Fisher example

kmg

2/7/2019

Prepare environment

```
library(lavaan)

## Warning: package 'lavaan' was built under R version 3.4.4

## This is lavaan 0.6-3

## lavaan is BETA software! Please report any bugs.

library(semPlot)

load("~/Dropbox/Classes/IAV/Data/Fisher/FisherData.Rdata")
```

Get data ready.

```
data <- FisherDataInterp[[7]] # used person number 7
data_lag <- FisherDataInterp[[7]][1:114,]
data_lag[,27:52] <- FisherDataInterp[[7]][2:115,]
names1 <- colnames(data_lag[,1:26])
colnames(data_lag) <- c(names1, paste0(names1, "C"))
```

Run PFA model

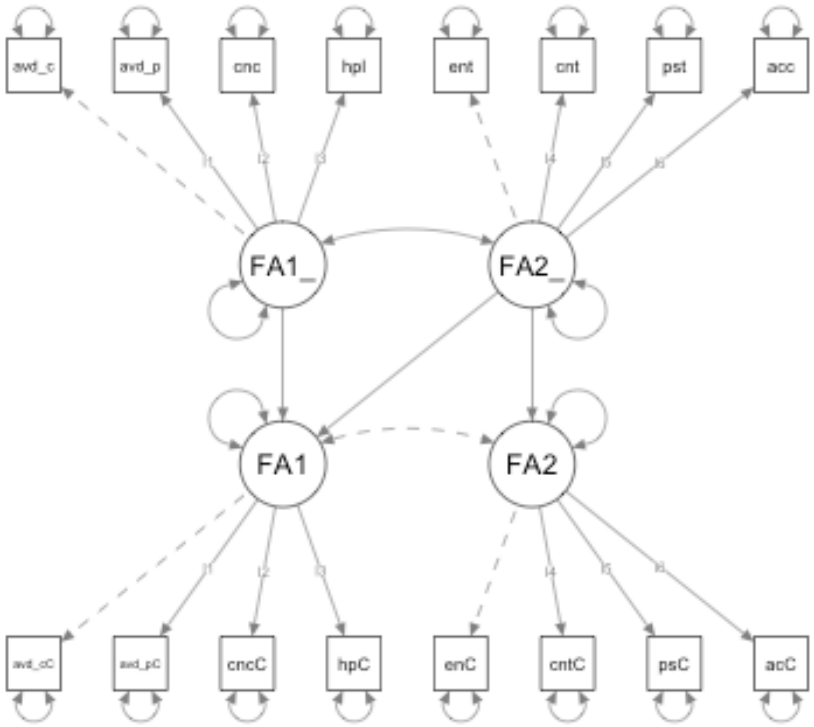
```
  model <- paste("FA1~FA2_lag", "FA1~FA1_lag", "FA2~FA2_lag", # VAR
                 "FA1~~FA1", "FA2 ~~ FA2",                # variance of LVs
                 "FA1 ~~ 0*FA2",                          # constraints
                 sep = "\n")

  # measurement models:
  FA1 <- "FA1 =~ 1*avoid_actC + l1*avoid_peopleC + l2*concentrateC + l3*hopel
essC"
  FA2 <- "FA2 =~ 1*enthusiasticC + l4*contentC + l5*positiveC + l6*acceptedC"

  FA1_lag <- "FA1_lag =~ avoid_act + l1*avoid_people + l2*concentrate + l3*ho
peless"
  FA2_lag <- "FA2_lag =~ enthusiastic + l4*content + l5*positive + l6*accepte
d"

  # Notice that lambdas are constrained to be equal with the "l#" indicators.

  # Paste it all together.
  model_full <- paste(model, FA1, FA2, FA1_lag, FA2_lag, sep = "\n")

  fit_all <- lavaan::sem(model_full, as.data.frame(data_lag))
```

Plot results.

You can also embed plots, for example:

```
semPaths(fit_all)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



Examine estimates.

Let's take a look at the measurement model parameters.

```
measurement <- parameterEstimates(fit_all)[parameterEstimates(fit_all)$op %in
% c("=~"),1:8 ]

print(measurement)
```

```
##        lhs op           rhs label   est    se      z pvalue
## 7      FA1 =~    avoid_actC       1.000 0.000     NA     NA
## 8      FA1 =~ avoid_peopleC    l1 0.853 0.081 10.558      0
## 9      FA1 =~  concentrateC    l2 0.562 0.079  7.154      0
## 10     FA1 =~     hopelessC    l3 0.670 0.084  7.988      0
## 11     FA2 =~ enthusiasticC       1.000 0.000     NA     NA
## 12     FA2 =~      contentC    l4 0.772 0.074 10.487      0
## 13     FA2 =~     positiveC    l5 0.992 0.081 12.295      0
## 14     FA2 =~     acceptedC    l6 0.728 0.075  9.699      0
## 15 FA1_lag =~     avoid_act       1.000 0.000     NA     NA
## 16 FA1_lag =~  avoid_people    l1 0.853 0.081 10.558      0
## 17 FA1_lag =~   concentrate    l2 0.562 0.079  7.154      0
## 18 FA1_lag =~      hopeless    l3 0.670 0.084  7.988      0
## 19 FA2_lag =~  enthusiastic       1.000 0.000     NA     NA
## 20 FA2_lag =~       content    l4 0.772 0.074 10.487      0
## 21 FA2_lag =~      positive    l5 0.992 0.081 12.295      0
## 22 FA2_lag =~      accepted    l6 0.728 0.075  9.699      0
```

Let's take a look at the structural model parameters.

```
structural <- parameterEstimates(fit_all)[parameterEstimates(fit_all)$op %in%
c("~"), 1:8]

print(structural)
```

```
##   lhs op     rhs label   est    se     z pvalue
## 1 FA1  ~ FA2_lag       0.082 0.113 0.724  0.469
## 2 FA1  ~ FA1_lag       0.924 0.107 8.650  0.000
## 3 FA2  ~ FA2_lag       0.547 0.097 5.647  0.000
```

Take a look at fit.

```
fitMeasures(fit_all, c("cfi","rmsea","srmr"))
```

```
##   cfi rmsea  srmr
## 0.778 0.128 0.105
```