Package 'iMRMC'

May 14, 2024

Type Package

Title Multi-Reader, Multi-Case Analysis Methods (ROC, Agreement, and Other Metrics)

Version 2.0.0

Author Brandon Gallas

Maintainer Brandon Gallas <Brandon.Gallas@fda.hhs.gov>

Description This software does Multi-Reader, Multi-Case (MRMC) analyses of data from imaging studies where clinicians (readers) evaluate patient images (cases). What does this mean? ... Many imaging studies are designed so that every reader reads every case in all modalities, a fully-crossed study. In this case, the data is cross-correlated, and we consider the readers and cases to be cross-correlated random effects. An MRMC analysis accounts for the variability and correlations from the readers and cases when estimating variances, confidence intervals, and p-values. The functions in this package can treat arbitrary study designs and studies with missing data, not just fully-crossed study designs. An overview of this software, including references presenting details on the methods, can be found here: <https://www.fda.gov/medical-devices/science-and-research-medical-devices/imrmc-software-do-multi-reader-multi-case-statistical-analysis-reader-studies>.

URL https://github.com/DIDSR/iMRMC, https://cran.r-project.org/web/packages/iMRMC/index.html

License CC0

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

RoxygenNote 7.3.1

Imports stats

Suggests testthat

R topics documented:

---

convertDF   *Convert MRMC data frames*

Description

Convert MRMC data frames

Usage

```
convertDF(inDF, inDFtype, outDFtype, readers)
```

Arguments

inDF	An MRMC dataframe with reading study results in `inDFtype` format
inDFtype	A string indicating the format type of the input MRMC data frame
outDFtype	A string indicating the format type of the output MRMC data frame.
readers	A character array holding the column names (readerIDs) corresponding to the data from different readers

Details

MRMC data frames contain scores from readers, cases, and sometimes modalities. This packages deals with (currently) two MRMC data frame formats. These are the formats:

matrixMode For this format, each row contains all the information for a case, including the reader study result for several readers, ground truth and other information as seperate columns. This mode can only hold data from one modality.

listMode For this format, each row contains the data for one observation. The columns specify the readerID, caseID, score, ground truth, and other information if there is any. This mode can hold data from multiple modalities.

Value

An MRMC dataframe with reading study results in `outDFtype` format

---

convertDFtoDesignMatrix

Convert an MRMC data frame to a design matrix

Description

Convert an MRMC data frame to a design matrix, dropping readers or cases with no observations

Usage

```
convertDFtoDesignMatrix(dfMRMC, modality = NULL, dropFlag = TRUE)
```

Arguments

dfMRMC	An MRMC data frame
modality	The score matrix depends on the modality. If more than one modality exists in the data frame, you must specify which modality to subset.
dropFlag	[logical] The default setting (TRUE) removes readers and cases that have no observations. Dropping them by default will speed up analyses. Leaving the levels (dropFlag = FALSE) is useful if you need the entire score or design matrix when comparing or doing analyses with two modalities.

Value

A matrix that is *nCases* by *nReaders* indicating which scores were reported for each reader and case

---

convertDFtoScoreMatrix

Convert an MRMC data frame to a score matrix

Description

Convert an MRMC data frame to a score matrix, dropping readers or cases with no observations

Usage

```
convertDFtoScoreMatrix(dfMRMC, modality = NULL, dropFlag = TRUE)
```

Arguments

dfMRMC	An MRMC data frame
modality	The score matrix depends on the modality. If more than one modality exists in the data frame, you must specify which modality to subset.
dropFlag	[logical] The default setting (TRUE) removes readers and cases that have no observations. Dropping them by default will speed up analyses. Leaving the levels (dropFlag = FALSE) is useful if you need the entire score or design matrix when comparing or doing analyses with two modalities.

Value

A matrix that is *nCases* by *nReaders* of the scores each reader reported for each case

---

createGroups

Assign a group label to items in a vector

Description

Assign a group label to items in a vector

Usage

```
createGroups(items, nG)
```

Arguments

items	A vector of items
nG	The number of groups

Value

A data frame containing the items and their group labels

Examples

```
x <- paste("item", 1:10, sep = "")
df <- createGroups(x, 3)
print(df)
```

---

createIMRMCdf   *Convert a data frame with all needed factors to doIMRMC formatted*
                                    data frame

Description

Convert a data frame with all needed factors to doIMRMC formatted data frame

Usage

```
createIMRMCdf(
  dFrame,
 keyColumns = list(readerID = "readerID", caseID = "caseID", modalityID = "modalityID",
    score = "score", truth = "truth"),
  truePositiveFactor = "cancer"
)
```

Arguments

dFrame	This data frame includes columns for readerID, caseID, modalityID, score, and truth. These columns are not expected to be named as such and other columns may exist.
keyColumns	This list identifies the column names of the data frame to be used for the analysis. list(readerID = "***", caseID = "***", modalityID = "***", score = "***", truth="***")
truePositiveFactor	
	The true positive label, such as "cancer" or "1"

Value

output a doIMRMC formatted data frame: rows for truth and rows for data. The results will be an iMRMC formatted data frame, see dfMRMC_example

---

deleteCol   *Delete a data frame column*

Description

Delete a data frame column

Usage

```
deleteCol(df, colName)
```

Arguments

df	A data frame
colName	Column name or list of column names to be deleted

Value

The data frame without the deleted column or columns

---

dfMRMC_example   *Example of iMRMC formatted data frame*

Description

An example data frame formatted for 'doIMRMC' and other iMRMC functions.

Usage

```
dfMRMC_example
```

Format

A data frame with 880 rows and 4 columns:

readerID Factor with 5 levels like "reader1", "reader2", ... As well as the special reader "truth"

caseID Factor with 80 levels like "case1", "case2", ...

modalityID Factor with 2 levels like "modality1", "modality2", ... As well as the special modality "truth"

score Numeric reader score

Each row of this data frame corresponds to an observation. For every caseID, there must be a row corresponding to the truth observation. The readerID for a truth observation is "truth". The modalityID for a truth observation is "truth". The score for a truth observation must be either 0 (signal-absent) or 1 (signal-present).

Examples

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dfMRMC_example <- sim.gRoeMetz(config)
# Analyze the MRMC ROC data
result <- doIMRMC(dfMRMC_example)
```

---

doAUCmrmc   *MRMC analysis for arbitrary design dataset*

Description

Execute a Multi-Reader, Multi-Case (MRMC) analysis of ROC data from imaging studies where clinicians (readers) evaluate patient images (cases). An overview of this software, including references presenting details on the methods, can be found HERE or as an entry in the FDA/CDRH Regulatory Science Tool Catalog HERE.

Usage

```
doAUCmrmc(data, flagROC = FALSE)
```

Arguments

data   an iMRMC formatted data frame, see dfMRMC_example

flagROC   boolean indicating if ROC results should be computed.

Value

The MRMC analysis results as a list, below is a quick summary:

- summaryMRMC, list a list of summary study design information.

    - nM, num number of modalities
    - nR, num number of readers
    - nC.neg, num number of signal-present caeses
    - nC.pos, num number of signal-absent cases
    - modalites, char names of modalities
    - readers, char names of modalities
    - cases.neg, char names of modalities
    - cases.pos, char names of modalities

- perReader.full, data.frame this data frame contains the performance results for each reader and modality comparison. The analysis returns the final AUC results and the moments, coefficients of those moments. Key variables of this data frame are AUC.1 (where '.1' indicates the row's reader and modality '.1' pair), AUC.2 ('.2' indicates the '.2' reader and modality pair), and covAUC.

- Ustat.full, data.frame this data frame contains the reader-average AUC performance results. The analysis results are based on U-statistics. Key variables of this data frame are AUC.1, AUC.2, AUC1minusAUC2 and the corresponding variances, confidence intervals, degrees of freedom and p-values.

- ROC, list each object of this list is an object containing an ROC curve. There is an ROC curve for every combination of reader and modality. For every modality, there are also four average ROC curves. These are discussed in Chen2014_Br-J-Radiol_v87p20140016. The diagonal average averages the reader-specific ROC curves along $y = -x + b$ for b in (0,1). The horizontal average averages the reader specific ROC curves along $y = b$ for b in (0,1). The vertical average averages the reader specific ROC curves along $x = b$ for b in (0,1). The pooled average ignores readerID and pools all the scores together to create one ROC curve.

Examples

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dFrame.imrmc <- sim.gRoeMetz(config)
# Analyze the MRMC ROC data and compute ROC curves
aucResult <- doAUCmrmc(dFrame.imrmc, flagROC = TRUE)
```

---

doIMRMC   *MRMC analysis for arbitrary design dataset*

Description

Execute a Multi-Reader, Multi-Case (MRMC) analysis of ROC data from imaging studies where clinicians (readers) evaluate patient images (cases). This function is a wrapper that executes doAUCmrmc and formats the output to generally match the output of doIMRMC version 1.2.5. An overview of this software, including references presenting details on the methods, can be found HERE or as an entry in the FDA/CDRH Regulatory Science Tool Catalog HERE.

Usage

```
doIMRMC(data)
```

Arguments

data   an iMRMC formatted data frame, see dfMRMC_example

Value

The MRMC analysis results, below is a quick summary:

- perReader, data.frame The performance results for each combination of reader and pair of modalities. Key variables of this data frame are AUCA, AUCB, AUCAminusAUCB and the corresponding variances. When the modalities differ, the variance is understood to be the covariance between the modalities.

- Ustat, data.frame Reader-averaged performance results for each pair of modalities. The analysis results are based on U-statistics. Key variables of this data frame are AUCA, AUCB, AUCAminusAUCB and the corresponding variances, confidence intervals, degrees of freedom, and p-values. When the modalities differ, the variance is understood to be the covariance between the modalities.

- MLEstat, data.frame Reader-average performance results for each pair of modalities. The analysis results are based on V-statistics, which approximates the true distribution with the empirical distribution. The empirical distribution equals the nonparametric MLE estimate of the true distribution, which is also equivalent to the ideal bootstrap estimate. Key variables of this data frame are AUCA, AUCB, AUCAminusAUCB and the corresponding variances, confidence intervals, degrees of freedom, and p-values. When the modalities differ, the variance is understood to be the covariance between the modalities.

- varDecomp, list list of data frames of the coefficient and components of variance. The analysis includes variance decomposition based off both the BDG and BCK MRMC methods, and Ustat and MLE statistical methods. Each MRMC and statistical method combination is contained within this list of lists.

- ROC, list each object of this list is an object containing an ROC curve. There is an ROC curve for every combination of reader and modality. For every modality, there are also four average ROC curves. These are discussed in Chen2014_Br-J-Radiol_v87p20140016. The diagonal average averages the reader-specific ROC curves along y = -x + b for b in (0,1). The horizontal average averages the reader specific ROC curves along y = b for b in (0,1). The vertical average averages the reader specific ROC curves along x = b for b in (0,1). The pooled average ignores readerID and pools all the scores together to create one ROC curve.

- full, list This returns the same result as doAUCmrmc.

Examples

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dFrame.imrmc <- sim.gRoeMetz(config)
# Analyze the MRMC ROC data
result <- doIMRMC(dFrame.imrmc)
```

---

doIMRMC_java   *MRMC analysis of the area under the ROC curve*

Description

doIMRMC_java takes ROC data as a data frame and runs a multi-reader multi-case analysis based on U-statistics as described in the following papers Gallas2006_Acad-Radiol_v13p353 (single-modality), Gallas2008_Neural-Networks_v21p387 (multiple modalities, arbitrary study designs), Gallas2009_Commun-Stat-A-Theor_v38p2586 (framework paper). This function is deprecated, please use doIMRMC instead.

Usage

```
doIMRMC_java(
  data = NULL,
  fileName = NULL,
  workDir = NULL,
  iMRMCjarFullPath = NULL,
  stripDatesForTests = FALSE
)
```

Arguments

data   an iMRMC formatted data frame, see dfMRMC_example

fileName   This character string identifies the location of an iMRMC input file. The input file is identical to data except there is a free text section to start, then a line with "BEGIN DATA:", then the data frame info.

workDir   This character string determines the directory where intermediate results are
                     written. If this parameter is not set, the program writes the intermediate results
                     to the directory specified by tempdir() and then deletes them.

iMRMCjarFullPath
                     This character string identifies the location of the iMRMC.jar file this jar file can
                     be downloaded from https://github.com/DIDSR/iMRMC/releases this R pro-
                     gram supports version iMRMC-v3p2.jar

stripDatesForTests
                     Since results include a date and time stamp, these need to be stripped out when
                     doing the package tests. This parameter flags whether or not the dates should be
                     stripped out.

Details

In detail, this procedure reads the name of an input file from the local file system, or takes a
data frame and writes it to the local file system formatted for the iMRMC program (found at
https://github.com/DIDSR/iMRMC/releases), it executes a java app, the iMRMC engine, which
writes the results to the local files system, it reads the analysis results from the local file system,
packs the analysis results into a list object, deletes the data and analysis results from the local file
system, and returns the list object.

This software requires Java(>=8).

The examples took too long for CRAN to accept. So here is an example:

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dFrame.imrmc <- sim.gRoeMetz(config)
# Analyze the MRMC ROC data
result <- doIMRMC_java(dFrame.imrmc)
```

Value

[list] iMRMC outputs. The objects of this list are described in detail in the iMRMC documentation
which can be found at <http://didsr.github.io/iMRMC/000_iMRMC/userManualHTML/index.htm>

Here is a quick summary:

- perReader data.frame containing the performance results for each reader. Key variables of
  this data frame are AUCA, AUCB, AUCAminusAUCB and the corresponding variances, con-
  fidence intervals, degrees of freedom and p-values.

- Ustat data.frame containing the reader-average performance results. The analysis results are
  based on U-statistics and the papers listed above. Key variables of this data frame are AUCA,
  AUCB, AUCAminusAUCB and the corresponding variances, confidence intervals, degrees of
  freedom and p-values.

- MLEstat data.frame containing the reader-average performance results. The analysis results
  are based on V-statistics, which approximates the true distribution with the empirical distri-
  bution. The empirical distribution equals the nonparametric MLE estimate of the true dis-
  tribution, which is also equivalent to the ideal bootstrap estimate. Please refer to the papers
  listed above. Key variables of this data frame are AUCA, AUCB, AUCAminusAUCB and the
  corresponding variances, confidence intervals, degrees of freedom and p-values.

- ROC list containing ROC curves There is an ROC curve for every combination of reader and
  modality. For every modality, there are also four average ROC curves. These are discussed

in Chen2014_Br-J-Radiol_v87p20140016. The diagonal average averages the reader-specific ROC curves along y = -x + b for b in (0,1). The horizontal average averages the reader specific ROC curves along y = b for b in (0,1). The vertical average averages the reader specific ROC curves along x = b for b in (0,1). The pooled average ignores readerID and pools all the scores together to create one ROC curve.

- varDecomp list containing different decompositions of the total variance. Please refer to Gallas2009_Commun-Stat-A-Theor_v38p2586 (framework paper). The different decompositions are BCK, BDG, DBM, MS, OR.

---

doROCavg   *Empirically average over multiple empirical ROC curves*

Description

Empirically average over multiple empirical ROC curves

Usage

```
doROCavg(ROC, direction = "SeSp")
```

Arguments

ROC   list of ROC curves. Each element of the list is a data frame with pairs of (fpf, tpf) operating points.

direction   the direction over which to average

- SeSp (default) The ROC curves are averaged diagonally. Average Se+Sp of all the input ROC curves for every possible Se-Sp.
- Se The ROC curves are averaged vertically. Average the sensitivity of all the input ROC curves for every possible specificity.
- Sp The ROC curves are averaged horizontally. Average the specificity of all the input ROC curves for every possible specificity.

Value

data frame of an ROC curve

- fpf False-positive fractions (1-specificity)
- tpf True-positive fractions (sensitivity)

doROCcurveMRMC   *Create a standard set of ROC curves from an MRMC data frame*

Description

Create a standard set of ROC curves from an MRMC data frame

Usage

```
doROCcurveMRMC(mrmcAlternate)
```

Arguments

mrmcAlternate   data frame

- `readerID`
- `caseID`
- `modalityID`
- `score`
- `truthLabel`

Value

list of ROC curves corresponding to all reader x modality combinations, all modalities with the reader scores pooled, all modalities with the per-reader ROC curves, horizontally, diagonally, and vertically averaged.

doROCxy   *Create empirical ROC curve*

Description

Create empirical ROC curve

Usage

```
doROCxy(sa, sp)
```

Arguments

sa   signal-absent scores
sp   signal-present scores

Value

data frame of an ROC curve

- `fpf` False-positive fractions (1-specificity)
- `tpf` True-positive fractions (sensitivity)
- `threshold` Threshold corresponding to each fpf, tpf operating point

Examples

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dFrame.imrmc <- sim.gRoeMetz(config)
# Isolate signal absent scores
indexSA <- grep("negCase", dFrame.imrmc$caseID)
sa <- dFrame.imrmc[indexSA, ]$score
# Isolate signal present scores
indexSP <- grep("posCase", dFrame.imrmc$caseID)
sp <- dFrame.imrmc[indexSP, ]$score
# Compute empirical ROC curve
result <- doROCxy(sa, sp)
```

---

doROCxyMRMC   *Create empirical ROC curve from an MRMC formatted data frame*

Description

Create empirical ROC curve from an MRMC formatted data frame

Usage

```
doROCxyMRMC(mrmcAlternate)
```

Arguments

mrmcAlternate   data frame

- readerID
- caseID
- modalityID
- score
- truthLabel

Value

data frame of an ROC curve

- modalityID
- readerID
- fpf False-positive fractions (1-specificity)
- tpf True-positive fractions (sensitivity)
- threshold Threshold corresponding to each fpf, tpf

extractPairedComparisonsBRBM
Extract between-reader between-modality pairs of scores

Description

Extract between-reader between-modality pairs of scores

Usage

```
extractPairedComparisonsBRBM(
  data0,
  modalities = c("testA", "testB"),
 keyColumns = list(readerID = "readerID", caseID = "caseID", modalityID = "modalityID",
    score = "score")
)
```

Arguments

data0	This data frame includes columns for readerID, caseID, modalityID, score.
modalities	The modalities (testA, testB) for the scores to be paired
keyColumns	This list identifies the column names of the data frame to be used for the analysis. list(readerID = "***", caseID = "***", modalityID = "***", score = "***", truth="***")

Value

A data frame of all paired observations. Each observation comes from a pair of readers evaluating a case in two modalities. The first column corresponds to one reader evaluating the case in testA. The second column corresonds to the other reader evaluating the case in testB.

---

extractPairedComparisonsWRBM
Extract within-reader between-modality pairs of scores

Description

Extract within-reader between-modality pairs of scores

Usage

```
extractPairedComparisonsWRBM(
  data0,
  modalities = "testA",
 keyColumns = list(readerID = "readerID", caseID = "caseID", modalityID = "modalityID",
    score = "score")
)
```

Arguments

`data0`	This data frame includes columns for readerID, caseID, modalityID, score.
`modalities`	The modalities (testA, testB) for the scores to be paired
`keyColumns`	This list identifies the column names of the data frame to be used for the analysis. list(readerID = "***", caseID = "***", modalityID = "***", score = "***", truth="***")

Value

A data frame of all paired observations. Each observation comes from a one reader evaluating a case in two modalities The first column corresponds to one reader evaluating the case in "testA". The second column corresonds to the same reader evaluating the case in "testB".

---

getBRBM	*Get between-reader, between-modality paired data from an MRMC data frame*

Description

Get between-reader, between-modality paired data from an MRMC data frame

Usage

```
getBRBM(mcsData, modality.X, modality.Y)
```

Arguments

`mcsData`	A data frame with the following columns: readerID, caseID, modalityID, score
`modality.X`	The name of one modality
`modality.Y`	The name of one modality.

Details

If modality.Y = modality.X, then the data would be between-reader, within-modality (BRWM).

Value

The result of merging the modality.X and modality.Y subsets of mcsData by caseID for every pair of readers

getMRMCdataset	*Import MRMC dataset from the web (https://github.com/DIDSR/iMRMC/wiki/iMRMC-Datasets)*

Description

Import MRMC dataset from the web (https://github.com/DIDSR/iMRMC/wiki/iMRMC-Datasets)

Usage

```
getMRMCdataset(
  dataset = c("pilotHTT", "viperObs", "viperObs365", "viperObs455",
    "MFcounts_dfClassify", "MFcounts_dfCountROI", "MFcounts_dfCountWSI",
  "cardioStudyTruth", "cardioStudyRawData", "prostateTruth", "prostateRawData")
)
```

Arguments

dataset   Possible dataset options available:

- **pilotHTT**: Breast cancer annotation data that is the aggregate of all clean data from the HTT project pilot study. https://github.com/DIDSR/HTT
- **viperObs**: Individual observations of each reader reading each case from the FDA Validation of Imaging Premarket Evaluation and Regulation (VIPER) Study. https://github.com/DIDSR/viperData
- **viperObs365**: Individual observations of each reader reading each case from the FDA Validation of Imaging Premarket Evaluation and Regulation (VIPER) Study, truth is based on cancer status at 365 days. https://github.com/DIDSR/viperData
- **viperObs455**: Individual observations of each reader reading each case from the FDA Validation of Imaging Premarket Evaluation and Regulation (VIPER) Study, truth is based on cancer status at 455 days. https://github.com/DIDSR/viperData
- **MFcounts_dfClassify**: A data frame comparing mitotic figure counting performance based on whole slide images (WSI images) from four scanners to the counts from a microscope, data is per candidate mitotic figure and modality. https://github.com/DIDSR/mitoticFigureCounts/tree/master
- **MFcounts_dfCountROI**: A data frame comparing mitotic figure counting performance based on whole slide images (WSI images) from four scanners to the counts from a microscope, data is per ROI and modality. https://github.com/DIDSR/mitoticFigureCounts/tree/master
- **MFcounts_dfCountWSI**: A data frame comparing mitotic figure counting performance based on whole slide images (WSI images) from four scanners to the counts from a microscope, data is per WSI and modality. https://github.com/DIDSR/mitoticFigureCounts/tree/master
- **cardioStudyTruth**: Cardio CT data comparing display color scale effects on diagnostic performance and reader agreement, ground truth data. https://github.com/DIDSR/c
- **cardioStudyRawData**: Cardio CT data comparing display color scale effects on diagnostic performance and reader agreement, reader data. https://github.com/DIDSR/cc
- **prostateTruth**: Prostate MRI data comparing display color scale effects on diagnostic performance and reader agreement, ground truth data. https://github.com/DIDSR/colo
- **prostateRawData**: Prostate MRI data comparing display color scale effects on diagnostic performance and reader agreement, reader data. https://github.com/DIDSR/cc

Value

desired dataset downloaded from the web as a csv

Examples

```
# Save Prostate MRI ground truth and reader data
truthData <- getMRMCdataset("prostateTruth")
rawData <- getMRMCdataset("prostateRawData")
```

---

getMRMCscore   *Get a score from an MRMC data frame*

Description

Get a score from an MRMC data frame

Usage

```
getMRMCscore(df, iR, iC, modality)
```

Arguments

df	An MRMC data frame
iR	The numeric index of the readerID
iC	The numeric index of the caseID
modality	The character description of the modalityID

Value

The score

---

getWRBM   *Get within-reader, between-modality paired data from an MRMC data frame*

Description

Get within-reader, between-modality paired data from an MRMC data frame

Usage

```
getWRBM(mrmcDF, modality.X, modality.Y)
```

Arguments

mrmcDF	A data frame with the following columns: readerID, caseID, modalityID, score
modality.X	The name of one modality
modality.Y	The name of one modality. This should be different from modality.X

Value

The result of merging the modality.X and modality.Y subsets of mrmcDF by readerID and caseID

init.lecuyerRNG	*Initialize the l'Ecuyer random number generator*

Description

See the documentation for the `parallel` package. If you require backwards compatibility, please run `RNGversion("3.5.0")`.

Usage

```
init.lecuyerRNG(seed = 1, stream = 2)
```

Arguments

seed	This determines the position in each stream
stream	This determines the stream

Value

Nothing

laBRBM	*MRMC analysis of between-reader between-modality limits of agreement*

Description

The core analysis is done by ustat11 with the identity kernel (kernelFlag = 1).

Usage

```
laBRBM(
  df,
  modalitiesToCompare,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)
```

Arguments

df	an iMRMC formatted data frame, see dfMRMC_example
modalitiesToCompare	
	The factors identifying the modalities to compare.
keyColumns	Identify the factors corresponding to the readerID, caseID, modalityID, and score (or alternative random and fixed effects).

---

laWRBM   *MRMC analysis of within-reader between-modality limits of agreement*

Description

The core analysis is done by ustat11 with the difference kernel (kernelFlag = 2). This calculation can also be accomplished by ustat11 with the identity kernel (kernelFlag = 1), and the code to do that is provided after the return statement so it never gets executed.

Usage

```
laWRBM(
  df,
  modalitiesToCompare,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)
```

Arguments

df   an iMRMC formatted data frame, see dfMRMC_example

modalitiesToCompare
                    The factors identifying the modalities to compare.

keyColumns   Identify the factors corresponding to the readerID, caseID, modalityID, and
                    score (or alternative random and fixed effects).

---

renameCol   *Rename a data frame column name or a list object name*

Description

Rename a data frame column name or a list object name

Usage

```
renameCol(df, oldColName, newColName)
```

Arguments

df   A data frame

oldColName   Old column name

newColName   New column name

Value

the data frame with the updated column name

roc2binary	*Convert ROC data formatted for doIMRMC to TPF and FPF data formatted for doIMRMC*

Description

Convert ROC data formatted for doIMRMC to TPF and FPF data formatted for doIMRMC

Usage

```
roc2binary(df.auc, threshold)
```

Arguments

df.auc	data frame of roc scores formatted for doIMRMC
threshold	The threshold for determining binary decisions

Value

a list of two data frames (df.tpf and df.fpf) both formatted for doIMRMC

Examples

```
# Create a sample configuration file
config <- sim.gRoeMetz.config()
# Simulate an MRMC ROC data set
dFrame.imrmc <- sim.gRoeMetz(config)
# Convert ROC MRMC data to TPF and FPF data frames
result <- roc2binary(dFrame.imrmc, threshold = 0.9)
# Analyze TPF data using doIMRMC
tpf_result <- doIMRMC(result$df.tpf)
# View(tpf_result$perReader)
```

roeMetzConfigs	*roeMetzConfigs*

Description

This is a data frame containing the configuration parameters used in Roe1997_Acad-Radiol_v4p298. Each row corresponds to one of the twelve configurations appearing in Table 1 of that paper in a format that can be the input to sim.gRoeMetz.

Details

The columns of this data frame are as follows

- Experiment labels and size
  - modalityID.A: [character] label modality A
  - modalityID.B: [character] label modality B
  - nR: [numeric] number of readers

- – nC.neg: [numeric] number of signal-absent cases
    - – nC.pos: [numeric] number of signal-present cases
- There are six fixed effects:
    - – mu.neg: [numeric] signal-absent (neg, global mean)
    - – mu.pos: [numeric] signal-present (pos, global mean)
    - – mu.Aneg: [numeric] modality A signal-absent (Aneg, modality effect)
    - – mu.Bneg: [numeric] modality B signal-absent (Bneg, modality effect)
    - – mu.Apos: [numeric] modality A signal-present (Apos, modality effect)
    - – mu.Bpos: [numeric] modality B signal-present (Bpos, modality effect)
- There are six random effects that are independent of modality
    - – var_r.neg: [numeric] variance of random reader effect
    - – var_c.neg: [numeric] variance of random case effect
    - – var_rc.neg: [numeric] variance of random reader by case effect
    - – var_r.pos: [numeric] variance of random reader effect
    - – var_c.pos: [numeric] variance of random case effect
    - – var_rc.pos: [numeric] variance of random reader by case effect
- There are six random effects that are specific to modality A
    - – var_r.Aneg: [numeric] variance of random reader effect
    - – var_c.Aneg: [numeric] variance of random case effect
    - – var_rc.Aneg: [numeric] variance of random reader by case effect
    - – var_r.Apos: [numeric] variance of random reader effect
    - – var_c.Apos: [numeric] variance of random case effect
    - – var_rc.Apos: [numeric] variance of randome reader by case effect
- There are six random effects that are specific to modality B
    - – var_r.Bneg: [numeric] variance of random reader effect
    - – var_c.Bneg: [numeric] variance of random case effect
    - – var_rc.Bneg: [numeric] variance of random reader by case effect
    - – var_r.Bpos: [numeric] variance of random reader effect
    - – var_c.Bpos: [numeric] variance of random case effect
    - – var_rc.Bpos: [numeric] variance of randome reader by case effect

---

sim.gRoeMetz	*Simulate an MRMC data set of an ROC experiment comparing two modalities*

Description

This procedure simulates an MRMC data set of an ROC experiment comparing two modalities. It is based on Gallas2014_J-Med-Img_v1p031006, which generalizes of the model in Roe1997_Acad-Radiol_v4p298 and Roe1997_Acad-Radiol_v4p587. Specifically, it allows the variance components to depend on the truth and the modality. For the simpler Roe and Metz model, you can enter the smaller set of parameters into sim.gRoeMetz.config and it will return a larger set of parameters that can be used with this function.

Usage

```
sim.gRoeMetz(config)
```

Arguments

config   [list] of simulation parameters:

- Experiment labels and size
    - modalityID.A: [character] label modality A
    - modalityID.B: [character] label modality B
    - nR: [numeric] number of readers
    - nC.neg: [numeric] number of signal-absent cases
    - nC.pos: [numeric] number of signal-present cases
- There are six fixed effects:
    - mu.neg: [numeric] signal-absent (neg, global mean)
    - mu.pos: [numeric] signal-present (pos, global mean)
    - mu.Aneg: [numeric] modality A signal-absent (Aneg, modality effect)
    - mu.Bneg: [numeric] modality B signal-absent (Bneg, modality effect)
    - mu.Apos: [numeric] modality A signal-present (Apos, modality effect)
    - mu.Bpos: [numeric] modality B signal-present (Bpos, modality effect)
- There are six random effects that are independent of modality
    - var_r.neg: [numeric] variance of random reader effect
    - var_c.neg: [numeric] variance of random case effect
    - var_rc.neg: [numeric] variance of random reader by case effect
    - var_r.pos: [numeric] variance of random reader effect
    - var_c.pos: [numeric] variance of random case effect
    - var_rc.pos: [numeric] variance of random reader by case effect
- There are six random effects that are specific to modality A
    - var_r.Aneg: [numeric] variance of random reader effect
    - var_c.Aneg: [numeric] variance of random case effect
    - var_rc.Aneg: [numeric] variance of random reader by case effect
    - var_r.Apos: [numeric] variance of random reader effect
    - var_c.Apos: [numeric] variance of random case effect
    - var_rc.Apos: [numeric] variance of randome reader by case effect
- There are six random effects that are specific to modality B
    - var_r.Bneg: [numeric] variance of random reader effect
    - var_c.Bneg: [numeric] variance of random case effect
    - var_rc.Bneg: [numeric] variance of random reader by case effect
    - var_r.Bpos: [numeric] variance of random reader effect
    - var_c.Bpos: [numeric] variance of random case effect
    - var_rc.Bpos: [numeric] variance of randome reader by case effect

Details

The simulation is a linear model with six fixed effects related to modality and truth and 18 normally distributed independent random effects for readers, cases, and the interaction between the two. Here is the linear model:

L.mrct = mu.t + mu.mt
+ reader.rt + case.ct + readerXcase.rct
+ modalityXreader.mrt + modalityXcase.mct + modalityXreaderXcase.mrct

- m=modality (levels: A and b)
- t=truth (levels: neg and Pos)
- mu.t is the global mean for t=neg and t=pos cases
- mu.mt is the modality specific fixed effects for t=neg and t=pos cases
- the remaining terms are the random effects: all independent normal random variables

Value

dFrame.imrmc [data.frame] with (nC.neg + nC.pos)*(nR+1) rows including

- readerID: [factor] w/ nR levels "reader1", "reader2", ...
- caseID: [factor] w/ nC levels "case1", "case2", ...
- modalityID: [factor] w/ 1 level config$modalityID
- score: [numeric] reader score

Note that the first nC.neg + nC.pos rows specify the truth labels for each case. For these rows, the readerID must be "truth" and the score must be 0 for negative cases and 1 for positive cases.

---

sim.gRoeMetz.config   *Create a configuration object for the sim.gRoeMetz program*

Description

This function creates a configuration object for the Roe & Metz simulation model to be used as input for the sim.gRoeMetz program. The default model returned when there are no arguments given to the function is the "HH" model from Roe1987_Acad-Radiol_v4p298. Following that paper, The user can specify three parameters related to experiment size (nR, nC.neg, nC.pos) and five parameters parameters specifying a linear model that does not depend on modality or truth (mu.neg, mu.pos, var_r, var_c, var_rc). The mu.pos is set to 1.0, which yields a reader averaged AUC of approximately 0.765.

Usage

```
sim.gRoeMetz.config(
  nR = 5,
  nC.neg = 40,
  nC.pos = 40,
  mu.neg = 0,
  mu.pos = 1,
  var_r = 0.03,
  var_c = 0.3,
  var_rc = 0.2
)
```

Arguments

nR	Number of readers (default = 5)
nC.neg	Number of signal-absent cases (default = 25)
nC.pos	Number of signal-present cases (default = 25)
mu.neg	Mean fixed effect of signal-absent distribution (default = 0.0) Modality specific parameters are set to zero: mu.Aneg = mu.Bneg = 0
mu.pos	Mean fixed effect of signal-present distribution (default = 1.0) Modality specific parameters are set to zero: mu.Apos = mu.Bpos = 0
var_r	Variance of reader random effect (default = 0.03) var_r.neg = var_r.pos = var_r.Aneg = var_r.Apos = var_r.Bneg = var_r.Bpos = var_r
var_c	Variance of case random effect (default = 0.30) var_c.neg = var_c.pos = var_c.Aneg = var_c.Apos = var_c.Bneg = var_c.Bpos = var_c
var_rc	Variance of reader.by.case random effect (default = 0.20) var_rc.neg = var_rc.pos = var_rc.Aneg = var_rc.Apos = var_rc.Bneg = var_rc.Bpos = var_rc

Details

If no arguments, this function returns a default simulation configuration for sim.gRoeMetz

Value

config [list] Refer to the sim.gRoeMetz input variable

---

simMRMC	*Simulate an MRMC data set*

Description

This program simulates observations from one set of readers scoring one set of cases. It produces one modality and one truth state of ROC data following Roe1997_Acad-Radiol_v4p298 and Roe1997_Acad-Radiol_v4p587. In order to produce an entire ROC data set, please use sim.gRoeMetz.

Usage

```
simMRMC(simMRMC.config)
```

Arguments

simMRMC.config [list] of simulation parameters:

- modalityID [character] label modalityID
- readerIDs [factor] the ID of each reader
- caseIDs [factor] the ID of each case

- mu [numeric] mean
- var_r [numeric] variance of random reader effect
- var_c [numeric] variance of random case effect
- var_rc [numeric] variance of random reader by case effect

Details

The simulation is a linear model with one fixed effect and three normally distributed independent random effects corresponding to readers, cases, and an interaction between the two.

L.rc = mu + readerEffect.r + caseEffect.c + readerXcaseEffect.rc

Value

L [data.frame] with nC*nR rows of 4 variables

- L$modalityID [factor] determined by input modalityID

- L$readerID [factor] determined by input readerIDs

- L$caseID [factor] determined by input caseIDs

- L$score [numeric] R.r + C.c + RC.rc

  - r = 1,2,...,nR
  - c = 1,2,...,nC
  - R.r ~ N(0,var_r)
  - C.c ~ N(0,var_c)
  - RC.rc ~ N(0,var_rc)

---

simRoeMetz.example   *Simulates a sample MRMC ROC experiment*

Description

Simulates a sample MRMC ROC experiment

Usage

```
simRoeMetz.example()
```

Value

dFrame.imrmc [data.frame] Please refer to the description of the sim.gRoeMetz return variable

successDFtoROCdf	*Convert an MRMC data frame of successes to one formatted for doIM-RMC*

Description

Convert an MRMC data frame of successes to one formatted for doIMRMC

Usage

```
successDFtoROCdf(df)
```

Arguments

df   Each row contains a success observation for one reader evaluating one case

Value

data frame ready for doIMRMC

undoIMRMCdf	*Convert a doIMRMC formatted data frame to a standard data frame with all factors.*

Description

Convert a doIMRMC formatted data frame to a standard data frame with all factors.

Usage

```
undoIMRMCdf(df.MRMC)
```

Arguments

df.MRMC   This data frame includes columns for readerID, caseID, modalityID, score. Each row is a reader x case x modality observation from the study In addition to observations from the study, this data frame requires rows specifying the truth for each caseID. For truth specifications, the readerID needs to equal "truth" or "-1", modalityID can be anything ("truth" is a good choice), and score should be 0 for signal-absent normal case, 1 for signal-present disease case.

Details

Delete rows specifying truth and put the truth information on every row.

Value

output a data frame with columns readerID, caseID, modalityID, score, truth

---

uStat11 *Analysis of U-statistics degree 1,1*

Description

These two functions calculate the mean and variance of a user-specified U-statistic kernel, which is a function of cross-correlated scores.

The motivation for this analysis is data collected in imaging studies where multiple readers read multiple cases in different modes or modalities. The goal is to evaluate the variance of a reader- and case-averaged endpoint, accounting for cross-correlated data arising from two random effects: the random reader skill and the random case difficulty. This analysis is sometimes referred to as an MRMC analysis. Of course, the random effects can be from sources other than readers and cases.

Usage

```
uStat11.jointD(
  df.input,
  modalitiesToCompare,
  kernelFlag = 1,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)

uStat11.conditionalD(
  df.input,
  modalitiesToCompare,
  kernelFlag = 1,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)
```

Arguments

df.input   an iMRMC formatted data frame, see dfMRMC_example

modalitiesToCompare

                The factors identifying the modalities to compare.

kernelFlag   This determines the kernel function

- kernelFlag = 1 == identity kernel: requires two modalities: A,B.
- kernelFlag = 2 == kernel of the difference in modalities: requires four modalities: A,B,C,D.

keyColumns   Identify the factors corresponding to the readerID, caseID, modalityID, and score (or alternative random and fixed effects).

Details

uStat11.conditionalD is identical to uStat11.jointD when the study is fully-crossed: when every reader readers all the cases in both modalities. For arbitrary study designs the two functions differ according to how the components of variance are estimated.

- uStat11.conditionalD follows Gallas2007_J-Opt-Soc-Am-A_v24pB70 <doi:10.1364/JOSAA.24.000B70> and estimates the components of variance (which isolate combinations of different random effects) with nested conditional means.

- uStat11.jointD is analogous to the method in Gallas2008_Neural-Networks_v21p387 <doi:10.1016/j.neunet.2007 and estimates the components of variance (which isolate combinations of different random effects) with a joint distribution over all the observations giving equal weight to each one.

Both functions yield unbiased variance estimates. Our simulations find that uStat11.conditionalD is statistically more efficient than uStat11.jointD (its variance estimate is more precise), but it is slower.

Please refer to the tests/testthat folder of the package for examples using these functions.

Value

This function calculates the mean and variance of the indicated U-statistic kernel, which is a function of the scores. For the identity kernel, we simply return the mean and variance of the scores.

The function returns a list of outputs. Many of these outputs have three elements.

- If kernelFlag = 1 == identity kernel, the first element corresponds to the mean score of modality A, the second corresponds to mean score of modality B, and the third corresponds to the mean of the difference in scores from modality A and B.

- If kernelFlag = 2 == difference kernel, the first element corresponds to the mean difference in scores from modalities A and B, the second element corresponds to the mean difference in scores from modalities C and D, and the third elements corresponds to the difference of the just-mentioned differences.

There are 16 outputs:

- mean: See description above.
- var: The variance of the mean.
- var.1obs: The variance of one reader-case-modality observation.
- meanPerR The reader-specific means.
- nR The number of readers in the study.
- nC The number of cases in the study.
- nCperR The number of cases evaluated by each reader for each modality.
- moments The second order moments of the problem.
- coeff The coefficients corresponding to the second-order moments such that the scalar product between the moments and coefficients yields the variance.
- kernel.A A matrix showing the kernel evaluated for each combination of each reader and case for modality A (or AB).
- design.A A matrix showing the what data exists for each combination of each reader and case for modality A (or AB).
- kernel.B A matrix showing the kernel evaluated for each combination of each reader and case for modality B (or CD).
- design.B A matrix showing the what data exists for each combination of each reader and case for modality B (or CD).

Examples

```
# Create an MRMC data frame
# Refer to Gallas2014_J-Med-Img_v1p031006
simRoeMetz.config <- sim.gRoeMetz.config()

# Simulate data
df.MRMC <- sim.gRoeMetz(simRoeMetz.config)

# Reformat data
df <- undoIMRMCdf(df.MRMC)

# Grab part of the data
df <- droplevels(df[grepl("pos", df$caseID), ])

#### uStat11.jointD.identity ####
# Calculate the reader- and case-averaged difference in scores from testA and testB
# (kernelFlag = 1 specifies the U-statistics kernel to be the identity)
result.jointD.identity <- uStat11.jointD(
  df,
  kernelFlag = 1,
  keyColumns = c("readerID", "caseID", "modalityID", "score"),
  modalitiesToCompare = c("testA", "testB"))

cat("\n")
cat("uStat11.jointD.identity \n")
print(result.jointD.identity[1:2])
```

---

uStat11.diff	*Create the kernel and design matrices for uStat11*

Description

The kernel is the difference kernel

Usage

```
uStat11.diff(
  df.input,
  modalitiesToCompare,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)
```

Arguments

df.input   Data frame of observations, one per row. Columns also identify random and
                    fixed effects.

modalitiesToCompare
                    The factors identifying the modalities to compare

keyColumns   The required columns

| uStat11.identity | *Create the kernel and design matrices for uStat11* |

Description

The kernel is the identity kernel

Usage

```
uStat11.identity(
  df.input,
  modalitiesToCompare,
  keyColumns = c("readerID", "caseID", "modalityID", "score")
)
```

Arguments

df.input	Data frame of observations, one per row. Columns also identify random and fixed effects.
modalitiesToCompare	
	The factors identifying the modalities to compare
keyColumns	The required columns

Index