignoriert__

  appears, click at "Ok" and ignore the message.

- Click at the "Open" symbol 📂 in the R menu
- Select `TMC_seg000_Start_Test.R`
  This file controls the operation of the TMC programme. See the comment in the first paragraph of 3. for the role of the start segment file. Its syntax is the usual R syntax, specifically:
  - Everything in a line after a '#' is a comment – R does nothing with it
  - The value NA means 'not applicable'
  - Commands beginning with 'source(" …' call other R programmes – do not change these commands

- The user has to enter the following parameters in `TMC_seg000_Start_Test.R.`

1. The number of the file to analyse (`FileNo`). This number points to a line in the input data information file (see 2.3), which contains the information about the file structure (name, storage directory, column numbers for value, age, ...).
   This parameter is always required.

2. The scaling factor (`scale.fact`). All numbers in the value column of the data file are multiplied by this number, usually to switch to another measurement unit. Note that subsequent entries like axis scales refer to the scaled values.
   No scaling: set `scale.fact <- 1`

3. `x.lo.limit` and `x.hi.limit`. Values < `x.lo.limit` and values > `x.hi.limit` are excluded from all further calculations and displays.
   No exclusion: set `x.lo.limit <- NA and x.hi.limit <- NA.`

4. Selection of the outpatient / hospitalized subset from data. This selection is possible only if a column 'OH' is given in the data file information and available in the data file. `Use.oh` gives the denomination of the subset to use. Example:
   `Use.oh <- "h"` selects those patients who have the value "h" in the variable OH. Note that character strings must always be written exactly as in the data file, i.e. lower and upper case letters differ.
   No selection: set `Use.oh <- NA`

5. Selection of the device to use in the analysis. This selection is possible only if a column 'Device' is given in the data file information and available in the data file. `Use.dev` decsribes the subset to use. Example:
   `Use.dev <- "A&B_2000"` selects those measurements which were made by the device 'A&B_2000' and have the corresponding entry in the data file. Note that character strings are must be given exactly as in the data file, i.e. lower and upper case letters differ.
   No selection: set `Use.dev <- NA`

6. Daytime selection: use only data obtained between ...

```
ana.hour.min <- NA       #  Example: 06
ana.minu.min <- NA       #  Example: 30
```

7. and ...

```
ana.hour.max <- NA       #  Example: 18
ana.minu.max <- NA       #  Example: 45
```

   No selection: set all 4 values to NA

8. Date selection: Split the data at a given date and time and select the part to analyse

```
split.date <- NA      # Example: "04.02.2019",
                      # format is "dd.mm.yyyy"
split.time <- NA      # Example: "13:25",
                      # format is "hh:mm"
use.split  <- NA      # 1: before /  2: after given
                      # date-time
```

No selection: set all three variables to NA

9. Age interval limits. Given are the left interval limits. The left limits belong to the interval, the right ones not, except for the last interval, which includes both limits. Example:

   `age.limits <- c(18, 30, 40, 50)` defines intervals 18-29, 30-39, 40-50

   This parameter is always required.

10. Probability levels for the RILs:

`RL1.p`	probability level for the first RIL (typical: `RL1.p <- 0.025`)
`RL2.p`	probability level for the second RIL (typical: `RL2.p <- 0.975`)

    This parameter is always required.

11. Truncation intervals used for TMC estimation must contain at least a proportion of ... of all data

    ```
    x.tr.prop.min <- 0.60
    ```

    Recommendation: do not change

12. Truncation intervals used for TMC estimation may contain at most a proportion of ... of all data

    ```
    x.tr.prop.max <- 0.85
    ```

    Recommendation: do not change

13. Define lower and upper x limits for better readable detail plots of the fitted distribution.

    These limits affect only the display, not the calculation.

    Limits apply to scaled values.

`x.clip.min`	axis minimum
`x.clip.max`	axis maximum
`x.clip.by1`	distances between tick marks with label
`x.clip.by2`	distances between tick marks without label

    No clipping: set all 4 values to NA

14. x limits and tick marks for the age scale in plots of age vs RL

    These plots are produced only if at least 4 age groups exist.

`age.clip.min`	axis minimum
`age.clip.max`	axis maximum
`age.clip.by1`	distances between tick marks with label
`age.clip.by2`	distances between tick marks without label

    No scaling of the age axis: set all 4 values to NA

15. y limits and tick marks for the RL scale in plots of age vs RL.

    These plots are produced only if at least 4 age groups exist.

    ```
    RL.clip.min   <-   0  # lower limit
    RL.clip.max   <- 100  # upper limit
    RL.clip.by1   <-  10  # vertical tickmarks with labels
    ```

```
RL.clip.by2   <-   5  # vertical tickmarks without labels
```

No scaling of the RL axis: set all 4 values to NA

16. Only If this is the analysis of a file sequence (usually simulated data):
    Specify the sequence
    First and last replicate

```
r.start     <-   NA        # Example:   1
r.ende      <-   NA        # Example: 100
```

If this is not the analysis of a file sequence: set both variables to NA.

Many more settings affect the execution of the programme. They are located in TMC_seg015_DefaultSettings.R and are not required for the standard use of the programme. Nevertheless, all settings in TMC_seg015_DefaultSettings.R can be changed by the user (at his/her own risk …). If changes of the defaults are intended, it is recommended to put these in the start segment after the line

```
####  Put changes of default settings after this line
```

When the starting segment is completed, it should be saved and executed:

- Save TMC_seg000_Start_Test.R . Easiest way: Ctrl+s (keep the 'ctrl' key down and press the 's' key).
- Run the TMC_seg000_Start_Test.R . Easiest way: Ctrl+a , then Ctrl+r.
  The programme shows text results in the console window and writes essential results to text files in the tab directory (see the subdirectory description above). Plots are shown in their own windows, essential plots are written as *.emf files to the figs directory (see the subdirectory description above). *.emf files can be included in Word, Excel and Powerpoint. For other text systems, e.g. LaTex, bmp, pdf or png files can be produced (by setting the appropriate keyword in TMC_seg015_DefaultSettings.R).
- Executing the start segment is easiest done by going into the start segment and the pressing Ctrl+s , then Ctrl+a and Ctrl+r, where 'Ctrl+s' means to keep the 'ctrl' key down and then press the 's' key. Commands in the start segment beginning with 'source(" ' must not be changed. They do the calls to other required files.

3.2 Running the R version of TMC for the analysis of a file sequence

A sequence of data files is typically analysed in a simulation study. The TMC programme requires that files in the sequence have names ending with a number (e.g. AST_r001.csv, AST_r002.csv, ...). At present, file numbers must consist of 3 digits with leading zeroes, and gaps in the number list (like 001, 002, 004) are not allowed.

For a file sequence, the entry in DataFileInfo.csv for the whole sequence is the same as for the analysis of single files described under 2.3, with two exceptions:

FileNo	File sequence number, identifies a file **sequence** uniquely.
FileName	Basic component of the file name, without the numbering and **without** the extension. Example: AST_r for the sequence AST_r001.csv, AST_r002.csv, ...

The range of file numbers to analyse is defined in the start segment, position 16, through the variables `r.start` and `r.ende`, which contain the numbers of the first and the last file to analyse. The TMC script produces the output file `<FileName>-S-TMC.csv`, which contains the essential estimation results.

If the simulation shall include also results from the RLE (TML) software, the TML estimation has to be done before running the TMC software. The call of TML can be done via the script `Call_TML_sequence.R`. This script runs the TML analysis for each file having the basic name FileName and a number in the sequence `r.start` to `r.ende`. `FileName`, `r.start` and `r.ende` are specified in the script. The script produces the output file `<FileName>-S-TML.csv`.

If this file is present at the end of a TMC sequence analysis, results from the files `<FileName>-S-TMC.csv` and `<FileName>-S-TML.csv` are jointly tabulated and plotted.

4 TMC output

The R script writes status messages and actual results to the R console window. Essential results are always stored in the TMC/figs and TMC/tabs (see 2.2.1), and a log file describing all parameters of an analysis is written to TMC/log.

The name of a result file for figures has following structure:

data file name – column of values - stratum – code for output type – suffix

The "stratum" describes the sex and age group (e.g. F 18-29 for females, 10-29 years old).

The "column of values" is the number of the column in the data file containing the values.

The following table gives the "codes" for the default output figures. More or less plots can be requested by setting the corresponding parameter in `TMC_seg045_PlotRequest.R`. As a default, plots contain no legends or other information describing the result in detail. This information can be switched on, see 2.3.4 . Plots and output tables always contain the `RunId` (see below) and the name of the analysed file. The appearance of plots (colors, line types, …) is controlled in the style file `TMC_seg030_Style.R`.

Code	In directory	Contents
F100.005	1_stra	Compares empirical and kernel density distribution function
F100.010	1_stra	Shows frequencies of observed values. Useful for checking extreme outliers and rounding artefacts, see the "Remark on rounding" (5.5)
F100.030	1_stra	Empirical histogram and fitted PND with RL estimates
F100.040	1_stra	Optimal solution for the QQ plot
F102.010	1_stra	Optimality criterion and goodness of fit
F090.010	2_file	Contour plot of value vs age, all data, females only
F090.020	2_file	Contour plot of value vs age, all data, males only
F095.010	2_file	Mean and confidence interval per weekday and time of day, all data

F095.020	2_file	Median +/- 2*MAD/sqrt(n) per weekday and time of day, all data
F095.030	2_file	Mean and confidence interval per time of day, Mo – Fr only
F095.040	2_file	Median +/- 2*MAD/sqrt(n) per time of day, Mo – Fr only
F150.010	2_file	Estimated RLs vs age, females (red) and males (blue) separated.
F160.010	2_file	Estimated RLs vs age, females and males jointly

Tabular output lies in the **TMC/tabs** directory. The file names are constructed as for figures, but no codes are used. The *.txt files in **TMC/tabs/1_stra** contain detailed results per stratum, the *.txt files in the **TMC/tabs/2_file** contain summary results as simple text files. The *.csv files in **TMC/tabs/2_file** contain the summary as csv files.

The suffix of every file describes the technical form of the file (csv, txt, emf, …).

Example:

TMC4_Test_3_F_60-69-F100.030.emf

is the name of the figure file showing the empirical histogram and the fitted PND density for the file TMC4_Test.csv, values in column 3, for the females in the age group 60-29 years. The file format is emf (expanded windows metafile).

The parameter settings of each run are written to a log file in **TMC/log**. The log file and all output have a run identification (**RunId**), which identifies the time of the run, and the settings of the run including the versions of all programme segments involved are recorded in the log file.

The main result table written to the R console and the text files has following form (example from the provided test data):

```
------------------------------------------------------------------------
       Table 1: Main results
------------------------------------------------------------------------
   Sex    Age      n pct.DL meth   x.RL1   x.RL2  prev.c  p.fit err
1  F+M 18-100 80000    0.0  tmc 10.0696 35.1180  0.9420 0.2108
2    F 18-100 36040    0.0  tmc 10.1767 35.4200  0.9344 0.2206
3    M 18-100 43960    0.0  tmc 10.1014 34.7625  0.9370 0.1886    !
4  F+M  18-29 11039    0.0  tmc  9.9999 33.9922  0.9302 0.2473
5  F+M  30-39  9940    0.0  tmc 10.0533 34.4063  0.9457 0.6662
6  F+M  40-49  9807    0.0  tmc  9.8983 33.1222  0.8926 0.3010
7  F+M  50-59  9775    0.0  tmc 10.1534 36.3122  0.9444 0.0785    !
8  F+M  60-69  9758    0.0  tmc 10.4091 33.8710  0.8993 0.2720
9  F+M  70-79  9630    0.0  tmc 10.0677 33.7907  0.9176 0.3087
10 F+M  80-89  9753    0.0  tmc 10.0507 34.6669  0.9470 0.6621
11 F+M 90-100 10298    0.0  tmc  9.9822 35.9533  0.9592 0.4390
12   F  18-29  4970    0.0  tmc 10.1153 34.5422  0.9310 0.2669
13   F  30-39  4456    0.0  tmc 10.1056 34.7045  0.9471 0.2678
14   F  40-49  4469    0.0  tmc  9.9832 33.8127  0.9056 0.1714    !
. . .
```

where

Sex and **Age** define the stratum
n number of values in the stratum
pct.DL percentage of values below determination / detection limit

meth is the estimation procedure

x.RL1 estimated lower reference limit, original scale

x.RL2 estimated upper reference limit, original scale

prev.c estimated prevalence of non-pathological values (the central part of the total distribution)

p.fit p value for goodness of fit in the truncation interval, should be ≥ p.fit.min (default: 0.20)

err a “!” indicates insufficient goodness of fit

5 Theoretical background of the TMC approach

The text below is an update of the method and procedures described in (2). Thanks go to Farhad Arzideh and Theo Postma, who indicated editorial mistakes in an earlier text (which luckily had no consequence for the analysis).

5.1 Assumptions of the TMC approach

Central assumptions of the TMC approach are: (i) The data contains an interval without values from diseased persons, the unaffected interval. (ii) Values are stochastically independent from another (no multiple values from the same subject). (iii) Values from non-diseased persons follow a power normal distribution.

The first two assumptions are shared by all indirect methods. The last one is more general than in many other approaches, which assume a normal or a log-normal distribution of values from non-diseased persons.

The Power Normal distribution has density (2)

$$f(x; \lambda, \mu, \sigma) = \frac{1}{K \cdot \sigma\sqrt{2\pi}} x^{\lambda-1} \cdot \exp\left(-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right) \qquad \text{(A1)}$$

where

$$K = \phi\left(\frac{1}{\lambda \cdot \sigma} + \frac{\mu}{\sigma}\right) \qquad \text{(A2)}$$

with ϕ denoting the standard normal (Gaussian) cumulative density function, and y is the Box-Cox transformation of x with parameter λ: $y = \left(x^{\lambda} - 1\right)/\lambda$ for $0 < \lambda \leq 1$ and $y = \ln(x)$ for $\lambda = 0$. The corresponding cumulative distribution function is

$$F(x; \lambda, \mu, \sigma) = \int_{0}^{x} f(s; \lambda, \mu, \sigma)\, ds \qquad \text{(A3)}$$

5.2 Rationale of the TMC approach

The TMC approach considers reported values as describing intervals on the real line. This means that a reported value “x” is interpreted as the information that the measured quantity lies somewhere in the interval [x-d/2, x+d/2), where d is the rounding unit. The notation [a, b) describes a right-open interval with the left limit (a) belonging to the interval, but the right limit (b) not. As example: d = 0.1

means that values are reported rounded to one decimal place, and a reported value of 61.4 g/L corresponds to the interval [61.35 g/L, 61.45 g/L). A reported value “<x” is understood as the interval [0, x), where usually x is the limit of detection or determination.

Treating reported data generally as interval data is not only the correct interpretation of the data, but doing so allows also a uniform processing of all data, no matter if it is presented as a single value (“x”) or an interval (“<x”). This approach avoids the need of (arbitrarily) setting surrogate values.

Interval data is appropriately presented as a histogram.

5.3 Procedural steps of the TMC approach

The TMC approach has 6 steps, which are implemented in the R script “TMC”

1. Define the age / sex stratum to analyse
2. Construct a histogram for the values in this stratum
3. Obtain an initial estimate for the PND parameters λ, μ, σ from a sequence of QQ plots
4. Obtain improved estimates of the PND parameters by the TMC procedure, while considering various truncation intervals
5. Identify the optimal PND parameter among the candidates considered in step 4
6. Calculate the RLs from the optimal PND parameters from step 5

These steps are described in detail below.

5.3.1 Define the age / sex strata to analyse

Age groups are defined by the user. If age is given in years, a typical definition of age groups could be approximately 10 years intervals (e.g. 18-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-100) years. With 8 age groups and two sexes in the data there are 16 age/ sex strata in the data, which are analysed independently and consecutively.

5.3.2 Construction of a histogram for the stratum

The data used for TMC estimation corresponds to the data that is needed to construct a histogram for the data. A histogram consists of k bins with limits c_i, and c_{i+1}, i = 1,2, …k, and a set of counts n_1, n_2, …, n_k, where n_i, i = 1,2, …,k, are the number of values lying between c_i and c_{i+1}, i = 1,2, …k. The lower limit is included in each bin, for n_k also the upper limit. Initial values for the c_i are the mean values between reported values, completed by the maximum of (reported minimum-0.5 rounding units, 0) and (reported maximum + 0.5 rounding units) for the outer intervals. All reported values lie between the outer interval limits c_1 and c_{k+1}. If the rounding unit is small, this initial construction generates many bins with small counts n_i, which is unfavourable for the subsequent calculation of the TMC optimisation criterion, a χ^2 based quantity. Therefore, bins containing less than the pre-specified minimum (variable `n.per.bins.min`, default 50) of values are aggregated with their neighbours until the required minimum count is achieved. If the aggregation of bins leaves less than `x.bins.min` (default 8) bins, a warning is issued and the actual stratum is not evaluated. Both conditions together imply a minimal number of n = 400 values per stratum. Bin limits need not be, and usually are not, equidistant. Note that there are k+1 bin limits, but only k bins.

The aggregated data can be represented graphically as a histogram with the c_i as bin limits and the bar heights chosen such that (bin width) * (bin height) = n_i / n. Where needed in the equations below, bins are indexed by the index of their lower limit: the bin [c_i, c_{i+1}) has index i.

5.3.3 Obtain an initial estimate for the PND parameters **λ, μ, σ** from a sequence of QQ plots

Rationale: Under the assumptions of the TMC approach there must be an interval consisting of several bins in the data which contains only values from a single PND. In a QQ plot of the data, transformed by the correct λ, this interval appears as a nearly straight line. This λ and the corresponding data interval are searched for by a simple grid search, using the coefficient of determination as measure for linearity. Regression parameters from the interval provide initial values for the subsequent TMC estimation. However, these initial values cannot serve as authoritative estimates of the PND parameters for reasons that are explained in 5.4 below.

5.3.3.1 Initialize the QQ plot optimality criterion

Set $\mathrm{r}_{\max}^2 = 0$

5.3.3.2 Initialize the λ sequence

Set λ = 0

5.3.3.3 Transform the data by the actual λ

Transform the data in the actual stratum by the Cox-Box transformation:

$$y_i = \frac{x_i^{\lambda}-1}{\lambda} \quad \text{if } \lambda > 0, \qquad y_i = \ln(x_i) \quad \text{if } \lambda = 0$$

5.3.3.4 Construct the QQ plot

Construct a QQ plot of y_i:

abscissa: expected values of the order statistics of a standard Gaussian distributi
$E\left(x_{[i]}\right) = \Phi^{-1}\left((i-0.5)/n\right)$, with Φ^{-1} denoting the inverse standard Gaussian distribution function

ordinate: y_i

5.3.3.5 Fit regression lines per search interval

Fit a linear regression line in each of the search intervals (P15, P65), (P25, P75), (P35, P85), where Pxx is the xxth percentile of the data, and calculate the associated r^2.

5.3.3.6 Update the optimality criterion

If $r^2 > r_{max}^2$, set $\mathrm{r}_{\max}^2 = \mathrm{r}^2$, $\lambda_{\mathrm{ini}} = \lambda$, $\mu_{\mathrm{ini}} = \beta_0$, $\sigma_{\mathrm{ini}} = \beta_1$.

β_0 and β_1 are the regression coefficients (intercept and slope) from 5.3.3.5.

5.3.3.7 Increase λ until its maximum

Increase λ by 0.2. If the resulting λ is <= 1, continue with 5.3.3.3, otherwise continue with 5.3.3.8.

5.3.3.8 Report the initial value

The values λ_{ini}, μ_{ini} and σ_{ini} recorded in 5.3.3.6 are the initial values for the subsequent TMC procedure.

5.3.4 Estimate PND parameters by the TMC procedure for all truncation interval candidates

Rationale: Results from step 5.3.3 are only approximations for the reasons given in 5.4 below. This holds for the parameter estimates and the location of the truncation interval found by the QQ plot approach. QQ plot parameter estimates are, however, precise enough to serve as initial values for the subsequent iterative TMC approach, which does not suffer from the structural problems that the QQ plot has.

The TMC approach has to find a truncation interval with the properties given in 5.2. To this end, a sequence of permissible truncation interval candidates is defined, a parameter estimate for (λ, μ, σ) is calculated for each candidate interval, and the final estimate is selected as the estimate with optimal properties. All truncation interval candidates contain the mode of the data. They contain at least `x.tr.bins.min` bins (default: 5). Also, the percentage of data contained in the truncation interval is restricted to lie between `x.tr.prop.min` (default: 0.60) and `x.tr.prop.max` (default: 0.85). For each truncation interval candidate the iterative estimation procedure below is executed.

5.3.4.1 Find the first truncation interval candidate

The first truncation interval that is considered in the following search is the smallest interval with following properties: the empirical mode of the data is contained in the interval, the number of bins is at least `x.tr.bins.`min and it contains at least a proportion of `x.tr.prop.`min of all values. If there is more than 1 interval with these properties, the leftmost of these is used.

5.3.4.2 Calculate PND estimates for the actual truncation interval

Calculate estimates for the PND parameters λ, μ, σ from the data in the actual truncation interval candidate. Parameters are chosen such that the estimated distribution fits as good as possible to the empirical histogram in the truncation interval, while predictions outside the truncation interval must not produce illogical results (like predicting the uncontaminated part of the dataset being larger than the total dataset).

The parameters λ, μ, σ are estimated by an iterative Newton-Raphson procedure. The procedure uses the result λ_{ini}, μ_{ini} and σ_{ini} from 3.10 as initial values.

The criterion to minimise is the penalised chi-square distance

$$D(\lambda,\mu,\sigma) = \sum_{i\in T} g_i + \sum_{j\notin T} w_j \qquad \text{(A4)}$$

Here, T is the set of bin indices [ci, ci+1) contained in the truncation interval, g_i is the chi-square contribution from interval i, and w_j is the penalty term for interval j. The χ^2 contribution of interval [ci, ci+1) is defined by

$$g_i = \frac{\left(n_i - N_i(\lambda,\mu,\sigma)\right)^2}{N_i(\lambda,\mu,\sigma)} \qquad \text{(A5)}$$

where N_i is the expected number of values in interval i, given by

$$N_i(\lambda,\mu,\sigma) = \frac{F(c_{i+1};\lambda,\mu,\sigma) - F(c_i;\lambda,\mu,\sigma)}{F(t_{hi};\lambda,\mu,\sigma) - F(t_{lo};\lambda,\mu,\sigma)} \sum_{k\in T} n_k \qquad \text{(A6)}$$

This is a conditional expectation, as only the distribution of values in the truncation interval is considered. The sum of all χ^2 contributions from the truncation interval has an asymptotical χ^2 distribution and is used to test the goodness of fit in the truncation interval. The corresponding p value p_{fit} is

$$p_{fit} = P\left(\sum_{i\in T} g_i > \chi^2_{|T|-4}\right) \qquad \text{(A7)}$$

It is an approximate measure, because the previous operations for finding the truncation interval are not accounted for.

The penalty term for interval j in (A4) is defined by

$$w_j = \varepsilon\, g_j\, \mathrm{P}(\chi^2_1 < \delta) \qquad \text{(A8)}$$

where $\mathrm{P}(\chi^2_1 < \delta)$ is the χ2 distribution function with 1 degree of freedom, δ is the difference between expected and observed counts, if this is positive and otherwise zero,

$$\delta = \max(N_j - n_j, 0) \qquad \text{(A9)}$$

and ε is a weighting factor (`w.fact`, default: 1). The penalty term contributes to the optimality criterion (A4) only in those data intervals, for which the predicted count is larger than the observed one. Also, w_j gives a considerable contribution only if δ is outside the range of random fluctuation of a $\chi^2(1)$ random variable. Fig. 1 below displays a histogram with marked truncation intervals, observed and expected counts, and Table 2 in (2) provides details of the calculation leading to Fig. 1.

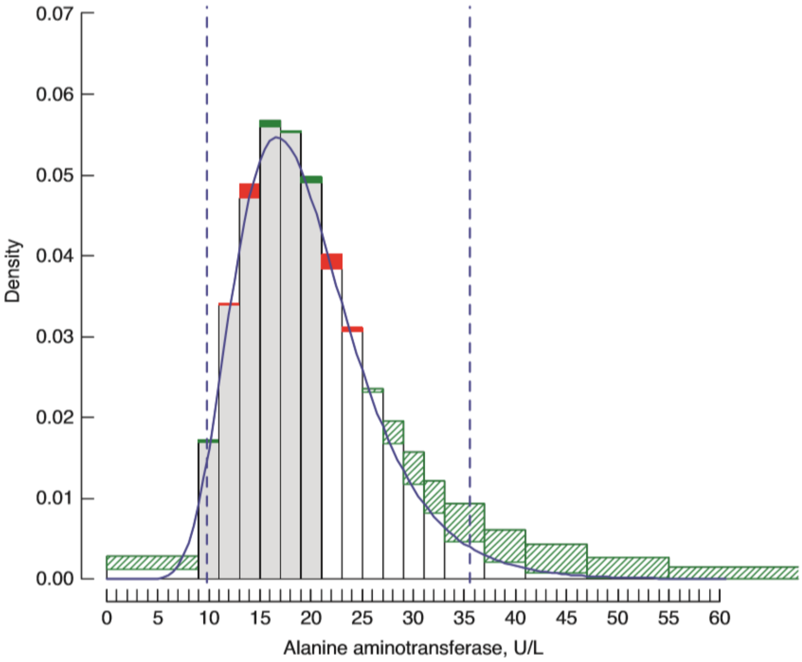


Fig. 1 (reproduced from (2)): Grey bins indicate the truncation interval, white bins lie outside the truncation interval. Observed bin proportions are white coloured areas + green areas and white coloured areas without red areas. The blue PND probability density curve is fitted by the TMC approach. Solid red and green rectangles indicate the differences between observed and expected counts which contribute to the χ^2 criterion (A5). Red rectangles indicate bins in which the expected count is larger than the observed. These rectangles contribute to (A5) inside and to (A8) outside the truncation interval. Bins outside the truncation interval with expected count smaller than observed, marked by green hatched rectangles, do not contribute to (A5). The vertical dashed blue lines indicate the 2.5% and 97.5% RILs.

5.3.4.3 Record relevant information for the actual truncation interval

Record the truncation interval limits, the parameter estimates λ, μ, σ for the actual truncation interval, the fit parameters ($D(\lambda, \mu, \sigma)$ from A4, p_{fit} from A7) and the degrees of freedom $df = |T| - 4$ for the fit parameter (A7).

5.3.4.4 Consider the next truncation interval candidate

If the right limit of the actual truncation interval is below the maximum of the data, shift the truncation interval one bin to the right and continue with step 5.3.4.2.

If the right limit of the actual truncation interval is equal to the maximum of the data, move the truncation interval to its leftmost position and try to add one bin at the right. If this is possible, continue with step 5.3.4.2., otherwise continue with step 5.1

5.3.5 Identify the optimal PND parameter among the candidates considered in step 5.3.4

Sort the table set up in step 5.3.4.3 by p_{fit}. First consider only truncation intervals with $p_{fit} > 0.20$. This limit is set by the parameter `p.fit.min`. If such intervals exist, the one with largest proportion of data in it is the optimal truncation interval. The corresponding parameters λ, μ, σ define the optimal PND for unaffected values.

If there are no truncation intervals with fit $p_{fit} > 0.20$, a warning is issued. The truncation interval with minimal optimality criterion $D(\lambda, \mu, \sigma)/df$, where $df = |T| - 4$ is the number of degrees of freedom from (A7), is now used as optimal truncation interval. This situation indicates an insufficient fit of the estimated distribution to the observed data. Possible reasons are inappropriate stratification or a violation of the essential assumptions formulated in 5.2.

The decision process is for each stratum displayed in a figure the characterization F102.010.

5.3.6 Calculate the RLs

The RILs are calculated as quantiles of the optimal PND found in step 5. Typically, the 2.5% and the 97.5% quantiles are used. Quantiles are calculated by inversion of equation (A3).

5.4 A remark on QQ plot regression for incomplete data

A QQ plot is a convenient method to display graphically some distributional properties of a data set. In a normal QQ plot, normally (Gaussian) distributed data fluctuate around a straight line with intercept and slope of this line approximating mean and standard deviation of the distribution. If the data consists of two normally and not overlapping distributions, the QQ plot shows two different nearly straight lines. If the distributions overlap, curved structures arise, but there may still be (nearly) linear components. In either of these situations one might try to estimate intercepts and slopes and use them as means and standard deviations of the distributions involved. However, this is only an approximation, as is easily seen from the construction of a QQ plot. The horizontal axis carries the expected positions of the ordered values. These positions, denoted by E(x[i]), i = 1, 2, …, n, where n is the dataset size, are calculated as

$$\mathrm{E}\left(\mathrm{x}_{[\mathrm{i}]}\right) = \Phi^{-1}((\mathrm{i} - 0.5)/\mathrm{n}) \tag{A10}$$

where Φ is the standard normal distribution function. This formula contains the sample size n, which is clearly available if only one distribution is involved. If two (different) distributions are involved, a QQ plot will no more show points fluctuating around straight lines, because all data points have wrong positions on the abscissa. This is due to the construction in (A10), which treats all data as coming from the same distribution, while in fact positions for two datasets are needed. These cannot be calculated, because the size of the samples is unknown as well as the sample membership of each value. The consequence of this situation can be seen with artificial test data consisting of two datasets which each contain the expected order statistics (A10) as values. A QQ plot showing correctly the associated straight lines with the correct parameters of the underlying distributions could only be constructed if the sample sizes were known as well as the origin of each point (first or second distribution?). For indirect methods of RI estimation, this information is not available. However, as indirect methods use to operate with large datasets, the error in estimating distributional parameters from a QQ plot is small enough to allow using these estimates as starting values for a Newton-Raphson procedure.

5.5 A remark on rounding

Laboratory data is usually rounded data. Amount and style of rounding influence the analysis of an empirical distribution, in particular the possibility of detecting a deviation from an assumed shape, which is a core component of an indirect method. Several software products, including the R package, use the principle of “go to the even digit”, following IEC 60559. This principle introduces an artificial fluctuation in an empirical distribution (see Fig. 2 in (2) or try `table(round(seq(0,4.9, by=0.1)))` in R) and is therefore unfavourable for an indirect method. Therefore, TMC rounds values < 0.5* rounding unit to the lower unit and ≥ 0.5*rounding unit to the higher unit. The plot with code F100.010 (see 4) is an aid to detect unfavourable rounding that may have happened during data recording. If this plot suggests unfavourable rounding present in the data, additional roundíng by TMC using the parameter `round.unit`, might remove the problem.

6 References

1. **R Core Team.** R: a language and environment for statistical computing. [Hrsg.] R Foundation for Statistical Computing. 2017.

2. **Wosniok, W. und Haeckel, R.** A new indirect estimation of reference intervals: truncated minimum chi-square (TMC) approach. *Clin Chem Lab Med.* 2019, Bd. 57, 12, S. 1933-1947. https://doi.org/10.1515/cclm-2018-1341.

3. **Freeman J, Modarres R.** Inverse Box-Cox: the power-normal distribution. *Stat Probabil Lett.* 2016, Bd. 76, S. 764–772.