Companion material

A Tutorial on Systematic Methodology for Psychometric Model Selection: Case Study of the European Obesity Academy Questionnaire — Expectations of Surgical Treatment

Paulo Sergio Panse Silveira and Jose de Oliveira Siqueira

Department of Pathology, School of Medicine, University of Sao Paulo, Sao Paulo, SP, Brazil

September 13, 2024

Contents

What is Confirmatory Factor Analysis? **3**

Descriptive statistics **4**

Confirmatory Factor Analysis in R **7**

Implementation of CFA **10**
Unifactor model, preliminary analysis 14
Unifactor model, configural invariance 18
Bifactor model, preliminary and configural invariance 20
Bifactor model, metric invariance 26
Bifactor model, scalar invariance 29
– partial scalar invariance 31
Bifactor model, strict invariance 36
Sintaxe for first and second order models 38

Final words **42**

This tutorial explains the process of selecting the best candidate models, utilizing data from 137 obese female patients. Data and R scripts are available at Harvard Dataverse (`https://doi.org/10.7910/DVN/FCBOGB`)

What is Confirmatory Factor Analysis?

By presenting various model possibilities derived from either expert opinions or exploratory factor analysis (EFA), validation relies on the application of Confirmatory Factor Analysis (CFA). Unlike EFA, CFA requires explicitly specifying the model by connecting the common factors to the desired items. It is important to note that EFA is not mandatory before conducting CFA, as research supports the independence of these two methodologies[1].

Measurement invariance is a general term that can be applied to various components of measurement models.

There are two categories of invariance:

1. related to the psychometric properties of the measurement scales[2], and includes
    - **Configural** invariance (Model 0): the base model.
    - **Metric** or weak invariance (Model 1): factor loadings are constrained to be equal.
    - **Scalar** or strong invariance (Model 2): equal item intercepts in addition to metric invariance.
    - **Strict invariance** (Model 3): equal measurement error variances.
2. related to variances and covariances of latent means (not discussed here).

The first category is the construct validation, estimating and testing parameters related to the items. The first category of invariance is a prerequisite for interpreting the second category, which tests to between-group differences[3]. In this text, the focus is on the construct validation only, which confirms that both groups are assessed using the same psychometric model (informally considered as the same 'rule').

The tests for construct validation models are hierarchical, with each level nested within the previous one. Unlike traditional statistics, the goal in Confirmatory Factor Analysis (CFA) is to find a model that fits the data well. Therefore, it is desirable not to reject the null hypothesis; in other words, non-rejection is treated as acceptance of the model[4]. Once a certain level of invariance is accepted, it functions as an assumption for the next (nested) test.

The analysis starts with the measurement model. The so-called Model 0 is the test of configural invariance, verifying the equivalence of the factor structure (i.e., equal forms) between independent conditions (groups). Since this first level is not nested within a previous reference model, the test is direct, and we will use the p-value of the root mean square error approximation (RMSEA) in its original or robust versions.

Metric invariance (Model 1) verifies the equivalence of factor loadings between conditions. When accepted, it suggests that the common factors have the same meanings for the participants, i.e., for individuals from different groups, the factors influence the items in the same way.

For scalar invariance (Model 2), the equality of item (manifest) means is evaluated. However, these sample means are not directly used; instead, the means estimated from the psychometric model (model-estimated parameters) are used. Full scalar invariance may be rejected, but the requirement for equality of item parameters can be relaxed through partial scalar invariance tests. One approach is to identify items with the most discrepant estimated means and relax their constraints one by one until acceptable invariance is achieved.

Finally, strict invariance (Model 3) is analyzed. Ideally, this test assumes full scalar invariance. In the absence of this, partial scalar invariance can be taken as the reference model, provided that we have not had to relax the equality of many items. Strict invariance implies that the items are equally precise (with equal reliability) between conditions.

If strict invariance is achieved, scores obtained by summing the observed item values would be valid for comparing groups, as their mean values and precisions are equal across groups. This is rarely achieved, and psychometric models are necessary for calculating the factorial score of each individual. In other words, if the psychometric instrument has strict invariance, there is no instrument effect, and any measurement differences are due to the group or experimental condition effect.

Since each test depends on the results of the previous ones, the procedure is carried out at each step until the model is rejected. When the model is eventually rejected, it is said that validation was achieved up to the last accepted level. If more than one model is accepted at the same level, criteria related to the quality of the model fit can be used to rank them.

Descriptive statistics

This example utilizes data from a study focused on the translation and cross-cultural adaptation of the European Obesity Academy Questionnaire: Expectations about Surgical Treatment (EOAQ-EST) into Brazilian Portuguese. The data comprises 137 female patients with obesity who are candidates for bariatric surgery. Given the general association between menopause and weight gain[5], two age groups were created to demonstrate the process for assessing construct validity at higher levels: women under 48 years old and those over 48 years old. This division is based on the median age of menopause among Brazilian women[6]. Data are available in `data.xlsx` (Excel format).

For any analysis, one or more procedures must be carried out to verify the integrity of the data. For example, the following R code performs some of the main assessments.

```
# CFA_descriptive.R

# data
data <- readxl::read_excel("data.xlsx")
data$ID <- as.character(data$ID)
data$Group <- factor(data$Group)
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

# sink("CFA_descriptive.txt")

cat("\n*** Descriptive statistics ***\n")

# n
print(table(data$Group))

# Sex
data$Sex <- factor(data$Sex)
print(table(data$Sex))

# BMI
cat("\nBMI\n")
cat("\n\tless or equal to 48\n")
print(summary(data$BMI[data$Group=="less or equal to 48"]))
print(sd(data$BMI[data$Group=="less or equal to 48"]))
cat("\n\tmore than 48\n")
print(summary(data$BMI[data$Group=="more than 48"]))
print(sd(data$BMI[data$Group=="more than 48"]))
# Schooling
cat("\nStatus\n")
data$Status <- factor(data$Status,
                          levels=c("Single",
                                   "Married",
                                   "Common-law",
                                   "Separated",
                                   "Divorced",
                                   "Widowed")
                         )
print(table(data$Group,data$Status))
# Schooling
cat("\nSchool\n")
data$Schooling <- factor(data$Schooling,
                          levels=c("Elementary",
                                   "High",
                                   "Technical",
                                   "College/University")
)
print(table(data$Group,data$Schooling))
# Working
cat("\nWorking\n")
data$Working <- factor(data$Working,
                          levels=c("Unemployed",
                                   "Medical leave",
                                   "Homemaker",
                                   "Currently working",
                                   "Retired")
)
print(table(data$Group,data$Working))
# Diabetes
cat("\nDiabetes\n")
data$Diabetes <- factor(data$Diabetes)
print(table(data$Group,data$Diabetes))
# Hypertension
cat("\nHypertension\n")
data$Hypertension <- factor(data$Hypertension)
print(table(data$Group,data$Hypertension))
```

```
# Checking
cat("\nNumber of valid answers (checking for missing data)\n")
dt <- data.frame(sapply(data,function(x){sum(!is.na(x))}))
names(dt) <- NULL
print(dt)

# Descriptive
responses <- paste0("Q",1:14)
cat("\nDescriptive statistics (psych::describe)\n")
dt <- as.data.frame(psych::describe(data[,c(responses)]))
print(dt)

# Correlation matrix
cat("\nCorrelation matrix (polycor::hetcor)\n")
Items <- data.frame(data[,responses])
r.het <- polycor::hetcor(Items,use="complete.obs")
print(round(r.het[[1]],3))

# sink()
```

It generates the following output:

```
*** Descriptive statistics ***

less or equal to 48        more than 48
                 82                  55

Female
   137

BMI

 less or equal to 48
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  33.79   45.19   48.73   49.88   54.10   80.47
[1] 8.525376

 more than 48
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  32.46   40.42   46.09   46.16   51.47   71.92
[1] 7.526294

Status

                     Single Married Common-law Separated Divorced Widowed
  less or equal to 48     34      30          7         1        8       2
  more than 48             8      28          2         1        4      12

School

                     Elementary High Technical College/University
  less or equal to 48         14   41         4                 23
  more than 48                24   21         1                  9

Working

                     Unemployed Medical leave Homemaker Currently working Retired
  less or equal to 48         15             1        12                53       1
  more than 48                 1             4        23                19       8

Diabetes

                     No Yes
  less or equal to 48 58  24
  more than 48        24  31
```

```
Hypertension

                     No Yes
  less or equal to 48 32  50
  more than 48        10  45

Number of valid answers (checking for missing data)

ID           137
Group        137
Sex          137
Weight       137
Height       137
BMI          137
Status       137
Schooling    137
Working      137
Diabetes     137
Hypertension 137
Q1           137
Q2           137
Q3           137
Q4           137
Q5           137
Q6           137
Q7           137
Q8           137
Q9           137
Q10          137
Q11          137
Q12          137
Q13          137
Q14          137

Descriptive statistics (psych::describe)
    vars   n     mean        sd median  trimmed    mad min max range       skew   kurtosis         se
Q1     1 137 4.540146 1.0074327      5 4.810811 0.0000   1   5     4 -2.3362611  4.5381892 0.08607078
Q2     2 137 4.788321 0.4444206      5 4.873874 0.0000   3   5     2 -1.8828871  2.6753938 0.03796941
Q3     3 137 3.496350 1.4709145      4 3.612613 1.4826   1   5     4 -0.5708839 -1.0432260 0.12566871
Q4     4 137 4.335766 1.0591157      5 4.558559 0.0000   1   5     4 -1.6518331  1.9665852 0.09048636
Q5     5 137 1.233577 0.8158609      1 1.000000 0.0000   1   5     4  3.6669716 12.7013029 0.06970370
Q6     6 137 3.503650 1.4759049      4 3.621622 1.4826   1   5     4 -0.4498104 -1.2364254 0.12609507
Q7     7 137 4.810219 0.6363134      5 4.990991 0.0000   1   5     4 -3.7315772 14.4506174 0.05436393
Q8     8 137 4.890511 0.3962848      5 5.000000 0.0000   2   5     3 -4.4340966 23.3080776 0.03385690
Q9     9 137 4.489051 1.0786483      5 4.765766 0.0000   1   5     4 -2.1704767  3.6760179 0.09215514
Q10   10 137 3.562044 1.5943896      4 3.693694 1.4826   1   5     4 -0.5443438 -1.3558547 0.13621790
Q11   11 137 4.175182 1.2998745      5 4.450450 0.0000   1   5     4 -1.4805661  0.8650035 0.11105577
Q12   12 137 4.905109 0.3404843      5 5.000000 0.0000   3   5     2 -3.7714226 14.5947469 0.02908954
Q13   13 137 4.583942 0.8881767      5 4.801802 0.0000   1   5     4 -2.2213412  4.2795218 0.07588206
Q14   14 137 4.912409 0.4770382      5 5.000000 0.0000   1   5     4 -6.3053423 42.3454229 0.04075613

Correlation matrix (polycor::hetcor)
        Q1     Q2     Q3     Q4     Q5     Q6     Q7     Q8    Q9    Q10    Q11    Q12    Q13    Q14
Q1   1.000  0.307  0.344  0.353  0.114  0.360 -0.011  0.370 0.215  0.235  0.129  0.408  0.557  0.130
Q2   0.307  1.000  0.162  0.261 -0.106  0.231  0.169  0.201 0.233  0.086  0.090  0.206  0.315  0.051
Q3   0.344  0.162  1.000  0.237 -0.012  0.416  0.094  0.094 0.152  0.234  0.189  0.153  0.283  0.062
Q4   0.353  0.261  0.237  1.000  0.028  0.225  0.128  0.053 0.106  0.205  0.101  0.273  0.345 -0.043
Q5   0.114 -0.106 -0.012  0.028  1.000 -0.007 -0.169  0.034 0.045  0.209  0.162  0.054  0.013  0.015
Q6   0.360  0.231  0.416  0.225 -0.007  1.000  0.056  0.145 0.269  0.326  0.164  0.198  0.397  0.021
Q7  -0.011  0.169  0.094  0.128 -0.169  0.056  1.000  0.209 0.308 -0.119 -0.093  0.290  0.041  0.114
Q8   0.370  0.201  0.094  0.053  0.034  0.145  0.209  1.000 0.315  0.040 -0.020  0.195  0.141  0.066
Q9   0.215  0.233  0.152  0.106  0.045  0.269  0.308  0.315 1.000  0.070  0.085  0.247  0.245  0.098
Q10  0.235  0.086  0.234  0.205  0.209  0.326 -0.119  0.040 0.070  1.000  0.396 -0.036  0.146  0.133
Q11  0.129  0.090  0.189  0.101  0.162  0.164 -0.093 -0.020 0.085  0.396  1.000 -0.045 -0.032 -0.082
Q12  0.408  0.206  0.153  0.273  0.054  0.198  0.290  0.195 0.247 -0.036 -0.045  1.000  0.452  0.220
Q13  0.557  0.315  0.283  0.345  0.013  0.397  0.041  0.141 0.245  0.146 -0.032  0.452  1.000  0.122
Q14  0.130  0.051  0.062 -0.043  0.015  0.021  0.114  0.066 0.098  0.133 -0.082  0.220  0.122  1.000
```

Confirmatory Factor Analysis

This section shows how to apply the `lavaan` package version 0.6.14[7] in R to invariance analysis.

Since its inception, `lavaan` has become a popular choice among researchers and statisticians for conducting confirmatory factor analysis (CFA), structural equation modeling (SEM), and other related analyses. The package supports complex model structures, allowing for multiple groups, multilevel data, and mixture models. Additionally, it offers extensive functionality for model assessment, including robust fit indices and support for various estimation methods. It can be integrated with other R packages to further enhance its capabilities, such as `semTools` and `semPlot` applied below.

The `lavaan` package was developed and is currently maintained at CRAN by Yves Rosseel, a professor at Ghent University in Belgium, and has received several collaborations from the broader R community, enhancing its functionality and robustness for various applications in latent variable modeling. Its intuitive syntax, which is easy to learn and use, is inspired by the general approach of structural equation modeling software like Mplus but has its own unique syntax tailored to the R programming environment.

In order to explore several possibilities, the main types of psychometric models can be taken into account, according to their structure (Figure 1):

- Unidimensional Model: This is the simplest model, where a general factor influences all items.
- First-Order Model: At the other end of the spectrum, this model proposes independent specific factors with possible correlations to explain subgroups of items.
- Bifactor Model: This model combines elements of both unidimensional and first-order models by proposing a general factor and specific factors that simultaneously influence items. In this model, there is no correlation between the specific factors, nor between the specific factors and the general factor.
- Second-Order Model: This is another intermediate model, proposing a hierarchy where a general factor is mediated by specific factors that influence the items.

In structural equation modeling diagrams, specific notations are used to visually represent different components of the model in a path diagram, summarizing the hypothesized relationships among latent and observed variables:

- Circles (or Ovals): These represent latent variables. Latent variables are not directly observed but are inferred from observed data. They represent

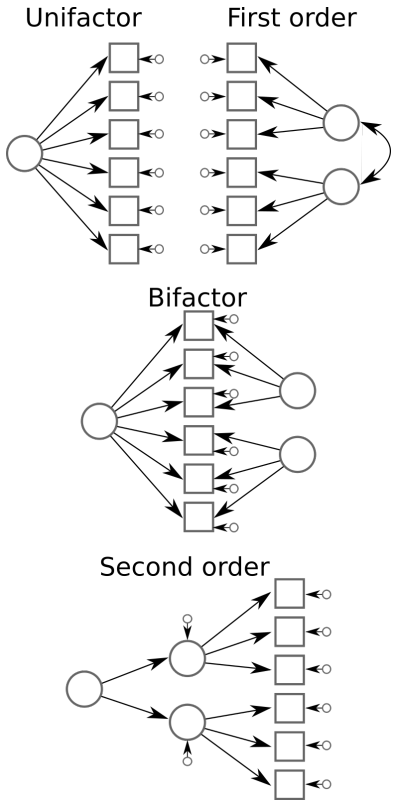


Figure 1: Path diagram of the main model structures.

underlying constructs that are theorized to explain the patterns in the observed data. For example, in a psychological study, latent variables might represent abstract concepts like intelligence, motivation, or anxiety.

- Squares (or Rectangles): These represent observed variables. Observed variables are the measurable items or indicators collected during the study. These could include survey responses, test scores, or other empirical data that are used to measure the latent variables.
- Triangles: Tipically used to denote a constant or a fixed parameter within the model. This symbol is used to indicate that the value it points to is not estimated from the data but is a fixed part of the model structure.
- Arrows:
  - Single-headed arrows indicate causal relationships or regressions. They show the direction of influence from one variable to another. For example, an arrow from a latent variable to an observed

variable indicates that the latent variable is hypothesized to influence the observed variable.

  - Double-headed arrows represent correlations or covariances between variables. They indicate a relationship between two variables without specifying the direction of causality.

- Error Terms: These are depicted as circles attached to observed variables, with single-headed arrows pointing from the error term to the observed variable (represented in Figure 1 as smaller circles and arrows). These error terms are, themselves, latent variables and they represent the measurement error or the variance in the observed variable that is not accounted for by the latent variable. Typically, error terms are often omitted in path diagrams to maintain a cleaner and more straightforward visual presentation.

Implementation of CFA

The R codes in the following continue the analysis of the EOAQ-EST data, by implementing a full example of each model structure at all validation levels. In order to simplify the analysis, two generic scripts were developed:

- `CFA.R` to compute the models:

```
# CFA.R

group.equal <- ""
intercepts <- FALSE
residuals <- FALSE
layout <- "tree3"
if(invariance == "configural")
{
  group.equal <- ""
}
if(invariance == "metric")
{
  group.equal <- "loadings"
}
if(invariance == "scalar")
{
  group.equal <- c("loadings","intercepts")
  intercepts <- TRUE
  layout <- "tree"
  bifactor <- ""
}
if(invariance == "strict")
{
  group.equal <- c("loadings","intercepts","residuals")
  intercepts <- TRUE
  residuals <- TRUE
  layout <- "tree"
  bifactor <- ""
}

if(!is.na(filename))
```

```
{sink(paste0(filename,".txt"))}

cat("\n*** Confirmatory Factor Analysis: ",title," ***")
if(is.null(group)) # previous analysis
{cat("\n      (preliminary analysis)\n")} else
{cat("\n      (group.equal = ",paste0(group.equal,collapse=", "),")\n",sep="")}

# show current model
cat(model)

o.data <- data
groups <- ""
if(is.null(group)) # previous analysis
{
  groups <- c("All",levels(data$Group))
}
for (g.aux in groups)
{
  data <- o.data
  if(nchar(g.aux)>0 & g.aux!="All")
  {
    data <- subset(data,Group==g.aux)
  }
  if(is.null(group))
  {
    cat("\n\n-----------\n",g.aux,"\n-----------\n",sep="")
  }
  model.estim <- lavaan::cfa(model,
                             estimator="MLR",
                             optim.method="BFGS",
                             missing="fiml",
                             std.lv=TRUE,
                             data=data,
                             group=group,
                             group.equal=group.equal)
  fitmeasure <- lavaan::fitMeasures(model.estim,output="text")
  # print(fitmeasure) # uncomment to see the full output
  lavsummary <- lavaan::summary(model.estim, standardized=TRUE)
  # print(lavsummary) # uncomment to see the full output

  cat("\nEXACT TEST (robust):\n")
  X2 <- as.numeric(fitmeasure["chisq.scaled"])
  df <- as.numeric(fitmeasure["df.scaled"])
  # metric, scalar, strict

  # heuristic
  X2.by.df <- X2/df
  # p value of X^2
  X2.p <- pchisq(q=X2,df=df,lower.tail=FALSE)
  cat("\n* X^2(",df,"): ",X2,sep="")
  cat("\n* p: ",X2.p,sep="")

  cat("\n\nTEST WITH TOLERANCE (robust):\n")
  RMSEA <- fitmeasure["rmsea.robust"]
  RMSEA.lwr <- fitmeasure["rmsea.ci.lower.robust"]
  RMSEA.upr <- fitmeasure["rmsea.ci.upper.robust"]
  RMSEA.pclose <- fitmeasure["rmsea.pvalue.robust"]
  H0.pclose <- ifelse(RMSEA.pclose<0.05,"rejected","not rejected")
  acceptance <- "plausible"
  if(RMSEA.upr < 0.05) {acceptance <- "highly plausible"}
  if(RMSEA.lwr > 0.05) {acceptance <- "not plausible"}
  cat("\n* RMSEA: ",round(RMSEA,3),
      " [",round(RMSEA.lwr,3),",",round(RMSEA.upr,3),"]",sep="")
  cat("\n* pclose: ",RMSEA.pclose,sep="")

  # reliability analysis
  cat("\n")
```

```
cat("\nReliability analysis (McDonald's omega):\n")
factors <- c("G","P","S","H")
dt.reliability <- semTools::compRelSEM(model.estim)
if(is.null(group))
{
  names <- c("group",names(dt.reliability))
  dt.reliability <- data.frame(matrix(data=c(g.aux,as.vector(dt.reliability)),
                                      nrow=1,ncol=length(names)))
  names(dt.reliability) <- names
}
for(o.aux in factors)
{
  if(sum(names(dt.reliability)==o.aux)==0)
  {
    dt.reliability$new <- NA
    names(dt.reliability) <- c(names(dt.reliability)[1:(ncol(dt.reliability)-1)],o.aux)
  }
}
dt.reliability <- dt.reliability[,c("group",factors)]
print(dt.reliability)
omega.G1 <- dt.reliability$G[1]
omega.P1 <- dt.reliability$P[1]
omega.S1 <- dt.reliability$S[1]
omega.H1 <- dt.reliability$H[1]
omega.G2 <- dt.reliability$G[2]
omega.P2 <- dt.reliability$P[2]
omega.S2 <- dt.reliability$S[2]
omega.H2 <- dt.reliability$H[2]
omega.valid <- TRUE
if(
  sum(dt.reliability[,2:ncol(dt.reliability)]<0,na.rm=TRUE)>0 |
  sum(dt.reliability[,2:ncol(dt.reliability)]>1,na.rm=TRUE)>0 |
  sum(is.na(dt.reliability[,2:ncol(dt.reliability)])>0)>0 |
  sum(!is.finite(as.numeric(unlist(dt.reliability[,2:ncol(dt.reliability)]))))>0
)
{
  omega.valid <- FALSE
}

# model selection
CFI <- fitmeasure["cfi.robust"]
TLI <- fitmeasure["tli.robust"]
PNFI <- fitmeasure["pnfi.scaled"]
AGFI <- fitmeasure["agfi"]
SABIC <- fitmeasure["bic2"]
cat("\n")
cat("\nModel selection:\n")
cat("\n* CFI = ",CFI,sep="")
cat("\n* TLI = ",TLI,sep="")
cat("\n* PNFI = ",PNFI,sep="")
cat("\n* SABIC = ",SABIC,sep="")

# graphic representation
if(!is.na(filename))
{
  sufix <- ""
  if(what!="std") # non-standard estimates
  {
    sufix <- "_est"
  }
  filepng <- paste0(filename,sufix,"_",g.aux,".png")
} else
{
  filepng <- NA
}

# graphic representation
```

```
  if(is.null(group)) # previous analysis
  {
    main <- paste0(title,": ",g.aux)
  } else
  {
    main <- paste0(title,":\nage ",lavsummary$data$group.label)
  }
  source("CFA_graphic.R")
}
cat("\n")

if(!is.na(filename))
{
  sink()
  cat("\nReport written on ",filename,".txt\n",
      "and image(s) stored in ",filename,"*.png\n",
      sep="")
}

# results
dt.result <- data.frame(
  X2, df, X2.p, X2.by.df,
  RMSEA, RMSEA.lwr, RMSEA.upr, RMSEA.pclose, H0.pclose,
  acceptance, invariance,
  CFI, PNFI, AGFI, SABIC,
  omega.G1, omega.P1, omega.S1, omega.H1,
  omega.G2, omega.P2, omega.S2, omega.H2, omega.valid)
```

- `CFA_graphic.R`, which is its companion to show the graphic representation of the models:

```
# CFA_graphic.R

extra.width <- 1
if(intercepts | residuals){extra.width <- 1.25}
if(!is.na(filepng))
{png(filepng, width=600*length(main)*extra.width, height=600)}
o.par <- par(no.readonly=TRUE)
m <- matrix(data=1:(2*length(main)),
            nrow=2,ncol=length(main),byrow=TRUE)
layout(m,heights=0.15,1)
par(mai=c(0,0,0,0))
for (i in 1:length(main))
{
  plot(NA,xlim=c(0,10),ylim=c(0,1),
       main="",xlab="",ylab="",axes=FALSE)
  text(5,1,main[i],cex=1.5,pos=1)
}
grf <- semPlot::semPaths(model.estim,
                         bifactor=bifactor,
                         ask = FALSE,
                         style="lisrel",
                         what=what,
                         whatLabels=what,
                         layout=layout,
                         rotation=2,
                         edge.color="black",
                         intercepts=intercepts,
                         residuals=residuals,
                         edge.label.cex=1.2,
                         curvePivot=TRUE,
                         intAtSide=FALSE,
                         fade=FALSE,
                         sizeLat=8,
```

```
                        nCharNodes=5,
                        sizeInt=3,
                        sizeMan=5,
                        label.cex=1.5,
                        thresholds=FALSE,
                        title=FALSE)
par(o.par)
if(!is.na(filepng))
{dev.off()}
```

The core of these scripts is composed of three `lavaan` functions: `lavaan::cfa` to compute the model estimates, and `lavaan::fitMeasures` and `lavaan::summary` to report the results. The outputs of these last two functions are detailed; thus, our script suggests some main estimates that can be used for statistical decision-making. One may explore the outputs according to their needs. The proper calls to `CFA.R` (which, in turn, calls `CFA_graphic.R`) are shown below.

Unifactor model, preliminary analysis

Before initiating construct validation, let's apply a global model to both the entire sample and each subgroup. For statistical decisions, it is recommended to use the Bonferroni correction when testing each group separately. Specifically, this involves comparing the *RMSEA pclose* value with $\alpha/2$ for two, or $\alpha/3$ for groups, etc. It is implemented by `CFA_unifactor_previous.R`:

```
# CFA_unifactor_previous.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_unifactor_previous" # base filename without .ext

# Unifactor model
model <- "
G =~ Q1+Q2+Q3+Q4+Q5+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14
"
title <- paste0("Unifactor model")
group <- NULL
invariance <- "preliminary"
bifactor <- ""
what <- "std" #"est"

# CFA
source("CFA.R")
```

This prepares for call `CFA.R`, defining:

- the observed data (which, in this example, is in an Excel file named `data.xlsx`;

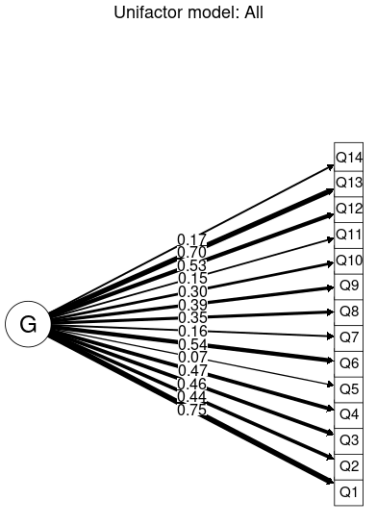
- the output is defined by `filename`, which can be sent to screen (`filename <- NA`) or to files (report goes to a file named `<filename>.txt` and graphics go to `<filename>*.png`;
- the `model` structure with the proper sintax (in this example, an unifactor model with a general factor, `G`);
- the `title` that will be used for the report and graphics;
- the parameters `group`, and `group.equal` for the function `lavaan::cfa` implemented on `CFA.R`, and `bifactor` and `what` for `semPlot::semPaths` on `CFA_graphic.R` (it only applies to bifactor models, below).

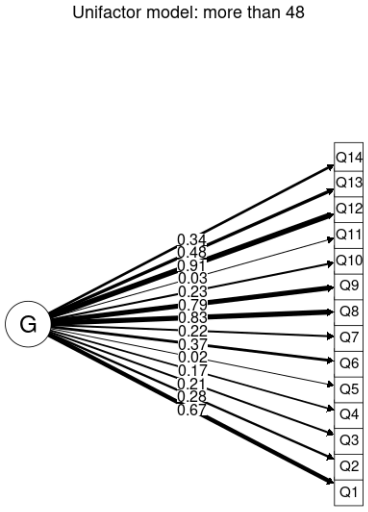
by calling `CFA_unifactor_previous.R`, it is generated:

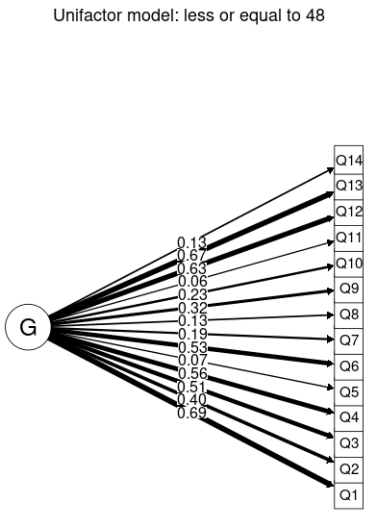
```
*** Confirmatory Factor Analysis:  Unifactor model  ***
      (preliminary analysis)

G =~ Q1+Q2+Q3+Q4+Q5+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14


-----------
All
-----------

EXACT TEST (robust):

* X^2(77): 149.7978
* p: 1.344581e-06

TEST WITH TOLERANCE (robust):

* RMSEA: 0.078 [0.054,0.1]
* pclose: 0.03000436

Reliability analysis (McDonald's omega):
  group                 G  P  S  H
1   All 0.646025403013114 NA NA NA


Model selection:

* CFI = 0.7642517
* TLI = 0.7213883
* PNFI = 0.4686339
* SABIC = 4537.704

-----------
less or equal to 48
-----------

EXACT TEST (robust):

* X^2(77): 140.116
* p: 1.482316e-05

TEST WITH TOLERANCE (robust):

* RMSEA: 0.091 [0.061,0.118]
* pclose: 0.0147078

Reliability analysis (McDonald's omega):
```

```
               group               G  P  S  H
1 less or equal to 48 0.61031240338967 NA NA NA


Model selection:

* CFI = 0.6803205
* TLI = 0.622197
* PNFI = 0.3791139
* SABIC = 2651.41

-----------
more than 48
-----------

EXACT TEST (robust):

* X^2(77): 319.2997
* p: 2.963366e-31

TEST WITH TOLERANCE (robust):

* RMSEA: 0.177 [0.145,0.209]
* pclose: 8.370508e-09

Reliability analysis (McDonald's omega):
         group                 G  P  S  H
1 more than 48 0.402025918825654 NA NA NA


Model selection:

* CFI = 0.4795493
* TLI = 0.3849218
* PNFI = 0.1563769
* SABIC = 1471.371
```

and the following images:







It is observed, even in this preliminary analysis, that the model is rejected for the three situations ($RMSEA.pclose < 0.05$ from the attempts with data of the whole set and of each group; the same conclusion for the robust version of $RMSEA$). It suggests that the construct validation might not be achieved. However, to provide a comprehensive guide, the analysis will proceed up to the strict invariance model in the following examples.

Unifactor model, configural invariance

It is implemented by `unifactor_configural.R`. This script is very similar `CFA_unifactor_previous.R`, with two important differences:

1. the parameters `group` is provided, becoming:

```
group <- "Group"
```

   Since the configural level of invariance is the base for the following models, it has no assumptions regarding equality of factor loadings, item means or measurement error variances, which is informed with the parameter `group.equal=""`. Thus, the core of `CFA.R` executes:

```
model.estim <- lavaan::cfa(model,
                           estimator="MLR",
                           optim.method="BFGS",
                           missing="fiml",
                           std.lv=TRUE,
                           data=data,
                           group="Group",
                           group.equal="")
```

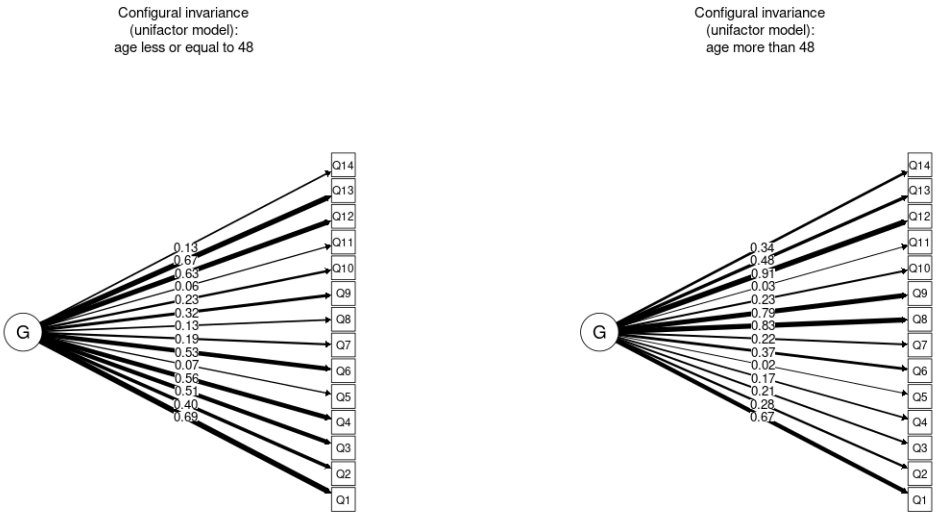
2. for the output, the parameter model estimates are computed simultaneously for both groups, thus only a set of estimates and a single graph are created.

Thus,

```
# CFA_unifactor_configural.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_unifactor_configural" # base filename without .ext

# Unifactor model
model <- "
G =~ Q1+Q2+Q3+Q4+Q5+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14
"
title <- paste0("Configural invariance\n(unifactor model)")
group <- "Group"
group.equal <- "configural"
bifactor <- ""
what <- "std" #"est"

# CFA
source("CFA.R")
```

generates:

```
*** Confirmatory Factor Analysis:  Configural invariance
(unifactor model)  ***
      (group.equal = )

G =~ Q1+Q2+Q3+Q4+Q5+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14

EXACT TEST (robust):

* X^2(154): 433.7164
* p: 1.902791e-28

TEST WITH TOLERANCE (robust):

* RMSEA: 0.132 [0.112,0.153]
* pclose: 3.830097e-09

Reliability analysis (McDonald's omega):
                group     G  P  S  H
1 less or equal to 48 0.610 NA NA NA
2        more than 48 0.402 NA NA NA


Model selection:

* CFI = 0.5577232
* TLI = 0.4773093
* PNFI = 0.2740556
* SABIC = 4181.375
```



In this example, the null hypothesis is rejected by the exact test with $X^2(77) = 357.115, p = 3.56 \cdot 10^{-18}$.

This chi-squared test, due to its zero-tolerance nature, tends to reject nearly all models, irrespective of their plausibility. Therefore, psychometric methods often utilize a tolerance level of 0.05 to test the null hypothesis of a population's existence, referred to as *pclose*. Alternatively, the 90% confidence interval of the robust root mean square error of approximation (RMSEA) can be used. RMSEA evaluates the model's fit to the data, determining its suitability for the intended population. A model is deemed **highly plausible** if the upper bound of this confidence interval is below 0.05, **plausible** if 0.05 falls within the interval, and **not plausible** if the lower bound is above 0.05[4,7,8]. In this example,
$RMSEA = 0.139\ [0.12, 0.158], pclose = 2.63 \cdot 10^{-12}$
or, by the more robust method,
$RMSEA = 0.132\ [0.112, 0.153], pclose = 3.83 \cdot 10^{-9}$,
thus rejecting the null hypothesis of adherence of the unifactor model to the data.

Again, this analysis could stop here: the unifactor model is not a choice to explain the obesity data presented here, therefore some other model, with a different structure, must be attempted.

Bifactor model, preliminary and configural invariance

In contrast, the best model found with the present data was a bifactor model without item Q5 and three specific factors. The main change in the previous

scripts is the model:

```
model <- "
G =~ Q1+Q2+Q3+Q4+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11+Q14
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
H ~~ 0*G
P ~~ 0*G
S ~~ 0*G
"
```

It reads that there are a **general** factor (`G`) which includes all items (from `Q1` to `Q14`, except `Q5`); the specific factors are **health** (`H`) with `Q7`, `Q8`, `Q9`, **personal** (`P`) with `Q1`, `Q2`, `Q12`, `Q13`, and **social** (`S`) with `Q3`, `Q4`, `Q6`, `Q10`, `Q11`, `Q14`. In addition, `H ~~ 0*P`, `H ~~ 0*S`, and `P ~~ 0*S` are instructions to remove the correlations between the specific latent factors, while `H ~~ 0*G`, `P ~~ 0*G`, and `S ~~ 0*G` remove the correlations between specific and the general latent factor, which is recommended for the bifactor model. These modifications are implemented in `CFA_bifactor_previous.R` and `CFA_bifactor_configural.R`, implementing the preliminary and the configural analysis, respectively.

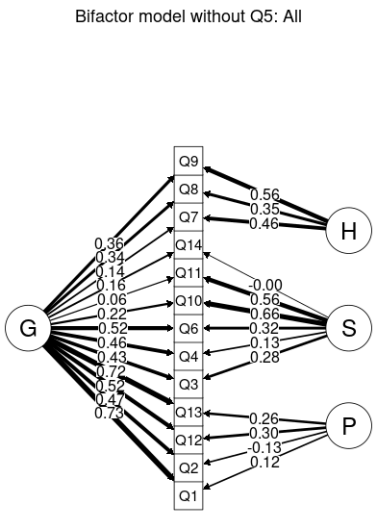
The preliminary analysis is:

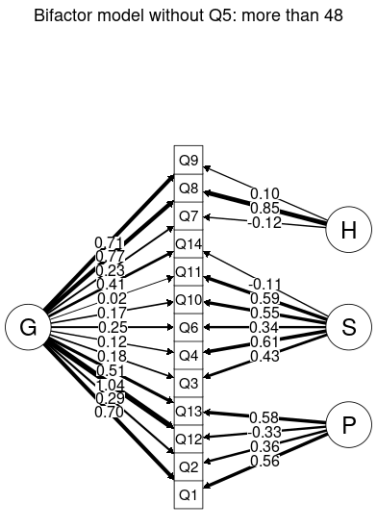
```
# CFA_bifactor_previous.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_bifactor_previous" # base filename without .ext

# Bifactor model
model <- "
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G
"
title <- paste0("Bifactor model without Q5")
group <- NULL
invariance <- "preliminary"
bifactor <- "G"
what <- "std" #"est"

# CFA
source("CFA.R")
```

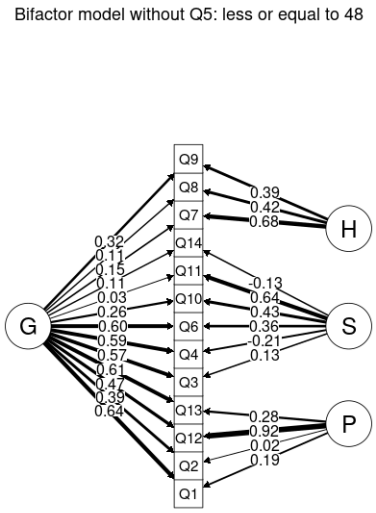
with the output:

```
*** Confirmatory Factor Analysis:  Bifactor model without Q5  ***
      (preliminary analysis)

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G


-----------
All
-----------

EXACT TEST (robust):

* X^2(52): 88.34941
* p: 0.00123712

TEST WITH TOLERANCE (robust):

* RMSEA: 0.052 [0,0.083]
* pclose: 0.438316

Reliability analysis (McDonald's omega):
  group                 G                 P                 S                 H
1   All 0.564202789385776 0.036970596330678 0.394030067352062 0.420812880457222


Model selection:

* CFI = 0.9270614
* TLI = 0.8905921
* PNFI = 0.4738591
* SABIC = 4165.464

-----------
less or equal to 48
-----------

EXACT TEST (robust):

* X^2(52): 102.5878
* p: 3.638281e-05

TEST WITH TOLERANCE (robust):

* RMSEA: 0.068 [0.007,0.106]
* pclose: 0.2360225

Reliability analysis (McDonald's omega):
                group                 G                 P                 S                 H
1 less or equal to 48 0.624956568146555 0.166954804711499 0.20101835589194 0.417270276707764


Model selection:

* CFI = 0.8770533
* TLI = 0.8155799
```

```
* PNFI = 0.3675582
* SABIC = 2385.747

-----------
more than 48
-----------

EXACT TEST (robust):

* X^2(52): 105.8168
* p: 1.531773e-05

TEST WITH TOLERANCE (robust):

* RMSEA: 0.081 [0.016,0.124]
* pclose: 0.15574

Reliability analysis (McDonald's omega):
         group               G                 P                 S                 H
1 more than 48 0.3433455622113 0.333632277415942 0.545929966237037 0.131431216991374


Model selection:

* CFI = 0.9054623
* TLI = 0.8581935
* PNFI = 0.4189128
* SABIC = 1475.011
```







The configural analysis is:

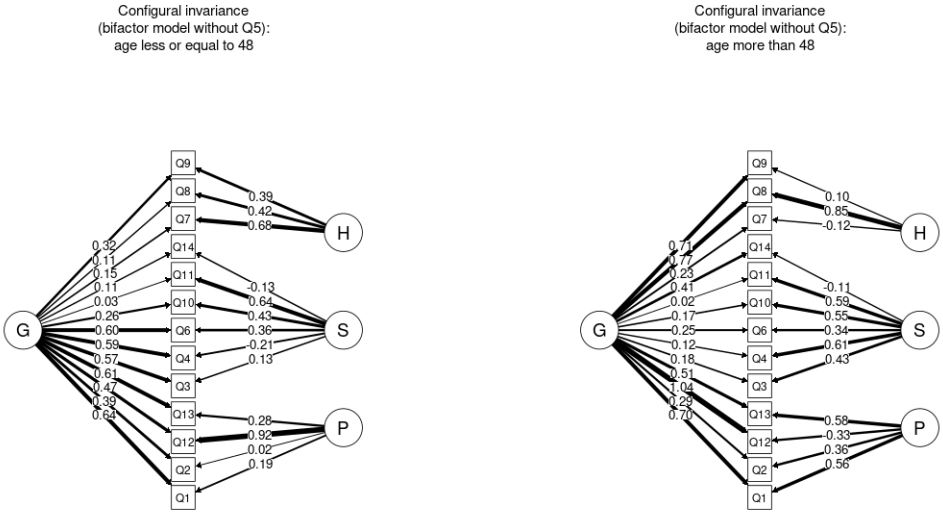
```
# CFA_bifactor_configural.R

# observed data
```

```
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_bifactor_configural" # base filename without .ext

# Bifactor model
model <- "
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G
"
title <- paste0("Configural invariance\n(bifactor model without Q5)")
group <- "Group"
invariance <- "configural"
bifactor <- "G"
what <- "std" #"est"

# CFA
source("CFA.R")
```

obtaining:

```
*** Confirmatory Factor Analysis:  Configural invariance
(bifactor model without Q5)  ***
      (group.equal = )

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(104): 208.2209
* p: 5.915631e-09

TEST WITH TOLERANCE (robust):

* RMSEA: 0.073 [0.038,0.102]
* pclose: 0.1214061

Reliability analysis (McDonald's omega):
                group     G     P     S     H
1 less or equal to 48 0.625 0.167 0.201 0.417
2        more than 48 0.343 0.334 0.546 0.131


Model selection:
```

```
* CFI = 0.892773
* TLI = 0.8391595
* PNFI = 0.3956444
* SABIC = 3933.301
```



This configural model was deemed acceptable by $RMSEA = 0.076$ $[0.044, 0.104], pclose = 0.084$ or robust $RMSEA = 0.073$ $[0.038, 0.102], pclose = 0.121$), and McDonald's omegas inside the interval $[0, 1]$.

Bifactor model, metric invariance

The modification resides in set to `group.equal="loadings"`, meaning that the model has factor loadings constrained to be equal between groups (female pacients below and above 48 years-old in this example):

```
model.estim <- lavaan::cfa(model,
                           estimator="MLR",
                           optim.method="BFGS",
                           missing="fiml",
                           std.lv=TRUE,
                           data=data,
                           group="Group",
                           group.equal="loadings")
```

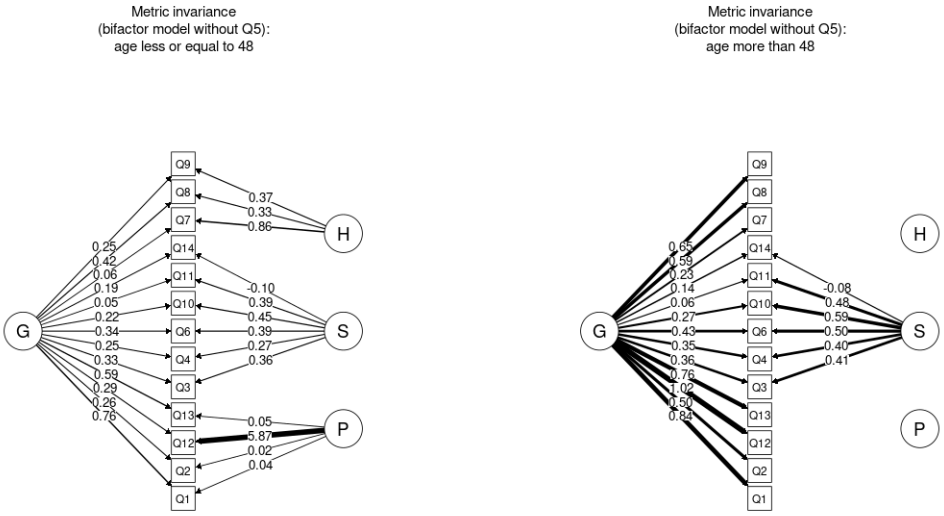
The implementation in `bifactor_metric.R`:

```
# CFA_bifactor_metric.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_bifactor_metric" # base filename without .ext

# Bifactor model
model <- "
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G
"
title <- paste0("Metric invariance\n(bifactor model without Q5)")
group <- "Group"
invariance <- "metric"
bifactor <- "G"
what <- "std" #"est"

# CFA
source("CFA.R")
```

creates the output:

```
*** Confirmatory Factor Analysis:  Metric invariance
(bifactor model without Q5)  ***
      (group.equal = loadings)

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(126): 243.0445
* p: 1.893278e-09

TEST WITH TOLERANCE (robust):

* RMSEA: 0.09 [0.062,0.116]
* pclose: 0.01293431

Reliability analysis (McDonald's omega):
               group     G      P     S     H
1 less or equal to 48 0.341  1.659 0.343 0.487
2        more than 48 0.608 -0.742 0.587 0.000
```

```
Model selection:

* CFI = 0.801137
* TLI = 0.7537887
* PNFI = 0.4244233
* SABIC = 3962.822
```

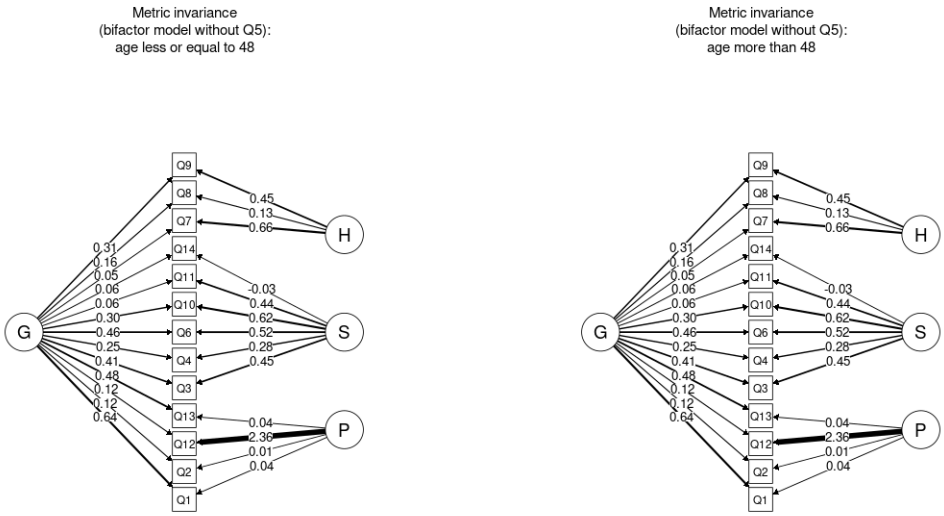


This metric model was rejected by
$RMSEA = 0.1$ $[0.076, 0.123], pclose = 0.0008$ or
robust $RMSEA = 0.09$ $[0.062, 0.116], pclose = 0.013$),
and all McDonald's omegas inside the interval $[0, 1]$, except a negative value for `P` in the group with age `less or equal 48 years-old`. In addition, the factor loadings for the `P` factor for the group "less than 48" could not be computed (one may check it is NA).

The analysis should stop here, stating that these groups were similar, under this model, up to the configural level.

An interesting point to note is that the metric invariance requires the equality of factor loadings. However, this is not reflected in the figure above. This discrepancy occurs because `what="std"` in `bifactor_metric.R` instructs the program to display correlations instead of average estimates. When the procedure is repeated with `what="est"`, the figure accurately reflects the

expected equality of loading factors, becoming:



Bifactor model, scalar invariance

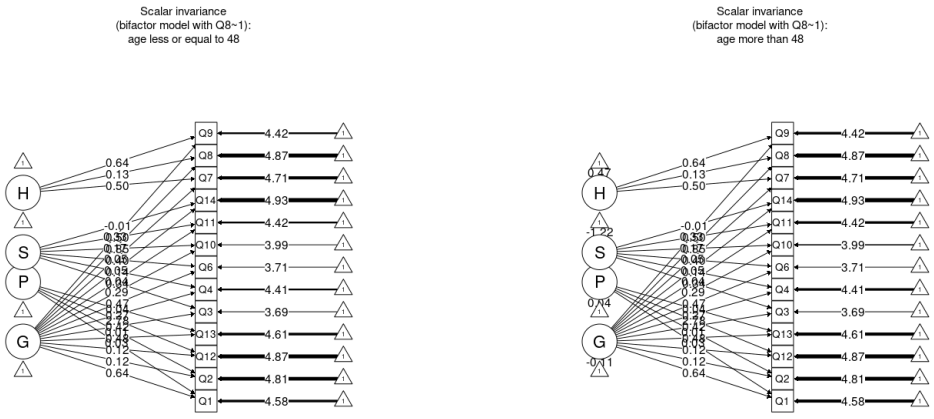
For the sake of completeness, the following scripts show how to configure the analysis for the scalar and strict levels of invariance, changing the `equal.group` parameter only by adding, progressively, more constraints. The scalar invariance is implemented by:

```
# CFA_bifactor_scalar.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_bifactor_scalar" # base filename without .ext

# Bifactor model
model <- "'
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G
```

```
"
title <- paste0("Scalar invariance\n(bifactor model with Q8~1)")
group <- "Group"
invariance <- "scalar"
bifactor <- "G"
what <- "std" # "std" or "est"

# CFA
source("CFA.R")
```

Since our focus here is on equality of item means (=intercepts), we changed the parameter `what="est"`. Thus, keep in mind that the values on the arrows connecting latent factors to observed items are no more correlations. It is obtained:

```
*** Confirmatory Factor Analysis:  Scalar invariance
(bifactor model with Q8~1)  ***
      (group.equal = loadings, intercepts)
'
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(135): 258.746
* p: 8.129625e-10

TEST WITH TOLERANCE (robust):

* RMSEA: 0.092 [0.065,0.117]
* pclose: 0.007474179

Reliability analysis (McDonald's omega):
               group     G      P     S     H
1 less or equal to 48 0.351  2.264 0.329 0.509
2        more than 48 0.589 -0.911 0.459 0.004


Model selection:

* CFI = 0.7799323
* TLI = 0.7456996
* PNFI = 0.4282102
* SABIC = 3962.272
```



In this new figure, the parameter `intercepts=TRUE` is set to show the intercepts. The model presents a simple regression for each item. The value on the arrow from a factor (independent variable) to an item (dependent variable) represents the slope coefficient, while the value on the arrow from the triangle (which has a constant equal to 1) to the same item represents the intercept. Notice that the intercepts have been equalized between the two groups of women.

– partial scalar invariance

Complete scalar invariance implies that all item intercepts (= means) are equivalent across groups. However, in many practical cases, this may not be realistic or achievable. When complete scalar invariance is not obtained, partial scalar invariance can be a practical approach to achieve comparability between groups in situations where complete scalar invariance is unfeasible.

Partial scalar invariance allows some intercepts to vary across groups while most remain invariant. This is done to allow the model to maintain an acceptable degree of comparability between groups without requiring all intercepts to be equal.

To establish partial scalar invariance, the following steps are generally followed:

1. **Test of Complete Scalar Invariance:** First, complete scalar invariance is tested by constraining all intercepts to be equal across groups.

2. **Identification of Non-Invariant Items:** If the test of complete scalar invariance fails, specific non-invariant intercepts are identified. This can be done by examining the model's modification indices or using other diagnostic techniques.

3. **Relaxation of Constraints:** Invariance constraints are then relaxed for the intercepts that are non-invariant, allowing these intercepts to vary across groups.

4. **Test of the New Model:** The adjusted model is tested again to check if it provides a good fit to the data while maintaining most intercepts invariant.

This process is repeated, one item at a time, trying to find an acceptable model. One possible criteria to find a candidate item to intercept relaxation is implemented in `CFA_scoretest.R`:

```
# CFA_scoretest.R

if(!exists("old_items")){old_items <- c()}
ts <- lavaan::lavTestScore(model.estim)$uni
ts <- ts[order(ts$X2, decreasing = TRUE),]
labels <- lavsummary$pe$label
rhss <- lavsummary$pe$rhs
rhss <- rhss[which(labels!="")]
labels <- labels[which(labels!="")]
labels <- labels[which(rhss!="")]
rhss <- rhss[which(rhss!="")]
p_item <- ""
p <- 0
ok <- FALSE
if(length(old_items)<length(unique(rhss)))
{
  while(!ok)
  {
    p <- p+1
    position <- which(labels==ts$lhs[p])
    if(length(position)>0)
    {
      p_item <- unique(rhss[position])
      tsok <- ts[which(ts$lhs==unique(labels[position])),]
      ok <- TRUE
      for (i in old_items)
      {
        if(p_item==i) # already used
        {
          ok <- FALSE
        }
      }
    } else
    {
      ok <- FALSE
    }
  }
  old_items <- c(old_items,p_item)
  cat("\nItem with highest discrepancy is ",p_item,sep="")
  cat("\nX^2(",tsok$df,") = ",tsok$X2," p = ",tsok$p.value,"\n",sep="")
} else
{
  cat("\nNo more items available\n")
}
```

For instance, the bifactor model computed above at the scalar level was computed by `CFA_bifactor_scalar.R` and rejected with the model:

```
model <- "
G =~ Q1+Q2+Q3+Q4+Q6+Q7+Q8+Q9+Q10+Q11+Q12+Q13+Q14
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11+Q14
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
H ~~ 0*G
P ~~ 0*G
S ~~ 0*G
"
```

Applying:

```
source("CFA_scoretest.R")
```

it is obtained:

```
Item with highest discrepancy is Q8
X^2(1) = 13.96756 p = 0.0001859924
```

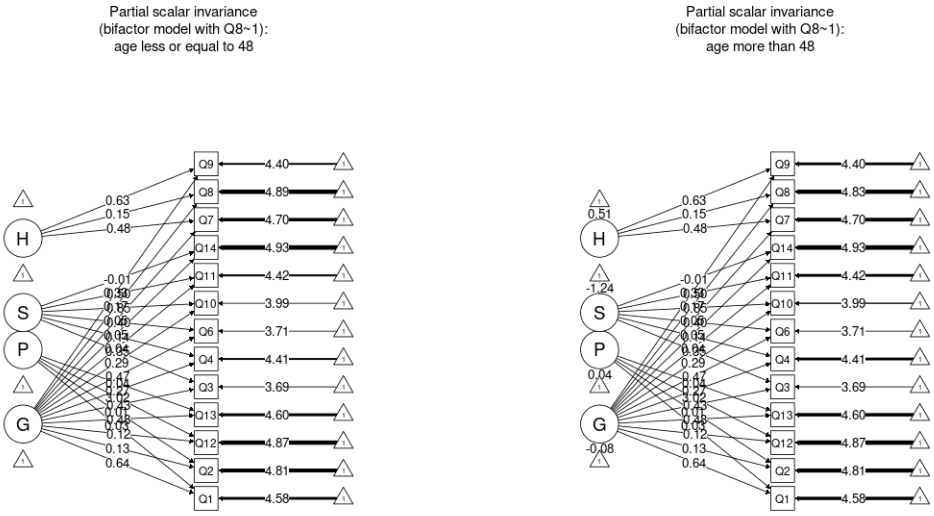
In order to process a partial scalar model relaxing the intercept of `Q8`, `lavaan::cfa` requires the addition of the parameter `group.partial="Q8~1"`. A variant script, `CFApartial.R`, replaces `CFA.R` to include this parameter, running:

```
  model.estim <- lavaan::cfa(model,
                             estimator="MLR",
                             optim.method="BFGS",
                             missing="fiml",
                             std.lv=TRUE,
                             data=data,
                             group="Group",
                             group.equal=c("loadings","intercepts"),
                             group.partial="Q8~1")
```

and obtaining:

```
*** Confirmatory Factor Analysis:  Partial scalar invariance
(bifactor model with Q8~1)  ***
      (group.equal = loadings, intercepts)

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(134): 256.8023
```

```
* p: 9.381434e-10

TEST WITH TOLERANCE (robust):

* RMSEA: 0.092 [0.065,0.117]
* pclose: 0.007556288

Reliability analysis (McDonald's omega):
                group     G      P     S     H
1 less or equal to 48 0.354  2.638 0.329 0.500
2        more than 48 0.588 -1.059 0.461 0.006


Model selection:

* CFI = 0.780685
* TLI = 0.744678
* PNFI = 0.4282981
* SABIC = 3963.237
```

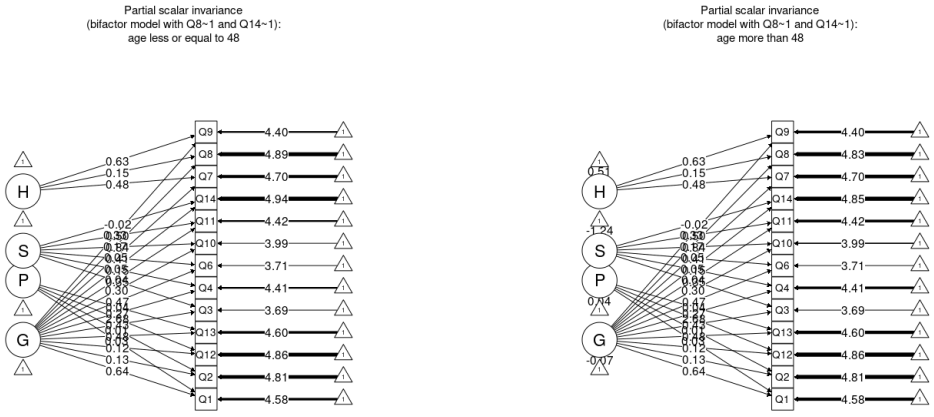


Observe that the intercept value of `Q8` differs between the groups. Since the model is still rejected, call `CFA_scoretest.R` to locate the next candidate item, which is:

```
Item with highest discrepancy is Q14
X^2(1) = 8.767032 p = 0.003067242
```

The parameter is, then, altered to `group.partial=c("Q8~1","Q14~1")`. Running `CFA_bifactor_scalar.R` again:

```
*** Confirmatory Factor Analysis:  Partial scalar invariance
(bifactor model with Q8~1 and Q14~1)  ***
      (group.equal = loadings, intercepts)

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(133): 255.8955
* p: 8.376427e-10

TEST WITH TOLERANCE (robust):

* RMSEA: 0.093 [0.066,0.117]
* pclose: 0.007315694

Reliability analysis (McDonald's omega):
                group     G      P     S     H
1 less or equal to 48 0.355  2.098 0.324 0.498
2       more than 48 0.590 -0.844 0.456 0.006


Model selection:

* CFI = 0.7805582
* TLI = 0.7426096
* PNFI = 0.4266112
* SABIC = 3964.266
```



which now released the intercepts of `Q8` and `Q14`. The model is still rejected,

thus `CFA_scoretest.R` now provides:

```
Item with highest discrepancy is Q4
X^2(1) = 7.323649 p = 0.006805309
```

One may proceed by setting `group.partial=c("Q81","Q141","Q4~1")` and so on, iterating until an acceptable model is achieved or no more eligible items are found. However, it is important to note that relaxing too many items can be counterproductive, as it may effectively revert to the metric level, albeit with a different model structure. If a reasonable model of partial scalar invariance is found, it then serves as the reference for the strict invariance model.

Bifactor model, strict invariance

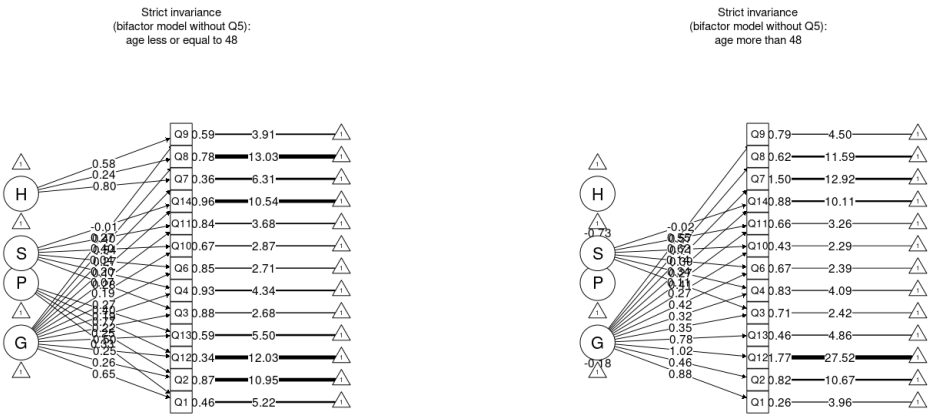
The strict invariance is implemented by:

```
# CFA_bifactor_strict.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_bifactor_strict" # base filename without .ext

# Bifactor model
model <- "
G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G
"
title <- paste0("Strict invariance\n(bifactor model without Q5)")
group <- "Group"
invariance <- "strict"
bifactor <- "G"
what <- "std" # "std" or "est"

# CFA
source("CFA.R")
```

obtaining:

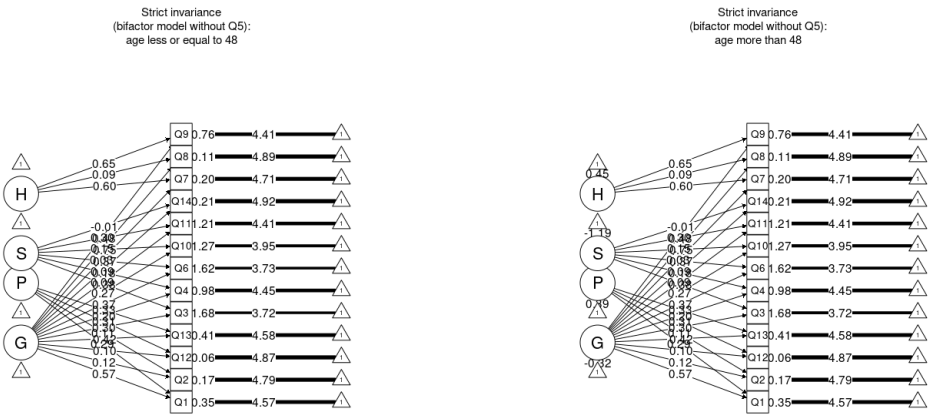
```
*** Confirmatory Factor Analysis:  Strict invariance
(bifactor model without Q5)  ***
      (group.equal = loadings, intercepts, residuals)

G =~ Q1 + Q2 + Q7 + Q8 + Q9 + Q12 + Q13 +  Q3 + Q4 + Q6 + Q10 + Q11 + Q14
P =~ Q1 + Q2 + Q12 + Q13
S =~ Q3 + Q4 + Q6 + Q10 + Q11 + Q14
H =~ Q7 + Q8 + Q9
```

```
H ~~ 0*P
H ~~ 0*S
P ~~ 0*S
P ~~ 0*G
S ~~ 0*G
H ~~ 0*G

EXACT TEST (robust):

* X^2(148): 340.897
* p: 2.956264e-17

TEST WITH TOLERANCE (robust):

* RMSEA: 0.112 [0.087,0.135]
* pclose: 0.0001380836

Reliability analysis (McDonald's omega):
               group     G      P     S      H
1 less or equal to 48 0.265  0.302 0.305  0.570
2        more than 48 0.625 -0.123 0.552 -0.259


Model selection:

* CFI = 0.6452173
* TLI = 0.6260399
* PNFI = 0.3172778
* SABIC = 4023.754
```



The residual appears as an additional number next to each item. Changing the parameter to `what="est"`, it is possible to verify that both, intercepts and

residuals, are equal between groups:



Sintaxe for first and second order models

It is important, to complete this guide, to show how is the structure of a first and second order models. The first order model has two differences in implementation when compared with a bifactor model: (1) obviously, there is no general factor and (2) correlations between factors is allowed. For instance:

```
model <- "
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11
"
```

To a second order model it is only necessary to add the general factor now pointing to the specific factors, instead of pointing to items. For instance:

```
model <- "
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11
G =~ H+P+S
"
```

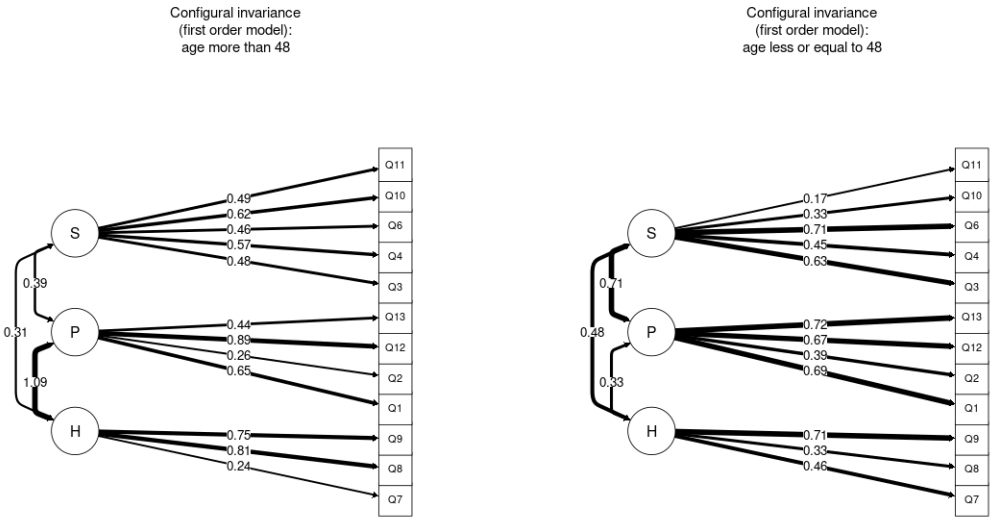
In this example, these two models were acceptable at the preliminary analysis, but they were rejected at the configural invariance level.

- First order model implemented in `CFA_1st_configural.R`:

```
# CFA_1st_configural.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_1st_configural" # base filename without .ext

# First order model
model <- "
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11
"
title <- paste0("Configural invariance\n(first order model)")
group <- "Group"
group.equal <- ""
bifactor <- ""

# CFA
source("CFA.R")
```

with output:

```
*** Confirmatory Factor Analysis:  Configural invariance
(first order model)  ***
      (group.equal = )

H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11

EXACT TEST:

* X^2(102) = 186.7919
* p = 6.229063e-07

TEST WITH TOLERANCE:

* RMSEA: 0.11 [0.085,0.135]
* RMSEA.pclose (p value of RMSEA) = 0.0002013301
* robust RMSEA: 0.104 [0.076,0.131]
* robust RMSEA.pclose (robust p value of RMSEA) = 0.001668227

Reliability analysis (McDonald's omega):
                group     H     P     S
1       more than 48 0.729 0.419 0.652
2 less or equal to 48 0.574 0.736 0.585


Model selection:

* CFI = 0.7739767
* PNFI = 0.4881169
* SABIC = 3826.97
```



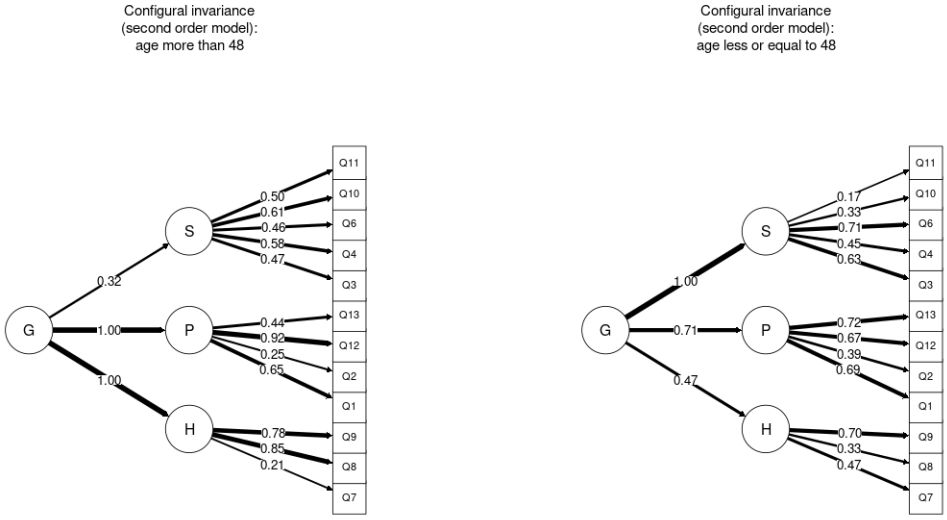
Observe that there are correlations between all pair of factors.

- Second order model implemented in `CFA_2nd_configural.R`:

```
# CFA_2nd_configural.R

# observed data
data <- readxl::read_excel("data.xlsx")
data$Group <- factor(data$Group,
                     levels = c("less or equal to 48","more than 48"))
responses <- paste0("Q",1:14)
data[,responses] <- lapply(data[,responses],as.numeric)

filename <- NA # screen output, otherwise
# filename <- "CFA_2nd_configural" # base filename without .ext

# Second order model
model <- "
H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11
G =~ H+P+S
"
title <- paste0("Configural invariance\n(second order model)")
group <- "Group"
group.equal <- ""
bifactor <- ""

# CFA
source("CFA.R")
```

with output:

```
*** Confirmatory Factor Analysis:  Configural invariance
(second order model)  ***
      (group.equal = )

H =~ Q7+Q8+Q9
P =~ Q1+Q2+Q12+Q13
S =~ Q3+Q4+Q6+Q10+Q11
G =~ H+P+S

EXACT TEST:

* X^2(102) = 190.3675
* p = 2.637446e-07

TEST WITH TOLERANCE:

* RMSEA: 0.112 [0.087,0.137]
* RMSEA.pclose (p value of RMSEA) = 0.0001080517
* robust RMSEA: NA [NA,NA]
* robust RMSEA.pclose (robust p value of RMSEA) = NA

Reliability analysis (McDonald's omega):
                group     H     P     S
1        more than 48 0.775 0.420 0.650
2 less or equal to 48 0.572 0.736 0.586


Model selection:

* CFI = 0.7644454
* PNFI = 0.4826688
* SABIC = 3830.546
```



Observe that the general factor has specific factors as intermediates to the items.

Final words

This appendix provides an in-depth look at the methodological framework used for questionnaire validation, focusing on the use of Confirmatory Factor Analysis (CFA). The detailed step-by-step process outlined includes the application of various invariance tests – configural, metric, scalar, and strict – in the pursuit of establishing the psychometric robustness of the measurement models. By illustrating the hierarchical nature of these tests and the significance of non-rejection in CFA, with the inclusion of practical examples, descriptive statistics, and R code scripts, we intend to offer guidance for researchers conducting similar validation studies.

References

[1] Brown TA. Confirmatory factor analysis for applied research. Guilford publications; 2015.

[2] Cheung GW, Rensvold RB. Evaluating goodness-of-fit indexes for testing measurement invariance. Structural Equation Modeling. Structural Equation Modeling: A Multidisciplinary Journal. 2002;9(2):233-55.

[3] Dimitrov DM. Testing for factorial invariance in the context of construct Validation. Measurement and Evaluation in Counseling and Development. 2010;43(2):121-49.

[4] MacCallum RC, Browne MW, Sugawara HM. Power analysis and determination of sample size for covariance structure modeling. Psychological Methods. 1996;1(2):130-49.

[5] Davis SR, Castelo-Branco C, Chedraui P, Lumsden MA, Nappi RE, Shah D, et al. Understanding weight gain at menopause. Climacteric. 2012;15(5):419-29.

[6] Pompei LM, Bonassi-Machado R, Steiner ML, Pompei IM, de Melo NR, Nappi RE, et al. Profile of Brazilian climacteric women: results from the Brazilian Menopause Study. Climacteric. 2022;25(5):523-9. Available from: https://doi.org/10.1080/13697137.2022.2088276.

[7] Rosseel Y. lavaan: An R Package for Structural Equation Modeling. Journal of Statistical Software. 2012;48(2):1-36.

[8] Curran PJ, Bollen KA, Chen F, Paxton P, Kirby JB. Finite Sampling Properties of the Point Estimates and Confidence Intervals of the RMSEA. Sociological Methods and Research. 2003;32(2):208-52.