

# Package ‘metaSEM’

October 31, 2015

**Type** Package

**Title** Meta-Analysis using Structural Equation Modeling

**Version** 0.9.6

**Date** 2015-10-31

**Depends** R (>= 3.2.0), OpenMx (>= 2.3.1)

**Imports** Matrix, MASS, ellipse, graphics, stats, utils

**Suggests** metafor, semPlot

**Author** Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Maintainer** Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Description** A collection of functions for conducting meta-analysis using a structural equation modeling (SEM) approach via the 'OpenMx' package. It also implements the two-stage SEM approach to conduct meta-analytic structural equation modeling on correlation and covariance matrices.

**License** GPL (>= 2)

**LazyLoad** yes

**LazyData** yes

**ByteCompile** yes

**URL** <https://github.com/mikewlcheung/metasem>

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-10-31 19:23:05

## R topics documented:

metaSEM-package . . . . .	3
Aloe14 . . . . .	4
anova . . . . .	5
as.mxMatrix . . . . .	7
asyCov . . . . .	8
BCG . . . . .	10

bdiagMat	11
bdiagRep	12
Becker09	13
Becker83	16
Becker92	17
Becker94	19
Berkey98	21
Bornmann07	22
Cheung00	24
Cheung09	26
coef	27
Cooper03	29
create.Fmatrix	30
create.mxMatrix	31
Diag	33
Digman97	34
HedgesOlkin85	36
homoStat	37
Hox02	38
Hunter83	40
indirectEffect	41
is.pd	43
issp05	44
issp89	46
Jaramillo05	48
list2matrix	49
Mak09	50
matrix2bdiag	51
meta	52
meta2semPlot	55
meta3	56
Norton13	58
pattern.n	59
pattern.na	60
plot	61
print	64
readData	65
reml	67
reml3	69
rerun	71
Roorda11	72
summary	74
tssem1	75
vcov	77
vec2symMat	79
wls	80
wvs94a	84
wvs94b	86

---

metaSEM-package	<i>Meta-Analysis using Structural Equation Modeling</i>
-----------------	---

---

### Description

A collection of functions for conducting meta-analysis using a structural equation modeling (SEM) approach via the 'OpenMx' package. It also implements the two-stage SEM approach to conduct meta-analytic structural equation modeling on correlation and covariance matrices.

### Details

Package:	metaSEM
Type:	Package
Version:	0.9.6
Date:	2015-10-31
License:	GPL (>=2)
LazyLoad:	yes

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

Maintainer: Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### References

- Cheung, M. W.-L. (2008). A model for integrating fixed-, random-, and mixed-effects meta-analyses into structural equation modeling. *Psychological Methods*, **13**, 182-202.
- Cheung, M. W.-L. (2009). Constructing approximate confidence intervals for parameters with structural equation models. *Structural Equation Modeling*, **16**, 267-294.
- Cheung, M. W.-L. (2010). Fixed-effects meta-analyses as multiple-group structural equation models. *Structural Equation Modeling*, **17**, 481-509.
- Cheung, M. W.-L. (2013). Implementing restricted maximum likelihood estimation in structural equation models. *Structural Equation Modeling*, **20**, 157-167.
- Cheung, M. W.-L. (2013). Multivariate meta-analysis as structural equation models. *Structural Equation Modeling*, **20**, 429-454.
- Cheung, M. W.-L. (2014). Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods*, **19**, 211-229.
- Cheung, M. W.-L. (2014). Fixed- and random-effects meta-analytic structural equation modeling: Examples and analyses in R. *Behavior Research Methods*, **46**, 29-40.

- Cheung, M. W.-L. (2015). metaSEM: An R package for meta-analysis using structural equation modeling. *Frontiers in Psychology*, **5**, 1521.
- Cheung, M. W.-L. (2015). *Meta-Analysis: A Structural Equation Modeling Approach*. Chichester, West Sussex: John Wiley & Sons, Inc.
- Cheung, M. W.-L., & Chan, W. (2004). Testing dependent correlation coefficients via structural equation modeling. *Organizational Research Methods*, **7**, 206-223.
- Cheung, M. W.-L., & Chan, W. (2005). Meta-analytic structural equation modeling: A two-stage approach. *Psychological Methods*, **10**, 40-64.
- Cheung, M. W.-L., & Chan, W. (2009). A two-stage approach to synthesizing covariance matrices in meta-analytic structural equation modeling. *Structural Equation Modeling*, **16**, 28-53.
- Cheung, M. W.-L., & Cheung, S.-F. (in press). Random-effects models for meta-analytic structural equation modeling: Review, issues, and illustrations. *Research Synthesis Methods*.

---

Aloe14

---

*Multivariate effect sizes between classroom management self-efficacy (CMSE) and other variables reported by Aloe et al. (2014)*


---

### Description

This study reports sixteen studies on the effect sizes (correlation coefficients) between CMSE and emotional exhaustion (EE), depersonalization (DP), and (lowered) personal accomplishment (PA) reported by Aloe et al. (2014).

### Usage

```
data("Aloe14")
```

### Format

A data frame with 16 observations on the following 14 variables.

Study a factor with levels Betoret Brouwers & Tomic Bumen Chang Durr Evers et al.  
 Friedman Gold Huk Kress Kumarakulasingam Martin et al. Ozdemir Skaalvik and Skaalvik  
 Williams

Year Year of publication

EE Emotional exhaustion

DP Depersonalization

PA (Lowered) personal accomplishment

V\_EE Sampling variance of emotional exhaustion

V\_DP Sampling variance of depersonalization

V\_PA Sampling variance of (lowered) personal accomplishment

C\_EE\_DP Sampling covariance between EE and DP

C\_EE\_PA Sampling covariance between EE and PA

C\_DP\_PA Sampling covariance between DP and PA  
 Publication\_type Either Dissertation or Journal  
 Percentage\_females Percentage of females in the study  
 Years\_experience Average years of experience

### Source

Aloe, A. M., Amo, L. C., & Shanahan, M. E. (2014). Classroom management self-efficacy and burnout: A multivariate meta-analysis. *Educational Psychology Review*, **26(1)**, 101-126. doi:10.1007/s10648-013-9244-0

### Examples

```
## Not run:
data(Aloe14)

## Random-effects meta-analysis
meta1 <- meta(cbind(EE,DP,PA), cbind(V_EE, C_EE_DP, C_EE_PA, V_DP, C_DP_PA, V_PA),
             data=Aloe14)
summary(meta1)

## Extract the coefficients for the variance component of the random effects
coef1 <- coef(meta1, select="random")

## Convert it into a symmetrix matrix by row major
my.cov <- vec2symMat(coef1, byrow=TRUE)

## Convert it into a correlation matrix
cov2cor(my.cov)

## Plot the multivariate effect sizes
plot(meta1)

## End(Not run)
```

---

 anova

---

*Compare Nested Models with Likelihood Ratio Statistic*


---

### Description

It compares nested models with the likelihood ratio statistic from either `wls`, `meta`, `meta3X` or `rem1` objects. It is a wrapper of [mxCompare](#).

### Usage

```
## S3 method for class 'wls'
anova(object, ..., all=FALSE)
## S3 method for class 'meta'
```

```
anova(object, ..., all=FALSE)
## S3 method for class 'meta3X'
anova(object, ..., all=FALSE)
## S3 method for class 'reml'
anova(object, ..., all=FALSE)
```

### Arguments

object	An object or a list of objects of either class <code>wls</code> , class <code>meta</code> , class <code>meta3</code> or class <code>reml</code> . It will be passed to the base argument in <code>mxCompare</code> .
...	An object or a list of objects of either class <code>wls</code> , class <code>meta</code> , class <code>meta3</code> or class <code>reml</code> . It will be passed to the comparison argument in <code>mxCompare</code> .
all	A boolean value on whether to compare all bases with all comparisons. It will be passed to the <code>all</code> argument in <code>mxCompare</code> .

### Value

A table of comparisons between the models in base and comparison.

### Note

When the objects are class `wls`, the degrees of freedom in the base and comparison models are incorrect, while the degrees of freedom on the difference between them is correct. If users want to obtain the correct degrees of freedom in the base and comparison models, they may individually apply the `summary` function on the base and comparison models.

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### Examples

```
## Test the significance of a predictor with likelihood ratio test
## Model0: No predictor
model0 <- meta(y=yi, v=vi, data=Hox02, model.name="No predictor")

## Model1: With a predictor
model1 <- meta(y=yi, v=vi, x=weeks, data=Hox02, model.name="One predictor")

## Compare these two models
anova(model1, model0)
```

as.mxMatrix

*Convert a Matrix into MxMatrix-class***Description**

It converts a matrix into MxMatrix-class via mxMatrix.

**Usage**

```
as.mxMatrix(x, name, ...)
```

**Arguments**

x	A character or numeric matrix. If x is not a matrix, as.matrix(x) is applied first.
name	An optional character string as the name of the MxMatrix object created by mxModel function. If name is missing, the name of x will be used.
...	Further arguments to be passed to <a href="#">mxMatrix</a> . It should be noted that type, nrow, ncol, values, free, name and labels will be created automatically. Thus, these arguments excepts labels should be avoided in ...

**Details**

If there are non-numeric values in x, they are treated as the labels of the free parameters. If an "\*" is present, the numeric value on the left hand side will be treated as the starting value for a free parameter or a fixed value for a fixed parameter. If it is a matrix of numeric values, there is no free parameters in the output matrix.

**Value**

A [MxMatrix-class](#) object with the same dimensions as x

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[mxMatrix](#), [create.mxMatrix](#), [create.Fmatrix](#)

**Examples**

```
## a and b are free parameters with starting values and labels
(a1 <- matrix(c(1:4, "5*a", 6, "7*b", 8, 9), ncol=3, nrow=3))
#      [,1] [,2] [,3]
# [1,] "1"  "4"  "7*b"
# [2,] "2"  "5*a" "8"
# [3,] "3"  "6"  "9"
```

```

a1 <- as.mxMatrix(a1)

## a and b are fixed parameters without any labels, name="new2"
(a2 <- matrix(1:9, ncol=3, nrow=3))
#      [,1] [,2] [,3]
# [1,]  1   4   7
# [2,]  2   5   8
# [3,]  3   6   9

new2 <- as.mxMatrix(a2, name="new2")

## Free parameters without starting values
(a3 <- matrix(c(1:4, "*a", 6, "*b", 8, 9), ncol=3, nrow=3))
#      [,1] [,2] [,3]
# [1,] "1"  "4"  "*b"
# [2,] "2"  "*a" "8"
# [3,] "3"  "6"  "9"

a3 <- as.mxMatrix(a3, lbound=0)

## A free parameter without label
(a4 <- matrix(c(1:4, "5*", 6, "7*b", 8, 9), ncol=3, nrow=3))
#      [,1] [,2] [,3]
# [1,] "1"  "4"  "7*b"
# [2,] "2"  "5*" "8"
# [3,] "3"  "6"  "9"

a4 <- as.mxMatrix(a4)

## Convert a scalar into mxMatrix object
## "name" is required as "3*a" is not a valid name.
(a5 <- as.mxMatrix("3*a", name="a5"))

```

---

asyCov

---

*Compute Asymptotic Covariance Matrix of a Correlation/Covariance Matrix*


---

### Description

It computes the asymptotic sampling covariance matrix of a correlation/covariance matrix under the assumption of multivariate normality.

### Usage

```

asyCov(x, n, cor.analysis = TRUE, dropNA = FALSE, as.matrix = TRUE,
       acov=c("individual", "unweighted", "weighted"),
       suppressWarnings = TRUE, silent = TRUE, run = TRUE, ...)

```



**Arguments**

x	A correlation/covariance matrix or a list of correlation/covariance matrices. NA on the variables or other values defined in <code>na.strings</code> will be removed before the analysis. Note that it only checks the diagonal elements of the matrices. If there are missing values, make sure that the diagonals are coded with NA or values defined in <code>na.strings</code> .
n	Sample size or a vector of sample sizes
cor.analysis	Logical. The output is either a correlation or covariance matrix.
dropNA	Logical. If it is TRUE, the resultant dimensions will be reduced by dropping the missing variables. If it is FALSE, the resultant dimensions are the same as the input by keeping the missing variables.
as.matrix	Logical. If it is TRUE and x is a list of correlation/covariance matrices with the same dimensions, the asymptotic covariance matrices will be column vectorized and stacked together. If it is FALSE, the output will be a list of asymptotic covariance matrices. Note that if it is TRUE, dropNA will be FALSE automatically. This option is useful when passing the asymptotic covariance matrices to <a href="#">meta</a>
acov	If it is individual (the default), the sampling variance covariance matrices are calculated based on individual correlation/covariance matrix. If it is either unweighted or weighted, the average correlation/covariance matrix is calculated based on the unweighted or weighted mean with the sample sizes. The average correlation/covariance matrix is used to calculate the sampling variance covariance matrices.
suppressWarnings	Logical. If TRUE, warnings are suppressed. Argument to be passed to <a href="#">mxRun</a> .
silent	Logical. Argument to be passed to <a href="#">mxRun</a>
run	Logical. If FALSE, only return the mx model without running the analysis.
...	Futher arguments to be passed to <a href="#">mxRun</a>

**Value**

An asymptotic covariance matrix of the vectorized correlation/covariance matrix or a list of these matrices. If `as.matrix=TRUE` and x is a list of matrices, the output is a stacked matrix.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

Cheung, M. W.-L., & Chan, W. (2004). Testing dependent correlation coefficients via structural equation modeling. *Organizational Research Methods*, *7*, 206-223.

**Examples**

```

C1 <- matrix(c(1,0.5,0.4,0.5,1,0.2,0.4,0.2,1), ncol=3)
asyCov(C1, n=100)

## Data with missing values
C2 <- matrix(c(1,0.4,NA,0.4,1,NA,NA,NA,NA), ncol=3)
C3 <- matrix(c(1,0.2,0.2,1), ncol=2)

## Output is a list of asymptotic covariance matrices
asyCov(list(C1,C2,C3), n=c(100,50,50), dropNA=TRUE, as.matrix=FALSE)

## Output is a stacked matrix of asymptotic covariance matrices
asyCov(list(C1,C2), n=c(100,50), as.matrix=TRUE)

## Output is a stacked matrix of asymptotic covariance matrices
asyCov(list(C3,C3), n=c(100,50), as.matrix=TRUE)

```

BCG

*Dataset on the Effectiveness of the BCG Vaccine for Preventing Tuberculosis*

**Description**

This dataset includes 13 studies on the effectiveness of the Bacillus Calmette-Guerin (BCG) vaccine for preventing tuberculosis (see van Houwelingen, Arends, & Stijnen (2002) for details).

**Usage**

```
data(BCG)
```

**Details**

A list of data with the following structure:

**Trial** Number of the trials

**Author** Authors of the original studies

**Year** Year of the publication

**VD** Vaccinated group with disease

**VWD** Vaccinated group without disease

**NVD** Not vaccinated group with disease

**NVWD** Not vaccinated group without disease

**Latitude** Geographic latitude of the place where the study was done

**Allocation** Method of treatment allocation

**ln\_OR** Natural logarithm of the odds ratio:  $\log((VD/VWD)/(NVD/NVWD))$

**v\_ln\_OR** Sampling variance of ln\_OR:  $1/VD+1/VWD+1/NVD+1/NVWD$

**ln\_Odd\_V** Natural logarithm of the odds of the vaccinated group:  $\log(\text{VD}/\text{VWD})$   
**ln\_Odd\_NV** Natural logarithm of the odds of the not vaccinated group:  $\log(\text{NVD}/\text{NVWD})$   
**v\_ln\_Odd\_V** Sampling variance of ln\_Odd\_V:  $1/\text{VD}+1/\text{VWD}$   
**cov\_V\_NV** Sampling covariance between ln\_Odd\_V and ln\_Odd\_NV: It is always 0  
**v\_ln\_Odd\_NV** Sampling variance of ln\_Odd\_NV:  $1/\text{NVD}+1/\text{NVWD}$

### Source

Colditz, G. A., Brewer, T. F., Berkey, C. S., Wilson, M. E., Burdick, E., Fineberg, H. V., & Mosteller, F. (1994). Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association*, **271**, 698–702.

### References

Berkey, C. S., Hoaglin, D. C., Mosteller, F., & Colditz, G. A. (1995). A random-effects regression model for meta-analysis. *Statistics in Medicine*, **14**, 395–411.

van Houwelingen, H. C., Arends, L. R., & Stijnen, T. (2002). Advanced methods in meta-analysis: Multivariate approach and meta-regression. *Statistics in Medicine*, **21**, 589–624.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

### Examples

```
data(BCG)

## Univariate meta-analysis on the log of the odds ratio
summary( meta(y=ln_OR, v=v_ln_OR, data=BCG,
             x=cbind(scale(Latitude,scale=FALSE),
                    scale(Year,scale=FALSE))) )

## Multivariate meta-analysis on the log of the odds
## The conditional sampling covariance is 0
bcg <- meta(y=cbind(ln_Odd_V, ln_Odd_NV), data=BCG,
           v=cbind(v_ln_Odd_V, cov_V_NV, v_ln_Odd_NV))
summary(bcg)

plot(bcg)
```

---

bdiagMat

*Create a Block Diagonal Matrix*


---

### Description

It creates a block diagonal matrix from a list of numeric or character matrices.

### Usage

```
bdiagMat(x)
```

**Arguments**

x                    A list of numeric or character matrices (or values)

**Value**

A numeric or character block diagonal matrix

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

It was based on a function posted by Scott Chasalow at <http://www.math.yorku.ca/Who/Faculty/Monette/pub/stmp/0827.html>.

**See Also**

[bdiagRep](#), [matrix2bdiag](#)

**Examples**

```
## Block diagonal matrix of numerics
bdiagMat( list(matrix(1:4,nrow=2,ncol=2),
               matrix(5:6,nrow=1,ncol=2)) )
#      [,1] [,2] [,3] [,4]
# [1,]  1   3   0   0
# [2,]  2   4   0   0
# [3,]  0   0   5   6

## Block diagonal matrix of characters
bdiagMat( list(matrix(letters[1:4],nrow=2,ncol=2),
               matrix(letters[5:6],nrow=1,ncol=2)) )
#      [,1] [,2] [,3] [,4]
# [1,] "a"  "c"  "0"  "0"
# [2,] "b"  "d"  "0"  "0"
# [3,] "0"  "0"  "e"  "f"
```

---

bdiagRep

---

*Create a Block Diagonal Matrix by Repeating the Input*


---

**Description**

It creates a block diagonal matrix by repeating the input matrix several times.

**Usage**

```
bdiagRep(x, times)
```

**Arguments**

x                    A numeric or character matrix (or values)  
 times                Number of times of x to be repeated

**Value**

A numeric or character block diagonal matrix

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[bdiagMat](#), [matrix2bdiag](#)

**Examples**

```
## Block diagonal matrix of numerics
bdiagRep( matrix(1:4,nrow=2,ncol=2), 2 )
#        [,1] [,2] [,3] [,4]
# [1,]    1    3    0    0
# [2,]    2    4    0    0
# [3,]    0    0    1    3
# [4,]    0    0    2    4

## Block diagonal matrix of characters
bdiagRep( matrix(letters[1:4],nrow=2,ncol=2), 2 )
#        [,1] [,2] [,3] [,4]
# [1,] "a" "c" "0" "0"
# [2,] "b" "d" "0" "0"
# [3,] "0" "0" "a" "c"
# [4,] "0" "0" "b" "d"
```

---

Becker09

*Ten Studies of Correlation Matrices used by Becker (2009)*

---

**Description**

This data set includes ten studies on the relationships between CSAI subscales and sport behavior. The original data were used in Craft et al. (2003), whereas a subset of them were illustrated in Becker (2009).

**Usage**

```
data("Becker09")
```

## Details

A list of data with the following structure:

**data** A list of 4x4 correlation matrices. The variables are *Performance*, *Cognitive*, *Somatic*, and *Self confidence*

**n** A vector of sample sizes

**Type\_of\_sport** Samples based on *Individual* or *Team*

## Source

Craft, L. L., Magyar, T. M., Becker, B. J., & Feltz, D. L. (2003). The relationship between the Competitive State Anxiety Inventory-2 and sport performance: a meta-analysis. *Journal of Sport and Exercise Psychology*, **25**(1), 44-65.

## References

Becker, B. J. (2009). Model-based meta-analysis. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 377-395). New York: Russell Sage Foundation.

## Examples

```
## Not run:
data(Becker09)

#### Fixed-effects model
## First stage analysis
fixed1 <- tssem1(Becker09$data, Becker09$n, method="FEM")
summary(fixed1)

## Prepare a regression model using create.mxMatrix()
A1 <- create.mxMatrix(c(0, "0.1*Cog2Per", "0.1*S02Per", "0.1*SC2Per",
                        0, 0, 0, 0,
                        0, 0, 0, 0,
                        0, "0.1*Cog2SC", "0.1*S02SC", 0),
                      type="Full", byrow=TRUE, ncol=4, nrow=4,
                      as.mxMatrix=FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c("Per", "Cog", "S0", "SC")

## Display A1
A1

S1 <- create.mxMatrix(c("0.1*var_Per",
                        0, 1,
                        0, "0.1*cor", 1,
                        0, 0, 0, "0.1*var_SC"), byrow=TRUE, type="Symm",
                      as.mxMatrix=FALSE)

## This step is not necessary but it is useful for inspecting the model.
```

```

dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c("Per", "Cog", "SO", "SC")

## Display S1
S1

## Second stage analysis
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, diag.constraints=TRUE,
  intervals.type="LB", model.name="TSSEM2 Becker09",
  mx.algebras=list( Cog=mxAlgebra(Cog2SC*SC2Per, name="Cog"),
    SO=mxAlgebra(SO2SC*SC2Per, name="SO"),
    Cog_SO=mxAlgebra(Cog2SC*SC2Per+SO2SC*SC2Per,
      name="Cog_SO")) )

summary(fixed2)

#### Fixed-effects model: with type of sport as cluster
## First stage analysis
cluster1 <- tssem1(Becker09$data, Becker09$n, method="FEM",
  cluster=Becker09$Type_of_sport)

summary(cluster1)

## Second stage analysis
cluster2 <- tssem2(cluster1, Amatrix=A1, Smatrix=S1, diag.constraints=TRUE,
  intervals.type="LB", model.name="TSSEM2 Becker09",
  mx.algebras=list( Cog=mxAlgebra(Cog2SC*SC2Per, name="Cog"),
    SO=mxAlgebra(SO2SC*SC2Per, name="SO"),
    Cog_SO=mxAlgebra(Cog2SC*SC2Per+SO2SC*SC2Per,
      name="Cog_SO")) )

summary(cluster2)

## Convert the model to semPlotModel object with 2 plots
## Use the short forms of the variable names
my.plots <- lapply(X=cluster2, FUN=meta2semPlot, manNames=c("Per", "Cog", "SO", "SC") )

## Load the library
library("semPlot")

## Setup two plots
layout(t(1:2))
## The labels are overlapped. We may modify it by using layout="spring"
semPaths(my.plots[[1]], whatLabels="est", nCharNodes=10, color="orange",
  layout="spring", edge.label.cex=0.8)
title("Individual sport")
semPaths(my.plots[[2]], whatLabels="est", nCharNodes=10, color="skyblue",
  layout="spring", edge.label.cex=0.8)
title("Team sport")

#### Random-effects model
## First stage analysis
random1 <- tssem1(Becker09$data, Becker09$n, method="REM", RE.type="Diag")
summary(random1)

## Second stage analysis

```

```

random2 <- tssem2(random1, Amatrix=A1, Smatrix=S1, diag.constraints=TRUE,
                 intervals.type="LB", model.name="TSSEM2 Becker09",
                 mx.algebras=list( Cog=mxAlgebra(Cog2SC*SC2Per, name="Cog"),
                                   SO=mxAlgebra(SO2SC*SC2Per, name="SO"),
                                   Cog_SO=mxAlgebra(Cog2SC*SC2Per+SO2SC*SC2Per,
                                                    name="Cog_SO")) )

summary(random2)

## Load the library
library("semPlot")

## Convert the model to semPlotModel object
my.plot <- meta2semPlot(random2, manNames=c("Per", "Cog", "SO", "SC"))

## Plot the model with labels
## The labels are overlapped.
## semPaths(my.plot, whatLabels="path", nCharEdges=10)
semPaths(my.plot, whatLabels="path", nCharEdges=10, nCharNodes=10, layout="spring", color="red")

## Plot the parameter estimates
semPaths(my.plot, whatLabels="est", nCharNodes=10, layout="spring", color="green")

## End(Not run)

```

---

Becker83

*Studies on Sex Differences in Conformity Reported by Becker (1983)*


---

### Description

The data set includes studies on sex differences in conformity using the fictitious norm group paradigm reported by Becker (1983).

### Usage

```
data(Becker83)
```

### Details

The variables are:

**study** study number

**di** Standardized mean difference

**vi** Sampling variance of the effect size

**percentage** Percentage of male authors

**items** Number of items



**Source**

Becker, B. J. (1983, April). Influence again: A comparison of methods for meta-analysis. *Paper presented at the annual meeting of the American Educational Research Association, Montreal.*

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. Orlando, FL: Academic Press.

**References**

Cheung, M. W.-L. (2010). Fixed-effects meta-analyses as multiple-group structural equation models. *Structural Equation Modeling*, **17**, 481-509.

**Examples**

```
data(Becker83)

## Random-effects meta-analysis
summary( meta(y=di, v=vi, data=Becker83) )

## Mixed-effects meta-analysis with log(items) as the predictor
summary( meta(y=di, v=vi, x=log(items), data=Becker83) )
```

---

Becker92

*Six Studies of Correlation Matrices reported by Becker (1992; 1995)*


---

**Description**

This data set includes six studies of correlation matrices reported by Becker (1992; 1995).

**Usage**

```
data(Becker92)
```

**Details**

A list of data with the following structure:

**data** A list of 6 studies of correlation matrices. The variables are *Math aptitude*, *Spatial ability* and *Verbal ability*

**n** A vector of sample sizes

**Source**

Becker, B. J. (1992). Using results from replicated studies to estimate linear models. *Journal of Educational Statistics*, **17(4)**, 341-362. doi:10.3102/10769986017004341

Becker, B. J. (1995). Corrections to "Using Results from Replicated Studies to Estimate Linear Models." *Journal of Educational and Behavioral Statistics*, **20(1)**, 100-102. doi:10.2307/1165390

## Examples

```

## Not run:
data(Becker92)

#### Fixed-effects model
## First stage analysis
## Replicate Becker's (1992) analysis using 4 studies only
fixed1 <- tssem1(Becker92$data[1:4], Becker92$N[1:4], method="FEM")
summary(fixed1)

## Prepare a regression model using create.mxMatrix()
A1 <- create.mxMatrix(c(0,0,0,"0.2*Spatial2Math",
                       0,0,"0.2*Verbal2Math",0,0), type="Full",
                      ncol=3, nrow=3, as.mxMatrix=FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c("Math","Spatial","Verbal")

## Display A1
A1

S1 <- create.mxMatrix(c("0.2*ErrorVarMath",0,0,1,"0.2*CorSpatialVerbal",1),
                      type="Symm", as.mxMatrix=FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c("Math","Spatial","Verbal")

## Display S1
S1

## Fixed-effects model: Second stage analysis
## Two equivalent versions to calculate the R2 and its 95% LBCI
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, intervals.type="LB",
                mx.algebras=list(R1=mxAlgebra(Spatial2Math^2+Verbal2Math^2
                                             +2*CorSpatialVerbal*Spatial2Math*Verbal2Math, name="R1"),
                                R2=mxAlgebra(One-Smatrix[1,1], name="R2"),
                                One=mxMatrix("Iden", ncol=1, nrow=1, name="One")))
summary(fixed2)

#### Random-effects model
## First stage analysis
## No random effects for off-diagonal elements
random1 <- tssem1(Becker92$data, Becker92$N, method="REM", RE.type="Diag")
summary(random1)

## Random-effects model: Second stage analysis
random2 <- tssem2(random1, Amatrix=A1, Smatrix=S1, intervals.type="z")
summary(random2)

#### Conventional fixed-effects GLS approach
## First stage analysis
## No random effects

```

```

## Replicate Becker's (1992) analysis using 4 studies only
gls1 <- tssem1(Becker92$data[1:4], Becker92$n[1:4], method="REM", RE.type="Zero",
              model.name="Fixed effects GLS Stage 1")
summary(gls1)

## Fixed-effects GLS model: Second stage analysis
gls2 <- tssem2(gls1, Amatrix=A1, Smatrix=S1, intervals.type="z",
              model.name="Fixed effects GLS Stage 2")
summary(gls2)

## Load the library
library("semPlot")

## Convert the model to semPlotModel object
my.plot <- meta2semPlot(gls2)

## Plot the model with labels
semPaths(my.plot, whatLabels="path", nCharEdges=10, nCharNodes=10, color="red")

## Plot the parameter estimates
semPaths(my.plot, whatLabels="est", nCharNodes=10, color="green")

## End(Not run)

```

---

Becker94

*Five Studies of Ten Correlation Matrices reported by Becker and Schram (1994)*


---

## Description

This data set includes five studies of ten correlation matrices reported by Becker and Schram (1994).

## Usage

```
data(Becker94)
```

## Details

A list of data with the following structure:

**data** A list of 10 correlation matrices. The variables are *SAT (Math)*, *Spatial* and *SAT (Verbal)*

**n** A vector of sample sizes

**gender** *Females* or *Males* samples

## Source

Becker, B. J., & Schram, C. M. (1994). Examining explanatory models through research synthesis. In H. Cooper & L. V. Hedges (Eds.), *The handbook of research synthesis* (pp. 357-381). New York: Russell Sage Foundation.

## Examples

```

## Not run:
data(Becker94)

#### Fixed-effects model
## First stage analysis
fixed1 <- tssem1(Becker94$data, Becker94$n, method="FEM")
summary(fixed1)

## Prepare a regression model using create.mxMatrix()
A1 <- create.mxMatrix(c(0,0,0,"0.2*Spatial2Aptitude",
  0,0,"0.2*Verbal2Aptitude",0,0), type="Full",
  ncol=3, nrow=3, name="A1")
S1 <-
create.mxMatrix(c("0.2*ErrorVarAptitude",0,0,1,"0.2*CorBetweenSpatialVerbal",1),
  type="Symm", name="S1")

## An alternative method to create a regression model using mxMatrix()
# A1 <- mxMatrix("Full", ncol=3, nrow=3, value=0, free=c(FALSE,FALSE,FALSE,TRUE,FALSE,
# FALSE,TRUE,FALSE,FALSE),
# label=c(NA,NA,NA,"Spatial2Aptitude",NA,NA,"Verbal2Aptitude",NA,NA),
# name="A1")
# S1 <- mxMatrix("Symm", ncol=3, nrow=3, value=c(0.5,0,0,1,0.2,1),
# free=c(TRUE,FALSE,FALSE,FALSE,TRUE,FALSE),
# label=c("ErrorVarAptitude",NA,NA,NA,"CorBetweenSpatialVerbal",NA),
# name="S1")

## Second stage analysis
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, intervals.type="LB")
summary(fixed2)

#### Fixed-effects model: with gender as cluster
## First stage analysis
cluster1 <- tssem1(Becker94$data, Becker94$n, method="FEM", cluster=Becker94$gender)
summary(cluster1)

## Second stage analysis
cluster2 <- tssem2(cluster1, Amatrix=A1, Smatrix=S1, intervals.type="LB")
summary(cluster2)

#### Conventional fixed-effects GLS approach
## First stage analysis
## No random effects
## Replicate Becker's (1992) analysis using 4 studies only
gls1 <- tssem1(Becker92$data[1:4], Becker92$n[1:4], method="REM", RE.type="Zero",
  model.name="Fixed effects GLS Stage 1")
summary(gls1)

## Fixed-effects GLS model: Second stage analysis
gls2 <- tssem2(gls1, Amatrix=A1, Smatrix=S1, intervals.type="LB",

```

```

                                model.name="Fixed effects GLS Stage 2")
summary(gls2)

## End(Not run)

```

---

 Berkey98

*Five Published Trails from Berkey et al. (1998)*


---

### Description

The data set includes five published trials, reported by Berkey et al. (1998), comparing surgical and non-surgical treatments for medium-severity periodontal disease, one year after treatment.

### Usage

```
data(Berkey98)
```

### Details

The variables are:

**trial** Trial number

**pub\_year** Publication year

**no\_of\_patients** Number of patients

**PD** Patient improvements (mm) in *probing depth*

**AL** Patient improvements (mm) in *attachment level*

**var\_PD** Sampling variance of PD

**cov\_PD\_AL** Sampling covariance between PD and AD

**var\_AL** Sampling variance of AL

### Source

Berkey, C. S., Hoaglin, D. C., Antczak-Bouckoms, A., Mosteller, F., & Colditz, G. A. (1998). Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine*, **17**, 2537-2550.

### Examples

```

data(Berkey98)

#### ML estimation method
## Multivariate meta-analysis
summary( meta(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL),
                                data=Berkey98, I2=c("I2q", "I2hm", "I2am")) )

## Multivariate meta-analysis with "publication year-1979" as the predictor
summary( meta(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL),

```

```

x=scale(pub_year, center=1979), data=Berkey98,
RE.lbound=NA) )

## Multivariate meta-analysis with equality constraint on the regression coefficients
summary( meta(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL),
x=scale(pub_year, center=1979), data=Berkey98,
coef.constraints=matrix(c("0.3*Eq_slope", "0.3*Eq_slope"),
nrow=2)) )

#### REML estimation method
## Multivariate meta-analysis
summary( reml(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL),
data=Berkey98,
model.name="Multivariate meta analysis with REML") )

## Multivariate meta-analysis with "publication year-1979" as the predictor
summary( reml(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL),
x=scale(pub_year, center=1979), data=Berkey98) )

```

---

Bornmann07

*A Dataset from Bornmann et al. (2007)*


---

## Description

A dataset from Bornmann et al. (2007) for three-level meta-analysis.

## Usage

```
data(Bornmann07)
```

## Details

The variables are:

**ID** ID of the study

**Study** Study name

**Cluster** Cluster for effect sizes

**logOR** Effect size: log odds ratio

**v** Sampling variance of logOR

**Year** Year of publication

**Type** Type of proposal: either **Grant** or **Fellowship**

**Discipline** Discipline of proposal: either **Physical sciences**, **Life sciences/biology**, **Social sciences/humanities** or **Multidisciplinary**)

**Country** Country of proposal: either **United States**, **Canada**, **Australia**, **United Kingdom** or **Europe**

## Source

Bornmann, L., Mutz, R., & Daniel, H.-D. (2007). Gender differences in grant peer review: A meta-analysis. *Journal of Informetrics*, **1**(3), 226-238. doi:10.1016/j.joi.2007.03.001

## References

Cheung, M. W.-L. (2014). Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods*, **19**, 211-229.

Marsh, H. W., Bornmann, L., Mutz, R., Daniel, H.-D., & O'Mara, A. (2009). Gender Effects in the Peer Reviews of Grant Proposals: A Comprehensive Meta-Analysis Comparing Traditional and Multilevel Approaches. *Review of Educational Research*, **79**(3), 1290-1326. doi:10.3102/0034654309334143

## Examples

```
## Not run:
data(Bornmann07)

#### ML estimation method
## No predictor
summary( meta3(y=logOR, v=v, cluster=Cluster, data=Bornmann07) )

## Type as a predictor
## Grant: 0
## Fellowship: 1
summary( meta3(y=logOR, v=v, x=(as.numeric(Type)-1),
               cluster=Cluster, data=Bornmann07) )

## Centered Year as a predictor
summary( meta3(y=logOR, v=v, x=scale(Year, scale=FALSE),
               cluster=Cluster, data=Bornmann07) )

#### REML estimation method
## No predictor
summary( reml3(y=logOR, v=v, cluster=Cluster, data=Bornmann07) )

## Type as a predictor
## Grants: 0
## Fellowship: 1
summary( reml3(y=logOR, v=v, x=(as.numeric(Type)-1),
               cluster=Cluster, data=Bornmann07) )

## Centered Year as a predictor
summary( reml3(y=logOR, v=v, x=scale(Year, scale=FALSE),
               cluster=Cluster, data=Bornmann07) )

## Handling missing covariates with FIML
## MCAR
## Set seed for replication
set.seed(1000000)

## Copy Bornmann07 to my.df
```

```

my.df <- Bornmann07
## "Fellowship": 1; "Grant": 0
my.df$Type_MCAR <- ifelse(Bornmann07$Type=="Fellowship", yes=1, no=0)

## Create 17 out of 66 missingness with MCAR
my.df$Type_MCAR[sample(1:66, 17)] <- NA
summary(meta3X(y=logOR, v=v, cluster=Cluster, x2=Type_MCAR, data=my.df))

## MAR
Type_MAR <- ifelse(Bornmann07$Type=="Fellowship", yes=1, no=0)

## Create 27 out of 66 missingness with MAR for cases Year<1996
index_MAR <- ifelse(Bornmann07$Year<1996, yes=TRUE, no=FALSE)
Type_MAR[index_MAR] <- NA

## Include auxiliary variable
summary(meta3X(y=logOR, v=v, cluster=Cluster, x2=Type_MAR, av2=Year, data=my.df))

## End(Not run)

```

---

Cheung00

*Fifty Studies of Correlation Matrices used in Cheung and Chan (2000)*

---

## Description

This data set includes fifty studies of correlation matrices on theory of planned theory reported by Cheung and Chan (2000).

## Usage

```
data(Cheung00)
```

## Details

A list of data with the following structure:

**data** A list of 50 studies of correlation matrices. The variables are attitude toward behavior *att*, subjective norm *sn*, behavioral intention *bi*, and behavior *beh*

**n** A vector of sample sizes

## Note

These studies were extracted from the original data set for illustration purpose. Some samples contained two or more correlation matrices and only one of them was arbitrarily selected to avoid the problem of dependence. Moreover, studies with less than 3 correlation coefficients were also excluded.



## Source

Cheung, S.-F., & Chan, D. K.-S. (2000). The role of perceived behavioral control in predicting human behavior: A meta-analytic review of studies on the theory of planned behavior. *Unpublished manuscript*, Chinese University of Hong Kong.

## References

Cheung, M.W.-L., & Cheung, S.-F. (in press). Random-effects models for meta-analytic structural equation modeling: Review, issues, and illustrations. *Research Synthesis Methods*.

## Examples

```
## Not run:
data(Cheung00)

## Variable labels
labels <- colnames(Cheung00$data[[1]])

## Full mediation model
S <- create.mxMatrix(c("1",
                      ".2*cov_att_sn", "1",
                      0, 0, ".2*e_bi",
                      0, 0, 0, ".2*e_beh"),
                    type="Symm", as.mxMatrix=FALSE, byrow=TRUE)
dimnames(S) <- list(labels, labels)
S

A <- matrix(c("0","0","0","0",
              "0","0","0","0",
              ".2*att2bi", ".2*sn2bi", "0", "0",
              "0", "0", ".2*bi2beh", "0"),
            byrow=TRUE, 4, 4)
dimnames(A) <- list(labels, labels)
A

#### Random-effects model

## Stage 1 analysis
random_1 <- tssem1(Cheung00$data, Cheung00$n, method="REM")
summary(random_1)

## Stage 2 analysis
random_2 <- tssem2(random_1, Amatrix=A, Smatrix=S, intervals.type="LB", diag.constraints=TRUE)
summary(random_2)

## Load the library
library("semPlot")

my.plot1 <- meta2semPlot(random_2)
semPaths(my.plot1, whatLabels="est", col="yellow")

## End(Not run)
```

Cheung09

*A Data Set from TSSEM User's Guide Version 1.11 by Cheung (2009)***Description**

Four studies were selected from the data set used by Cheung and Chan (2005; 2009). Some variables were randomly deleted to illustrate the analysis with missing data.

**Usage**

```
data(Cheung09)
```

**Details**

A list of data with the following structure:

**data** A list of 4 studies of correlation matrices

**n** A vector of sample sizes

**Source**

Cheung, M. W.-L. (2009). TSSEM: A LISREL syntax generator for two-stage structural equation modeling (Version 1.11) [Computer software]. Retrieved from <http://courses.nus.edu.sg/course/psycwlm/internet/tssem.zip>.

**References**

Cheung, M. W.-L., & Chan, W. (2005). Meta-analytic structural equation modeling: A two-stage approach. *Psychological Methods*, **10**, 40-64.

Cheung, M. W.-L., & Chan, W. (2009). A two-stage approach to synthesizing covariance matrices in meta-analytic structural equation modeling. *Structural Equation Modeling*, **16**, 28-53.

**Examples**

```
## Not run:
data(Cheung09)

#### Fixed-effects model: Stage 1 analysis
fixed1 <- tssem1(Cheung09$data, Cheung09$n, method="FEM")
summary(fixed1)

## Prepare a model implied matrix
## Factor correlation matrix
Phi <- create.mxMatrix( c("0.3*corf2f1", "0.3*corf3f1", "0.3*corf3f2"),
                        type="Stand", as.mxMatrix=FALSE )

## Error variances
Psi <- create.mxMatrix( paste("0.2*e", 1:9, sep=""), type="Diag",
                        as.mxMatrix=FALSE )
```

```

## Create Smatrix
S1 <- bdiagMat(list(Psi, Phi))
## dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c(paste("x",1:9,sep=""),
##
##                                     paste("f",1:3,sep=""))
## S1
S1 <- as.mxMatrix(S1)

## Factor loadings
Lambda <- create.mxMatrix( c(".3*f1x1", ".3*f1x2", ".3*f1x3", rep(0,9),
                             ".3*f2x4", ".3*f2x5", ".3*f2x6", ".3*f2x7",
                             rep(0,9), ".3*f3x8", ".3*f3x9"), type="Full",
                             ncol=3, nrow=9, as.mxMatrix=FALSE )
Zero1 <- matrix(0, nrow=9, ncol=9)
Zero2 <- matrix(0, nrow=3, ncol=12)

## Create Amatrix
A1 <- rbind( cbind(Zero1, Lambda),
             Zero2 )
## dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c(paste("x",1:9,sep=""),
##
##                                     paste("f",1:3,sep=""))
## A1
A1 <- as.mxMatrix(A1)

## Create Fmatrix
F1 <- create.Fmatrix(c(rep(1,9), rep(0,3)))

#### Fixed-effects model: Stage 2 analysis
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, Fmatrix=F1,
                 intervals.type="LB")
summary(fixed2)

## End(Not run)

```

---

coef *Extract Parameter Estimates from tssem1FEM, tssem1FEM.cluster, tssem1REM, wls, wls.cluster, meta, meta3X, reml and MxRAMModel Objects*

---

## Description

It extracts the parameter estimates from either `tssem1FEM`, `tssem1FEM.cluster`, `tssem1REM`, `wls`, `wls.cluster`, `meta`, `meta3X`, `reml` or `MxRAMModel` objects.

## Usage

```

## S3 method for class 'tssem1FEM'
coef(object, ...)
## S3 method for class 'tssem1FEM.cluster'
coef(object, ...)

```

```
## S3 method for class 'tssem1REM'
coef(object, select = c("all", "fixed", "random"), ...)
## S3 method for class 'wls'
coef(object, ...)
## S3 method for class 'wls.cluster'
coef(object, ...)
## S3 method for class 'meta'
coef(object, select = c("all", "fixed", "random"), ...)
## S3 method for class 'meta3X'
coef(object, select = c("all", "fixed", "random", "allX"), ...)
## S3 method for class 'reml'
coef(object, ...)
## S3 method for class 'MxRAMModel'
coef(object, ...)
```

### Arguments

object	An object returned from either class <code>tssem1FEM</code> , class <code>tssem1FEM.cluster</code> , class <code>tssem1REM</code> , class <code>wls</code> , class <code>wls.cluster</code> , class <code>meta</code> , class <code>reml</code> or class <code>MxRAMModel</code>
select	Select all for both fixed- and random-effects parameters, fixed for the fixed-effects parameters or random for the random-effects parameters. For <code>meta3X</code> objects, <code>allX</code> is used to extract all parameters including the predictors and auxiliary variables.
...	Further arguments; currently none is used

### Value

Parameter estimates for both fixed-effects (if any) and random-effects (if any)

### Note

`coef.MxRAMModel` is simply a wrapper of `omxGetParameters`. Extra arguments will be passed to it

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### See Also

[tssem1](#), [wls](#), [meta](#), [reml](#), [omxGetParameters](#)

### Examples

```
## Random-effects meta-analysis
model1 <- meta(y=yi, v=vi, data=Hox02)
coef(model1)

## Fixed-effects only
coef(model1, select="fixed")
```

Cooper03

*Selected effect sizes from Cooper et al. (2003)***Description**

Fifty-six effect sizes from 11 districts from Cooper et al. (2003) were reported by Konstantopoulos (2011).

**Usage**

```
data(Cooper03)
```

**Details**

The variables are:

**District** District ID

**Study** Study ID

**y** Effect size

**v** Sampling variance

**Year** Year of publication

**Source**

Cooper, H., Valentine, J. C., Charlton, K., & Melson, A. (2003). The Effects of Modified School Calendars on Student Achievement and on School and Community Attitudes. *Review of Educational Research*, **73**(1), 1-52. doi:10.3102/00346543073001001

**References**

Konstantopoulos, S. (2011). Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods*, **2**, 61-76. doi:10.1002/jrsm.35

**Examples**

```
## Not run:
data(Cooper03)

#### ML estimation method
## No predictor
summary( model1 <- meta3(y=y, v=v, cluster=District, data=Cooper03) )

## Show all heterogeneity indices and their 95% confidence intervals
summary( meta3(y=y, v=v, cluster=District, data=Cooper03,
              intervals.type="LB", I2=c("I2q", "I2hm", "I2am", "ICC")) )

## Year as a predictor
summary( meta3(y=y, v=v, cluster=District, x=scale(Year, scale=FALSE),
```

```

        data=Cooper03, model.name="Year as a predictor") )

## Equality of level-2 and level-3 heterogeneity
summary( model2 <- meta3(y=y, v=v, cluster=District, data=Cooper03,
                        RE2.constraints="0.2*EqTau2",
                        RE3.constraints="0.2*EqTau2",
                        model.name="Equal Tau2") )

## Compare model2 vs. model1
anova(model1, model2)

#### REML estimation method
## No predictor
summary( reml3(y=y, v=v, cluster=District, data=Cooper03) )

## Level-2 and level-3 variances are constrained equally
summary( reml3(y=y, v=v, cluster=District, data=Cooper03,
              RE.equal=TRUE, model.name="Equal Tau2") )

## Year as a predictor
summary( reml3(y=y, v=v, cluster=District, x=scale(Year, scale=FALSE),
              data=Cooper03, intervals.type="LB") )

## Handling missing covariates with FIML
## Create 20/56 MCAR data in Year
set.seed(10000)
Year_MCAR <- Cooper03$Year
Year_MCAR[sample(56, 20)] <- NA
summary( meta3X(y=y, v=v, cluster=District, x2=scale(Year_MCAR, scale=FALSE),
              data=Cooper03, model.name="NA in Year_MCAR") )

## End(Not run)

```

---

```
create.Fmatrix
```

```
Create an F matrix to select observed variables
```

---

## Description

It creates an F matrix to select observed variables for wls function.

## Usage

```
create.Fmatrix(x, name, as.mxMatrix=TRUE, ...)
```

## Arguments

x	A vector of logical type
name	Name of the matrix. If it is missing, "Fmatrix" will be used.
as.mxMatrix	Logical. If it is TRUE, the output is a matrix of MxMatrix-class. If it is FALSE, it is a numeric matrix.
...	Not used.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[as.mxMatrix](#), [create.mxMatrix](#), [wls](#)

**Examples**

```
## Select the first 3 variables while the other 2 variables are latent.
create.Fmatrix(c(1,1,1,0,0))
# FullMatrix 'Fmatrix'
#
# @labels: No labels assigned.
#
# @values
#      [,1] [,2] [,3] [,4] [,5]
# [1,]  1   0   0   0   0
# [2,]  0   1   0   0   0
# [3,]  0   0   1   0   0
#
# @free: No free parameters.
#
# @lbound: No lower bounds assigned.
#
# @ubound: No upper bounds assigned.

create.Fmatrix(c(1,1,1,0,0), as.mxMatrix=FALSE)
#      [,1] [,2] [,3] [,4] [,5]
# [1,]  1   0   0   0   0
# [2,]  0   1   0   0   0
# [3,]  0   0   1   0   0
```

---

create.mxMatrix

*Create a Vector into MxMatrix-class*

---

**Description**

It converts a vector into MxMatrix-class via mxMatrix.

**Usage**

```
create.mxMatrix(x, type=c("Full", "Symm", "Diag", "Stand"), ncol=NA,
nrow=NA, as.mxMatrix=TRUE, byrow=FALSE, ...)
```

**Arguments**

x	A character or numeric vector
type	Matrix type similar to those listed in <a href="#">mxMatrix</a>
ncol	Number of columns. It is necessary when type="Full". It is ignored and determined by the length of x for the other types of matrices.
nrow	Number of rows. It is necessary when type="Full". It is ignored and determined by the length of x for the other types of matrices.
as.mxMatrix	Logical. If it is TRUE, the output is a matrix of MxMatrix-class. If it is FALSE, it is a numeric matrix.
byrow	Logical. If FALSE (the default) the matrix is filled by columns, otherwise the matrix is filled by rows.
...	Further arguments to be passed to <a href="#">mxMatrix</a> . Please note that type, nrow, ncol, values, free and labels will be created automatically. Thus, these arguments excepts labels should be avoided in ...

**Details**

If there are non-numeric values in x, they are treated as the labels of the free parameters. If an "\*" is present, the numeric value on the left hand side will be treated as the starting value for a free parameter or a fixed value for a fixed parameter. If it is a matrix of numeric values, there is no free parameters in the output matrix. nrow and ncol will be calculated from the length of x unless type="Full" is specified.

**Value**

A [MxMatrix-class](#) object with the same dimensions as x

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[mxMatrix](#), [create.mxMatrix](#), [create.Fmatrix](#)

**Examples**

```
## a and b are free parameters with starting values and labels
(a1 <- c(1:4, "5*a", 6, "7*b", 8, 9))

(mat1 <- create.mxMatrix(a1, ncol=3, nrow=3, name="mat1"))

## Arrange the elements by row
(mat2 <- create.mxMatrix(a1, ncol=3, nrow=3, as.mxMatrix=FALSE, byrow=TRUE))

(a3 <- c(1:3, "4*f4", "5*f5", "6*f6"))

(mat3 <- create.mxMatrix(a3, type="Symm", name="mat3"))
```



```
## Create character matrix
(mat4 <- create.mxMatrix(a3, type="Symm", as.mxMatrix=FALSE))

## Arrange the elements by row
(mat5 <- create.mxMatrix(a3, type="Symm", as.mxMatrix=FALSE, byrow=TRUE))

(mat6 <- create.mxMatrix(a3, type="Diag", lbound=6:1, name="mat6"))
```

---

Diag

*Matrix Diagonals*

---

## Description

Extract or replace the diagonal of a matrix, or construct a diagonal matrix with the same behaviors as `diag` prior to R-3.0.0.

## Usage

```
Diag(x, ...)
Diag(x) <- value
```

## Arguments

<code>x</code>	a matrix, vector or 1D array, or missing.
<code>...</code>	Optional dimensions ( <code>nrow</code> and <code>ncol</code> ) for the result when <code>x</code> is not a matrix.
<code>value</code>	either a single value or a vector of length equal to that of the current diagonal. Should be of a mode which can be coerced to that of <code>x</code> .

## Details

Started from R-3.0.0, `diag(x)` returns a numeric matrix with NA in the diagonals when `x` is a character vector. Although this follows exactly what the manual says, this breaks the metaSEM. The `Diag` has the same functions as `diag` except that `Diag(x)` works for a character vector of `x` by returning a square matrix of character "0" with `x` as the diagonals.

## Note

See <http://r.789695.n4.nabble.com/Behaviors-of-diag-with-character-vector-in-R-3-0-0-td4663735.html> for the discussion.

## Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

## See Also

[diag](#)

**Examples**

```
v <- c("a", "b")
Diag(v)
```

---

 Digman97

---

*Factor Correlation Matrices of Big Five Model from Digman (1997)*


---

**Description**

The data set includes fourteen studies of the factor correlation matrices of the Five-Factor Model of personality reported by Digman (1997).

**Usage**

```
data(Digman97)
```

**Details**

A list of data with the following structure:

**data** A list of 14 studies of correlation matrices. The variables are *Agreeableness* (A), *Conscientiousness* (C), *Emotional Stability* (ES), *Extraversion* (E) and *Intellect* (I)

**n** A vector of sample sizes

**cluster** Types of participants of the studies

**Source**

Digman, J.M. (1997). Higher-order factors of the Big Five. *Journal of Personality and Social Psychology*, **73**, 1246-1256.

**References**

Cheung, M. W.-L., & Chan, W. (2005). Classifying correlation matrices into relatively homogeneous subgroups: A cluster analytic approach. *Educational and Psychological Measurement*, **65**, 954-979.

**Examples**

```
## Not run:
Digman97

##### Fixed-effects TSSEM
fixed1 <- tssem1(Digman97$data, Digman97$n, method="FEM")
summary(fixed1)

## Factor covariance among latent factors
Phi <- matrix(c(1,"0.3*cor","0.3*cor",1), ncol=2, nrow=2)
```

```

## Error covariance matrix
Psi <- Diag(c("0.2*e1", "0.2*e2", "0.2*e3", "0.2*e4", "0.2*e5"))

## S matrix
S1 <- bdiagMat(list(Psi, Phi))

## This step is not necessary but it is useful for inspecting the model.
dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c("A", "C", "ES", "E", "I", "Alpha", "Beta")

## Display S1
S1

## A matrix
Lambda <-
matrix(c(".3*Alpha_A", ".3*Alpha_C", ".3*Alpha_ES", rep(0,5), ".3*Beta_E", ".3*Beta_I"),
       ncol=2, nrow=5)
A1 <- rbind( cbind(matrix(0, ncol=5, nrow=5), Lambda),
            matrix(0, ncol=7, nrow=2) )

## This step is not necessary but it is useful for inspecting the model.
dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c("A", "C", "ES", "E", "I", "Alpha", "Beta")

## Display A1
A1

## F matrix to select the observed variables
F1 <- create.Fmatrix(c(1,1,1,1,1,0,0), as.mxMatrix=FALSE)

## Display F1
F1

fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, Fmatrix=F1,
                 model.name="TSSEM2 Digman97")
summary(fixed2)

#### Fixed-effects TSSEM with several clusters
#### Create a variable for different samples
#### Younger participants: Children and Adolescents
#### Older participants: others
cluster <- ifelse(Digman97$cluster %in% c("Children", "Adolescents"),
                 yes="Younger participants", no="Older participants")

#### Show the cluster
cluster

## Example of Fixed-effects TSSEM with several clusters
fixed1.cluster <- tssem1(Digman97$data, Digman97$n, method="FEM",
                        cluster=cluster)
summary(fixed1.cluster)

fixed2.cluster <- tssem2(fixed1.cluster, Amatrix=A1, Smatrix=S1, Fmatrix=F1)
#### Please note that the estimates for the younger participants are problematic.
summary(fixed2.cluster)

```

```

## Load the library
library("semPlot")

## Convert the model to semPlotModel object with 2 plots
my.plots <- lapply(X=fixed2.cluster, FUN=meta2semPlot, latNames=c("Alpha","Beta"))

## Setup two plots
layout(t(1:2))
semPaths(my.plots[[1]], whatLabels="est", nCharNodes=10, color="green")
semPaths(my.plots[[2]], whatLabels="est", nCharNodes=10, color="green")

#### Random-effects TSSEM with random effects on the diagonals
random1 <- tssem1(Digman97$data, Digman97$n, method="REM",
                 RE.type="Diag")
summary(random1)

random2 <- tssem2(random1, Amatrix=A1, Smatrix=S1, Fmatrix=F1)
summary(random2)

## Convert the model to semPlotModel object
my.plot <- meta2semPlot(random2, latNames=c("Alpha","Beta"))

## Plot the model with labels
semPaths(my.plot, whatLabels="path", nCharEdges=10, nCharNodes=10, color="red")

## Plot the parameter estimates
semPaths(my.plot, whatLabels="est", nCharNodes=10, color="green")

## End(Not run)

```

---

HedgesOlkin85

*Effects of Open Education Reported by Hedges and Olkin (1985)*


---

### Description

Effects of open education on attitude toward school and on reading achievement reported by Hedges and Olkin (1985).

### Usage

```
data(HedgesOlkin85)
```

### Details

The variables are:

**study** Study number

**d\_att** Standardized mean difference on *attitude*

**d\_ach** Standardized mean difference on *achievement*  
**var\_att** Sampling variance of the effect size of *attitude*  
**cov\_att\_ach** Sampling covariance between the effect sizes  
**var\_ach** Sampling variance of the effect size of *achievement*

### Source

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. Orlando, FL: Academic Press.

### References

Cheung, M. W.-L. (2010). Fixed-effects meta-analyses as multiple-group structural equation models. *Structural Equation Modeling*, **17**, 481-509.

### Examples

```
data(Hedges01kin85)

## Fixed-effects meta-analysis
summary( meta(y=cbind(d_att, d_ach),
                    v=cbind(var_att, cov_att_ach, var_ach),
                    data=Hedges01kin85,
                    RE.constraints=matrix(0, nrow=2, ncol=2)) )
```

---

homoStat

*Test the Homogeneity of Effect Sizes*

---

### Description

It tests the homogeneity of univariate and multivariate effect sizes.

### Usage

```
homoStat(y, v)
```

### Arguments

**y** A vector of effect size for univariate meta-analysis or a  $k \times p$  matrix of effect sizes for multivariate meta-analysis where  $k$  is the number of studies and  $p$  is the number of effect sizes.

**v** A vector of the sampling variance of the effect size for univariate meta-analysis or a  $k \times p^*$  matrix of the sampling covariance matrix of the effect sizes for multivariate meta-analysis where  $p^* = p(p + 1)/2$ . It is arranged by column major as used by `vech`. It is assumed that there is no missing value in  $v$  if  $y$  is complete. If there are missing values in  $v$  due to the missingness on  $y$ , the missing values in  $v$  will be removed automatically.

**Value**

A list of

Q	Q statistic on the null hypothesis of homogeneity of effect sizes. It has an approximate chi-square distribution under the null hypothesis.
Q.df	Degrees of freedom of the Q statistic
pval	p value on the test of homogeneity of effect sizes

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

- Becker, B. J. (1992). Using results from replicated studies to estimate linear models. *Journal of Educational Statistics*, **17**, 341-362.
- Cheung, M. W.-L. (2010). Fixed-effects meta-analyses as multiple-group structural equation models. *Structural Equation Modeling*, **17**, 481-509.
- Cochran, W. G. (1954). The combination of estimates from different experiments. *Biometrics*, **10**, 101-129.

**See Also**

[meta](#)

**Examples**

```
with( Hox02, homoStat(yi, vi) )

with( Hedges01kin85, homoStat(y=cbind(d_att, d_ach),
  v=cbind(var_att, cov_att_ach, var_ach)) )
```

---

Hox02

*Simulated Effect Sizes Reported by Hox (2002)*

---

**Description**

Twenty stimulated studies on standardized mean difference and one continuous study characteristic reported by Hox (2002).

**Usage**

```
data(Hox02)
```

## Details

The variables are:

**study** Study number

**yi** Effect size (standardized mean difference)

**vi** Sampling variance of the effect size

**weeks** Duration of the experimental intervention in terms of weeks

## Source

Hox, J. J. (2002). *Multilevel analysis: Techniques and applications*. Mahwah, N.J.: Lawrence Erlbaum Associates.

## References

Cheung, M. W.-L. (2008). A model for integrating fixed-, random-, and mixed-effects meta-analyses into structural equation modeling. *Psychological Methods*, **13**, 182-202.

## Examples

```
## Not run:
data(Hox02)

#### ML estimation method
## Random-effects meta-analysis
summary( meta(y=yi, v=vi, data=Hox02, I2=c("I2q", "I2hm"), intervals.type="LB") )

## Fixed-effects meta-analysis
summary( meta(y=yi, v=vi, data=Hox02, RE.constraints=0,
             model.name="Fixed effects model") )

## Mixed-effects meta-analysis with "weeks" as a predictor
## Request likelihood-based CI
summary( meta(y=yi, v=vi, x=weeks, data=Hox02, intervals.type="LB",
             model.name="Mixed effects meta analysis with LB CI") )

#### REML estimation method
## Random-effects meta-analysis with REML
summary( VarComp <- reml(y=yi, v=vi, data=Hox02) )

## Extract the variance component
VarComp_REML <- matrix( coef(VarComp), ncol=1, nrow=1 )

## Meta-analysis by treating the variance component as fixed
summary( meta(y=yi, v=vi, data=Hox02, RE.constraints=VarComp_REML) )

## Mixed-effects meta-analysis with "weeks" as a predictor
## Request likelihood-based CI
summary( reml(y=yi, v=vi, x=weeks, intervals.type="LB",
             data=Hox02, model.name="REML with LB CI") )
```

```
## End(Not run)
```

---

Hunter83

*Fourteen Studies of Correlation Matrices reported by Hunter (1983)*

---

### Description

This data set includes fourteen studies of Correlation Matrices reported by Hunter (1983)

### Usage

```
data(Hunter83)
```

### Details

A list of data with the following structure:

**data** A list of 14 studies of correlation matrices. The variables are *Ability*, *Job knowledge*, *Work sample* and *Supervisor rating*

**n** A vector of sample sizes

### Source

Hunter, J. E. (1983). A causal analysis of cognitive ability, job knowledge, job performance, and supervisor ratings. In F. Landy, S. Zedeck, & J. Cleveland (Eds.), *Performance Measurement and Theory* (pp. 257-266). Hillsdale, NJ: Erlbaum.

### Examples

```
## Not run:
data(Hunter83)

#### Fixed-effects model
## First stage analysis
fixed1 <- tssem1(Hunter83$data, Hunter83$n, method="FEM",
                 model.name="TSSEM1 fixed effects model")
summary(fixed1)

#### Second stage analysis
## Model without direct effect from Ability to Supervisor
A1 <- create.mxMatrix(c(0,"0.1*A2J","0.1*A2W",0,0,0,"0.1*J2W","0.1*J2S",
                       0,0,0,"0.1*W2S",0,0,0,0),
                      type="Full", ncol=4, nrow=4, as.mxMatrix=FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c("Ability","Job","Work","Supervisor")
A1
```



```

S1 <- create.mxMatrix(c(1,"0.1*Var_e_J", "0.1*Var_e_W", "0.1*Var_e_S"),
                      type="Diag", as.mxMatrix=FALSE)
dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c("Ability","Job","Work","Supervisor")
S1

## diag.constraints=TRUE is required as there are mediators
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, intervals.type="LB",
                 diag.constraints=FALSE,
                 model.name="TSSEM2 fixed effects model")
summary(fixed2)

## Coefficients
coef(fixed2)

## VCOV based on parametric bootstrap
vcov(fixed2)

#### Random-effects model with diagonal elements only
## First stage analysis
random1 <- tssem1(Hunter83$data, Hunter83$n, method="REM", RE.type="Diag",
                  model.name="TSSEM1 random effects model")
summary(random1)

## Second stage analysis
## Model without direct effect from Ability to Supervisor

## diag.constraints=TRUE is required as there are mediators
random2 <- tssem2(random1, Amatrix=A1, Smatrix=S1, intervals.type="LB",
                  diag.constraints=FALSE,
                  mx.algebras=
                    list( ind=mxAlgebra(A2J*J2S+A2J*J2W*W2S+A2W*W2S, name="ind") ),
                  model.name="TSSEM2 random effects model")
summary(random2)

## Load the library
library("semPlot")

## Convert the model to semPlotModel object
my.plot <- meta2semPlot(random2)

## Plot the model with labels
semPaths(my.plot, whatLabels="path", nCharEdges=10, nCharNodes=10, color="red")

## Plot the parameter estimates
semPaths(my.plot, whatLabels="est", nCharNodes=10, color="green")

## End(Not run)

```

---

indirectEffect

*Estimate the asymptotic covariance matrix of standardized or unstandardized indirect and direct effects*


---

**Description**

It estimates the standardized or unstandardized indirect and direct effects and their asymptotic sampling covariance matrix.

**Usage**

```
indirectEffect(x, n, standardized = TRUE, direct.effect = TRUE, run = TRUE)
```

**Arguments**

x	A 3x3 correlation/covariance matrix or a list of correlation/covariance matrices. Variables are arranged as dependent variable (y), mediator (m) and independent variable (x)
n	Sample size or a vector of sample sizes
standardized	Logical. Whether the indirect effect is standardized.
direct.effect	Logical. Whether the direct effect is estimated. If it is FALSE, the direct effect is fixed at zero.
run	Logical. If FALSE, only return the mx model without running the analysis.

**Details**

Cheung (2009) estimated the standardized indirect effect and its standard error with non-linear constraints. Since OpenMx does not generate standard errors when there are non-linear constraints, Kwan and Chan's (2011) approach is used in this function. Delta method is used to calculate the asymptotic covariance matrix.

**Value**

A vector (or a matrix if the input is a list of matrices) of (standardized) indirect effect, standardized direct effect and their asymptotic sampling covariance matrices

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

- Cheung, M. W.-L. (2009). Comparison of methods for constructing confidence intervals of standardized indirect effects. *Behavior Research Methods*, *41*, 425-438.
- Kwan, J., & Chan, W. (2011). Comparing standardized coefficients in structural equation modeling: a model reparameterization approach. *Behavior Research Methods*, *43*, 730-745.

## Examples

```
## A correlation matrix as input
x <- matrix(c(1, 0.4, 0.2, 0.4, 1, 0.3, 0.2, 0.3, 1), ncol=3)
dimnames(x) <- list( c("y", "m", "x"), c("y", "m", "x") )
indirectEffect(x, n=300)

## A list of correlation matrices
indirectEffect( list(x, x), n=c(300,500), standardized=FALSE )
```

---

is.pd

*Test Positive Definiteness of a List of Square Matrices*

---

## Description

It tests the positive definiteness of a square matrix or a list of square matrices. It returns TRUE if the matrix is positive definite. It returns FALSE if the matrix is either non-positive definite or not symmetric. Variables with NA in the diagonals will be removed before testing. It returns NA when there are missing correlations even after deleting the missing variables.

## Usage

```
is.pd(x, check.asyCov=FALSE, cor.analysis=TRUE, tol=1e-06)
```

## Arguments

x	A square matrix or a list of square matrices
check.asyCov	If it is TRUE, it mirrors the checking in <a href="#">asyCov</a> .
cor.analysis	Whether the input matrix is a correlation or a covariance matrix. It is ignored when check.asyCov=FALSE.
tol	Relative tolerance of positiveness of smallest eigenvalue compared to largest eigenvalue. The matrix is considered positive definite if the ratio of the smallest eigenvalue to the largest eigenvalue is larger than tol. See <a href="#">nearPD</a>

## Value

If the input is a matrix, it returns TRUE, FALSE or NA. If the input is a list of matrices, it returns a list of TRUE, FALSE or NA.

## Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Examples**

```

A <- diag(1,3)
is.pd(A)
# TRUE

B <- matrix(c(1,2,2,1), ncol=2)
is.pd(B)
# FALSE

is.pd(list(A, B))
# TRUE FALSE

C <- A
C[2,1] <- C[1,2] <- NA
is.pd(C)
# NA

```

---

issp05

*Data Set from ISSP (2005)*


---

**Description**

Thirty-two covariance matrices on work-related attitudes were extracted from the International Social Survey Programme 2005: Work Orientation III (ISSP, 2005). Seven variables were selected for demonstration purposes. They were grouped into three constructs: *Importance of Job Prospects* measured by job security (JP1), high income (JP2), and opportunity for advancement (JP3); *Importance of Job Autonomy* measured by work independently (JA1) and decide time of work (JA2); and *Importance of Contributions to Society* measured by help other people (CS1) and a job useful to society (CS2).

**Usage**

```
data(issp05)
```

**Details**

A list of data with the following structure:

**data** A list of 32 covariance matrices

**n** A vector of sample sizes

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Source**

ISSP Research Group (2007): *International Social Survey Programme 2005: Work Orientation III (ISSP 2005)*. GESIS Data Archive, Cologne. ZA4350 Data file Version 1.0.0, doi:10.4232/1.4350

**See Also**[issp89](#)**Examples**

```

## Not run:
data(issp05)

#### Fixed-effects TSSEM
fixed1 <- tssem1(issp05$data, issp05$n, method="FEM")
summary(fixed1)

## Prepare a model implied matrix
## Factor correlation matrix
Phi <- create.mxMatrix( c("0.3*corf2f1", "0.3*corf3f1", "0.3*corf3f2"),
                        type="Stand", as.mxMatrix=FALSE )

## Error variances
Psi <- create.mxMatrix( paste("0.2*e", 1:7, sep=""), type="Diag",
                        as.mxMatrix=FALSE )

## Create Smatrix
S1 <- bdiagMat(list(Psi, Phi))
## dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c(paste("x", 1:7, sep=""),
##                                           paste("f", 1:3, sep=""))
##
## S1
S1 <- as.mxMatrix(S1)

## Factor loadings
Lambda <- create.mxMatrix( c(".3*f1x1", ".3*f1x2", ".3*f1x3", rep(0,7),
                            ".3*f2x4", ".3*f2x5", rep(0,7), ".3*f3x6", ".3*f3x7"),
                          type="Full", ncol=3, nrow=7, as.mxMatrix=FALSE )
Zero1 <- matrix(0, nrow=7, ncol=7)
Zero2 <- matrix(0, nrow=3, ncol=10)

## Create Amatrix
A1 <- rbind( cbind(Zero1, Lambda),
            Zero2 )
## dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c(paste("x", 1:7, sep=""),
##                                           paste("f", 1:3, sep=""))
##
## A1
A1 <- as.mxMatrix(A1)

## Create Fmatrix
F1 <- create.Fmatrix(c(rep(1,7), rep(0,3)))

#### Fixed-effects model: Stage 2 analysis
fixed2 <- tssem2(fixed1, Amatrix=A1, Smatrix=S1, Fmatrix=F1,
                intervals.type="LB")
summary(fixed2)

```

```
## End(Not run)
```

---

issp89

*Data Set from Cheung and Chan (2005; 2009)*

---

## Description

Eleven covariance matrices on work-related attitudes were extracted from the Inter-University Consortium for Political and Social Research (1989). Nine variables were selected by Cheung and Chan (2005; 2009) for demonstration purposes. They were grouped into three constructs: *Job Prospects* measured by job security (JP1), income (JP2), and advancement opportunity (JP3); *Job Nature* measured by interesting job (JN1), independent work (JN2), help other people (JN3), and useful to society (JN4); and *Time Demand* measured by flexible working hours (TD1) and lots of leisure time (TD2).

## Usage

```
data(issp89)
```

## Details

A list of data with the following structure:

**data** A list of 11 studies of covariance matrices

**n** A vector of sample sizes

## Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

## Source

Inter-University Consortium for Political and Social Research. (1989). *International Social Survey Program: Work orientation*. Ann Arbor, MI: Author.

## References

Cheung, M. W.-L., & Chan, W. (2005). Meta-analytic structural equation modeling: A two-stage approach. *Psychological Methods*, **10**, 40-64.

Cheung, M. W.-L., & Chan, W. (2009). A two-stage approach to synthesizing covariance matrices in meta-analytic structural equation modeling. *Structural Equation Modeling*, **16**, 28-53.

## See Also

[issp05](#)

**Examples**

```

## Not run:
data(issp89)

#### Analysis of correlation structure in Cheung and Chan (2005)
#### Fixed-effects model: Stage 1 analysis
cor1 <- tssem1(issp89$data, issp89$n, method="FEM", cor.analysis=TRUE)
summary(cor1)

## Prepare a model implied matrix
## Factor correlation matrix
Phi <- create.mxMatrix( c("0.3*corf2f1", "0.3*corf3f1", "0.3*corf3f2"),
                        type="Stand", as.mxMatrix=FALSE )

## Error variances
Psi <- create.mxMatrix( paste("0.2*e", 1:9, sep=""), type="Diag",
                        as.mxMatrix=FALSE )

## Create Smatrix
S1 <- bdiagMat(list(Psi, Phi))
## dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c(paste("x", 1:9, sep=""),
##                                             paste("f", 1:3, sep=""))
## S1
S1 <- as.mxMatrix(S1)

## Factor loadings
Lambda <- create.mxMatrix( c(".3*f1x1", ".3*f1x2", ".3*f1x3", rep(0, 9),
                             ".3*f2x4", ".3*f2x5", ".3*f2x6", ".3*f2x7",
                             rep(0, 9), ".3*f3x8", ".3*f3x9"), type="Full",
                           ncol=3, nrow=9, as.mxMatrix=FALSE )
Zero1 <- matrix(0, nrow=9, ncol=9)
Zero2 <- matrix(0, nrow=3, ncol=12)

## Create Amatrix
A1 <- rbind( cbind(Zero1, Lambda),
            Zero2 )
## dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c(paste("x", 1:9, sep=""),
##                                             paste("f", 1:3, sep=""))
## A1
A1 <- as.mxMatrix(A1)

## Create Fmatrix
F1 <- create.Fmatrix(c(rep(1, 9), rep(0, 3)))

#### Fixed-effects model: Stage 2 analysis
cor2 <- tssem2(cor1, Amatrix=A1, Smatrix=S1, Fmatrix=F1, intervals.type="LB")
summary(cor2)

#### Analysis of covariance structure in Cheung and Chan (2009)
#### Fixed-effects model: Stage 1 analysis
cov1 <- tssem1(issp89$data, issp89$n, method="FEM", cor.analysis=FALSE)
summary(cov1)

```

```
#### Fixed-effects model: Stage 2 analysis
cov2 <- tssem2(cov1, Amatrix=A1, Smatrix=S1, Fmatrix=F1)
summary(cov2)

## End(Not run)
```

---

Jaramillo05

*Dataset from Jaramillo, Mulki & Marshall (2005)*

---

### Description

A dataset of the relationship between organizational commitment (OC) and salesperson job performance (JP) from Jaramillo, Mulki & Marshall (2005).

### Usage

```
data(Jaramillo05)
```

### Format

A data frame with 61 observations on the following 10 variables.

Author a character vector of study

Sample\_size sample size of the study

Sales sample type; either "mixed", "nonsales" or "sales"

Country a character vector of country of study

IDV Hofstede's (1997) individualism index

OC\_scale scale of OC; either "Porter or Mowday", "Meyer" or "other"

OC\_alpha Coefficient alpha of organizational commitment

JP\_alpha Coefficient alpha of job performance

r correlation between organizational commitment and job performance

r\_v sampling variance of r

### Source

Jaramillo, F., Mulki, J. P., & Marshall, G. W. (2005). A meta-analysis of the relationship between organizational commitment and salesperson job performance: 25 years of research. *Journal of Business Research*, **58**(6), 705-714. doi:10.1016/j.jbusres.2003.10.004



**Examples**

```
## Not run:
## Research question 4.4.1
summary(meta(r, r_v, data=Jaramillo05))

## Research question 4.4.2
## Select cases with either "sales" or "nonsales"
Sales.df <- subset(Jaramillo05, Sales %in% c("sales", "nonsales"))

## Create a predictor with 1 and 0 when they are "sales" or "nonsales", respectively
predictor <- ifelse(Jaramillo05$Sales=="sales", yes=1, no=0)

## Mixed-effects meta-analysis
summary( meta(y = r, v = r_v, x = predictor, data = Jaramillo05) )

## Research question 4.4.3
summary(meta(r, r_v, x=IDV, data=Jaramillo05))

## End(Not run)
```

---

list2matrix

---

*Convert a List of Symmetric Matrices into a Stacked Matrix*


---

**Description**

It converts a list of symmetric matrices into a stacked matrix. Dimensions of the symmetric matrices have to be the same. It tries to preserve the dimension names if possible. Dimension names will be created if there is no dimension names in the first symmetric matrix.

**Usage**

```
list2matrix(x, diag = FALSE)
```

**Arguments**

x	A list of $k$ $p \times p$ symmetric matrices.
diag	Logical. If it is TRUE, <a href="#">vech</a> is used to vectorize the (covariance) matrices. If it is FALSE, <a href="#">vechs</a> is used to vectorize the (correlation) matrices.

**Value**

A  $k \times p^*$  stacked matrix where  $p^* = p(p-1)/2$  for `diag=FALSE` or  $p^* = p(p+1)/2$  for `diag=TRUE`.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Examples**

```

C1 <- matrix(c(1,0.5,0.4,0.5,1,0.2,0.4,0.2,1), ncol=3)
C2 <- matrix(c(1,0.4,NA,0.4,1,NA,NA,NA,NA), ncol=3)

## A list without dimension names
list2matrix(list(C1, C2))
#      x2_x1 x3_x1 x3_x2
# [1,]  0.5  0.4  0.2
# [2,]  0.4  NA   NA

dimnames(C1) <- list( c("x","y","z"), c("x","y","z") )
dimnames(C2) <- list( c("x","y","z"), c("x","y","z") )

## A list with dimension names
list2matrix(list(C1, C2))
#      y_x z_x z_y
# [1,] 0.5 0.4 0.2
# [2,] 0.4 NA  NA

```

---

Mak09

*Eight studies from Mak et al. (2009)*


---

**Description**

Eight studies from Mak et al. (2009) were reported by Cheung et al. (2012).

**Usage**

```
data(Mak09)
```

**Format**

A data frame with 8 observations on the following 10 variables.

Study a character vector of study

type a character vector

AF.BP a numeric vector

Tot.BP a numeric vector

AF.non.BP a numeric vector

Tot.non.BP a numeric vector

yi a numeric vector

vi a numeric vector

age.mean a numeric vector

study.duration a numeric vector

**Source**

Mak, A., Cheung, M. W.-L., Ho, R. C. M., Cheak, A. A. C., & Lau, C. S. (2009). Bisphosphonate and atrial fibrillation: Bayesian meta-analyses of randomized controlled trials and observational studies. *BMC Musculoskeletal Disorders*, **10(113)**. doi:10.1186/1471-2474-10-113 Available at <http://www.biomedcentral.com/1471-2474/10/113>.

**References**

Cheung, M. W.-L., Ho, R. C. M., Lim, Y., & Mak, A. (2012). Conducting a meta-analysis: Basics and good practices. *International Journal of Rheumatic Diseases*, **15(2)**, 129-135. doi: 10.1111/j.1756-185X.2012.01712.x

**Examples**

```
## Random-effects meta-analysis
( meta1 <- summary(meta(y=yi, v=vi, data=Mak09, I2=c("I2q", "I2hm"))) )

## Convert the estimates back into odds ratio
OR <- with(coef(meta1), exp(c(Estimate[1], lbound[1], ubound[1])))
names(OR) <- c("Estimate in OR", "lbound in OR", "ubound in OR")
OR

## Mixed-effects meta-analysis with mean age as a predictor
summary( meta(y=yi, v=vi, x=age.mean, data=Mak09) )
```

---

matrix2bdiag

---

*Convert a Matrix into a Block Diagonal Matrix*


---

**Description**

It converts a matrix into a block diagonal matrix.

**Usage**

```
matrix2bdiag(x, ...)
```

**Arguments**

x                    A  $k \times p$  matrix of numerics or characters.  
...                    Further arguments to be passed to [vec2symMat](#)

**Details**

Each row of x is converted into a symmetric matrix via [vec2symMat](#). Then the list of the symmetric matrices is converted into a block diagonal matrix via a function written by Scott Chasalow posted at <http://www.math.yorku.ca/Who/Faculty/Monette/pub/stmp/0827.html>.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[vec2symMat](#)

**Examples**

```
(m1 <- matrix(1:12, ncol=6, byrow=TRUE))
#      [,1] [,2] [,3] [,4] [,5] [,6]
# [1,]  1  2  3  4  5  6
# [2,]  7  8  9 10 11 12

matrix2bdiag(m1)
#      [,1] [,2] [,3] [,4] [,5] [,6]
# [1,]  1  2  3  0  0  0
# [2,]  2  4  5  0  0  0
# [3,]  3  5  6  0  0  0
# [4,]  0  0  0  7  8  9
# [5,]  0  0  0  8 10 11
# [6,]  0  0  0  9 11 12
```

---

meta

*Univariate and Multivariate Meta-Analysis with Maximum Likelihood Estimation*

---

**Description**

It conducts univariate and multivariate meta-analysis with maximum likelihood estimation method. Mixed-effects meta-analysis can be conducted by including study characteristics as predictors. Equality constraints on intercepts, regression coefficients and variance components can be easily imposed by setting the same labels on the parameter estimates.

**Usage**

```
meta(y, v, x, data, intercept.constraints = NULL, coef.constraints = NULL,
      RE.constraints = NULL, RE.startvalues=0.1, RE.lbound = 1e-10,
      intervals.type = c("z", "LB"), I2="I2q", R2=TRUE,
      model.name="Meta analysis with ML", suppressWarnings = TRUE,
      silent = TRUE, run = TRUE, ...)
```

**Arguments**

**y** A vector of effect size for univariate meta-analysis or a  $k \times p$  matrix of effect sizes for multivariate meta-analysis where  $k$  is the number of studies and  $p$  is the number of effect sizes.

<code>v</code>	A vector of the sampling variance of the effect size for univariate meta-analysis or a $k \times p^*$ matrix of the sampling covariance matrix of the effect sizes for multivariate meta-analysis where $p^* = p(p + 1)/2$ . It is arranged by column major as used by <code>vech</code> .
<code>x</code>	A predictor or a $k \times m$ matrix of predictors where $m$ is the number of predictors.
<code>data</code>	An optional data frame containing the variables in the model.
<code>intercept.constraints</code>	A $1 \times p$ matrix specifying whether the intercepts of the effect sizes are fixed or free. If the input is not a matrix, the input is converted into a $1 \times p$ matrix with <code>t(as.matrix(intercept.constraints))</code> . The default is that the intercepts are free. When there is no predictor, these intercepts are the same as the pooled effect sizes. The format of this matrix follows <code>as.mxMatrix</code> . The intercepts can be constrained equally by using the same labels.
<code>coef.constraints</code>	A $p \times m$ matrix specifying how the predictors predict the effect sizes. If the input is not a matrix, it is converted into a matrix by <code>as.matrix()</code> . The default is that all $m$ predictors predict all $p$ effect sizes. The format of this matrix follows <code>as.mxMatrix</code> . The regression coefficients can be constrained equally by using the same labels.
<code>RE.constraints</code>	A $p \times p$ matrix specifying the variance components of the random effects. If the input is not a matrix, it is converted into a matrix by <code>as.matrix()</code> . The default is that all covariance/variance components are free. The format of this matrix follows <code>as.mxMatrix</code> . Elements of the variance components can be constrained equally by using the same labels. If a zero matrix is specified, it becomes a fixed-effects meta-analysis.
<code>RE.startvalues</code>	A vector of $p$ starting values on the diagonals of the variance component of the random effects. If only one scalar is given, it will be duplicated across the diagonals. Starting values for the off-diagonals of the variance component are all 0. A $p \times p$ symmetric matrix of starting values is also accepted.
<code>RE.lbound</code>	A vector of $p$ lower bounds on the diagonals of the variance component of the random effects. If only one scalar is given, it will be duplicated across the diagonals. Lower bounds for the off-diagonals of the variance component are set at NA. A $p \times p$ symmetric matrix of the lower bounds is also accepted.
<code>intervals.type</code>	Either <code>z</code> (default if missing) or <code>LB</code> . If it is <code>z</code> , it calculates the 95% Wald confidence intervals (CIs) based on the $z$ statistic. If it is <code>LB</code> , it calculates the 95% likelihood-based CIs on the parameter estimates. Note that the $z$ values and their associated $p$ values are based on the $z$ statistic. They are not related to the likelihood-based CIs.
<code>I2</code>	Possible options are <code>"I2q"</code> , <code>"I2hm"</code> and <code>"I2am"</code> . They represent the $I^2$ calculated by using a typical within-study sampling variance from the $Q$ statistic, the harmonic mean and the arithmetic mean of the within-study sampling variances (Xiong, Miller, & Morris, 2010). More than one options are possible. If <code>intervals.type="LB"</code> , 95% confidence intervals on the heterogeneity indices will be constructed.
<code>R2</code>	Logical. If TRUE and there are predictors, $R^2$ is calculated (Raudenbush, 2009).
<code>model.name</code>	A string for the model name in <code>mxModel</code> .

suppressWarnings	Logical. If TRUE, warnings are suppressed. Argument to be passed to <code>mxRun</code> .
silent	Logical. Argument to be passed to <code>mxRun</code>
run	Logical. If FALSE, only return the mx model without running the analysis.
...	Futher arguments to be passed to <code>mxRun</code>

**Value**

An object of class `meta` with a list of

<code>call</code>	Object returned by <code>match.call</code>
<code>data</code>	A data matrix of <code>y</code> , <code>v</code> and <code>x</code>
<code>no.y</code>	No. of effect sizes
<code>no.x</code>	No. of predictors
<code>miss.x</code>	A vector indicating whether the predictors are missing. Studies will be removed before the analysis if they are TRUE
<code>I2</code>	Types of I2 calculated
<code>R2</code>	Logical
<code>mx.fit</code>	A fitted object returned from <code>mxRun</code>
<code>mx0.fit</code>	A fitted object without any predictor returned from <code>mxRun</code>

**Note**

Missing values (NA) in `y` and their related elements in `v` will be removed automatically. When there are missing values in `v` but not in `y`, missing values will be replaced by `1e5`. Effectively, these effect sizes will have little impact on the analysis.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

- Cheung, M. W.-L. (2008). A model for integrating fixed-, random-, and mixed-effects meta-analyses into structural equation modeling. *Psychological Methods*, **13**, 182-202.
- Cheung, M. W.-L. (2009). Constructing approximate confidence intervals for parameters with structural equation models. *Structural Equation Modeling*, **16**, 267-294.
- Cheung, M. W.-L. (2013). Multivariate meta-analysis as structural equation models. *Structural Equation Modeling*, **20**, 429-454.
- Hardy, R. J., & Thompson, S. G. (1996). A likelihood approach to meta-analysis with random effects. *Statistics in Medicine*, **15**, 619-629.
- Neale, M. C., & Miller, M. B. (1997). The use of likelihood-based confidence intervals in genetic models. *Behavior Genetics*, **27**, 113-120.
- Raudenbush, S. W. (2009). Analyzing effect sizes: random effects models. In H. M. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295-315). New York: Russell Sage Foundation.

Xiong, C., Miller, J. P., & Morris, J. C. (2010). Measuring study-specific heterogeneity in meta-analysis: application to an antecedent biomarker study of alzheimer's disease. *Statistics in Biopharmaceutical Research*, **2**(3), 300-309. doi:10.1198/sbr.2009.0067

### See Also

[reml](#), [Hox02](#), [Berkey98](#), [wvs94a](#)

---

meta2semPlot

*Convert metaSEM objects into semPlotModel objects for plotting*

---

### Description

It converts objects in class `wls` into objects of class `semPlotModel`.

### Usage

```
meta2semPlot(object, manNames = NULL, latNames = NULL, labels = c("labels", "RAM"), ...)
```

### Arguments

<code>object</code>	An object of class <code>wls</code> returned from <code>wls()</code> or <code>tssem2()</code> .
<code>manNames</code>	A character vector of the manifest names. The program will try to get it from the object if it is not given.
<code>latNames</code>	A character vector of the latent names. The program will create it by using "L1", "L2", etc if it is not given.
<code>labels</code>	Either <code>labels</code> (default if missing) or <code>RAM</code> . If <code>labels</code> , the labels of the parameters are used in plotting. If <code>RAM</code> , the RAM notations are used in plotting.
<code>...</code>	Further arguments to be passed to <a href="#">ramModel</a>

### Details

It uses the `ramModel()` to do the conversion.

### Value

A "semPlotModel" object.

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### See Also

[ramModel](#), [Becker92](#), [Becker09](#), [Digman97](#), [Hunter83](#)

---

meta3	<i>Three-Level Univariate Meta-Analysis with Maximum Likelihood Estimation</i>
-------	--

---

### Description

It conducts three-level univariate meta-analysis with maximum likelihood estimation method. Mixed-effects meta-analysis can be conducted by including study characteristics as predictors. Equality constraints on the intercepts, regression coefficients and variance components on the level-2 and on the level-3 can be easily imposed by setting the same labels on the parameter estimates.

### Usage

```
meta3(y, v, cluster, x, data, intercept.constraints = NULL,
      coef.constraints = NULL, RE2.constraints = NULL,
      RE2.lbound = 1e-10, RE3.constraints = NULL, RE3.lbound = 1e-10,
      intervals.type = c("z", "LB"), I2="I2q",
      R2=TRUE, model.name = "Meta analysis with ML",
      suppressWarnings = TRUE, silent = TRUE, run = TRUE, ...)
meta3X(y, v, cluster, x2, x3, av2, av3, data, intercept.constraints=NULL,
      coef.constraints=NULL, RE2.constraints=NULL, RE2.lbound=1e-10,
      RE3.constraints=NULL, RE3.lbound=1e-10, intervals.type=c("z", "LB"),
      R2=TRUE, model.name="Meta analysis with ML",
      suppressWarnings=TRUE, silent = TRUE, run = TRUE, ...)
```

### Arguments

y	A vector of $k$ studies of effect size.
v	A vector of $k$ studies of sampling variance.
cluster	A vector of $k$ characters or numbers indicating the clusters.
x	A predictor or a $k \times m$ matrix of level-2 and level-3 predictors where $m$ is the number of predictors.
x2	A predictor or a $k \times m$ matrix of level-2 predictors where $m$ is the number of predictors.
x3	A predictor or a $k \times m$ matrix of level-3 predictors where $m$ is the number of predictors.
av2	A predictor or a $k \times m$ matrix of level-2 auxiliary variables where $m$ is the number of variables.
av3	A predictor or a $k \times m$ matrix of level-3 auxiliary variables where $m$ is the number of variables.
data	An optional data frame containing the variables in the model.
intercept.constraints	A $1 \times 1$ matrix specifying whether the intercept of the effect size is fixed or constrained. The format of this matrix follows <a href="#">as.mxMatrix</a> . The intercept can be constrained with other parameters by using the same label.



<code>coef.constraints</code>	A $1 \times m$ matrix specifying how the level-2 and level-3 predictors predict the effect sizes. If the input is not a matrix, it is converted into a matrix by <code>as.matrix()</code> . The default is that all $m$ predictors predict the effect size. The format of this matrix follows <code>as.mxMatrix</code> . The regression coefficients can be constrained equally by using the same labels.
<code>RE2.constraints</code>	A scalar or a $1 \times 1$ matrix specifying the variance components of the random effects. The default is that the variance components is free. The format of this matrix follows <code>as.mxMatrix</code> . Elements of the variance components can be constrained equally by using the same label.
<code>RE2.lbound</code>	A scalar or a $1 \times 1$ matrix of lower bound on the level-2 variance component of the random effects.
<code>RE3.constraints</code>	A scalar or a $1 \times 1$ matrix specifying the variance components of the random effects at level-3. The default is that the variance components is free. The format of this matrix follows <code>as.mxMatrix</code> . Elements of the variance components can be constrained equally by using the same label.
<code>RE3.lbound</code>	A scalar or a $1 \times 1$ matrix of lower bound on the level-3 variance component of the random effects.
<code>intervals.type</code>	Either <code>z</code> (default if missing) or <code>LB</code> . If it is <code>z</code> , it calculates the 95% Wald confidence intervals (CIs) based on the $z$ statistic. If it is <code>LB</code> , it calculates the 95% likelihood-based CIs on the parameter estimates. Note that the $z$ values and their associated $p$ values are based on the $z$ statistic. They are not related to the likelihood-based CIs.
<code>I2</code>	Possible options are <code>"I2q"</code> , <code>"I2hm"</code> , <code>"I2am"</code> and <code>"ICC"</code> . They represent the $I^2$ calculated by using a typical within-study sampling variance from the $Q$ statistic, the harmonic mean, the arithmetic mean of the within-study sampling variances, and the intra-class correlation. More than one options are possible. If <code>intervals.type="LB"</code> , 95% confidence intervals on the heterogeneity indices will be constructed.
<code>R2</code>	Logical. If <code>TRUE</code> and there are predictors, $R^2$ is calculated.
<code>model.name</code>	A string for the model name in <code>mxModel</code> .
<code>suppressWarnings</code>	Logical. If <code>TRUE</code> , warnings are suppressed. Argument to be passed to <code>mxRun</code> .
<code>silent</code>	Logical. Argument to be passed to <code>mxRun</code>
<code>run</code>	Logical. If <code>FALSE</code> , only return the <code>mx</code> model without running the analysis.
<code>...</code>	Futher arguments to be passed to <code>mxRun</code>

## Details

$$y_{ij} = \beta_0 + \beta' * \mathbf{x}_{ij} + u_{(2)ij} + u_{(3)j} + e_{ij}$$

where  $y_{ij}$  is the effect size for the  $i$ th study in the  $j$ th cluster,  $\beta_0$  is the intercept,  $\beta$  is the regression coefficients,  $\mathbf{x}_{ij}$  is a vector of predictors,  $u_{(2)ij} \sim N(0, \tau_2^2)$  and  $u_{(3)j} \sim N(0, \tau_3^2)$  are the level-2

and level-3 heterogeneity variances, respectively, and  $e_{ij} \sim N(0, v_{ij})$  is the conditional known sampling variance.

`meta3()` does not differentiate between level-2 or level-3 variables in  $x$  since both variables are treated as a design matrix. When there are missing values in  $x$ , the data will be deleted. `meta3X()` treats the predictors  $x_2$  and  $x_3$  as level-2 and level-3 variables. Thus, their means and covariance matrix will be estimated. Missing values in  $x_2$  and  $x_3$  will be handled by (full information) maximum likelihood (FIML) in `meta3X()`. Moreover, auxiliary variables  $av_2$  at level-2 and  $av_3$  at level-3 may be included to improve the estimation. Although `meta3X()` is more flexible in handling missing covariates, it is more likely to encounter estimation problems.

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### References

- Cheung, M. W.-L. (2014). Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods*, **19**, 211-229.
- Enders, C. K. (2010). *Applied missing data analysis*. New York: Guilford Press.
- Graham, J. (2003). Adding missing-data-relevant variables to FIML-based structural equation models. *Structural Equation Modeling: A Multidisciplinary Journal*, **10**(1), 80-100.
- Konstantopoulos, S. (2011). Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods*, **2**, 61-76.

### See Also

[rem13](#), [Cooper03](#), [Bornmann07](#)

---

Norton13

*Studies on the Hospital Anxiety and Depression Scale Reported by Norton et al. (2013)*

---

### Description

The data set includes 28 studies on 14 items measuring the Hospital Anxiety and Depression Scale (HADS) Reported by Norton et al. (2013).

### Usage

`data(Norton13)`

**Details**

The variables are:

**data** A list of 28 studies of correlation matrices. The variables are 14 items (x1 to x14) measuring HADS.

**n** A vector of sample sizes

**population** A vector of population of the data

**group** A vector of classification into *patients* vs. *non-patients* based on population

**Source**

Norton, S., Cosco, T., Doyle, F., Done, J., & Sacker, A. (2013). The Hospital Anxiety and Depression Scale: A meta confirmatory factor analysis. *Journal of Psychosomatic Research*, 74(1), 74-81.

**References**

Jak, S., & Cheung, M. W.-L. (2015). Addressing heterogeneity in meta-analytic structural equation modeling using subgroup analysis. *Manuscript in preparation*.

**Examples**

```
data(Norton13)
```

---

pattern.n

*Display the Accumulative Sample Sizes for the Covariance Matrix*

---

**Description**

It displays the accumulative sample sizes for the covariance matrix.

**Usage**

```
pattern.n(x, n)
```

**Arguments**

**x** A list of square matrices  
**n** A vector of sample sizes.

**Value**

A square matrix of the accumulative sample sizes of the input matrices.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Examples**

```
## Show the pattern of missing data
pattern.n(Hunter83$data, Hunter83$n)

#           Ability Knowledge Work sample Supervisor
# Ability   3815         3372      3281      3605
# Knowledge  3372         3532      2998      3322
# Work sample 3281         2998      3441      3231
# Supervisor 3605         3322      3231      3765
```

---

pattern.na

*Display the Pattern of Missing Data of a List of Square Matrices*

---

**Description**

It displays the pattern of missing data (or pattern of data that are present) of a list of square matrices with the same dimensions.

**Usage**

```
pattern.na(x, show.na = TRUE)
```

**Arguments**

x	A list of square matrices
show.na	If it is TRUE, it shows the pattern of missing data. If it is FALSE, it shows the pattern of data that are present.

**Value**

A square matrix of numerical values with the same dimensions of the input matrices.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Examples**

```
## Show the pattern of missing data
pattern.na(Hunter83$data, show.na=TRUE)

#           Ability Knowledge Work sample Supervisor
# Ability     1         3       3         2
# Knowledge   3         2       4         3
# Work sample 3         4       2         3
# Supervisor 2         3       3         1

## Show the pattern of data that are present
```

```
pattern.na(Hunter83$data, show.na=FALSE)

#           Ability Knowledge Work sample Supervisor
# Ability           13           11           11           12
# Knowledge          11           12           10           11
# Work sample        11           10           12           11
# Supervisor         12           11           11           13
```

---

plot

*Plot method for meta objects*


---

## Description

It plots the pooled and the individual effect sizes and their confidence ellipses for meta objects.

## Usage

```
## S3 method for class 'meta'
plot(x, effect.sizes, add.margin = 0.1, interval = 0.95,
     main= "Effect Sizes and their Confidence Ellipses",
     axis.labels= paste("Effect size ", effect.sizes, sep = ""),
     study.col = "black", study.pch = 19, study.min.cex = 0.8,
     study.weight.plot = FALSE, study.ellipse.plot = TRUE,
     study.ellipse.col = "black", study.ellipse.lty = 2,
     study.ellipse.lwd = 0.5,
     estimate.col = "blue", estimate.pch = 18, estimate.cex = 2,
     estimate.ellipse.plot = TRUE, estimate.ellipse.col = "red",
     estimate.ellipse.lty = 1, estimate.ellipse.lwd = 2,
     randeff.ellipse.plot = TRUE, randeff.ellipse.col = "green",
     randeff.ellipse.lty = 1, randeff.ellipse.lwd = 2,
     univariate.plot = TRUE, univariate.lines.col = "gray",
     univariate.lines.lty = 3,
     univariate.lines.lwd = 1, univariate.polygon.width = 0.02,
     univariate.polygon.col = "red", univariate.arrows.col = "green",
     univariate.arrows.lwd = 2,
     diag.panel = FALSE, xlim=NULL, ylim=NULL, ...)
```

## Arguments

<code>x</code>	An object of class <a href="#">meta</a> .
<code>effect.sizes</code>	Numeric values indicating which effect sizes to be plotted. At least two effect sizes are required. To plot the effect sizes of $y_1$ and $y_2$ , one may use <code>effect.sizes=c(1,2)</code> . If it is missing, all effect sizes will be plotted in a pairwise way.
<code>add.margin</code>	Value for additional margins on the left and bottom margins.
<code>interval</code>	Interval for the confidence ellipses.

<code>main</code>	Main title of each plot. If there are multiple plots, a vector of character titles may be used.
<code>axis.labels</code>	Labels for the effect sizes.
<code>study.col</code>	Color of individual studies. See <code>col</code> in <a href="#">par</a> .
<code>study.pch</code>	Plotting character of individual studies. See <code>pch</code> in <a href="#">points</a> .
<code>study.min.cex</code>	Minimum value of <code>cex</code> for individual studies. See <code>cex</code> in <a href="#">par</a> .
<code>study.weight.plot</code>	Logical. If TRUE, the plotting size of individual studies ( <code>cex</code> ) will be proportional to one over the square root of the determinant of the sampling covariance matrix of the study.
<code>study.ellipse.plot</code>	Logical. If TRUE, the confidence ellipses of individual studies are plotted.
<code>study.ellipse.col</code>	Color of the confidence ellipses of individual studies. See <code>col</code> in <a href="#">par</a> .
<code>study.ellipse.lty</code>	Line type of the confidence ellipse of individual studies. See <code>lty</code> in <a href="#">par</a> .
<code>study.ellipse.lwd</code>	Line width of the confidence ellipse of individual studies. See <code>lwd</code> in <a href="#">par</a> .
<code>estimate.col</code>	Color of the estimated effect size. See <code>col</code> in <a href="#">par</a> .
<code>estimate.pch</code>	Plotting character of the estimated effect sizes. See <code>pch</code> in <a href="#">points</a> .
<code>estimate.cex</code>	Amount of plotting of the estimated effect sizes. See <code>cex</code> in <a href="#">par</a> .
<code>estimate.ellipse.plot</code>	Logical. If TRUE, the confidence ellipse of the estimated effect sizes will be plotted.
<code>estimate.ellipse.col</code>	Color of the confidence ellipse of the estimated effect sizes. See <code>col</code> in <a href="#">par</a> .
<code>estimate.ellipse.lty</code>	Line type of the confidence ellipse of the estimated effect sizes. See <code>lty</code> in <a href="#">par</a> .
<code>estimate.ellipse.lwd</code>	Line width of the confidence ellipse of the estimated effect sizes. See <code>lwd</code> in <a href="#">par</a> .
<code>randeff.ellipse.plot</code>	Logical. If TRUE, the confidence ellipses of the random effects will be plotted.
<code>randeff.ellipse.col</code>	Color of the confidence ellipses of the random effects. See <code>col</code> in <a href="#">par</a> .
<code>randeff.ellipse.lty</code>	Line type of the confidence ellipses of the random effects. See <code>lty</code> in <a href="#">par</a> .
<code>randeff.ellipse.lwd</code>	Line width of the confidence ellipses of the random effects. See <code>lwd</code> in <a href="#">par</a> .
<code>univariate.plot</code>	Logical. If TRUE, the estimated univariate effect sizes will be plotted.
<code>univariate.lines.col</code>	Color of the estimated univariate effect sizes. See <code>col</code> in <a href="#">par</a> .

univariate.lines.lty	Line type of the estimated univariate effect sizes. See lty in <a href="#">par</a> .
univariate.lines.lwd	Line width of the estimated univariate effect sizes. See lwd in <a href="#">par</a> .
univariate.polygon.width	Width of the polygon of the estimated univariate effect sizes.
univariate.polygon.col	Color of the polygon of the estimated univariate effect sizes.
univariate.arrows.col	Color of the arrows of the estimated univariate effect sizes.
univariate.arrows.lwd	Line width of the arrows of the estimated univariate effect sizes.
diag.panel	Logical. If TRUE, diagonal panels will be created. They can then be used for forest plots for univariate meta-analysis.
xlim	NULL or a numeric vector of length 2; if it is NULL, it provides defaults estimated from the data.
ylim	NULL or a numeric vector of length 2; if it is NULL, it provides defaults estimated from the data.
...	Further arguments to be passed to the plot function.

**Note**

The estimated effect sizes and random effects are based on the labels Intercept1, Intercept2, ... and Tau2\_1\_1, Tau2\_2\_1, Tau2\_2\_2, etc. At least two effect sizes are required for this function. The .ellipse() function is adopted from the ellipse package written by Duncan Murdoch and E. D. Chow.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

Cheung, M. W.-L. (2013). Multivariate meta-analysis as structural equation models. *Structural Equation Modeling*, **20**, 429-454.

**Examples**

```
## Not run:
## Multivariate meta-analysis
x <- meta(y=cbind(PD, AL), v=cbind(var_PD, cov_PD_AL, var_AL), data=Berkey98)
plot(x)

## Plot individual studies proportional to the weights
plot(x, study.weight.plot=TRUE)

## Include forest plot from the metafor package
library(metafor)
plot(x, diag.panel=TRUE, main="Multivariate meta-analysis",
```

```
axis.label=c("PD", "AL"))
forest( rma(yi=PD, vi=var_PD, data=Berkey98) )
title("Forest plot of PD")
forest( rma(yi=AL, vi=var_AL, data=Berkey98) )
title("Forest plot of AL")

## End(Not run)
```

---

print	<i>Print Methods for tssem1FEM, tssem1FEM.cluster, tssem1REM, wls, meta, meta3X and reml Objects</i>
-------	--

---

## Description

Print methods for either `tssem1FEM`, `tssem1FEM.cluster`, `tssem1REM`, `wls`, `meta`, `meta3X` or `reml` objects.

## Usage

```
## S3 method for class 'tssem1FEM'
print(x, ...)
## S3 method for class 'tssem1FEM.cluster'
print(x, ...)
## S3 method for class 'tssem1REM'
print(x, ...)
## S3 method for class 'wls'
print(x, ...)
## S3 method for class 'meta'
print(x, ...)
## S3 method for class 'meta3X'
print(x, ...)
## S3 method for class 'reml'
print(x, ...)
```

## Arguments

x	An object returned from either class <code>tssem1FEM</code> , class <code>tssem1FEM.cluster</code> , class <code>tssem1REM</code> , class <code>wls</code> , class <code>meta</code> , class <code>meta3X</code> or class <code>reml</code>
...	Further arguments to be passed to <code>summary.default</code>

## Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

## See Also

[tssem1](#), [wls](#), [meta](#), [reml](#)



---

readData	<i>Read External Correlation/Covariance Matrices</i>
----------	--

---

### Description

It reads full/lower triangle/stacked vectors of correlation/covariance data into a list of correlation/covariance matrices.

### Usage

```
readFullMat(file, ...)
readStackVec(file, ...)
readLowTriMat(file, no.var, ...)
```

### Arguments

file	File name of the data.
no.var	Number of variables in the data.
...	Further arguments to be passed to <code>scan</code> for <code>readLowTriMat</code> and to <code>read.table</code> for <code>readFullMat</code> and <code>readStackVec</code> .

### Value

A list of correlation/covariance matrices.

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### Examples

```
## Not run:
## Write two full correlation matrices into a file named "fullmat.dat".
## x2 is missing in the second matrix.
## The content of "fullmat.dat" is
# 1.0 0.3 0.4
# 0.3 1.0 0.5
# 0.4 0.5 1.0
# 1.0 NA 0.4
# NA NA NA
# 0.4 NA 1.0
cat("1.0 0.3 0.4\n0.3 1.0 0.5\n0.4 0.5 1.0
1.0 NA 0.4\nNA NA NA\n0.4 NA 1.0",
file="fullmat.dat", sep="")

## Read the correlation matrices
my.full <- readFullMat("fullmat.dat")

my.full
```

```

# `$1`
#   x1 x2 x3
# x1 1.0 0.3 0.4
# x2 0.3 1.0 0.5
# x3 0.4 0.5 1.0
#
# `$2`
#   x1 x2 x3
# x1 1.0 NA 0.4
# x2 NA NA NA
# x3 0.4 NA 1.0

## Write two lower triangle correlation matrices into a file named "lowertriangle.dat".
## x2 is missing in the second matrix.
## The content of "lowertriangle.dat" is
# 1.0
# 0.3 1.0
# 0.4 0.5 1.0
# 1.0
# NA NA
# 0.4 NA 1.0
cat("1.0\n0.3 1.0\n0.4 0.5 1.0\n1.0\nNA NA\n0.4 NA 1.0",
    file="lowertriangle.dat", sep="")

## Read the lower triangle correlation matrices
my.lowertri <- readLowTriMat(file = "lowertriangle.dat", no.var = 3)

my.lowertri
# `$1`
#   x1 x2 x3
# x1 1.0 0.3 0.4
# x2 0.3 1.0 0.5
# x3 0.4 0.5 1.0
#
# `$2`
#   x1 x2 x3
# x1 1.0 NA 0.4
# x2 NA NA NA
# x3 0.4 NA 1.0

## Write two vectors of correlation coefficients based on
## column major into a file named "stackvec.dat".
## x2 is missing in the second matrix.
## The content of "stackvec.dat" is
# 1.0 0.3 0.4 1.0 0.5 1.0
# 1.0 NA 0.4 NA NA 1.0
cat("1.0 0.3 0.4 1.0 0.5 1.0\n1.0 NA 0.4 NA NA 1.0\n",
    file="stackvec.dat", sep="")

my.vec <- readStackVec("stackvec.dat")

my.vec
# `$1`

```

```

#      x1  x2  x3
# x1  1.0  0.3  0.4
# x2  0.3  1.0  0.5
# x3  0.4  0.5  1.0
#
# `$2`
#      x1  x2  x3
# x1  1.0  NA  0.4
# x2  NA  NA  NA
# x3  0.4  NA  1.0

## End(Not run)

```

---

reml	<i>Estimate Variance Components with Restricted (Residual) Maximum Likelihood Estimation</i>
------	--

---

## Description

It estimates the variance components of random-effects in univariate and multivariate meta-analysis with restricted (residual) maximum likelihood (REML) estimation method.

## Usage

```

reml(y, v, x, data, RE.constraints = NULL, RE.startvalues = 0.1,
      RE.lbound = 1e-10, intervals.type = c("z", "LB"),
      model.name="Variance component with REML",
      suppressWarnings = TRUE, silent = TRUE, run = TRUE, ...)

```

## Arguments

y	A vector of effect size for univariate meta-analysis or a $k \times p$ matrix of effect sizes for multivariate meta-analysis where $k$ is the number of studies and $p$ is the number of effect sizes.
v	A vector of the sampling variance of the effect size for univariate meta-analysis or a $k \times p^*$ matrix of the sampling covariance matrix of the effect sizes for multivariate meta-analysis where $p^* = p(p + 1)/2$ . It is arranged by column major as used by <a href="#">vech</a> .
x	A predictor or a $k \times m$ matrix of predictors where $m$ is the number of predictors.
data	An optional data frame containing the variables in the model.
RE.constraints	A $p \times p$ matrix specifying the variance components of the random effects. If the input is not a matrix, it is converted into a matrix by <code>as.matrix()</code> . The default is that all covariance/variance components are free. The format of this matrix follows <a href="#">as.mxMatrix</a> . Elements of the variance components can be constrained equally by using the same labels. If a zero matrix is specified, it becomes a fixed-effects meta-analysis.

RE.startvalues	A vector of $p$ starting values on the diagonals of the variance component of the random effects. If only one scalar is given, it will be repeated across the diagonals. Starting values for the off-diagonals of the variance component are all 0. A $p \times p$ symmetric matrix of starting values is also accepted.
RE.lbound	A vector of $p$ lower bounds on the diagonals of the variance component of the random effects. If only one scalar is given, it will be repeated across the diagonals. Lower bounds for the off-diagonals of the variance component are set at NA. A $p \times p$ symmetric matrix of the lower bounds is also accepted.
intervals.type	Either z (default if missing) or LB. If it is z, it calculates the 95% Wald confidence intervals (CIs) based on the z statistic. If it is LB, it calculates the 95% likelihood-based CIs on the parameter estimates. Note that the z values and their associated p values are based on the z statistic. They are not related to the likelihood-based CIs.
model.name	A string for the model name in <code>mxModel</code> .
suppressWarnings	Logical. If TRUE, warnings are suppressed. Argument to be passed to <code>mxRun</code> .
silent	Logical. Argument to be passed to <code>mxRun</code>
run	Logical. If FALSE, only return the mx model without running the analysis.
...	Futher arguments to be passed to <code>mxRun</code>

### Details

Restricted (residual) maximum likelihood obtains the parameter estimates on the transformed data that do not include the fixed-effects parameters. A transformation matrix  $M = I - X(X'X)^{-1}X'$  is created based on the design matrix  $X$  which is just a column vector when there is no predictor in  $x$ . The last  $N$  redundant rows of  $M$  is removed where  $N$  is the rank of  $X$ . After pre-multiplying by  $M$  on  $y$ , the parameters of fixed-effects are removed from the model. Thus, only the parameters of random-effects are estimated.

An alternative but equivalent approach is to minimize the  $-2 \times \log$ -likelihood function:

$$\log(\det |V + T^2|) + \log(\det |X'(V + T^2)^{-1}X|) + (y - X\hat{\alpha})'(V + T^2)^{-1}(y - X\hat{\alpha})$$

where  $V$  is the known conditional sampling covariance matrix of  $y$ ,  $T^2$  is the variance component of the random effects, and  $\hat{\alpha} = (X'(V + T^2)^{-1}X)^{-1}X'(V + T^2)^{-1}y$ . `reml()` minimizes the above likelihood function to obtain the parameter estimates.

### Value

An object of class `reml` with a list of

call	Object returned by <code>match.call</code>
data	A data matrix of $y$ , $v$ and $x$
no.y	No. of effect sizes
no.x	No. of predictors
miss.vec	A vector indicating missing data. Studies will be removed before the analysis if they are TRUE
mx.fit	A fitted object returned from <code>mxRun</code>

**Note**

reml is more computational intensive than meta. Moreover, reml is more likely to encounter errors during optimization. Since a likelihood function is directly employed to obtain the parameter estimates, there is no number of studies and number of observed statistics returned by `mxRun`. Ad-hoc steps are used to modify `mx.fit@runstate$objectives[[1]]@numObs` and `mx.fit@runstate$objectives[[1]]@numSta`

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

- Cheung, M. W.-L. (2013). Implementing restricted maximum likelihood estimation in structural equation models. *Structural Equation Modeling*, **20**(1), 157-167.
- Mehta, P. D., & Neale, M. C. (2005). People Are Variables Too: Multilevel Structural Equations Modeling. *Psychological Methods*, **10**(3), 259-284.
- Searle, S. R., Casella, G., & McCulloch, C. E. (1992). *Variance components*. New York: Wiley.
- Viechtbauer, W. (2005). Bias and efficiency of meta-analytic variance estimators in the random-effects model. *Journal of Educational and Behavioral Statistics*, **30**(3), 261-293.

**See Also**

[meta](#), [reml3](#), [Hox02](#), [Berkey98](#)

---

reml3

*Estimate Variance Components in Three-Level Univariate Meta-Analysis with Restricted (Residual) Maximum Likelihood Estimation*

---

**Description**

It estimates the variance components of random-effects in three-level univariate meta-analysis with restricted (residual) maximum likelihood (REML) estimation method.

**Usage**

```
reml3(y, v, cluster, x, data, RE2.startvalue=0.1, RE2.lbound=1e-10,
      RE3.startvalue=RE2.startvalue, RE3.lbound=RE2.lbound, RE.equal=FALSE,
      intervals.type=c("z", "LB"), model.name="Variance component with REML",
      suppressWarnings=TRUE, silent=TRUE, run=TRUE, ...)
```

**Arguments**

<code>y</code>	A vector of $k$ studies of effect size.
<code>v</code>	A vector of $k$ studies of sampling variance.
<code>cluster</code>	A vector of $k$ characters or numbers indicating the clusters.
<code>x</code>	A predictor or a $k \times m$ matrix of level-2 and level-3 predictors where $m$ is the number of predictors.
<code>data</code>	An optional data frame containing the variables in the model.
<code>RE2.startvalue</code>	Starting value for the level-2 variance.
<code>RE2.lbound</code>	Lower bound for the level-2 variance.
<code>RE3.startvalue</code>	Starting value for the level-3 variance.
<code>RE3.lbound</code>	Lower bound for the level-3 variance.
<code>RE.equal</code>	Logical. Whether the variance components at level-2 and level-3 are constrained equally.
<code>intervals.type</code>	Either <code>z</code> (default if missing) or <code>LB</code> . If it is <code>z</code> , it calculates the 95% Wald confidence intervals (CIs) based on the $z$ statistic. If it is <code>LB</code> , it calculates the 95% likelihood-based CIs on the parameter estimates. Note that the $z$ values and their associated $p$ values are based on the $z$ statistic. They are not related to the likelihood-based CIs.
<code>model.name</code>	A string for the model name in <code>mxModel</code> .
<code>suppressWarnings</code>	Logical. If <code>TRUE</code> , warnings are suppressed. Argument to be passed to <code>mxRun</code> .
<code>silent</code>	Logical. Argument to be passed to <code>mxRun</code>
<code>run</code>	Logical. If <code>FALSE</code> , only return the <code>mx</code> model without running the analysis.
<code>...</code>	Futher arguments to be passed to <code>mxRun</code>

**Details**

Restricted (residual) maximum likelihood obtains the parameter estimates on the transformed data that do not include the fixed-effects parameters. A transformation matrix  $M = I - X(X'X)^{-1}X'$  is created based on the design matrix  $X$  which is just a column vector when there is no predictor in  $x$ . The last  $N$  redundant rows of  $M$  is removed where  $N$  is the rank of  $X$ . After pre-multiplying by  $M$  on  $y$ , the parameters of fixed-effects are removed from the model. Thus, only the parameters of random-effects are estimated.

An alternative but equivalent approach is to minimize the  $-2 \times \log$ -likelihood function:

$$\log(\det |V + T^2|) + \log(\det |X'(V + T^2)^{-1}X|) + (y - X\hat{\alpha})'(V + T^2)^{-1}(y - X\hat{\alpha})$$

where  $V$  is the known conditional sampling covariance matrix of  $y$ ,  $T^2$  is the variance component combining level-2 and level-3 random effects, and  $\hat{\alpha} = (X'(V + T^2)^{-1}X)^{-1}X'(V + T^2)^{-1}y$ . `reml()` minimizes the above likelihood function to obtain the parameter estimates.

**Value**

An object of class `reml` with a list of

<code>call</code>	Object returned by <a href="#">match.call</a>
<code>data</code>	A data matrix of <code>y</code> , <code>v</code> and <code>x</code>
<code>mx.fit</code>	A fitted object returned from <a href="#">mxRun</a>

**Note**

`reml` is more computational intensive than `meta`. Moreover, `reml` is more likely to encounter errors during optimization. Since a likelihood function is directly employed to obtain the parameter estimates, there is no number of studies and number of observed statistics returned by [mxRun](#). Ad-hoc steps are used to modify `mx.fit@runstate$objectives[[1]]@numObs` and `mx.fit@runstate$objectives[[1]]@numSta`

**Author(s)**

Mike W.-L. Cheung <[mikewlcheung@nus.edu.sg](mailto:mikewlcheung@nus.edu.sg)>

**References**

- Cheung, M. W.-L. (2013). Implementing restricted maximum likelihood estimation in structural equation models. *Structural Equation Modeling*, **20**(1), 157-167.
- Cheung, M. W.-L. (2014). Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods*, **19**, 211-229.
- Mehta, P. D., & Neale, M. C. (2005). People Are Variables Too: Multilevel Structural Equations Modeling. *Psychological Methods*, **10**(3), 259-284.
- Searle, S. R., Casella, G., & McCulloch, C. E. (1992). *Variance components*. New York: Wiley.

**See Also**

[meta3](#), [reml](#), [Cooper03](#), [Bornmann07](#)

---

rerun *Rerun models via mxTryHard()*

---

**Description**

It reruns models via `mxTryHard()`.

**Usage**

```
rerun(object, ...)
```

**Arguments**

**object** An object of either class `tssem1FEM`, class `tssem1REM`, class `wls`, class `meta` or class `reml`.

**...** Further arguments to be passed to `mxTryHard`

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**Examples**

```
## Not run:
random1 <- tssem1(Digman97$data, Digman97$n, method="REM", RE.type="Diag")
random1_rerun <- rerun(random1)
summary(random1_rerun)

## End(Not run)
```

---

Roorda11

*Studies on Students' School Engagement and Achievement Reported by Roorda et al. (2011)*

---

**Description**

The data set includes 45 studies on the influence of affective teacher-student relationships on students' school engagement and achievement reported by Roorda et al. (2011).

**Usage**

```
data(Roorda11)
```

**Details**

The variables are:

**data** A list of 45 studies of correlation matrices. The variables are *pos* (positive teacher-student relations), *neg* (negative teacher-student relations), *enga* (student engagement), and *achiev* (student achievement).

**n** A vector of sample sizes

**SES** A vector of average socio-economic status (SES) of the samples

**Source**

Roorda, D. L., Koomen, H. M. Y., Spilt, J. L., & Oort, F. J. (2011). The influence of affective teacher-student relationships on students' school engagement and achievement a meta-analytic approach. *Review of Educational Research*, 81(4), 493-529.



## References

Jak, S., & Cheung, M. W.-L. (2015). Addressing heterogeneity in meta-analytic structural equation modeling using subgroup analysis. *Manuscript in preparation*.

## Examples

```
## Not run:

## Random-effects model: First stage analysis
random1 <- tssem1(my.df = Roorda11$data, n = Roorda11$n, method = "REM", RE.type = "Diag")
summary(random1)

varnames <- c("pos", "neg", "enga", "achiev")

## Prepare a regression model using create.mxMatrix()
A <- create.mxMatrix(c(0,0,0,0,
                      0,0,0,0,
                      "0.1*b31", "0.1*b32", 0, 0,
                      0, 0, "0.1*b43", 0),
                    type = "Full", nrow = 4, ncol = 4, byrow = TRUE,
                    name = "A", as.mxMatrix = FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(A) <- list(varnames, varnames)
A

S <- create.mxMatrix(c(1,
                      ".5*p21", 1,
                      0, 0, "0.6*p33",
                      0, 0, 0, "0.6*p44"),
                    type="Symm", byrow = TRUE,
                    name="S", as.mxMatrix = FALSE)

## This step is not necessary but it is useful for inspecting the model.
dimnames(S) <- list(varnames, varnames)
S

## Random-effects model: Second stage analysis
random2 <- tssem2(random1, Amatrix=A, Smatrix=S, diag.constraints=TRUE,
                  intervals="LB")
summary(random2)

## Load the library
library("semPlot")

## Convert the model to semPlotModel object
my.plot <- meta2semPlot(random2)

## Plot the parameter estimates
semPaths(my.plot, whatLabels="est", nCharNodes=10, color="green")

## End(Not run)
```

summary

*Summary Method for tssem1, wls, meta and meta3X Objects***Description**

It summaries results for either class `tssem1FEM`, class `tssem1FEM.cluster`, class `tssem1REM`, class `wls`, class `wls.cluster`, class `meta`, class `meta3X` or class `reml`.

**Usage**

```
## S3 method for class 'tssem1FEM'
summary(object, ...)
## S3 method for class 'tssem1FEM.cluster'
summary(object, ...)
## S3 method for class 'tssem1REM'
summary(object, ...)
## S3 method for class 'wls'
summary(object, df.adjustment=0, R=50, ...)
## S3 method for class 'wls.cluster'
summary(object, df.adjustment=0, R=50, ...)
## S3 method for class 'meta'
summary(object, homoStat=TRUE, ...)
## S3 method for class 'meta3X'
summary(object, allX=FALSE, ...)
## S3 method for class 'reml'
summary(object, ...)
## S3 method for class 'tssem1FEM'
print.summary(x, ...)
## S3 method for class 'wls'
print.summary(x, ...)
## S3 method for class 'meta'
print.summary(x, ...)
## S3 method for class 'meta3X'
print.summary(x, ...)
## S3 method for class 'reml'
print.summary(x, ...)
```

**Arguments**

<code>object</code>	An object returned from either class <code>tssem1FEM</code> , class <code>tssem1FEM.cluster</code> , class <code>tssem1REM</code> , class <code>wls</code> , class <code>wls.cluster</code> , class <code>meta</code> , class <code>meta3X</code> or class <code>reml</code> .
<code>x</code>	An object returned from either class <code>summary.tssem1FEM</code> , class <code>summary.tssem1FEM.cluster</code> , class <code>summary.wls</code> , class <code>summary.meta</code> , class <code>summary.meta3X</code> or class <code>summary.reml</code> .
<code>homoStat</code>	Logical. Whether to conduct a homogeneity test on the effect sizes.
<code>allX</code>	Logical. Whether to report the predictors and the auxiliary variables.

df.adjustment	Numeric. Adjust the degrees of freedom manually. It may be necessary if the df calculated is incorrect when diag.constraints=TRUE.
R	Positive integer. The number of parameter bootstrap replicates when diag.constraints=TRUE.
...	Further arguments to be passed to <a href="#">printCoefmat</a>

### Note

If the OpenMx status1 is either 0 or 1, the estimation is considered fine. If the OpenMx status1 is other values, it indicates estimation problems. Users should refer to <http://openmx.psyc.virginia.edu/wiki/errors> for more details.

### Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

### See Also

[tssem1](#), [wls](#), [meta](#), [reml](#)

---

 tssem1

---

*First Stage of the Two-Stage Structural Equation Modeling (TSSEM)*


---

### Description

It conducts the first stage analysis of TSSEM by pooling correlation/covariance matrices. `tssem1FEM()` and `tssem1REM()` use fixed- and random-effects models, respectively. `tssem1()` is a wrapper of these functions.

### Usage

```
tssem1(my.df, n, method=c("FEM","REM"), cor.analysis = TRUE, cluster=NULL,
       RE.type=c("Symm", "Diag", "Zero", "User"), RE.startvalues=0.1,
       RE.lbound=1e-10, RE.constraints=NULL, I2="I2q",
       acov=c("individual", "unweighted", "weighted"),
       model.name=NULL, suppressWarnings=TRUE, silent=TRUE, run=TRUE, ...)
tssem1FEM(my.df, n, cor.analysis=TRUE, model.name=NULL,
          cluster=NULL, suppressWarnings=TRUE, silent=TRUE, run=TRUE, ...)
tssem1REM(my.df, n, cor.analysis=TRUE, RE.type=c("Symm", "Diag", "Zero","User"),
          RE.startvalues=0.1, RE.lbound=1e-10, RE.constraints=NULL,
          I2="I2q", acov=c("individual", "unweighted", "weighted"),
          model.name=NULL, suppressWarnings=TRUE,
          silent=TRUE, run=TRUE, ...)
```

**Arguments**

<code>my.df</code>	A list of correlation/covariance matrices
<code>n</code>	A vector of sample sizes
<code>method</code>	Either "FEM" (default if missing) or "REM". If it is "FEM", fixed-effects meta-analysis will be applied. If it is "REM", random-effects meta-analysis will be applied.
<code>cor.analysis</code>	Logical. The output is either a pooled correlation or a covariance matrix.
<code>cluster</code>	A vector of characters or numbers indicating the clusters. Analyses will be conducted for each cluster. It will be ignored when <code>method="REM"</code> .
<code>RE.type</code>	Either "Symm", "Diag", "Zero" or "User". If it is "Symm" (default if missing), a symmetric matrix is used for the random effects on the covariances among the correlation (or covariance) vectors. If it is "Diag", a diagonal matrix is used for the random effects meaning that the random effects are independent. If it is "Zero", there is no random effects which is similar to the conventional Generalized Least Squares (GLS) approach to fixed-effects analysis. "User", user has to specify the variance component via the <code>RE.constraints</code> argument. This argument will be ignored when <code>method="FEM"</code> .
<code>RE.startvalues</code>	Starting values on the diagonals of the variance component of the random effects. It will be ignored when <code>method="FEM"</code> .
<code>RE.lbound</code>	Lower bounds on the diagonals of the variance component of the random effects. It will be ignored when <code>method="FEM"</code> .
<code>RE.constraints</code>	A $p^* \times p^*$ matrix specifying the variance components of the random effects, where $p^*$ is the number of effect sizes. If the input is not a matrix, it is converted into a matrix by <code>as.matrix()</code> . The default is that all covariance/variance components are free. The format of this matrix follows <code>as.mxMatrix</code> . Elements of the variance components can be constrained equally by using the same labels. If a zero matrix is specified, it becomes a fixed-effects meta-analysis.
<code>I2</code>	Possible options are "I2q", "I2hm" and "I2am". They represent the I2 calculated by using a typical within-study sampling variance from the Q statistic, the harmonic mean and the arithmetic mean of the within-study sampling variances (Xiong, Miller, & Morris, 2010). More than one options are possible. If <code>intervals.type="LB"</code> , 95% confidence intervals on the heterogeneity indices will be constructed.
<code>acov</code>	If it is individual (the default), the sampling variance covariance matrices are calculated based on individual correlation/covariance matrix. If it is either unweighted or weighted, the average correlation/covariance matrix is calculated based on the unweighted or weighted mean with the sample sizes. The average correlation/covariance matrix is used to calculate the sampling variance covariance matrices. This argument is ignored with the <code>method="FEM"</code> argument.
<code>model.name</code>	A string for the model name in <code>mxModel</code> .
<code>suppressWarnings</code>	Logical. If TRUE, warnings are suppressed. Argument to be passed to <code>mxRun</code> .
<code>silent</code>	Logical. Argument to be passed to <code>mxRun</code>

run Logical. If FALSE, only return the mx model without running the analysis.  
 ... Further arguments to be passed to [mxRun](#)

### Value

Either an object of class `tssem1FEM` for fixed-effects TSSEM, an object of class `tssem1FEM.cluster` for fixed-effects TSSEM with `cluster` argument, or an object of class `tssem1REM` for random-effects TSSEM.

### Note

If the `cluster` argument is used, it returns a list of results based for each cluster.

### Author(s)

Mike W.-L. Cheung <[mikewlcheung@nus.edu.sg](mailto:mikewlcheung@nus.edu.sg)>

### References

- Cheung, M. W.-L. (2014). Fixed- and random-effects meta-analytic structural equation modeling: Examples and analyses in R. *Behavior Research Methods*, **46**, 29-40.
- Cheung, M. W.-L. (2013). Multivariate meta-analysis as structural equation models. *Structural Equation Modeling*, **20**, 429-454.
- Cheung, M. W.-L., & Chan, W. (2005). Meta-analytic structural equation modeling: A two-stage approach. *Psychological Methods*, **10**, 40-64.
- Cheung, M. W.-L., & Chan, W. (2009). A two-stage approach to synthesizing covariance matrices in meta-analytic structural equation modeling. *Structural Equation Modeling*, **16**, 28-53.

### See Also

[wls](#), [Cheung09](#), [Becker92](#), [Digman97](#), [issp89](#), [issp05](#)

### Description

It extracts the variance-covariance matrix of the parameter estimates from either `tssem1FEM`, `tssem1FEM.cluster`, `tssem1REM`, `wls`, `wls.cluster`, `meta`, `meta3X`, `reml` or `MxRAMModel` objects.

**Usage**

```
## S3 method for class 'tssem1FEM'
vcov(object, ...)
## S3 method for class 'tssem1FEM.cluster'
vcov(object, ...)
## S3 method for class 'tssem1REM'
vcov(object, select = c("all", "fixed", "random"), ...)
## S3 method for class 'wls'
vcov(object, R=50, ...)
## S3 method for class 'wls.cluster'
vcov(object, R=50, ...)
## S3 method for class 'meta'
vcov(object, select = c("all", "fixed", "random"), ...)
## S3 method for class 'meta3X'
vcov(object, select = c("all", "fixed", "random", "allX"), ...)
## S3 method for class 'reml'
vcov(object, ...)
## S3 method for class 'MxRAMModel'
vcov(object, ...)
```

**Arguments**

object	An object returned from either class <code>tssem1FEM</code> , class <code>tssem1FEM.cluster</code> , class <code>tssem1REM</code> , class <code>wls</code> , class <code>wls.cluster</code> , class <code>meta</code> , class <code>reml</code> or class <code>MxRAMModel</code>
select	Select all for both fixed- and random-effects parameters, fixed for the fixed-effects parameters or random for the random-effects parameters. For <code>meta3X</code> objects, <code>allX</code> is used to extract all parameters including the predictors and auxiliary variables.
R	Positive integer. The number of parameter bootstrap replicates when <code>diag.constraints=TRUE</code> .
...	Further arguments; currently none is used

**Value**

A variance-covariance matrix of the parameter estimates.

**Note**

`vcov` returns NA when `diag.constraints=TRUE` argument is used in `wls` objects.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**See Also**

[tssem1](#), [wls](#), [meta](#), [reml](#)

## Examples

```
## Random-effects meta-analysis
model1 <- meta(y=yi, v=vi, data=Hox02)
vcov(model1)

## Fixed-effects only
vcov(model1, select="fixed")

## Random-effects only
vcov(model1, select="random")
```

---

vec2symMat

*Convert a Vector into a Symmetric Matrix*

---

## Description

It converts a vector into a symmetric matrix.

## Usage

```
vec2symMat(x, diag = TRUE, byrow = FALSE)
```

## Arguments

x	A vector of numerics or characters
diag	Logical. If it is TRUE (the default), the diagonals of the created matrix are replaced by elements of x; otherwise, the diagonals of the created matrix are replaced by "1".
byrow	Logical. If it is FALSE (the default), the created matrix is filled by columns; otherwise, the matrix is filled by rows.

## Value

A symmetric square matrix based on column major

## Author(s)

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

## See Also

[matrix2bdiag](#)

**Examples**

```
vec2symMat(1:6)
#      [,1] [,2] [,3]
# [1,]  1   2   3
# [2,]  2   4   5
# [3,]  3   5   6

vec2symMat(1:6, diag=FALSE)
#      [,1] [,2] [,3] [,4]
# [1,]  1   1   2   3
# [2,]  1   1   4   5
# [3,]  2   4   1   6
# [4,]  3   5   6   1

vec2symMat(letters[1:6])
#      [,1] [,2] [,3]
# [1,] "a"  "b"  "c"
# [2,] "b"  "d"  "e"
# [3,] "c"  "e"  "f"
```

wls

*Conduct a Correlation/Covariance Structure Analysis with WLS***Description**

It fits a correlation or covariance structure with weighted least squares (WLS) estimation method where the inverse of the asymptotic covariance matrix is used as the weight matrix. `tssem2` conducts the second stage analysis of the two-stage structural equation modeling (TSSEM). `tssem2` is a wrapper of `wls`.

**Usage**

```
wls(Cov, asyCov, n, Amatrix=NULL, Smatrix=NULL, Fmatrix=NULL,
    diag.constraints=FALSE, cor.analysis=TRUE, intervals.type=c("z", "LB"),
    mx.algebras=NULL, model.name=NULL, suppressWarnings=TRUE,
    silent=TRUE, run=TRUE, ...)
tssem2(tssem1.obj, Amatrix=NULL, Smatrix=NULL, Fmatrix=NULL,
    diag.constraints=FALSE, intervals.type=c("z", "LB"), mx.algebras=NULL,
    model.name=NULL, suppressWarnings=TRUE, silent=TRUE, run=TRUE, ...)
```

**Arguments**

<code>tssem1.obj</code>	An object of either class <code>tssem1FEM</code> , class <code>tssem1FEM.cluster</code> or class <code>tssem1REM</code> returned from <code>tssem1()</code>
<code>Cov</code>	A $p \times p$ sample correlation/covariance matrix where $p$ is the number of variables.
<code>asyCov</code>	A $p^* \times p^*$ asymptotic sampling covariance matrix of either <code>vechs</code> ( <code>Cov</code> ) or <code>vech</code> ( <code>Cov</code> ) where $p^* = p(p - 1)/2$ for correlation matrix and $p^* = p(p + 1)/2$ for covariance matrix.



n	Sample size.
Amatrix	An asymmetric matrix in the RAM approach with <code>MxMatrix-class</code> . If it is NULL, a matrix of zero will be created. If it is a matrix, it will be converted into <code>MxMatrix-class</code> by the <code>as.mxMatrix</code> function.
Smatrix	A symmetric matrix in the RAM approach with <code>MxMatrix-class</code> . If it is a matrix, it will be converted into <code>MxMatrix-class</code> by the <code>as.mxMatrix</code> function.
Fmatrix	A filter matrix in the RAM approach with <code>MxMatrix-class</code> . If it is NULL (the default), an identity matrix with the same dimensions of Cov will be created. If it is a matrix, it will be converted into <code>MxMatrix-class</code> by the <code>as.mxMatrix</code> function. It is not required when there is no latent variable.
diag.constraints	Logical. This argument is ignored when <code>cor.analysis=FALSE</code> . If <code>diag.constraints=TRUE</code> , the diagonals of the model implied matrix will be constrained at 1 by nonlinear constraints. The drawback is that standard error will not be generated. Parametric bootstrap is used to estimate the standard error by drawing samples from $\mathcal{N}(\text{vech}(Cov), \text{asyCov})$ for covariance analysis and $\mathcal{N}(\text{vechs}(Cov), \text{asyCov})$ for correlation analysis while <code>asyCov</code> is treated as fixed. This process is computational intensive. A better approach is to request likelihood-based confidence intervals (CIs) by specifying <code>intervals.type="LB"</code> . If <code>diag.constraints=FALSE</code> and <code>cor.analysis=TRUE</code> , the diagonals are automatically constrained as ones by treating the error variances as computed values rather than as parameters. Since the error variances are not parameters, they are not reported.
cor.analysis	Logical. Analysis of correlation or covariance structure. If <code>cor.analysis=TRUE</code> , <code>vechs</code> is used to vectorize S; otherwise, <code>vech</code> is used to vectorize S.
intervals.type	Either z (default if missing) or LB. If it is z, it calculates the 95% Wald CIs based on the z statistic. If it is LB, it calculates the 95% likelihood-based CIs on the parameter estimates. Please note that the z values and their associated p values are based on the z statistic. They are not related to the likelihood-based CIs.
mx.algebras	A list of <code>mxMatrix</code> or <code>mxAlgebra</code> objects on the <code>Amatrix</code> , <code>Smatrix</code> and <code>Fmatrix</code> . It can be used to define new functions of parameters and their LBCIs. For example, if the regression coefficients to calculate an indirect effect are stored in <code>A[1,2]</code> and <code>A[1,3]</code> , we may define <code>list(ind=mxAlgebra(Amatrix[1,2]*Amatrix[1,3], name="ind")</code> . See the examples in <a href="#">Becker92</a> and <a href="#">Hunter83</a> . It should be noted that <code>Fmatrix</code> , <code>Amatrix</code> , <code>Smatrix</code> , <code>Iden</code> (a $p \times p$ identity matrix), <code>sampleS</code> (sample correlation or covariance matrix), <code>impliedS1</code> , <code>impliedS</code> (model implied correlation or covariance matrix), <code>vecS</code> , <code>invAcov</code> , <code>obj</code> , <code>One</code> , <code>select</code> and <code>constraint</code> and <code>Ematrix</code> (computed error variances when <code>diag.constraints=FALSE</code> ) have been defined internally. You should not create new matrices using these names.
model.name	A string for the model name in <code>mxModel</code> . If it is missing, the default is "TSSEM2 (or WLS) Analysis of Correlation Structure" for <code>cor.analysis=TRUE</code> and "TSSEM2 (or WLS) Analysis of Covariance Structure" for <code>cor.analysis=FALSE</code> .
suppressWarnings	Logical. If TRUE, warnings are suppressed. Argument to be passed to <code>mxRun</code> .
silent	Logical. Argument to be passed to <code>mxRun</code>
run	Logical. If FALSE, only return the mx model without running the analysis.
...	Futher arguments to be passed to <code>mxRun</code> .

**Value**

An object of class `wls` with a list of

<code>call</code>	The matched call
<code>Cov</code>	Input data of either a covariance or correlation matrix
<code>asyCov</code>	Asymptotic covariance matrix of the input data
<code>noObservedStat</code>	Number of observed statistics
<code>n</code>	Sample size
<code>cor.analysis</code>	logical
<code>noConstraints</code>	Number of constraints imposed on S
<code>indepModelChisq</code>	Chi-square statistic of the independent model returned by <code>.indepwlsChisq</code>
<code>indepModelDf</code>	Degrees of freedom of the independent model returned by <code>.indepwlsChisq</code>
<code>mx.fit</code>	A fitted object returned from <code>mxRun</code>

**Note**

If the input is a list of `tssem1.obj`, it returns a list of results for each cluster.

**Author(s)**

Mike W.-L. Cheung <mikewlcheung@nus.edu.sg>

**References**

- Bentler, P.M., & Savalei, V. (2010). Analysis of correlation structures: current status and open problems. In Kolenikov, S., Thombs, L., & Steinley, D. (Eds.). *Recent Methodological Developments in Social Science Statistics* (pp. 1-36). Hoboken, NJ: Wiley.
- Cheung, M. W.-L. (2010). Fixed-effects meta-analyses as multiple-group structural equation models. *Structural Equation Modeling*, **17**, 481-509.
- Cheung, M. W.-L. (2014). Fixed- and random-effects meta-analytic structural equation modeling: Examples and analyses in R. *Behavior Research Methods*, **46**, 29-40.
- Cheung, M. W.-L., & Chan, W. (2005). Meta-analytic structural equation modeling: A two-stage approach. *Psychological Methods*, **10**, 40-64.
- Cheung, M. W.-L., & Chan, W. (2009). A two-stage approach to synthesizing covariance matrices in meta-analytic structural equation modeling. *Structural Equation Modeling*, **16**, 28-53.
- Joreskog, K. G., Sorbom, D., Du Toit, S., & Du Toit, M. (1999). *LISREL 8: New Statistical Features*. Chicago: Scientific Software International.
- McArdle, J. J., & MacDonald, R. P. (1984). Some algebraic properties of the Reticular Action Model for moment structures. *British Journal of Mathematical and Statistical Psychology*, **37**, 234-251.

**See Also**

[tssem1](#), [Becker92](#), [Digman97](#), [Hunter83](#), [issp89](#), [issp05](#)

**Examples**

```

#### Analysis of correlation structure
R1 <- matrix(c(1.00, 0.22, 0.24, 0.18,
              0.22, 1.00, 0.30, 0.22,
              0.24, 0.30, 1.00, 0.24,
              0.18, 0.22, 0.24, 1.00), ncol=4, nrow=4)

n <- 1000
acovR1 <- asyCov(R1, n)

## One-factor CFA model
(A1 <- cbind(matrix(0, nrow=5, ncol=4),
            matrix(c("0.2*a1", "0.2*a2", "0.2*a3", "0.2*a4", 0),
                  ncol=1)))

(S1 <- Diag(c("0.2*e1", "0.2*e2", "0.2*e3", "0.2*e4", 1)))

## The first 4 variables are observed while the last one is latent.
(F1 <- create.Fmatrix(c(1,1,1,1,0), name="F1"))
wls.fit1 <- wls(Cov=R1, asyCov=acovR1, n=n, Fmatrix=F1, Smatrix=S1, Amatrix=A1,
               cor.analysis=TRUE, intervals="LB")
summary(wls.fit1)

#### Multiple regression analysis
## Variables in R2: y, x1, x2
R2 <- matrix(c(1.00, 0.22, 0.24,
              0.22, 1.00, 0.30,
              0.24, 0.30, 1.00,
              0.18, 0.22, 0.24), ncol=3, nrow=3)
acovR2 <- asyCov(R2, n)

## A2: Regression coefficients
#   y x1 x2
# y  F T T
# x1 F F F
# x2 F F F
(A2 <- mxMatrix("Full", ncol=3, nrow=3, byrow=TRUE,
               free=c(FALSE, rep(TRUE, 2), rep(FALSE, 6)), name="A2"))

## S2: Covariance matrix of free parameters
#   y x1 x2
# y  T F F
# x1 F F F
# x2 F T F
(S2 <- mxMatrix("Symm", ncol=3, nrow=3, values=c(0.2,0,0,1,0.2,1),
               free=c(TRUE,FALSE,FALSE,FALSE,TRUE,FALSE), name="S2"))

## F may be ignored as there is no latent variable.
wls.fit2 <- wls(Cov=R2, asyCov=acovR2, n=n, Amatrix=A2, Smatrix=S2,
               cor.analysis=TRUE, intervals="LB")
summary(wls.fit2)

```

```
#### Analysis of covariance structure
R3 <- matrix(c(1.50, 0.22, 0.24, 0.18,
              0.22, 1.60, 0.30, 0.22,
              0.24, 0.30, 1.80, 0.24,
              0.18, 0.22, 0.24, 1.30), ncol=4, nrow=4)
n <- 1000
acovS3 <- asyCov(R3, n, cor.analysis=FALSE)

(A3 <- cbind(matrix(0, nrow=5, ncol=4),
             matrix(c("0.2*a1", "0.2*a2", "0.2*a3", "0.2*a4", 0), ncol=1)))

(S3 <- Diag(c("0.2*e1", "0.2*e2", "0.2*e3", "0.2*e4", 1)))

F3 <- c(TRUE, TRUE, TRUE, TRUE, FALSE)
(F3 <- create.Fmatrix(F3, name="F3", as.mxMatrix=FALSE))

wls.fit3 <- wls(Cov=R3, asyCov=acovS3, n=n, Amatrix=A3, Smatrix=S3,
               Fmatrix=F3, cor.analysis=FALSE)
summary(wls.fit3)
```

---

wvs94a

*Forty-four Studies from Cheung (2013)*


---

## Description

Between 1990 and 1993, 57,561 adults aged 18 and above from 42 nations were interviewed by local academic institutes in Eastern European nations and by professional survey organizations in other nations. Standardized mean difference (SMD) between males and females on life satisfaction and on life control in each country were calculated as the effect sizes. Positive values indicate that males have higher scores than females do.

## Usage

```
data(wvs94a)
```

## Details

The variables are:

**country** Country

**lifesat** SMD on life satisfaction

**lifecon** SMD on life control

**lifesat\_var** Sampling variance of lifesat

**inter\_cov** Sampling covariance between lifesat and lifecon

**lifecon\_var** Sampling variance of lifecon

**gnp** Gross National Product

## Source

World Values Study Group. (1994). World Values Survey, 1981-1984 and 1990-1993 [Computer file]. *Ann Arbor, MI: Inter-university Consortium for Political and Social Research*.

## References

Au, K., & Cheung, M. W.-L. (2004). Intra-cultural variation and job autonomy in 42 countries. *Organization Studies*, **25**, 1339-1362.

Cheung, M. W.-L. (2013). Multivariate meta-analysis as structural equation models. *Structural Equation Modeling*, **20**, 429-454.

## Examples

```
## Not run:
data(wvs94a)

## Random-effects model
random.ma1 <- meta(y=cbind(lifesat, lifecon),
                  v=cbind(lifesat_var, inter_cov, lifecon_var), data=wvs94a,
                  model.name="Random effects model")
summary(random.ma1)

## Random-effects model with both population effect sizes fixed at 0
random.ma2 <- meta(y=cbind(lifesat, lifecon),
                  v=cbind(lifesat_var, inter_cov, lifecon_var), data=wvs94a,
                  intercept.constraints=matrix(0, nrow=1, ncol=2),
                  model.name="Effect sizes are fixed at 0")
summary(random.ma2)

## Compare the nested models
anova(random.ma1, random.ma2)

## Fixed-effects model by fixing the variance component at 0
fixed.ma <- meta(y=cbind(lifesat, lifecon),
                v=cbind(lifesat_var, inter_cov, lifecon_var), data=wvs94a,
                RE.constraints=matrix(0, ncol=2, nrow=2),
                model.name="Fixed effects model")
summary(fixed.ma)

## Mixed-effects model
## gnp is divided by 10000 and centered by using
## scale(gnp/10000, scale=FALSE)
mixed.ma1 <- meta(y=cbind(lifesat, lifecon),
                 v=cbind(lifesat_var, inter_cov, lifecon_var),
                 x=scale(gnp/10000, scale=FALSE), data=wvs94a,
                 model.name="GNP as a predictor")
summary(mixed.ma1)

## Mixed-effects model with equal regression coefficients
mixed.ma2 <- meta(y=cbind(lifesat, lifecon),
                 v=cbind(lifesat_var, inter_cov, lifecon_var),
```

```

x=scale(gnp/10000, scale=FALSE), data=wvs94a,
coef.constraints=matrix(c("0.0*Eq_slope",
                          "0.0*Eq_slope"), nrow=2),
model.name="GNP as a predictor with equal slope")
summary(mixed.ma2)

## Compare the nested models
anova(mixed.ma1, mixed.ma2)

## Plot the multivariate effect sizes
plot(random.ma1, main="Estimated effect sizes and their 95% confidence ellipses",
      axis.label=c("Gender difference on life satisfaction",
                  "Gender difference on life control"))

## End(Not run)

```

wvs94b

*Forty-four Covariance Matrices on Life Satisfaction, Job Satisfaction,  
and Job Autonomy*

### Description

Between 1990 and 1993, 57,561 adults aged 18 and above from 42 nations were interviewed by local academic institutes in Eastern European nations and by professional survey organizations in other nations. The covariance matrices among Life Satisfaction, Job Satisfaction, and Job Autonomy were calculated.

### Usage

```
data(wvs94b)
```

### Details

The variables are:

**data** Covariance matrix among Life Satisfaction (LS), Job Satisfaction (JS), and Job Autonomy (JA)

**n** Sample size in the country

### Source

World Values Study Group. (1994). World Values Survey, 1981-1984 and 1990-1993 [Computer file]. *Ann Arbor, MI: Inter-university Consortium for Political and Social Research.*

### References

Au, K., & Cheung, M. W.-L. (2004). Intra-cultural variation and job autonomy in 42 countries. *Organization Studies*, **25**, 1339-1362.

Cheung, M.W.-L., & Cheung, S.-F. (in press). Random-effects models for meta-analytic structural equation modeling: Review, issues, and illustrations. *Research Synthesis Methods*.

**Examples**

```
## Not run:
data(wvs94b)

## Get the indirect and the direct effects and
## their sampling covariance matrices for each study
indirect1 <- indirectEffect(wvs94b$data, wvs94b$n)
indirect1

## Multivariate meta-analysis on the indirect and direct effects
indirect2 <- meta(indirect1[, c("ind_eff", "dir_eff")],
                 indirect1[, c("ind_var", "ind_dir_cov", "dir_var")])

summary(indirect2)

## End(Not run)
```

# Index

- \*Topic **compute effect sizes**
  - indirectEffect, 41
- \*Topic **datasets**
  - Aloe14, 4
  - BCG, 10
  - Becker09, 13
  - Becker83, 16
  - Becker92, 17
  - Becker94, 19
  - Berkey98, 21
  - Bornmann07, 22
  - Cheung00, 24
  - Cheung09, 26
  - Cooper03, 29
  - Digman97, 34
  - HedgesOlkin85, 36
  - Hox02, 38
  - Hunter83, 40
  - issp05, 44
  - issp89, 46
  - Jaramillo05, 48
  - Mak09, 50
  - Norton13, 58
  - Roorda11, 72
  - wvs94a, 84
  - wvs94b, 86
- \*Topic **meta-analysis**
  - asyCov, 8
  - homoStat, 37
  - meta, 52
  - meta3, 56
  - rem1, 67
  - rem13, 69
- \*Topic **methods**
  - anova, 5
  - coef, 27
  - meta2semPlot, 55
  - plot, 61
  - print, 64
  - summary, 74
  - vcov, 77
- \*Topic **package**
  - metaSEM-package, 3
- \*Topic **tssem**
  - meta2semPlot, 55
  - rerun, 71
  - tssem1, 75
  - wls, 80
- \*Topic **utilities**
  - as.mxMatrix, 7
  - bdiagMat, 11
  - bdiagRep, 12
  - create.Fmatrix, 30
  - create.mxMatrix, 31
  - Diag, 33
  - is.pd, 43
  - list2matrix, 49
  - matrix2bdiag, 51
  - pattern.n, 59
  - pattern.na, 60
  - readData, 65
  - vec2symMat, 79
- Aloe14, 4
- anova, 5
- as.mxMatrix, 7, 31, 53, 56, 57, 67, 76
- asyCov, 8, 43
- BCG, 10
- bdiagMat, 11, 13
- bdiagRep, 12, 12
- Becker09, 13, 55
- Becker83, 16
- Becker92, 17, 55, 77, 81, 82
- Becker94, 19
- Berkey98, 21, 55, 69
- Bornmann07, 22, 58, 71
- Cheung00, 24



- Cheung09, [26](#), [77](#)  
 coef, [27](#)  
 Cooper03, [29](#), [58](#), [71](#)  
 create.Fmatrix, [7](#), [30](#), [32](#)  
 create.mxMatrix, [7](#), [31](#), [31](#), [32](#)
- Diag, [33](#)  
 diag, [33](#)  
 Diag<- (Diag), [33](#)  
 Digman97, [34](#), [55](#), [77](#), [82](#)
- HedgesO1kin85, [36](#)  
 homoStat, [37](#)  
 Hox02, [38](#), [55](#), [69](#)  
 Hunter83, [40](#), [55](#), [81](#), [82](#)
- indirectEffect, [41](#)  
 is.pd, [43](#)  
 issp05, [44](#), [46](#), [77](#), [82](#)  
 issp89, [45](#), [46](#), [77](#), [82](#)
- Jaramillo05, [48](#)
- list2matrix, [49](#)
- Mak09, [50](#)  
 match.call, [54](#), [68](#), [71](#)  
 matrix2bdiag, [12](#), [13](#), [51](#), [79](#)  
 meta, [6](#), [9](#), [28](#), [38](#), [52](#), [61](#), [64](#), [69](#), [75](#), [78](#)  
 meta2semPlot, [55](#)  
 meta3, [6](#), [56](#), [71](#)  
 meta3X (meta3), [56](#)  
 metaSEM (metaSEM-package), [3](#)  
 metaSEM-package, [3](#)  
 mxAlgebra, [81](#)  
 mxCompare, [5](#), [6](#)  
 mxMatrix, [7](#), [32](#), [81](#)  
 mxModel, [53](#), [57](#), [68](#), [70](#), [76](#), [81](#)  
 mxRun, [9](#), [54](#), [57](#), [68–71](#), [76](#), [77](#), [81](#), [82](#)  
 mxTryHard, [72](#)
- nearPD, [43](#)  
 Norton13, [58](#)
- omxGetParameters, [28](#)
- par, [62](#), [63](#)  
 pattern.n, [59](#)  
 pattern.na, [60](#)  
 plot, [61](#)
- points, [62](#)  
 print, [64](#)  
 print.summary.meta (summary), [74](#)  
 print.summary.meta3X (summary), [74](#)  
 print.summary.reml (summary), [74](#)  
 print.summary.tssem1FEM (summary), [74](#)  
 print.summary.wls (summary), [74](#)  
 printCoefmat, [75](#)
- ramModel, [55](#)  
 read.table, [65](#)  
 readData, [65](#)  
 readFullMat (readData), [65](#)  
 readLowTriMat (readData), [65](#)  
 readStackVec (readData), [65](#)  
 reml, [6](#), [28](#), [55](#), [64](#), [67](#), [71](#), [75](#), [78](#)  
 reml3, [58](#), [69](#), [69](#)  
 rerun, [71](#)  
 Roorda11, [72](#)
- scan, [65](#)  
 summary, [6](#), [74](#)  
 summary.meta (summary), [74](#)  
 summary.meta3X (summary), [74](#)  
 summary.reml (summary), [74](#)  
 summary.tssem1FEM (summary), [74](#)  
 summary.tssem1REM (summary), [74](#)  
 summary.wls (summary), [74](#)
- tssem1, [28](#), [64](#), [75](#), [75](#), [78](#), [82](#)  
 tssem1FEM (tssem1), [75](#)  
 tssem1REM (tssem1), [75](#)  
 tssem2 (wls), [80](#)
- vcov, [77](#)  
 vec2symMat, [51](#), [52](#), [79](#)  
 vech, [37](#), [49](#), [53](#), [67](#), [80](#), [81](#)  
 vechs, [49](#), [80](#), [81](#)
- wls, [6](#), [28](#), [31](#), [64](#), [75](#), [77](#), [78](#), [80](#)  
 wvs94a, [55](#), [84](#)  
 wvs94b, [86](#)