Confirmatory Factor Analysis

Isabella Chittumuri

11/8/2020

13.9 Carry out a factor analysis of the rootstock data of Table 6.2. Combine the six groups into a single sample.

```
getwd()
```

```
## [1] "/Users/isabellachittumuri/Documents/Hunter College/Fall 2020/Stat 717/HW"
```

```
df <- read.table("T6_2_ROOT.DAT")
root <- df[ -c(1) ]
```

(a) Estimate the loadings for two factors by the principal component method and do a varimax rotation.

```
# 1 PC method
Rmat <- cor(root)
(e <- eigen(Rmat))
```

```
## eigen() decomposition
## $values
## [1] 2.78462702 1.05412174 0.11733950 0.04391174
##
## $vectors
##           [,1]       [,2]       [,3]       [,4]
## [1,] 0.4713465  0.5600120  0.6431731  0.2248274
## [2,] 0.5089667  0.4544775 -0.7142114 -0.1559013
## [3,] 0.5243109 -0.4431448  0.2413716 -0.6859012
## [4,] 0.4938456 -0.5324091 -0.1340527  0.6743048
```

```
# Proportion of var explained
pca <- princomp(covmat=Rmat)
(s <- summary(pca, loadings = TRUE))
```

```
## Importance of components:
##                           Comp.1    Comp.2     Comp.3     Comp.4
## Standard deviation     1.6687202 1.0267043 0.34254854 0.20955128
## Proportion of Variance 0.6961568 0.2635304 0.02933488 0.01097793
## Cumulative Proportion  0.6961568 0.9596872 0.98902207 1.00000000
##
## Loadings:
##    Comp.1 Comp.2 Comp.3 Comp.4
```

```
## V2  0.471  0.560  0.643  0.225
## V3  0.509  0.454 -0.714 -0.156
## V4  0.524 -0.443  0.241 -0.686
## V5  0.494 -0.532 -0.134  0.674
```

```r
# Define loadings
PC <- -e$vectors[ ,c(1,2)]
(Load1 <- sqrt(e$values[1])*PC[,1])
```

```
## [1] -0.7865453 -0.8493229 -0.8749282 -0.8240901
```

```r
(Load2 <- sqrt(e$values[2])*PC[,2])
```

```
## [1] -0.5749668 -0.4666140  0.4549787  0.5466267
```

```r
p <- nrow(Rmat)
```

```r
# 1-factor solution
LL  <- Load1 %*% t(Load1)
comm <- Load1^2
Psi <- diag(rep(1,p) - comm)
round(Rmat - (LL + Psi), 3)
```

```
##         V2     V3     V4     V5
## V2  0.000  0.213 -0.250 -0.318
## V3  0.213  0.000 -0.228 -0.248
## V4 -0.250 -0.228  0.000  0.225
## V5 -0.318 -0.248  0.225  0.000
```

```r
# 2-factor solution
( L2  <- cbind(Load1,Load2) )
```

```
##           Load1      Load2
## [1,] -0.7865453 -0.5749668
## [2,] -0.8493229 -0.4666140
## [3,] -0.8749282  0.4549787
## [4,] -0.8240901  0.5466267
```

```r
LL  <- L2 %*% t(L2)
comm <- Load1^2 + Load2^2
Psi <- diag(rep(1,p) - comm)
round(Rmat - LL - Psi,3)
```

```
##         V2     V3     V4     V5
## V2  0.000 -0.055  0.011 -0.003
## V3 -0.055  0.000 -0.016  0.007
## V4  0.011 -0.016  0.000 -0.024
## V5 -0.003  0.007 -0.024  0.000
```

(b) Did the rotation improve the loadings?

```r
# (a) No rotation
( mle <- factanal(root, factors = 1, rotation="none") )
```

```
##
## Call:
## factanal(x = root, factors = 1, rotation = "none")
##
## Uniquenesses:
##    V2    V3    V4    V5
## 0.809 0.733 0.005 0.102
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
##
## Test of the hypothesis that 1 factor is sufficient.
## The chi square statistic is 63.06 on 2 degrees of freedom.
## The p-value is 2.03e-14
```

```r
attributes(mle)
```

```
## $names
##  [1] "converged"    "loadings"     "uniquenesses" "correlation"  "criteria"
##  [6] "factors"      "dof"          "method"       "STATISTIC"    "PVAL"
## [11] "n.obs"        "call"
##
## $class
## [1] "factanal"
```

```r
# control loading suppression by "cutoff"
print(loadings(mle), cutoff=0.00001)
```

```
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
```

```r
print(loadings(mle), cutoff=0.05)
```

```
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
```

```r
mle$uniquenesses
```

```
##        V2        V3        V4        V5
## 0.8085905 0.7331676 0.0050000 0.1020230
```

```r
# Error matrix
est <- tcrossprod(mle$loadings) + diag(mle$uniquenesses)
( ret <- round(Rmat - est, 3) )
```

```
##        V2     V3    V4     V5
## V2  0.000  0.655 0.002 -0.084
## V3  0.655  0.000 0.000 -0.038
## V4  0.002  0.000 0.000  0.000
## V5 -0.084 -0.038 0.000  0.000
```

```r
# Test for the # of factors
mle$PVAL
```

```
##    objective
## 2.027119e-14
```

```r
sapply(1:1, function(nf) factanal(x=root, factors = nf)$PVAL)
```

```
##    objective
## 2.027119e-14
```

```r
# (b) Varimax rotation
( mle2 <- factanal(root, factors = 1, rotation="varimax") )
```

```
##
## Call:
## factanal(x = root, factors = 1, rotation = "varimax")
##
## Uniquenesses:
##    V2    V3    V4    V5
```

```
## 0.809 0.733 0.005 0.102
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
##
## Test of the hypothesis that 1 factor is sufficient.
## The chi square statistic is 63.06 on 2 degrees of freedom.
## The p-value is 2.03e-14
```

```r
# control loading suppression by "cutoff"
print(loadings(mle2), cutoff=0.00001)
```

```
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
```

```r
mle$uniquenesses
```

```
##        V2        V3        V4        V5
## 0.8085905 0.7331676 0.0050000 0.1020230
```

```r
# Error matrix
est <- tcrossprod(mle2$loadings) + diag(mle2$uniquenesses)
( ret <- round(Rmat - est, 3) )
```

```
##        V2     V3    V4     V5
## V2  0.000  0.655 0.002 -0.084
## V3  0.655  0.000 0.000 -0.038
## V4  0.002  0.000 0.000  0.000
## V5 -0.084 -0.038 0.000  0.000
```

```r
# Test for the # of factors
mle$PVAL
```

```
##    objective
## 2.027119e-14
```

```r
sapply(1:1, function(nf) factanal(x=root, factors = nf)$PVAL)
```

```
##    objective
## 2.027119e-14
```

```r
# (c) Factor scores
( mle3 <- factanal(root, factors = 1, rotation="varimax", scores="regression") )
```

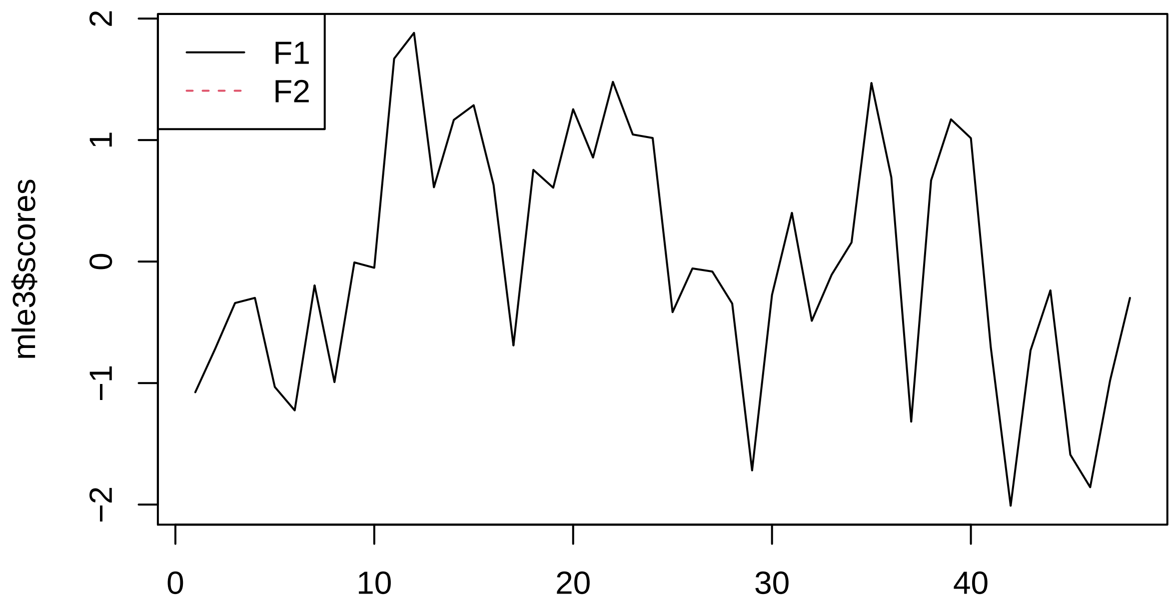
```
##
## Call:
## factanal(x = root, factors = 1, scores = "regression", rotation = "varimax")
##
## Uniquenesses:
##    V2    V3    V4    V5
## 0.809 0.733 0.005 0.102
##
## Loadings:
##    Factor1
## V2 0.438
## V3 0.517
## V4 0.998
## V5 0.948
##
##                Factor1
## SS loadings      2.351
## Proportion Var   0.588
##
## Test of the hypothesis that 1 factor is sufficient.
## The chi square statistic is 63.06 on 2 degrees of freedom.
## The p-value is 2.03e-14
```

```r
mle3$scores
```

```
##           Factor1
##  [1,] -1.075894700
##  [2,] -0.718331767
##  [3,] -0.341198228
##  [4,] -0.298849786
##  [5,] -1.031896202
##  [6,] -1.224844946
##  [7,] -0.195800897
##  [8,] -0.992300323
##  [9,] -0.006872836
## [10,] -0.051012053
## [11,]  1.670343786
## [12,]  1.882564941
## [13,]  0.611600928
## [14,]  1.166048117
## [15,]  1.286862464
## [16,]  0.631131105
## [17,] -0.690408057
## [18,]  0.755112248
```

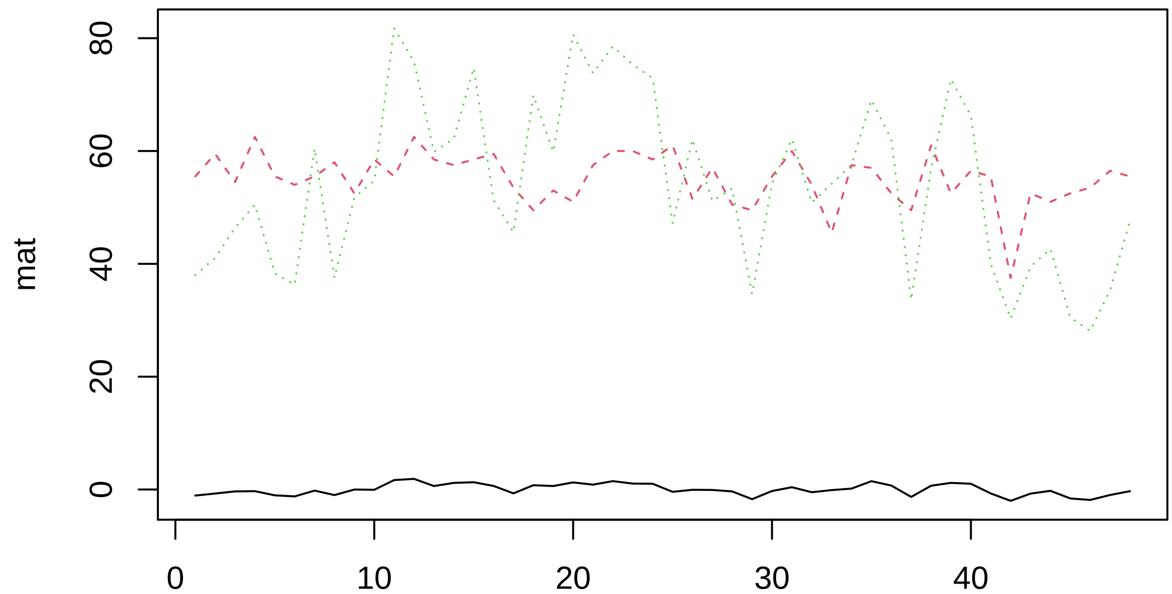
```
## [19,]  0.607838738
## [20,]  1.253940598
## [21,]  0.856203651
## [22,]  1.479499875
## [23,]  1.045930962
## [24,]  1.017287281
## [25,] -0.416305152
## [26,] -0.056489307
## [27,] -0.081960876
## [28,] -0.345549882
## [29,] -1.719014901
## [30,] -0.274300042
## [31,]  0.401054203
## [32,] -0.487855330
## [33,] -0.108051153
## [34,]  0.157448194
## [35,]  1.469903101
## [36,]  0.693291401
## [37,] -1.317695325
## [38,]  0.668203733
## [39,]  1.170790895
## [40,]  1.015961037
## [41,] -0.704108696
## [42,] -2.009658878
## [43,] -0.731035587
## [44,] -0.236974925
## [45,] -1.589301452
## [46,] -1.857350604
## [47,] -0.979113458
## [48,] -0.298841895
```

```
# plot the factors wk over wk
matplot(mle3$scores,type="l",lty=1:2, col=1:2)
legend("topleft", legend=c("F1", "F2"), lty=1:2, col=1:2)
```




```
# try to add the stock returns as well - scale to see them
mat <- data.frame(mle3$scores, 50*root[,c(1,4)])
matplot(mat,type="l")

# (d) Varimax rotation for the PC method compared to no rotation
library(psych)
```




```r
(fit1 <- principal(root, nfactors=2, rotate="none") )
```

```
## Principal Components Analysis
## Call: principal(r = root, nfactors = 2, rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##     PC1   PC2   h2    u2 com
## V2 0.79  0.57 0.95 0.051 1.8
## V3 0.85  0.47 0.94 0.061 1.6
## V4 0.87 -0.45 0.97 0.027 1.5
## V5 0.82 -0.55 0.98 0.022 1.7
##
##                        PC1  PC2
## SS loadings           2.78 1.05
## Proportion Var        0.70 0.26
## Cumulative Var        0.70 0.96
## Proportion Explained  0.73 0.27
## Cumulative Proportion 0.73 1.00
##
## Mean item complexity =  1.7
## Test of the hypothesis that 2 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.03
##  with the empirical chi square  0.39  with prob <  NA
##
## Fit based upon off diagonal values = 1
```

```r
(fit2 <- principal(root, nfactors=2, rotate="varimax") )
```

```
## Principal Components Analysis
## Call: principal(r = root, nfactors = 2, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##     RC1  RC2   h2    u2 com
## V2 0.16 0.96 0.95 0.051 1.1
## V3 0.28 0.93 0.94 0.061 1.2
## V4 0.94 0.29 0.97 0.027 1.2
## V5 0.97 0.19 0.98 0.022 1.1
##
##                        RC1  RC2
## SS loadings           1.94 1.90
## Proportion Var        0.48 0.48
## Cumulative Var        0.48 0.96
## Proportion Explained  0.50 0.50
## Cumulative Proportion 0.50 1.00
##
## Mean item complexity =  1.1
## Test of the hypothesis that 2 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.03
##  with the empirical chi square  0.39  with prob <  NA
##
## Fit based upon off diagonal values = 1
```

Yes, the rotation improved the loadings. The varimax rotated loadings have higher factor loadings in the .90's compared to the regular loadings without rotation.

14.6 Use the football data of Table 8.3, combining the three groups into a single sample. Conduct a confirmatory factor analysis of the covariance matrix using maximum likelihood to fit the model. Test the hypothesis that the observations are driven by two factors related to head size:

f1 = "horizontal dimension" f2 = "vertical dimension"

To fit an identifiable model, define the observed variable "head circumference" to be equal to /i plus error, and define the observed variable "eye-to-top-of- head measurement" to be equal to f2 plus error. In your initial model, allow the other 4 variables to be functions of both factors, for a total of 8 factor loadings to be estimated.

```r
getwd()
```

```
## [1] "/Users/isabellachittumuri/Documents/Hunter College/Fall 2020/Stat 717/HW"
```

```r
df2 <- read.table("T8_3_FOOTBALL.DAT")
df3 <- as.data.frame(df2)
head <- df3[ -c(1) ]
colnames(head) <- c("WDIM", "CIRCUM", "FBEYE", "EYEHD", "EARHD", "JAW")
```

(a) Assess goodness of fit with the criteria discussed in Section 14.3.3.

```r
library(sem)
library(semPlot)
head.cov <- cov(head); head.cov
```

```
##               WDIM    CIRCUM     FBEYE       EYEHD      EARHD       JAW
## WDIM    0.44314607 0.4836517 0.1342135 -0.06797753 0.05022472 0.1925843
## CIRCUM  0.48365169 3.5327021 1.0898340  1.48174157 0.70300125 0.6049376
## FBEYE   0.13421348 1.0898340 0.5541326  0.25027528 0.20449750 0.1820462
## EYEHD  -0.06797753 1.4817416 0.2502753  2.81560674 1.01219101 0.2482247
## EARHD   0.05022472 0.7030012 0.2044975  1.01219101 0.91566167 0.1055680
## JAW     0.19258427 0.6049376 0.1820462  0.24822472 0.10556804 0.4051486
```

```r
# Specify the model - recticular action model (RAM)
model.head <- specifyModel(text="
                           F1 ->  CIRCUM, NA, 1
                           F1 ->  WDIM,  lam1, NA
                           F1 ->  FBEYE, lam3, NA
                           F1 ->  EARHD, lam5, NA
                           F1 ->  JAW, lam6, NA
                           F2 ->  EYEHD, NA, 1
                           F2 ->  WDIM, lam2, NA
                           F2 ->  FBEYE, lam4, NA
                           F2 ->  EARHD, lam6, NA
                           F2 ->  JAW, lam8, NA
                           CIRCUM <-> CIRCUM, psi1, NA
                           EYEHD <-> EYEHD, psi2, NA
                           WDIM  <-> WDIM,  psi3, NA
                           FBEYE <-> FBEYE, psi4, NA
                           EARHD <-> EARHD, psi5, NA
                           JAW <-> JAW, psi6, NA
                           F1 <-> F1, phi1,  NA
                           F2 <-> F2, phi2,  NA
                           F1 <-> F2, phi12, NA
                             ")
```

```
## NOTE: it is generally simpler to use specifyEquations() or cfa()
##       see ?specifyEquations
```

```r
# Fit the model
head.sem <- sem(model.head, head.cov, nrow(head))

# Print results (fit indices, paramters, hypothesis tests)
summary(head.sem)
```

```
##
##  Model Chisquare =  12.28093   Df =  5 Pr(>Chisq) = 0.03113465
##  AIC =  44.28093
##  BIC =  -10.21811
##
##  Normalized Residuals
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.5495360 -0.1602494  0.0004129  0.1172366  0.2112464  2.6022203
```

```
##
##  R-square for Endogenous Variables
## CIRCUM   WDIM  FBEYE  EARHD    JAW  EYEHD
## 0.9960 0.1930 0.6298 0.2797 0.2399 1.8914
##
##  Parameter Estimates
##       Estimate    Std Error  z value     Pr(>|z|)
## lam1   0.16376182 0.04093254  4.00077368 6.313572e-05 WDIM <--- F1
## lam3   0.33013986 0.04551172  7.25395268 4.047816e-13 FBEYE <--- F1
## lam5   0.13337452 0.04638958  2.87509662 4.039039e-03 EARHD <--- F1
## lam6   0.15815065 0.03361466  4.70481153 2.541006e-06 JAW <--- F1
## lam2  -0.06253527 0.02876479 -2.17402122 2.970354e-02 WDIM <--- F2
## lam4  -0.04872922 0.02461075 -1.97999743 4.770382e-02 FBEYE <--- F2
## lam8   0.01280782 0.01665686  0.76892168 4.419398e-01 JAW <--- F2
## psi1   0.01400221 0.32366539  0.04326139 9.654932e-01 CIRCUM <--> CIRCUM
## psi2  -2.58230050 1.25394098 -2.05934771 3.946094e-02 EYEHD <--> EYEHD
## psi3   0.35762758 0.05368246  6.66190725 2.702966e-11 WDIM <--> WDIM
## psi4   0.20513739 0.04749556  4.31908578 1.566769e-05 FBEYE <--> FBEYE
## psi5   0.67488416 0.10562122  6.38966459 1.662500e-10 EARHD <--> EARHD
## psi6   0.30077191 0.04573322  6.57666237 4.811255e-11 JAW <--> JAW
## phi1   3.51914060 0.62068946  5.66972828 1.430242e-08 F1 <--> F1
## phi2   5.47919818 1.37831681  3.97528212 7.029588e-05 F2 <--> F2
## phi12  1.47862488 0.37357621  3.95802737 7.557130e-05 F2 <--> F1
##
##  Iterations =  64
```

```
# Print coefficients (loadings)
coef(head.sem)
```

```
##        lam1        lam3        lam5        lam6        lam2        lam4
##  0.16376182  0.33013986  0.13337452  0.15815065 -0.06253527 -0.04872922
##        lam8        psi1        psi2        psi3        psi4        psi5
##  0.01280782  0.01400221 -2.58230050  0.35762758  0.20513739  0.67488416
##        psi6        phi1        phi2       phi12
##  0.30077191  3.51914060  5.47919818  1.47862488
```

```
# Print standardized coefficients (loadings)
stdCoef(head.sem)
```

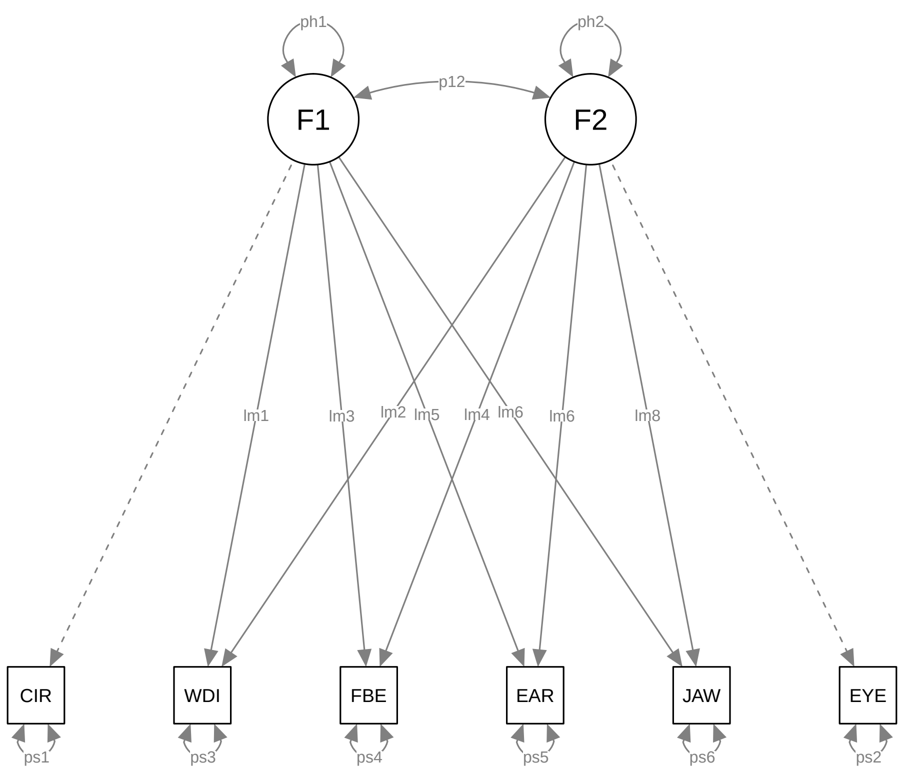
```
##          Std. Estimate
## 1           0.998016481     CIRCUM <--- F1
## 2    lam1   0.461484928       WDIM <--- F1
## 3    lam3   0.831973332      FBEYE <--- F1
## 4    lam5   0.258489516      EARHD <--- F1
## 5    lam6   0.471647020        JAW <--- F1
## 6           1.375282530      EYEHD <--- F2
## 7    lam2  -0.219892317       WDIM <--- F2
## 8    lam4  -0.153229057      FBEYE <--- F2
## 9    lam6   0.382455783      EARHD <--- F2
## 10   lam8   0.047660834        JAW <--- F2
## 11   psi1   0.003963104 CIRCUM <--> CIRCUM
## 12   psi2  -0.891402037   EYEHD <--> EYEHD
## 13   psi3   0.807019627     WDIM <--> WDIM
```

```
## 14  psi4   0.370195466   FBEYE <--> FBEYE
## 15  psi5   0.720332038   EARHD <--> EARHD
## 16  psi6   0.760138787       JAW <--> JAW
## 17  phi1   1.000000000         F1 <--> F1
## 18  phi2   1.000000000         F2 <--> F2
## 19 phi12   0.336729523         F2 <--> F1
```

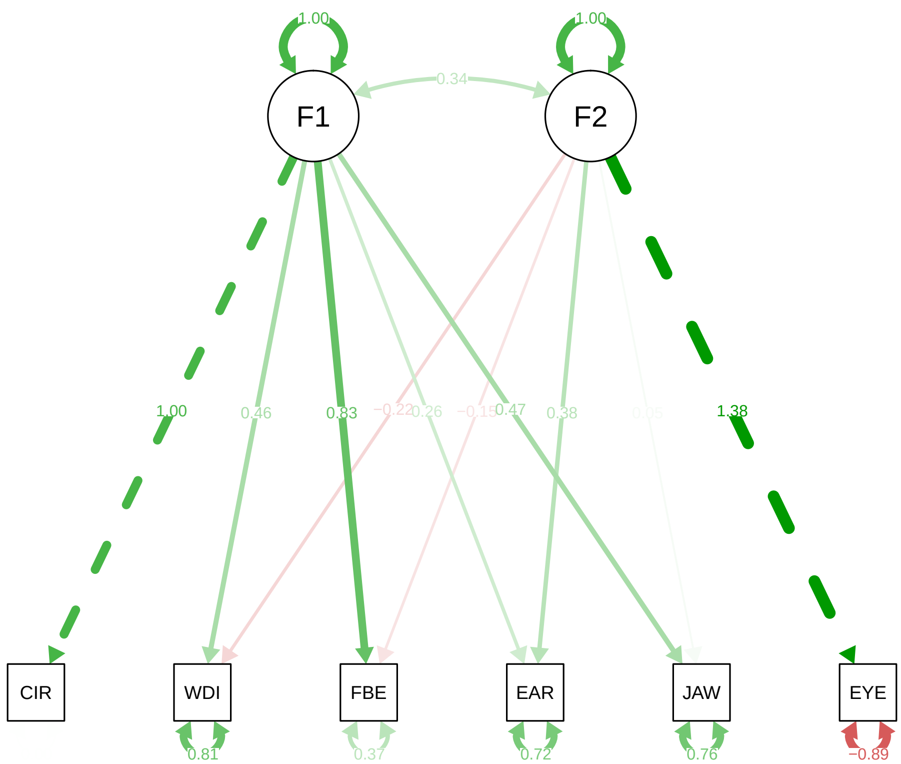
```
head.sem$t
```

```
## [1] 16
```

```
# Plotting the graph representation of the model
semPlot::semPaths(head.sem)
```




```
semPlot::semPaths(head.sem, "std")
```




```
#=== Hypothesis testing - not rejected

head.sem$criterion
```

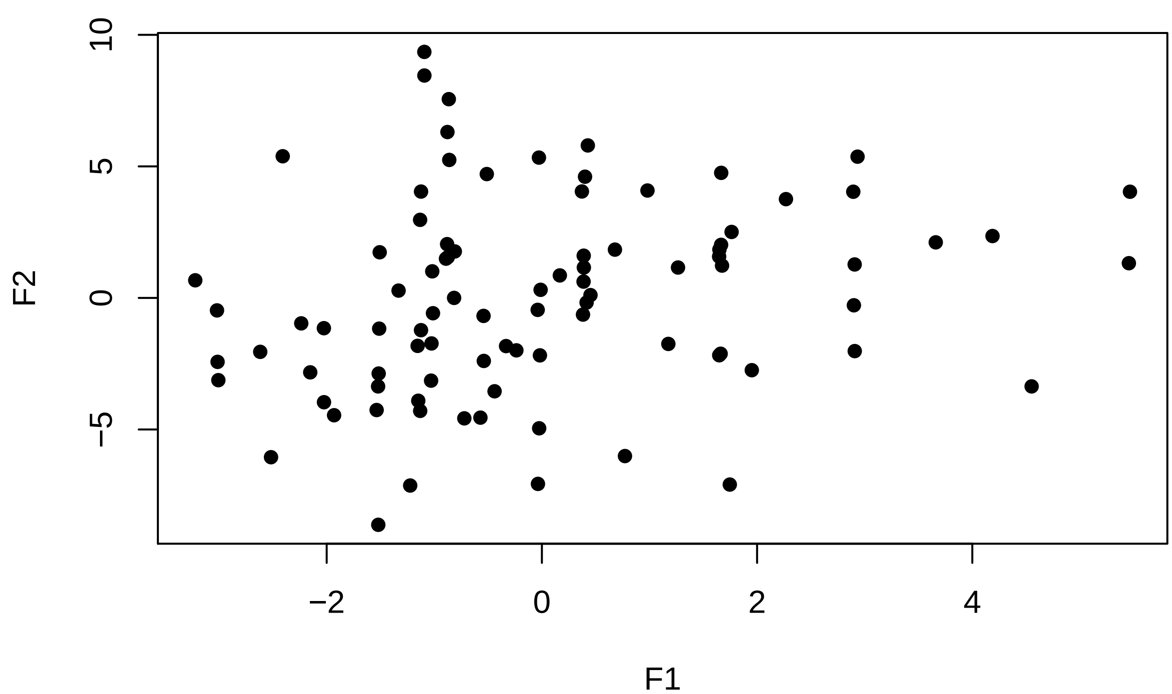
```
## [1] 0.137988
```

```
## 'objectiveML'
summary(head.sem, conf.level=.90, robust=FALSE
        , analytic.se=head.sem$t <= 100
        , fit.indices=c("GFI", "RMSEA", "SRMR") #, "AIC", "AICc", "BIC", "CAIC")
        )
```

```
##
##  Model Chisquare =  12.28093   Df =  5 Pr(>Chisq) = 0.03113465
##  Goodness-of-fit index =  0.9579365
##  RMSEA index =  0.1279127   90% CI: (0.03560232, 0.2204232)
##  SRMR =  0.06565363
##
##  Normalized Residuals
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.5495360 -0.1602494  0.0004129  0.1172366  0.2112464  2.6022203
##
##  R-square for Endogenous Variables
## CIRCUM   WDIM  FBEYE  EARHD    JAW  EYEHD
## 0.9960 0.1930 0.6298 0.2797 0.2399 1.8914
```

```
##
##  Parameter Estimates
##       Estimate    Std Error  z value     Pr(>|z|)
## lam1   0.16376182 0.04093254  4.00077368 6.313572e-05 WDIM <--- F1
## lam3   0.33013986 0.04551172  7.25395268 4.047816e-13 FBEYE <--- F1
## lam5   0.13337452 0.04638958  2.87509662 4.039039e-03 EARHD <--- F1
## lam6   0.15815065 0.03361466  4.70481153 2.541006e-06 JAW <--- F1
## lam2  -0.06253527 0.02876479 -2.17402122 2.970354e-02 WDIM <--- F2
## lam4  -0.04872922 0.02461075 -1.97999743 4.770382e-02 FBEYE <--- F2
## lam8   0.01280782 0.01665686  0.76892168 4.419398e-01 JAW <--- F2
## psi1   0.01400221 0.32366539  0.04326139 9.654932e-01 CIRCUM <--> CIRCUM
## psi2  -2.58230050 1.25394098 -2.05934771 3.946094e-02 EYEHD <--> EYEHD
## psi3   0.35762758 0.05368246  6.66190725 2.702966e-11 WDIM <--> WDIM
## psi4   0.20513739 0.04749556  4.31908578 1.566769e-05 FBEYE <--> FBEYE
## psi5   0.67488416 0.10562122  6.38966459 1.662500e-10 EARHD <--> EARHD
## psi6   0.30077191 0.04573322  6.57666237 4.811255e-11 JAW <--> JAW
## phi1   3.51914060 0.62068946  5.66972828 1.430242e-08 F1 <--> F1
## phi2   5.47919818 1.37831681  3.97528212 7.029588e-05 F2 <--> F2
## phi12  1.47862488 0.37357621  3.95802737 7.557130e-05 F2 <--> F1
##
##  Iterations =  64
```

```r
#=== Factor Scores
fs <- fscores(head.sem, data=head)
plot(fs, pch=16)
```



According to the CFI (comparative fit index), which was 0.957, the model is an okay fit because this value is just barely greater than 0.95. According to the RMSEA (Root mean sq. error approx.), which was 0.127, the model is not a good fit because this value is greater than 0.06. According to the SRMR (standardized root mean sq. res.), which was 0.065, the model is a good fit because it's less than 0.08.

F1, the effort (score), and F2, the knowledge mastery (score), are calculated and plotted against each other. There doesn't seem to be an correlation between the two scores.

(b) For comparison, fit the 2-factor model with simple structure. That is, fit the model with head width, head circumference, front-to-back measurement at eye level, and jaw width loading only on fi . Similarly, let eye- to-top-of-head measurement and ear-to-top-of-head measurement load only on f2. Use goodness-of-fit criteria and hypothesis tests on factor loadings to compare the initial model with this simple-structure model. Which model is preferable?

```
# Specify the model - reticular action model (RAM)
model.head <- specifyModel(text="
                           F1 ->  CIRCUM, NA, 1
                           F1 ->  WDIM,  lam1, NA
                           F1 ->  FBEYE, lam3, NA
                           F1 ->  JAW, lam6, NA
                           F2 ->  EYEHD, NA, 1
                           F2 ->  EARHD, lam6, NA
                           CIRCUM <-> CIRCUM, psi1, NA
                           EYEHD <-> EYEHD, psi2, NA
                           WDIM  <-> WDIM,  psi3, NA
                           FBEYE <-> FBEYE, psi4, NA
                           EARHD <-> EARHD, psi5, NA
                           JAW <-> JAW, psi6, NA
                           F1 <-> F1, phi1,  NA
                           F2 <-> F2, phi2,  NA
                           F1 <-> F2, phi12, NA
                             ")
```

```
## NOTE: it is generally simpler to use specifyEquations() or cfa()
##       see ?specifyEquations
```

```
# Fit the model
head.sem <- sem(model.head, data=head)

# Print results (fit indices, parameters, hypothesis tests)
summary(head.sem)
```

```
##
##  Model Chisquare =  39.78585   Df =  9 Pr(>Chisq) = 8.308242e-06
##  AIC =  63.78585
##  BIC =  -0.7124419
##
##  Normalized Residuals
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.92933 -0.08624  0.29067  0.46797  1.14795  2.65088
##
##  R-square for Endogenous Variables
## CIRCUM   WDIM  FBEYE    JAW  EYEHD  EARHD
```

```
## 1.1708 0.1194 0.5265 0.2228 2.0307 0.1416
##
##  Parameter Estimates
##       Estimate   Std Error  z value   Pr(>|z|)
## lam1   0.1123294 0.03198319  3.512139 4.445155e-04 WDIM <--- F1
## lam3   0.2637613 0.03511289  7.511809 5.831603e-14 FBEYE <--- F1
## lam6   0.1515509 0.03106977  4.877762 1.072962e-06 JAW <--- F1
## psi1  -0.6118571 0.39328861 -1.555746 1.197685e-01 CIRCUM <--> CIRCUM
## psi2  -2.6791181 1.30210285 -2.057532 3.963509e-02 EYEHD <--> EYEHD
## psi3   0.3902270 0.05757137  6.778142 1.217308e-11 WDIM <--> WDIM
## psi4   0.2623578 0.04636429  5.658617 1.525976e-08 FBEYE <--> FBEYE
## psi5   0.7350658 0.11380405  6.459048 1.053637e-10 EARHD <--> EARHD
## psi6   0.3359546 0.04978829  6.747663 1.502459e-11 JAW <--> JAW
## phi1   4.1939732 0.65416642  6.411172 1.444055e-10 F1 <--> F1
## phi2   5.2783583 1.43124182  3.687957 2.260618e-04 F2 <--> F2
## phi12  1.3687390 0.34045235  4.020354 5.811074e-05 F2 <--> F1
##
##  Iterations =  49
```

```
# Print coefficients (loadings)
coef(head.sem)
```

```
##       lam1       lam3       lam6       psi1       psi2       psi3       psi4
##  0.1123294  0.2637613  0.1515509 -0.6118571 -2.6791181  0.3902270  0.2623578
##       psi5       psi6       phi1       phi2      phi12
##  0.7350658  0.3359546  4.1939732  5.2783583  1.3687390
```

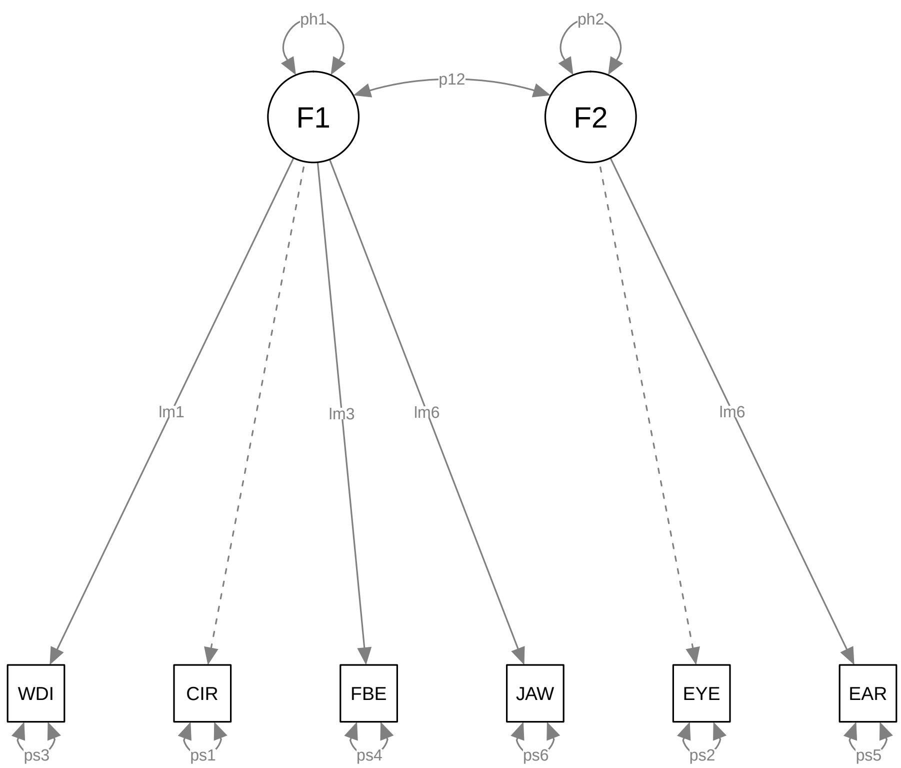
```
# Print standardized coefficients (loadings)
stdCoef(head.sem)
```

```
##          Std. Estimate
## 1            1.0820392     CIRCUM <--- F1
## 2   lam1     0.3455674       WDIM <--- F1
## 3   lam3     0.7256331      FBEYE <--- F1
## 4   lam6     0.4720507        JAW <--- F1
## 5            1.4250373      EYEHD <--- F2
## 6   lam6     0.3762666      EARHD <--- F2
## 7   psi1    -0.1708088 CIRCUM <--> CIRCUM
## 8   psi2    -1.0307312   EYEHD <--> EYEHD
## 9   psi3     0.8805832     WDIM <--> WDIM
## 10  psi4     0.4734566   FBEYE <--> FBEYE
## 11  psi5     0.8584234   EARHD <--> EARHD
## 12  psi6     0.7771681       JAW <--> JAW
## 13  phi1     1.0000000         F1 <--> F1
## 14  phi2     1.0000000         F2 <--> F2
## 15 phi12     0.2909098         F2 <--> F1
```

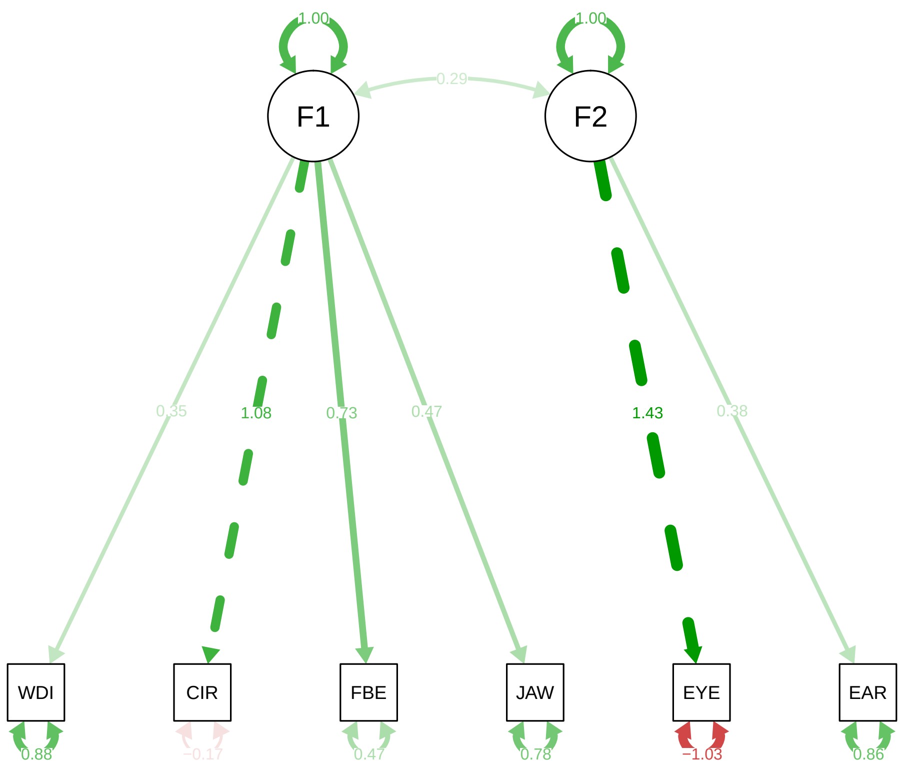
```
head.sem$t
```

```
## [1] 12
```

```r
# Plotting the graph representation of the model
semPlot::semPaths(head.sem)
```




```r
semPlot::semPaths(head.sem, "std")
```




```r
#=== Hypothesis testing - not rejected

head.sem$criterion
```

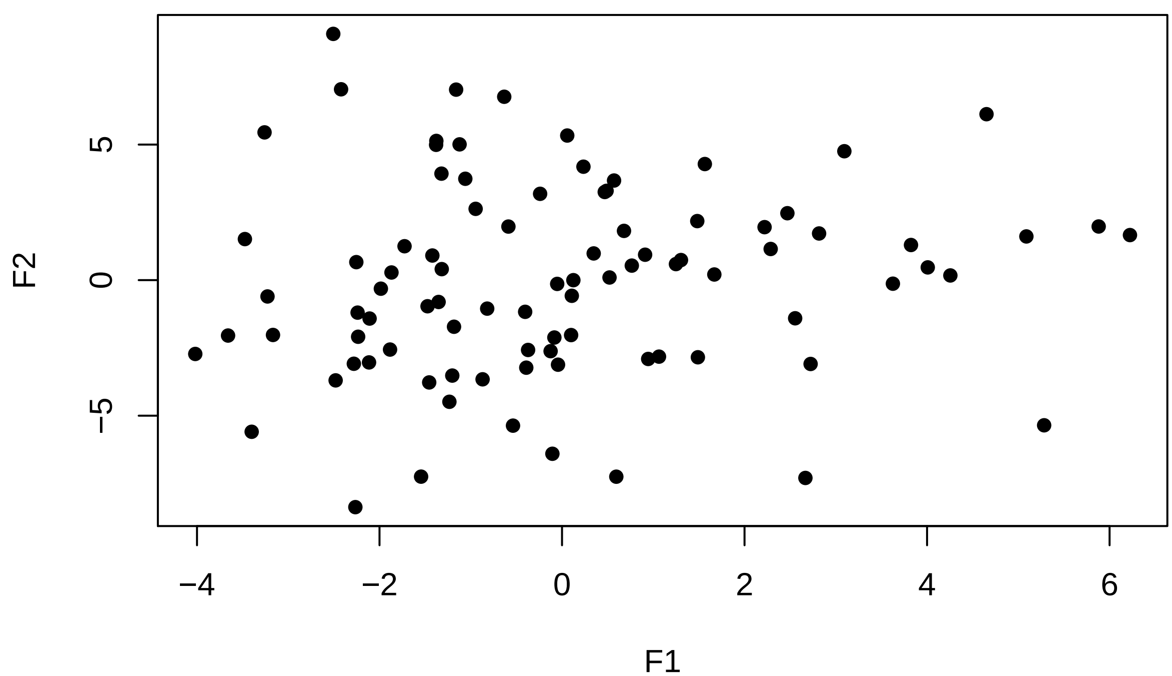
```
## [1] 0.447032
```

```r
## 'objectiveML'
summary(head.sem, conf.level=.90, robust=FALSE
        , analytic.se=head.sem$t <= 100
        , fit.indices=c("GFI", "RMSEA", "SRMR") #, "AIC", "AICc", "BIC", "CAIC")
        )
```

```
##
##  Model Chisquare =  39.78585   Df =  9 Pr(>Chisq) = 8.308242e-06
##  Goodness-of-fit index =  0.8855191
##  RMSEA index =  0.1960466   90% CI: (0.1363188, 0.260189)
##  SRMR =  0.1201244
##
##  Normalized Residuals
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.92933 -0.08624  0.29067  0.46797  1.14795  2.65088
##
##  R-square for Endogenous Variables
## CIRCUM   WDIM  FBEYE    JAW  EYEHD  EARHD
## 1.1708 0.1194 0.5265 0.2228 2.0307 0.1416
```

```
##
##  Parameter Estimates
##       Estimate   Std Error  z value   Pr(>|z|)
## lam1   0.1123294 0.03198319  3.512139 4.445155e-04 WDIM <--- F1
## lam3   0.2637613 0.03511289  7.511809 5.831603e-14 FBEYE <--- F1
## lam6   0.1515509 0.03106977  4.877762 1.072962e-06 JAW <--- F1
## psi1  -0.6118571 0.39328861 -1.555746 1.197685e-01 CIRCUM <--> CIRCUM
## psi2  -2.6791181 1.30210285 -2.057532 3.963509e-02 EYEHD <--> EYEHD
## psi3   0.3902270 0.05757137  6.778142 1.217308e-11 WDIM <--> WDIM
## psi4   0.2623578 0.04636429  5.658617 1.525976e-08 FBEYE <--> FBEYE
## psi5   0.7350658 0.11380405  6.459048 1.053637e-10 EARHD <--> EARHD
## psi6   0.3359546 0.04978829  6.747663 1.502459e-11 JAW <--> JAW
## phi1   4.1939732 0.65416642  6.411172 1.444055e-10 F1 <--> F1
## phi2   5.2783583 1.43124182  3.687957 2.260618e-04 F2 <--> F2
## phi12  1.3687390 0.34045235  4.020354 5.811074e-05 F2 <--> F1
##
##  Iterations =  49
```

```
#=== Factor Scores
fs <- fscores(head.sem, data=head)
plot(fs, pch=16)
```



According to the CFI, which was 0.885, the model was not a good fit. According to the RMSEA, which was 0.196, the model was not a good fit. According to the SRMR, which was 0.12, the model was not a good fit.

Looking at the plotted F1 and F2, there seems to be no correlation between the two scores.

Because of these results, the first model is preferable.