Assignment 2: Task 1 - CFA

Group 11

30.10.2018

0. Load Libraries

```
library(tidyverse)
library(lavaan)
library(semPlot)
library(lessR)
```

1. Problem Statement

We are given a covariance matrix and other descriptive statistics (mean, standard deviation, number of observations) of 8 variables that are assumed to measure abilities of professional basketball players. The task is to check whether a hypothesized model with two latent factors called "frontcourt skills" and "backcourt skills" fits the variance/covariance structure of the given data well, compare this model to another with three latent factors and finally perform a test for equality of two parameters. The goal is to find shared structures in the variances of the variables, i.e. factors that have an influence on groups of variables and can therefore be determined underlying unobservable factors that help explaining the variance/covariance structure of the data.

2. Descriptive Statistics

With only the variance matrix there is no possibility of checking for outliers in the data. One can see from looking at the complete table that the means of X3 and X8 and are substantially higher than the rest. This is not important for the analysis and can be safely ignored.

```
basketdf <- read.table("http://feb.kuleuven.be/martina.vandebroek/public/STATdata/basketball.txt",
                       header=T)
basket_cov <- basketdf %>% filter(X_TYPE_ == "COV") %>% select(2:ncol(basketdf))
rownames(basket_cov) <- colnames(basket_cov)
basket_cov <- as.matrix(basket_cov)
basketdf %>% filter(X_TYPE_ != "COV")
```

```
##   X_TYPE_          X1         X2          X3          X4          X5
## 1    MEAN   0.313544   0.757588   3.354687   1.137500   0.508394
## 2     STD   0.126765   0.101748   2.051235   0.420963   0.063846
## 3       N 320.000000 320.000000 320.000000 320.000000 320.000000
##            X6         X7         X8
## 1   0.755000   1.495000   5.056875
## 2   0.655414   1.178294   2.068820
## 3 320.000000 320.000000 320.000000
```

3. Assumptions

The most important condition to obtain meaningful results is that the number of inputs (unique values in variance/covariance matrix) is higher than the number of estimated parameters. Here we have 8 variables so our number of inputs is (8*(8+1))/2=36. For the first model we estimate 8 loadings, 8 error variances of the variables and 1 covariance between factors which lead to a total of 17 estimated parameters. The model is therefore well identified (as is the second model where 19 parameters are estimated). We confirm this with the inspect function where nonzero integers in the output are parameters that are to be estimated. A (approximate) multivariate normal distribution of the data also has to be assumed in order to estimate the paramters with maximum likelihood.

```
# Specify two-factor model
model1 <- '
# latent variables
backcourt =~ X1 + X2 + X3 + X4
frontcourt =~ X5 + X6 + X7 + X8
'
fit1 <- lavaan::sem(model1, sample.nobs = 320, sample.cov = basket_cov, orthogonal = F)

# Show number of estimated parameters
inspect(fit1)$psi
```

```
##            bckcrt frntcr
## backcourt  15
## frontcourt 17     16
```

4. Method and interpretation

Part A:

The model was already fit in section 3 with the sem function from the lavaan package. To assess the model fit we constructed a function that computes the goodness-of-fit (GFI) because this indicator is not given in the output of lavaan.

```
# Specify GFI function
GFI <- function(Si, Sobs){
  if( class(Si) == "list"){
    Si <- as.matrix(as.data.frame(Si$cov))
  }

  nominator <- sum(diag(solve(Si) %*% Sobs - diag(ncol(Sobs))))^2
  denominator <- sum(diag(solve(Si) %*% Sobs))^2
  return(1 - (nominator / denominator))
}
```

By using the summary (see output in Appendix), resid and modindices functions of the lavaan package the model are inspected. Overall the model does not fit the data well, for example the null hypothesis that the estimated covariance matrix is equal to the observed covariance matrix is rejected with a p-value of 0.

Overall the model does not fit the data well, the null hypothesis that the estimated covariance matrix is equal to the observed covariance matrix is rejected with a p-value of 0 and the comparative fit index (CFI) which compares the model with a baseline null model is 0.892. This value indicates that the model, while being better than the baseline model, can not be considered a good enough improvement to the baseline model. The GFI on the other hand is equal to 0.9999902. This tells us that the calculated covariance matrix (Si) is very similar to the observed one (Sobs). Finally, all estimated parameters are significant (the output of

the summary function was deprecated, so that it does not show the parameter estimates because we do not interpret them in any way, the estimates were, however, checked and turned out to be significant).

```
# Display standardized residuals
resid(fit1, type = "standardized")
```

```
## $type
## [1] "standardized"
##
## $cov
##    X1     X2     X3     X4     X5     X6     X7     X8
## X1  0.000
## X2  3.282  0.000
## X3 -3.381 -1.090  0.000
## X4 -2.510 -1.897  6.089  0.000
## X5 -0.124  0.643 -0.648  0.859  0.000
## X6  0.583 -0.638 -1.506  1.465  2.924  0.000
## X7 -1.864  1.439 -2.563 -1.476 -1.325 -5.744  0.000
## X8  3.842  2.795  1.109 -0.764 -0.526  3.243  1.592  0.000
```

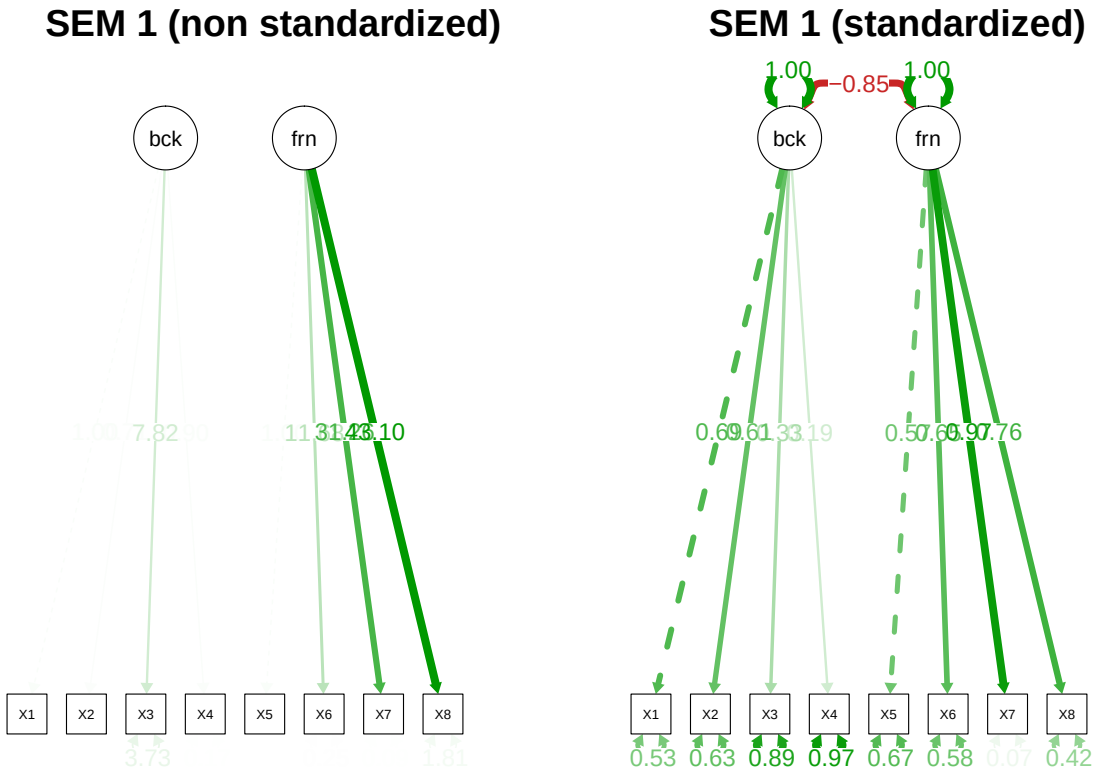
The resid function gives us the matrix of the standardized residuals. As a heuristic rule, standardized residuals over 1.96 (absolute value) are considered bad and indicate a bad fit. This is the case for the following 9 covariances: (X4,X3), (X7,X6), (X8,X1), (X3,X1), (X2,X1), (X8,X6), (X6,X5), (X8,X2) and (X7,X3).

```
# Show modification indices greater than 10
modindices(fit1, sort. = T, minimum.value = 10)
```

```
##          lhs op rhs     mi     epc sepc.lv sepc.all sepc.nox
## 41        X3 ~~  X4 43.828   0.302   0.302    0.378    0.378
## 53        X6 ~~  X7 23.085  -0.136  -0.136   -0.898   -0.898
## 23 backcourt =~  X8 22.190  17.306   1.510    0.731    0.731
## 22 backcourt =~  X7 18.418 -10.325  -0.901   -0.766   -0.766
## 28        X1 ~~  X2 16.622   0.004   0.004    0.506    0.506
## 54        X6 ~~  X8 11.856   0.154   0.154    0.230    0.230
## 34        X1 ~~  X8 10.642   0.028   0.028    0.223    0.223
## 29        X1 ~~  X3 10.552  -0.040  -0.040   -0.223   -0.223
```

The modindices function outputs the modification indices that tell us the approximate decrease in χ^2 if that parameter would be estimated, mi is the decrease in the χ^2 statistic (LM Stat) in SAS) and epc is the expected change of the estimated parameter (Parm Change in SAS). Here, we see that for example the estimation of the covariance between X3 and X4 (or equivalenty lifting the constraint that this covariance is 0) would result in a decrease of the χ^2 statistic of approximately 43.828.

```
# Plot of standardized and unstandardized loading estimates
par(mfrow=c(1,2))
semPaths(fit1, "est",edge.label.cex = 1.4, curvePivot = TRUE)
title("SEM 1 (non standardized)", line = 3)
semPaths(fit1, "std",edge.label.cex = 1.4, curvePivot = TRUE)
title("SEM 1 (standardized)", line = 3)
```




```
par(mfrow=c(1,1))
```

Here, two path plots are shown to illustrate the underlying model specification and the estimated parameters. The left one uses non standardized parameters, which is why most of the paths are hardly visible because the parameters are low. In order to be able to see all paths, also the standarrdized plot, where the problem of paths fading away is greatly reduced, is shown.

Part B:

The model is respecified by splitting the backcourt factor into 2 factors: “shooting skills” and “neuromuscular coordination” while the frontcourt factor is equivalent to the first model. See again the output values of the summary function in the Appendix.

```
# Specify three factor model
model2 <- '
# latent variables
shoot =~ X1 + X2
neuro =~ X3 + X4
frontcourt =~ X5 + X6 + X7 + X8
'
# Fitting the model
fit2 <- lavaan::sem(model2, sample.nobs = 320, sample.cov = basket_cov, orthogonal = F)
```

The p-value still indicates a good fit, but the CFI improved a lot in comparison to the first model (0.975 instead of 0.892), all other indicators are not significantly different to the first model (p-value still bad, GFI still good, parameters still significant). Because these indicators do not allow us to definitively say one model is better than another, both the AIC and BIC, which include a penalty for model complexity, are compared between the models. The second model has lower values on both of them (AIC: 2100.703 $<$ 2156.084, BIC:

2172.301 < 2220.145), so one can safely assume that the second model fits the data better.

```
resid(fit2, type = "standardized")
```

```
## $type
## [1] "standardized"
##
## $cov
##    X1     X2     X3     X4     X5     X6     X7     X8
## X1  0.000
## X2  0.000  0.000
## X3 -0.553  1.247  0.000
## X4 -0.902 -0.666  0.000  0.000
## X5 -0.131  0.316 -0.154  1.266  0.000
## X6  0.566 -1.047 -0.949  1.981  2.952  0.000
## X7 -2.386 -1.501 -1.286  0.092 -1.439 -5.924  0.000
## X8  3.867  2.249  2.218 -0.234 -0.474  3.280  1.740  0.000
```

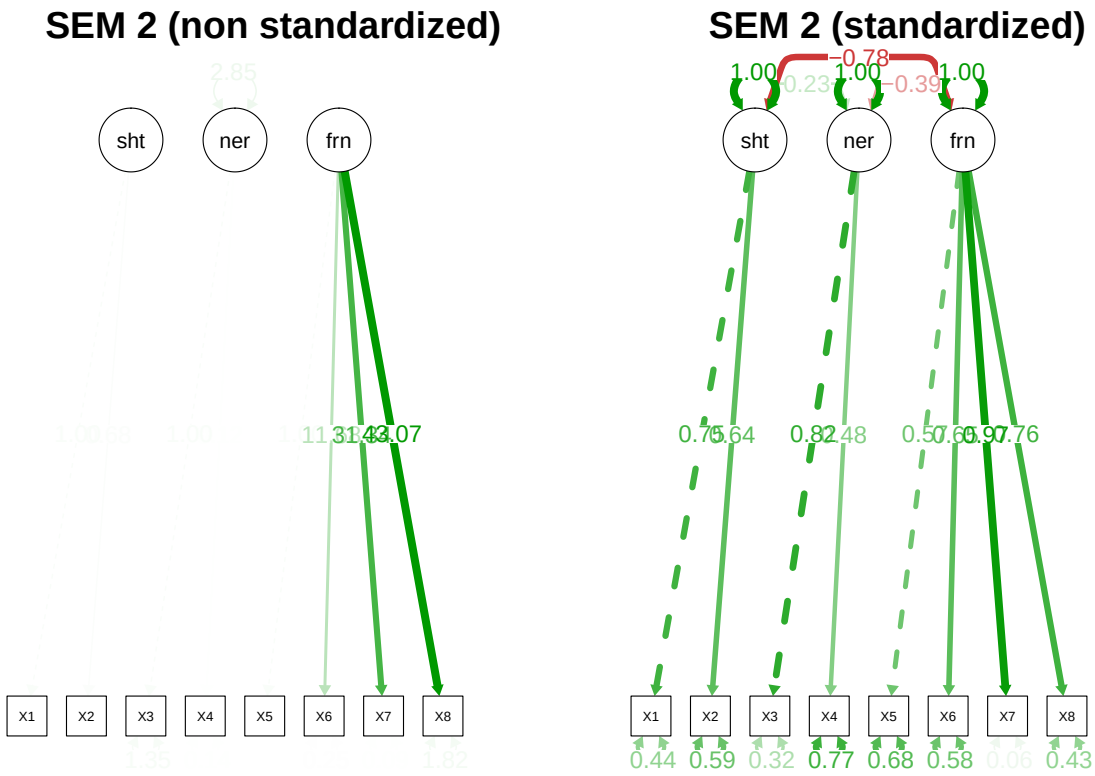
There are now less standardized residuals with a n absolute value over 1.96 (7 instead of 9) and also the values are smaller than those for the first model. From this one can conclude that the second model is an improvement to the first one.

```
# Show modification indices greater than 10
modindices(fit2, sort. = T, minimum.value = 10)
```

```
##      lhs op rhs     mi    epc sepc.lv sepc.all sepc.nox
## 64    X6 ~~  X7 24.343 -0.139  -0.139   -0.936   -0.936
## 28 shoot =~  X8 14.341  8.533   0.812    0.393    0.393
## 65    X6 ~~  X8 12.153  0.156   0.156    0.232    0.232
## 45    X1 ~~  X8 10.823  0.027   0.027    0.242    0.242
## 27 shoot =~  X7 10.078 -4.389  -0.417   -0.355   -0.355
```

The mi values are now lower, so it is more difficult than in the first case to improve the model further.

```
# Plot of standardized and unstandardized loading estimates
par(mfrow=c(1,2))
semPaths(fit2, "est",edge.label.cex = 1.4, curvePivot = TRUE)
title("SEM 2 (non standardized)", line = 3)
semPaths(fit2, "std",edge.label.cex = 1.4, curvePivot = TRUE)
title("SEM 2 (standardized)", line = 3)
```




```
par(mfrow=c(1,1))
```

The model could probably be improved by introducing new factor for X6 and X7 and/or a loading of the factor shooting skills on X8 (these two have the highest mi value in the modindices output).

Part C:

In order to be able to test whether the loadings of X7 and x8 are the same, a two sided t-test using the standardized loadings needs to be applied. The null hypothesis is that the loadings are equal, whereas the alternative hypothesis is that both values are not equal. The significance level is set to five percent and the remaining degrees of freedom are 301, as 19 variables were estimated.

```
# Extract standardized estimates and standard errors
estX7X8 <- standardizedSolution(fit2)$est.std[7:8]
seX7X8 <- standardizedSolution(fit2)$se[7:8]

# Set significance level
alpha <- 0.05

# Calculate t-statistic
tStat <- ((estX7X8[1] - estX7X8[2]) / seX7X8[1])

# Decide whether to reject H0 or not
# TRUE: Reject H0
# FALSE: Accept H0
ifelse(tStat < 0, tStat < -qt(p=1-(alpha/2), df=320-19),
                  tStat > qt(p=1-(alpha/2), df=320-19))
```

```
## [1] TRUE
```

The resulting test statistic is equal to 13.34606, which is much larger then the threshold value of approximately 1.94. Therefore the null hypothesis can be rejected, meaning that the loadings are significantly different from each other.

5. Alternative solutions

We want to test another two factor model, which consists offensive and defensive attributes. Therefore, X1, X2, X3 and X5 are measures of offensive attributes and the remaining observed variables are measures of the defensive abilities.

```
modelAlt <- '
# latent variables
offensive =~ X1 + X2 + X3 + X5
defensive =~ X4 + X6 + X7 + X8
'

# Fitting the model
fit3 <- lavaan::sem(modelAlt, sample.nobs = 320, sample.cov = basket_cov, orthogonal = F)
```

Analysing the fit criteria (see output in Appendix) results in the model being a bad one. The null hypothesis that the estimated covariance matrix is equal to the observed covariance matrix is rejected. The CFI value of 0.885 is lower than the desired value of 0.95 and the GFI value of approximately 0.2136 is far below the threshold value aswell. The AIC and BIC values are also higher than the values from the other two values.

```
# Show modification indices greater than 10
modindices(fit3, sort. = T, minimum.value = 10)
```

```
##          lhs op rhs     mi     epc sepc.lv sepc.all sepc.nox
## 42        X3 ~~  X4 43.595   0.298   0.298    0.372    0.372
## 23 offensive =~  X8 24.623  45.228   3.633    1.759    1.759
## 53        X6 ~~  X7 24.582  -0.146  -0.146   -1.043   -1.043
## 28        X1 ~~  X2 19.814   0.003   0.003    0.309    0.309
## 54        X6 ~~  X8 13.529   0.167   0.167    0.246    0.246
## 34        X1 ~~  X8 12.337   0.030   0.030    0.230    0.230
## 22 offensive =~  X7 11.219 -24.551  -1.972   -1.676   -1.676
```

One could improve this model by creating a new factor containing X3 and X4, which would result in the χ^2 statistic being improved by 43.828.

6. Conclusion

The main purpose of CFA is to identify a certain factor model, which fits the data well based on several criteria, as mentioned above. After respecifying the first factor model, we came out with a better solution in terms of model fit. Nonetheless, this model could still not be completely validated, as the null hypothesis of the χ^2 test is rejected. The proposed alternative model turns out to be worse than all the other models.

Appendix

```
# Output of model 1 fit measures
summary(fit1,fit.measures = TRUE, estimates = F)
```

```
## lavaan 0.6-3 ended normally after 88 iterations
##
##   Optimization method                           NLMINB
##   Number of free parameters                         17
##
##   Number of observations                           320
##
##   Estimator                                         ML
##   Model Fit Test Statistic                     113.897
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              909.437
##   Degrees of freedom                                28
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.892
##   Tucker-Lewis Index (TLI)                       0.841
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1061.042
##   Loglikelihood unrestricted model (H1)      -1004.093
##
##   Number of free parameters                         17
##   Akaike (AIC)                                2156.084
##   Bayesian (BIC)                              2220.145
##   Sample-size adjusted Bayesian (BIC)         2166.224
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.125
##   90 Percent Confidence Interval          0.103  0.148
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.073
```

```
GFI(Si = fitted(fit1), Sobs = basket_cov)
```

```
## [1] 0.9999902
```

```r
# Output of model 2 fit measures
summary(fit2,fit.measures = TRUE, estimates = F)
```

```
## lavaan 0.6-3 ended normally after 96 iterations
##
##   Optimization method                           NLMINB
##   Number of free parameters                         19
##
##   Number of observations                           320
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      54.516
##   Degrees of freedom                                17
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              909.437
##   Degrees of freedom                                28
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.957
##   Tucker-Lewis Index (TLI)                       0.930
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1031.351
##   Loglikelihood unrestricted model (H1)      -1004.093
##
##   Number of free parameters                         19
##   Akaike (AIC)                                2100.703
##   Bayesian (BIC)                              2172.301
##   Sample-size adjusted Bayesian (BIC)         2112.036
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.083
##   90 Percent Confidence Interval          0.059  0.108
##   P-value RMSEA <= 0.05                          0.014
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.038
```

```r
GFI(Si = fitted(fit2), Sobs = basket_cov)
```

```
## [1] 0.9999902
```

```r
# Output of model 3 fit measures
summary(fit3,fit.measures = TRUE, estimates = F)
```

```
## lavaan 0.6-3 ended normally after 90 iterations
##
##   Optimization method                           NLMINB
##   Number of free parameters                         17
##
##   Number of observations                           320
##
##   Estimator                                         ML
##   Model Fit Test Statistic                     120.647
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              909.437
##   Degrees of freedom                                28
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.885
##   Tucker-Lewis Index (TLI)                       0.830
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1064.417
##   Loglikelihood unrestricted model (H1)      -1004.093
##
##   Number of free parameters                         17
##   Akaike (AIC)                                2162.834
##   Bayesian (BIC)                              2226.895
##   Sample-size adjusted Bayesian (BIC)         2172.974
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.129
##   90 Percent Confidence Interval          0.108  0.152
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.075
```

```r
GFI(Si = fitted(fit3), Sobs = basket_cov)
```

```
## [1] 0.2136608
```