

Package ‘MplusAutomation’

April 3, 2012

Type Package

Title Automating Mplus Model Estimation and Interpretation

Version 0.5-1

Date 2012-04-03

Author Michael Hallquist

Maintainer Michael Hallquist <michael.hallquist@gmail.com>

Description The MplusAutomation package leverages the flexibility of the R language to automate latent variable model estimation and interpretation using Mplus, a powerful latent variable modeling program developed by Muthen and Muthen (www.statmodel.com). Specifically, MplusAutomation provides routines for creating related groups of models, running batches of models, and extracting and tabulating model parameters and fit statistics.

License LGPL-3

Depends gsubfn, xtable, tcltk, plyr, relimp

LazyLoad yes

Repository CRAN

Date/Publication 2012-04-03 21:21:29

R topics documented:

MplusAutomation-package	2
compareModels	3
createModels	5
extractModelParameters	6
extractModelSummaries	8
extractModIndices	11
getSavedata_Bparams	12

getSavedata_Data	13
getSavedata_Fileinfo	14
HTMLSummaryTable	15
LatexSummaryTable	16
lookupTech1Parameter	17
prepareMplusData	18
readModels	19
runModels	20
runModels_Interactive	21
showSummaryTable	22
testBParamCompoundConstraint	23
testBParamConstraint	25

Index	27
--------------	-----------

MplusAutomation-package

Automating Mplus Model Estimation and Interpretation

Description

The MplusAutomation package leverages the flexibility of the R language to automate latent variable model estimation and interpretation using Mplus, a powerful latent variable modeling program developed by Muthen and Muthen (www.statmodel.com). Specifically, MplusAutomation provides routines for creating related groups of models, running batches of models, and extracting and tabulating model parameters and fit statistics.

Details

Package:	MplusAutomation
Type:	Package
Version:	0.5-1
Date:	2012-04-03
License:	LGPL-3
LazyLoad:	yes

The MplusAutomation package has four primary purposes:

1. To automatically run groups/batches of models.
2. To provide routines to extract model fit statistics, parameter estimates, and raw data from Mplus output files.
3. To facilitate comparisons among models
4. To provide a template language that allows for the creation of related input files.

The core routine for running batches of models is [runModels](#), with an easy-to-use GUI wrapper, [runModels_Interactive](#).

To extract model summary statistics from one or more output files, see [extractModelSummaries](#), which returns a `data.frame` of fit statistics for models located within a directory. Model fit results can be summarized in tabular form (for comparing among models) using [showSummaryTable](#) (displays table in separate window), [HTMLSummaryTable](#) (creates HTML file containing summary table), or [LatexSummaryTable](#) (returns a LaTeX-formatted table of summary statistics).

To extract raw data created by the `SAVEDATA` command (e.g., class membership probabilities or factor scores), see [getSavedata_Data](#).

To extract unstandardized or standardized parameter estimates from a single output file, see [extractModelParameters](#).

Summaries, parameters, modification indices, and `SAVEDATA` output can be extracted simultaneously using the [readModels](#) function, and this is the recommended way to extract output with this package.

Model fit and parameter comparisons between models can be obtained using [compareModels](#).

To create a group of related models from a single template, see [createModels](#). Please read the `MplusAutomation` vignette provided along with the package (and on the CRAN website) in order to understand the template language.

In addition to the major functions above, a function for converting an R `data.frame` for use with Mplus is provided: [prepareMplusData](#). This converts the `data.frame` to a tab-delimited file and provides an Mplus syntax stub for variable names.

Author(s)

Michael Hallquist

Maintainer: Michael Hallquist <michael.hallquist@gmail.com>

References

Mplus software. Muthen and Muthen. <http://www.statmodel.com>

compareModels

Compare the output of two Mplus models

Description

The `compareModels` function compares the output of two Mplus files and prints similarities and differences in the model summary statistics and parameter estimates. Options are provided for filtering out fixed parameters and nonsignificant parameters. When requested, `compareModels` will compute the chi-square difference test for nested models (does not apply to MLMV, WLSM, and WLSMV estimators, where `DIFFTEST` in Mplus is needed).

Model outputs to be compared can be full summaries and parameters (generated by `readModels`), summary statistics only (`extractModelSummaries`), or parameters only (`extractModelParameters`).

Usage

```
compareModels(m1, m2, show="all", equalityMargin=c(param=0.0001, pvalue=0.0001),
sort="none", showFixed=FALSE, showNS=TRUE, diffTest=FALSE)
```

Arguments

m1	The first Mplus model to be compared. Generated by readModels, extractModelSummaries, or extractModelParameters.
m2	The second Mplus model to be compared.
show	What aspects of the models should be compared. Options are "all", "summaries", "equal", "diff", "pdiff", and "unique". See below for details.
equalityMargin	Defines the discrepancy between models that is considered equal. Different margins can be specified for p-value equality versus parameter equality. Defaults to .0001 for both.
sort	How to sort the output of parameter comparisons. Options are "none", "type", "alphabetical", and "maxDiff". See below for details.
showFixed	Whether to display fixed parameters in the output (identified where the est/se = 999.000, per Mplus convention. Default to FALSE.
showNS	Whether to display non-significant parameter estimates. Can be TRUE or FALSE, or a numeric value (e.g., .10) that defines what p-value is filtered as non-significant.
diffTest	Whether to compute a chi-square difference test between the models. Assumes that the models are nested. Not available for MLMV, WLSMV, and ULSMV estimators. Use DIFFTEST in Mplus instead.

Details

The show parameter can be one or more of the following, which can be passed as a vector, such as c("equal", "pdiff").

show "all" Display all available model comparison. Equivalent to c("summaries", "equal", "diff", "pdiff", "unique").

"summaries" Print a comparison of model summary statistics. Compares the following summary statistics (where available): c("Title", "Observations", "Estimator", "Parameters", "LL", "AIC", "BIC", "ChiSqM_Value", "ChiSqM_DF", "CFI", "TLI", "RMSEA", "SRMR", "WRMR")

"allsummaries" Prints a comparison of all summary statistics available in each model. May generate a lot of output.

"equal" Print parameter estimates that are equal between models (i.e., \leq equalityMargin["param"]).

"diff" Print parameter estimates that are different between models (i.e., $>$ equalityMargin["param"]).

"pdiff" Print parameter estimates where the p-values differ between models (i.e., $>$ equalityMargin["pvalue"]).

"unique" Print parameter estimates that are unique to each model.

The sort parameter determines the order in which parameter estimates are displayed. The following options are available:

sort "none" No sorting is performed, so parameters are output in the order presented in Mplus. (Default)

"type" Sort parameters by their role in the model. This groups output by regression coefficient (ON), factor loadings (BY), covariances (WITH), and so on. Within each type, output is alphabetical.

"alphabetical" Sort parameters in alphabetical order.

"maxDiff" Sort parameter output by the largest differences between models (high to low).

Value

No value is returned by this function. It is used to print model differences to the R console.

Author(s)

Michael Hallquist

createModels

Create Mplus Input Files from Template

Description

The createModels function processes a single Mplus template file and creates a group of related model input files. Definitions and examples for the template language are provided in the MplusAutomation vignette and are not duplicated here at the moment. See this PDF: <http://cran.r-project.org/web/packages/MplusAutomation/vignettes/Vignette.pdf>

Usage

```
createModels(templatefile)
```

Arguments

templatefile The filename (absolute or relative path) of an Mplus template file to be processed. Example "C:/MplusTemplate.txt"

Value

No value is returned by this function. It is solely used to process an Mplus template file.

Author(s)

Michael Hallquist

Examples

```
## Not run:  
createModels("L2 Multimodel Template No iter.txt")  
  
## End(Not run)
```

 extractModelParameters

Extract model parameters from MODEL RESULTS section.

Description

Extracts the model parameters from the MODEL RESULTS section of one or more Mplus output files. If a particular output file has more than one results section (unstandardized, stdyx, stdy, and/or std), a list will be returned. If the `target` is a directory, all `.out` files therein will be parsed and a single list will be returned, where the list elements are named by the output file name. Returned parameters often include the parameter estimate, `std. err`, `param/s.e.`, and two-tailed `p-value`.

Usage

```
extractModelParameters(target=getwd(), recursive=FALSE, filefilter, dropDimensions=FALSE, resultType)
```

Arguments

<code>target</code>	the directory containing Mplus output files (<code>.out</code>) to parse OR the single output file to be parsed. May be a full path, relative path, or a filename within the working directory. Defaults to the current working directory. Example: <code>"C:/Users/Michael/Mplus Runs"</code>
<code>recursive</code>	optional. If <code>TRUE</code> , parse all models nested in subdirectories within <code>target</code> . Defaults to <code>FALSE</code> .
<code>filefilter</code>	a Perl regular expression (PCRE-compatible) specifying particular output files to be parsed within directory. See <code>regex</code> or http://www.pcre.org/pcre.txt for details about regular expression syntax.
<code>dropDimensions</code>	Relevant only for multi-file parsing. If <code>TRUE</code> , then if only one output section (usually unstandardized) is present for all files in the parsed list, then eliminate the second-level list (which contains elements for each output section). The result is that the elements of the returned list are <code>data.frame</code> objects with the relevant parameters.
<code>resultType</code>	N.B.: this parameter is deprecated and will be removed in a future version. The new default is to extract all results that are present and return a list (see below for details). <code>resultType</code> specified the results section to extract. If <code>raw</code> , the unstandardized estimates will be returned. <code>"stdyx"</code> , <code>"stdy"</code> , and <code>"std"</code> are the other options, which extract different standardized solutions. See the Mplus User's Guide for additional details about the differences in these standardizations.

Value

If `target` is a single file, a list containing unstandardized and standardized results will be returned. If all standardized solutions are available, the list element will be named: `unstandardized`, `stdyx.standardized`, `stdy.standardized`, and `std.standardized`. If confidence intervals are output using `OUTPUT:CINTERVAL`, then a list element named `ci.unstandardized` will be included. Each of these list elements is a `data.frame` containing relevant model parameters.

If `target` is a directory, a list will be returned, where each element contains the results for a single file, and the top-level elements are named after the corresponding output file name. Each element within this list is itself a list, with elements as in the single file case above.

The core `data.frame` for each MODEL RESULTS section typically has the following structure:

<code>paramHeader</code>	The header that begins a given parameter set. Example: "FACTOR1 BY"
<code>param</code>	The particular parameter being measured (within <code>paramHeader</code>). Example: "ITEM1"
<code>est</code>	Parameter estimate value.
<code>se</code>	Standard error of the estimate
<code>est_se</code>	Quotient of <code>est/se</code> , representing z-test/t-test in large samples
<code>pval</code>	Two-tailed p-value for the <code>est_se</code> quotient.

In the case of output from Bayesian estimation (ESTIMATOR=BAYES), the `data.frame` will contain a different set of variables, including some of the above, as well as

<code>posterior_sd</code>	Posterior standard deviation of the estimate.
<code>lower_2.5ci</code>	Lower 2.5 percentile of the estimate.
<code>upper_2.5ci</code>	Upper 2.5 percentile (aka 97.5 percentile) of the estimate.

Also note that the `pval` column for Bayesian output represents a one-tailed estimate.

In the case of output from a Monte Carlo study (MONTECARLO: and MODEL POPULATION:), the `data.frame` will contain a different set of variables, including some of the above, as well as

<code>population</code>	Population parameter value.
<code>average</code>	Average parameter estimate across replications.
<code>population_sd</code>	Standard deviation of parameter value in population across replications.
<code>average_se</code>	Average standard error of estimated parameter value across replications.
<code>mse</code>	Mean squared error.
<code>cover_95</code>	Proportion of replications whose 95% confidence interval for the parameter includes the population value.
<code>pct_sig_coef</code>	Proportion of replications for which the two-tailed significance test of the parameter is significant ($p < .05$).

In the case of confidence interval output (OUTPUT:CINTERVAL), the list element `ci.unstandardized` will contain a different set of variables, including some of the above, as well as

<code>low.5</code>	Lower 0.5% CI estimate.
<code>low2.5</code>	Lower 2.5% CI estimate.
<code>low5</code>	Lower 5% CI estimate.
<code>est</code>	Parameter estimate value.
<code>up5</code>	Upper 5% (i.e., 95%) CI estimate.
<code>up2.5</code>	Upper 2.5% (i.e., 97.5%) CI estimate.
<code>up.5</code>	Upper 0.5% (i.e., 99.5%) CI estimate.

If the model contains multiple latent classes, an additional variable, LatentClass, will be included, specifying the latent class number. Also, the Categorical Latent Variables section will be included as LatentClass "Categorical.Latent.Variables."

If the model contains multiple groups, Group will be included.

If the model contains two-level output (between/within), BetweenWithin will be included.

Author(s)

Michael Hallquist

See Also

[extractModelSummaries](#)

Examples

```
## Not run:
ex3.14 <- extractModelParameters(
"C:/Program Files/Mplus/Mplus Examples/User's Guide Examples/ex3.14.out")

## End(Not run)
```

extractModelSummaries *Extract summary statistics from a single output file or from a group of Mplus models within a directory*

Description

Parses a group of Mplus model output files (.out extension) for model fit statistics. At this time, the details extracted are fixed and include: Filename, InputInstructions, Title, Estimator, LL, BIC, aBIC, AIC, AICC, Parameters, Observations, CFI, TLI, RMSEA_Estimate, RMSEA_90CI_LB, RMSEA_90CI_UB, RMSEA_pLT05, ChiSqM_Value, ChiSqM_DF, ChiSq_PValue, BLRT_KM1LL, BLRT_PValue, BLRT_Numdraws). The infrastructure is in place to allow for user-specified selection of summary statistics in future versions.

Usage

```
extractModelSummaries(target = getwd(), recursive = FALSE, filefilter)
```

Arguments

target	the directory containing Mplus output files (.out) to parse OR the single output file to be parsed. Defaults to the current working directory. Example: "C:/Users/Michael/Mplus Runs"
recursive	optional. If TRUE, parse all models nested in subdirectories within directory. Defaults to FALSE.
filefilter	a Perl regular expression (PCRE-compatible) specifying particular output files to be parsed within directory. See regex or http://www.pcre.org/pcre.txt for details about regular expression syntax.

Value

Returns a `data.frame` containing model fit statistics for all output files within directory. The `data.frame` contains some of the following variables (depends on model type):

Title	Title for the model, specified by the <code>TITLE:</code> command
Filename	Filename of the output file
Estimator	Estimator used for the model (e.g., ML, MLR, WLSMV, etc.)
LL	Log-likelihood of the model
BIC	Bayesian Information Criterion
aBIC	Sample-Size-Adjusted BIC (Sclove, 1987)
AIC	Akaike's Information Criterion
AICC	Corrected AIC, based on Sugiura (1978) and recommended by Burnham & Anderson (2002)
DIC	Deviance Information Criterion. Available in <code>ESTIMATOR=BAYES</code> output.
Parameters	Number of parameters estimated by the model
pD	Estimated number of parameters in Bayesian output
Observations	The number of observations for the model (does not support multiple-groups analysis at this time)
CFI	Confirmatory Fit Index
TLI	Tucker-Lewis Index
RMSEA_Estimate	Point estimate of root mean squared error of approximation
RMSEA_90CI_LB	Lower bound of the 90% Confidence Interval around the RMSEA estimate.
RMSEA_90CI_UB	Upper bound of the 90% Confidence Interval around the RMSEA estimate.
RMSEA_pLT05	Probability that the RMSEA estimate falls below .05, indicating good fit.
ChiSqM_Value	Model chi-squared value
ChiSqM_DF	Model chi-squared degrees of freedom
ChiSqM_PValue	Model chi-squared p value
ObsRepChiSqDiff_95CI_LB	Lower bound of 95% confidence interval for the difference between observed and replicated chi-square values
ObsRepChiSqDiff_95CI_UB	Upper bound of 95% confidence interval for the difference between observed and replicated chi-square values
PostPred_PValue	Posterior predictive p-value
BLRT_KM1LL	Log-likelihood of the K-1 model (one less class) for the Bootstrapped Likelihood Ratio Test (TECH14).
BLRT_PValue	P-value of the Bootstrapped Likelihood Ratio Test (TECH14) testing whether the K class model is significantly better than K-1

BLRT_Numdraws	The number of bootstrapped samples used in the Bootstrapped Likelihood Ratio Test
SRMR	Standardized root mean square residual
SRMR.Between	For TYPE=TWOLEVEL output, standardized root mean square residual for between level
SRMR.Within	For TYPE=TWOLEVEL output, standardized root mean square residual for within level
WRMR	Weighted root mean square residual
ChiSqBaseline_Value	Baseline (unstructured) chi-squared value
ChiSqBaseline_DF	Baseline (unstructured) chi-squared degrees of freedom
ChiSqBaseline_PValue	Baseline (unstructured) chi-squared p value
NumFactors	For TYPE=EFA output, the number of factors
T11_KM1Starts	TECH11: Number of initial stage random starts for k-1 model
T11_KM1Final	TECH11: Number of final stage optimizations for k-1 model
T11_KM1LL	TECH11: Log-likelihood of the K-1 model used for the Vuong-Lo-Mendell-Rubin LRT
T11_VLMR_2xLLDiff	TECH11: 2 * Log-likelihood Difference of K-class vs. K-1-class model for the Vuong-Lo-Mendell-Rubin LRT
T11_VLMR_ParamDiff	TECH11: Difference in number of parameters between K-class and K-1-class model for the Vuong-Lo-Mendell-Rubin LRT
T11_VLMR_Mean	TECH11: Vuong-Lo-Mendell-Rubin LRT mean
T11_VLMR_SD	TECH11: Vuong-Lo-Mendell-Rubin LRT standard deviation
T11_VLMR_PValue	TECH11: Vuong-Lo-Mendell-Rubin LRT p-value
T11_LMR_Value	TECH11: Lo-Mendell-Rubin Adjusted LRT value
T11_LMR_PValue	TECH11: Lo-Mendell-Rubin Adjusted LRT p-value

Author(s)

Michael Hallquist

See Also

[regex](#), [runModels](#), [readModels](#)

Examples

```
## Not run:
allExamples <- extractModelSummaries(
"C:/Program Files/Mplus/Mplus Examples/User's Guide Examples")
## End(Not run)
```

extractModIndices	<i>Extract model modification indices.</i>
-------------------	--

Description

Extracts the model modification indices from the MODEL MODIFICATION INDICES section of one or more Mplus output files. If the `target` is a directory, all `.out` files therein will be parsed and a single list will be returned, where the list elements are named by the output file name. Returned parameters typically include the pairwise relationships between variables to be freed, the change in model chi-square (M.I.), and the expected parameter change (E.P.C.).

Usage

```
extractModIndices(target=getwd(), recursive=FALSE, filefilter)
```

Arguments

<code>target</code>	the directory containing Mplus output files (<code>.out</code>) to parse OR the single output file to be parsed. May be a full path, relative path, or a filename within the working directory. Defaults to the current working directory. Example: "C:/Users/Michael/Mplus Runs"
<code>recursive</code>	optional. If TRUE, parse all models nested in subdirectories within <code>target</code> . Defaults to FALSE.
<code>filefilter</code>	a Perl regular expression (PCRE-compatible) specifying particular output files to be parsed within directory. See regex or http://www.pcre.org/pcre.txt for details about regular expression syntax.

Value

If `target` is a single file, a `data.frame` containing modification results for the target output file will be returned. If `target` is a directory, a list will be returned, where each element contains a `data.frame` of the modification indices for a single file, and the top-level elements are named after the corresponding output file name.

The basic `data.frame` containing the MODEL MODIFICATION INDICES section of `outfile`. Variables include

<code>modV1</code>	The first variable in the pair to be freed according to the M.I.
<code>operator</code>	The suggested relationship between <code>modV1</code> and <code>modV2</code> (e.g., WITH for freeing the covariance between <code>modV1</code> and <code>modV2</code>)
<code>modV2</code>	The first variable in the pair to be freed according to the M.I.
<code>MI</code>	The decrease in model chi-square if the specified relationship is freely estimated
<code>EPC</code>	The expected parameter estimate between <code>modV1</code> and <code>modV2</code> if freed.
<code>Std_EPC</code>	The EPC value standardized using the variances of the continuous latent variables.
<code>StdYX_EPC</code>	The EPC value standardized using the variances of the continuous latent variables as well as the variances of the background and/or outcome variables.

Author(s)

Michael Hallquist

See Also

[readModels](#), [extractModelSummaries](#), [extractModelParameters](#)

Examples

```
## Not run:  
ex3.14 <- extractModIndices(  
  "C:/Program Files/Mplus/Mplus Examples/User's Guide Examples/ex3.14.out")  
## End(Not run)
```

getSavedata_Bparams *Load the draws from the Bayesian model posterior distribution (SAVEDATA BPARAMETERS) command into an R data.frame*

Description

This function reads a the BPARAMETERS output file from the Mplus SAVEDATA BPARAMETERS command and returns an R data.frame object.

Usage

```
getSavedata_Bparams(outfile)
```

Arguments

`outfile` required. The name of the Mplus output file to read. Can be an absolute or relative path. If `outfile` is a relative path or just the filename, then it is assumed that the file resides in the working directory `getwd()`.

Value

a data.frame containing the draws from the posterior distribution for a Bayesian model that uses the SAVEDATA BPARAMETERS command.

Note

Note that the `outfile` parameter should refer to the Mplus output file (.out extension), not the actual dataset generated by SAVEDATA. This function reads information about the dataset from the .out file and loads the dataset accordingly.

Author(s)

Michael Hallquist

See Also

[getSavedata_Fileinfo](#), [getSavedata_Data](#)

getSavedata_Data	<i>Load an analysis dataset from the SAVEDATA command into an R data.frame</i>
------------------	--

Description

This function reads an analysis dataset generated by the Mplus SAVEDATA command and returns an R data.frame object.

Usage

```
getSavedata_Data(outfile)
```

Arguments

outfile	required. The name of the Mplus output file to read. Can be an absolute or relative path. If outfile is a relative path or just the filename, then it is assumed that the file resides in the working directory getwd().
---------	--

Value

a data.frame containing the analysis dataset generated by the SAVEDATA command.

Note

Note that the outfile parameter should refer to the Mplus output file (.out extension), not the actual dataset generated by SAVEDATA. This function reads information about the dataset from the .out file and loads the dataset accordingly.

Author(s)

Michael Hallquist

See Also

[getSavedata_Fileinfo](#)

Examples

```
## Not run:  
savedat <- getSavedata_Data("C:/Program Files/Mplus/Test Output.out")  
  
## End(Not run)
```

getSavedata_Fileinfo *Read Variable Names, Formats, and Widths from data generated by the SAVEDATA Command*

Description

This function reads the SAVEDATA INFORMATION section from an Mplus output file that used the SAVEDATA command, and it returns a list with the filename, variable names, variable formats, and variable widths of the SAVEDATA file. If present, the function also parses information about the Bayesian Parameters (BPARAMETERS) file.

Usage

```
getSavedata_Fileinfo(outfile)
```

Arguments

`outfile` required. The name of the Mplus output file to read. Can be an absolute or relative path. If `outfile` is a relative path or just the filename, then it is assumed that the file resides in the working directory `getwd()`.

Value

Returns a list of SAVEDATA file information that includes:

<code>fileName</code>	The name of the file containing the analysis dataset created by the Mplus SAVE-DATA command.
<code>fileVarNames</code>	A character vector containing the names of variables in the dataset.
<code>fileVarFormats</code>	A character vector containing the Fortran-style formats of variables in the dataset.
<code>fileVarWidths</code>	A numeric vector containing the widths of variables in the dataset (which is stored in fixed-width format).
<code>bayesFile</code>	The name of the BPARAMETERS file containing draws from the posterior distribution created by the Mplus SAVEDATA BPARAMETERS command.
<code>bayesVarNames</code>	A character vector containing the names of variables in the BPARAMETERS dataset.

Author(s)

Michael Hallquist

See Also

[getSavedata_Data](#)

Examples

```
## Not run:
fileInfo <- getSavedata_Fileinfo("C:/Program Files/Mplus/Test Output.out")

## End(Not run)
```

HTMLSummaryTable	<i>Create an HTML file containing a summary table of Mplus model statistics</i>
------------------	---

Description

Creates an HTML file containing a summary table of model fit statistics extracted using the `extractModelSummaries` function. By default, the following summary statistics are included: Title, LL, Parameters, AIC, AICC, BIC, RMSEA_Estimate, but these are customizable using the `keepCols` and `dropCols` parameters.

Usage

```
HTMLSummaryTable(modellist, filename=file.path(getwd(), "Model Comparison.html"),
  keepCols, dropCols, sortBy, display=FALSE)
```

Arguments

<code>modellist</code>	A list of models (as a <code>data.frame</code>) returned from the <code>extractModelSummaries</code> function.
<code>filename</code>	The name of the HTML file to be created. Can be an absolute or relative path. If <code>filename</code> is a relative path or just the filename, then it is assumed that the file resides in the working directory <code>getwd()</code> . Example: "Mplus_Summary.html"
<code>keepCols</code>	A vector of character strings indicating which columns/variables to display in the summary. Only columns included in this list will be displayed (all others excluded). By default, <code>keepCols</code> is: <code>c("Title", "LL", "Parameters", "AIC", "AICC", "BIC", "RMSEA_Estimate")</code> . Example: <code>c("Title", "LL", "AIC", "CFI")</code>
<code>dropCols</code>	A vector of character strings indicating which columns/variables to omit from the summary. Any column not included in this list will be displayed. By default, <code>dropCols</code> is <code>NULL</code> . Example: <code>c("InputInstructions", "TLI")</code>
<code>sortBy</code>	optional. Field name (as character string) by which to sort the table. Typically an information criterion (e.g., "AIC" or "BIC") is used to sort the table. Defaults to "AICC".
<code>display</code>	optional. This parameter specifies whether to display the table in a web browser upon creation (TRUE or FALSE).

Value

No value is returned by this function. It is solely used to create an HTML file containing summary statistics.

Note

You must choose between `keepCols` and `dropCols` because it's not sensible to use these together to include and exclude columns. The function will error if you include both parameters.

Author(s)

Michael Hallquist

See Also

[extractModelSummaries](#), [showSummaryTable](#), [LatexSummaryTable](#)

LatexSummaryTable	<i>Display summary table of Mplus model statistics in separate window</i>
-------------------	---

Description

Creates a LaTeX-formatted summary table of model fit statistics extracted using the `extractModelSummaries` function. The table syntax is returned by the function, which is useful for embedding LaTeX tables using Sweave. By default, the following summary statistics are included: Title, LL, Parameters, AIC, AICC, BIC, RMSEA_Estimate, but these are customizable using the `keepCols` and `dropCols` parameters.

Usage

```
LatexSummaryTable(modellist, keepCols, dropCols, sortBy, label=NULL, caption=NULL)
```

Arguments

<code>modellist</code>	A list of models (as a <code>data.frame</code>) returned from the <code>extractModelSummaries</code> function.
<code>keepCols</code>	A vector of character strings indicating which columns/variables to display in the summary. Only columns included in this list will be displayed (all others excluded). By default, <code>keepCols</code> is <code>c("Title", "LL", "Parameters", "AIC", "AICC", "BIC", "RMSEA_Estimate")</code> . Example: <code>c("Title", "LL", "AIC", "CFI")</code>
<code>dropCols</code>	A vector of character strings indicating which columns/variables to omit from the summary. Any column not included in this list will be displayed. By default, <code>dropCols</code> is <code>NULL</code> . Example: <code>c("InputInstructions", "TLI")</code>
<code>sortBy</code>	optional. Field name (as character string) by which to sort the table. Typically an information criterion (e.g., "AIC" or "BIC") is used to sort the table. Defaults to "AICC"
<code>label</code>	optional. A character string specifying the label for the LaTeX table, which can be used for referencing the table.
<code>caption</code>	optional. A character string specifying the caption for the LaTeX table.

Value

A LaTeX-formatted table summarizing the modelList is returned (created by xtable).

Note

You must choose between keepCols and dropCols because it's not sensible to use these together to include and exclude columns. The function will error if you include both parameters.

Author(s)

Michael Hallquist

See Also

[extractModelSummaries](#), [HTMLSummaryTable](#), [showSummaryTable](#), [Sweave](#)

lookupTech1Parameter *Lookup the matrix element for a give parameter number*

Description

The lookupTech1Parameter function identifies the position in the Mplus model matrices corresponding to a given parameter defined in the TECHNICAL 1 PARAMETER SPECIFICATION OUTPUT. The goal of this function is to aid in identifying problematic parameters often printed in the warnings and errors section of Mplus output.

Usage

```
lookupTech1Parameter(tech1Output, paramNumber)
```

Arguments

tech1Output	The object corresponding to the TECH1 parameter specification from readModels.
paramNumber	The parameter number to lookup

Value

A data.frame containing the row(s) and column(s) of TECH1 parameter specification matching the requested paramNumber.

Author(s)

Michael Hallquist

See Also

[readModels](#)

Examples

```
## Not run: models <- readModels("test1.out")
param <- lookupTech1Parameter(models$tech1, 16)

## End(Not run)
```

prepareMplusData	<i>Create tab-delimited file and Mplus input syntax from R data.frame</i>
------------------	---

Description

The prepareMplusData function converts an R data.frame object into a tab-delimited file (without header) to be used in an Mplus input file. The corresponding Mplus syntax, including the data file definition and variable names, is printed to the console.

Usage

```
prepareMplusData(df, filename, keepCols, dropCols)
```

Arguments

df	The R data.frame to be prepared for Mplus
filename	The path and filename for the tab-delimited data file for use with Mplus. Example: "C:/Mplusdata/data1.dat"
keepCols	A character vector specifying the variable names within df to be output to filename.
dropCols	A character vector specifying the variable names within df to be omitted from the data output to filename.

Value

No value is returned by this function. It is solely used to convert an R data.frame to an Mplus data file.

Author(s)

Michael Hallquist

Examples

```
## Not run:
library(foreign)

study5 <- read.spss("reanalysis-study-5-mt-fall-08.sav", to.data.frame=TRUE)
ASData5 <- subset(study5, select=c("ppnum", paste("as", 1:33, sep="")))

prepareMplusData(ASData5, "study5.dat")

## End(Not run)
```

readModels	<i>Read Parameters, Summary Statistics, and Savedata from Mplus Output</i>
------------	--

Description

Extracts the model parameters, summary statistics, and savedata from the one or more Mplus output files. This function is essentially a wrapper around `extractModelParameters`, `extractModelSummaries`, and `getSavedata_Data`, respectively. The goal is to have a single function that parses all (supported) aspects of Mplus output and to combine these into a list object, with one element per output file identified.

Usage

```
readModels(target=getwd(), recursive=FALSE, filefilter)
```

Arguments

target	the directory containing Mplus output files (.out) to parse OR the single output file to be parsed. May be a full path, relative path, or a filename within the working directory. Defaults to the current working directory. Example: "C:/Users/Michael/Mplus Runs"
recursive	optional. If TRUE, parse all models nested in subdirectories within target. Defaults to FALSE.
filefilter	a Perl regular expression (PCRE-compatible) specifying particular output files to be parsed within directory. See <code>regex</code> or http://www.pcre.org/pcre.txt for details about regular expression syntax.

Value

A list with one element per file. Each element is composed of up to three subelements: summaries, parameters, and savedata. If target is a single file, then the top-level elements will be the summaries, parameters, and savedata, not a list of files.

summaries	Summary statistics from <code>extractModelSummaries</code> , having structure as specified by that function
parameters	Model parameters from <code>extractModelParameters</code> , having structure as specified by that function
mod_indices	Model modification indices from <code>extractModIndices</code> , having structure as specified by that function
savedata_info	File information about SAVEDATA files related to this output
savedata	SAVEDATA file as an R <code>data.frame</code> , as described in <code>getSavedata_Data</code>
bparameters	a <code>data.frame</code> containing the draws from the posterior distribution for a Bayesian model that uses the SAVEDATA BPARAMETERS command
residuals	a list containing relevant information from OUTPUT: RESIDUALS

tech1	a list containing parameter specification and starting values from OUTPUT: TECH1
tech4	a list containing means, covariances, and correlations for latent variables from OUTPUT: TECH4

Author(s)

Michael Hallquist

See Also

[extractModelSummaries](#), [extractModelParameters](#), [extractModIndices](#), [getSavedata_Fileinfo](#), [getSavedata_Data](#), [getSavedata_Bparams](#)

Examples

```
## Not run:
allOutput <- readModels(
"C:/Program Files/Mplus/Mplus Examples/User's Guide Examples", recursive=TRUE)

## End(Not run)
```

runModels	<i>Run Mplus Models</i>
-----------	-------------------------

Description

This function runs a group of Mplus models (.inp files) located within a single directory or nested within subdirectories.

Usage

```
runModels(directory = getwd(), recursive = FALSE, showOutput = FALSE,
replaceOutfile = "always", logFile = "Mplus Run Models.log", Mplus_command="Mplus")
```

Arguments

directory	the directory containing Mplus input files (.inp) to run. Defaults to the current working directory. Example: "C:/Users/Michael/Mplus Runs"
recursive	optional. If TRUE, run all models nested in subdirectories within directory. Defaults to FALSE.
showOutput	optional. If TRUE, show estimation output (TECH8) in the R console. Note that if run within Rgui, output will display within R, but if run via Rterm, a separate window will appear during estimation.

- `replaceOutfile` optional. Currently supports three settings: "always", which runs all models, regardless of whether an output file for the model exists; "never", which does not run any model that has an existing output file; and "modifiedDate", which only runs a model if the modified date for the input file is more recent than the output file modified date (implying there have been updates to the model).
- `logFile` optional. If non-null, specifies a file (and optionally, directory) that records the settings passed into the function and the models run (or skipped) during the run.
- `Mplus_command` optional. N.B.: No need to pass this parameter for most users (has intelligent defaults). Allows the user to specify the name/path of the Mplus executable to be used for running models. This covers situations where Mplus is not in the system's path, or where one wants to test different versions of the Mplus program.

Value

None. Function is used for its side effects (running models).

Author(s)

Michael Hallquist

See Also

[runModels_Interactive](#)

Examples

```
## Not run:
runModels("C:/Users/Michael/Mplus Runs", recursive=TRUE, showOutput=TRUE,
replaceOutfile="modifiedDate", logFile="MH_RunLog.txt",
Mplus_command="C:\Users\Michael\Mplus Install\Mplus51.exe")
## End(Not run)
```

`runModels_Interactive` *Run Mplus Models Using Graphical Interface*

Description

This function provides a graphical user interface to the `runModels` function. It uses Tcl/Tk to display a window in which the user can specify parameters for `runModels`, including the directory for runs, recursing through subdirectories, displaying output on the console, and replacing existing outfiles.

Usage

```
runModels_Interactive(directory = getwd(), recursive = "0",
showOutput = "1", replaceOutfile = "1", checkDate = "0",
logFile="1")
```

Arguments

directory	optional. The starting directory that will display in the dialog window. Defaults to the current working directory.
recursive	optional. Whether the recursive checkbox should be checked when the window opens. "0" for FALSE, "1" for TRUE.
showOutput	optional. Whether the show output checkbox should be checked when the window opens. "0" for FALSE, "1" for TRUE.
replaceOutfile	optional. Whether the replace outfile checkbox should be checked when the window opens. "0" for FALSE, "1" for TRUE.
checkDate	optional. Whether the check modified date checkbox should be checked when the window opens. "0" for FALSE, "1" for TRUE.
logFile	optional. Whether the log file checkbox should be checked when the window opens. "0" for FALSE, "1" for TRUE.

Details

This function exists as a GUI wrapper for `runModels` and does not provide any distinct functionality.

Value

None. Function is used to display user interface for running models.

Author(s)

Michael Hallquist

See Also

[runModels](#)

showSummaryTable	<i>Display summary table of Mplus model statistics in separate window</i>
------------------	---

Description

Displays a summary table of model fit statistics extracted using the `extractModelSummaries` function. This function relies on the `showData` function from the `relimp` package, which displays data in a Tk-based window. By default, the following summary statistics are included: Title, LL, Parameters, AIC, AICC, BIC, RMSEA_Estimate, but these are customizable using the `keepCols` and `dropCols` parameters.

Usage

```
showSummaryTable(modelList, keepCols, dropCols, sortBy, font="Courier 9")
```

Arguments

modellist	A list of models (as a <code>data.frame</code>) returned from the <code>extractModelSummaries</code> function.
keepCols	A vector of character strings indicating which columns/variables to display in the summary. Only columns included in this list will be displayed (all others excluded). By default, <code>keepCols</code> is: <code>c("Title", "LL", "Parameters", "AIC", "AICC", "BIC", "RMSEA_Estimate")</code> . Example: <code>c("Title", "LL", "AIC", "CFI")</code>
dropCols	A vector of character strings indicating which columns/variables to omit from the summary. Any column not included in this list will be displayed. By default, <code>dropCols</code> is <code>NULL</code> . Example: <code>c("InputInstructions", "TLI")</code>
sortBy	optional. Field name (as character string) by which to sort the table. Typically an information criterion (e.g., "AIC" or "BIC") is used to sort the table. Defaults to "AICC"
font	optional. The font to be used to display the summary table. Defaults to Courier 9

Value

No value is returned by this function. It is solely used to display the summary table in a separate window.

Note

You must choose between `keepCols` and `dropCols` because it's not sensible to use these together to include and exclude columns. The function will error if you include both parameters.

Author(s)

Michael Hallquist

See Also

[extractModelSummaries](#)

[HTMLSummaryTable](#)

[LatexSummaryTable](#)

testBParamCompoundConstraint

Test inequality-constrained hypothesis for two or more parameters based on iterations of MCMC chains

Description

Tests an inequality-constrained hypothesis (van de Schoot, Hoijtink, Hallquist, & Boelen, submitted) based on draws from the posterior distribution of the model parameters, which provides information about the proportion of the distribution that is in agreement with a given hypothesis. This function is used for more complex hypotheses about three or more parameters, whereas testBParamConstraint tests a simple two-parameter hypothesis.

This function accepts a data.frame containing iterations of the MCMC chains (rows) for each model parameter (columns) and prints out the number and proportion of draws that are consistent with the requested hypothesis test.

The test argument is evaluated directly as R code, with the bparams data.frame attached so that variable names are available directly in the environment. Because the goal is to evaluate the test for each draw from the posterior distribution, remember to use vector-based logic operators, not boolean operators. That is, stick to & or | for joining tests of parameters, rather than && or || since the latter will return a single TRUE/FALSE, which is irrelevant.

An example test in R logic would be "(STAITOT.ON.CG > STAITOT.ON.UG) & (BDIM.ON.CG > BDIM.ON.UG)".

Usage

```
testBParamCompoundConstraint(bparams, test)
```

Arguments

bparams	A data.frame containing draws from the posterior distribution. Obtained by SAVEDATA:BPARAMETERS in Mplus and getSavedata_Bparams or readModels in MplusAutomation.
test	The R code defining the parameter test of three or more parameters. Example: "(STAITOT.ON.CG > STAITOT.ON.UG) & (BDIM.ON.CG > BDIM.ON.UG)".

Value

No value is returned by this function. Instead, two summary tables are printed to the screen containing the number and proportion of draws consistent with the hypothesis.

Author(s)

Michael Hallquist

See Also

[testBParamConstraint](#)

Examples

```
## Not run:
btest <- getSavedata_Bparams("model vb1_simple_b.out")

testBParametersCompoundConstraint(btest,
```

```
"(STDYX_STAITOT.ON.CG > STDYX_STAITOT.ON.UCG) & (STDYX_BDIM.ON.CG > STDYX_BDIM.ON.UCG)")
## End(Not run)
```

testBParamConstraint *Test inequality-constrained hypothesis for two parameters based on iterations of MCMC chains*

Description

Tests a simple inequality-constrained hypothesis (van de Schoot, Hoijtink, Hallquist, & Boelen, submitted) based on draws from the posterior distribution of the model parameters, which provides information about the proportion of the distribution that is in agreement with a given hypothesis. This function is used for simple hypothesis for two parameters, whereas testBParamCompoundConstraint gives full access to multiple parameters and R's logic syntax.

This function accepts a data.frame containing iterations of the MCMC chains (rows) for each model parameter (columns) and prints out the number and proportion of draws that are consistent with the requested hypothesis test.

The coef1, operator, and coef2 arguments are appended in sequence, so that the hypothesis test is constructed from left-to-right. e.g., testBParamConstraint(bparamsDF, "MGM.TRT1", ">", "MGM.EX2").

Usage

```
testBParamConstraint(bparams, coef1, operator, coef2)
```

Arguments

bparams	A data.frame containing draws from the posterior distribution. Obtained by SAVEDATA:BPARAMETERS in Mplus and getSavedata_Bparams or readModels in MplusAutomation.
coef1	The name of the first parameter to be compared. Example: "MGM.TRT1"
operator	A logical operator to compare the two parameters. Should be one of >=, >, <, or <=. Example: ">="
coef2	The name of the first parameter to be compared. Example: "MGM.EX2"

Value

No value is returned by this function. Instead, two summary tables are printed to the screen containing the number and proportion of draws consistent with the hypothesis.

Author(s)

Michael Hallquist

See Also[testBParamCompoundConstraint](#)**Examples**

```
## Not run:  
btest <- getSavedata_Bparams("model vb1_simpel_b.out")  
  
testBParametersConstraint(btest, "STDYX_STAITOT.ON.CG", ">", "STDYX_STAITOT.ON.UCG")  
  
## End(Not run)
```

Index

*Topic **interface**

- compareModels, [3](#)
- createModels, [5](#)
- extractModelParameters, [6](#)
- extractModelSummaries, [8](#)
- extractModIndices, [11](#)
- getSavedata_Bparams, [12](#)
- getSavedata_Data, [13](#)
- getSavedata_Fileinfo, [14](#)
- HTMLSummaryTable, [15](#)
- LatexSummaryTable, [16](#)
- lookupTech1Parameter, [17](#)
- prepareMplusData, [18](#)
- readModels, [19](#)
- runModels, [20](#)
- runModels_Interactive, [21](#)
- showSummaryTable, [22](#)
- testBParamCompoundConstraint, [23](#)
- testBParamConstraint, [25](#)

*Topic **package**

- MplusAutomation-package, [2](#)

compareModels, [3](#), [3](#)

createModels, [3](#), [5](#)

extractModelParameters, [3](#), [6](#), [12](#), [20](#)

extractModelSummaries, [3](#), [8](#), [8](#), [12](#), [16](#), [17](#),
[20](#), [23](#)

extractModIndices, [11](#), [20](#)

getSavedata_Bparams, [12](#), [20](#), [24](#), [25](#)

getSavedata_Data, [3](#), [13](#), [13](#), [14](#), [20](#)

getSavedata_Fileinfo, [13](#), [14](#), [20](#)

HTMLSummaryTable, [3](#), [15](#), [17](#), [23](#)

LatexSummaryTable, [3](#), [16](#), [16](#), [23](#)

lookupTech1Parameter, [17](#)

MplusAutomation

(MplusAutomation-package), [2](#)

MplusAutomation-package, [2](#)

prepareMplusData, [3](#), [18](#)

readModels, [3](#), [10](#), [12](#), [17](#), [19](#), [24](#), [25](#)

regex, [10](#)

runModels, [2](#), [10](#), [20](#), [22](#)

runModels_Interactive, [2](#), [21](#), [21](#)

showSummaryTable, [3](#), [16](#), [17](#), [22](#)

Sweave, [17](#)

testBParamCompoundConstraint, [23](#), [26](#)

testBParamConstraint, [24](#), [25](#)