Appendix A

This report narrates the process of factor analizing the proposed scale, eliminating items, and fitting the resulting scale to a confirmatory model.

library(magrittr) # enables piping : %>%   
library(psych)  
library(ggplot2)# graphing  
library(sem)  
library(GPArotation)  
requireNamespace("readr") # data input  
requireNamespace("tidyr") # data manipulation  
requireNamespace("dplyr") # Avoid attaching dplyr, b/c its function names conflict with a lot of packages (esp base, stats, and plyr).  
requireNamespace("testit") # For asserting conditions meet expected patterns.  
requireNamespace("readxl") # for inputing Excel files

source("./shinyApp/sourced/SteigerRLibraryFunctions.txt")  
source("./shinyApp/sourced/AdvancedFactorFunctions\_CF.R")  
source("./scripts/common-functions.R") # used in multiple reports  
source("./scripts/graph-presets.R") # fonts, colors, themes   
source("./scripts/fa-utility-functions.R") # to graph factor patterns

sample\_size <- 643  
opt <- options(fit.indices = c("GFI", "AGFI", "RMSEA", "NFI", "NNFI", "CFI", "RNI", "IFI", "SRMR", "AIC", "AICc", "BIC", "CAIC"))

dto <- readRDS("./data/unshared/derived/dto.rds")  
unitData <- dto$unitData  
metaData <- dto$metaData   
ds <- dto$analytic

# Introduction

The purpose of this research was to develop a new measure of fear of childbirth (the Childbirth Fear Questionnaire; CFQ) that would address the limitations of existing measures. Participants were 643 pregnant women residing in English speaking countries, and were recruited via online forums. Participants completed a set of questionnaires, including the CFQ, via an online survey.

The administered CFQ contained 49 items grouped into 9 subgroups. The review of items on the questionnaire, their descriptive statistics, and group correlations are available in the [Appendix C](https://rawgit.com/andkov/fear-of-childbirth/master/reports/appendix-c/appendix-c.html).

# Overview

This section give a summary of the steps applied during the analysis.

Out analysis consists of a series of phases. We start with the original Fear of Childbirth Questionnaire (FCQ) items, conduct exploratory factor analysis and use the results to diagnose those items exhibiting poor performance. At the end of each step we eliminate **one** item and repeat the steps of the analysis. This process is repeated until we no longer have the basis for item elimination.

By comparing various rotations and considering interpretive qualities of the solutions, we have decided to use orthogonal bifactor rotation, as the one that offers the greatest interpretability. To examine the various solutions in details consult [Appendix B](https://rawgit.com/andkov/fear-of-childbirth/master/reports/appendix-b/appendix-b-0.html)

## Analysis Steps

The following steps are repeated for each of the analytic phases:

#### 1.Scree

Scree plot is plotted and top eigen values are displayed

#### 2.MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

#### 3.Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

#### 4.Fit

psych::fa call is applied to conduct maximum likelihood factor analysis (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed, following the formulae:

CFI = ((chisq\_null-df\_null) - (chisq-df))/(chisq\_null-df\_null)  
 TLI = ((chisq\_null/df\_null) - (chisq/df))/((chisq\_null/df\_null)-1)

For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

#### 5.RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

#### 6.Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

#### 7.Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

## Elimination

The decisions to eliminate an item from the scale is guided by the following considerations:  
1. Does the item have a non-trivial (>.30) loading on a subfactor?  
2. Does the item have a non-trivial (>.50) loading on the general factor?  
3. Does item load on more than two subfactors?  
4. Does item have substantive relevance and aid interpretability of the solution?  
5. Does item have a substantial contribution to the R-square of the model relative other items?

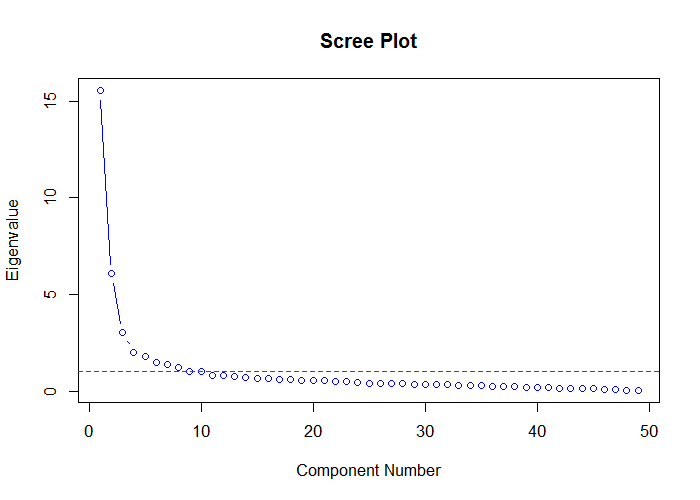
# Phase 0

We create a correlation matrix using all 49 items on the administered scale and conduct eigen diagnostics:

# Phase 0  
items\_phase\_0 <- c(paste0("foc\_0",1:9), paste0("foc\_",10:49))  
R0 <- make\_cor(ds, metaData, items\_phase\_0)

## Scree

# Diagnosing number of factors  
Scree.Plot(R0)



#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R0)),  
 value = eigen(R0)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

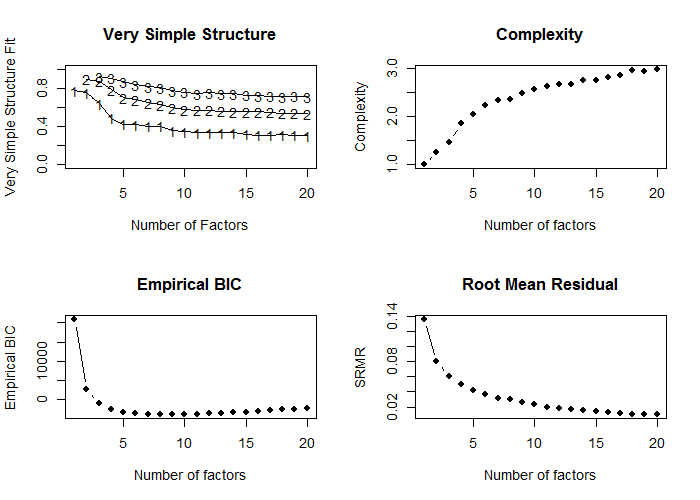
eigen value  
1 1 15.5464443  
2 2 6.0701008  
3 3 3.0215047  
4 4 2.0079960  
5 5 1.7679392  
6 6 1.4593287  
7 7 1.3778156  
8 8 1.2151014  
9 9 1.0094502  
10 10 0.9982523  
11 11 0.8091173  
12 12 0.7928341  
13 13 0.7500816  
14 14 0.7169123  
15 15 0.6768888

Scree plot is somewhat ambiguious, suggesting a solution involving 4-5 factors, while Keiser rule (eigenvalue > 1) suggests up to 9 factors.

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R0,n.obs = 643)

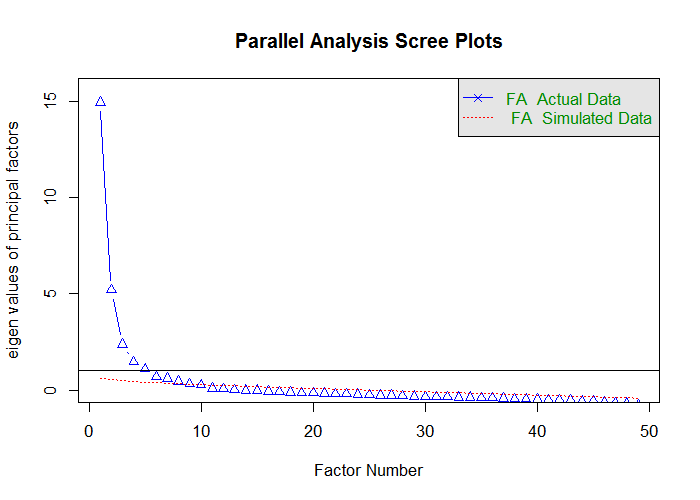


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.78 with 1 factors  
VSS complexity 2 achieves a maximimum of 0.89 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 9 factors   
Empirical BIC achieves a minimum of -3999.01 with 7 factors  
Sample Size adjusted BIC achieves a minimum of -791.46 with 16 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.78 0.00 0.037 1127 14390 0.0e+00 69.3 0.78 0.137 7103 10681 1.0 28254 0.1367 0.140 20966  
2 0.75 0.89 0.023 1079 10796 0.0e+00 32.5 0.89 0.120 3819 7245 1.3 9667 0.0799 0.083 2690  
3 0.64 0.87 0.018 1032 8000 0.0e+00 23.9 0.92 0.104 1327 4604 1.5 5587 0.0608 0.065 -1086  
4 0.49 0.78 0.015 986 5813 0.0e+00 19.7 0.94 0.089 -562 2568 1.9 3720 0.0496 0.054 -2656  
5 0.42 0.70 0.014 941 4970 0.0e+00 16.6 0.95 0.083 -1114 1873 2.0 2671 0.0420 0.047 -3413  
6 0.42 0.68 0.013 897 4362 0.0e+00 14.5 0.95 0.079 -1438 1410 2.2 2061 0.0369 0.042 -3739  
7 0.40 0.65 0.013 854 3754 0.0e+00 12.7 0.96 0.074 -1768 943 2.3 1523 0.0317 0.037 -3999  
8 0.40 0.63 0.013 812 2935 2.5e-237 12.3 0.96 0.065 -2315 263 2.4 1405 0.0305 0.037 -3846  
9 0.36 0.60 0.012 771 2571 4.6e-192 10.7 0.97 0.062 -2414 34 2.5 999 0.0257 0.032 -3986  
10 0.34 0.58 0.012 731 2228 6.5e-151 9.9 0.97 0.058 -2499 -178 2.6 810 0.0231 0.029 -3916  
11 0.34 0.57 0.013 692 1898 6.9e-113 8.9 0.97 0.054 -2577 -380 2.6 585 0.0197 0.026 -3890  
12 0.33 0.57 0.014 654 1681 1.8e-91 8.4 0.97 0.051 -2548 -472 2.7 483 0.0179 0.024 -3746  
13 0.34 0.56 0.014 617 1325 9.8e-54 8.0 0.97 0.044 -2665 -706 2.7 407 0.0164 0.023 -3583  
14 0.33 0.56 0.015 581 1181 3.8e-43 7.5 0.98 0.042 -2576 -731 2.7 334 0.0149 0.021 -3422  
15 0.32 0.55 0.016 546 1015 1.4e-30 7.0 0.98 0.038 -2516 -782 2.8 276 0.0135 0.020 -3254  
16 0.31 0.55 0.017 512 894 3.7e-23 6.7 0.98 0.036 -2417 -791 2.8 234 0.0124 0.019 -3077  
17 0.31 0.55 0.018 479 795 4.8e-18 6.5 0.98 0.034 -2303 -782 2.9 213 0.0119 0.019 -2885  
18 0.31 0.54 0.019 447 702 1.3e-13 6.2 0.98 0.032 -2188 -769 3.0 175 0.0108 0.017 -2715  
19 0.31 0.54 0.021 416 608 2.2e-09 5.9 0.98 0.029 -2082 -761 3.0 159 0.0102 0.017 -2531  
20 0.30 0.53 0.022 386 550 7.4e-08 5.7 0.98 0.028 -1946 -720 3.0 139 0.0096 0.017 -2357

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R0,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 9 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 14.94487089 0.6225188  
2 5.21708368 0.5279537  
3 2.37587802 0.4887215  
4 1.45560672 0.4438806  
5 1.09227743 0.4160784  
6 0.69844153 0.3880725  
7 0.61012843 0.3659361  
8 0.45511336 0.3384841  
9 0.32340554 0.3079898  
10 0.27259629 0.2824877  
11 0.08722195 0.2620589  
12 0.06374368 0.2393029  
13 0.01338476 0.2156926  
14 -0.01480093 0.1936663  
15 -0.01810249 0.1741381

## Fit

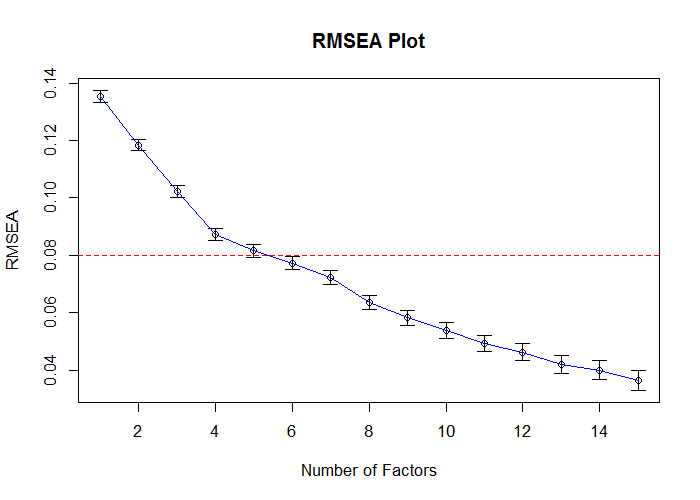
psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

ls\_solution <- solve\_factors(R0,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)

## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R0,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 15.54644 14390.417 1127 0 0.13539363 0.13342391 0.13737263  
 [2,] 2 21.61655 10789.396 1079 0 0.11839687 0.11637184 0.12043263  
 [3,] 3 24.63805 7965.640 1032 0 0.10229945 0.10021106 0.10440020  
 [4,] 4 26.64605 5796.907 986 0 0.08717812 0.08501450 0.08935580  
 [5,] 5 28.41399 4959.072 941 0 0.08155423 0.07932515 0.08379840  
 [6,] 6 29.87331 4332.693 897 0 0.07724020 0.07494366 0.07955277  
 [7,] 7 31.25113 3712.555 854 0 0.07220659 0.06983360 0.07459650  
 [8,] 8 32.46623 2923.307 812 0 0.06364000 0.06116116 0.06613606  
 [9,] 9 33.47568 2448.905 771 0 0.05822225 0.05563812 0.06082320  
[10,] 10 34.47393 2092.926 731 0 0.05387044 0.05117417 0.05658229  
[11,] 11 35.28305 1766.844 692 0 0.04918721 0.04635593 0.05203071  
[12,] 12 36.07588 1549.766 654 0 0.04618925 0.04322740 0.04915953  
[13,] 13 36.82597 1318.401 617 0 0.04207975 0.03894254 0.04521586  
[14,] 14 37.54288 1176.173 581 0 0.03994533 0.03665421 0.04322725  
[15,] 15 38.21977 1014.674 546 0 0.03656552 0.03305376 0.04004819

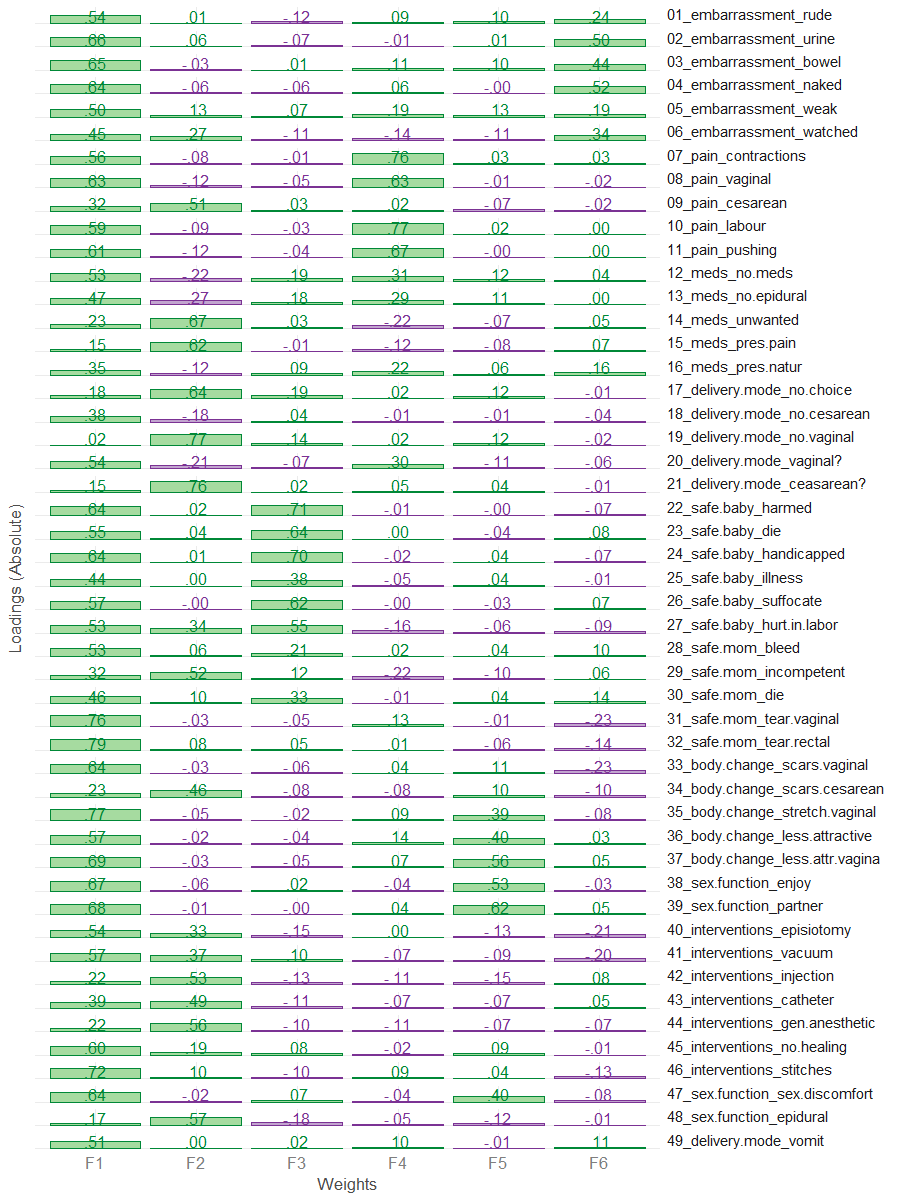
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

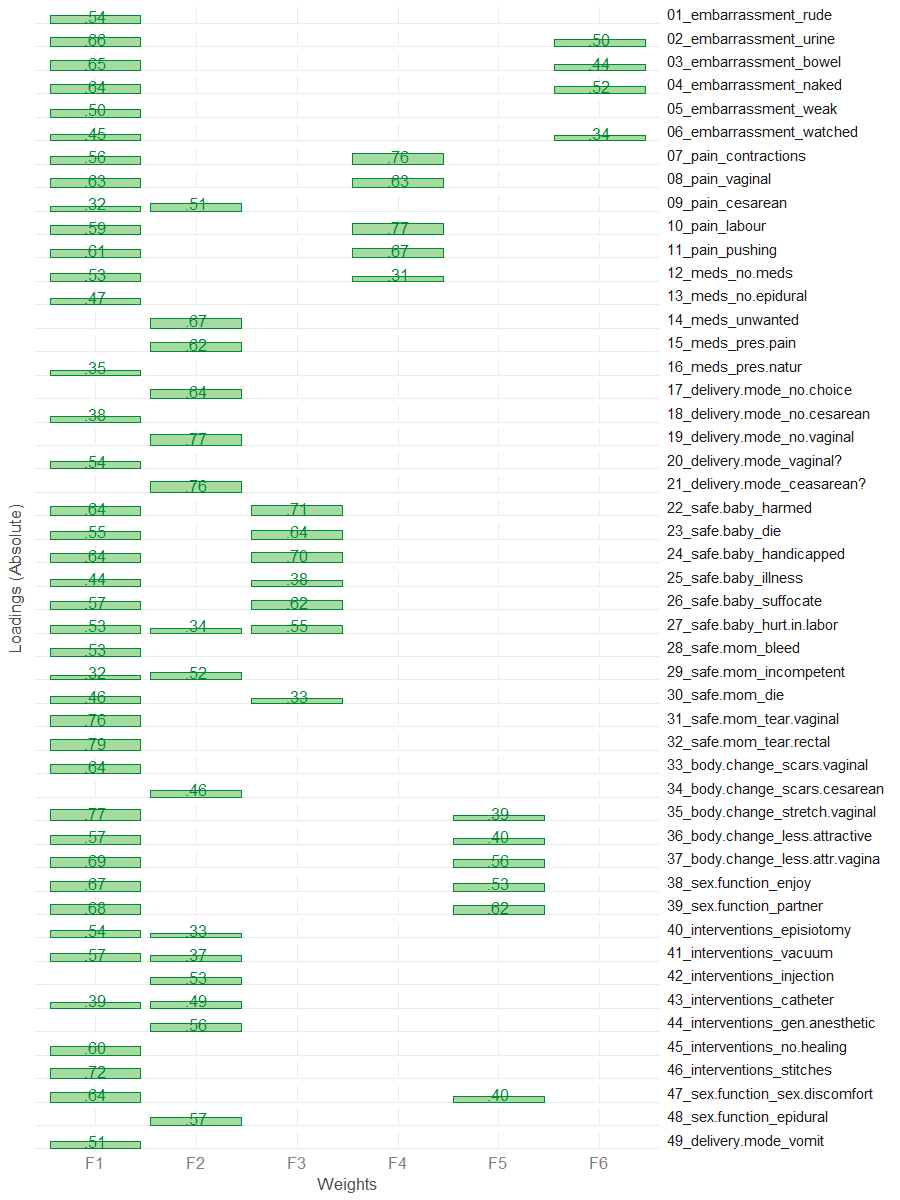
fit\_efa\_0 <- MLFA(  
 Correlation.Matrix = R0,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_0[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)

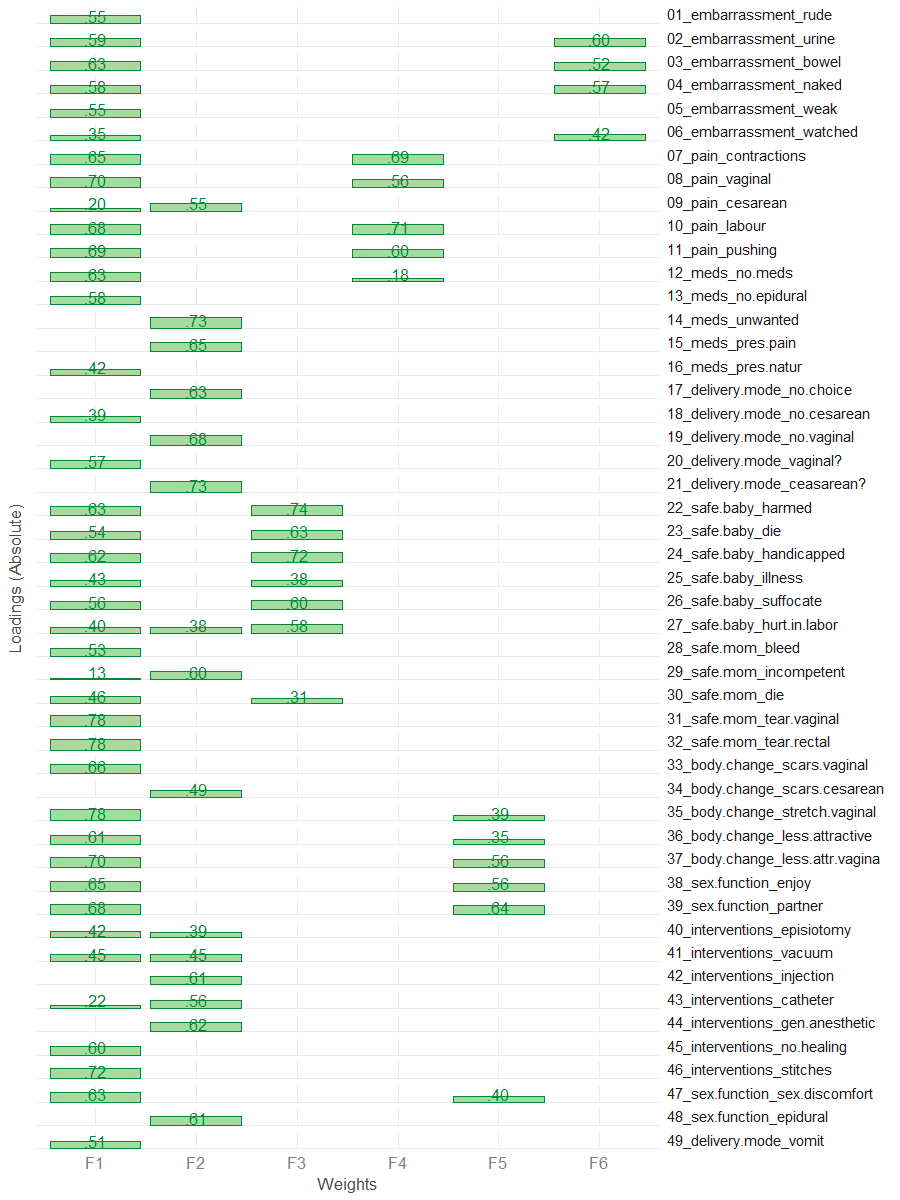


## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

# These values are translated into CFA model and used as starting values  
model\_0 <- FAtoSEM(  
 x = fit\_efa\_0[["Bifactor"]] ,  
 cutoff = 0.30,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

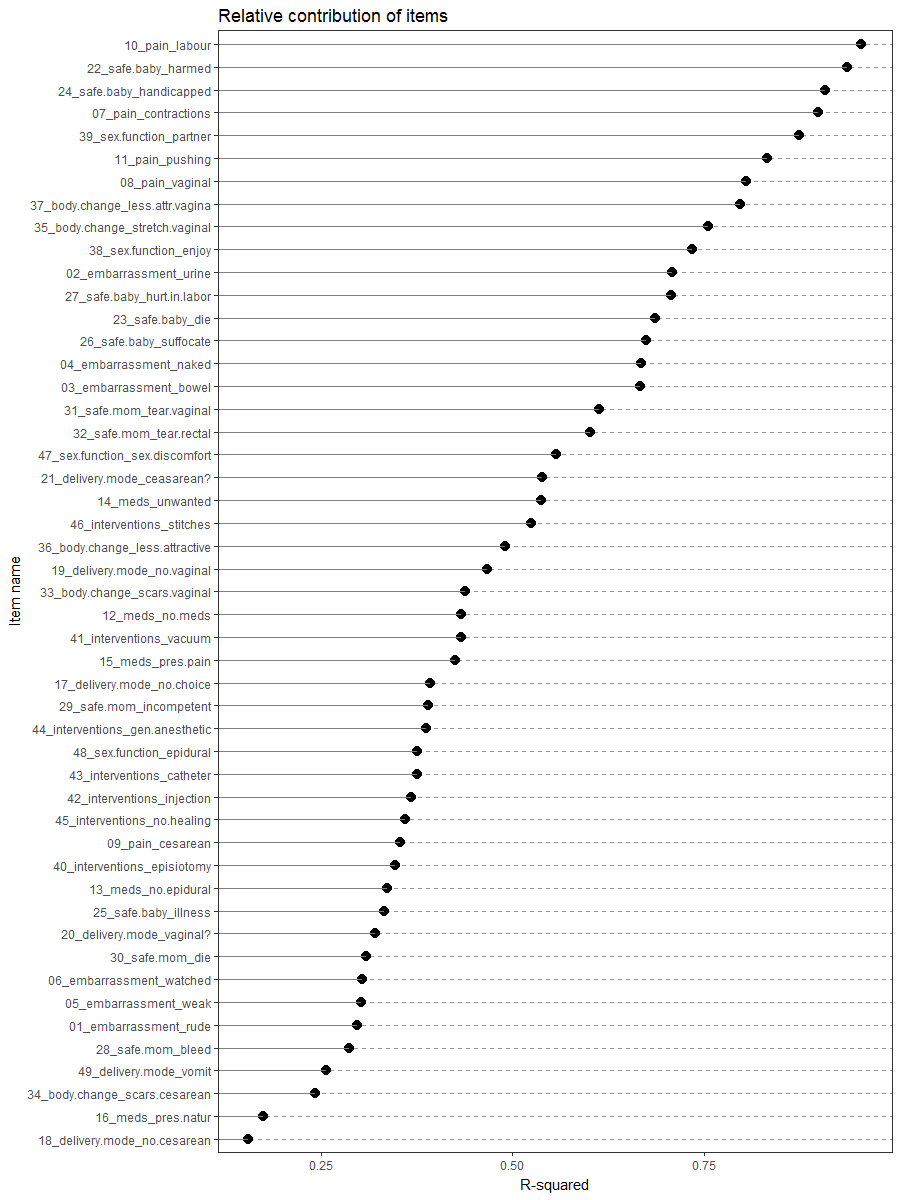
# the model is estimated using sem package  
fit\_0 <- sem::sem(model\_0,R0,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_0)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_0)

Model Chiquare = 6071.013 | df model = 1099 | df null = 1176  
Goodness-of-fit index = 0.7018118  
Adjusted Goodness-of-fit index = 0.6676246  
RMSEA index = .0839 90% CI: (.082,.086)  
Comparitive Fit Index (CFI = 0.7923754  
Tucker Lewis Index (TLI/NNFI) = 0.7778285  
Akaike Information Criterion (AIC) = 6323.013  
Bayesian Information Criterion (BIC) = -1035.281

#Relative contribudion of items   
sort(summary(fit\_0)$Rsq) %>% dot\_plot()



# Phase 1

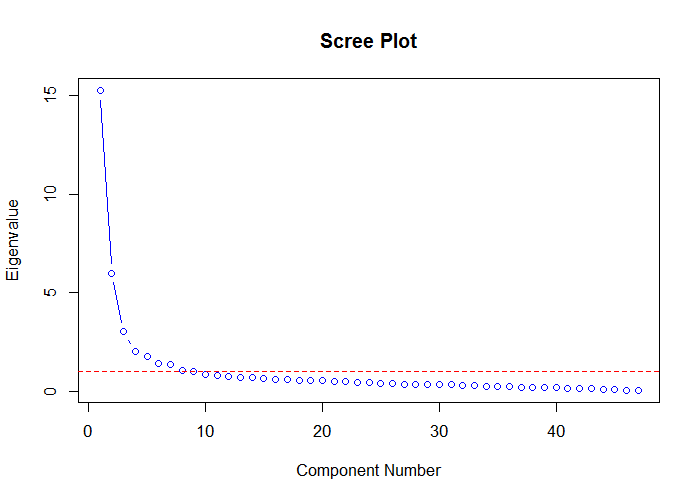
Based on the results from phase 0 we identified that items 18 and item 16 are performing poorly in the scale, based on the consideration we have [outlined](#elimination). They both do not have a non-trivial loading on any subscales (1), have small loadings on the general factor (2) and contribute marketly less to the R-square compared to other items (5). We also did not percieve them to be crucial for interpretive validity of the solution (4).

Thus we remove item 16 and item 18 from the pool of items and repeat the analytical steps.

drop\_items\_1 <- c("foc\_18","foc\_16")  
items\_phase\_1 <- setdiff(items\_phase\_0, drop\_items\_1)  
R1 <- make\_cor(ds, metaData, items\_phase\_1)

## Scree

# Diagnosing number of factors  
Scree.Plot(R1)



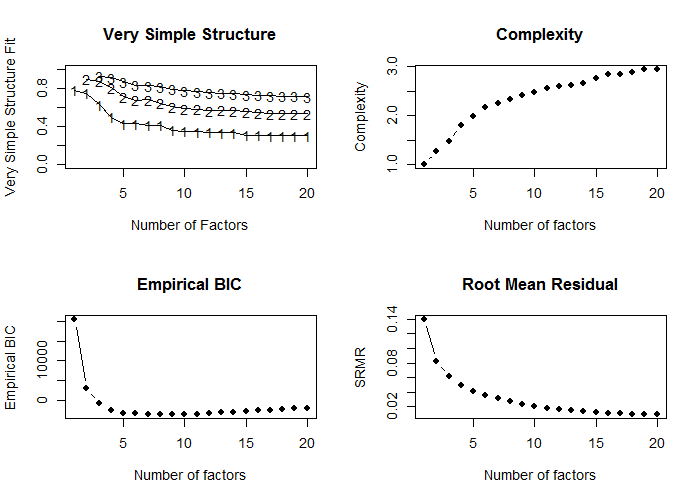
#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R1)),  
 value = eigen(R1)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

eigen value  
1 1 15.2610009  
2 2 5.9755365  
3 3 3.0168417  
4 4 1.9983595  
5 5 1.7537678  
6 6 1.3995805  
7 7 1.3757185  
8 8 1.0354289  
9 9 0.9894350  
10 10 0.8274985  
11 11 0.8074161  
12 12 0.7185295  
13 13 0.6993490  
14 14 0.6689659  
15 15 0.6543354

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R1,n.obs = 643)

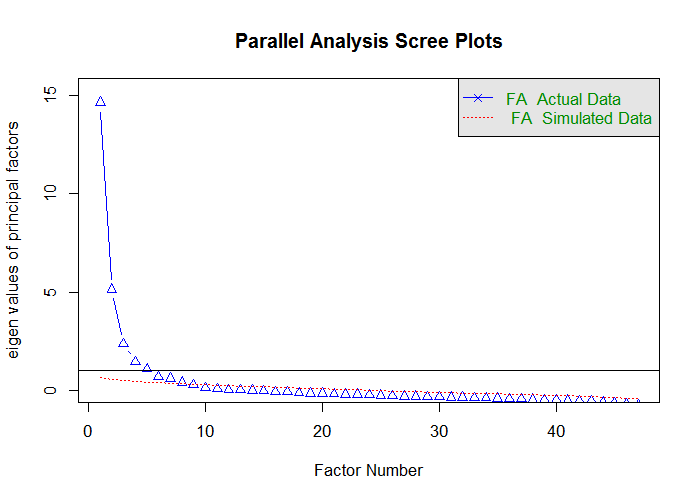


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.78 with 1 factors  
VSS complexity 2 achieves a maximimum of 0.9 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 8 factors   
Empirical BIC achieves a minimum of -3717.33 with 9 factors  
Sample Size adjusted BIC achieves a minimum of -737.44 with 15 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.78 0.00 0.039 1034 13986 0.0e+00 66.3 0.78 0.142 7300 10583 1.0 27314 0.1402 0.143 20628  
2 0.75 0.90 0.025 988 10420 0.0e+00 30.8 0.90 0.124 4032 7169 1.3 9377 0.0821 0.086 2988  
3 0.63 0.87 0.020 943 7620 0.0e+00 22.0 0.93 0.107 1522 4516 1.5 5243 0.0614 0.066 -854  
4 0.50 0.80 0.015 899 5435 0.0e+00 17.7 0.94 0.090 -378 2476 1.8 3328 0.0489 0.054 -2485  
5 0.43 0.71 0.014 856 4618 0.0e+00 14.6 0.95 0.084 -917 1801 2.0 2291 0.0406 0.046 -3244  
6 0.43 0.67 0.013 814 4024 0.0e+00 13.1 0.96 0.080 -1240 1345 2.2 1819 0.0362 0.042 -3444  
7 0.41 0.68 0.013 773 3430 0.0e+00 11.7 0.96 0.075 -1568 886 2.2 1438 0.0322 0.038 -3560  
8 0.41 0.64 0.012 733 2748 4.2e-230 10.3 0.97 0.067 -1991 336 2.3 1062 0.0276 0.034 -3678  
9 0.36 0.61 0.012 694 2267 6.7e-166 9.2 0.97 0.061 -2221 -17 2.4 770 0.0235 0.029 -3717  
10 0.35 0.58 0.013 656 1919 4.7e-124 8.5 0.97 0.056 -2323 -240 2.5 590 0.0206 0.026 -3652  
11 0.35 0.58 0.013 619 1623 4.3e-91 7.9 0.97 0.052 -2379 -414 2.5 458 0.0182 0.024 -3544  
12 0.34 0.57 0.014 583 1288 3.1e-55 7.5 0.97 0.045 -2481 -630 2.6 389 0.0167 0.023 -3381  
13 0.33 0.57 0.015 548 1138 1.5e-43 7.1 0.98 0.043 -2405 -665 2.6 331 0.0154 0.022 -3212  
14 0.33 0.57 0.016 514 988 2.7e-32 6.7 0.98 0.040 -2336 -704 2.7 275 0.0141 0.020 -3048  
15 0.31 0.56 0.017 481 846 1.9e-22 6.2 0.98 0.036 -2265 -737 2.8 205 0.0122 0.018 -2905  
16 0.31 0.54 0.018 449 748 2.7e-17 6.0 0.98 0.034 -2155 -730 2.8 182 0.0114 0.018 -2721  
17 0.31 0.53 0.019 418 639 1.6e-11 5.8 0.98 0.031 -2064 -736 2.8 163 0.0108 0.017 -2540  
18 0.30 0.53 0.021 388 562 1.6e-08 5.6 0.98 0.028 -1947 -715 2.9 135 0.0099 0.016 -2374  
19 0.31 0.53 0.022 359 495 2.4e-06 5.3 0.98 0.026 -1826 -686 2.9 113 0.0090 0.016 -2208  
20 0.30 0.53 0.025 331 496 1.0e-08 5.4 0.98 0.030 -1644 -593 2.9 113 0.0090 0.016 -2027

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R1,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 8 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 14.662802646 0.6512997  
2 5.128267536 0.5193880  
3 2.370174574 0.4840731  
4 1.445452900 0.4462623  
5 1.077566306 0.4076140  
6 0.657691710 0.3778618  
7 0.598542772 0.3475295  
8 0.387275698 0.3194363  
9 0.264490654 0.2948566  
10 0.117594262 0.2720865  
11 0.062287508 0.2493177  
12 0.015919955 0.2260899  
13 0.004584041 0.2004393  
14 -0.022108291 0.1815642  
15 -0.037872320 0.1596377

## Fit

psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

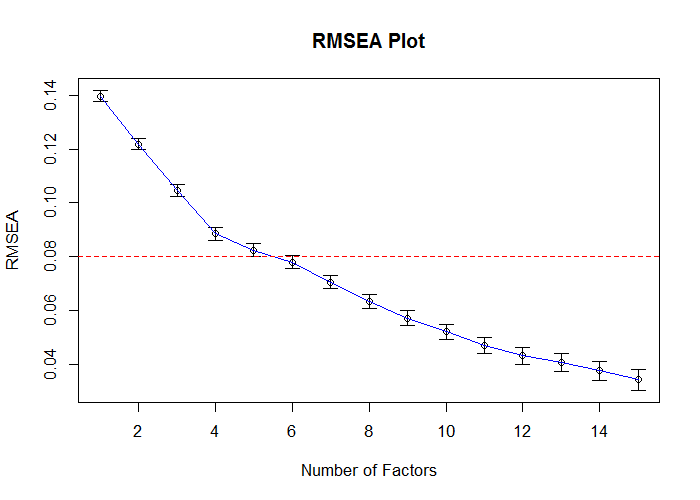
ls\_solution <- solve\_factors(R1,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)  
ds\_index

n\_factors chisq\_null df\_null chisq df CFI TLI  
1 1 23866.17 1081 27313.5521 1034 -0.1533621 -0.2057876  
2 2 23866.17 1081 9376.9482 988 0.6318242 0.5971680  
3 3 23866.17 1081 5243.4121 943 0.8112627 0.7836426  
4 4 23866.17 1081 3328.4142 899 0.8933774 0.8717920  
5 5 23866.17 1081 2290.9707 856 0.9370217 0.9204679  
6 6 23866.17 1081 1818.9932 814 0.9558927 0.9414250  
7 7 23866.17 1081 1438.2916 773 0.9708016 0.9591675  
8 8 23866.17 1081 1061.5430 733 0.9855808 0.9787352  
9 9 23866.17 1081 770.1728 694 0.9966569 0.9947927  
10 10 23866.17 1081 589.6575 656 1.0029117 1.0047980  
11 11 23866.17 1081 458.1732 619 1.0070584 1.0123265  
12 12 23866.17 1081 388.5802 583 1.0085327 1.0158214  
13 13 23866.17 1081 331.4643 548 1.0095034 1.0187466  
14 14 23866.17 1081 275.1349 514 1.0104834 1.0220477  
15 15 23866.17 1081 205.2746 481 1.0121011 1.0271960

## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R1,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 15.26100 13986.3420 1034 0 0.13968391 0.13763014 0.14174752  
 [2,] 2 21.23654 10402.0125 988 0 0.12182640 0.11971339 0.12395087  
 [3,] 3 24.25338 7582.5146 943 0 0.10472366 0.10254262 0.10691803  
 [4,] 4 26.25174 5415.9730 899 0 0.08846600 0.08620346 0.09074386  
 [5,] 5 28.00551 4590.7989 856 0 0.08243834 0.08010419 0.08478901  
 [6,] 6 29.40509 3985.3098 814 0 0.07790039 0.07549237 0.08032603  
 [7,] 7 30.78081 3246.6647 773 0 0.07060137 0.06810014 0.07312128  
 [8,] 8 31.81623 2623.4219 733 0 0.06338111 0.06077085 0.06601042  
 [9,] 9 32.80567 2149.5036 694 0 0.05715568 0.05442262 0.05990698  
[10,] 10 33.63317 1796.4776 656 0 0.05203840 0.04917031 0.05492240  
[11,] 11 34.44058 1497.2259 619 0 0.04701000 0.04398061 0.05004978  
[12,] 12 35.15911 1281.7351 583 0 0.04320704 0.04000694 0.04640942  
[13,] 13 35.85846 1127.4416 548 0 0.04058327 0.03721317 0.04394638  
[14,] 14 36.52743 980.3371 514 0 0.03759247 0.03401236 0.04114885  
[15,] 15 37.18176 843.5561 481 0 0.03426475 0.03041564 0.03805916

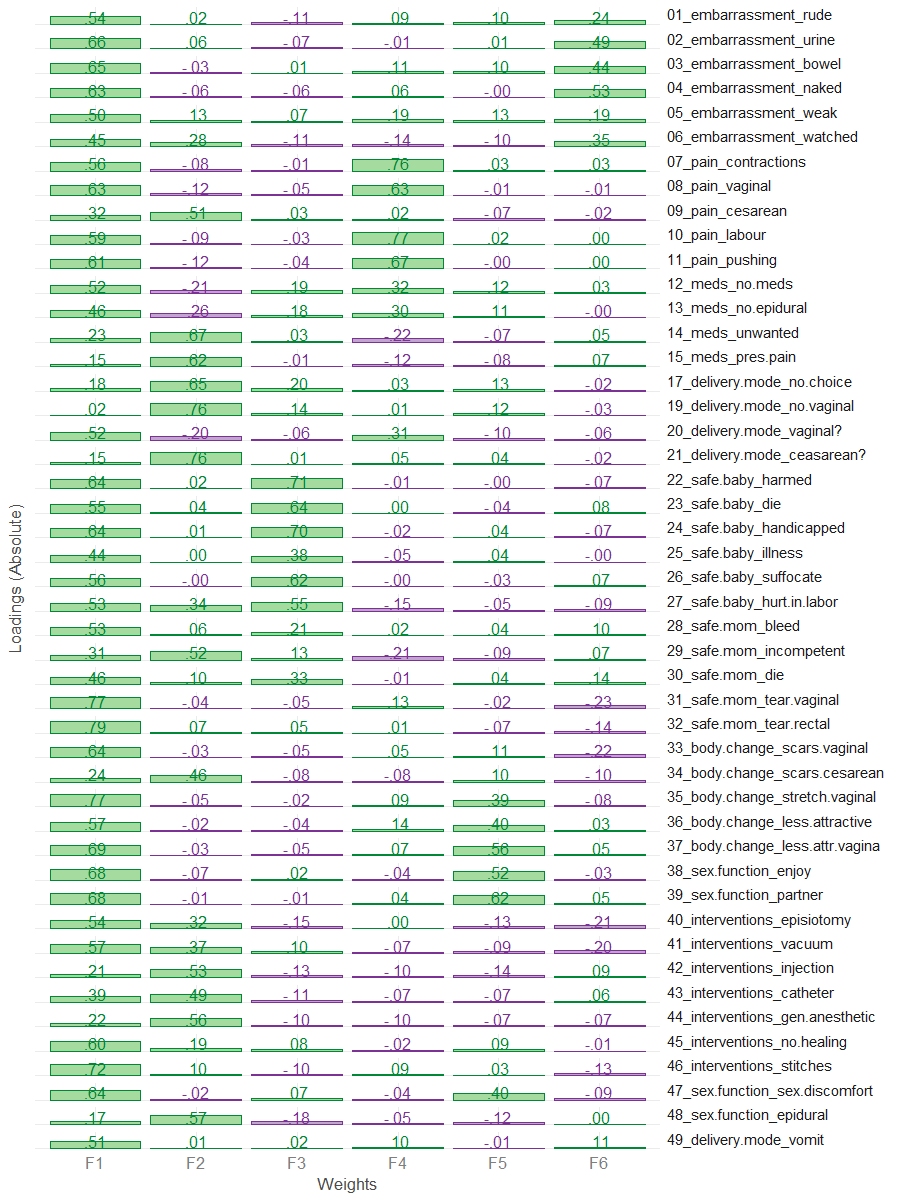
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

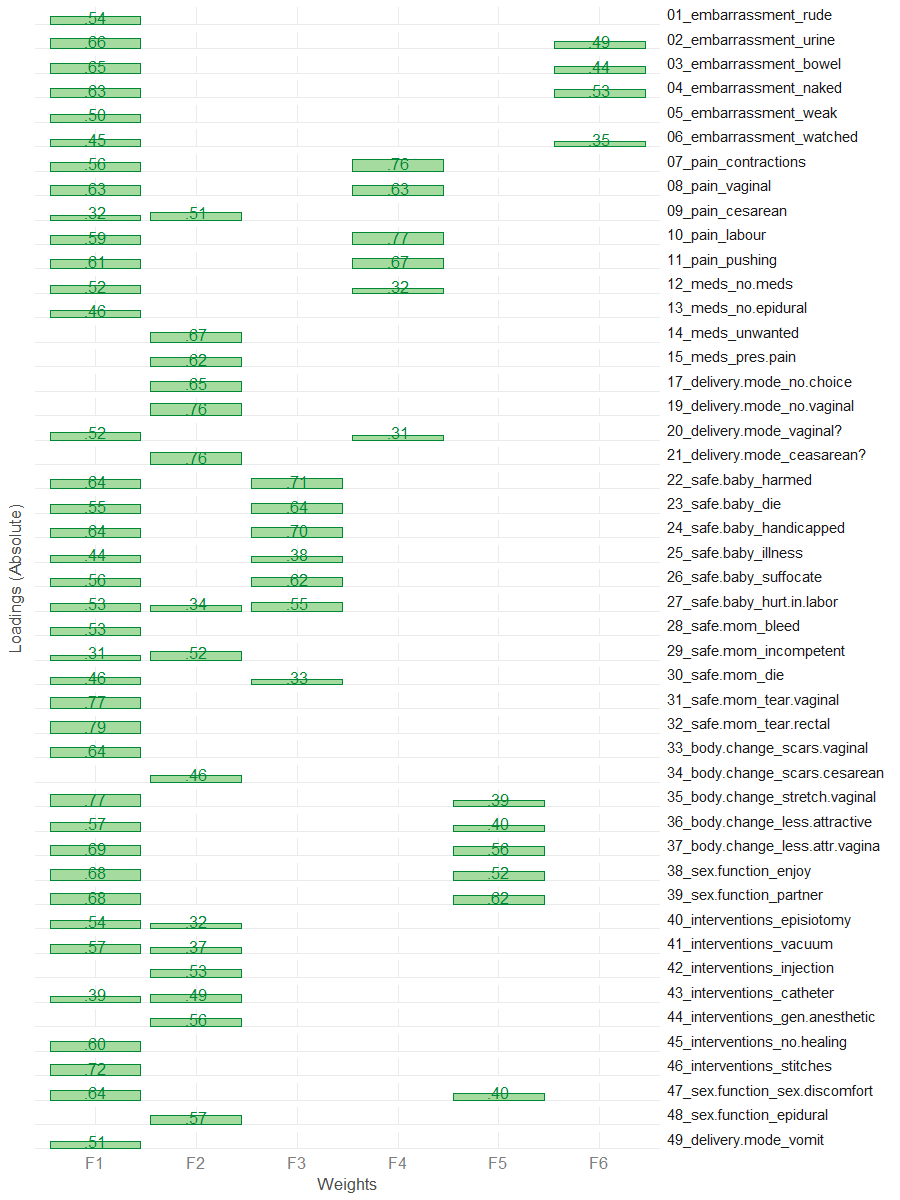
fit\_efa\_1 <- MLFA(  
 Correlation.Matrix = R1,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_1[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)

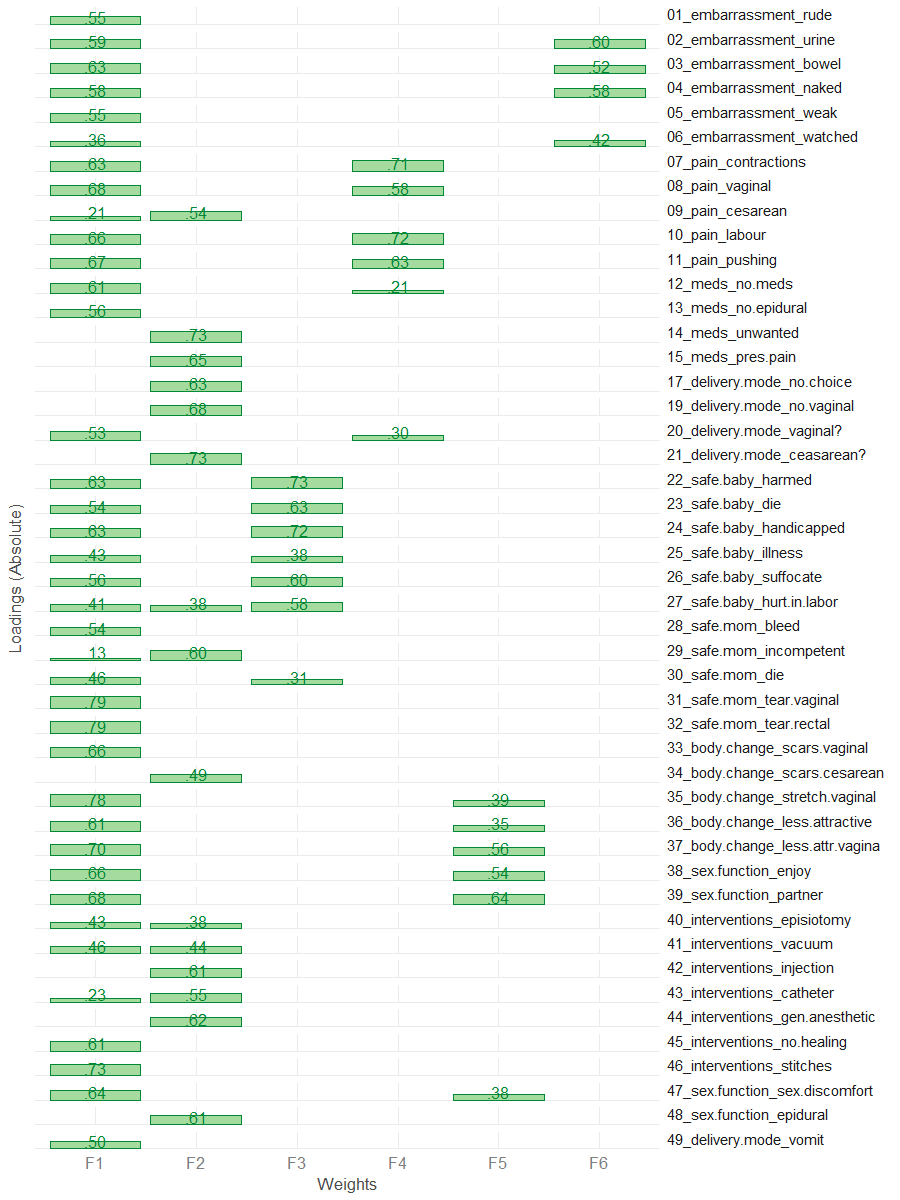


## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

# These values are translated into CFA model and used as starting values  
model\_1 <- FAtoSEM(  
 x = fit\_efa\_1[["Bifactor"]] ,  
 cutoff = 0.30,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

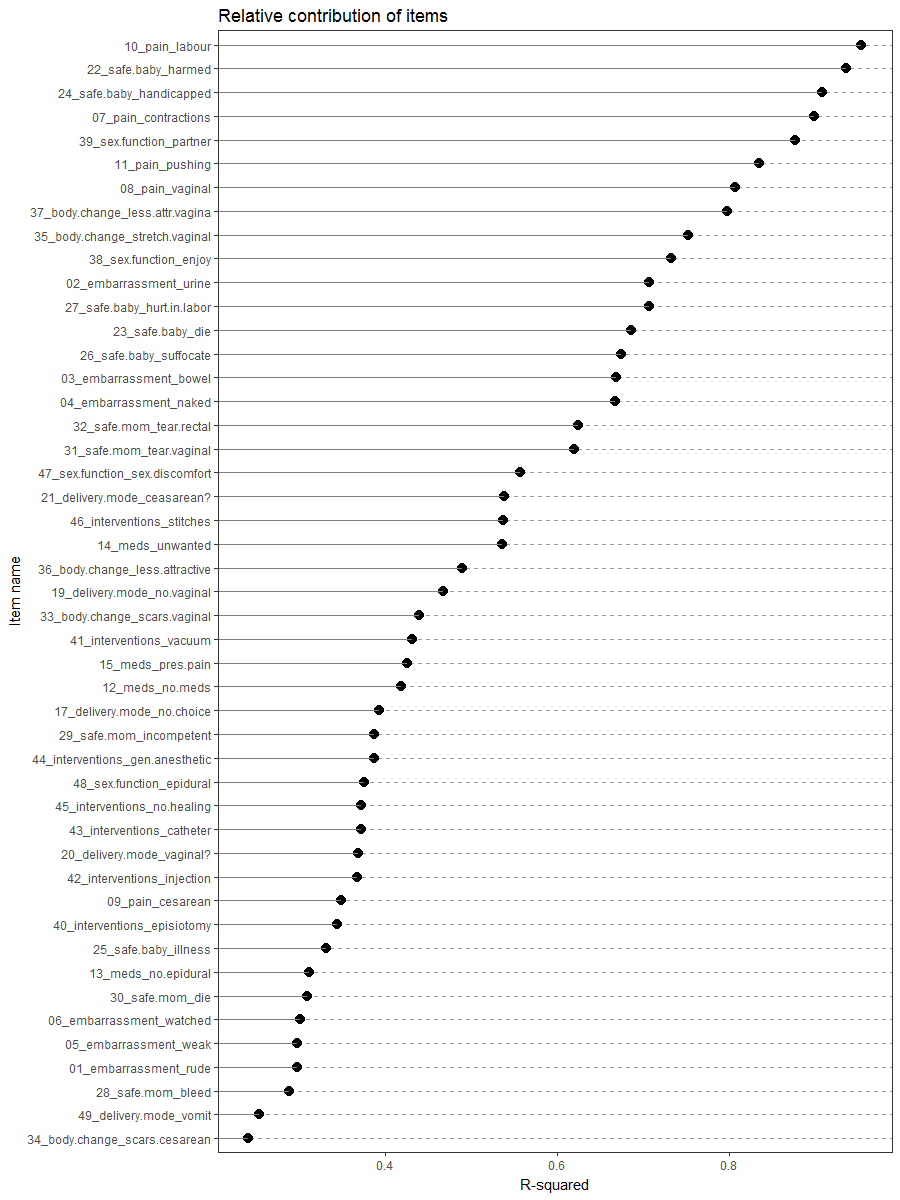
# the model is estimated using sem package  
fit\_1 <- sem::sem(model\_1,R1,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_1)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_1)

Model Chiquare = 5579.292 | df model = 1005 | df null = 1081  
Goodness-of-fit index = 0.7145923  
Adjusted Goodness-of-fit index = 0.6796618  
RMSEA index = .0842 90% CI: (.082,.086)  
Comparitive Fit Index (CFI = 0.8046405  
Tucker Lewis Index (TLI/NNFI) = 0.789867  
Akaike Information Criterion (AIC) = 5825.292  
Bayesian Information Criterion (BIC) = -919.1839

#Relative contribudion of items   
sort(summary(fit\_1)$Rsq) %>% dot\_plot()



# Phase 2

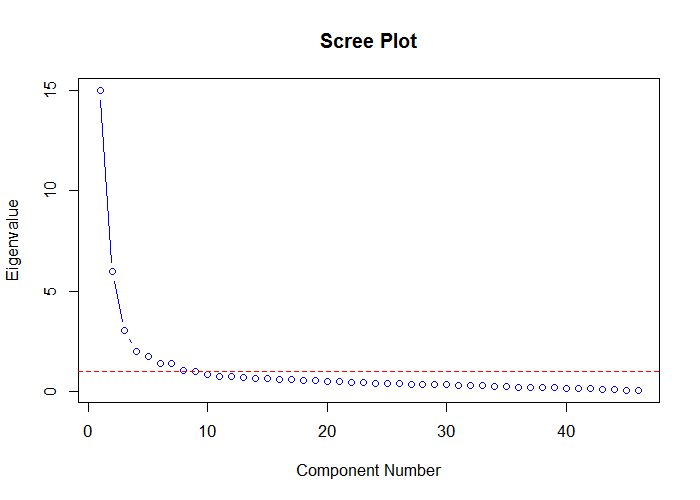
Based on the results from phase 1 we identified that item 49 performs poorly on the scale, according to the consideration we have [outlined](#elimination). It does not load on any of the subscales (1) and has a borderline non-trivial loading on the general factor (2). Also, we didn't think it was contributing coherence or interpretability to the scale (4). Although item 34 had a lower R-square contribution than item 49, we did not remove it from the scale because it has a stong subscale loading and contributed to interpretability.

Thus we removed item 49 from the pool of items and repeat the analytical steps.

drop\_items\_2 <- c("foc\_49")  
items\_phase\_2 <- setdiff(items\_phase\_1, drop\_items\_2)  
R2 <- make\_cor(ds, metaData, items\_phase\_2)

## Scree

# Diagnosing number of factors  
Scree.Plot(R2)



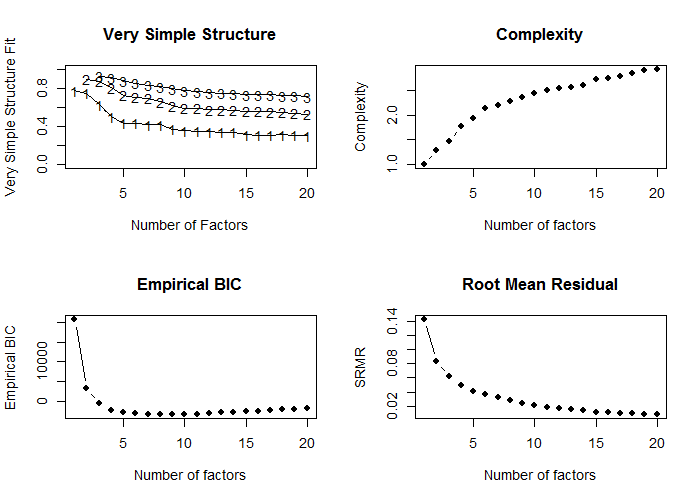
#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R2)),  
 value = eigen(R2)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

eigen value  
1 1 15.0033580  
2 2 5.9706177  
3 3 3.0128922  
4 4 1.9974413  
5 5 1.7318636  
6 6 1.3978818  
7 7 1.3694589  
8 8 1.0354151  
9 9 0.9731544  
10 10 0.8227477  
11 11 0.7544838  
12 12 0.7076260  
13 13 0.6771612  
14 14 0.6551042  
15 15 0.6087487

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R2,n.obs = 643)

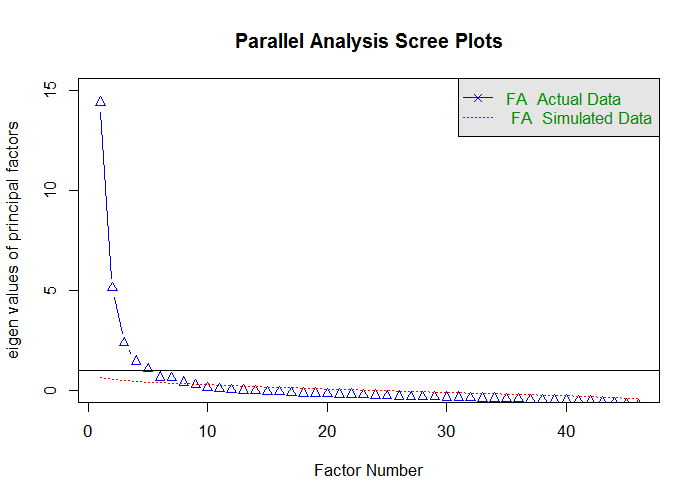


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.77 with 1 factors  
VSS complexity 2 achieves a maximimum of 0.9 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 8 factors   
Empirical BIC achieves a minimum of -3512.42 with 9 factors  
Sample Size adjusted BIC achieves a minimum of -709.97 with 16 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.77 0.00 0.041 989 13903 0.0e+00 65.6 0.77 0.145 7508 10648 1.0 27208 0.1430 0.146 20813  
2 0.75 0.90 0.026 944 10337 0.0e+00 30.1 0.90 0.126 4233 7230 1.3 9291 0.0836 0.087 3187  
3 0.63 0.87 0.020 900 7532 0.0e+00 21.4 0.93 0.109 1712 4570 1.5 5183 0.0624 0.067 -636  
4 0.50 0.81 0.016 857 5345 0.0e+00 17.0 0.94 0.092 -196 2525 1.8 3245 0.0494 0.054 -2296  
5 0.44 0.73 0.014 815 4543 0.0e+00 14.0 0.95 0.086 -727 1861 1.9 2247 0.0411 0.046 -3023  
6 0.43 0.70 0.014 774 3969 0.0e+00 12.4 0.96 0.082 -1035 1422 2.1 1750 0.0363 0.042 -3255  
7 0.42 0.70 0.014 734 3360 0.0e+00 11.2 0.96 0.076 -1386 945 2.2 1426 0.0327 0.039 -3320  
8 0.42 0.66 0.013 695 2680 3.0e-230 9.8 0.97 0.068 -1814 393 2.3 1020 0.0277 0.034 -3474  
9 0.37 0.62 0.013 657 2204 4.5e-166 8.7 0.97 0.062 -2044 42 2.4 736 0.0235 0.030 -3512  
10 0.36 0.59 0.013 620 1859 6.7e-124 8.0 0.97 0.057 -2150 -181 2.4 562 0.0206 0.027 -3447  
11 0.35 0.59 0.014 584 1564 1.6e-90 7.4 0.97 0.053 -2212 -358 2.5 432 0.0180 0.024 -3345  
12 0.35 0.57 0.014 549 1230 3.4e-54 7.0 0.98 0.045 -2320 -577 2.5 365 0.0165 0.023 -3185  
13 0.34 0.58 0.015 515 1082 1.8e-42 6.6 0.98 0.043 -2248 -613 2.5 307 0.0152 0.022 -3023  
14 0.34 0.57 0.016 482 925 3.2e-30 6.1 0.98 0.039 -2192 -662 2.6 246 0.0136 0.020 -2871  
15 0.31 0.56 0.017 450 784 1.7e-20 5.7 0.98 0.036 -2125 -697 2.7 180 0.0116 0.018 -2730  
16 0.31 0.56 0.018 419 669 8.7e-14 5.4 0.98 0.032 -2040 -710 2.7 155 0.0108 0.017 -2554  
17 0.31 0.56 0.020 389 582 8.0e-10 5.3 0.98 0.030 -1934 -699 2.8 136 0.0101 0.017 -2379  
18 0.31 0.55 0.021 360 500 1.4e-06 5.0 0.98 0.027 -1828 -685 2.8 110 0.0091 0.015 -2218  
19 0.31 0.54 0.024 332 440 6.5e-05 4.8 0.98 0.025 -1707 -653 2.9 93 0.0084 0.015 -2054  
20 0.30 0.53 0.026 305 369 7.1e-03 4.6 0.98 0.020 -1603 -635 2.9 81 0.0078 0.014 -1891

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R2,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 8 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 14.407375452 0.6377829  
2 5.124208493 0.5138322  
3 2.366897162 0.4711463  
4 1.443153319 0.4341293  
5 1.059351774 0.4051553  
6 0.644409830 0.3740431  
7 0.597139091 0.3416047  
8 0.387436292 0.3118740  
9 0.247329236 0.2826577  
10 0.112050211 0.2597786  
11 0.057576688 0.2358313  
12 0.006134471 0.2127153  
13 -0.020916243 0.1922420  
14 -0.031829792 0.1711617  
15 -0.084688321 0.1534352

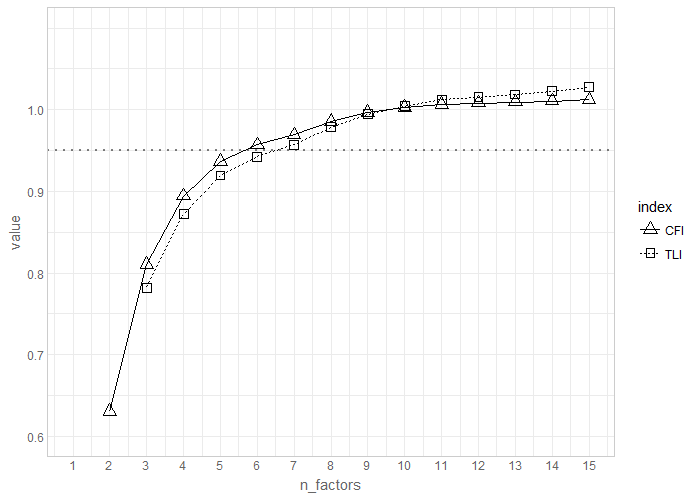
## Fit

psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

ls\_solution <- solve\_factors(R2,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)  
ds\_index %>% print()

n\_factors chisq\_null df\_null chisq df CFI TLI  
1 1 23609.91 1035 27207.9185 989 -0.1614183 -0.2154378  
2 2 23609.91 1035 9291.2546 944 0.6302420 0.5945980  
3 3 23609.91 1035 5183.1288 900 0.8102704 0.7818110  
4 4 23609.91 1035 3245.0735 857 0.8942156 0.8722441  
5 5 23609.91 1035 2246.9747 815 0.9365679 0.9194451  
6 6 23609.91 1035 1750.2273 774 0.9567561 0.9421739  
7 7 23609.91 1035 1425.9096 734 0.9693505 0.9567817  
8 8 23609.91 1035 1020.3561 695 0.9855877 0.9785371  
9 9 23609.91 1035 735.8401 657 0.9965076 0.9944983  
10 10 23609.91 1035 562.3911 620 1.0025519 1.0042600  
11 11 23609.91 1035 431.6603 584 1.0067482 1.0119595  
12 12 23609.91 1035 364.5061 549 1.0081725 1.0154072  
13 13 23609.91 1035 306.6366 515 1.0092299 1.0185493  
14 14 23609.91 1035 245.9683 482 1.0104555 1.0224511  
15 15 23609.91 1035 180.0110 450 1.0119597 1.0275073

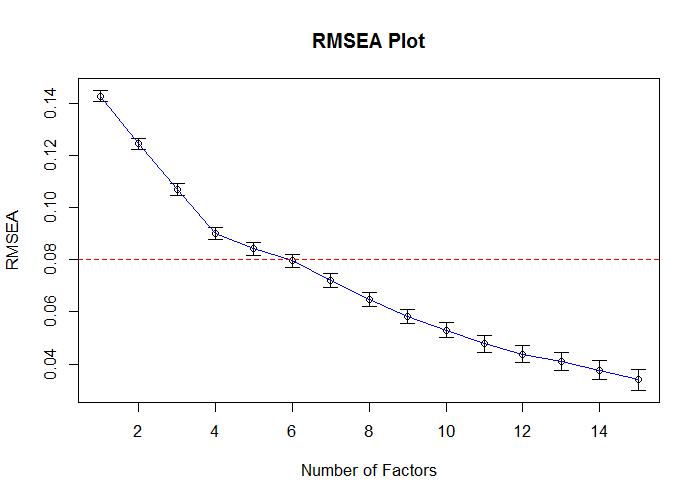
ds\_index %>% plot\_fit\_indices()



## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R2,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 15.00336 13903.0467 989 0 0.14261510 0.14051681 0.14472353  
 [2,] 2 20.97398 10317.3180 944 0 0.12436357 0.12220411 0.12653485  
 [3,] 3 23.98687 7493.2207 900 0 0.10682184 0.10459227 0.10906520  
 [4,] 4 25.98431 5323.4596 857 0 0.09009979 0.08778639 0.09242913  
 [5,] 5 27.71617 4516.7319 815 0 0.08411165 0.08172453 0.08651601  
 [6,] 6 29.11405 3915.2005 774 0 0.07950782 0.07704420 0.08198989  
 [7,] 7 30.48351 3175.5703 734 0 0.07198120 0.06942126 0.07456082  
 [8,] 8 31.51893 2554.7581 695 0 0.06456071 0.06188834 0.06725337  
 [9,] 9 32.49208 2087.0070 657 0 0.05822626 0.05542771 0.06104454  
[10,] 10 33.31483 1736.7420 620 0 0.05296794 0.05002994 0.05592359  
[11,] 11 34.06931 1437.7686 584 0 0.04771953 0.04461352 0.05083777  
[12,] 12 34.77694 1223.7316 549 0 0.04375340 0.04046865 0.04704219  
[13,] 13 35.45410 1068.0991 515 0 0.04090066 0.03743347 0.04436180  
[14,] 14 36.10921 921.6697 482 0 0.03769400 0.03400031 0.04136305  
[15,] 15 36.71795 784.2585 450 0 0.03401476 0.03002067 0.03794733

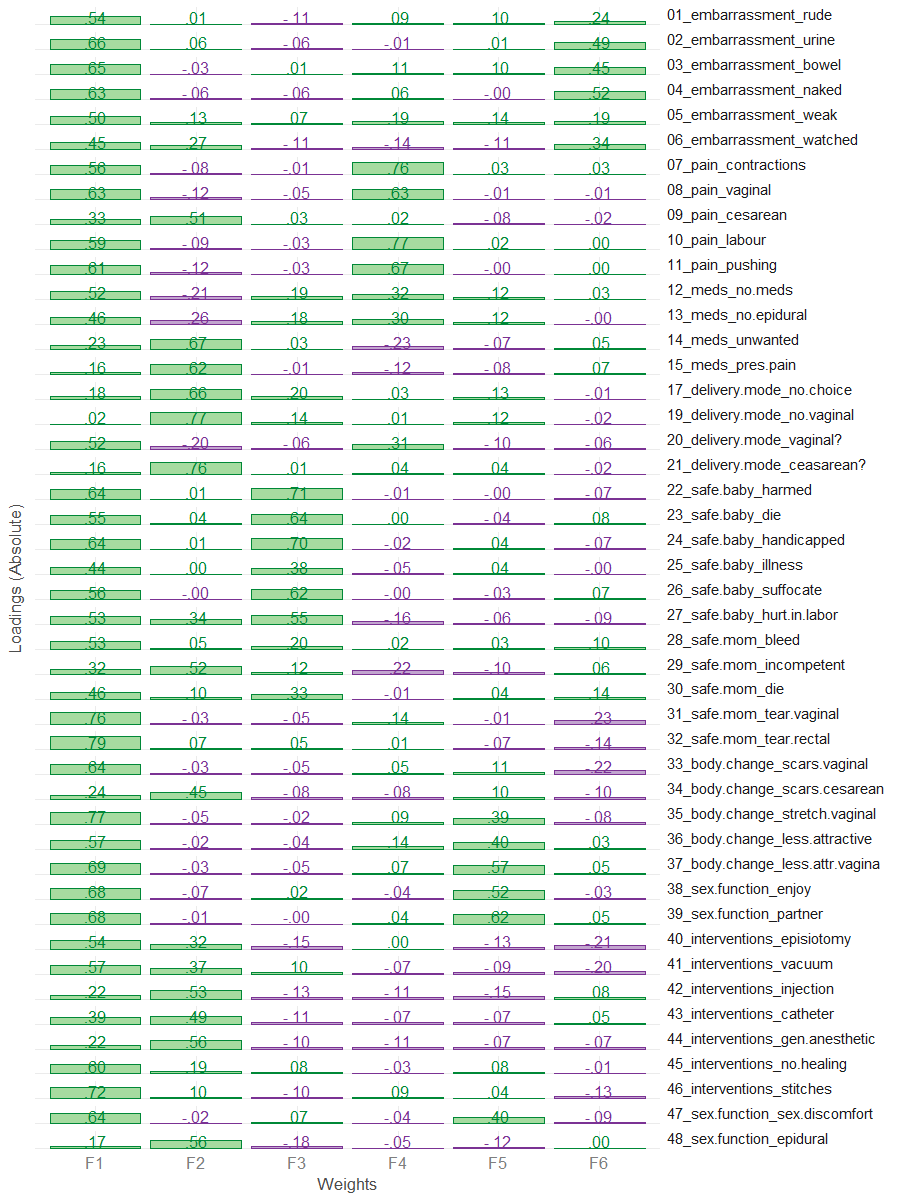
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

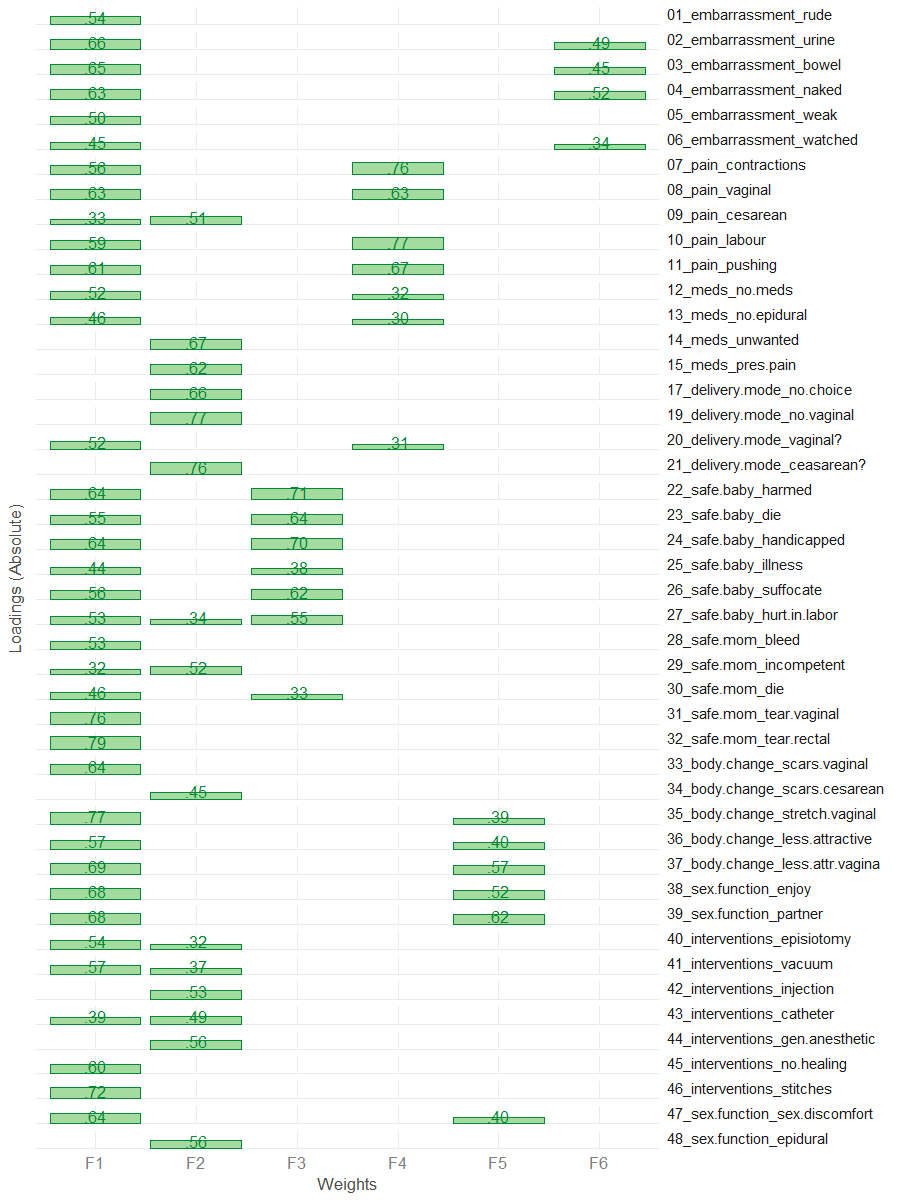
fit\_efa\_2 <- MLFA(  
 Correlation.Matrix = R2,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_2[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)

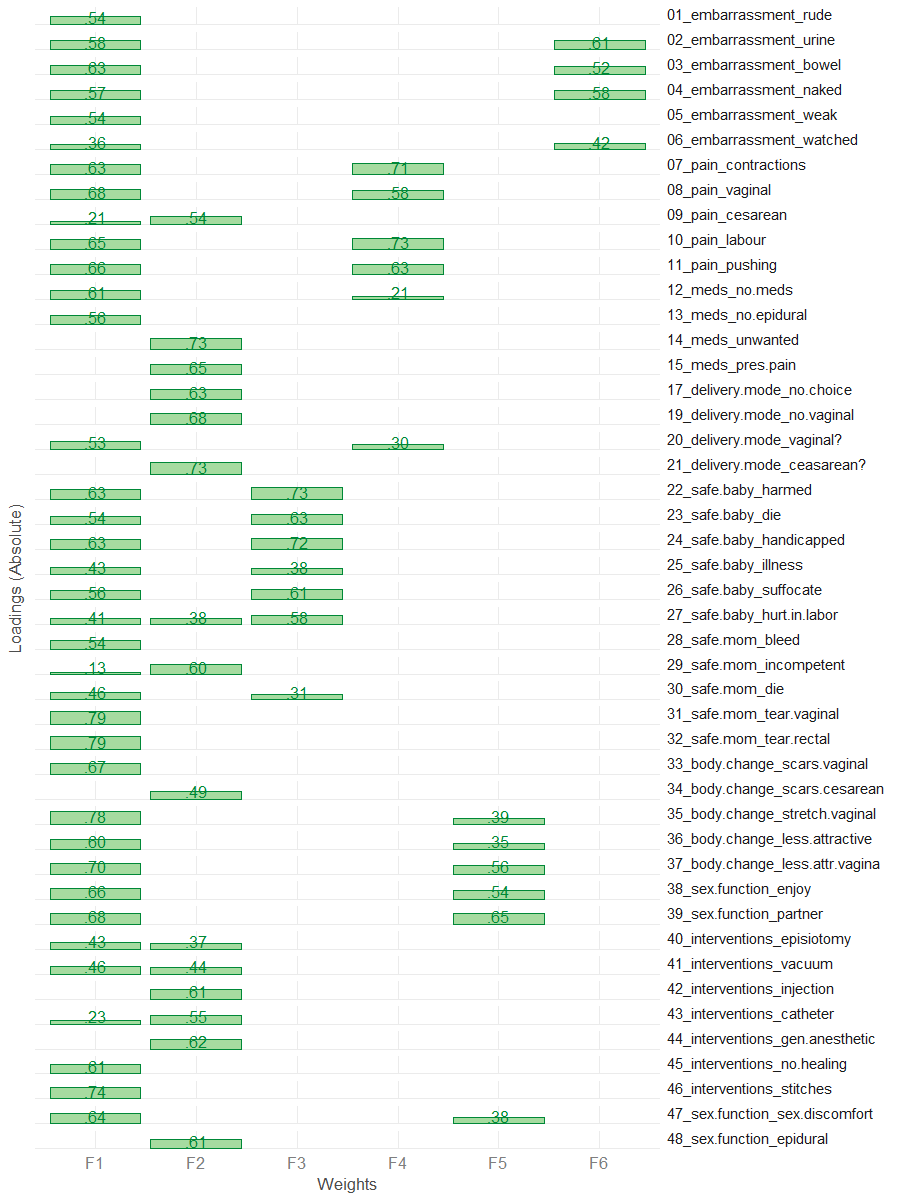


## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

# These values are translated into CFA model and used as starting values  
model\_2 <- FAtoSEM(  
 x = fit\_efa\_2[["Bifactor"]] ,  
 cutoff = 0.30,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

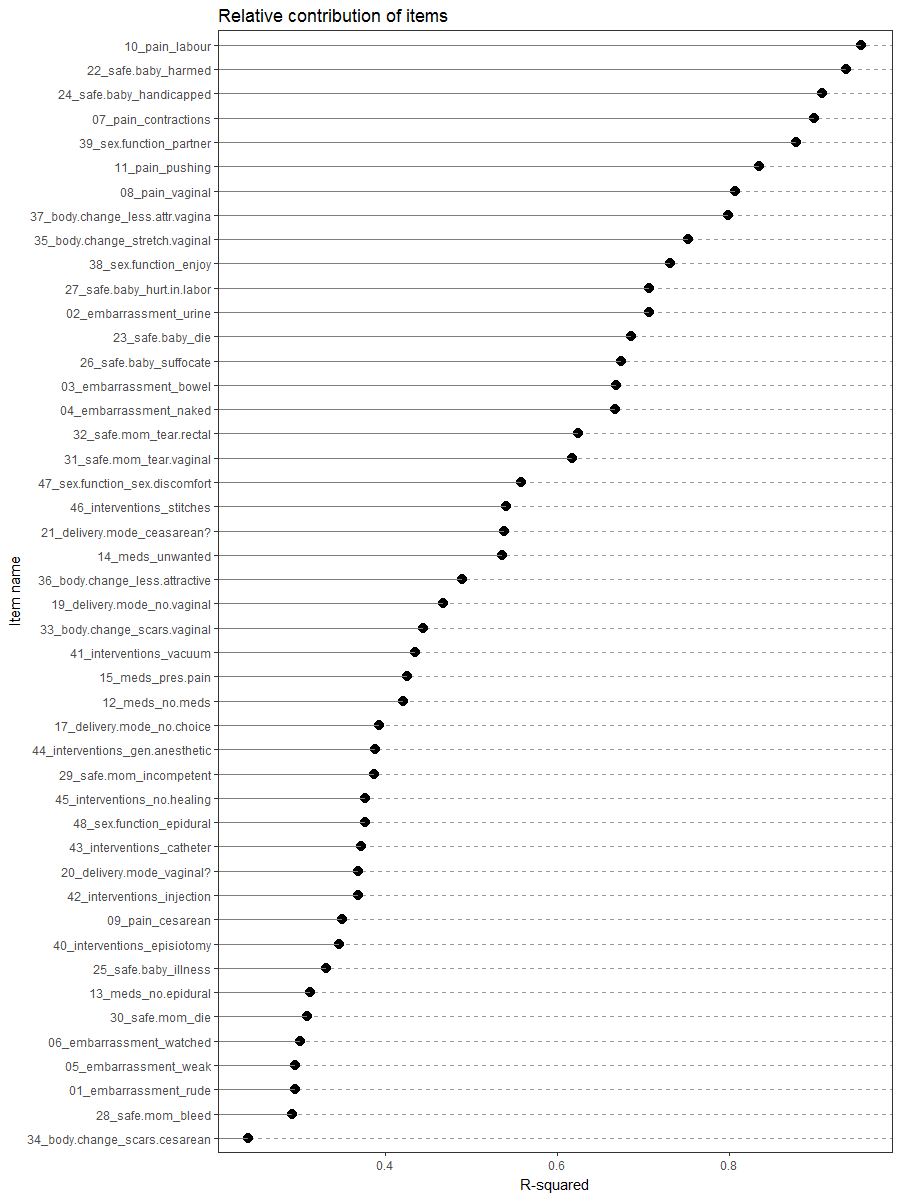
# the model is estimated using sem package  
fit\_2 <- sem::sem(model\_2,R2,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_2)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_2)

Model Chiquare = 5476.325 | df model = 960 | df null = 1035  
Goodness-of-fit index = 0.7139425  
Adjusted Goodness-of-fit index = 0.6778874  
RMSEA index = .0856 90% CI: (.083,.088)  
Comparitive Fit Index (CFI = 0.8052033  
Tucker Lewis Index (TLI/NNFI) = 0.7899848  
Akaike Information Criterion (AIC) = 5718.325  
Bayesian Information Criterion (BIC) = -731.1739

#Relative contribudion of items   
sort(summary(fit\_2)$Rsq) %>% dot\_plot()



# Phase 3

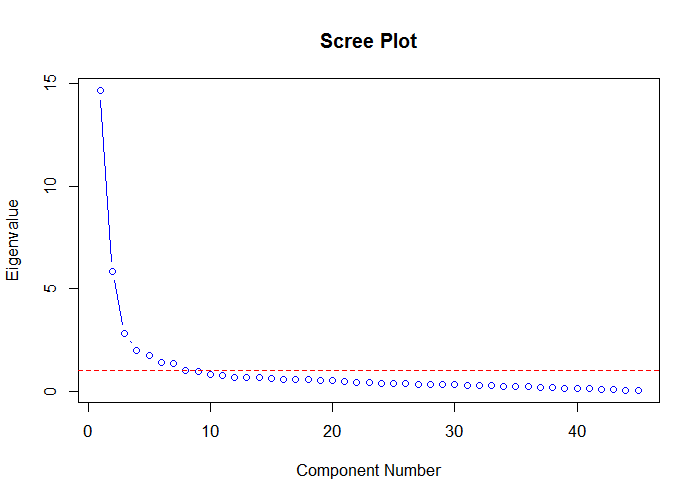
Based on the results from phase 1 we identified that item 27 performs poorly on the scale, according to the consideration we have [outlined](#elimination). It loads above .30 on two subfactors(1), and has a borderline loading on the general factor(2). We also don't think that the removal of this item will affect the interpretabiliyt of the scale adversly (4)

Thus we removed item 27 from the pool of items and repeat the analytical steps.

drop\_items\_3 <- c("foc\_27")  
items\_phase\_3 <- setdiff(items\_phase\_2, drop\_items\_3)  
R3 <- make\_cor(ds, metaData, items\_phase\_3)

## Scree

# Diagnosing number of factors  
Scree.Plot(R3)



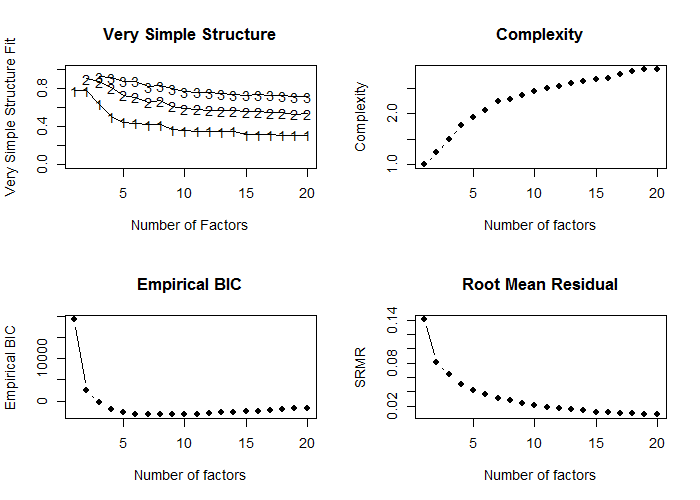
#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R3)),  
 value = eigen(R3)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

eigen value  
1 1 14.6487510  
2 2 5.8489709  
3 3 2.8037872  
4 4 1.9872289  
5 5 1.7314314  
6 6 1.3964740  
7 7 1.3571005  
8 8 1.0309481  
9 9 0.9631267  
10 10 0.8068630  
11 11 0.7538167  
12 12 0.6892828  
13 13 0.6724461  
14 14 0.6547391  
15 15 0.6036542

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R3,n.obs = 643)

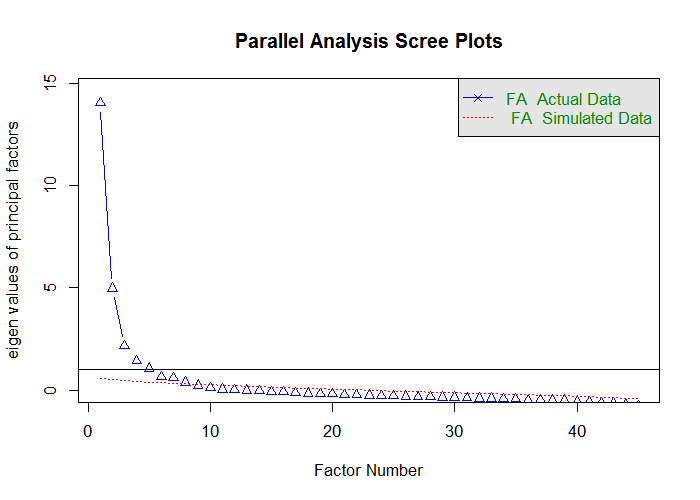


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.77 with 2 factors  
VSS complexity 2 achieves a maximimum of 0.9 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 8 factors   
Empirical BIC achieves a minimum of -3302.52 with 9 factors  
Sample Size adjusted BIC achieves a minimum of -684.33 with 15 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.77 0.00 0.039 945 13224 0.0e+00 62.7 0.77 0.144 7113 10114 1.0 25558 0.1417 0.145 19447  
2 0.77 0.90 0.025 901 9852 0.0e+00 28.2 0.90 0.126 4026 6887 1.2 8345 0.0810 0.085 2519  
3 0.63 0.87 0.021 858 7365 0.0e+00 21.2 0.92 0.110 1817 4541 1.5 5156 0.0636 0.068 -392  
4 0.51 0.81 0.016 816 5174 0.0e+00 16.7 0.94 0.093 -102 2489 1.8 3169 0.0499 0.055 -2107  
5 0.45 0.73 0.015 775 4390 0.0e+00 13.8 0.95 0.087 -621 1840 1.9 2190 0.0415 0.047 -2821  
6 0.44 0.71 0.014 735 3787 0.0e+00 12.0 0.96 0.082 -966 1368 2.1 1616 0.0356 0.041 -3136  
7 0.42 0.65 0.014 696 3163 7.9e-310 10.5 0.96 0.076 -1338 872 2.2 1202 0.0307 0.037 -3298  
8 0.42 0.67 0.013 658 2556 4.8e-221 9.6 0.97 0.069 -1699 390 2.3 975 0.0277 0.034 -3280  
9 0.37 0.61 0.013 621 2069 6.8e-155 8.6 0.97 0.062 -1946 25 2.4 713 0.0237 0.030 -3303  
10 0.36 0.59 0.013 585 1738 1.2e-114 7.9 0.97 0.057 -2045 -188 2.4 539 0.0206 0.027 -3244  
11 0.35 0.58 0.014 550 1465 3.4e-84 7.3 0.97 0.052 -2092 -346 2.5 408 0.0179 0.024 -3149  
12 0.35 0.57 0.015 516 1122 6.6e-47 6.9 0.98 0.044 -2215 -577 2.5 341 0.0164 0.023 -2996  
13 0.34 0.57 0.016 483 973 2.9e-35 6.5 0.98 0.041 -2150 -617 2.6 291 0.0151 0.022 -2832  
14 0.35 0.57 0.017 451 838 1.3e-25 6.1 0.98 0.038 -2078 -646 2.6 230 0.0134 0.020 -2686  
15 0.32 0.55 0.018 420 698 3.7e-16 5.6 0.98 0.034 -2018 -684 2.7 168 0.0115 0.018 -2548  
16 0.32 0.55 0.019 390 606 1.4e-11 5.4 0.98 0.031 -1916 -678 2.7 145 0.0107 0.017 -2376  
17 0.31 0.55 0.021 361 526 3.2e-08 5.2 0.98 0.028 -1809 -663 2.8 127 0.0100 0.017 -2207  
18 0.31 0.54 0.022 333 455 9.6e-06 5.0 0.98 0.026 -1698 -641 2.8 106 0.0091 0.016 -2047  
19 0.31 0.53 0.025 306 393 5.5e-04 4.7 0.98 0.023 -1585 -614 2.9 85 0.0082 0.015 -1894  
20 0.31 0.54 0.028 280 335 1.4e-02 4.6 0.98 0.020 -1476 -587 2.9 74 0.0076 0.014 -1736

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R3,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 8 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 14.058670441 0.5830145  
2 4.985410045 0.4986126  
3 2.159309700 0.4546319  
4 1.442010244 0.4139609  
5 1.058692478 0.3910414  
6 0.642168563 0.3590889  
7 0.587735037 0.3303169  
8 0.385927276 0.3056843  
9 0.235506246 0.2786641  
10 0.102261017 0.2554779  
11 0.034752955 0.2287904  
12 0.002964522 0.2062449  
13 -0.021035732 0.1877216  
14 -0.040372286 0.1607391  
15 -0.089048394 0.1423353

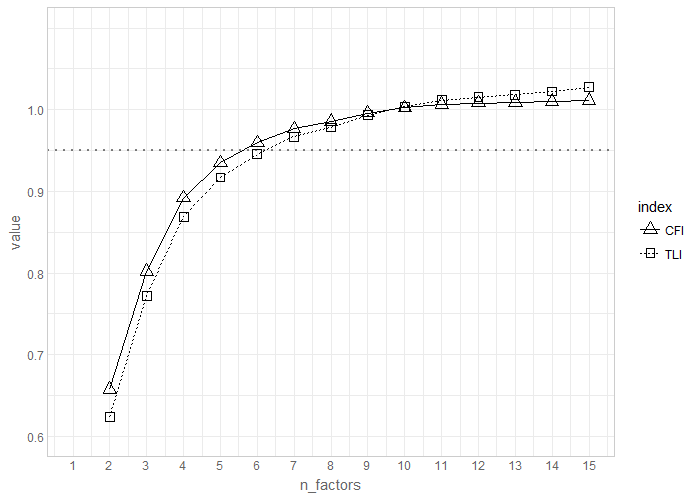
## Fit

psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

ls\_solution <- solve\_factors(R3,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)  
ds\_index %>% print()

n\_factors chisq\_null df\_null chisq df CFI TLI  
1 1 22721.04 990 25557.9339 945 -0.1326165 -0.1865507  
2 2 22721.04 990 8344.9922 901 0.6574489 0.6236120  
3 3 22721.04 990 5155.8623 858 0.8022247 0.7717978  
4 4 22721.04 990 3169.0071 816 0.8917214 0.8686325  
5 5 22721.04 990 2189.8412 775 0.9348931 0.9168311  
6 6 22721.04 990 1616.3575 735 0.9594425 0.9453715  
7 7 22721.04 990 1202.0989 696 0.9767108 0.9668731  
8 8 22721.04 990 974.7391 658 0.9854246 0.9780704  
9 9 22721.04 990 712.9519 621 0.9957686 0.9932544  
10 10 22721.04 990 538.6414 585 1.0021333 1.0036102  
11 11 22721.04 990 407.5341 550 1.0065559 1.0118006  
12 12 22721.04 990 340.8359 516 1.0080605 1.0154650  
13 13 22721.04 990 291.3286 483 1.0088202 1.0180786  
14 14 22721.04 990 230.1720 451 1.0101619 1.0223065  
15 15 22721.04 990 167.5029 420 1.0116192 1.0273881

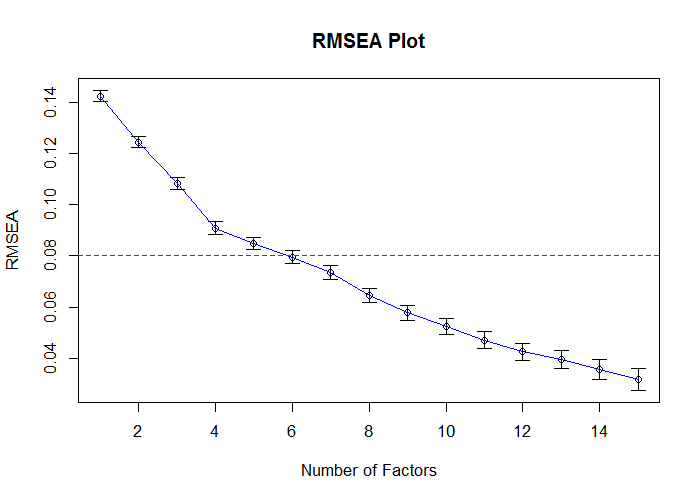
ds\_index %>% plot\_fit\_indices()



## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R3,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 14.64875 13223.7249 945 0.000000e+00 0.14226341 0.14011676 0.14442069  
 [2,] 2 20.49772 9849.7966 901 0.000000e+00 0.12438053 0.12217030 0.12660315  
 [3,] 3 23.30151 7321.5114 858 0.000000e+00 0.10832360 0.10604224 0.11061931  
 [4,] 4 25.28874 5144.8461 816 0.000000e+00 0.09090201 0.08853320 0.09328749  
 [5,] 5 27.02017 4360.7671 775 0.000000e+00 0.08489315 0.08244764 0.08735674  
 [6,] 6 28.41664 3717.1101 735 0.000000e+00 0.07949698 0.07696907 0.08204433  
 [7,] 7 29.77374 3109.9461 696 0.000000e+00 0.07350073 0.07087918 0.07614304  
 [8,] 8 30.80469 2419.6435 658 0.000000e+00 0.06457710 0.06183103 0.06734460  
 [9,] 9 31.76782 1957.6662 621 0.000000e+00 0.05790261 0.05502129 0.06080468  
[10,] 10 32.57468 1617.9716 585 0.000000e+00 0.05244435 0.04941315 0.05549379  
[11,] 11 33.32850 1336.6346 550 0.000000e+00 0.04719953 0.04398959 0.05042151  
[12,] 12 34.01778 1116.8090 516 0.000000e+00 0.04258688 0.03916956 0.04600469  
[13,] 13 34.69023 969.0159 483 0.000000e+00 0.03958988 0.03596783 0.04319892  
[14,] 14 35.34497 823.2240 451 0.000000e+00 0.03585472 0.03195670 0.03971103  
[15,] 15 35.94862 695.9829 420 5.551115e-16 0.03199256 0.02773291 0.03615571

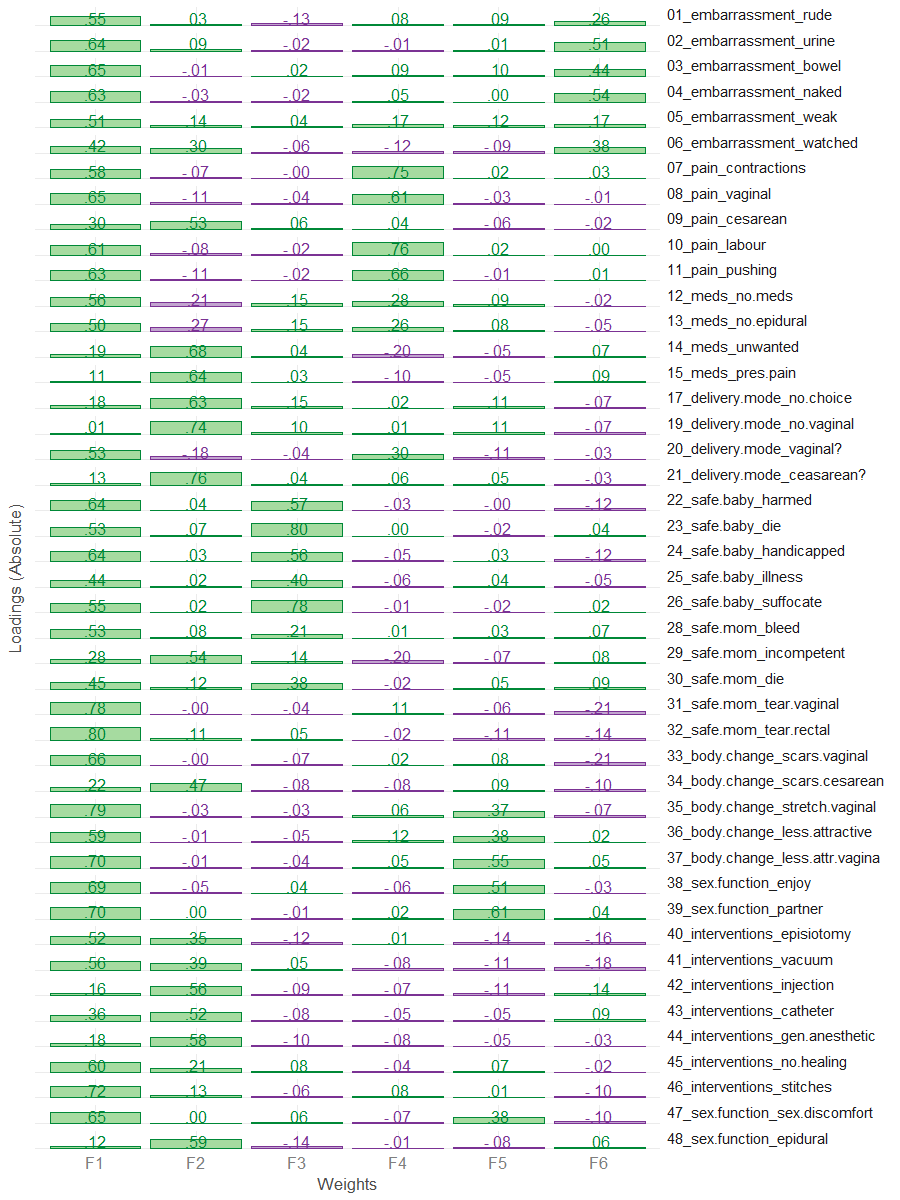
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

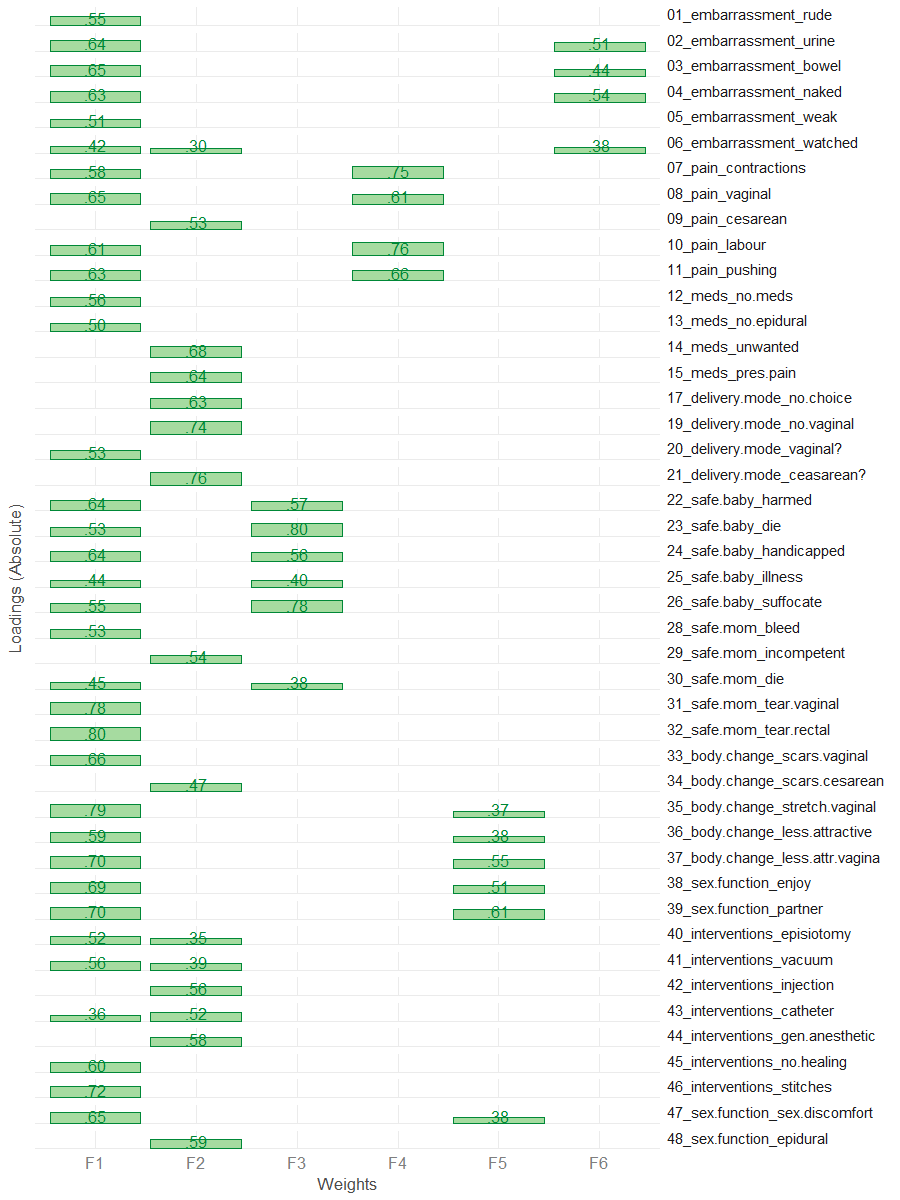
fit\_efa\_3 <- MLFA(  
 Correlation.Matrix = R3,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_3[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



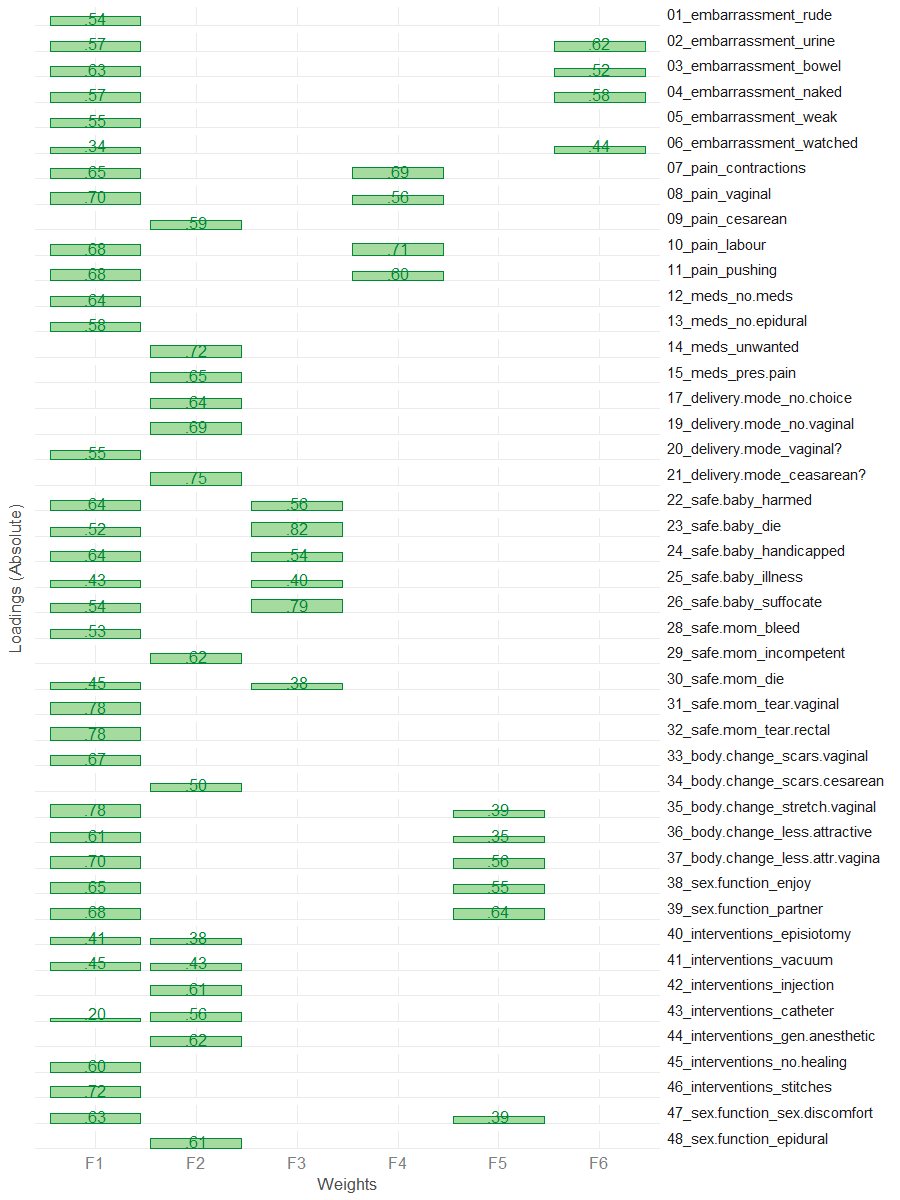
## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

NOTE: we chose to fix the loading of item 6 on subscale 2 to 0 because it was borderline trivial (.30), and because it had substantial loadings on the subscale 6. We decided not to eliminate it from the scale because it had a strong conceptual fit to subscale 6 and contributed to the interpretability of the overall scale.

# These values are translated into CFA model and used as starting values  
model\_3 <- FAtoSEM(  
 x = fit\_efa\_3[["Bifactor"]] ,  
 cutoff = 0.31,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

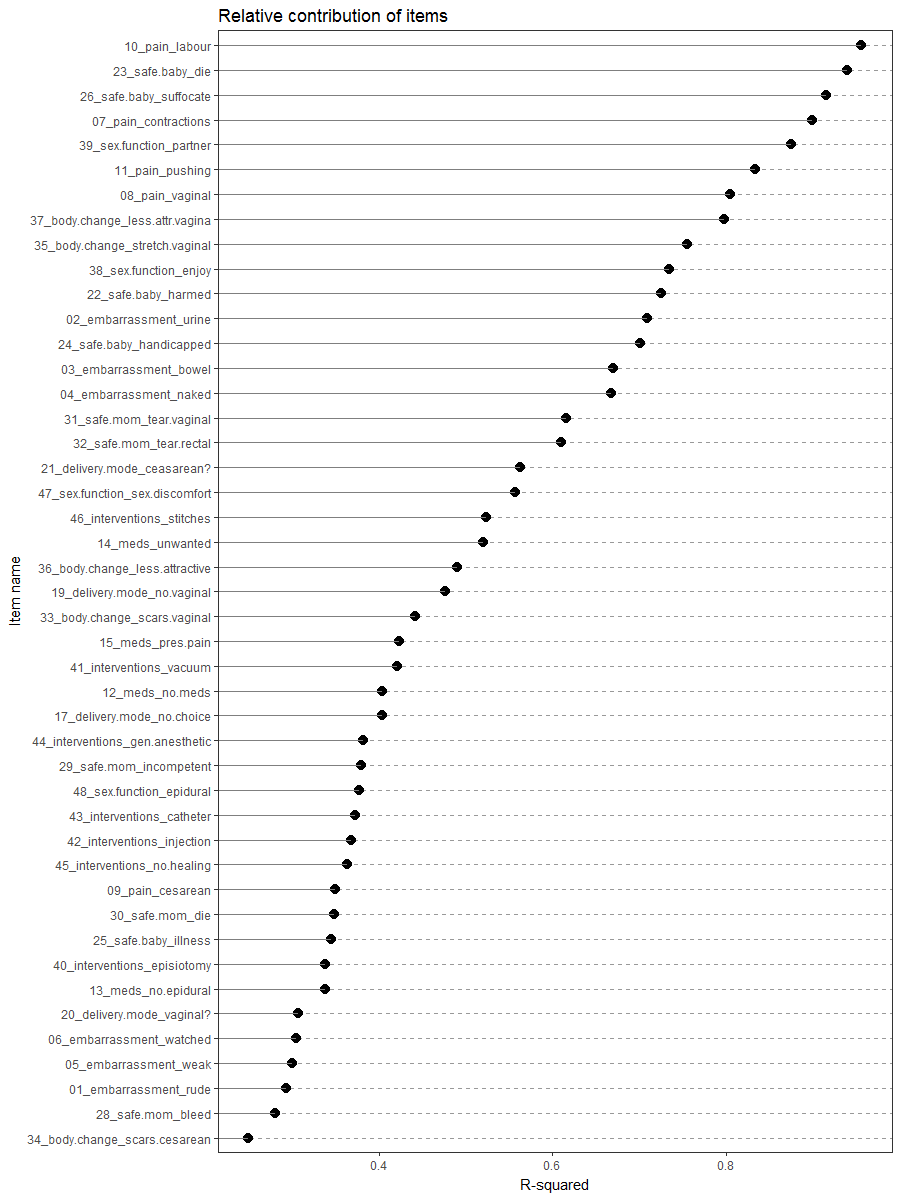
# the model is estimated using sem package  
fit\_3 <- sem::sem(model\_3,R3,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_3)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_3)

Model Chiquare = 5361.8 | df model = 922 | df null = 990  
Goodness-of-fit index = 0.7077978  
Adjusted Goodness-of-fit index = 0.6719856  
RMSEA index = .0866 90% CI: (.084,.089)  
Comparitive Fit Index (CFI = 0.8009555  
Tucker Lewis Index (TLI/NNFI) = 0.7862754  
Akaike Information Criterion (AIC) = 5587.8  
Bayesian Information Criterion (BIC) = -599.9849

#Relative contribudion of items   
sort(summary(fit\_3)$Rsq) %>% dot\_plot()



# Phase 4

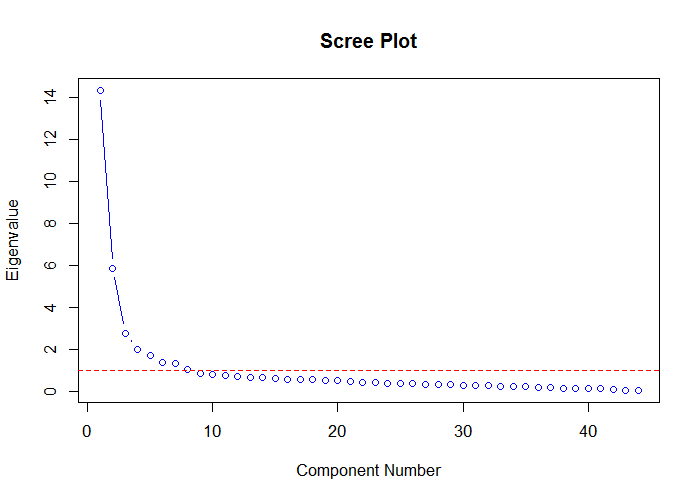
Based on the results from phase 3 we identified that item 28 performs poorly on the scale, according to the consideration we have [outlined](#elimination). It does not have non-trivial loadings on any of the subscale (1), has a modest loading on the general factor(2), and has a relatively small R-square contribution(5).

Thus we removed item 28 from the pool of items and repeat the analytical steps.

drop\_items\_4 <- c("foc\_28")  
items\_phase\_4 <- setdiff(items\_phase\_3, drop\_items\_4)  
R4 <- make\_cor(ds, metaData, items\_phase\_4)

## Scree

# Diagnosing number of factors  
Scree.Plot(R4)



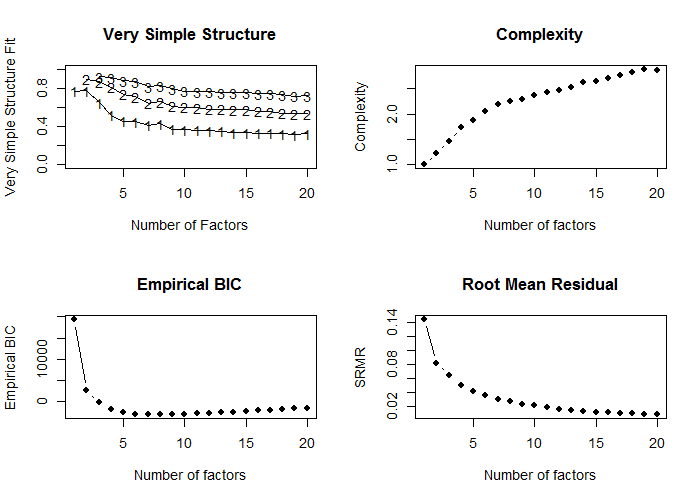
#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R4)),  
 value = eigen(R4)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

eigen value  
1 1 14.3240834  
2 2 5.8485873  
3 3 2.7589871  
4 4 1.9863038  
5 5 1.7204655  
6 6 1.3897594  
7 7 1.3239562  
8 8 1.0274052  
9 9 0.8489992  
10 10 0.8041700  
11 11 0.7456201  
12 12 0.6879099  
13 13 0.6724458  
14 14 0.6388179  
15 15 0.5979046

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R4,n.obs = 643)

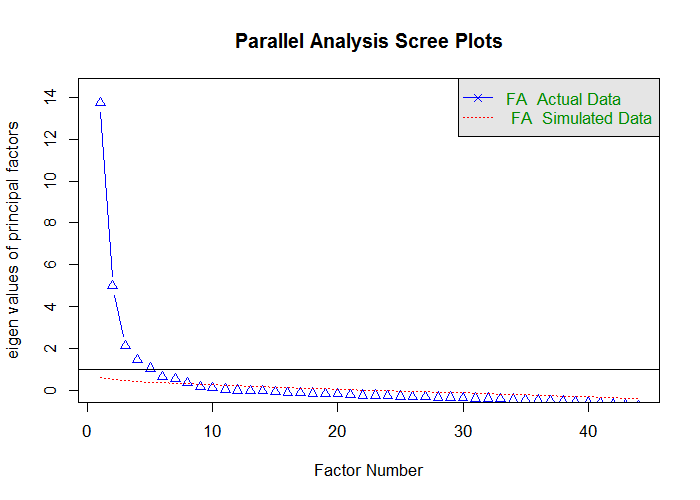


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.77 with 2 factors  
VSS complexity 2 achieves a maximimum of 0.9 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 8 factors   
Empirical BIC achieves a minimum of -3181.92 with 7 factors  
Sample Size adjusted BIC achieves a minimum of -656.88 with 14 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.77 0.00 0.040 902 13025 0.0e+00 61.9 0.77 0.147 7193 10056.6 1.0 25301 0.1442 0.148 19468  
2 0.77 0.90 0.025 859 9653 0.0e+00 27.3 0.90 0.128 4099 6826.2 1.2 8067 0.0814 0.085 2513  
3 0.65 0.87 0.022 817 7188 0.0e+00 20.6 0.92 0.112 1905 4499.0 1.5 5001 0.0641 0.069 -282  
4 0.52 0.81 0.016 776 4997 0.0e+00 16.1 0.94 0.094 -20 2443.5 1.7 3023 0.0498 0.055 -1995  
5 0.45 0.74 0.015 736 4221 0.0e+00 13.2 0.95 0.087 -538 1799.0 1.9 2045 0.0410 0.046 -2714  
6 0.45 0.71 0.014 697 3601 0.0e+00 11.4 0.96 0.082 -906 1306.9 2.0 1503 0.0351 0.041 -3004  
7 0.41 0.65 0.014 659 2995 1.4e-293 9.9 0.96 0.076 -1266 826.3 2.2 1079 0.0298 0.036 -3182  
8 0.43 0.66 0.013 622 2375 1.6e-202 9.2 0.97 0.068 -1647 327.9 2.2 883 0.0269 0.033 -3139  
9 0.37 0.61 0.014 586 1925 4.0e-142 8.1 0.97 0.061 -1864 -3.7 2.3 614 0.0225 0.029 -3176  
10 0.37 0.60 0.014 551 1635 6.3e-108 7.6 0.97 0.057 -1928 -178.2 2.4 506 0.0204 0.027 -3056  
11 0.36 0.60 0.014 517 1359 7.8e-77 7.1 0.97 0.052 -1984 -342.9 2.4 376 0.0176 0.024 -2967  
12 0.36 0.58 0.015 484 1001 2.7e-38 6.6 0.98 0.042 -2128 -591.5 2.5 305 0.0158 0.022 -2824  
13 0.35 0.58 0.016 452 858 1.6e-27 6.2 0.98 0.039 -2065 -629.6 2.5 247 0.0142 0.021 -2676  
14 0.33 0.57 0.017 421 729 8.1e-19 5.7 0.98 0.035 -1994 -656.9 2.6 183 0.0123 0.018 -2539  
15 0.33 0.58 0.019 391 637 4.6e-14 5.5 0.98 0.033 -1891 -649.7 2.6 159 0.0114 0.018 -2369  
16 0.33 0.56 0.020 362 558 1.5e-10 5.3 0.98 0.031 -1783 -633.3 2.7 139 0.0107 0.017 -2202  
17 0.33 0.56 0.022 334 492 3.6e-08 5.1 0.98 0.029 -1667 -606.9 2.8 120 0.0099 0.017 -2040  
18 0.32 0.54 0.024 307 418 2.6e-05 4.8 0.98 0.026 -1567 -592.6 2.8 96 0.0089 0.016 -1889  
19 0.32 0.53 0.027 281 361 9.3e-04 4.5 0.98 0.023 -1456 -564.2 2.9 78 0.0080 0.015 -1739  
20 0.33 0.53 0.030 256 308 1.4e-02 4.4 0.98 0.020 -1347 -534.4 2.9 67 0.0074 0.014 -1588

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R4,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 8 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 13.73717088 0.6089597  
2 4.98409850 0.4855647  
3 2.11160620 0.4487151  
4 1.44444157 0.4109633  
5 1.04706370 0.3804090  
6 0.64272471 0.3497754  
7 0.54265829 0.3207520  
8 0.35350708 0.2939980  
9 0.15896605 0.2705606  
10 0.10302055 0.2487366  
11 0.03236708 0.2193822  
12 -0.01975138 0.1989695  
13 -0.03073731 0.1741913  
14 -0.04237818 0.1556325  
15 -0.09141584 0.1358143

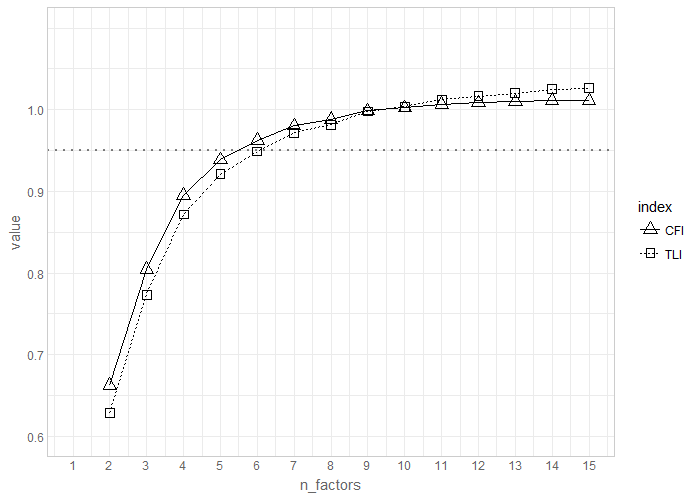
## Fit

psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

ls\_solution <- solve\_factors(R4,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)  
ds\_index %>% print()

n\_factors chisq\_null df\_null chisq df CFI TLI  
1 1 22310.05 946 25300.8085 902 -0.1420496 -0.1977593  
2 2 22310.05 946 8067.2858 859 0.6625975 0.6284252  
3 3 22310.05 946 5001.0721 817 0.8041536 0.7732305  
4 4 22310.05 946 3022.9844 776 0.8948241 0.8717829  
5 5 22310.05 946 2045.3943 736 0.9387104 0.9212229  
6 6 22310.05 946 1502.5982 697 0.9622919 0.9488208  
7 7 22310.05 946 1079.2691 659 0.9803282 0.9717610  
8 8 22310.05 946 882.5096 622 0.9878062 0.9814544  
9 9 22310.05 946 613.5776 586 0.9987092 0.9979162  
10 10 22310.05 946 506.4825 551 1.0020838 1.0035776  
11 11 22310.05 946 376.4499 517 1.0065788 1.0120378  
12 12 22310.05 946 305.2191 484 1.0083683 1.0163562  
13 13 22310.05 946 246.7056 452 1.0096093 1.0201116  
14 14 22310.05 946 183.4404 421 1.0111196 1.0249861  
15 15 22310.05 946 159.3592 391 1.0108426 1.0262329

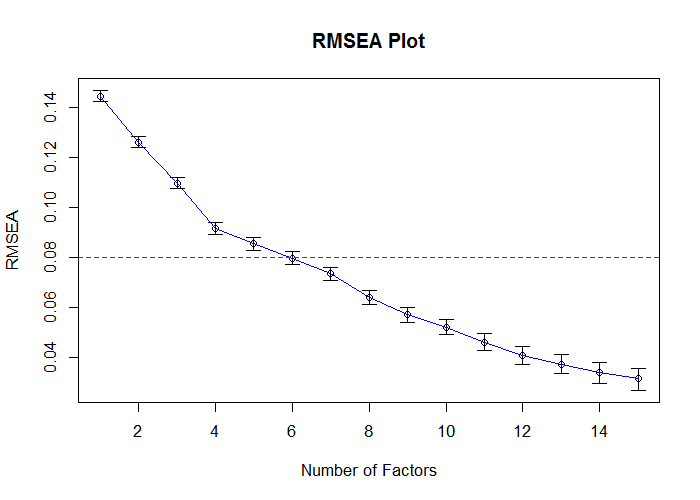
ds\_index %>% plot\_fit\_indices()



## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R4,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 14.32408 13025.1688 902 0.000000e+00 0.14468959 0.14249378 0.14689639  
 [2,] 2 20.17267 9650.7716 859 0.000000e+00 0.12626242 0.12400049 0.12853719  
 [3,] 3 22.93166 7143.6228 817 0.000000e+00 0.10982657 0.10749077 0.11217730  
 [4,] 4 24.91796 4966.7400 776 0.000000e+00 0.09171640 0.08928935 0.09416091  
 [5,] 5 26.63843 4187.2874 736 0.000000e+00 0.08546418 0.08295657 0.08799077  
 [6,] 6 28.02819 3544.0994 697 0.000000e+00 0.07976593 0.07717125 0.08238109  
 [7,] 7 29.35214 2940.7987 659 0.000000e+00 0.07343928 0.07074517 0.07615533  
 [8,] 8 30.37955 2252.7348 622 0.000000e+00 0.06390414 0.06107525 0.06675556  
 [9,] 9 31.22855 1810.9567 586 0.000000e+00 0.05706161 0.05408727 0.06005751  
[10,] 10 32.03272 1507.6455 551 0.000000e+00 0.05200346 0.04887430 0.05515152  
[11,] 11 32.77834 1219.7354 517 0.000000e+00 0.04601327 0.04267891 0.04935789  
[12,] 12 33.46625 1000.8388 484 0.000000e+00 0.04078377 0.03720352 0.04435701  
[13,] 13 34.13869 854.0272 452 0.000000e+00 0.03722125 0.03338719 0.04102542  
[14,] 14 34.77751 726.7271 421 0.000000e+00 0.03363244 0.02947987 0.03771406  
[15,] 15 35.37542 635.8871 391 5.795364e-14 0.03123396 0.02676172 0.03558734

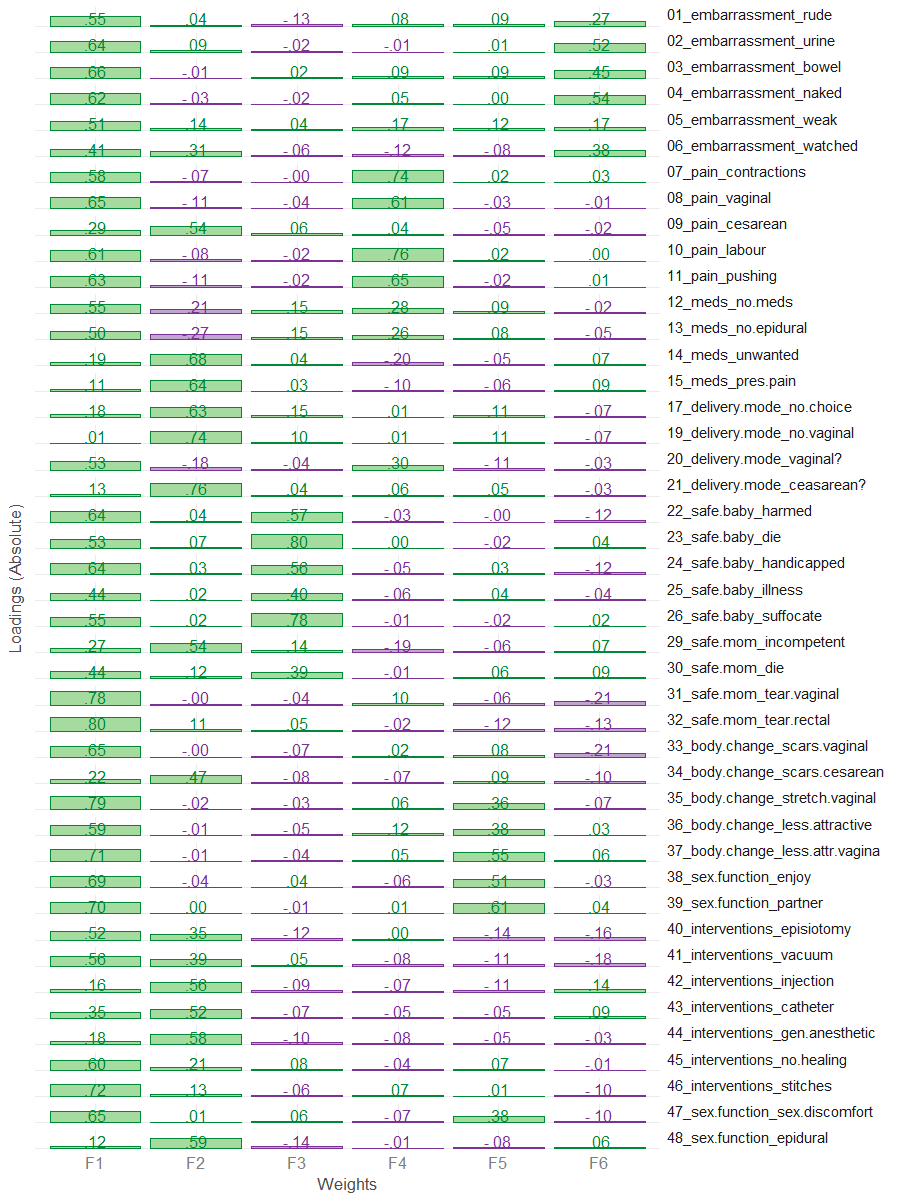
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

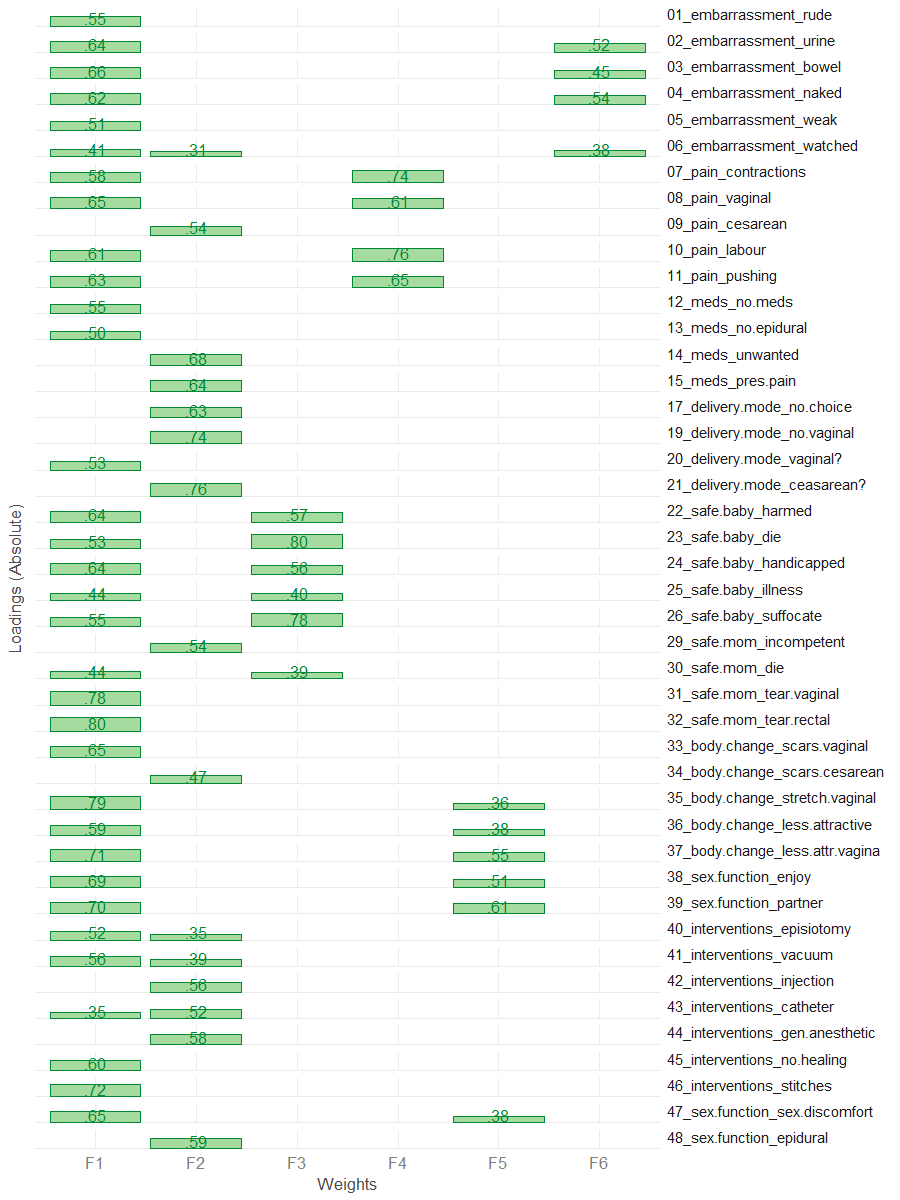
fit\_efa\_4 <- MLFA(  
 Correlation.Matrix = R4,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_4[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



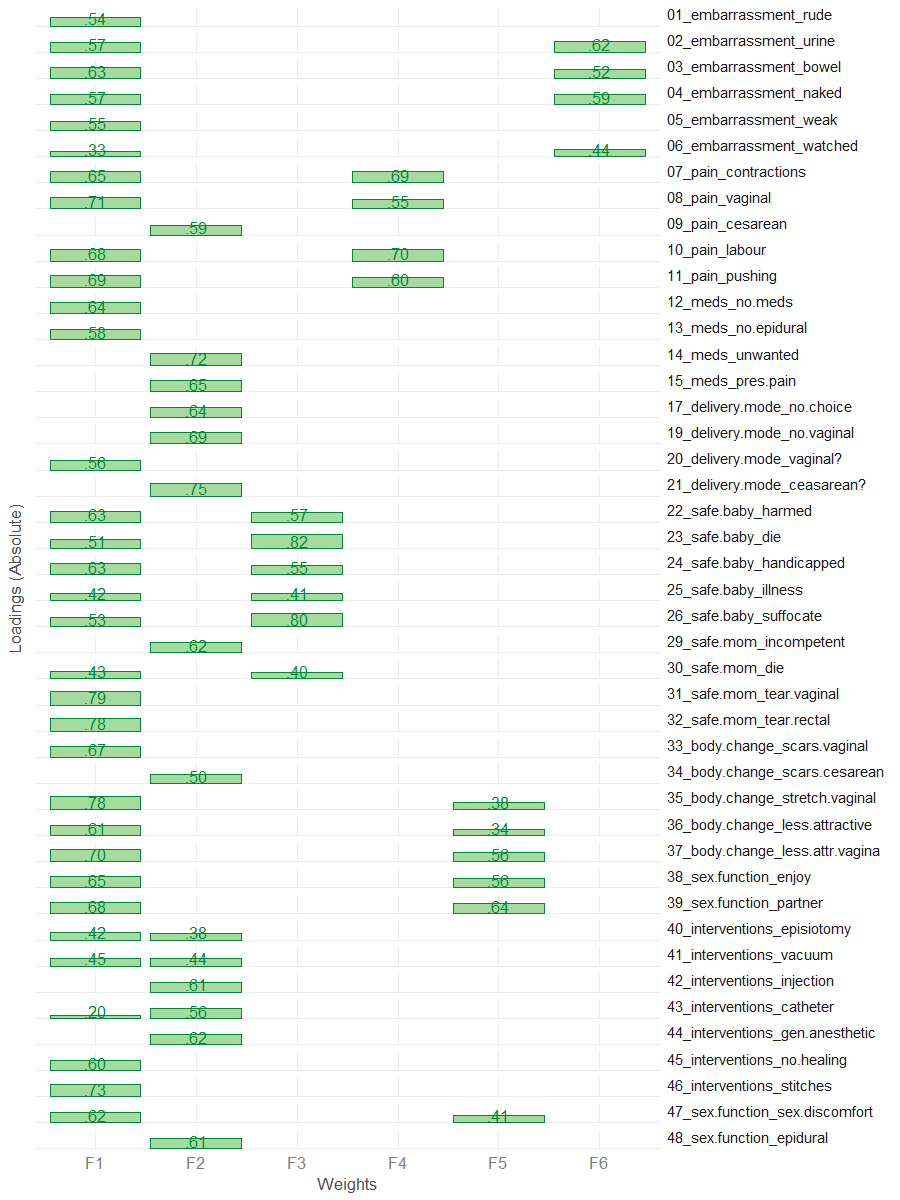
## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

NOTE: we chose to fix the loading of item 6 on subscale 2 to 0 because it was borderline trivial (.31), and because it had substantial loadings on the subscale 6. We decided not to eliminate it from the scale because it had a strong conceptual fit to subscale 6 and contributed to the interpretability of the overall scale.

# These values are translated into CFA model and used as starting values  
model\_4 <- FAtoSEM(  
 x = fit\_efa\_4[["Bifactor"]] ,  
 cutoff = 0.31,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

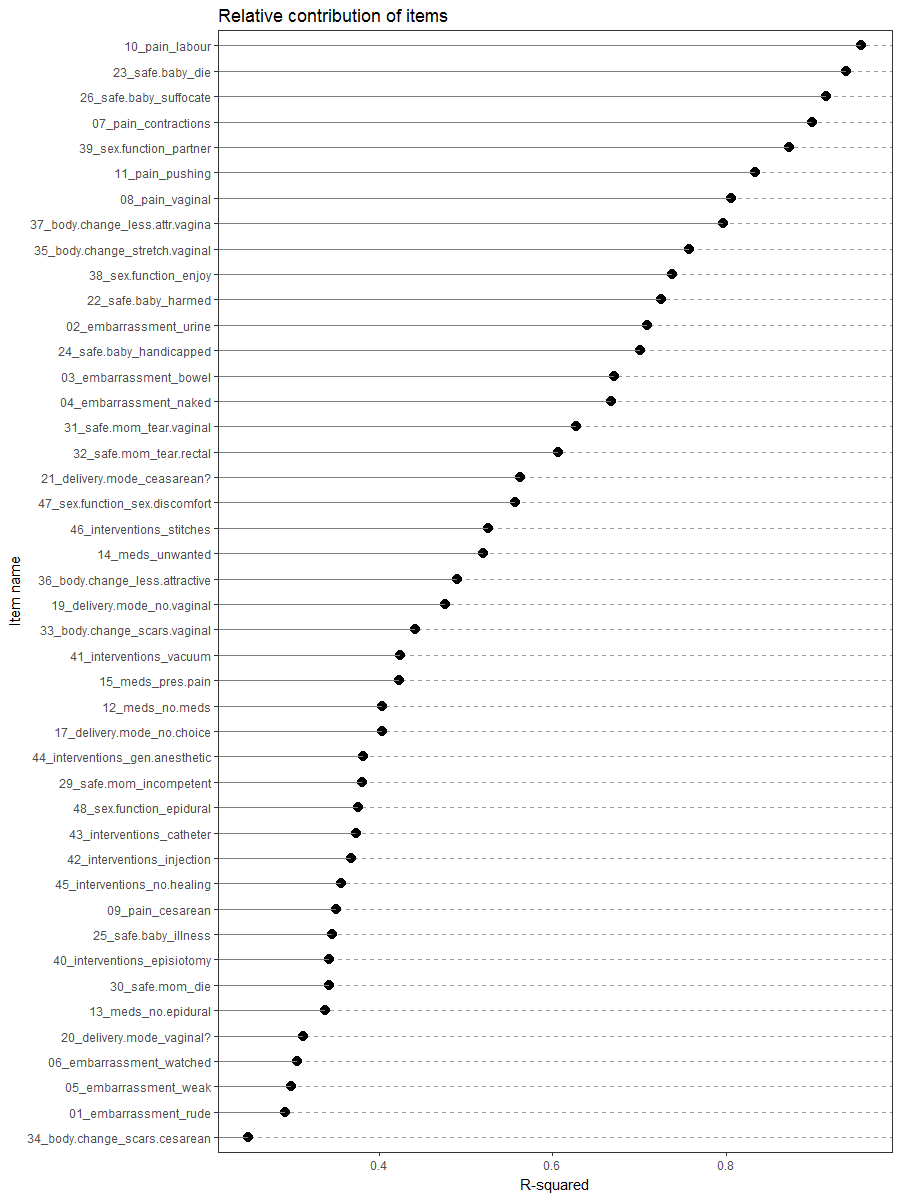
# the model is estimated using sem package  
fit\_4 <- sem::sem(model\_4,R4,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_4)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_4)

Model Chiquare = 5121.14 | df model = 879 | df null = 946  
Goodness-of-fit index = 0.7148281  
Adjusted Goodness-of-fit index = 0.6788166  
RMSEA index = .0867 90% CI: (.084,.089)  
Comparitive Fit Index (CFI = 0.8064366  
Tucker Lewis Index (TLI/NNFI) = 0.7916826  
Akaike Information Criterion (AIC) = 5343.14  
Bayesian Information Criterion (BIC) = -562.6013

#Relative contribudion of items   
sort(summary(fit\_4)$Rsq) %>% dot\_plot()



# Phase 5

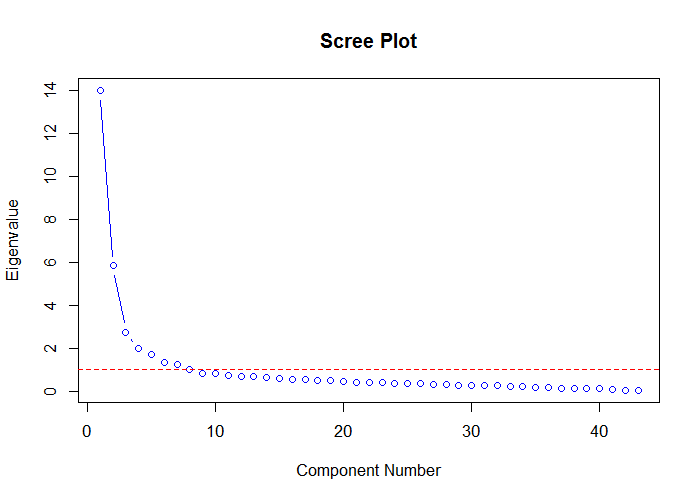
Based on the results from phase 3 we identified that item 5 performs poorly on the scale, according to the consideration we have [outlined](#elimination). It does not have non-trivial loadings on any of the subscale (1), has a modest loading on the general factor(2), and has a relatively small R-square contribution(5).

Thus we removed item 5 from the pool of items and repeat the analytical steps.

drop\_items\_5 <- c("foc\_05")  
items\_phase\_5 <- setdiff(items\_phase\_4, drop\_items\_5)  
R5 <- make\_cor(ds, metaData, items\_phase\_5)

## Scree

# Diagnosing number of factors  
Scree.Plot(R5)



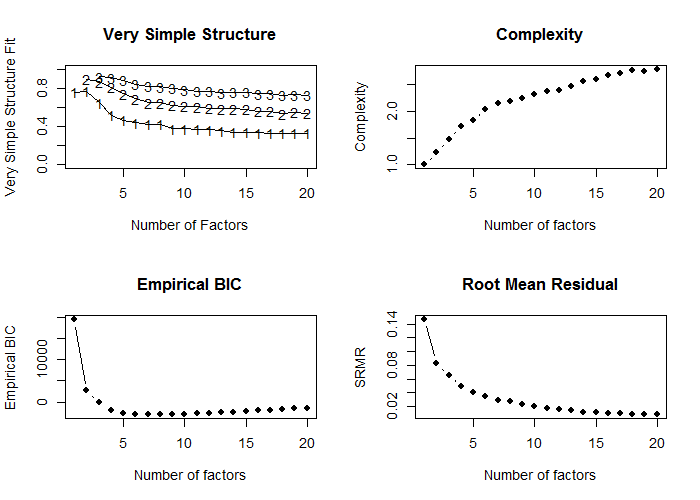
#The first 15 eigen values  
data.frame(  
 eigen = c(1:nrow(R5)),  
 value = eigen(R5)$values  
) %>%  
 dplyr::filter(eigen < 16) %>%  
 print()

eigen value  
1 1 13.9937705  
2 2 5.8485737  
3 3 2.7530230  
4 4 1.9861503  
5 5 1.7072120  
6 6 1.3495091  
7 7 1.2617484  
8 8 1.0211763  
9 9 0.8488369  
10 10 0.8030910  
11 11 0.7321124  
12 12 0.6872050  
13 13 0.6643892  
14 14 0.6281910  
15 15 0.5933009

## MAP

psych::nfactors call is applied, producing Very Simple Structure, Velicer's MAP, and other criteria to determine the appropriate number of factors. See [documentation](http://www.personality-project.org/r/html/VSS.html)

# MAP  
psych::nfactors(R5,n.obs = 643)

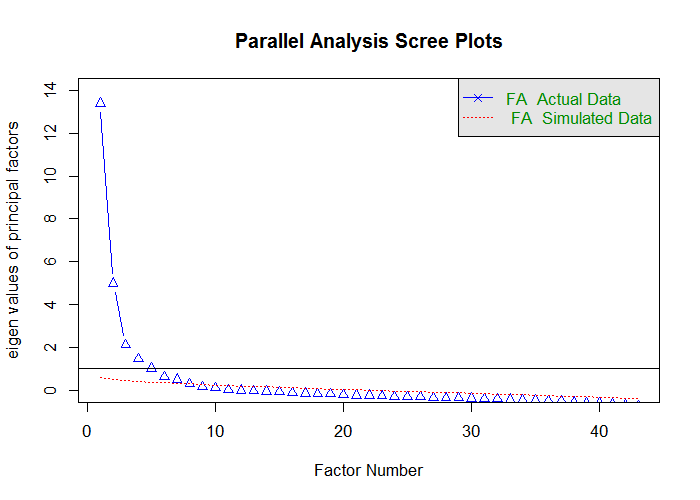


Number of factors  
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,   
 n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)  
VSS complexity 1 achieves a maximimum of 0.77 with 2 factors  
VSS complexity 2 achieves a maximimum of 0.9 with 2 factors  
The Velicer MAP achieves a minimum of 0.01 with 8 factors   
Empirical BIC achieves a minimum of -3015.61 with 7 factors  
Sample Size adjusted BIC achieves a minimum of -585.82 with 14 factors  
  
Statistics by number of factors   
 vss1 vss2 map dof chisq prob sqresid fit RMSEA BIC SABIC complex eChisq SRMR eCRMS eBIC  
1 0.76 0.00 0.042 860 12930 0.0e+00 61.2 0.76 0.150 7369 10100 1.0 25151 0.1472 0.151 19590  
2 0.77 0.90 0.026 818 9564 0.0e+00 26.6 0.90 0.131 4275 6872 1.2 7917 0.0826 0.087 2628  
3 0.65 0.87 0.022 777 7094 0.0e+00 20.0 0.92 0.114 2070 4537 1.5 4886 0.0649 0.070 -138  
4 0.52 0.81 0.017 737 4906 0.0e+00 15.5 0.94 0.095 140 2480 1.7 2896 0.0499 0.055 -1870  
5 0.46 0.74 0.016 698 4148 0.0e+00 12.6 0.95 0.089 -365 1851 1.8 1945 0.0409 0.047 -2568  
6 0.44 0.69 0.015 660 3534 0.0e+00 10.9 0.96 0.084 -733 1362 2.0 1439 0.0352 0.041 -2829  
7 0.42 0.65 0.015 623 2927 5.1e-294 9.4 0.96 0.077 -1101 877 2.1 1013 0.0295 0.036 -3016  
8 0.43 0.65 0.014 587 2320 5.9e-204 8.8 0.97 0.069 -1476 388 2.2 862 0.0272 0.034 -2934  
9 0.38 0.63 0.014 552 1896 1.2e-146 7.8 0.97 0.063 -1673 79 2.2 600 0.0227 0.029 -2969  
10 0.38 0.61 0.014 518 1606 1.3e-111 7.3 0.97 0.059 -1744 -99 2.3 493 0.0206 0.027 -2857  
11 0.37 0.61 0.015 485 1331 5.7e-80 6.7 0.97 0.054 -1805 -265 2.4 363 0.0177 0.024 -2773  
12 0.36 0.59 0.016 453 977 1.7e-40 6.3 0.98 0.044 -1953 -514 2.4 292 0.0159 0.022 -2637  
13 0.36 0.59 0.017 422 832 4.5e-29 5.9 0.98 0.040 -1897 -557 2.5 236 0.0143 0.021 -2493  
14 0.34 0.59 0.018 392 704 4.0e-20 5.4 0.98 0.037 -1830 -586 2.6 171 0.0121 0.018 -2364  
15 0.34 0.58 0.020 363 613 4.7e-15 5.2 0.98 0.034 -1735 -582 2.6 147 0.0113 0.018 -2200  
16 0.33 0.56 0.022 335 539 9.7e-12 5.0 0.98 0.032 -1627 -564 2.7 130 0.0106 0.017 -2036  
17 0.33 0.56 0.023 308 474 3.7e-09 4.8 0.98 0.031 -1518 -540 2.7 112 0.0098 0.017 -1880  
18 0.33 0.54 0.026 282 401 4.1e-06 4.5 0.98 0.027 -1423 -527 2.8 91 0.0088 0.016 -1733  
19 0.33 0.55 0.029 257 352 7.6e-05 4.4 0.98 0.026 -1310 -494 2.7 78 0.0082 0.015 -1583  
20 0.33 0.54 0.032 233 329 3.5e-05 4.3 0.98 0.027 -1178 -438 2.8 78 0.0082 0.016 -1428

## Parallel

psych::fa.parallel call is applied, comparing the number of factors in the correlation matrix to random "parallel" matrices. For details, see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa.parallel?)

pa\_results <- psych::fa.parallel(R5,643,fm = "ml",fa="fa")



Parallel analysis suggests that the number of factors = 8 and the number of components = NA

ds\_pa <- data.frame(  
 observed\_eigens = pa\_results$fa.values,  
 simulated\_eigens = pa\_results$fa.sim  
) %>% head(15) %>% print()

observed\_eigens simulated\_eigens  
1 13.409132874 0.5852112  
2 4.983414832 0.4924904  
3 2.106034339 0.4439112  
4 1.444644011 0.4036302  
5 1.022113644 0.3731454  
6 0.627427125 0.3397537  
7 0.494522349 0.3165334  
8 0.303714564 0.2857996  
9 0.153846436 0.2574921  
10 0.103496336 0.2343488  
11 0.006408525 0.2147485  
12 -0.019857351 0.1908902  
13 -0.033101598 0.1702708  
14 -0.053269349 0.1453240  
15 -0.093190585 0.1250946

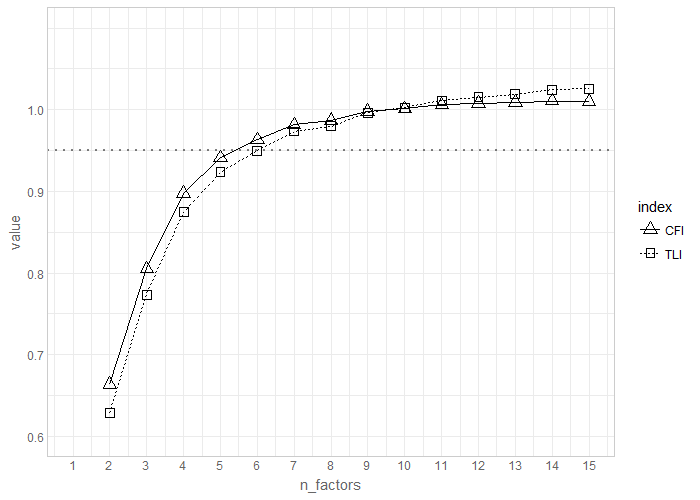
## Fit

psych::fa call is applied to conduct maximum likelihood factor analysls (fm="ml") in order to obtain the chi-square of the proposed models, which incrementally increase the number of retained factors. CFI and TLI indices are then computed from the produced criteria. For details on psych::fa see [documentation](https://www.rdocumentation.org/packages/psych/versions/1.6.9/topics/fa)

ls\_solution <- solve\_factors(R5,min=1,max=15,sample\_size = 643)  
ds\_index <- get\_indices(ls\_solution)  
ds\_index %>% print()

n\_factors chisq\_null df\_null chisq df CFI TLI  
1 1 21986.15 903 25150.6363 860 -0.1521349 -0.2097417  
2 2 21986.15 903 7917.1765 818 0.6632773 0.6282877  
3 3 21986.15 903 4886.3462 777 0.8050886 0.7734814  
4 4 21986.15 903 2896.0445 737 0.8975938 0.8745281  
5 5 21986.15 903 1945.0540 698 0.9408507 0.9234787  
6 6 21986.15 903 1438.6500 660 0.9630677 0.9494699  
7 7 21986.15 903 1012.8028 623 0.9815112 0.9732016  
8 8 21986.15 903 861.8346 587 0.9869643 0.9799467  
9 9 21986.15 903 600.2811 552 0.9977100 0.9962538  
10 10 21986.15 903 492.7624 518 1.0011971 1.0020867  
11 11 21986.15 903 363.2638 485 1.0057741 1.0107505  
12 12 21986.15 903 292.2585 453 1.0076242 1.0151978  
13 13 21986.15 903 235.8425 422 1.0088297 1.0188938  
14 14 21986.15 903 170.8848 392 1.0104878 1.0241593  
15 15 21986.15 903 147.2154 363 1.0102349 1.0254605

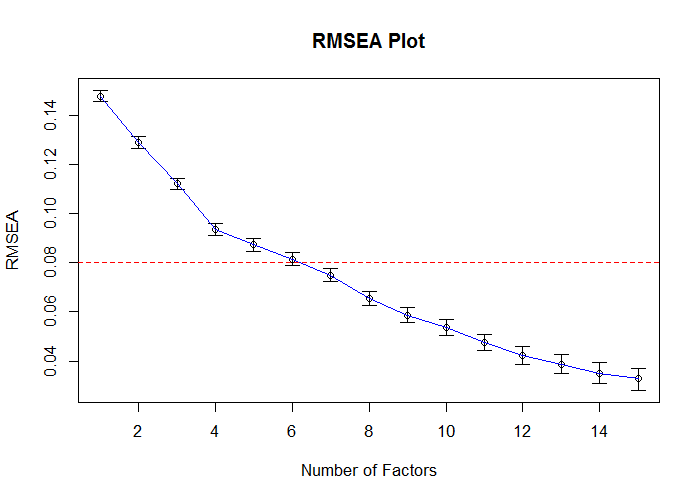
ds\_index %>% plot\_fit\_indices()



## RMSEA

RMSEA diagnostic is conducted using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger. The routine relies on the maxim likelihood factor analysis conducted by stats::factanal call. For details on the latter see [here](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/factanal.html)

FA.Stats(Correlation.Matrix = R5,n.obs = 643,n.factors = 1:15,RMSEA.cutoff = .08)



Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
 [1,] 1 13.99377 12930.1304 860 0.000000e+00 0.14785610 0.14560905 0.15011445  
 [2,] 2 19.84234 9561.7857 818 0.000000e+00 0.12903442 0.12671886 0.13136325  
 [3,] 3 22.59537 7050.1229 777 0.000000e+00 0.11214086 0.10974875 0.11454844  
 [4,] 4 24.58152 4875.2856 737 0.000000e+00 0.09352096 0.09103473 0.09602541  
 [5,] 5 26.28873 4112.5593 698 0.000000e+00 0.08729153 0.08472192 0.08988099  
 [6,] 6 27.63824 3471.7930 660 0.000000e+00 0.08146146 0.07880130 0.08414313  
 [7,] 7 28.89999 2874.9150 623 0.000000e+00 0.07503510 0.07227194 0.07782141  
 [8,] 8 29.92116 2199.0201 587 0.000000e+00 0.06540315 0.06250227 0.06832818  
 [9,] 9 30.77000 1774.4539 552 0.000000e+00 0.05873260 0.05568551 0.06180342  
[10,] 10 31.57309 1470.7884 518 0.000000e+00 0.05352613 0.05032034 0.05675360  
[11,] 11 32.30520 1189.7536 485 0.000000e+00 0.04757516 0.04416467 0.05100009  
[12,] 12 32.99241 976.2787 453 0.000000e+00 0.04241797 0.03876589 0.04606990  
[13,] 13 33.65680 829.8782 422 0.000000e+00 0.03880087 0.03489595 0.04268571  
[14,] 14 34.28499 702.2533 392 0.000000e+00 0.03511136 0.03088971 0.03927637  
[15,] 15 34.87829 612.5021 363 4.884981e-15 0.03272022 0.02818422 0.03715811

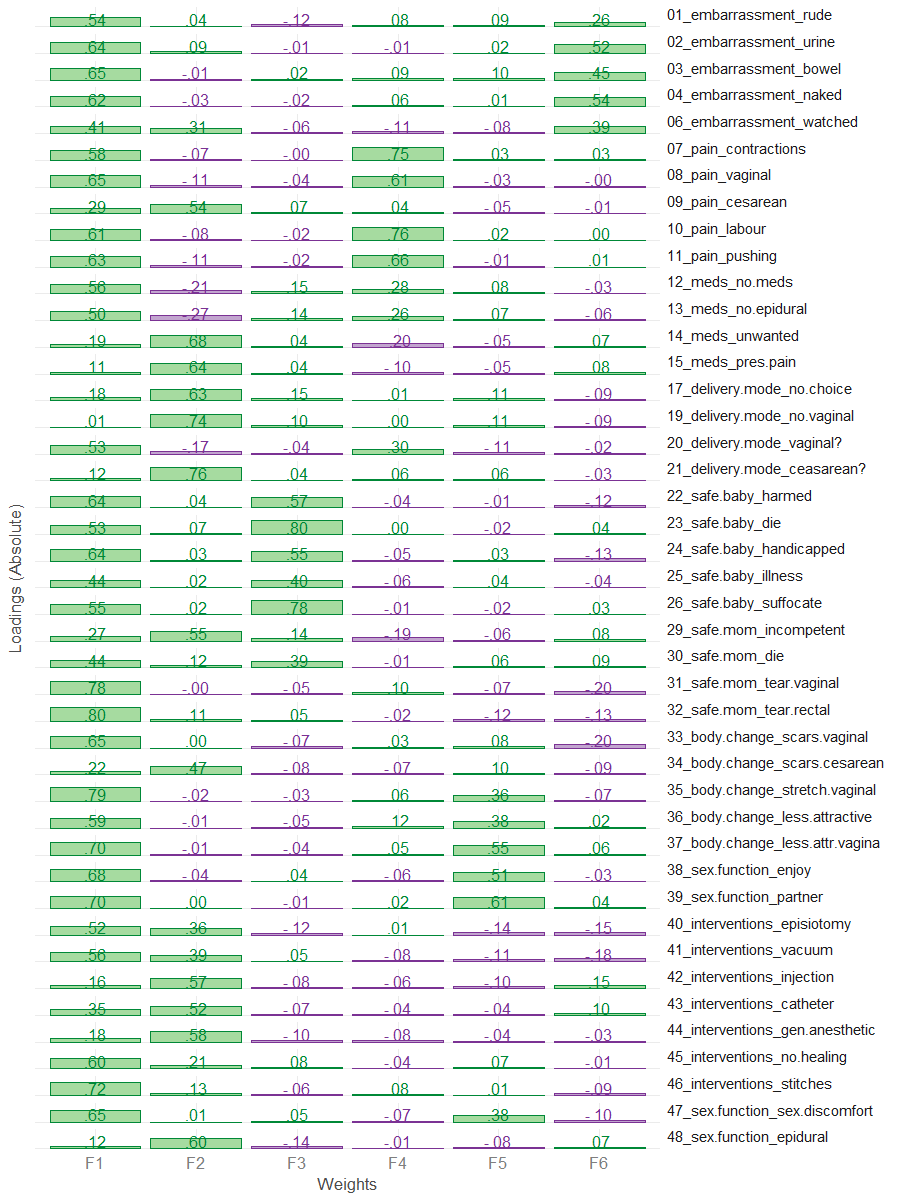
## Estimate

Using [Advanced Factor Function](http://statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt) by James Steiger, we conduct maximum likelihood factor analysis, by obtaining the unrotated solution from stats::factanal call and then rotating solution using gradient projection algorithms (Bernaards & Jennrich, 2005).

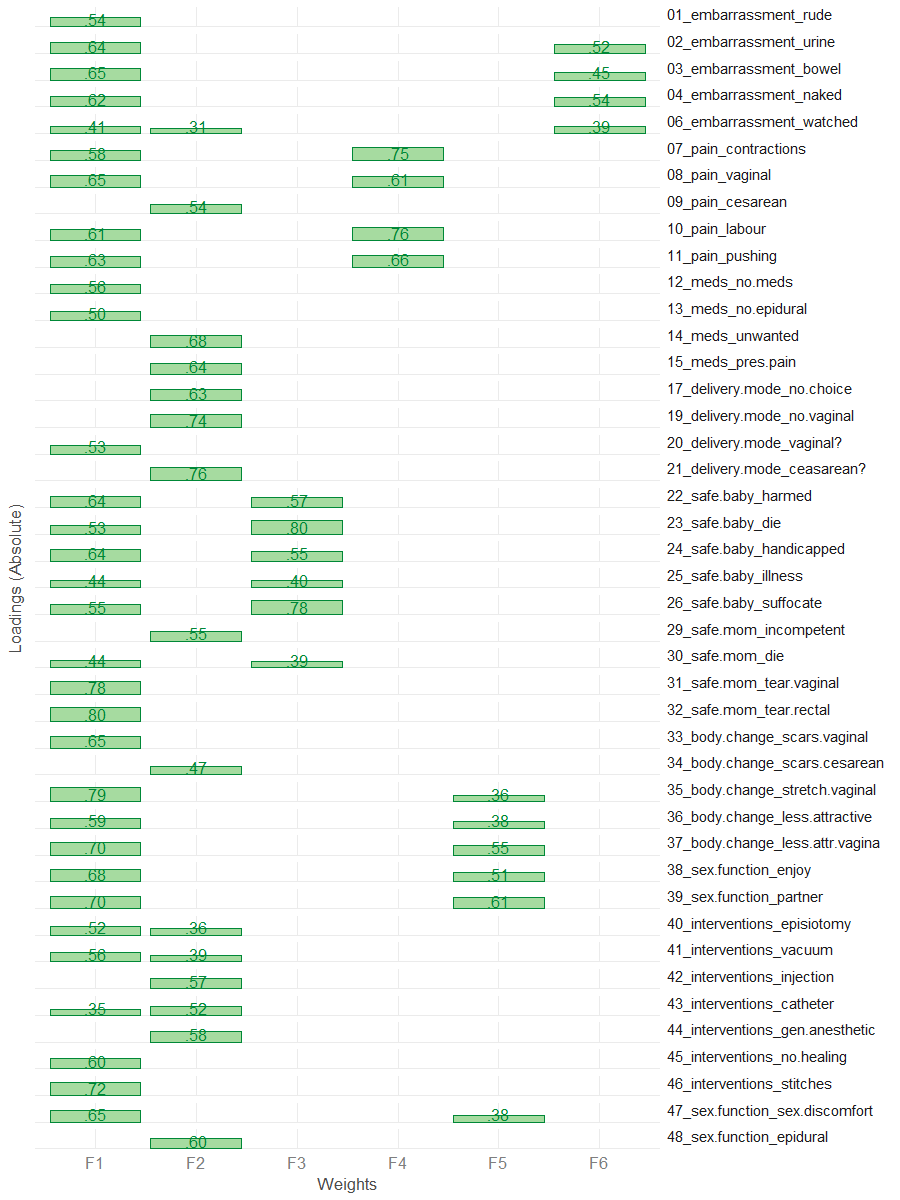
fit\_efa\_5 <- MLFA(  
 Correlation.Matrix = R5,  
 n.factors = 6,  
 n.obs = 643,  
 sort = FALSE  
)

This will take a moment..........exiting

#Loadings from the EFA solution\n")  
f\_pattern <- fit\_efa\_5[['Bifactor']]$F   
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# Loadings above threashold (.3) are masked to see the simpler structure  
f\_pattern[f\_pattern<.30] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



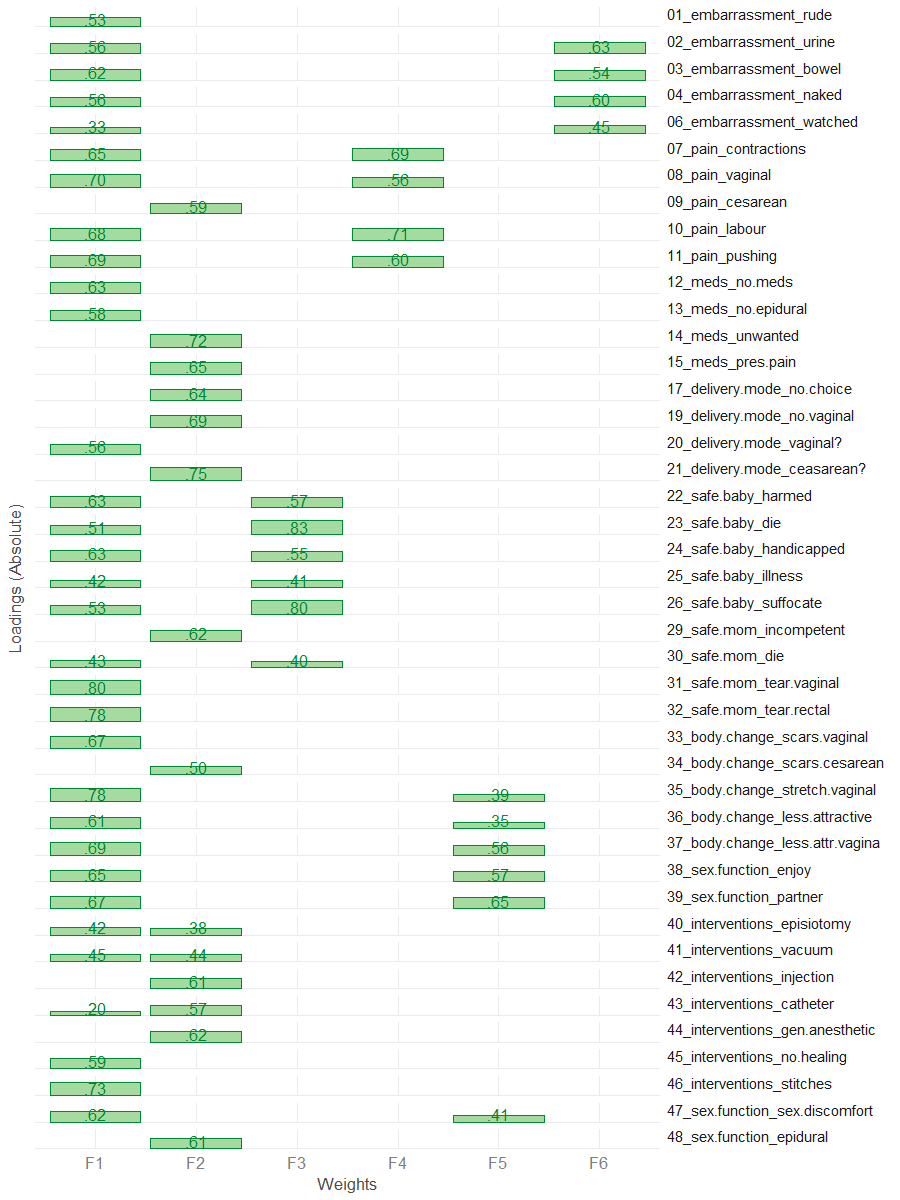
## Confirm

Applying "Exploratory-Confirmatory" procedure described by [Joreskog(1978)](https://scholar.google.ca/scholar?q=Structural+analysis+of+covariance+and+correlation+matrices&btnG=&hl=en&as_sdt=0%2C33), we find the largest loading for each column of the factor pattern, then constrain all the other loadings in that row to be zero, and fit the resulting model as a confirmatory factor model. Given that we chose the orthogonal bifactor solution, we permit the the cross-loadings between general factor and subfactors.

NOTE: we chose to fix the loading of item 6 on subscale 2 to 0 because it was borderline trivial (.31), and because it had substantial loadings on the subscale 6. We decided not to eliminate it from the scale because it had a strong conceptual fit to subscale 6 and contributed to the interpretability of the overall scale.

# These values are translated into CFA model and used as starting values  
model\_5 <- FAtoSEM(  
 x = fit\_efa\_5[["Bifactor"]] ,  
 cutoff = 0.315,  
 factor.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame"),  
 make.start.values = TRUE,  
 cov.matrix = FALSE, # TRUE - oblique, FALSE - orthogonal  
 num.digits = 4  
)

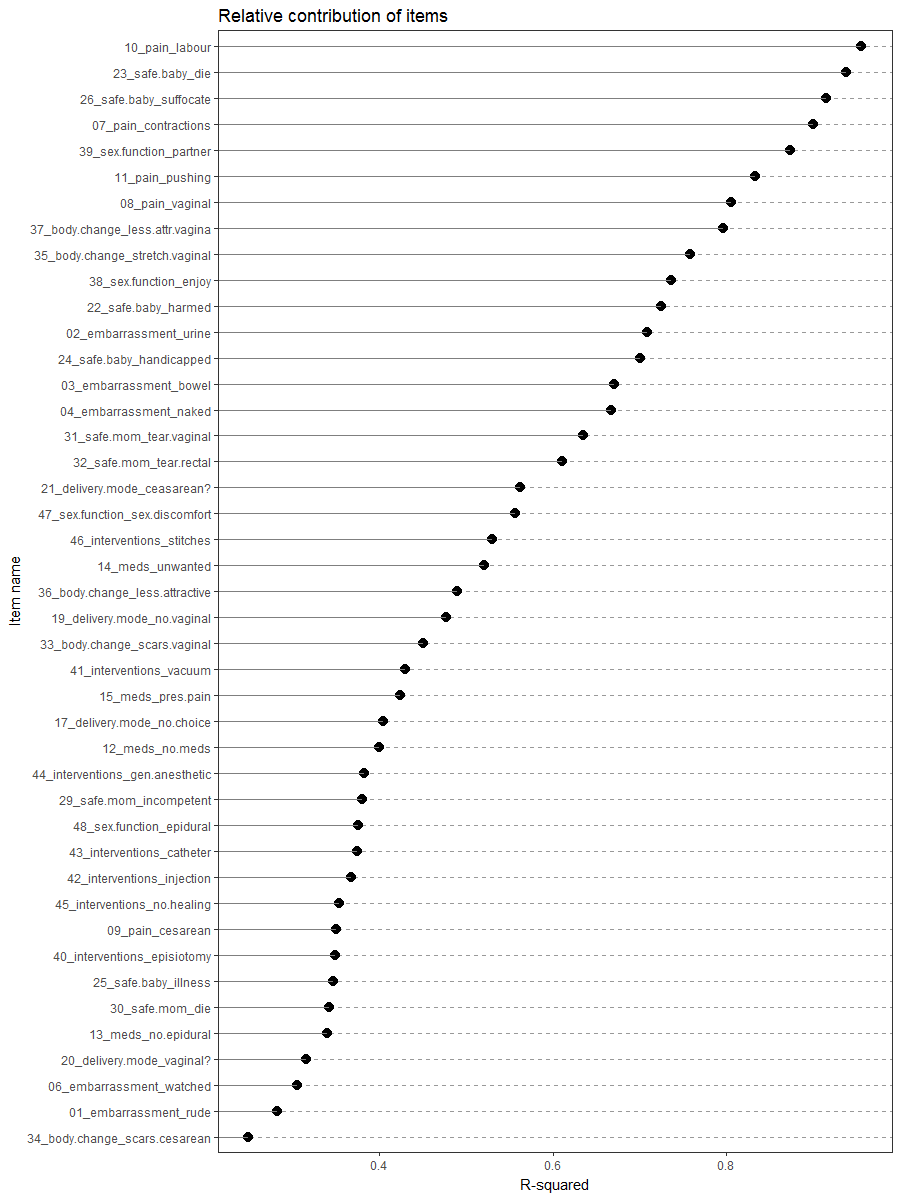
# the model is estimated using sem package  
fit\_5 <- sem::sem(model\_5,R5,sample\_size)  
# the pattern of the solution  
m <- GetPattern(fit\_5)$F  
m[m==0] <- NA  
m %>% plot\_factor\_pattern(factor\_width=6)



# Summary of the fitted model  
sem\_model\_summary(fit\_5)

Model Chiquare = 4985.451 | df model = 837 | df null = 903  
Goodness-of-fit index = 0.7175682  
Adjusted Goodness-of-fit index = 0.680788  
RMSEA index = .0879 90% CI: (.086,.090)  
Comparitive Fit Index (CFI = 0.8080764  
Tucker Lewis Index (TLI/NNFI) = 0.7929427  
Akaike Information Criterion (AIC) = 5203.451  
Bayesian Information Criterion (BIC) = -426.7118

#Relative contribudion of items   
sort(summary(fit\_5)$Rsq) %>% dot\_plot()



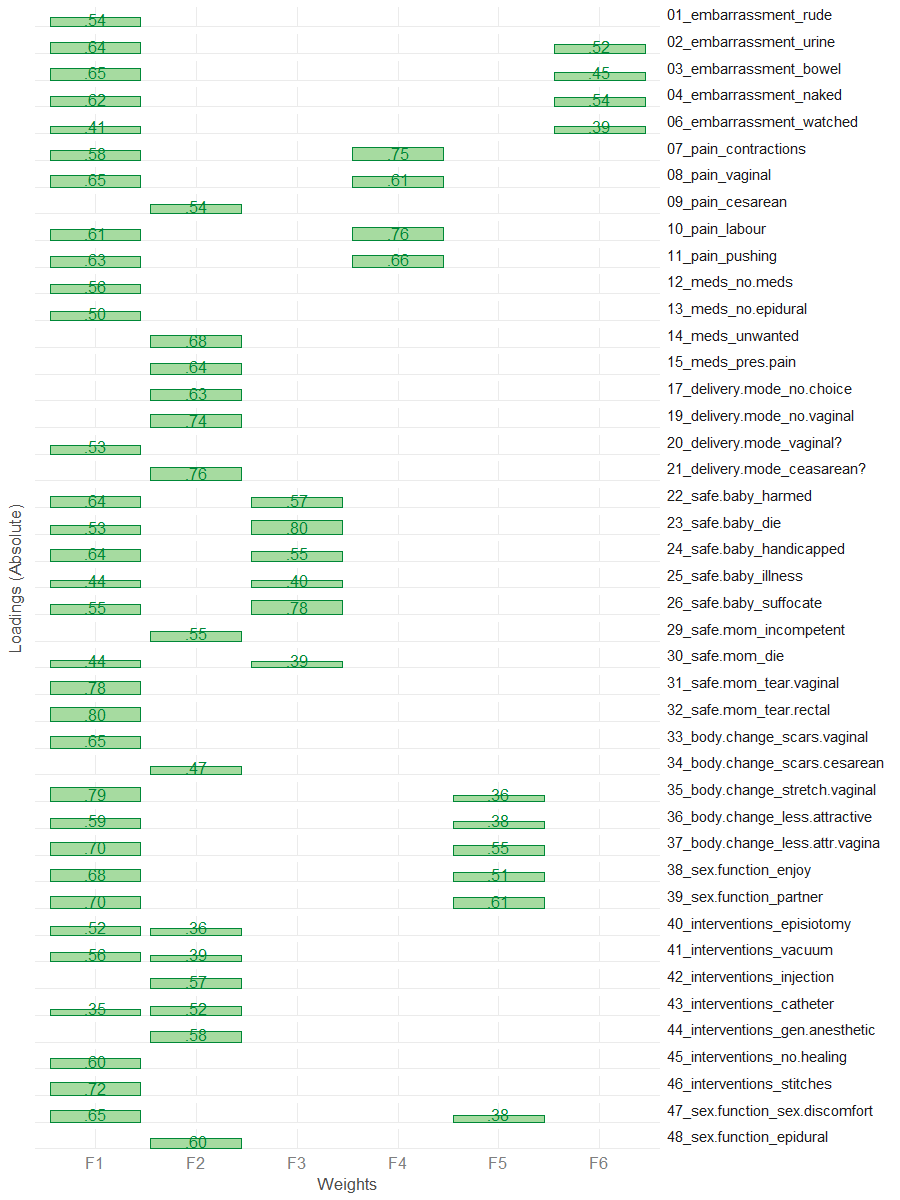
# Conclusion

The remaining items do not violate the consideraton we have [outlined](#elimination).

One contention item is item 6 which consistently exhibits crossloading between subscale 2 and subscale 6. Given that one of the loadings is borderline trivial (.31) and that item 6 has a good conceptual fit to subscale 6, we chose to fix the loadings of item 6 on subscale 2 to 0, instead of eliminating it from the scale.

Thus the remaing simple structure the we recommend as the final solution is as follow:

f\_pattern <- fit\_efa\_5[['Bifactor']]$F   
f\_pattern[f\_pattern<.315] <- NA  
f\_pattern %>% plot\_factor\_pattern(factor\_width = 6)



# in tabular form with trival loading masked  
f\_pattern %>%   
 knitr::kable(  
 format = "pandoc",  
 col.names = c("General","Interventions","Safety","Pain","Sex & Body","Shame")  
 ) %>%   
 print()

General Interventions Safety Pain Sex & Body Shame  
-------------------------------- ---------- -------------- ---------- ---------- ----------- ----------  
01\_embarrassment\_rude 0.5423293 NA NA NA NA NA  
02\_embarrassment\_urine 0.6360395 NA NA NA NA 0.5191324  
03\_embarrassment\_bowel 0.6524797 NA NA NA NA 0.4453770  
04\_embarrassment\_naked 0.6197220 NA NA NA NA 0.5421293  
06\_embarrassment\_watched 0.4102881 NA NA NA NA 0.3885423  
07\_pain\_contractions 0.5792023 NA NA 0.7472456 NA NA  
08\_pain\_vaginal 0.6510845 NA NA 0.6142759 NA NA  
09\_pain\_cesarean NA 0.5399707 NA NA NA NA  
10\_pain\_labour 0.6059648 NA NA 0.7589788 NA NA  
11\_pain\_pushing 0.6304261 NA NA 0.6557546 NA NA  
12\_meds\_no.meds 0.5559435 NA NA NA NA NA  
13\_meds\_no.epidural 0.5044178 NA NA NA NA NA  
14\_meds\_unwanted NA 0.6820275 NA NA NA NA  
15\_meds\_pres.pain NA 0.6355915 NA NA NA NA  
17\_delivery.mode\_no.choice NA 0.6270099 NA NA NA NA  
19\_delivery.mode\_no.vaginal NA 0.7357642 NA NA NA NA  
20\_delivery.mode\_vaginal? 0.5310804 NA NA NA NA NA  
21\_delivery.mode\_ceasarean? NA 0.7644856 NA NA NA NA  
22\_safe.baby\_harmed 0.6426486 NA 0.5674377 NA NA NA  
23\_safe.baby\_die 0.5285091 NA 0.8038722 NA NA NA  
24\_safe.baby\_handicapped 0.6397053 NA 0.5516317 NA NA NA  
25\_safe.baby\_illness 0.4386078 NA 0.3980388 NA NA NA  
26\_safe.baby\_suffocate 0.5508336 NA 0.7802002 NA NA NA  
29\_safe.mom\_incompetent NA 0.5475151 NA NA NA NA  
30\_safe.mom\_die 0.4403417 NA 0.3855358 NA NA NA  
31\_safe.mom\_tear.vaginal 0.7821562 NA NA NA NA NA  
32\_safe.mom\_tear.rectal 0.8018144 NA NA NA NA NA  
33\_body.change\_scars.vaginal 0.6538049 NA NA NA NA NA  
34\_body.change\_scars.cesarean NA 0.4706662 NA NA NA NA  
35\_body.change\_stretch.vaginal 0.7877252 NA NA NA 0.3639645 NA  
36\_body.change\_less.attractive 0.5902975 NA NA NA 0.3754950 NA  
37\_body.change\_less.attr.vagina 0.7047147 NA NA NA 0.5502074 NA  
38\_sex.function\_enjoy 0.6845825 NA NA NA 0.5136751 NA  
39\_sex.function\_partner 0.6977893 NA NA NA 0.6099491 NA  
40\_interventions\_episiotomy 0.5190806 0.3561446 NA NA NA NA  
41\_interventions\_vacuum 0.5623879 0.3907158 NA NA NA NA  
42\_interventions\_injection NA 0.5678000 NA NA NA NA  
43\_interventions\_catheter 0.3476105 0.5249393 NA NA NA NA  
44\_interventions\_gen.anesthetic NA 0.5834468 NA NA NA NA  
45\_interventions\_no.healing 0.5974314 NA NA NA NA NA  
46\_interventions\_stitches 0.7191906 NA NA NA NA NA  
47\_sex.function\_sex.discomfort 0.6514208 NA NA NA 0.3793385 NA  
48\_sex.function\_epidural NA 0.5965353 NA NA NA NA

# in tabular form with trival loading not masked  
fit\_efa\_5[['Bifactor']]$F %>%   
 knitr::kable()

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Factor1 | Factor2 | Factor3 | Factor4 | Factor5 | Factor6 |
| 01\_embarrassment\_rude | 0.5423293 | 0.0354241 | -0.1244893 | 0.0804393 | 0.0929842 | 0.2630145 |
| 02\_embarrassment\_urine | 0.6360395 | 0.0917088 | -0.0138083 | -0.0077248 | 0.0161971 | 0.5191324 |
| 03\_embarrassment\_bowel | 0.6524797 | -0.0111211 | 0.0215541 | 0.0944636 | 0.0957167 | 0.4453770 |
| 04\_embarrassment\_naked | 0.6197220 | -0.0270479 | -0.0159253 | 0.0594578 | 0.0063370 | 0.5421293 |
| 06\_embarrassment\_watched | 0.4102881 | 0.3109870 | -0.0564050 | -0.1140105 | -0.0767501 | 0.3885423 |
| 07\_pain\_contractions | 0.5792023 | -0.0717080 | -0.0033733 | 0.7472456 | 0.0258661 | 0.0254417 |
| 08\_pain\_vaginal | 0.6510845 | -0.1050601 | -0.0358711 | 0.6142759 | -0.0258560 | -0.0030035 |
| 09\_pain\_cesarean | 0.2867437 | 0.5399707 | 0.0667470 | 0.0436433 | -0.0470763 | -0.0070985 |
| 10\_pain\_labour | 0.6059648 | -0.0799260 | -0.0196977 | 0.7589788 | 0.0171547 | 0.0023748 |
| 11\_pain\_pushing | 0.6304261 | -0.1071267 | -0.0218411 | 0.6557546 | -0.0148246 | 0.0102663 |
| 12\_meds\_no.meds | 0.5559435 | -0.2135852 | 0.1508774 | 0.2763698 | 0.0846058 | -0.0258050 |
| 13\_meds\_no.epidural | 0.5044178 | -0.2694990 | 0.1423879 | 0.2556404 | 0.0735433 | -0.0571140 |
| 14\_meds\_unwanted | 0.1865154 | 0.6820275 | 0.0409606 | -0.2032958 | -0.0486401 | 0.0661205 |
| 15\_meds\_pres.pain | 0.1117344 | 0.6355915 | 0.0353919 | -0.0984584 | -0.0543572 | 0.0849449 |
| 17\_delivery.mode\_no.choice | 0.1766761 | 0.6270099 | 0.1474785 | 0.0101468 | 0.1055101 | -0.0853619 |
| 19\_delivery.mode\_no.vaginal | 0.0104164 | 0.7357642 | 0.1033089 | 0.0049232 | 0.1077822 | -0.0868324 |
| 20\_delivery.mode\_vaginal? | 0.5310804 | -0.1719837 | -0.0357867 | 0.2998018 | -0.1106821 | -0.0194421 |
| 21\_delivery.mode\_ceasarean? | 0.1231446 | 0.7644856 | 0.0381583 | 0.0624416 | 0.0551232 | -0.0310248 |
| 22\_safe.baby\_harmed | 0.6426486 | 0.0360814 | 0.5674377 | -0.0366921 | -0.0069491 | -0.1245322 |
| 23\_safe.baby\_die | 0.5285091 | 0.0663237 | 0.8038722 | 0.0010820 | -0.0152627 | 0.0389358 |
| 24\_safe.baby\_handicapped | 0.6397053 | 0.0288145 | 0.5516317 | -0.0475646 | 0.0293705 | -0.1254217 |
| 25\_safe.baby\_illness | 0.4386078 | 0.0218463 | 0.3980388 | -0.0617734 | 0.0407836 | -0.0429735 |
| 26\_safe.baby\_suffocate | 0.5508336 | 0.0169084 | 0.7802002 | -0.0085089 | -0.0181696 | 0.0261672 |
| 29\_safe.mom\_incompetent | 0.2680192 | 0.5475151 | 0.1417759 | -0.1893575 | -0.0593872 | 0.0823268 |
| 30\_safe.mom\_die | 0.4403417 | 0.1228104 | 0.3855358 | -0.0085270 | 0.0599991 | 0.0918993 |
| 31\_safe.mom\_tear.vaginal | 0.7821562 | -0.0003814 | -0.0460863 | 0.1006444 | -0.0667876 | -0.2022514 |
| 32\_safe.mom\_tear.rectal | 0.8018144 | 0.1113117 | 0.0501259 | -0.0194241 | -0.1179708 | -0.1302555 |
| 33\_body.change\_scars.vaginal | 0.6538049 | 0.0029122 | -0.0732032 | 0.0257085 | 0.0807509 | -0.1964019 |
| 34\_body.change\_scars.cesarean | 0.2191653 | 0.4706662 | -0.0787972 | -0.0711825 | 0.0974594 | -0.0890362 |
| 35\_body.change\_stretch.vaginal | 0.7877252 | -0.0219142 | -0.0302175 | 0.0644025 | 0.3639645 | -0.0684696 |
| 36\_body.change\_less.attractive | 0.5902975 | -0.0094281 | -0.0506879 | 0.1176374 | 0.3754950 | 0.0246573 |
| 37\_body.change\_less.attr.vagina | 0.7047147 | -0.0111246 | -0.0445706 | 0.0477964 | 0.5502074 | 0.0580497 |
| 38\_sex.function\_enjoy | 0.6845825 | -0.0425303 | 0.0363985 | -0.0591871 | 0.5136751 | -0.0262829 |
| 39\_sex.function\_partner | 0.6977893 | 0.0039927 | -0.0129411 | 0.0169166 | 0.6099491 | 0.0419385 |
| 40\_interventions\_episiotomy | 0.5190806 | 0.3561446 | -0.1177258 | 0.0079074 | -0.1407206 | -0.1467145 |
| 41\_interventions\_vacuum | 0.5623879 | 0.3907158 | 0.0512029 | -0.0810893 | -0.1149485 | -0.1769570 |
| 42\_interventions\_injection | 0.1577588 | 0.5678000 | -0.0820984 | -0.0643750 | -0.0993342 | 0.1516477 |
| 43\_interventions\_catheter | 0.3476105 | 0.5249393 | -0.0712807 | -0.0419917 | -0.0430658 | 0.1019760 |
| 44\_interventions\_gen.anesthetic | 0.1785587 | 0.5834468 | -0.0952341 | -0.0791251 | -0.0425807 | -0.0278484 |
| 45\_interventions\_no.healing | 0.5974314 | 0.2125631 | 0.0761573 | -0.0375136 | 0.0715903 | -0.0119218 |
| 46\_interventions\_stitches | 0.7191906 | 0.1324744 | -0.0565872 | 0.0766869 | 0.0088176 | -0.0909816 |
| 47\_sex.function\_sex.discomfort | 0.6514208 | 0.0082455 | 0.0544070 | -0.0651988 | 0.3793385 | -0.0953781 |
| 48\_sex.function\_epidural | 0.1195484 | 0.5965353 | -0.1392454 | -0.0095462 | -0.0772314 | 0.0745112 |

# Reproducibility

sessionInfo()

R version 3.3.1 (2016-06-21)  
Platform: x86\_64-w64-mingw32/x64 (64-bit)  
Running under: Windows >= 8 x64 (build 9200)  
  
locale:  
[1] LC\_COLLATE=English\_United States.1252 LC\_CTYPE=English\_United States.1252 LC\_MONETARY=English\_United States.1252  
[4] LC\_NUMERIC=C LC\_TIME=English\_United States.1252   
  
attached base packages:  
[1] stats graphics grDevices utils datasets methods base   
  
other attached packages:  
[1] knitr\_1.14 plotrix\_3.6-3 GPArotation\_2014.11-1 sem\_3.1-8 ggplot2\_2.2.0   
[6] psych\_1.6.9 magrittr\_1.5   
  
loaded via a namespace (and not attached):  
 [1] reshape2\_1.4.1 splines\_3.3.1 lattice\_0.20-34 colorspace\_1.2-7 htmltools\_0.3.5 stats4\_3.3.1   
 [7] yaml\_2.1.13 nloptr\_1.0.4 foreign\_0.8-67 DBI\_0.5-1 RColorBrewer\_1.1-2 readxl\_0.1.1   
[13] plyr\_1.8.4 stringr\_1.1.0 munsell\_0.4.3 gtable\_0.2.0 coda\_0.18-1 evaluate\_0.10   
[19] labeling\_0.3 mi\_1.0 extrafont\_0.17 parallel\_3.3.1 highr\_0.6 Rttf2pt1\_1.3.4   
[25] Rcpp\_0.12.7 readr\_1.0.0 formatR\_1.4 scales\_0.4.1 arm\_1.9-1 abind\_1.4-5   
[31] lme4\_1.1-12 testit\_0.5 mnormt\_1.5-5 digest\_0.6.10 stringi\_1.1.2 dplyr\_0.5.0   
[37] grid\_3.3.1 tools\_3.3.1 lazyeval\_0.2.0 tibble\_1.2 dichromat\_2.0-0 tidyr\_0.6.0   
[43] extrafontdb\_1.0 MASS\_7.3-45 Matrix\_1.2-7.1 rsconnect\_0.5 matrixcalc\_1.0-3 assertthat\_0.1   
[49] minqa\_1.2.4 rmarkdown\_1.1 R6\_2.2.0 boot\_1.3-18 nlme\_3.1-128