CFA_Bifactor_NoCovariate

Contents

1 Load packages & set working directory & read in data

```
library(matrixcalc);library(MASS);library(Matrix)
```

```
## Warning:   'matrixcalc' R 4.3.1

## Warning:   'Matrix' R 4.3.1
```

```
library(coda);library(R2OpenBUGS);library(metaSEM)
```

```
## Warning:   'coda' R 4.3.1

## Warning:   'R2OpenBUGS' R 4.3.2

##      OpenMx

##
##    'OpenMx'

## The following objects are masked from 'package:Matrix':
##
##     %&%, expm

## The following object is masked from 'package:matrixcalc':
##
##     vech

## "SLSQP" is set as the default optimizer in OpenMx.

## mxOption(NULL, "Gradient algorithm") is set at "central".

## mxOption(NULL, "Optimality tolerance") is set at "6.3e-14".

## mxOption(NULL, "Gradient iterations") is set at "2".
```

```
# Working directory
wd = 'D:/Research/2023/CompareMASEM/CFA/Bifactor/WithCovariate/'
setwd(wd)
```

2 Functions

```
# vector to matrix
v2m <- function(vec,p,corr= T){
    M = matrix(0,p,p)
    M[lower.tri(M)] = vec
    M = M + t(M)
    if(corr==TRUE){
        diag(M) = 1
    }else{
        diag(M) = diag(M)/2
    }
    return(M)
}

# impute missing values in covariance / correlation matrices of each study
# to obtain a rough estimate of the covariance matrix of covariance / correlation matrix
# weighted average correlation
Mimpute <- function(R,N,missing){
    if(is.null(missing)){
        return(R)
    }else{
        na.pos = which(is.na(R),arr.ind = TRUE)
        mu.N = mean(N)
        Rbar = apply(R,2,mean,na.rm = TRUE)# Becker's mean r

        for(coli in unique(na.pos[,2])){
            id = na.pos[(na.pos[,2] == coli),1]
            R[id,coli] = Rbar[coli]
        }
        return(R)
    }
}

# change the coordinating system of a vectorized matrix to the coordinating system of
# the original matrix
# e.g., from vS to S, the former uses one coordinate (vil), whereas the latter uses two (j,k).
Get.vi2jk <- function(p,diag.incl=FALSE,byrow=FALSE){
    A = matrix(1,p,p)
    if(diag.incl ==FALSE){
        pp = p*(p-1)/2
        vi2jk <- matrix(NA,pp,3)
        vi2jk[,3] <- 1:pp
        if(byrow == FALSE){
            vi2jk[,1:2] <- which(lower.tri(A)==1,arr.ind = TRUE)
        }else{
            vi2jk[,1:2] <- which(upper.tri(A)==1,arr.ind = TRUE)
        }
        colnames(vi2jk) = c('j','k','vi')
    }else{
        pp = p*(p+1)/2
        vi2jk <- matrix(NA,pp,3)
        vi2jk[,3] <- 1:pp
        if(byrow == FALSE){
```

```
            vi2jk[,1:2] <- which(lower.tri(A,diag = TRUE)==1,arr.ind = TRUE)
        }else{
            vi2jk[,1:2] <- which(upper.tri(A,diag = TRUE)==1,arr.ind = TRUE)
        }
        colnames(vi2jk) = c('j','k','vi')
    }
    return(vi2jk)
}

# change the coordinating system of a matrix to the coordinating system of
# the corresponding vectorized matrix
# e.g., from S to vS, the former uses two coordinates (j,k), whereas the latter uses only one (vil).
Get.jk2vi <- function(vi2jk,p,diag.incl=FALSE){
    jk2vi = matrix(0,p,p)
    jk2vi[vi2jk[,1:2]] = vi2jk[,3]
    if(diag.incl){
        jk2vi = jk2vi + t(jk2vi)
        diag(jk2vi) = diag(jk2vi)/2
    }else{
        pp = p*(p-1)/2
        jk2vi = jk2vi + t(jk2vi) + diag(rep(pp+1,p))
    }
    return(jk2vi)
}

jkvil <- function(p){
    vi2jk   = Get.vi2jk(p)
    j   = vi2jk[,1]
    k   = vi2jk[,2]
    vil = Get.jk2vi(vi2jk,p)
    return(list(j=j,k=k,vil=vil))
}

# compute the covariance matrix of correlation matrix
# based on Steiger (1980)
Corr.Cov <- function(vR,N,index.list){
    nvR = length(vR)
    vR  = c(vR,1)
    NvR.cov = matrix(NA,nvR,nvR)
    j = index.list$j
    k = index.list$k
    vil = index.list$vil

    for(vi in 1:nvR){
        NvR.cov[vi,vi] = (1-(vR[vi])^2)^2
    }
    for(vi in 1:(nvR-1)){
    for(vj in (vi+1):nvR){
        NvR.cov[vi,vj] = ((vR[vil[j[vi],j[vj]]]-vR[vi]*vR[vil[k[vi],j[vj]]])*(vR[vil[k[vi],k[vj]]]-vR[v
         +(vR[vil[j[vi],k[vj]]]-vR[vil[j[vi],j[vj]]]*vR[vj])*(vR[vil[k[vi],j[vj]]]-vR[vi]*vR[vil[j[vi],
         +(vR[vil[j[vi],j[vj]]]-vR[vil[j[vi],k[vj]]]*vR[vj])*(vR[vil[k[vi],k[vj]]]-vR[vi]*vR[vil[j[vi],
         +(vR[vil[j[vi],k[vj]]]-vR[vi]*vR[vil[k[vi],k[vj]]])*(vR[vil[j[vj],k[vi]]]-vR[vil[k[vi],k[vj]]]
        NvR.cov[vj,vi] <- NvR.cov[vi,vj]
```

```
    }
    }

    vR.cov = NvR.cov/(N)
    vR.cov = as.matrix(nearPD(vR.cov,posd.tol = 1e-5)$mat)
    return(vR.cov)
}

# Use average correlation vector to compute V_psi
Vj <- function(vR.bar,N,pp,Nstudy,index.list){

    mu.N = mean(N)
    S.vR.bar = Corr.Cov(vR.bar,mu.N,index.list)
    inv.S.vR.bar = solve(S.vR.bar)
    tau.vR = array(NA,dim = c(Nstudy,pp,pp))
    S.vR = array(NA,dim = c(Nstudy,pp,pp))
    for(i in 1:Nstudy){
        S.vR[i,,]<- S.vR.bar/N[i]*mu.N
        tau.vR[i,,] <- inv.S.vR.bar/mu.N*N[i]
    }
    return(list(S.vR = S.vR,tau.vR = tau.vR))
}

# Use individual correlation vectors to compute V_psi
Vj2 <- function(vR.impute,N,pp,Nstudy,index.list){

    tau.vR = array(NA,dim = c(Nstudy,pp,pp))
    S.vR = array(NA,dim = c(Nstudy,pp,pp))
    for(i in 1:Nstudy){
        S.vR[i,,] = Corr.Cov(vR.impute[i,],N[i],index.list)
        tau.vR[i,,] <- solve(S.vR[i,,])
    }
    return(list(S.vR = S.vR,tau.vR = tau.vR))
}

# generate data for meta-analytic CFA
# the two-level model of OSMASEM is used
Gen.CFA.data <- function(Nstudy,mu.N,Model.list,p,missing,N=NULL){

    beta = Model.list$beta
    tau = Model.list$tau
    ind = Model.list$ind
    Z = Model.list$Z
    pp = Model.list$pp
    j = Model.list$j
    j10 = Model.list$j10
    k = Model.list$k
    k10 = Model.list$k10
    vil = Model.list$vil

    # predicted SEM parameters
    coefM <- Z%*%t(beta)
```

```
    # predicted part of the true correlation vector for each study
    vPs = t(apply(coefM,1,function(x,pp,j,k,j10,k10,ind){
        r = rep(NA,pp)
        for(vi in 1:pp){
          r[vi] = x[j[vi]]*x[k[vi]]+x[j10[vi]]*x[k10[vi]]*ind[vi]
        }
        return(r)
    },pp=pp,j=j,k=k,j10=j10,k10=k10,ind=ind) )

    # true correlation vector for each study
    if(tau[1]>0){
       vP = t(apply(vPs,1,function(x,tau,pp){
        r = rep(NA,pp)
        for(vi in 1:pp){ r[vi] = rnorm(1,x[vi],sd=tau[vi]) }
        return(r)
       },tau=tau,pp=pp) )
    }else{ vP=vPs }

    # sample size for each study
    if(is.null(N)){
      N <- rzinb(n =Nstudy, k =0.8, lambda=round(mu.N*0.2), omega = 0)
      N <- N + round(mu.N*0.8)
    }

    # observed correlations
    vR = matrix(NA,Nstudy,pp)
    for(studyi in 1:Nstudy){
        Pm = v2m(vP[studyi,],p,T)
        Pm = nearPD(Pm,corr=T)$mat
        Ri = cor(mvrnorm(N[studyi],rep(0,p),Pm))
        vR[studyi,] = Ri[lower.tri(Ri)]
    }

    #source(paste(wd,'RealData.R',sep=''))
    #vR = Make.Missing2(vR,missing,miss.rate,N) # generate missing values
    return(list(j=j,k=k,vil=vil,pp=pp,N=N,vR=vR,Z=Z))
}

d4osmasem <- function(dsim){
    j = dsim$j
    vR = dsim$vR
    N = dsim$N
    Z = as.matrix(dsim$Z)

    p = max(j)
    R.l = as.list(as.data.frame(t(vR)))
    Mat = lapply(R.l,function(x,p) v2m(x,p,T),p=p)
    my.df = Cor2DataFrame(Mat,N,acov = 'weighted')
    my.df$data = data.frame(my.df$data,covariate=scale(Z[,1]),check.names = FALSE)
    return(my.df)
}

wbugs <-function(data,initsl,prm,mfn,
```

```r
    nchains=1,niter=60000,nburnin=30000,nthin=1,wd,
    diagm){
# data: a named list of the data in the likelihood model for OpenBUGS
# initsl: a list with nchains elements; each element is a list of starting values
# prm: vector of names of the parameters to save
# mfn: the file name of the likelihood model for OpenBUGS
# diagm: name of the convergence diagnostic method; either 'Geweke' or 'Gelman'
# The function checks convergence every niter-nburnin iterations

    fit = bugs(data,initsl,prm,mfn,
        n.chains=nchains,n.iter=niter,n.burnin=nburnin,n.thin=1,
        debug=F,saveExec=T,working.directory = wd)

    for(tryi in 2:20){
        print(paste0('Iteration: ',tryi*(niter-nburnin)))
        fit.coda = read.openbugs(stem="",thin = nthin)
        del.id = na.omit(match(c('ppp'),varnames(fit.coda)))
        print(summary(fit.coda),3)
        if(diagm=='Geweke'){
            if(length(del.id)>0){
                tmp.conv = geweke.diag(fit.coda[,-del.id])[[1]]$z
            }else{ tmp.conv = geweke.diag(fit.coda)[[1]]$z }
            crit = (sum((abs(tmp.conv)>1.96),na.rm = T)==0)
        }else if(diagm=='Gelman'){
            if(length(del.id)>0){
                tmp.conv = gelman.diag(fit.coda)$psrf[-del.id,2]
            }else{ tmp.conv = gelman.diag(fit.coda)$psrf[,2] }
            crit = (sum((tmp.conv>1.1),na.rm = T)==0)
        }
        if(crit){
            print(tmp.conv)
            print(summary(fit.coda),3)
            break
        }else{
            fit = bugs(data,initsl,prm,mfn,
            n.chains=nchains,n.iter=niter-nburnin+1,n.burnin=1,n.thin=1,
            restart=T,saveExec=T,working.directory = wd)
        }
    }
    ppp.id = match('ppp',prm)
    sel = NA
    if(is.na(ppp.id)){
        nprm = length(prm)
        for(i in 1:nprm){
            sel = c(sel,grep(prm[i],rownames(summary(fit.coda)$quantiles)))
        }
    }else{
        prm = prm[-ppp.id]
        nprm = length(prm)
        for(i in 1:nprm){
            sel = c(sel,grep(prm[i],rownames(summary(fit.coda)$quantiles)))
        }
    }
```

```
    sel = sel[-1]
    sel = unique(sel)

    if(is.na(ppp.id)){ est = round(summary(fit.coda)$quantiles[sel,'50%'],3)
    }else{
        est = round(c(summary(fit.coda)$quantiles[sel,'50%'],
        summary(fit.coda)$statistics['ppp','Mean']),3)
    }
    psd = round(summary(fit.coda)$statistics[sel,'SD'],3)
    if(diagm=='Geweke'){
        CIl = round(HPDinterval(fit.coda,prob = .95)[[1]][sel,1],3)
        CIu = round(HPDinterval(fit.coda,prob = .95)[[1]][sel,2],3)
    }else if(diagm=='Gelman'){
        fit.coda.l = do.call(rbind,fit.coda)
        HPDCI = HPDinterval(mcmc(fit.coda.l),prob = .95)
        CIl = HPDCI[sel,1]
        CIu = HPDCI[sel,2]
    }
    sel.muL = grep('mu.L',names(est))
    sel.sdL = grep('sd.L',names(est))
    CVl = round(est[sel.muL] - 1.28*est[sel.sdL],3)
    CVu = round(est[sel.muL] + 1.28*est[sel.sdL],3)

    conv = round(c(tryi,tmp.conv),3)
    return(list(est=est,psd=psd,CIl=CIl,CIu=CIu,CVl=CVl,CVu=CVu,conv=conv,
        DIC=fit$DIC,fit.coda=fit.coda))
}
```

3 BMASEM

3.1 Data preparation

```
## Remove studies that did not report Individualism or bivariate correlations
index_na <- is.na(Gnambs18$Individualism)
Gnambs18 <- lapply(Gnambs18,function(x) x[!index_na])
index    <- Gnambs18$CorMat==1
Gnambs18 <- lapply(Gnambs18, function(x) x[index])

# Standardize Individualism
M <- Gnambs18$Individualism
M <- (M-mean(M))/sd(M)

# Convert correlation matrices to correlation vectors
mR = Gnambs18$data
vR = sapply(mR,function(x){ x = x[c(1,3,4,7,10,2,5,6,8,9),c(1,3,4,7,10,2,5,6,8,9)]
    return(x[lower.tri(x)]) })
vR = t(vR)

N      = Gnambs18$n # sample sizes within primary studies
mu.N   = mean(N) # mean sample size
Nstudy = length(Gnambs18$data) # the number of primary studies
```

```r
# Coordinates of correlation matrices and vectors
p  = 10 # number of variables
pp = p*(p-1)/2  # number of bivariate correlations
index.list = jkvil(p)
j = index.list$j
k = index.list$k
vil = index.list$vil
j10 = j+10
k10 = k+10
# Do items load on the same factor? 1=No; 0 = Yes
ind = (j>(p+1)/2)*(k<(p+2)/2)

# Covariance matrices of sample correlation vectors
vR.bar = apply(vR,2,mean,na.rm = TRUE)
Stau.vR = Vj(vR.bar,N,pp,Nstudy,index.list)
tau.vR = Stau.vR$tau.vR

# information for the additional error term
mu.vR.psi = rep(0,pp)
df.prelim = 100*pp/mu.N+pp
alpha.prior.vE = (df.prelim-pp+1)/2
beta.prior.vE = alpha.prior.vE*(0.3/mu.N)
```

3.2 Model fitting

```r
# Data
data<-list("Nstudy","N","mu.N",'p',"pp","j","k",'j10','k10','ind',
   "vR","tau.vR",'M',"mu.vR.psi",'alpha.prior.vE','beta.prior.vE')

#Initial values
vR.inits = vR;vR.inits[which(is.na(vR)==0,arr.ind = TRUE)] = NA
vL.inits = matrix(0.6,Nstudy,p*2);vL.inits[,15] = NA
initsl <- list(list(a=rep(0.6,p*2),b=c(rep(0,14),NA,rep(0,5)),
   sd.uL = c(rep(0.1,14),NA,rep(0.1,5)),tau.R=mu.N*3,
   vR.psi = matrix(0,Nstudy,pp),vR = vR.inits,vR.rep = vR,vL = vL.inits))

prm =c('a','b','sd.uL','vLH.pred','vLL.pred') # Parameters to save
model.fn = paste(wd,'CFACovariate.txt',sep='') # model file name


# stop every 10000 iterations to check whether convergence is achieved
fit = wbugs(data,initsl,prm,model.fn,
        nchains=1,niter=60000,nburnin=30000,nthin=1,wd,diagm='Geweke')
```

```
## [1] "Iteration: 60000"
## Abstracting a[1] ... 30000 valid values
## Abstracting a[2] ... 30000 valid values
## Abstracting a[3] ... 30000 valid values
## Abstracting a[4] ... 30000 valid values
## Abstracting a[5] ... 30000 valid values
## Abstracting a[6] ... 30000 valid values
## Abstracting a[7] ... 30000 valid values
## Abstracting a[8] ... 30000 valid values
```

```
## Abstracting a[9] ... 30000 valid values
## Abstracting a[10] ... 30000 valid values
## Abstracting a[11] ... 30000 valid values
## Abstracting a[12] ... 30000 valid values
## Abstracting a[13] ... 30000 valid values
## Abstracting a[14] ... 30000 valid values
## Abstracting a[15] ... 30000 valid values
## Abstracting a[16] ... 30000 valid values
## Abstracting a[17] ... 30000 valid values
## Abstracting a[18] ... 30000 valid values
## Abstracting a[19] ... 30000 valid values
## Abstracting a[20] ... 30000 valid values
## Abstracting b[1] ... 30000 valid values
## Abstracting b[2] ... 30000 valid values
## Abstracting b[3] ... 30000 valid values
## Abstracting b[4] ... 30000 valid values
## Abstracting b[5] ... 30000 valid values
## Abstracting b[6] ... 30000 valid values
## Abstracting b[7] ... 30000 valid values
## Abstracting b[8] ... 30000 valid values
## Abstracting b[9] ... 30000 valid values
## Abstracting b[10] ... 30000 valid values
## Abstracting b[11] ... 30000 valid values
## Abstracting b[12] ... 30000 valid values
## Abstracting b[13] ... 30000 valid values
## Abstracting b[14] ... 30000 valid values
## Abstracting b[16] ... 30000 valid values
## Abstracting b[17] ... 30000 valid values
## Abstracting b[18] ... 30000 valid values
## Abstracting b[19] ... 30000 valid values
## Abstracting b[20] ... 30000 valid values
## Abstracting deviance ... 30000 valid values
## Abstracting sd.uL[1] ... 30000 valid values
## Abstracting sd.uL[2] ... 30000 valid values
## Abstracting sd.uL[3] ... 30000 valid values
## Abstracting sd.uL[4] ... 30000 valid values
## Abstracting sd.uL[5] ... 30000 valid values
## Abstracting sd.uL[6] ... 30000 valid values
## Abstracting sd.uL[7] ... 30000 valid values
## Abstracting sd.uL[8] ... 30000 valid values
## Abstracting sd.uL[9] ... 30000 valid values
## Abstracting sd.uL[10] ... 30000 valid values
## Abstracting sd.uL[11] ... 30000 valid values
## Abstracting sd.uL[12] ... 30000 valid values
## Abstracting sd.uL[13] ... 30000 valid values
## Abstracting sd.uL[14] ... 30000 valid values
## Abstracting sd.uL[16] ... 30000 valid values
## Abstracting sd.uL[17] ... 30000 valid values
## Abstracting sd.uL[18] ... 30000 valid values
## Abstracting sd.uL[19] ... 30000 valid values
## Abstracting sd.uL[20] ... 30000 valid values
## Abstracting vLH.pred[1] ... 30000 valid values
## Abstracting vLH.pred[2] ... 30000 valid values
## Abstracting vLH.pred[3] ... 30000 valid values
```

```
## Abstracting vLH.pred[4] ... 30000 valid values
## Abstracting vLH.pred[5] ... 30000 valid values
## Abstracting vLH.pred[6] ... 30000 valid values
## Abstracting vLH.pred[7] ... 30000 valid values
## Abstracting vLH.pred[8] ... 30000 valid values
## Abstracting vLH.pred[9] ... 30000 valid values
## Abstracting vLH.pred[10] ... 30000 valid values
## Abstracting vLL.pred[1] ... 30000 valid values
## Abstracting vLL.pred[2] ... 30000 valid values
## Abstracting vLL.pred[3] ... 30000 valid values
## Abstracting vLL.pred[4] ... 30000 valid values
## Abstracting vLL.pred[5] ... 30000 valid values
## Abstracting vLL.pred[6] ... 30000 valid values
## Abstracting vLL.pred[7] ... 30000 valid values
## Abstracting vLL.pred[8] ... 30000 valid values
## Abstracting vLL.pred[9] ... 30000 valid values
## Abstracting vLL.pred[10] ... 30000 valid values
##
## Iterations = 30001:60000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 30000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                   Mean       SD Naive SE Time-series SE
## a[1]          7.47e-01  0.01880 1.09e-04       0.000146
## a[2]          5.96e-01  0.01681 9.71e-05       0.000201
## a[3]          5.24e-01  0.01700 9.81e-05       0.000157
## a[4]          6.14e-01  0.02178 1.26e-04       0.000174
## a[5]          8.02e-01  0.01171 6.76e-05       0.000117
## a[6]          5.39e-01  0.01727 9.97e-05       0.000157
## a[7]          5.23e-01  0.02397 1.38e-04       0.000185
## a[8]          5.19e-01  0.01274 7.35e-05       0.000143
## a[9]          3.85e-01  0.03484 2.01e-04       0.000241
## a[10]         5.99e-01  0.01863 1.08e-04       0.000156
## a[11]        -5.12e-02  0.01948 1.12e-04       0.000498
## a[12]         5.21e-01  0.02693 1.56e-04       0.000656
## a[13]         3.04e-01  0.02147 1.24e-04       0.000335
## a[14]         3.28e-01  0.03643 2.10e-04       0.000389
## a[15]        -3.20e-02  0.01353 7.81e-05       0.000327
## a[16]         5.69e-01  0.01580 9.12e-05       0.000209
## a[17]         3.27e-01  0.02003 1.16e-04       0.000223
## a[18]         5.91e-01  0.01496 8.64e-05       0.000217
## a[19]         3.58e-01  0.02348 1.36e-04       0.000222
## a[20]         3.89e-01  0.02498 1.44e-04       0.000230
## b[1]          4.30e-02  0.01990 1.15e-04       0.000180
## b[2]         -3.89e-04  0.01775 1.02e-04       0.000196
## b[3]         -2.13e-02  0.01783 1.03e-04       0.000179
## b[4]          1.84e-02  0.02269 1.31e-04       0.000188
## b[5]          2.00e-02  0.01265 7.31e-05       0.000127
## b[6]          6.82e-02  0.01801 1.04e-04       0.000173
## b[7]          4.83e-02  0.02506 1.45e-04       0.000202
```

```
## b[8]          6.59e-02  0.01364 7.87e-05       0.000162
## b[9]          1.71e-01  0.03590 2.07e-04       0.000258
## b[10]         4.21e-02  0.01947 1.12e-04       0.000175
## b[11]         9.13e-03  0.02213 1.28e-04       0.000556
## b[12]         4.74e-02  0.02846 1.64e-04       0.000673
## b[13]         2.86e-02  0.02370 1.37e-04       0.000369
## b[14]         2.39e-03  0.03993 2.31e-04       0.000505
## b[16]        -3.62e-02  0.01689 9.75e-05       0.000234
## b[17]        -4.60e-02  0.02105 1.22e-04       0.000231
## b[18]        -3.67e-02  0.01632 9.42e-05       0.000252
## b[19]         1.10e-02  0.02467 1.42e-04       0.000235
## b[20]        -6.71e-05  0.02654 1.53e-04       0.000273
## deviance     -5.14e+03 56.01973 3.23e-01       0.663395
## sd.uL[1]      9.77e-02  0.01521 8.78e-05       0.000135
## sd.uL[2]      8.24e-02  0.01308 7.55e-05       0.000126
## sd.uL[3]      8.50e-02  0.01298 7.50e-05       0.000115
## sd.uL[4]      1.15e-01  0.01715 9.90e-05       0.000151
## sd.uL[5]      5.41e-02  0.00959 5.54e-05       0.000101
## sd.uL[6]      8.78e-02  0.01329 7.67e-05       0.000117
## sd.uL[7]      1.29e-01  0.01910 1.10e-04       0.000162
## sd.uL[8]      5.83e-02  0.01000 5.77e-05       0.000110
## sd.uL[9]      1.93e-01  0.02725 1.57e-04       0.000215
## sd.uL[10]     9.55e-02  0.01459 8.42e-05       0.000129
## sd.uL[11]     4.97e-02  0.01921 1.11e-04       0.000538
## sd.uL[12]     6.37e-02  0.02832 1.64e-04       0.000968
## sd.uL[13]     7.64e-02  0.01984 1.15e-04       0.000354
## sd.uL[14]     1.67e-01  0.03358 1.94e-04       0.000437
## sd.uL[16]     5.66e-02  0.01512 8.73e-05       0.000272
## sd.uL[17]     8.71e-02  0.01736 1.00e-04       0.000236
## sd.uL[18]     4.90e-02  0.01676 9.68e-05       0.000397
## sd.uL[19]     1.11e-01  0.02016 1.16e-04       0.000239
## sd.uL[20]     1.19e-01  0.02220 1.28e-04       0.000256
## vLH.pred[1]   8.04e-01  0.03168 1.83e-04       0.000259
## vLH.pred[2]   5.95e-01  0.02823 1.63e-04       0.000306
## vLH.pred[3]   4.96e-01  0.02830 1.63e-04       0.000280
## vLH.pred[4]   6.38e-01  0.03641 2.10e-04       0.000279
## vLH.pred[5]   8.29e-01  0.01965 1.13e-04       0.000182
## vLH.pred[6]   6.28e-01  0.02886 1.67e-04       0.000264
## vLH.pred[7]   5.86e-01  0.04036 2.33e-04       0.000305
## vLH.pred[8]   6.06e-01  0.02149 1.24e-04       0.000242
## vLH.pred[9]   6.08e-01  0.05843 3.37e-04       0.000406
## vLH.pred[10]  6.54e-01  0.03118 1.80e-04       0.000266
## vLL.pred[1]   6.48e-01  0.05006 2.89e-04       0.000467
## vLL.pred[2]   5.97e-01  0.04472 2.58e-04       0.000524
## vLL.pred[3]   5.73e-01  0.04511 2.60e-04       0.000455
## vLL.pred[4]   5.72e-01  0.05711 3.30e-04       0.000482
## vLL.pred[5]   7.56e-01  0.03211 1.85e-04       0.000353
## vLL.pred[6]   3.82e-01  0.04535 2.62e-04       0.000443
## vLL.pred[7]   4.11e-01  0.06282 3.63e-04       0.000500
## vLL.pred[8]   3.68e-01  0.03439 1.99e-04       0.000412
## vLL.pred[9]  -8.05e-03  0.08984 5.19e-04       0.000653
## vLL.pred[10]  5.02e-01  0.04902 2.83e-04       0.000441
##
## 2. Quantiles for each variable:
```

```
##
##                  2.5%       25%       50%       75%     97.5%
## a[1]          7.10e-01  7.35e-01  7.47e-01  7.60e-01  7.85e-01
## a[2]          5.63e-01  5.85e-01  5.96e-01  6.07e-01  6.29e-01
## a[3]          4.91e-01  5.13e-01  5.24e-01  5.35e-01  5.58e-01
## a[4]          5.71e-01  6.00e-01  6.14e-01  6.29e-01  6.57e-01
## a[5]          7.79e-01  7.95e-01  8.02e-01  8.10e-01  8.26e-01
## a[6]          5.05e-01  5.27e-01  5.39e-01  5.50e-01  5.73e-01
## a[7]          4.75e-01  5.07e-01  5.23e-01  5.38e-01  5.69e-01
## a[8]          4.94e-01  5.11e-01  5.19e-01  5.28e-01  5.44e-01
## a[9]          3.16e-01  3.61e-01  3.84e-01  4.08e-01  4.53e-01
## a[10]         5.62e-01  5.87e-01  5.99e-01  6.11e-01  6.36e-01
## a[11]        -8.93e-02 -6.43e-02 -5.15e-02 -3.84e-02 -1.16e-02
## a[12]         4.70e-01  5.03e-01  5.20e-01  5.39e-01  5.76e-01
## a[13]         2.62e-01  2.90e-01  3.04e-01  3.18e-01  3.46e-01
## a[14]         2.56e-01  3.04e-01  3.28e-01  3.52e-01  4.00e-01
## a[15]        -5.85e-02 -4.12e-02 -3.21e-02 -2.28e-02 -5.63e-03
## a[16]         5.39e-01  5.59e-01  5.69e-01  5.80e-01  6.02e-01
## a[17]         2.88e-01  3.14e-01  3.27e-01  3.40e-01  3.67e-01
## a[18]         5.62e-01  5.81e-01  5.91e-01  6.00e-01  6.21e-01
## a[19]         3.12e-01  3.42e-01  3.57e-01  3.73e-01  4.05e-01
## a[20]         3.39e-01  3.72e-01  3.89e-01  4.06e-01  4.38e-01
## b[1]          3.78e-03  2.98e-02  4.29e-02  5.61e-02  8.22e-02
## b[2]         -3.52e-02 -1.22e-02 -5.34e-04  1.13e-02  3.51e-02
## b[3]         -5.62e-02 -3.32e-02 -2.14e-02 -9.33e-03  1.37e-02
## b[4]         -2.56e-02  3.36e-03  1.81e-02  3.34e-02  6.35e-02
## b[5]         -5.21e-03  1.17e-02  2.02e-02  2.85e-02  4.45e-02
## b[6]          3.28e-02  5.63e-02  6.80e-02  8.01e-02  1.04e-01
## b[7]         -1.02e-03  3.18e-02  4.82e-02  6.50e-02  9.76e-02
## b[8]          3.91e-02  5.70e-02  6.58e-02  7.50e-02  9.28e-02
## b[9]          1.00e-01  1.47e-01  1.70e-01  1.94e-01  2.41e-01
## b[10]         3.41e-03  2.92e-02  4.22e-02  5.50e-02  8.01e-02
## b[11]        -3.48e-02 -5.32e-03  9.10e-03  2.37e-02  5.32e-02
## b[12]        -1.04e-02  2.92e-02  4.77e-02  6.64e-02  1.03e-01
## b[13]        -1.91e-02  1.32e-02  2.88e-02  4.45e-02  7.42e-02
## b[14]        -7.58e-02 -2.40e-02  1.96e-03  2.86e-02  8.17e-02
## b[16]        -6.88e-02 -4.73e-02 -3.64e-02 -2.50e-02 -2.13e-03
## b[17]        -8.72e-02 -5.99e-02 -4.60e-02 -3.21e-02 -3.92e-03
## b[18]        -6.83e-02 -4.76e-02 -3.69e-02 -2.62e-02 -3.99e-03
## b[19]        -3.75e-02 -5.30e-03  1.08e-02  2.73e-02  5.93e-02
## b[20]        -5.26e-02 -1.74e-02 -2.17e-04  1.75e-02  5.20e-02
## deviance     -5.25e+03 -5.18e+03 -5.14e+03 -5.10e+03 -5.03e+03
## sd.uL[1]      7.24e-02  8.69e-02  9.59e-02  1.07e-01  1.32e-01
## sd.uL[2]      6.07e-02  7.32e-02  8.10e-02  9.01e-02  1.12e-01
## sd.uL[3]      6.34e-02  7.59e-02  8.37e-02  9.27e-02  1.14e-01
## sd.uL[4]      8.71e-02  1.03e-01  1.14e-01  1.26e-01  1.54e-01
## sd.uL[5]      3.81e-02  4.73e-02  5.31e-02  5.97e-02  7.57e-02
## sd.uL[6]      6.55e-02  7.85e-02  8.65e-02  9.59e-02  1.18e-01
## sd.uL[7]      9.73e-02  1.15e-01  1.27e-01  1.40e-01  1.72e-01
## sd.uL[8]      4.17e-02  5.12e-02  5.73e-02  6.44e-02  8.06e-02
## sd.uL[9]      1.48e-01  1.74e-01  1.90e-01  2.09e-01  2.55e-01
## sd.uL[10]     7.11e-02  8.52e-02  9.41e-02  1.04e-01  1.28e-01
## sd.uL[11]     1.52e-02  3.63e-02  4.84e-02  6.16e-02  9.17e-02
## sd.uL[12]     1.53e-02  4.38e-02  6.13e-02  8.07e-02  1.26e-01
```

```
## sd.uL[13]     4.27e-02  6.24e-02  7.46e-02  8.85e-02  1.20e-01
## sd.uL[14]     1.10e-01  1.43e-01  1.64e-01  1.87e-01  2.43e-01
## sd.uL[16]     3.12e-02  4.59e-02  5.52e-02  6.55e-02  9.08e-02
## sd.uL[17]     5.86e-02  7.50e-02  8.52e-02  9.72e-02  1.27e-01
## sd.uL[18]     2.03e-02  3.73e-02  4.74e-02  5.90e-02  8.64e-02
## sd.uL[19]     7.78e-02  9.64e-02  1.09e-01  1.22e-01  1.57e-01
## sd.uL[20]     8.19e-02  1.04e-01  1.17e-01  1.33e-01  1.69e-01
## vLH.pred[1]   7.42e-01  7.83e-01  8.03e-01  8.25e-01  8.67e-01
## vLH.pred[2]   5.40e-01  5.77e-01  5.95e-01  6.14e-01  6.51e-01
## vLH.pred[3]   4.41e-01  4.78e-01  4.96e-01  5.15e-01  5.53e-01
## vLH.pred[4]   5.68e-01  6.14e-01  6.38e-01  6.63e-01  7.11e-01
## vLH.pred[5]   7.90e-01  8.16e-01  8.29e-01  8.42e-01  8.67e-01
## vLH.pred[6]   5.72e-01  6.09e-01  6.28e-01  6.47e-01  6.85e-01
## vLH.pred[7]   5.06e-01  5.59e-01  5.86e-01  6.13e-01  6.65e-01
## vLH.pred[8]   5.64e-01  5.92e-01  6.06e-01  6.20e-01  6.48e-01
## vLH.pred[9]   4.93e-01  5.70e-01  6.08e-01  6.47e-01  7.23e-01
## vLH.pred[10]  5.93e-01  6.33e-01  6.54e-01  6.75e-01  7.15e-01
## vLL.pred[1]   5.49e-01  6.15e-01  6.49e-01  6.81e-01  7.46e-01
## vLL.pred[2]   5.08e-01  5.67e-01  5.97e-01  6.26e-01  6.85e-01
## vLL.pred[3]   4.85e-01  5.44e-01  5.73e-01  6.03e-01  6.62e-01
## vLL.pred[4]   4.59e-01  5.34e-01  5.73e-01  6.10e-01  6.83e-01
## vLL.pred[5]   6.94e-01  7.35e-01  7.56e-01  7.77e-01  8.21e-01
## vLL.pred[6]   2.92e-01  3.52e-01  3.82e-01  4.12e-01  4.71e-01
## vLL.pred[7]   2.88e-01  3.70e-01  4.12e-01  4.53e-01  5.34e-01
## vLL.pred[8]   3.00e-01  3.45e-01  3.68e-01  3.91e-01  4.35e-01
## vLL.pred[9]  -1.88e-01 -6.75e-02 -7.96e-03  5.19e-02  1.70e-01
## vLL.pred[10]  4.06e-01  4.70e-01  5.02e-01  5.35e-01  5.99e-01
##
##         a[1]         a[2]         a[3]         a[4]         a[5]         a[6]
##  0.381961809 -0.144694493 -0.089128971 -0.646156904  0.931957811 -0.305544639
##         a[7]         a[8]         a[9]        a[10]        a[11]        a[12]
##  0.428616718  1.325841485  0.049983314  0.700771090  1.335331400  0.969253138
##        a[13]        a[14]        a[15]        a[16]        a[17]        a[18]
##  0.465752005  0.088693900  0.419291297 -0.359520019 -0.927183138 -0.814469286
##        a[19]        a[20]         b[1]         b[2]         b[3]         b[4]
##  0.476778005 -1.902646862  1.489030049  0.870305511  0.014296256  1.094065852
##         b[5]         b[6]         b[7]         b[8]         b[9]        b[10]
## -1.204428022 -0.137138909 -1.644749324 -1.510508977 -1.781844920 -1.062402169
##        b[11]        b[12]        b[13]        b[14]        b[16]        b[17]
## -1.101860875 -1.343866534 -0.304540666  0.822871286 -0.385521527  0.824065634
##        b[18]        b[19]        b[20]     deviance     sd.uL[1]     sd.uL[2]
##  1.617703745  0.014690085  0.854322766 -0.966210000  1.231786335 -0.990087989
##     sd.uL[3]     sd.uL[4]     sd.uL[5]     sd.uL[6]     sd.uL[7]     sd.uL[8]
## -1.889827396 -0.874900004  0.618812981  1.605918789 -0.988671734 -0.093161208
##     sd.uL[9]    sd.uL[10]    sd.uL[11]    sd.uL[12]    sd.uL[13]    sd.uL[14]
## -0.894195613  0.191256659  0.047699157 -0.263007947 -0.173302735 -0.450586383
##    sd.uL[16]    sd.uL[17]    sd.uL[18]    sd.uL[19]    sd.uL[20]  vLH.pred[1]
##  0.137947797 -0.765938644 -0.015402527 -1.039041712  0.119248103  1.488909292
##  vLH.pred[2]  vLH.pred[3]  vLH.pred[4]  vLH.pred[5]  vLH.pred[6]  vLH.pred[7]
##  0.674786321 -0.042651781  0.535416679 -0.543205196 -0.335533130 -1.041812399
##  vLH.pred[8]  vLH.pred[9] vLH.pred[10]  vLL.pred[1]  vLL.pred[2]  vLL.pred[3]
## -0.422747669 -1.418785870 -0.462189613 -1.152351272 -0.831512014 -0.044405788
##  vLL.pred[4]  vLL.pred[5]  vLL.pred[6]  vLL.pred[7]  vLL.pred[8]  vLL.pred[9]
## -1.210859511  1.364997719  0.009796852  1.682761597  1.859099215  1.635614681
```

```
## vLL.pred[10]
##  1.207003664
##
## Iterations = 30001:60000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 30000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                    Mean       SD Naive SE Time-series SE
## a[1]           7.47e-01  0.01880 1.09e-04       0.000146
## a[2]           5.96e-01  0.01681 9.71e-05       0.000201
## a[3]           5.24e-01  0.01700 9.81e-05       0.000157
## a[4]           6.14e-01  0.02178 1.26e-04       0.000174
## a[5]           8.02e-01  0.01171 6.76e-05       0.000117
## a[6]           5.39e-01  0.01727 9.97e-05       0.000157
## a[7]           5.23e-01  0.02397 1.38e-04       0.000185
## a[8]           5.19e-01  0.01274 7.35e-05       0.000143
## a[9]           3.85e-01  0.03484 2.01e-04       0.000241
## a[10]          5.99e-01  0.01863 1.08e-04       0.000156
## a[11]         -5.12e-02  0.01948 1.12e-04       0.000498
## a[12]          5.21e-01  0.02693 1.56e-04       0.000656
## a[13]          3.04e-01  0.02147 1.24e-04       0.000335
## a[14]          3.28e-01  0.03643 2.10e-04       0.000389
## a[15]         -3.20e-02  0.01353 7.81e-05       0.000327
## a[16]          5.69e-01  0.01580 9.12e-05       0.000209
## a[17]          3.27e-01  0.02003 1.16e-04       0.000223
## a[18]          5.91e-01  0.01496 8.64e-05       0.000217
## a[19]          3.58e-01  0.02348 1.36e-04       0.000222
## a[20]          3.89e-01  0.02498 1.44e-04       0.000230
## b[1]           4.30e-02  0.01990 1.15e-04       0.000180
## b[2]          -3.89e-04  0.01775 1.02e-04       0.000196
## b[3]          -2.13e-02  0.01783 1.03e-04       0.000179
## b[4]           1.84e-02  0.02269 1.31e-04       0.000188
## b[5]           2.00e-02  0.01265 7.31e-05       0.000127
## b[6]           6.82e-02  0.01801 1.04e-04       0.000173
## b[7]           4.83e-02  0.02506 1.45e-04       0.000202
## b[8]           6.59e-02  0.01364 7.87e-05       0.000162
## b[9]           1.71e-01  0.03590 2.07e-04       0.000258
## b[10]          4.21e-02  0.01947 1.12e-04       0.000175
## b[11]          9.13e-03  0.02213 1.28e-04       0.000556
## b[12]          4.74e-02  0.02846 1.64e-04       0.000673
## b[13]          2.86e-02  0.02370 1.37e-04       0.000369
## b[14]          2.39e-03  0.03993 2.31e-04       0.000505
## b[16]         -3.62e-02  0.01689 9.75e-05       0.000234
## b[17]         -4.60e-02  0.02105 1.22e-04       0.000231
## b[18]         -3.67e-02  0.01632 9.42e-05       0.000252
## b[19]          1.10e-02  0.02467 1.42e-04       0.000235
## b[20]         -6.71e-05  0.02654 1.53e-04       0.000273
## deviance      -5.14e+03 56.01973 3.23e-01       0.663395
## sd.uL[1]       9.77e-02  0.01521 8.78e-05       0.000135
## sd.uL[2]       8.24e-02  0.01308 7.55e-05       0.000126
```

```
## sd.uL[3]      8.50e-02  0.01298 7.50e-05       0.000115
## sd.uL[4]      1.15e-01  0.01715 9.90e-05       0.000151
## sd.uL[5]      5.41e-02  0.00959 5.54e-05       0.000101
## sd.uL[6]      8.78e-02  0.01329 7.67e-05       0.000117
## sd.uL[7]      1.29e-01  0.01910 1.10e-04       0.000162
## sd.uL[8]      5.83e-02  0.01000 5.77e-05       0.000110
## sd.uL[9]      1.93e-01  0.02725 1.57e-04       0.000215
## sd.uL[10]     9.55e-02  0.01459 8.42e-05       0.000129
## sd.uL[11]     4.97e-02  0.01921 1.11e-04       0.000538
## sd.uL[12]     6.37e-02  0.02832 1.64e-04       0.000968
## sd.uL[13]     7.64e-02  0.01984 1.15e-04       0.000354
## sd.uL[14]     1.67e-01  0.03358 1.94e-04       0.000437
## sd.uL[16]     5.66e-02  0.01512 8.73e-05       0.000272
## sd.uL[17]     8.71e-02  0.01736 1.00e-04       0.000236
## sd.uL[18]     4.90e-02  0.01676 9.68e-05       0.000397
## sd.uL[19]     1.11e-01  0.02016 1.16e-04       0.000239
## sd.uL[20]     1.19e-01  0.02220 1.28e-04       0.000256
## vLH.pred[1]   8.04e-01  0.03168 1.83e-04       0.000259
## vLH.pred[2]   5.95e-01  0.02823 1.63e-04       0.000306
## vLH.pred[3]   4.96e-01  0.02830 1.63e-04       0.000280
## vLH.pred[4]   6.38e-01  0.03641 2.10e-04       0.000279
## vLH.pred[5]   8.29e-01  0.01965 1.13e-04       0.000182
## vLH.pred[6]   6.28e-01  0.02886 1.67e-04       0.000264
## vLH.pred[7]   5.86e-01  0.04036 2.33e-04       0.000305
## vLH.pred[8]   6.06e-01  0.02149 1.24e-04       0.000242
## vLH.pred[9]   6.08e-01  0.05843 3.37e-04       0.000406
## vLH.pred[10]  6.54e-01  0.03118 1.80e-04       0.000266
## vLL.pred[1]   6.48e-01  0.05006 2.89e-04       0.000467
## vLL.pred[2]   5.97e-01  0.04472 2.58e-04       0.000524
## vLL.pred[3]   5.73e-01  0.04511 2.60e-04       0.000455
## vLL.pred[4]   5.72e-01  0.05711 3.30e-04       0.000482
## vLL.pred[5]   7.56e-01  0.03211 1.85e-04       0.000353
## vLL.pred[6]   3.82e-01  0.04535 2.62e-04       0.000443
## vLL.pred[7]   4.11e-01  0.06282 3.63e-04       0.000500
## vLL.pred[8]   3.68e-01  0.03439 1.99e-04       0.000412
## vLL.pred[9]  -8.05e-03  0.08984 5.19e-04       0.000653
## vLL.pred[10]  5.02e-01  0.04902 2.83e-04       0.000441
##
## 2. Quantiles for each variable:
##
##                   2.5%       25%       50%       75%     97.5%
## a[1]          7.10e-01  7.35e-01  7.47e-01  7.60e-01  7.85e-01
## a[2]          5.63e-01  5.85e-01  5.96e-01  6.07e-01  6.29e-01
## a[3]          4.91e-01  5.13e-01  5.24e-01  5.35e-01  5.58e-01
## a[4]          5.71e-01  6.00e-01  6.14e-01  6.29e-01  6.57e-01
## a[5]          7.79e-01  7.95e-01  8.02e-01  8.10e-01  8.26e-01
## a[6]          5.05e-01  5.27e-01  5.39e-01  5.50e-01  5.73e-01
## a[7]          4.75e-01  5.07e-01  5.23e-01  5.38e-01  5.69e-01
## a[8]          4.94e-01  5.11e-01  5.19e-01  5.28e-01  5.44e-01
## a[9]          3.16e-01  3.61e-01  3.84e-01  4.08e-01  4.53e-01
## a[10]         5.62e-01  5.87e-01  5.99e-01  6.11e-01  6.36e-01
## a[11]        -8.93e-02 -6.43e-02 -5.15e-02 -3.84e-02 -1.16e-02
## a[12]         4.70e-01  5.03e-01  5.20e-01  5.39e-01  5.76e-01
## a[13]         2.62e-01  2.90e-01  3.04e-01  3.18e-01  3.46e-01
```

```
## a[14]          2.56e-01  3.04e-01  3.28e-01  3.52e-01  4.00e-01
## a[15]         -5.85e-02 -4.12e-02 -3.21e-02 -2.28e-02 -5.63e-03
## a[16]          5.39e-01  5.59e-01  5.69e-01  5.80e-01  6.02e-01
## a[17]          2.88e-01  3.14e-01  3.27e-01  3.40e-01  3.67e-01
## a[18]          5.62e-01  5.81e-01  5.91e-01  6.00e-01  6.21e-01
## a[19]          3.12e-01  3.42e-01  3.57e-01  3.73e-01  4.05e-01
## a[20]          3.39e-01  3.72e-01  3.89e-01  4.06e-01  4.38e-01
## b[1]           3.78e-03  2.98e-02  4.29e-02  5.61e-02  8.22e-02
## b[2]          -3.52e-02 -1.22e-02 -5.34e-04  1.13e-02  3.51e-02
## b[3]          -5.62e-02 -3.32e-02 -2.14e-02 -9.33e-03  1.37e-02
## b[4]          -2.56e-02  3.36e-03  1.81e-02  3.34e-02  6.35e-02
## b[5]          -5.21e-03  1.17e-02  2.02e-02  2.85e-02  4.45e-02
## b[6]           3.28e-02  5.63e-02  6.80e-02  8.01e-02  1.04e-01
## b[7]          -1.02e-03  3.18e-02  4.82e-02  6.50e-02  9.76e-02
## b[8]           3.91e-02  5.70e-02  6.58e-02  7.50e-02  9.28e-02
## b[9]           1.00e-01  1.47e-01  1.70e-01  1.94e-01  2.41e-01
## b[10]          3.41e-03  2.92e-02  4.22e-02  5.50e-02  8.01e-02
## b[11]         -3.48e-02 -5.32e-03  9.10e-03  2.37e-02  5.32e-02
## b[12]         -1.04e-02  2.92e-02  4.77e-02  6.64e-02  1.03e-01
## b[13]         -1.91e-02  1.32e-02  2.88e-02  4.45e-02  7.42e-02
## b[14]         -7.58e-02 -2.40e-02  1.96e-03  2.86e-02  8.17e-02
## b[16]         -6.88e-02 -4.73e-02 -3.64e-02 -2.50e-02 -2.13e-03
## b[17]         -8.72e-02 -5.99e-02 -4.60e-02 -3.21e-02 -3.92e-03
## b[18]         -6.83e-02 -4.76e-02 -3.69e-02 -2.62e-02 -3.99e-03
## b[19]         -3.75e-02 -5.30e-03  1.08e-02  2.73e-02  5.93e-02
## b[20]         -5.26e-02 -1.74e-02 -2.17e-04  1.75e-02  5.20e-02
## deviance      -5.25e+03 -5.18e+03 -5.14e+03 -5.10e+03 -5.03e+03
## sd.uL[1]       7.24e-02  8.69e-02  9.59e-02  1.07e-01  1.32e-01
## sd.uL[2]       6.07e-02  7.32e-02  8.10e-02  9.01e-02  1.12e-01
## sd.uL[3]       6.34e-02  7.59e-02  8.37e-02  9.27e-02  1.14e-01
## sd.uL[4]       8.71e-02  1.03e-01  1.14e-01  1.26e-01  1.54e-01
## sd.uL[5]       3.81e-02  4.73e-02  5.31e-02  5.97e-02  7.57e-02
## sd.uL[6]       6.55e-02  7.85e-02  8.65e-02  9.59e-02  1.18e-01
## sd.uL[7]       9.73e-02  1.15e-01  1.27e-01  1.40e-01  1.72e-01
## sd.uL[8]       4.17e-02  5.12e-02  5.73e-02  6.44e-02  8.06e-02
## sd.uL[9]       1.48e-01  1.74e-01  1.90e-01  2.09e-01  2.55e-01
## sd.uL[10]      7.11e-02  8.52e-02  9.41e-02  1.04e-01  1.28e-01
## sd.uL[11]      1.52e-02  3.63e-02  4.84e-02  6.16e-02  9.17e-02
## sd.uL[12]      1.53e-02  4.38e-02  6.13e-02  8.07e-02  1.26e-01
## sd.uL[13]      4.27e-02  6.24e-02  7.46e-02  8.85e-02  1.20e-01
## sd.uL[14]      1.10e-01  1.43e-01  1.64e-01  1.87e-01  2.43e-01
## sd.uL[16]      3.12e-02  4.59e-02  5.52e-02  6.55e-02  9.08e-02
## sd.uL[17]      5.86e-02  7.50e-02  8.52e-02  9.72e-02  1.27e-01
## sd.uL[18]      2.03e-02  3.73e-02  4.74e-02  5.90e-02  8.64e-02
## sd.uL[19]      7.78e-02  9.64e-02  1.09e-01  1.22e-01  1.57e-01
## sd.uL[20]      8.19e-02  1.04e-01  1.17e-01  1.33e-01  1.69e-01
## vLH.pred[1]    7.42e-01  7.83e-01  8.03e-01  8.25e-01  8.67e-01
## vLH.pred[2]    5.40e-01  5.77e-01  5.95e-01  6.14e-01  6.51e-01
## vLH.pred[3]    4.41e-01  4.78e-01  4.96e-01  5.15e-01  5.53e-01
## vLH.pred[4]    5.68e-01  6.14e-01  6.38e-01  6.63e-01  7.11e-01
## vLH.pred[5]    7.90e-01  8.16e-01  8.29e-01  8.42e-01  8.67e-01
## vLH.pred[6]    5.72e-01  6.09e-01  6.28e-01  6.47e-01  6.85e-01
## vLH.pred[7]    5.06e-01  5.59e-01  5.86e-01  6.13e-01  6.65e-01
## vLH.pred[8]    5.64e-01  5.92e-01  6.06e-01  6.20e-01  6.48e-01
```

```
## vLH.pred[9]   4.93e-01  5.70e-01  6.08e-01  6.47e-01  7.23e-01
## vLH.pred[10]  5.93e-01  6.33e-01  6.54e-01  6.75e-01  7.15e-01
## vLL.pred[1]   5.49e-01  6.15e-01  6.49e-01  6.81e-01  7.46e-01
## vLL.pred[2]   5.08e-01  5.67e-01  5.97e-01  6.26e-01  6.85e-01
## vLL.pred[3]   4.85e-01  5.44e-01  5.73e-01  6.03e-01  6.62e-01
## vLL.pred[4]   4.59e-01  5.34e-01  5.73e-01  6.10e-01  6.83e-01
## vLL.pred[5]   6.94e-01  7.35e-01  7.56e-01  7.77e-01  8.21e-01
## vLL.pred[6]   2.92e-01  3.52e-01  3.82e-01  4.12e-01  4.71e-01
## vLL.pred[7]   2.88e-01  3.70e-01  4.12e-01  4.53e-01  5.34e-01
## vLL.pred[8]   3.00e-01  3.45e-01  3.68e-01  3.91e-01  4.35e-01
## vLL.pred[9]  -1.88e-01 -6.75e-02 -7.96e-03  5.19e-02  1.70e-01
## vLL.pred[10]  4.06e-01  4.70e-01  5.02e-01  5.35e-01  5.99e-01
```

```
fit[-9]
```

```
## $est
##         a[1]         a[2]         a[3]         a[4]         a[5]         a[6]
##        0.747        0.596        0.524        0.614        0.802        0.539
##         a[7]         a[8]         a[9]        a[10]        a[11]        a[12]
##        0.523        0.519        0.384        0.599       -0.052        0.520
##        a[13]        a[14]        a[15]        a[16]        a[17]        a[18]
##        0.304        0.328       -0.032        0.569        0.327        0.591
##        a[19]        a[20]     deviance         b[1]         b[2]         b[3]
##        0.358        0.389    -5139.000        0.043       -0.001       -0.021
##         b[4]         b[5]         b[6]         b[7]         b[8]         b[9]
##        0.018        0.020        0.068        0.048        0.066        0.170
##        b[10]        b[11]        b[12]        b[13]        b[14]        b[16]
##        0.042        0.009        0.048        0.029        0.002       -0.036
##        b[17]        b[18]        b[19]        b[20]     sd.uL[1]     sd.uL[2]
##       -0.046       -0.037        0.011        0.000        0.096        0.081
##     sd.uL[3]     sd.uL[4]     sd.uL[5]     sd.uL[6]     sd.uL[7]     sd.uL[8]
##        0.084        0.114        0.053        0.086        0.127        0.057
##     sd.uL[9]    sd.uL[10]    sd.uL[11]    sd.uL[12]    sd.uL[13]    sd.uL[14]
##        0.190        0.094        0.048        0.061        0.075        0.164
##    sd.uL[16]    sd.uL[17]    sd.uL[18]    sd.uL[19]    sd.uL[20]  vLH.pred[1]
##        0.055        0.085        0.047        0.109        0.117        0.803
##  vLH.pred[2]  vLH.pred[3]  vLH.pred[4]  vLH.pred[5]  vLH.pred[6]  vLH.pred[7]
##        0.595        0.496        0.638        0.829        0.628        0.586
##  vLH.pred[8]  vLH.pred[9] vLH.pred[10]  vLL.pred[1]  vLL.pred[2]  vLL.pred[3]
##        0.606        0.608        0.654        0.649        0.597        0.573
##  vLL.pred[4]  vLL.pred[5]  vLL.pred[6]  vLL.pred[7]  vLL.pred[8]  vLL.pred[9]
##        0.573        0.756        0.382        0.412        0.368       -0.008
## vLL.pred[10]
##        0.502
##
## $psd
##         a[1]         a[2]         a[3]         a[4]         a[5]         a[6]
##        0.019        0.017        0.017        0.022        0.012        0.017
##         a[7]         a[8]         a[9]        a[10]        a[11]        a[12]
##        0.024        0.013        0.035        0.019        0.019        0.027
##        a[13]        a[14]        a[15]        a[16]        a[17]        a[18]
##        0.021        0.036        0.014        0.016        0.020        0.015
##        a[19]        a[20]     deviance         b[1]         b[2]         b[3]
##        0.023        0.025       56.020        0.020        0.018        0.018
##         b[4]         b[5]         b[6]         b[7]         b[8]         b[9]
```

```
##       0.023       0.013       0.018       0.025       0.014       0.036
##       b[10]       b[11]       b[12]       b[13]       b[14]       b[16]
##       0.019       0.022       0.028       0.024       0.040       0.017
##       b[17]       b[18]       b[19]       b[20]    sd.uL[1]    sd.uL[2]
##       0.021       0.016       0.025       0.027       0.015       0.013
##    sd.uL[3]    sd.uL[4]    sd.uL[5]    sd.uL[6]    sd.uL[7]    sd.uL[8]
##       0.013       0.017       0.010       0.013       0.019       0.010
##    sd.uL[9]   sd.uL[10]   sd.uL[11]   sd.uL[12]   sd.uL[13]   sd.uL[14]
##       0.027       0.015       0.019       0.028       0.020       0.034
##   sd.uL[16]   sd.uL[17]   sd.uL[18]   sd.uL[19]   sd.uL[20] vLH.pred[1]
##       0.015       0.017       0.017       0.020       0.022       0.032
## vLH.pred[2] vLH.pred[3] vLH.pred[4] vLH.pred[5] vLH.pred[6] vLH.pred[7]
##       0.028       0.028       0.036       0.020       0.029       0.040
## vLH.pred[8] vLH.pred[9] vLH.pred[10] vLL.pred[1] vLL.pred[2] vLL.pred[3]
##       0.021       0.058       0.031       0.050       0.045       0.045
## vLL.pred[4] vLL.pred[5] vLL.pred[6] vLL.pred[7] vLL.pred[8] vLL.pred[9]
##       0.057       0.032       0.045       0.063       0.034       0.090
## vLL.pred[10]
##       0.049
##
## $CIl
##        a[1]        a[2]        a[3]        a[4]        a[5]        a[6]
##       0.710       0.563       0.491       0.572       0.779       0.506
##        a[7]        a[8]        a[9]       a[10]       a[11]       a[12]
##       0.475       0.494       0.316       0.561      -0.091       0.468
##       a[13]       a[14]       a[15]       a[16]       a[17]       a[18]
##       0.264       0.254      -0.058       0.540       0.288       0.561
##       a[19]       a[20]    deviance        b[1]        b[2]        b[3]
##       0.311       0.340   -5254.000       0.004      -0.034      -0.057
##        b[4]        b[5]        b[6]        b[7]        b[8]        b[9]
##      -0.025      -0.005       0.033      -0.002       0.039       0.102
##       b[10]       b[11]       b[12]       b[13]       b[14]       b[16]
##       0.004      -0.034      -0.010      -0.018      -0.076      -0.069
##       b[17]       b[18]       b[19]       b[20]    sd.uL[1]    sd.uL[2]
##      -0.086      -0.068      -0.037      -0.052       0.070       0.059
##    sd.uL[3]    sd.uL[4]    sd.uL[5]    sd.uL[6]    sd.uL[7]    sd.uL[8]
##       0.062       0.085       0.037       0.064       0.093       0.040
##    sd.uL[9]   sd.uL[10]   sd.uL[11]   sd.uL[12]   sd.uL[13]   sd.uL[14]
##       0.144       0.070       0.012       0.009       0.039       0.106
##   sd.uL[16]   sd.uL[17]   sd.uL[18]   sd.uL[19]   sd.uL[20] vLH.pred[1]
##       0.029       0.056       0.018       0.074       0.078       0.742
## vLH.pred[2] vLH.pred[3] vLH.pred[4] vLH.pred[5] vLH.pred[6] vLH.pred[7]
##       0.540       0.440       0.567       0.791       0.571       0.506
## vLH.pred[8] vLH.pred[9] vLH.pred[10] vLL.pred[1] vLL.pred[2] vLL.pred[3]
##       0.564       0.496       0.593       0.549       0.511       0.487
## vLL.pred[4] vLL.pred[5] vLL.pred[6] vLL.pred[7] vLL.pred[8] vLL.pred[9]
##       0.459       0.694       0.293       0.285       0.301      -0.184
## vLL.pred[10]
##       0.406
##
## $CIu
##        a[1]        a[2]        a[3]        a[4]        a[5]        a[6]
##       0.784       0.628       0.558       0.657       0.825       0.574
##        a[7]        a[8]        a[9]       a[10]       a[11]       a[12]
```

```
##        0.569        0.545        0.454        0.635       -0.014        0.574
##        a[13]        a[14]        a[15]        a[16]        a[17]        a[18]
##        0.347        0.398       -0.005        0.602        0.366        0.620
##        a[19]        a[20]     deviance         b[1]         b[2]         b[3]
##        0.404        0.438    -5035.000        0.082        0.036        0.013
##         b[4]         b[5]         b[6]         b[7]         b[8]         b[9]
##        0.064        0.045        0.104        0.096        0.093        0.242
##        b[10]        b[11]        b[12]        b[13]        b[14]        b[16]
##        0.080        0.053        0.103        0.075        0.082       -0.003
##        b[17]        b[18]        b[19]        b[20]     sd.uL[1]     sd.uL[2]
##       -0.003       -0.004        0.060        0.052        0.128        0.109
##     sd.uL[3]     sd.uL[4]     sd.uL[5]     sd.uL[6]     sd.uL[7]     sd.uL[8]
##        0.111        0.151        0.073        0.115        0.166        0.078
##     sd.uL[9]    sd.uL[10]    sd.uL[11]    sd.uL[12]    sd.uL[13]    sd.uL[14]
##        0.248        0.126        0.088        0.117        0.115        0.236
##    sd.uL[16]    sd.uL[17]    sd.uL[18]    sd.uL[19]    sd.uL[20]  vLH.pred[1]
##        0.087        0.122        0.083        0.151        0.163        0.866
##  vLH.pred[2]  vLH.pred[3]  vLH.pred[4]  vLH.pred[5]  vLH.pred[6]  vLH.pred[7]
##        0.651        0.552        0.709        0.868        0.684        0.664
##  vLH.pred[8]  vLH.pred[9] vLH.pred[10]  vLL.pred[1]  vLL.pred[2]  vLL.pred[3]
##        0.648        0.725        0.715        0.746        0.687        0.663
##  vLL.pred[4]  vLL.pred[5]  vLL.pred[6]  vLL.pred[7]  vLL.pred[8]  vLL.pred[9]
##        0.682        0.821        0.472        0.532        0.436        0.172
## vLL.pred[10]
##        0.599
##
## $CVl
## named numeric(0)
##
## $CVu
## named numeric(0)
##
## $conv
##                     a[1]         a[2]         a[3]         a[4]         a[5]
##        2.000        0.382       -0.145       -0.089       -0.646        0.932
##         a[6]         a[7]         a[8]         a[9]        a[10]        a[11]
##       -0.306        0.429        1.326        0.050        0.701        1.335
##        a[12]        a[13]        a[14]        a[15]        a[16]        a[17]
##        0.969        0.466        0.089        0.419       -0.360       -0.927
##        a[18]        a[19]        a[20]         b[1]         b[2]         b[3]
##       -0.814        0.477       -1.903        1.489        0.870        0.014
##         b[4]         b[5]         b[6]         b[7]         b[8]         b[9]
##        1.094       -1.204       -0.137       -1.645       -1.511       -1.782
##        b[10]        b[11]        b[12]        b[13]        b[14]        b[16]
##       -1.062       -1.102       -1.344       -0.305        0.823       -0.386
##        b[17]        b[18]        b[19]        b[20]     deviance     sd.uL[1]
##        0.824        1.618        0.015        0.854       -0.966        1.232
##     sd.uL[2]     sd.uL[3]     sd.uL[4]     sd.uL[5]     sd.uL[6]     sd.uL[7]
##       -0.990       -1.890       -0.875        0.619        1.606       -0.989
##     sd.uL[8]     sd.uL[9]    sd.uL[10]    sd.uL[11]    sd.uL[12]    sd.uL[13]
##       -0.093       -0.894        0.191        0.048       -0.263       -0.173
##    sd.uL[14]    sd.uL[16]    sd.uL[17]    sd.uL[18]    sd.uL[19]    sd.uL[20]
##       -0.451        0.138       -0.766       -0.015       -1.039        0.119
##  vLH.pred[1]  vLH.pred[2]  vLH.pred[3]  vLH.pred[4]  vLH.pred[5]  vLH.pred[6]
```

```
##        1.489        0.675       -0.043        0.535       -0.543       -0.336
##  vLH.pred[7]  vLH.pred[8]  vLH.pred[9] vLH.pred[10]  vLL.pred[1]  vLL.pred[2]
##       -1.042       -0.423       -1.419       -0.462       -1.152       -0.832
##  vLL.pred[3]  vLL.pred[4]  vLL.pred[5]  vLL.pred[6]  vLL.pred[7]  vLL.pred[8]
##       -0.044       -1.211        1.365        0.010        1.683        1.859
##  vLL.pred[9] vLL.pred[10]
##        1.636        1.207
##
## $DIC
## [1] -4566
```

4 OSMASEM

4.1 Data preparation

```
# Modified based on the code from Jak & Cheung (2019)
## Exclude studies with missing values on Individualism
index_na <- is.na(Gnambs18$Individualism)
Gnambs18 <- lapply(Gnambs18, function(x) x[!index_na])

# Exclude studies that reported CFA results only
index <- Gnambs18$CorMat==1
Gnambs18 <- lapply(Gnambs18, function(x) x[index])

## Create a dataframe with the data and the asymptotic variances and covariances (acov)
my.df <- Cor2DataFrame(Gnambs18$data, Gnambs18$n, acov = "weighted")

## Add the standardized individualism as the moderator
## Standardization of the moderator improves the convergence.
my.df$data <- data.frame(my.df$data,
                         Individualism=scale(Gnambs18$Individualism),
                         check.names=FALSE)
summary(my.df)
```

```
##           Length Class      Mode
## data      1081   data.frame list
## n           34   -none-     numeric
## obslabels   10   -none-     character
## ylabels     45   -none-     character
## vlabels   1035   -none-     character
```

4.2 Model fitting

```
## Specify the bifactor model
model0 <- "G =~  g1*I1 + g2*I2 + g3*I3 + g4*I4 + g5*I5 +
                 g6*I6 + g7*I7 + g8*I8 + g9*I9 + g10*I10
          POS =~ p1*I1 + p3*I3 + p4*I4 + p7*I7 + p10*I10
          NEG =~ n2*I2 + n5*I5 + n6*I6 + n8*I8 + n9*I9"
RAM0 <- lavaan2RAM(model0, obs.variables = paste0("I", 1:10),std.lv = TRUE)


## Create heterogeneity variances
T0 <- create.Tau2(RAM=RAM0, RE.type="Diag", Transform="expLog", RE.startvalues=0.05)
```

```r
## Create the A1 matrix with moderator effects of "Individualism"
Ax1 <- RAM0$A
Ax1[grep("\\*", Ax1)] <- "0*data.Individualism"
Ax1
```

```
##     I1  I2  I3  I4  I5  I6  I7  I8  I9  I10 G
## I1  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I2  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I3  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I4  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I5  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I6  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I7  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I8  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I9  "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## I10 "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0*data.Individualism"
## G   "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## POS "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## NEG "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
##     POS                    NEG
## I1  "0*data.Individualism" "0"
## I2  "0"                    "0*data.Individualism"
## I3  "0*data.Individualism" "0"
## I4  "0*data.Individualism" "0"
## I5  "0"                    "0*data.Individualism"
## I6  "0"                    "0*data.Individualism"
## I7  "0*data.Individualism" "0"
## I8  "0"                    "0*data.Individualism"
## I9  "0"                    "0*data.Individualism"
## I10 "0*data.Individualism" "0"
## G   "0"                    "0"
## POS "0"                    "0"
## NEG "0"                    "0"
```

```r
## Create matrices with implicit diagonal constraints
M1 <- create.vechsR(A0=RAM0$A, S0=RAM0$S, F0=RAM0$F, Ax=list(Ax1))

## Fit the bifactor model with One-Stage MASEM
fit1 <- osmasem(model.name="Moderated by individualism", Mmatrix=M1,
                Tmatrix=T0, data=my.df)
summary(fit1, fitIndices= T)
```

```
## Summary of Moderated by individualism
##
## free parameters:
##       name  matrix row col      Estimate  Std.Error A      z value     Pr(>|z|)
## 1       g1      A0  I1   G  0.7267701507 0.01945695    37.35272726 0.000000e+00
## 2       g2      A0  I2   G  0.5513972665 0.01850697    29.79403360 0.000000e+00
## 3       g3      A0  I3   G  0.5685774763 0.02076070    27.38719721 0.000000e+00
## 4       g4      A0  I4   G  0.5043285283 0.01818280    27.73657039 0.000000e+00
## 5       g5      A0  I5   G  0.5462132466 0.02102773    25.97585717 0.000000e+00
## 6       g6      A0  I6   G  0.5307454018 0.01691066    31.38526261 0.000000e+00
## 7       g7      A0  I7   G  0.5959707117 0.01882232    31.66297311 0.000000e+00
## 8       g8      A0  I8   G  0.3917127022 0.01951388    20.07354156 0.000000e+00
```

```
## 9         g9      A0  I9   G  0.6162442886 0.02143710    28.74661894 0.000000e+00
## 10       g10      A0 I10   G  0.7773662362 0.02123763    36.60324882 0.000000e+00
## 11        p1      A0  I1 POS  0.0032438158 0.05182178     0.06259561 9.500885e-01
## 12        p3      A0  I3 POS  0.5045006914 0.04440483    11.36139208 0.000000e+00
## 13        p4      A0  I4 POS  0.3537941419 0.03871764     9.13780118 0.000000e+00
## 14        p7      A0  I7 POS  0.3585442832 0.04183061     8.57133833 0.000000e+00
## 15       p10      A0 I10 POS  0.0157760266 0.05396921     0.29231534 7.700455e-01
## 16        n2      A0  I2 NEG  0.5690587050 0.02840268    20.03538689 0.000000e+00
## 17        n5      A0  I5 NEG  0.3003206055 0.03141487     9.55982189 0.000000e+00
## 18        n6      A0  I6 NEG  0.5804909678 0.02622181    22.13771907 0.000000e+00
## 19        n8      A0  I8 NEG  0.3904778253 0.03059096    12.76448263 0.000000e+00
## 20        n9      A0  I9 NEG  0.3578788477 0.03402054    10.51949237 0.000000e+00
## 21      g1_1      A1  I1   G  0.0875106295 0.02096142     4.17484321 2.981913e-05
## 22      g2_1      A1  I2   G  0.0221066335 0.01956045     1.13016972 2.584047e-01
## 23      g3_1      A1  I3   G  0.0727629290 0.01649127     4.41221020 1.023207e-05
## 24      g4_1      A1  I4   G  0.0353540106 0.01633820     2.16388678 3.047304e-02
## 25      g5_1      A1  I5   G  0.0006427825 0.02102243     0.03057603 9.756077e-01
## 26      g6_1      A1  I6   G  0.0200873872 0.01811576     1.10883510 2.675013e-01
## 27      g7_1      A1  I7   G  0.0727057407 0.01762023     4.12626420 3.687037e-05
## 28      g8_1      A1  I8   G  0.1228580011 0.02008026     6.11834697 9.455101e-10
## 29      g9_1      A1  I9   G -0.0168883628 0.02261616    -0.74673887 4.552212e-01
## 30     g10_1      A1 I10   G  0.0658329331 0.02271738     2.89791087 3.756573e-03
## 31      p1_1      A1  I1 POS -0.1620758723 0.03394846    -4.77417385 1.804465e-06
## 32      p3_1      A1  I3 POS -0.0600522886 0.02963874    -2.02614186 4.275025e-02
## 33      p4_1      A1  I4 POS -0.0424609008 0.02578751    -1.64656853 9.964676e-02
## 34      p7_1      A1  I7 POS -0.0749924361 0.03046906    -2.46126527 1.384480e-02
## 35     p10_1      A1 I10 POS -0.1798659236 0.03488781    -5.15555199 2.528849e-07
## 36      n2_1      A1  I2 NEG  0.0067380318 0.02955382     0.22799193 8.196525e-01
## 37      n5_1      A1  I5 NEG -0.0012606561 0.03076838    -0.04097246 9.673179e-01
## 38      n6_1      A1  I6 NEG  0.0045899643 0.02693411     0.17041456 8.646841e-01
## 39      n8_1      A1  I8 NEG  0.0329755683 0.03050530     1.08097843 2.797067e-01
## 40      n9_1      A1  I9 NEG  0.0650178696 0.03508332     1.85324187 6.384770e-02
## 41    Tau1_1 vecTau1   1   1 -4.9271860882 0.26167349   -18.82951959 0.000000e+00
## 42    Tau1_2 vecTau1   2   1 -5.2013970114 0.26651810   -19.51611173 0.000000e+00
## 43    Tau1_3 vecTau1   3   1 -5.0697804707 0.26572746   -19.07887328 0.000000e+00
## 44    Tau1_4 vecTau1   4   1 -4.4381217375 0.26399654   -16.81128724 0.000000e+00
## 45    Tau1_5 vecTau1   5   1 -5.8440955736 0.28841453   -20.26283335 0.000000e+00
## 46    Tau1_6 vecTau1   6   1 -4.6644024158 0.25776110   -18.09583538 0.000000e+00
## 47    Tau1_7 vecTau1   7   1 -4.1166659930 0.25211835   -16.32830784 0.000000e+00
## 48    Tau1_8 vecTau1   8   1 -5.0380671709 0.26464341   -19.03719090 0.000000e+00
## 49    Tau1_9 vecTau1   9   1 -4.6929422828 0.27976275   -16.77472157 0.000000e+00
## 50   Tau1_10 vecTau1  10   1 -5.2213032680 0.26368947   -19.80095508 0.000000e+00
## 51   Tau1_11 vecTau1  11   1 -5.3886483970 0.26781801   -20.12055994 0.000000e+00
## 52   Tau1_12 vecTau1  12   1 -5.3175763326 0.27293152   -19.48318849 0.000000e+00
## 53   Tau1_13 vecTau1  13   1 -4.6928265352 0.27681309   -16.95305133 0.000000e+00
## 54   Tau1_14 vecTau1  14   1 -4.9310293957 0.25530249   -19.31445867 0.000000e+00
## 55   Tau1_15 vecTau1  15   1 -4.4450994939 0.25726415   -17.27834804 0.000000e+00
## 56   Tau1_16 vecTau1  16   1 -5.4009803526 0.28186559   -19.16154540 0.000000e+00
## 57   Tau1_17 vecTau1  17   1 -5.2106450385 0.25906431   -20.11332612 0.000000e+00
## 58   Tau1_18 vecTau1  18   1 -5.6057411516 0.27704858   -20.23378429 0.000000e+00
## 59   Tau1_19 vecTau1  19   1 -4.7494537988 0.27057640   -17.55309717 0.000000e+00
## 60   Tau1_20 vecTau1  20   1 -5.5666747577 0.28104726   -19.80689946 0.000000e+00
## 61   Tau1_21 vecTau1  21   1 -4.0983812880 0.25475259   -16.08769218 0.000000e+00
## 62   Tau1_22 vecTau1  22   1 -4.5994789668 0.26616279   -17.28069891 0.000000e+00
```

```
## 63 Tau1_23 vecTau1  23   1 -5.2994763262 0.27002978   -19.62552579 0.000000e+00
## 64 Tau1_24 vecTau1  24   1 -5.4141004999 0.27442890   -19.72860938 0.000000e+00
## 65 Tau1_25 vecTau1  25   1 -5.1905445326 0.27720019   -18.72489561 0.000000e+00
## 66 Tau1_26 vecTau1  26   1 -5.5475442122 0.27269366   -20.34350264 0.000000e+00
## 67 Tau1_27 vecTau1  27   1 -4.4759907086 0.26606269   -16.82306813 0.000000e+00
## 68 Tau1_28 vecTau1  28   1 -4.4100062278 0.26314104   -16.75909691 0.000000e+00
## 69 Tau1_29 vecTau1  29   1 -5.1406347790 0.26838024   -19.15429713 0.000000e+00
## 70 Tau1_30 vecTau1  30   1 -5.6316052979 0.27561830   -20.43262481 0.000000e+00
## 71 Tau1_31 vecTau1  31   1 -5.6163828994 0.28702931   -19.56728026 0.000000e+00
## 72 Tau1_32 vecTau1  32   1 -4.8097952773 0.26030576   -18.47748289 0.000000e+00
## 73 Tau1_33 vecTau1  33   1 -4.9557632008 0.27108639   -18.28112149 0.000000e+00
## 74 Tau1_34 vecTau1  34   1 -4.2421453256 0.26063011   -16.27649731 0.000000e+00
## 75 Tau1_35 vecTau1  35   1 -4.6718429488 0.27750431   -16.83520888 0.000000e+00
## 76 Tau1_36 vecTau1  36   1 -5.5232542043 0.26584452   -20.77625775 0.000000e+00
## 77 Tau1_37 vecTau1  37   1 -4.3851545391 0.25997633   -16.86751435 0.000000e+00
## 78 Tau1_38 vecTau1  38   1 -5.8431226475 0.28735207   -20.33436743 0.000000e+00
## 79 Tau1_39 vecTau1  39   1 -5.9169427985 0.27723470   -21.34272062 0.000000e+00
## 80 Tau1_40 vecTau1  40   1 -4.5048884607 0.25510446   -17.65899506 0.000000e+00
## 81 Tau1_41 vecTau1  41   1 -4.9724734012 0.25648662   -19.38687282 0.000000e+00
## 82 Tau1_42 vecTau1  42   1 -5.0623598730 0.25638637   -19.74504309 0.000000e+00
## 83 Tau1_43 vecTau1  43   1 -4.5778672215 0.26176672   -17.48834697 0.000000e+00
## 84 Tau1_44 vecTau1  44   1 -3.9722141050 0.25344811   -15.67269204 0.000000e+00
## 85 Tau1_45 vecTau1  45   1 -5.0808069935 0.26542052   -19.14248001 0.000000e+00
##
## To obtain confidence intervals re-run with intervals=TRUE
##
## Model Statistics:
##                |  Parameters  |  Degrees of Freedom  |  Fit (-2lnL units)
##        Model:            85                   1445              -2882.282
##    Saturated:            90                   1440              -2647.332
## Independence:            45                   1485               1502.869
## Number of observations/statistics: 104684/1530
##
## chi-square:  ² ( df=5 ) = -234.9505,  p = 1
## Information Criteria:
##       |  df Penalty  |  Parameters Penalty  |  Sample-Size Adjusted
## AIC:      -5772.282             -2712.282                -2712.142
## BIC:     -19584.606             -1899.792                -2169.925
## CFI: 1.05845
## TLI: 1.526053   (also known as NNFI)
## RMSEA: 0 *(Non-centrality parameter is negative)  [95% CI (0, 0)]
## Prob(RMSEA <= 0.05): 1
## timestamp: 2023-12-12 16:22:05
## Wall clock time: 356.5459 secs
## optimizer:  SLSQP
## OpenMx version number: 2.21.8
## Need help?  See help(mxSummary)
```