

Package ‘IsoCorrectoR’

July 23, 2025

Title Correction for natural isotope abundance and tracer purity in MS
and MS/MS data from stable isotope labeling experiments

Version 1.26.0

Imports dplyr, magrittr, methods, quadprog, readr, readxl, stringr,
tibble, tools, utils, pracma, WriteXLS

Description IsoCorrectoR performs the correction of mass spectrometry data from stable isotope labeling/tracing metabolomics experiments with regard to natural isotope abundance and tracer impurity. Data from both MS and MS/MS measurements can be corrected (with any tracer isotope: ^{13}C , ^{15}N , ^{18}O ...), as well as ultra-high resolution MS data from multiple-tracer experiments (e.g. ^{13}C and ^{15}N used simultaneously). See the Bioconductor package IsoCorrectoRGUI for a graphical user interface to IsoCorrectoR. NOTE: With R version 4.0.0, writing correction results to Excel files may currently not work on Windows. However, writing results to csv works as before.

Depends R (≥ 3.5)

URL <https://genomics.ur.de/files/IsoCorrectoR/>

License GPL-3

LazyData TRUE

NeedsCompilation no

biocViews Software, Metabolomics, MassSpectrometry, Preprocessing,
ImmunoOncology

RoxygenNote 6.1.1

Suggests IsoCorrectoRGUI, knitr, rmarkdown, testthat, BiocStyle

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/IsoCorrectoR>

git_branch RELEASE_3_21

git_last_commit 96af901

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-07-23

Author Christian Kohler [cre, aut],
Paul Heinrich [aut]

Maintainer Christian Kohler <christian.kohler@ur.de>

Contents

IsoCorrection	2
IsoCorrectoR	4
Index	6

IsoCorrection	<i>Algorithm For Natural Isotope Abundance And Tracer Purity Correction Of Data From Stable Isotope Labeling Experiments</i>
---------------	--

Description

IsoCorrection is the main function of the IsoCorrectoR package. It performs the correction of mass spectrometry data from stable isotope labeling experiments with regard to natural abundance and tracer purity. Data from both MS and MS/MS experiments can be corrected (with any tracer isotope: ^{13}C , ^{15}N , ^{18}O ...), as well as high resolution data from multiple-tracer experiments (e.g. ^{13}C and ^{15}N used simultaneously).

Usage

```
IsoCorrection(MeasurementFile = NA, ElementFile = NA, MoleculeFile = NA,
  CorrectTracerImpurity = FALSE, CorrectTracerElementCore = TRUE,
  CalculateMeanEnrichment = TRUE, UltraHighRes = FALSE, DirOut = ".",
  FileOut = "result", FileOutFormat = "csv", ReturnResultsObject = TRUE,
  CorrectAlsoMonoisotopic = FALSE, CalculationThreshold = 10^-8,
  CalculationThreshold_UHR = 8, verbose = FALSE, Testmode = FALSE)
```

Arguments

MeasurementFile	Required. The file that contains the measured data to be corrected. Only ".xls", ".xlsx" and ".csv" file formats are supported.
ElementFile	Required. The file that contains the element information required for correction. Only ".xls", ".xlsx" and ".csv" file formats are supported.
MoleculeFile	Required. The file that contains the information on the molecules for which data is to be corrected. Only ".xls", ".xlsx" and ".csv" file formats are supported.
CorrectTracerImpurity	Logical. If TRUE, correction for isotopic impurity of the tracer substrate is performed.

CorrectTracerElementCore	Logical. If TRUE (recommended!), the tracer element atoms in the core module (usually the part of the molecule that does not come from derivatization) are considered when correcting.
CalculateMeanEnrichment	Logical. If TRUE, the mean isotopic enrichment is calculated for each molecule.
UltraHighRes	Logical. If TRUE, high resolution correction is performed on the data. Should only be set to TRUE, if you know that you have high resolution data.
DirOut	Character String. Defines the directory the corrected data and log-file should be written to. Default directory is set to current working directory ('.').
FileOut	Character String. Defines the name of the file that contains the corrected data. The name of the file will be IsoCorrectoR_<FileOut>.<FileFormat>. If the format is set to "csv", the name will also contain the type of the corrected data in the respective file.
FileOutFormat	Character String. Defines the format of the files that contain the corrected data. Can either be "csv" or "xls". If set to "csv", multiple files will be generated, one for each type of corrected data (eg. corrected data, fractions, mean enrichment...). If set to "xls", all correction results are provided in one excel file in different sheets.
ReturnResultsObject	Logical. If TRUE, the correction results are returned as a list in the current R_session in addition to writing the results to a file. This is useful if the corrected data has to be further processed directly in R.
CorrectAlsoMonoisotopic	Logical. If TRUE, monoisotopic correction results are also provided.
CalculationThreshold	(Advanced Option) Numeric. Defines a threshold to stop probability calculations at for making correction faster (normal resolution mode). Should be left at the default value.
CalculationThreshold_UHR	(Advanced Option) Numeric. Defines a threshold to stop probability calculations at for making correction faster (high resolution mode). Should be left at the default value.
verbose	Logical. If TRUE, status messages are sent to standard output.
Testmode	Logical. If TRUE, starts a testmode for development purposes. Not required for users of IsoCorrectoR.

Value

The function returns a list with 4 elements

success: string that is "TRUE" if the correction was successful, "FALSE" if an error occurred and "WARNINGS" if warnings occurred

results: a list containing a dataframe for each type of corrected data (normal, fractions, mean enrichment ...). Will be NA if ReturnResultsObject is set to FALSE

log: list containing log information on the correction run (parameters, file names and paths, warnings and errors)

error: contains a string with the associated error message if an error occurred, empty otherwise

References

See Reference 1 [LinktoIsoCorrectoR-Paper](#)

Examples

```
# Normal resolution data

# 1) get path of example files
path.molecule <- system.file("extdata","normal_resolution","MoleculeFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);
path.element <- system.file("extdata","normal_resolution","ElementFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);
path.measurement <- system.file("extdata","normal_resolution","MeasurementFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);

# 2) run correction algorithm and save results in variable
correctionResults <- IsoCorrection(MeasurementFile=path.measurement,
  ElementFile=path.element,
  MoleculeFile=path.molecule)

# High resolution data

# 1) get path of example files
path.molecule <- system.file("extdata","high_resolution","MoleculeFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);
path.element <- system.file("extdata","high_resolution","ElementFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);
path.measurement <- system.file("extdata","high_resolution","MeasurementFile.csv",
  package = "IsoCorrectoR", mustWork = TRUE);

# 2) run correction algorithm and save results in variable
correctionResults <- IsoCorrection(MeasurementFile=path.measurement,
  ElementFile=path.element,
  MoleculeFile=path.molecule,UltraHighRes=TRUE)
```

IsoCorrectoR

IsoCorrectoR example data

Description

Example data for the IsoCorrectoR package

Usage

```
data(IsoCorrectoR)
```

Format

A list with four elements, namely

tool_features: overview of correction features as provided by various tools IsoCorrectoR is compared with.

element_file: example data for an Element file. This information is independent of the resolution type (normal or high)

normal_resolution: list with two elements, containing examples for normal resolution data: molecule information and measurement data

high_resolution: list with two elements, containing examples for high resolution data: molecule information and measurement data

Author(s)

Paul Heinrich, Christian Kohler

Index

* **datasets**

 IsoCorrector, [4](#)

 IsoCorrection, [2](#)

 IsoCorrector, [4](#)