Halal V2 Data Analysis

Rizqy Amelia Zein

March 24, 2019

Before performing data analysis, I cleaned the data by applying exclusion criteria.

## Tidying data

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.5.2

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

ds$RM01\_01 <- as.factor(ds$RM01\_01) # changing experimental group variable from character to factor  
  
# Merging columns (product perceptions items) to make it less confusing and filtering so that only Muslim participants are selected. I also applied an exclusion criteria, which is "time degradation of being fast" (TIME\_RSI). Participants with who score > 2 in TIME\_RSI, did not complete the questionnaire in a meaningful way. (see: https://www.soscisurvey.de/help/doku.php/en:results:variables)  
  
ds %<>% filter(DM08 == "Muslim", TIME\_RSI < 2) %>%  
 mutate\_at(vars(15:62), funs(replace(., is.na(.), 0))) %>%  
 mutate(tastebeverage=PC01\_01+PC02\_01+PC03\_01+PC04\_01,  
 tastecake=PC05\_01+PC06\_01+PC07\_01+PC08\_01,  
 healthybeverage=PC09\_01+PC10\_01+PC11\_01+PC12\_01,  
 healthycake=PC13\_01+PC14\_01+PC15\_01+PC16\_01,  
 qualitybeverage=PC17\_01+PC18\_01+PC19\_01+PC20\_01,  
 qualitycake=PC21\_01+PC22\_01+PC23\_01+PC24\_01,  
 willtry\_beverage=PC25\_01+PC26\_01+PC27\_01+PC28\_01,  
 willtry\_cake=PC29\_01+PC30\_01+PC31\_01+PC32\_01,  
 willbuy\_beverage=PC33\_01+PC34\_01+PC35\_01+PC36\_01,  
 willbuy\_cake=PC37\_01+PC38\_01+PC39\_01+PC40\_01,  
 entermarket\_beverage=PC41\_01+PC42\_01+PC43\_01+PC44\_01,  
 entermarket\_cake=PC45\_01+PC46\_01+PC47\_01+PC48\_01  
 )

## Warning: funs() is soft deprecated as of dplyr 0.8.0  
## please use list() instead  
##   
## # Before:  
## funs(name = f(.)  
##   
## # After:   
## list(name = ~f(.))  
## This warning is displayed once per session.

# Deleting original columns  
  
ds <- ds[, -c(15:62)]  
  
# give comments to the new variables  
comment(ds$tastebeverage) <- "What do you think about the taste of this beverage product?"  
comment(ds$tastecake) <- "What do you think about the taste of this bakery product?"  
comment(ds$healthybeverage) <- "What do you think about the health aspect of this beverage product?"  
comment(ds$healthycake) <- "What do you think about the health aspect of this bakery product?"  
comment(ds$qualitybeverage) <- "What do you think about the quality of this beverage product?"  
comment(ds$qualitycake) <- "What do you think about the quality of this cake product?"  
comment(ds$willtry\_beverage) <- "How much likely will you try this beverage product"  
comment(ds$willtry\_cake) <- "How much likely will you try this bakery product"  
comment(ds$willbuy\_beverage) <- "How much likely will you buy this beverage product"  
comment(ds$willbuy\_cake) <- "How much likely will you buy this bakery product"  
comment(ds$entermarket\_beverage) <- "Do you think this beverage product should enter Indonesian market?"  
comment(ds$entermarket\_cake) <- "Do you think this bakery product should enter Indonesian market?"  
  
# Replacing -9 to NA  
library(naniar)

## Warning: package 'naniar' was built under R version 3.5.3

ds <- replace\_with\_na\_all(ds, condition = ~.x %in% common\_na\_numbers)  
  
  
# Creating separate halal and coo variable  
ds$halal <- ifelse(ds$RM01\_01 == 1, 1, ifelse(ds$RM01\_01 == 3, 1, 0))  
ds$coo <- ifelse(ds$RM01\_01 == 3, 1, ifelse(ds$RM01\_01 == 4, 1, 0))

# Confirmatory Factor Analysis and Factor Scores Extraction

Instead of using sum scores, I used **factor scores** for data analysis. Using sum scores would potentially lead to higher measurement errors, so I tried to reduce it by treating product perception (of cake & beverage), need of cognitive closure, religious zeal and collective narcissism as latent variables.

I treated the items as ordinal (see: ordered option in fit feature, except for product perception (of cake & beverage), because the scales were styled as semi-continous (1-101)).

## CFA of product perception (cake)

library(lavaan)

## Warning: package 'lavaan' was built under R version 3.5.1

## This is lavaan 0.6-3

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

# Product Perception (cake)  
  
pp\_cake <- '  
  
pp\_cake =~ tastecake + healthycake + qualitycake + willtry\_cake + willbuy\_cake + entermarket\_cake  
  
# residuals  
  
qualitycake ~~ healthycake  
willtry\_cake ~~ willbuy\_cake  
willtry\_cake ~~ healthycake  
willtry\_cake ~~ tastecake  
  
'  
  
fitpp\_cake <- cfa(pp\_cake, data=ds, missing="ML")

## Warning in lav\_data\_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases are empty and will be ignored:  
## 243

summary(fitpp\_cake, fit.measures=T, modindices=T)

## lavaan 0.6-3 ended normally after 136 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 22  
##   
## Used Total  
## Number of observations 385 386  
## Number of missing patterns 19  
##   
## Estimator ML  
## Model Fit Test Statistic 8.566  
## Degrees of freedom 5  
## P-value (Chi-square) 0.128  
##   
## Model test baseline model:  
##   
## Minimum Function Test Statistic 1600.688  
## Degrees of freedom 15  
## P-value 0.000  
##   
## User model versus baseline model:  
##   
## Comparative Fit Index (CFI) 0.998  
## Tucker-Lewis Index (TLI) 0.993  
##   
## Loglikelihood and Information Criteria:  
##   
## Loglikelihood user model (H0) -9132.193  
## Loglikelihood unrestricted model (H1) -9127.910  
##   
## Number of free parameters 22  
## Akaike (AIC) 18308.386  
## Bayesian (BIC) 18395.357  
## Sample-size adjusted Bayesian (BIC) 18325.554  
##   
## Root Mean Square Error of Approximation:  
##   
## RMSEA 0.043  
## 90 Percent Confidence Interval 0.000 0.091  
## P-value RMSEA <= 0.05 0.527  
##   
## Standardized Root Mean Square Residual:  
##   
## SRMR 0.013  
##   
## Parameter Estimates:  
##   
## Information Observed  
## Observed information based on Hessian  
## Standard Errors Standard  
##   
## Latent Variables:  
## Estimate Std.Err z-value P(>|z|)  
## pp\_cake =~   
## tastecake 1.000   
## healthycake 0.650 0.067 9.627 0.000  
## qualitycake 0.678 0.054 12.445 0.000  
## willtry\_cake 1.405 0.072 19.473 0.000  
## willbuy\_cake 1.455 0.080 18.230 0.000  
## entermarket\_ck 1.227 0.075 16.311 0.000  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|)  
## .healthycake ~~   
## .qualitycake 103.974 14.082 7.384 0.000  
## .willtry\_cake ~~   
## .willbuy\_cake 56.164 16.491 3.406 0.001  
## .healthycake ~~   
## .willtry\_cake -6.506 8.809 -0.739 0.460  
## .tastecake ~~   
## .willtry\_cake 33.680 9.844 3.421 0.001  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|)  
## .tastecake 72.018 1.064 67.695 0.000  
## .healthycake 66.759 1.056 63.208 0.000  
## .qualitycake 70.710 0.869 81.404 0.000  
## .willtry\_cake 68.684 1.298 52.932 0.000  
## .willbuy\_cake 61.318 1.306 46.945 0.000  
## .entermarket\_ck 64.065 1.236 51.842 0.000  
## pp\_cake 0.000   
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|)  
## .tastecake 172.887 15.434 11.201 0.000  
## .healthycake 307.220 23.724 12.950 0.000  
## .qualitycake 165.684 13.458 12.311 0.000  
## .willtry\_cake 137.507 20.338 6.761 0.000  
## .willbuy\_cake 106.790 18.821 5.674 0.000  
## .entermarket\_ck 191.692 19.044 10.066 0.000  
## pp\_cake 255.859 29.728 8.607 0.000  
##   
## Modification Indices:  
##   
## lhs op rhs mi epc sepc.lv sepc.all  
## 25 tastecake ~~ healthycake 0.005 -0.831 -0.831 -0.004  
## 26 tastecake ~~ qualitycake 1.041 9.140 9.140 0.054  
## 27 tastecake ~~ willbuy\_cake 3.634 40.651 40.651 0.299  
## 28 tastecake ~~ entermarket\_cake 7.003 -42.189 -42.189 -0.232  
## 29 healthycake ~~ willbuy\_cake 0.104 -4.479 -4.479 -0.025  
## 30 healthycake ~~ entermarket\_cake 0.188 5.863 5.863 0.024  
## 31 qualitycake ~~ willtry\_cake 0.001 -0.304 -0.304 -0.002  
## 32 qualitycake ~~ willbuy\_cake 2.104 -11.688 -11.688 -0.088  
## 33 qualitycake ~~ entermarket\_cake 1.764 14.042 14.042 0.079  
## 34 willtry\_cake ~~ entermarket\_cake 0.001 0.550 0.550 0.003  
## 35 willbuy\_cake ~~ entermarket\_cake 0.395 10.140 10.140 0.071  
## sepc.nox  
## 25 -0.004  
## 26 0.054  
## 27 0.299  
## 28 -0.232  
## 29 -0.025  
## 30 0.024  
## 31 -0.002  
## 32 -0.088  
## 33 0.079  
## 34 0.003  
## 35 0.071

ds$pp\_cake <- lavPredict(fitpp\_cake) # extracting factor scores

## CFA of product perception (beverage)

# Product Perception (bev)  
  
pp\_bev <- '  
  
pp\_bev =~ tastebeverage + healthybeverage + qualitybeverage + willtry\_beverage + willbuy\_beverage + entermarket\_beverage  
  
# Residuals  
  
qualitybeverage ~~ healthybeverage  
willtry\_beverage ~~ willbuy\_beverage  
qualitybeverage ~~ entermarket\_beverage  
qualitybeverage ~~ willbuy\_beverage  
qualitybeverage ~~ willtry\_beverage  
healthybeverage ~~ entermarket\_beverage  
  
'  
  
fitpp\_bev <- cfa(pp\_bev, data=ds, missing="ML")

## Warning in lav\_data\_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases are empty and will be ignored:  
## 243

summary(fitpp\_bev, fit.measures=T, modindices=T)

## lavaan 0.6-3 ended normally after 175 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 24  
##   
## Used Total  
## Number of observations 385 386  
## Number of missing patterns 15  
##   
## Estimator ML  
## Model Fit Test Statistic 2.522  
## Degrees of freedom 3  
## P-value (Chi-square) 0.471  
##   
## Model test baseline model:  
##   
## Minimum Function Test Statistic 1862.877  
## Degrees of freedom 15  
## P-value 0.000  
##   
## User model versus baseline model:  
##   
## Comparative Fit Index (CFI) 1.000  
## Tucker-Lewis Index (TLI) 1.001  
##   
## Loglikelihood and Information Criteria:  
##   
## Loglikelihood user model (H0) -9138.105  
## Loglikelihood unrestricted model (H1) -9136.844  
##   
## Number of free parameters 24  
## Akaike (AIC) 18324.211  
## Bayesian (BIC) 18419.089  
## Sample-size adjusted Bayesian (BIC) 18342.940  
##   
## Root Mean Square Error of Approximation:  
##   
## RMSEA 0.000  
## 90 Percent Confidence Interval 0.000 0.081  
## P-value RMSEA <= 0.05 0.777  
##   
## Standardized Root Mean Square Residual:  
##   
## SRMR 0.004  
##   
## Parameter Estimates:  
##   
## Information Observed  
## Observed information based on Hessian  
## Standard Errors Standard  
##   
## Latent Variables:  
## Estimate Std.Err z-value P(>|z|)  
## pp\_bev =~   
## tastebeverage 1.000   
## healthybeverag 0.723 0.061 11.874 0.000  
## qualitybeverag 0.683 0.057 11.897 0.000  
## willtry\_beverg 1.437 0.079 18.139 0.000  
## willbuy\_beverg 1.340 0.078 17.147 0.000  
## entermrkt\_bvrg 1.103 0.065 17.054 0.000  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|)  
## .healthybeverage ~~   
## .qualitybeverag 150.319 17.419 8.630 0.000  
## .willtry\_beverage ~~   
## .willbuy\_beverg 43.561 22.291 1.954 0.051  
## .qualitybeverage ~~   
## .entermrkt\_bvrg 45.484 15.626 2.911 0.004  
## .willbuy\_beverg -24.661 12.350 -1.997 0.046  
## .willtry\_beverg -17.905 12.719 -1.408 0.159  
## .healthybeverage ~~   
## .entermrkt\_bvrg 45.377 15.966 2.842 0.004  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|)  
## .tastebeverage 56.957 1.188 47.956 0.000  
## .healthybeverag 70.408 1.106 63.632 0.000  
## .qualitybeverag 69.523 0.933 74.546 0.000  
## .willtry\_beverg 57.513 1.407 40.884 0.000  
## .willbuy\_beverg 54.341 1.368 39.712 0.000  
## .entermrkt\_bvrg 60.439 1.238 48.803 0.000  
## pp\_bev 0.000   
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|)  
## .tastebeverage 209.094 18.884 11.073 0.000  
## .healthybeverag 291.057 23.045 12.630 0.000  
## .qualitybeverag 176.167 18.394 9.577 0.000  
## .willtry\_beverg 90.976 24.110 3.773 0.000  
## .willbuy\_beverg 136.561 24.567 5.559 0.000  
## .entermrkt\_bvrg 191.579 19.770 9.691 0.000  
## pp\_bev 322.035 37.170 8.664 0.000  
##   
## Modification Indices:  
##   
## lhs op rhs mi epc sepc.lv sepc.all  
## 27 tastebeverage ~~ healthybeverage 0.150 -5.777 -5.777 -0.023  
## 29 tastebeverage ~~ willtry\_beverage 0.749 10.742 10.742 0.078  
## 30 tastebeverage ~~ willbuy\_beverage 0.749 -10.018 -10.018 -0.059  
## 31 tastebeverage ~~ entermarket\_beverage 0.150 8.819 8.819 0.044  
## 32 healthybeverage ~~ willtry\_beverage 1.344 12.457 12.457 0.077  
## 33 healthybeverage ~~ willbuy\_beverage 1.039 -10.797 -10.797 -0.054  
## 34 willtry\_beverage ~~ entermarket\_beverage 2.472 -18.474 -18.474 -0.140  
## 35 willbuy\_beverage ~~ entermarket\_beverage 2.170 16.592 16.592 0.103  
## sepc.nox  
## 27 -0.023  
## 29 0.078  
## 30 -0.059  
## 31 0.044  
## 32 0.077  
## 33 -0.054  
## 34 -0.140  
## 35 0.103

ds$pp\_bev <- lavPredict(fitpp\_bev)

## CFA of collective narcissism

# Collective Narcissism   
  
colnar <- '  
  
colnar =~ SK08\_01 + SK08\_02 + SK08\_03 + SK08\_04 + SK08\_05  
  
# Residuals  
SK08\_01 ~~ SK08\_04  
SK08\_03 ~~ SK08\_04  
  
'  
  
fitcn <- cfa(colnar, data=ds, ordered=c("SK08\_01", "SK08\_02", "SK08\_03",  
 "SK08\_04", "SK08\_05"))  
summary(fitcn, fit.measures=T, modindices=T)

## lavaan 0.6-3 ended normally after 15 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 22  
##   
## Number of observations 386  
##   
## Estimator DWLS Robust  
## Model Fit Test Statistic 5.320 11.526  
## Degrees of freedom 3 3  
## P-value (Chi-square) 0.150 0.009  
## Scaling correction factor 0.468  
## Shift parameter 0.147  
## for simple second-order correction (Mplus variant)  
##   
## Model test baseline model:  
##   
## Minimum Function Test Statistic 1269.455 916.061  
## Degrees of freedom 10 10  
## P-value 0.000 0.000  
##   
## User model versus baseline model:  
##   
## Comparative Fit Index (CFI) 0.998 0.991  
## Tucker-Lewis Index (TLI) 0.994 0.969  
##   
## Robust Comparative Fit Index (CFI) NA  
## Robust Tucker-Lewis Index (TLI) NA  
##   
## Root Mean Square Error of Approximation:  
##   
## RMSEA 0.045 0.086  
## 90 Percent Confidence Interval 0.000 0.106 0.037 0.141  
## P-value RMSEA <= 0.05 0.467 0.101  
##   
## Robust RMSEA NA  
## 90 Percent Confidence Interval NA NA  
##   
## Standardized Root Mean Square Residual:  
##   
## SRMR 0.027 0.027  
##   
## Parameter Estimates:  
##   
## Information Expected  
## Information saturated (h1) model Unstructured  
## Standard Errors Robust.sem  
##   
## Latent Variables:  
## Estimate Std.Err z-value P(>|z|)  
## colnar =~   
## SK08\_01 1.000   
## SK08\_02 0.943 0.055 17.163 0.000  
## SK08\_03 0.954 0.061 15.647 0.000  
## SK08\_04 0.411 0.078 5.243 0.000  
## SK08\_05 1.002 0.061 16.460 0.000  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK08\_01 ~~   
## .SK08\_04 -0.144 0.040 -3.580 0.000  
## .SK08\_03 ~~   
## .SK08\_04 0.088 0.041 2.146 0.032  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|)  
## .SK08\_01 0.000   
## .SK08\_02 0.000   
## .SK08\_03 0.000   
## .SK08\_04 0.000   
## .SK08\_05 0.000   
## colnar 0.000   
##   
## Thresholds:  
## Estimate Std.Err z-value P(>|z|)  
## SK08\_01|t1 -0.771 0.071 -10.822 0.000  
## SK08\_01|t2 0.523 0.067 7.787 0.000  
## SK08\_01|t3 1.604 0.105 15.300 0.000  
## SK08\_02|t1 -1.129 0.081 -13.916 0.000  
## SK08\_02|t2 0.464 0.066 6.987 0.000  
## SK08\_02|t3 1.476 0.097 15.239 0.000  
## SK08\_03|t1 -1.276 0.087 -14.686 0.000  
## SK08\_03|t2 -0.176 0.064 -2.744 0.006  
## SK08\_03|t3 0.992 0.077 12.930 0.000  
## SK08\_04|t1 -1.865 0.126 -14.777 0.000  
## SK08\_04|t2 -0.920 0.075 -12.311 0.000  
## SK08\_04|t3 0.754 0.071 10.631 0.000  
## SK08\_05|t1 -1.093 0.080 -13.681 0.000  
## SK08\_05|t2 0.379 0.066 5.781 0.000  
## SK08\_05|t3 1.496 0.098 15.263 0.000  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK08\_01 0.435   
## .SK08\_02 0.497   
## .SK08\_03 0.486   
## .SK08\_04 0.904   
## .SK08\_05 0.432   
## colnar 0.565 0.047 11.992 0.000  
##   
## Scales y\*:  
## Estimate Std.Err z-value P(>|z|)  
## SK08\_01 1.000   
## SK08\_02 1.000   
## SK08\_03 1.000   
## SK08\_04 1.000   
## SK08\_05 1.000   
##   
## Modification Indices:  
##   
## lhs op rhs mi epc sepc.lv sepc.all sepc.nox  
## 40 SK08\_01 ~~ SK08\_02 1.026 0.065 0.065 0.140 0.140  
## 41 SK08\_01 ~~ SK08\_03 0.178 0.028 0.028 0.061 0.061  
## 42 SK08\_01 ~~ SK08\_05 2.116 -0.102 -0.102 -0.234 -0.234  
## 43 SK08\_02 ~~ SK08\_03 0.397 -0.040 -0.040 -0.081 -0.081  
## 44 SK08\_02 ~~ SK08\_04 4.237 -0.138 -0.138 -0.206 -0.206  
## 45 SK08\_02 ~~ SK08\_05 0.178 0.028 0.028 0.060 0.060  
## 46 SK08\_03 ~~ SK08\_05 0.041 0.013 0.013 0.029 0.029  
## 47 SK08\_04 ~~ SK08\_05 4.237 0.147 0.147 0.235 0.235

ds$cn <- lavPredict(fitcn)

## CFA of religious zeal

# Religious Zeal  
  
rz <- '  
rz =~ SK09\_01 + SK09\_02 + SK09\_03 + SK09\_04 + SK09\_05 + SK09\_06 + SK09\_07 + SK09\_08 + SK09\_09 + SK09\_10 + SK09\_11 + SK09\_12 + SK09\_13 + SK09\_14 + SK09\_15 + SK09\_16  
  
# Residuals  
SK09\_01 ~~ SK09\_02  
SK09\_06 ~~ SK09\_13  
SK09\_10 ~~ SK09\_11  
SK09\_12 ~~ SK09\_16  
SK09\_03 ~~ SK09\_04  
SK09\_08 ~~ SK09\_12  
SK09\_06 ~~ SK09\_12  
'  
  
fitrz <- cfa(rz, data=ds, ordered=c("SK09\_01", "SK09\_02", "SK09\_03", "SK09\_04", "SK09\_05", "SK09\_06", "SK09\_07", "SK09\_08",  
 "SK09\_09", "SK09\_10", "SK09\_11", "SK09\_12", "SK09\_13", "SK09\_14", "SK09\_15", "SK09\_16"))  
summary(fitrz, fit.measures=T, modindices=T)

## lavaan 0.6-3 ended normally after 33 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 71  
##   
## Number of observations 386  
##   
## Estimator DWLS Robust  
## Model Fit Test Statistic 174.331 284.216  
## Degrees of freedom 97 97  
## P-value (Chi-square) 0.000 0.000  
## Scaling correction factor 0.674  
## Shift parameter 25.495  
## for simple second-order correction (Mplus variant)  
##   
## Model test baseline model:  
##   
## Minimum Function Test Statistic 20407.491 8313.897  
## Degrees of freedom 120 120  
## P-value 0.000 0.000  
##   
## User model versus baseline model:  
##   
## Comparative Fit Index (CFI) 0.996 0.977  
## Tucker-Lewis Index (TLI) 0.995 0.972  
##   
## Robust Comparative Fit Index (CFI) NA  
## Robust Tucker-Lewis Index (TLI) NA  
##   
## Root Mean Square Error of Approximation:  
##   
## RMSEA 0.046 0.071  
## 90 Percent Confidence Interval 0.034 0.056 0.061 0.080  
## P-value RMSEA <= 0.05 0.743 0.000  
##   
## Robust RMSEA NA  
## 90 Percent Confidence Interval NA NA  
##   
## Standardized Root Mean Square Residual:  
##   
## SRMR 0.054 0.054  
##   
## Parameter Estimates:  
##   
## Information Expected  
## Information saturated (h1) model Unstructured  
## Standard Errors Robust.sem  
##   
## Latent Variables:  
## Estimate Std.Err z-value P(>|z|)  
## rz =~   
## SK09\_01 1.000   
## SK09\_02 0.921 0.036 25.316 0.000  
## SK09\_03 1.046 0.038 27.663 0.000  
## SK09\_04 0.782 0.048 16.127 0.000  
## SK09\_05 0.905 0.044 20.498 0.000  
## SK09\_06 0.458 0.063 7.265 0.000  
## SK09\_07 1.044 0.035 29.490 0.000  
## SK09\_08 0.953 0.041 23.398 0.000  
## SK09\_09 1.009 0.041 24.758 0.000  
## SK09\_10 0.612 0.054 11.384 0.000  
## SK09\_11 0.720 0.050 14.405 0.000  
## SK09\_12 0.855 0.042 20.174 0.000  
## SK09\_13 0.104 0.065 1.590 0.112  
## SK09\_14 1.033 0.036 28.550 0.000  
## SK09\_15 1.078 0.038 28.162 0.000  
## SK09\_16 0.991 0.039 25.437 0.000  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK09\_01 ~~   
## .SK09\_02 0.142 0.029 4.913 0.000  
## .SK09\_06 ~~   
## .SK09\_13 -0.221 0.050 -4.396 0.000  
## .SK09\_10 ~~   
## .SK09\_11 0.197 0.038 5.219 0.000  
## .SK09\_12 ~~   
## .SK09\_16 0.154 0.030 5.211 0.000  
## .SK09\_03 ~~   
## .SK09\_04 0.127 0.032 3.997 0.000  
## .SK09\_08 ~~   
## .SK09\_12 0.133 0.036 3.704 0.000  
## .SK09\_06 ~~   
## .SK09\_12 0.169 0.041 4.140 0.000  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|)  
## .SK09\_01 0.000   
## .SK09\_02 0.000   
## .SK09\_03 0.000   
## .SK09\_04 0.000   
## .SK09\_05 0.000   
## .SK09\_06 0.000   
## .SK09\_07 0.000   
## .SK09\_08 0.000   
## .SK09\_09 0.000   
## .SK09\_10 0.000   
## .SK09\_11 0.000   
## .SK09\_12 0.000   
## .SK09\_13 0.000   
## .SK09\_14 0.000   
## .SK09\_15 0.000   
## .SK09\_16 0.000   
## rz 0.000   
##   
## Thresholds:  
## Estimate Std.Err z-value P(>|z|)  
## SK09\_01|t1 -2.313 0.188 -12.321 0.000  
## SK09\_01|t2 -1.537 0.100 -15.295 0.000  
## SK09\_01|t3 0.263 0.065 4.062 0.000  
## SK09\_02|t1 -2.419 0.209 -11.552 0.000  
## SK09\_02|t2 -1.628 0.106 -15.287 0.000  
## SK09\_02|t3 0.568 0.068 8.384 0.000  
## SK09\_03|t1 -2.796 0.323 -8.649 0.000  
## SK09\_03|t2 -1.829 0.123 -14.895 0.000  
## SK09\_03|t3 0.078 0.064 1.220 0.223  
## SK09\_04|t1 -2.796 0.323 -8.649 0.000  
## SK09\_04|t2 -1.496 0.098 -15.263 0.000  
## SK09\_04|t3 0.351 0.065 5.377 0.000  
## SK09\_05|t1 -2.313 0.188 -12.321 0.000  
## SK09\_05|t2 -0.891 0.074 -12.038 0.000  
## SK09\_05|t3 0.679 0.070 9.761 0.000  
## SK09\_06|t1 -0.249 0.065 -3.859 0.000  
## SK09\_06|t2 0.834 0.073 11.483 0.000  
## SK09\_06|t3 1.796 0.120 14.994 0.000  
## SK09\_07|t1 -1.945 0.134 -14.464 0.000  
## SK09\_07|t2 -1.129 0.081 -13.916 0.000  
## SK09\_07|t3 0.615 0.068 8.977 0.000  
## SK09\_08|t1 -1.581 0.103 -15.305 0.000  
## SK09\_08|t2 -0.900 0.074 -12.129 0.000  
## SK09\_08|t3 0.170 0.064 2.642 0.008  
## SK09\_09|t1 -1.990 0.140 -14.260 0.000  
## SK09\_09|t2 -0.798 0.072 -11.107 0.000  
## SK09\_09|t3 0.843 0.073 11.577 0.000  
## SK09\_10|t1 -1.219 0.085 -14.426 0.000  
## SK09\_10|t2 -0.216 0.064 -3.352 0.001  
## SK09\_10|t3 1.179 0.083 14.215 0.000  
## SK09\_11|t1 -1.764 0.117 -15.075 0.000  
## SK09\_11|t2 -0.737 0.071 -10.439 0.000  
## SK09\_11|t3 0.754 0.071 10.631 0.000  
## SK09\_12|t1 -1.276 0.087 -14.686 0.000  
## SK09\_12|t2 -0.223 0.064 -3.454 0.001  
## SK09\_12|t3 0.789 0.072 11.013 0.000  
## SK09\_13|t1 -1.604 0.105 -15.300 0.000  
## SK09\_13|t2 -0.872 0.074 -11.855 0.000  
## SK09\_13|t3 0.530 0.067 7.887 0.000  
## SK09\_14|t1 -2.094 0.153 -13.717 0.000  
## SK09\_14|t2 -1.369 0.091 -15.014 0.000  
## SK09\_14|t3 0.379 0.066 5.781 0.000  
## SK09\_15|t1 -2.094 0.153 -13.717 0.000  
## SK09\_15|t2 -1.516 0.099 -15.281 0.000  
## SK09\_15|t3 0.183 0.064 2.845 0.004  
## SK09\_16|t1 -1.829 0.123 -14.895 0.000  
## SK09\_16|t2 -0.638 0.069 -9.272 0.000  
## SK09\_16|t3 0.737 0.071 10.439 0.000  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK09\_01 0.374   
## .SK09\_02 0.469   
## .SK09\_03 0.316   
## .SK09\_04 0.617   
## .SK09\_05 0.487   
## .SK09\_06 0.869   
## .SK09\_07 0.319   
## .SK09\_08 0.432   
## .SK09\_09 0.364   
## .SK09\_10 0.766   
## .SK09\_11 0.676   
## .SK09\_12 0.543   
## .SK09\_13 0.993   
## .SK09\_14 0.333   
## .SK09\_15 0.273   
## .SK09\_16 0.386   
## rz 0.626 0.037 16.690 0.000  
##   
## Scales y\*:  
## Estimate Std.Err z-value P(>|z|)  
## SK09\_01 1.000   
## SK09\_02 1.000   
## SK09\_03 1.000   
## SK09\_04 1.000   
## SK09\_05 1.000   
## SK09\_06 1.000   
## SK09\_07 1.000   
## SK09\_08 1.000   
## SK09\_09 1.000   
## SK09\_10 1.000   
## SK09\_11 1.000   
## SK09\_12 1.000   
## SK09\_13 1.000   
## SK09\_14 1.000   
## SK09\_15 1.000   
## SK09\_16 1.000   
##   
## Modification Indices:  
##   
## lhs op rhs mi epc sepc.lv sepc.all sepc.nox  
## 122 SK09\_01 ~~ SK09\_03 6.564 0.097 0.097 0.283 0.283  
## 123 SK09\_01 ~~ SK09\_04 0.015 0.006 0.006 0.012 0.012  
## 124 SK09\_01 ~~ SK09\_05 0.346 -0.029 -0.029 -0.069 -0.069  
## 125 SK09\_01 ~~ SK09\_06 3.322 -0.116 -0.116 -0.203 -0.203  
## 126 SK09\_01 ~~ SK09\_07 0.003 -0.002 -0.002 -0.006 -0.006  
## 127 SK09\_01 ~~ SK09\_08 0.306 0.022 0.022 0.056 0.056  
## 128 SK09\_01 ~~ SK09\_09 1.443 -0.050 -0.050 -0.136 -0.136  
## 129 SK09\_01 ~~ SK09\_10 3.372 -0.106 -0.106 -0.198 -0.198  
## 130 SK09\_01 ~~ SK09\_11 0.195 -0.023 -0.023 -0.046 -0.046  
## 131 SK09\_01 ~~ SK09\_12 3.156 -0.090 -0.090 -0.200 -0.200  
## 132 SK09\_01 ~~ SK09\_13 1.055 -0.059 -0.059 -0.096 -0.096  
## 133 SK09\_01 ~~ SK09\_14 0.168 0.016 0.016 0.044 0.044  
## 134 SK09\_01 ~~ SK09\_15 0.828 0.036 0.036 0.111 0.111  
## 135 SK09\_01 ~~ SK09\_16 0.197 -0.019 -0.019 -0.051 -0.051  
## 136 SK09\_02 ~~ SK09\_03 9.667 0.115 0.115 0.299 0.299  
## 137 SK09\_02 ~~ SK09\_04 0.000 -0.001 -0.001 -0.002 -0.002  
## 138 SK09\_02 ~~ SK09\_05 0.047 -0.010 -0.010 -0.021 -0.021  
## 139 SK09\_02 ~~ SK09\_06 7.283 -0.168 -0.168 -0.264 -0.264  
## 140 SK09\_02 ~~ SK09\_07 4.215 0.072 0.072 0.187 0.187  
## 141 SK09\_02 ~~ SK09\_08 1.638 -0.058 -0.058 -0.128 -0.128  
## 142 SK09\_02 ~~ SK09\_09 0.444 0.029 0.029 0.070 0.070  
## 143 SK09\_02 ~~ SK09\_10 1.329 -0.068 -0.068 -0.114 -0.114  
## 144 SK09\_02 ~~ SK09\_11 2.883 -0.093 -0.093 -0.165 -0.165  
## 145 SK09\_02 ~~ SK09\_12 1.634 -0.067 -0.067 -0.133 -0.133  
## 146 SK09\_02 ~~ SK09\_13 5.929 -0.140 -0.140 -0.206 -0.206  
## 147 SK09\_02 ~~ SK09\_14 1.310 -0.049 -0.049 -0.125 -0.125  
## 148 SK09\_02 ~~ SK09\_15 0.405 -0.029 -0.029 -0.080 -0.080  
## 149 SK09\_02 ~~ SK09\_16 0.238 -0.022 -0.022 -0.052 -0.052  
## 150 SK09\_03 ~~ SK09\_05 1.441 -0.056 -0.056 -0.142 -0.142  
## 151 SK09\_03 ~~ SK09\_06 5.299 -0.153 -0.153 -0.291 -0.291  
## 152 SK09\_03 ~~ SK09\_07 0.952 -0.041 -0.041 -0.130 -0.130  
## 153 SK09\_03 ~~ SK09\_08 1.407 0.045 0.045 0.122 0.122  
## 154 SK09\_03 ~~ SK09\_09 7.130 -0.121 -0.121 -0.358 -0.358  
## 155 SK09\_03 ~~ SK09\_10 0.185 0.024 0.024 0.048 0.048  
## 156 SK09\_03 ~~ SK09\_11 0.337 -0.025 -0.025 -0.054 -0.054  
## 157 SK09\_03 ~~ SK09\_12 1.920 -0.068 -0.068 -0.163 -0.163  
## 158 SK09\_03 ~~ SK09\_13 4.168 0.111 0.111 0.198 0.198  
## 159 SK09\_03 ~~ SK09\_14 0.003 0.002 0.002 0.006 0.006  
## 160 SK09\_03 ~~ SK09\_15 0.001 0.001 0.001 0.003 0.003  
## 161 SK09\_03 ~~ SK09\_16 1.608 -0.055 -0.055 -0.158 -0.158  
## 162 SK09\_04 ~~ SK09\_05 0.563 -0.041 -0.041 -0.075 -0.075  
## 163 SK09\_04 ~~ SK09\_06 2.572 -0.106 -0.106 -0.144 -0.144  
## 164 SK09\_04 ~~ SK09\_07 0.200 -0.022 -0.022 -0.050 -0.050  
## 165 SK09\_04 ~~ SK09\_08 0.169 -0.019 -0.019 -0.037 -0.037  
## 166 SK09\_04 ~~ SK09\_09 1.280 -0.059 -0.059 -0.125 -0.125  
## 167 SK09\_04 ~~ SK09\_10 0.253 0.029 0.029 0.042 0.042  
## 168 SK09\_04 ~~ SK09\_11 2.340 0.078 0.078 0.121 0.121  
## 169 SK09\_04 ~~ SK09\_12 0.109 -0.017 -0.017 -0.029 -0.029  
## 170 SK09\_04 ~~ SK09\_13 7.586 0.147 0.147 0.187 0.187  
## 171 SK09\_04 ~~ SK09\_14 1.042 0.048 0.048 0.105 0.105  
## 172 SK09\_04 ~~ SK09\_15 1.577 0.055 0.055 0.133 0.133  
## 173 SK09\_04 ~~ SK09\_16 1.003 -0.051 -0.051 -0.105 -0.105  
## 174 SK09\_05 ~~ SK09\_06 9.324 0.160 0.160 0.246 0.246  
## 175 SK09\_05 ~~ SK09\_07 0.021 0.006 0.006 0.016 0.016  
## 176 SK09\_05 ~~ SK09\_08 1.284 -0.056 -0.056 -0.122 -0.122  
## 177 SK09\_05 ~~ SK09\_09 2.276 0.060 0.060 0.143 0.143  
## 178 SK09\_05 ~~ SK09\_10 0.247 -0.026 -0.026 -0.043 -0.043  
## 179 SK09\_05 ~~ SK09\_11 1.118 -0.054 -0.054 -0.095 -0.095  
## 180 SK09\_05 ~~ SK09\_12 1.907 0.062 0.062 0.120 0.120  
## 181 SK09\_05 ~~ SK09\_13 5.201 -0.134 -0.134 -0.192 -0.192  
## 182 SK09\_05 ~~ SK09\_14 1.423 -0.054 -0.054 -0.135 -0.135  
## 183 SK09\_05 ~~ SK09\_15 0.418 -0.028 -0.028 -0.078 -0.078  
## 184 SK09\_05 ~~ SK09\_16 3.530 0.072 0.072 0.167 0.167  
## 185 SK09\_06 ~~ SK09\_07 0.029 -0.010 -0.010 -0.019 -0.019  
## 186 SK09\_06 ~~ SK09\_08 0.023 0.009 0.009 0.014 0.014  
## 187 SK09\_06 ~~ SK09\_09 1.850 0.079 0.079 0.141 0.141  
## 188 SK09\_06 ~~ SK09\_10 6.153 0.135 0.135 0.166 0.166  
## 189 SK09\_06 ~~ SK09\_11 0.152 0.022 0.022 0.029 0.029  
## 190 SK09\_06 ~~ SK09\_14 0.016 0.008 0.008 0.015 0.015  
## 191 SK09\_06 ~~ SK09\_15 0.939 -0.061 -0.061 -0.125 -0.125  
## 192 SK09\_06 ~~ SK09\_16 2.833 0.094 0.094 0.162 0.162  
## 193 SK09\_07 ~~ SK09\_08 0.076 0.011 0.011 0.029 0.029  
## 194 SK09\_07 ~~ SK09\_09 0.894 0.033 0.033 0.096 0.096  
## 195 SK09\_07 ~~ SK09\_10 0.020 0.007 0.007 0.014 0.014  
## 196 SK09\_07 ~~ SK09\_11 1.858 -0.066 -0.066 -0.143 -0.143  
## 197 SK09\_07 ~~ SK09\_12 0.086 0.013 0.013 0.031 0.031  
## 198 SK09\_07 ~~ SK09\_13 2.680 -0.089 -0.089 -0.158 -0.158  
## 199 SK09\_07 ~~ SK09\_14 0.899 -0.033 -0.033 -0.102 -0.102  
## 200 SK09\_07 ~~ SK09\_15 0.573 0.027 0.027 0.092 0.092  
## 201 SK09\_07 ~~ SK09\_16 2.206 -0.061 -0.061 -0.173 -0.173  
## 202 SK09\_08 ~~ SK09\_09 0.054 0.009 0.009 0.024 0.024  
## 203 SK09\_08 ~~ SK09\_10 1.073 0.053 0.053 0.092 0.092  
## 204 SK09\_08 ~~ SK09\_11 1.722 0.060 0.060 0.112 0.112  
## 205 SK09\_08 ~~ SK09\_13 0.118 0.020 0.020 0.030 0.030  
## 206 SK09\_08 ~~ SK09\_14 0.006 0.003 0.003 0.008 0.008  
## 207 SK09\_08 ~~ SK09\_15 2.547 -0.065 -0.065 -0.188 -0.188  
## 208 SK09\_08 ~~ SK09\_16 0.000 0.000 0.000 0.001 0.001  
## 209 SK09\_09 ~~ SK09\_10 0.047 0.011 0.011 0.021 0.021  
## 210 SK09\_09 ~~ SK09\_11 0.513 -0.032 -0.032 -0.065 -0.065  
## 211 SK09\_09 ~~ SK09\_12 4.260 0.085 0.085 0.192 0.192  
## 212 SK09\_09 ~~ SK09\_13 0.329 -0.032 -0.032 -0.054 -0.054  
## 213 SK09\_09 ~~ SK09\_14 0.039 -0.007 -0.007 -0.021 -0.021  
## 214 SK09\_09 ~~ SK09\_15 5.942 -0.094 -0.094 -0.297 -0.297  
## 215 SK09\_09 ~~ SK09\_16 4.137 0.074 0.074 0.199 0.199  
## 216 SK09\_10 ~~ SK09\_12 1.949 0.068 0.068 0.106 0.106  
## 217 SK09\_10 ~~ SK09\_13 0.416 0.034 0.034 0.038 0.038  
## 218 SK09\_10 ~~ SK09\_14 0.522 0.036 0.036 0.071 0.071  
## 219 SK09\_10 ~~ SK09\_15 0.757 -0.047 -0.047 -0.104 -0.104  
## 220 SK09\_10 ~~ SK09\_16 2.044 -0.072 -0.072 -0.133 -0.133  
## 221 SK09\_11 ~~ SK09\_12 0.173 0.020 0.020 0.033 0.033  
## 222 SK09\_11 ~~ SK09\_13 5.222 0.124 0.124 0.152 0.152  
## 223 SK09\_11 ~~ SK09\_14 5.316 0.100 0.100 0.211 0.211  
## 224 SK09\_11 ~~ SK09\_15 0.304 0.026 0.026 0.059 0.059  
## 225 SK09\_11 ~~ SK09\_16 0.918 -0.049 -0.049 -0.096 -0.096  
## 226 SK09\_12 ~~ SK09\_13 1.942 -0.080 -0.080 -0.109 -0.109  
## 227 SK09\_12 ~~ SK09\_14 0.147 -0.017 -0.017 -0.040 -0.040  
## 228 SK09\_12 ~~ SK09\_15 0.047 -0.010 -0.010 -0.025 -0.025  
## 229 SK09\_13 ~~ SK09\_14 1.908 0.076 0.076 0.133 0.133  
## 230 SK09\_13 ~~ SK09\_15 3.017 0.097 0.097 0.186 0.186  
## 231 SK09\_13 ~~ SK09\_16 2.463 -0.087 -0.087 -0.140 -0.140  
## 232 SK09\_14 ~~ SK09\_15 0.724 0.028 0.028 0.092 0.092  
## 233 SK09\_14 ~~ SK09\_16 0.766 -0.037 -0.037 -0.102 -0.102  
## 234 SK09\_15 ~~ SK09\_16 2.414 0.058 0.058 0.178 0.178

ds$rz <- lavPredict(fitrz)

## CFA of need of cognitive closure

# NCC  
  
ncc <- '  
  
ncc =~ SK07\_01 + SK07\_02 + SK07\_03 + SK07\_04 + SK07\_05 + SK07\_06 + SK07\_07 + SK07\_08 + SK07\_09 + SK07\_10 + SK07\_11 + SK07\_12 + SK07\_13 + SK07\_14 + SK07\_15  
  
# Residuals  
SK07\_03 ~~ SK07\_12  
SK07\_01 ~~ SK07\_12  
SK07\_01 ~~ SK07\_15  
SK07\_03 ~~ SK07\_04  
SK07\_03 ~~ SK07\_06  
SK07\_03 ~~ SK07\_09  
SK07\_03 ~~ SK07\_10  
SK07\_03 ~~ SK07\_13  
SK07\_04 ~~ SK07\_09  
SK07\_04 ~~ SK07\_12  
SK07\_07 ~~ SK07\_08  
SK07\_08 ~~ SK07\_13  
SK07\_10 ~~ SK07\_11  
SK07\_12 ~~ SK07\_13  
SK07\_08 ~~ SK07\_12  
'  
  
fitncc <- cfa(ncc, data=ds, ordered=c("SK07\_01", "SK07\_02", "SK07\_03", "SK07\_04", "SK07\_05", "SK07\_06", "SK07\_07",  
 "SK07\_08", "SK07\_09", "SK07\_10", "SK07\_11", "SK07\_12", "SK07\_13", "SK07\_14",  
 "SK07\_15"))  
  
summary(fitncc, fit.measures=T, modindices=T)

## lavaan 0.6-3 ended normally after 25 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 75  
##   
## Number of observations 386  
##   
## Estimator DWLS Robust  
## Model Fit Test Statistic 194.706 212.888  
## Degrees of freedom 75 75  
## P-value (Chi-square) 0.000 0.000  
## Scaling correction factor 1.014  
## Shift parameter 20.784  
## for simple second-order correction (Mplus variant)  
##   
## Model test baseline model:  
##   
## Minimum Function Test Statistic 3837.946 2332.079  
## Degrees of freedom 105 105  
## P-value 0.000 0.000  
##   
## User model versus baseline model:  
##   
## Comparative Fit Index (CFI) 0.968 0.938  
## Tucker-Lewis Index (TLI) 0.955 0.913  
##   
## Robust Comparative Fit Index (CFI) NA  
## Robust Tucker-Lewis Index (TLI) NA  
##   
## Root Mean Square Error of Approximation:  
##   
## RMSEA 0.064 0.069  
## 90 Percent Confidence Interval 0.053 0.076 0.058 0.080  
## P-value RMSEA <= 0.05 0.017 0.002  
##   
## Robust RMSEA NA  
## 90 Percent Confidence Interval NA NA  
##   
## Standardized Root Mean Square Residual:  
##   
## SRMR 0.065 0.065  
##   
## Parameter Estimates:  
##   
## Information Expected  
## Information saturated (h1) model Unstructured  
## Standard Errors Robust.sem  
##   
## Latent Variables:  
## Estimate Std.Err z-value P(>|z|)  
## ncc =~   
## SK07\_01 1.000   
## SK07\_02 0.815 0.097 8.371 0.000  
## SK07\_03 0.766 0.093 8.219 0.000  
## SK07\_04 0.975 0.107 9.158 0.000  
## SK07\_05 0.562 0.091 6.147 0.000  
## SK07\_06 0.639 0.086 7.384 0.000  
## SK07\_07 0.326 0.098 3.328 0.001  
## SK07\_08 0.015 0.088 0.174 0.862  
## SK07\_09 0.815 0.093 8.745 0.000  
## SK07\_10 0.877 0.093 9.407 0.000  
## SK07\_11 0.854 0.093 9.174 0.000  
## SK07\_12 0.533 0.089 5.981 0.000  
## SK07\_13 0.505 0.088 5.764 0.000  
## SK07\_14 0.385 0.090 4.299 0.000  
## SK07\_15 1.151 0.087 13.299 0.000  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK07\_03 ~~   
## .SK07\_12 0.564 0.038 14.988 0.000  
## .SK07\_01 ~~   
## .SK07\_12 -0.003 0.040 -0.083 0.934  
## .SK07\_15 0.044 0.044 0.994 0.320  
## .SK07\_03 ~~   
## .SK07\_04 -0.167 0.047 -3.569 0.000  
## .SK07\_06 -0.192 0.042 -4.603 0.000  
## .SK07\_09 -0.150 0.043 -3.518 0.000  
## .SK07\_10 -0.145 0.043 -3.390 0.001  
## .SK07\_13 0.498 0.043 11.463 0.000  
## .SK07\_04 ~~   
## .SK07\_09 0.125 0.039 3.227 0.001  
## .SK07\_12 -0.095 0.045 -2.103 0.035  
## .SK07\_07 ~~   
## .SK07\_08 0.364 0.048 7.608 0.000  
## .SK07\_08 ~~   
## .SK07\_13 0.403 0.041 9.909 0.000  
## .SK07\_10 ~~   
## .SK07\_11 0.152 0.040 3.791 0.000  
## .SK07\_12 ~~   
## .SK07\_13 0.537 0.040 13.457 0.000  
## .SK07\_08 ~~   
## .SK07\_12 0.339 0.042 8.025 0.000  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|)  
## .SK07\_01 0.000   
## .SK07\_02 0.000   
## .SK07\_03 0.000   
## .SK07\_04 0.000   
## .SK07\_05 0.000   
## .SK07\_06 0.000   
## .SK07\_07 0.000   
## .SK07\_08 0.000   
## .SK07\_09 0.000   
## .SK07\_10 0.000   
## .SK07\_11 0.000   
## .SK07\_12 0.000   
## .SK07\_13 0.000   
## .SK07\_14 0.000   
## .SK07\_15 0.000   
## ncc 0.000   
##   
## Thresholds:  
## Estimate Std.Err z-value P(>|z|)  
## SK07\_01|t1 -1.903 0.130 -14.634 0.000  
## SK07\_01|t2 -0.853 0.073 -11.670 0.000  
## SK07\_01|t3 0.583 0.068 8.582 0.000  
## SK07\_02|t1 -1.581 0.103 -15.305 0.000  
## SK07\_02|t2 -0.058 0.064 -0.915 0.360  
## SK07\_02|t3 1.192 0.083 14.287 0.000  
## SK07\_03|t1 -1.706 0.112 -15.194 0.000  
## SK07\_03|t2 -0.435 0.066 -6.586 0.000  
## SK07\_03|t3 0.981 0.076 12.843 0.000  
## SK07\_04|t1 -1.706 0.112 -15.194 0.000  
## SK07\_04|t2 -0.530 0.067 -7.887 0.000  
## SK07\_04|t3 0.891 0.074 12.038 0.000  
## SK07\_05|t1 -1.206 0.084 -14.357 0.000  
## SK07\_05|t2 0.297 0.065 4.568 0.000  
## SK07\_05|t3 1.369 0.091 15.014 0.000  
## SK07\_06|t1 -1.321 0.089 -14.861 0.000  
## SK07\_06|t2 0.039 0.064 0.610 0.542  
## SK07\_06|t3 1.219 0.085 14.426 0.000  
## SK07\_07|t1 -2.419 0.209 -11.552 0.000  
## SK07\_07|t2 -1.104 0.080 -13.760 0.000  
## SK07\_07|t3 0.493 0.067 7.388 0.000  
## SK07\_08|t1 -2.313 0.188 -12.321 0.000  
## SK07\_08|t2 -1.247 0.086 -14.560 0.000  
## SK07\_08|t3 0.358 0.065 5.478 0.000  
## SK07\_09|t1 -1.558 0.102 -15.303 0.000  
## SK07\_09|t2 -0.443 0.066 -6.686 0.000  
## SK07\_09|t3 0.754 0.071 10.631 0.000  
## SK07\_10|t1 -1.321 0.089 -14.861 0.000  
## SK07\_10|t2 0.085 0.064 1.322 0.186  
## SK07\_10|t3 1.290 0.088 14.746 0.000  
## SK07\_11|t1 -1.678 0.110 -15.235 0.000  
## SK07\_11|t2 -0.583 0.068 -8.582 0.000  
## SK07\_11|t3 1.013 0.077 13.101 0.000  
## SK07\_12|t1 -1.865 0.126 -14.777 0.000  
## SK07\_12|t2 -0.780 0.071 -10.918 0.000  
## SK07\_12|t3 0.754 0.071 10.631 0.000  
## SK07\_13|t1 -2.228 0.173 -12.900 0.000  
## SK07\_13|t2 -1.093 0.080 -13.681 0.000  
## SK07\_13|t3 0.538 0.067 7.986 0.000  
## SK07\_14|t1 -0.961 0.076 -12.667 0.000  
## SK07\_14|t2 0.771 0.071 10.822 0.000  
## SK07\_14|t3 1.628 0.106 15.287 0.000  
## SK07\_15|t1 -1.764 0.117 -15.075 0.000  
## SK07\_15|t2 -0.576 0.068 -8.483 0.000  
## SK07\_15|t3 0.900 0.074 12.129 0.000  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|)  
## .SK07\_01 0.636   
## .SK07\_02 0.759   
## .SK07\_03 0.787   
## .SK07\_04 0.654   
## .SK07\_05 0.885   
## .SK07\_06 0.852   
## .SK07\_07 0.961   
## .SK07\_08 1.000   
## .SK07\_09 0.759   
## .SK07\_10 0.720   
## .SK07\_11 0.735   
## .SK07\_12 0.897   
## .SK07\_13 0.907   
## .SK07\_14 0.946   
## .SK07\_15 0.518   
## ncc 0.364 0.051 7.079 0.000  
##   
## Scales y\*:  
## Estimate Std.Err z-value P(>|z|)  
## SK07\_01 1.000   
## SK07\_02 1.000   
## SK07\_03 1.000   
## SK07\_04 1.000   
## SK07\_05 1.000   
## SK07\_06 1.000   
## SK07\_07 1.000   
## SK07\_08 1.000   
## SK07\_09 1.000   
## SK07\_10 1.000   
## SK07\_11 1.000   
## SK07\_12 1.000   
## SK07\_13 1.000   
## SK07\_14 1.000   
## SK07\_15 1.000   
##   
## Modification Indices:  
##   
## lhs op rhs mi epc sepc.lv sepc.all sepc.nox  
## 123 SK07\_01 ~~ SK07\_02 2.381 0.087 0.087 0.125 0.125  
## 124 SK07\_01 ~~ SK07\_03 1.057 0.054 0.054 0.076 0.076  
## 125 SK07\_01 ~~ SK07\_04 0.188 0.025 0.025 0.039 0.039  
## 126 SK07\_01 ~~ SK07\_05 3.764 -0.114 -0.114 -0.152 -0.152  
## 127 SK07\_01 ~~ SK07\_06 0.000 -0.001 -0.001 -0.001 -0.001  
## 128 SK07\_01 ~~ SK07\_07 0.176 0.024 0.024 0.031 0.031  
## 129 SK07\_01 ~~ SK07\_08 0.438 0.039 0.039 0.049 0.049  
## 130 SK07\_01 ~~ SK07\_09 0.431 -0.037 -0.037 -0.053 -0.053  
## 131 SK07\_01 ~~ SK07\_10 0.338 -0.032 -0.032 -0.048 -0.048  
## 132 SK07\_01 ~~ SK07\_11 0.800 -0.050 -0.050 -0.073 -0.073  
## 133 SK07\_01 ~~ SK07\_13 7.068 0.134 0.134 0.176 0.176  
## 134 SK07\_01 ~~ SK07\_14 11.657 -0.196 -0.196 -0.253 -0.253  
## 135 SK07\_02 ~~ SK07\_03 1.036 -0.058 -0.058 -0.075 -0.075  
## 136 SK07\_02 ~~ SK07\_04 0.395 0.036 0.036 0.051 0.051  
## 137 SK07\_02 ~~ SK07\_05 0.046 0.012 0.012 0.014 0.014  
## 138 SK07\_02 ~~ SK07\_06 0.045 -0.012 -0.012 -0.015 -0.015  
## 139 SK07\_02 ~~ SK07\_07 0.141 -0.022 -0.022 -0.026 -0.026  
## 140 SK07\_02 ~~ SK07\_08 0.000 -0.001 -0.001 -0.001 -0.001  
## 141 SK07\_02 ~~ SK07\_09 0.358 0.034 0.034 0.045 0.045  
## 142 SK07\_02 ~~ SK07\_10 0.621 -0.042 -0.042 -0.057 -0.057  
## 143 SK07\_02 ~~ SK07\_11 7.064 0.139 0.139 0.186 0.186  
## 144 SK07\_02 ~~ SK07\_12 0.276 -0.029 -0.029 -0.035 -0.035  
## 145 SK07\_02 ~~ SK07\_13 0.221 -0.027 -0.027 -0.032 -0.032  
## 146 SK07\_02 ~~ SK07\_14 0.377 0.032 0.032 0.038 0.038  
## 147 SK07\_02 ~~ SK07\_15 8.272 -0.169 -0.169 -0.269 -0.269  
## 148 SK07\_03 ~~ SK07\_05 0.388 -0.035 -0.035 -0.042 -0.042  
## 149 SK07\_03 ~~ SK07\_07 12.989 0.195 0.195 0.225 0.225  
## 150 SK07\_03 ~~ SK07\_08 39.370 0.314 0.314 0.354 0.354  
## 151 SK07\_03 ~~ SK07\_11 0.142 -0.022 -0.022 -0.029 -0.029  
## 152 SK07\_03 ~~ SK07\_14 0.307 0.030 0.030 0.035 0.035  
## 153 SK07\_03 ~~ SK07\_15 1.203 -0.066 -0.066 -0.104 -0.104  
## 154 SK07\_04 ~~ SK07\_05 0.139 -0.021 -0.021 -0.028 -0.028  
## 155 SK07\_04 ~~ SK07\_06 0.460 0.037 0.037 0.050 0.050  
## 156 SK07\_04 ~~ SK07\_07 3.326 -0.109 -0.109 -0.138 -0.138  
## 157 SK07\_04 ~~ SK07\_08 7.214 -0.153 -0.153 -0.189 -0.189  
## 158 SK07\_04 ~~ SK07\_10 0.119 -0.020 -0.020 -0.029 -0.029  
## 159 SK07\_04 ~~ SK07\_11 2.306 -0.088 -0.088 -0.127 -0.127  
## 160 SK07\_04 ~~ SK07\_13 2.142 -0.086 -0.086 -0.111 -0.111  
## 161 SK07\_04 ~~ SK07\_14 0.043 0.011 0.011 0.014 0.014  
## 162 SK07\_04 ~~ SK07\_15 1.866 0.075 0.075 0.128 0.128  
## 163 SK07\_05 ~~ SK07\_06 0.922 0.051 0.051 0.058 0.058  
## 164 SK07\_05 ~~ SK07\_07 3.313 -0.103 -0.103 -0.112 -0.112  
## 165 SK07\_05 ~~ SK07\_08 7.389 -0.154 -0.154 -0.164 -0.164  
## 166 SK07\_05 ~~ SK07\_09 1.301 0.060 0.060 0.073 0.073  
## 167 SK07\_05 ~~ SK07\_10 0.022 0.007 0.007 0.009 0.009  
## 168 SK07\_05 ~~ SK07\_11 1.481 0.064 0.064 0.080 0.080  
## 169 SK07\_05 ~~ SK07\_12 0.495 -0.039 -0.039 -0.044 -0.044  
## 170 SK07\_05 ~~ SK07\_13 0.023 0.008 0.008 0.009 0.009  
## 171 SK07\_05 ~~ SK07\_14 2.408 0.081 0.081 0.088 0.088  
## 172 SK07\_05 ~~ SK07\_15 0.008 -0.005 -0.005 -0.008 -0.008  
## 173 SK07\_06 ~~ SK07\_07 4.884 -0.118 -0.118 -0.131 -0.131  
## 174 SK07\_06 ~~ SK07\_08 0.000 0.001 0.001 0.001 0.001  
## 175 SK07\_06 ~~ SK07\_09 0.015 0.006 0.006 0.008 0.008  
## 176 SK07\_06 ~~ SK07\_10 6.216 0.126 0.126 0.161 0.161  
## 177 SK07\_06 ~~ SK07\_11 1.149 -0.057 -0.057 -0.072 -0.072  
## 178 SK07\_06 ~~ SK07\_12 0.117 -0.018 -0.018 -0.021 -0.021  
## 179 SK07\_06 ~~ SK07\_13 6.247 -0.141 -0.141 -0.160 -0.160  
## 180 SK07\_06 ~~ SK07\_14 0.054 0.012 0.012 0.013 0.013  
## 181 SK07\_06 ~~ SK07\_15 0.190 -0.023 -0.023 -0.035 -0.035  
## 182 SK07\_07 ~~ SK07\_09 0.879 -0.054 -0.054 -0.063 -0.063  
## 183 SK07\_07 ~~ SK07\_10 4.939 -0.126 -0.126 -0.152 -0.152  
## 184 SK07\_07 ~~ SK07\_11 0.545 0.042 0.042 0.050 0.050  
## 185 SK07\_07 ~~ SK07\_12 8.506 0.138 0.138 0.149 0.149  
## 186 SK07\_07 ~~ SK07\_13 14.765 0.211 0.211 0.226 0.226  
## 187 SK07\_07 ~~ SK07\_14 3.313 0.103 0.103 0.108 0.108  
## 188 SK07\_07 ~~ SK07\_15 1.528 -0.076 -0.076 -0.108 -0.108  
## 189 SK07\_08 ~~ SK07\_09 0.161 0.022 0.022 0.025 0.025  
## 190 SK07\_08 ~~ SK07\_10 10.343 -0.177 -0.177 -0.208 -0.208  
## 191 SK07\_08 ~~ SK07\_11 3.327 0.101 0.101 0.118 0.118  
## 192 SK07\_08 ~~ SK07\_14 0.349 0.032 0.032 0.033 0.033  
## 193 SK07\_08 ~~ SK07\_15 1.188 -0.064 -0.064 -0.089 -0.089  
## 194 SK07\_09 ~~ SK07\_10 0.014 0.007 0.007 0.009 0.009  
## 195 SK07\_09 ~~ SK07\_11 1.270 0.060 0.060 0.081 0.081  
## 196 SK07\_09 ~~ SK07\_12 0.850 -0.049 -0.049 -0.059 -0.059  
## 197 SK07\_09 ~~ SK07\_13 3.896 -0.109 -0.109 -0.131 -0.131  
## 198 SK07\_09 ~~ SK07\_14 2.563 -0.083 -0.083 -0.098 -0.098  
## 199 SK07\_09 ~~ SK07\_15 0.461 0.038 0.038 0.060 0.060  
## 200 SK07\_10 ~~ SK07\_12 1.660 -0.071 -0.071 -0.089 -0.089  
## 201 SK07\_10 ~~ SK07\_13 1.115 -0.057 -0.057 -0.071 -0.071  
## 202 SK07\_10 ~~ SK07\_14 0.207 0.024 0.024 0.030 0.030  
## 203 SK07\_10 ~~ SK07\_15 1.442 0.063 0.063 0.103 0.103  
## 204 SK07\_11 ~~ SK07\_12 0.000 -0.001 -0.001 -0.001 -0.001  
## 205 SK07\_11 ~~ SK07\_13 0.004 0.004 0.004 0.004 0.004  
## 206 SK07\_11 ~~ SK07\_14 0.031 0.009 0.009 0.011 0.011  
## 207 SK07\_11 ~~ SK07\_15 0.732 -0.046 -0.046 -0.074 -0.074  
## 208 SK07\_12 ~~ SK07\_14 1.485 0.066 0.066 0.071 0.071  
## 209 SK07\_12 ~~ SK07\_15 1.467 0.067 0.067 0.099 0.099  
## 210 SK07\_13 ~~ SK07\_14 0.050 0.012 0.012 0.013 0.013  
## 211 SK07\_13 ~~ SK07\_15 1.378 0.061 0.061 0.089 0.089  
## 212 SK07\_14 ~~ SK07\_15 0.164 0.022 0.022 0.031 0.031

ds$ncc <- lavPredict(fitncc)

# Mean centering and interaction terms

Before performing hypothesis testing, I mean centred all predictors (RZ, NCC & CN).

# Mean Centering RZ, NCC & ColNar  
  
ds <- mutate(ds, rz\_mc = rz - mean(rz),  
 ncc\_mc = ncc - mean(ncc),  
 cn\_mc = cn - mean(cn))

…and making interaction terms..

# Making interaction terms  
ds <- mutate(ds, halalXcoo = halal\*coo,  
 halalXncc = halal\*ncc\_mc,  
 halalXrz = halal\*rz\_mc,  
 halalXcn = halal\*cn\_mc,  
 cooXncc = coo\*ncc\_mc,  
 cooXrz = coo\*rz\_mc,  
 cooXcn = coo\*cn\_mc,  
 halalXcooXncc = halal\*coo\*ncc\_mc,  
 halalXcooXrz = halal\*coo\*rz\_mc,  
 halalXcooXcn = halal\*coo\*cn\_mc)

# Hypothesis testing

We are then performing hypothesis testing using **SEM-based path analysis** (with ML as the estimator). I **bootstrapped** the sample to 1000 iterations. As for the interaction plot, I’m currently trying to figure out how to make it nicely in R.

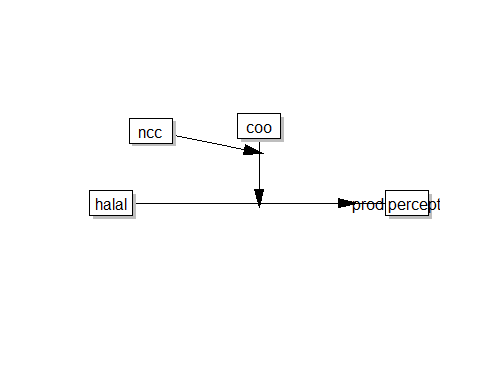
## Need for cognitive closure

Here is the plot

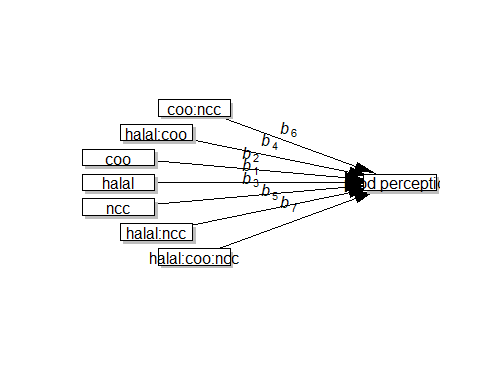
# Plot  
library(processR)

## Warning: package 'processR' was built under R version 3.5.2

labels=list(X="halal", W="coo", Z="ncc", Y="prod perception")  
pmacroModel(3,labels=labels)



statisticalDiagram(3,labels=labels)



This is the result of path analysis (using lavaan, when the dv is **product perception of cake** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

mod1ncc <- '  
# Model  
pp\_cake ~ b1\*halal + b2\*coo + b3\*ncc\_mc + b4\*halalXcoo + b5\*halalXncc + b6\*cooXncc + b7\*halalXcooXncc + DM02\_01 + DM01  
  
  
# NCC Mean and Variances  
ncc\_mc ~ a\*1  
ncc\_mc ~~ b\*ncc\_mc   
  
# Conditional effect  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high NCC  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turkey, low NCC  
cond3:= b1 + b5\*(a+sqrt(b)) # English, high NCC  
cond4:= b1 + b5\*(a-sqrt(b)) # English, low NCC  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high NCC  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low NCC  
cond7:= b5\*(a+sqrt(b)) # England, high NCC  
cond8 := b5\*(a-sqrt(b)) # England, low NCC  
  
  
'  
  
fitmod1ncc <- sem(mod1ncc, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = mod1ncc, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

summary(fitmod1ncc, standardized=T)

## lavaan 0.6-3 ended normally after 212 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 569.712  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_cake ~   
## halal (b1) 2.786 2.089 1.334 0.182 2.786 0.091  
## coo (b2) 3.472 2.359 1.471 0.141 3.472 0.113  
## ncc\_mc (b3) -1.285 3.569 -0.360 0.719 -1.285 -0.045  
## halalXcoo (b4) -0.393 3.180 -0.124 0.902 -0.393 -0.011  
## halalXncc (b5) 7.557 4.205 1.797 0.072 7.557 0.184  
## cooXncc (b6) 1.765 4.665 0.378 0.705 1.765 0.044  
## hllXcXncc (b7) -3.060 6.276 -0.488 0.626 -3.060 -0.051  
## DM02\_01 -0.071 0.139 -0.514 0.607 -0.071 -0.027  
## DM01 0.967 2.493 0.388 0.698 0.967 0.023  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.013 -0.352 0.725 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.494 0.000 0.124 0.577  
## halalXncc 0.005 0.009 0.566 0.571 0.005 0.029  
## cooXncc 0.013 0.009 1.387 0.166 0.013 0.067  
## halalXcooXncc 0.012 0.006 1.872 0.061 0.012 0.094  
## DM02\_01 0.030 0.145 0.204 0.838 0.030 0.010  
## DM01 -0.002 0.009 -0.218 0.827 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.405 0.000 0.122 0.565  
## halalXncc 0.019 0.009 1.952 0.051 0.019 0.099  
## cooXncc 0.011 0.010 1.142 0.254 0.011 0.057  
## halalXcooXncc 0.012 0.006 1.865 0.062 0.012 0.092  
## DM02\_01 -0.044 0.156 -0.284 0.777 -0.044 -0.015  
## DM01 -0.003 0.009 -0.319 0.750 -0.003 -0.016  
## halalXcoo ~~   
## halalXncc 0.021 0.010 2.073 0.038 0.021 0.132  
## cooXncc 0.018 0.010 1.828 0.068 0.018 0.111  
## halalXcooXncc 0.018 0.010 1.874 0.061 0.018 0.163  
## DM02\_01 -0.043 0.119 -0.360 0.719 -0.043 -0.017  
## DM01 -0.004 0.008 -0.474 0.635 -0.004 -0.024  
## halalXncc ~~   
## cooXncc 0.066 0.012 5.364 0.000 0.066 0.461  
## halalXcooXncc 0.066 0.012 5.372 0.000 0.066 0.690  
## DM02\_01 -0.039 0.132 -0.295 0.768 -0.039 -0.018  
## DM01 0.005 0.009 0.546 0.585 0.005 0.035  
## cooXncc ~~   
## halalXcooXncc 0.066 0.012 5.357 0.000 0.066 0.668  
## DM02\_01 0.020 0.133 0.154 0.878 0.020 0.009  
## DM01 0.002 0.008 0.187 0.851 0.002 0.011  
## halalXcooXncc ~~   
## DM02\_01 -0.020 0.101 -0.193 0.847 -0.020 -0.013  
## DM01 0.001 0.006 0.106 0.915 0.001 0.007  
## DM02\_01 ~~   
## DM01 -0.382 0.137 -2.776 0.005 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## ncc\_mc (a) -0.000 0.027 -0.005 0.996 -0.000 -0.000  
## .pp\_cake -3.333 6.404 -0.520 0.603 -3.333 -0.218  
## halal 0.496 0.025 19.548 0.000 0.496 0.992  
## coo 0.506 0.025 19.975 0.000 0.506 1.013  
## halalXcoo 0.247 0.023 10.883 0.000 0.247 0.572  
## halalXncc 0.011 0.019 0.565 0.572 0.011 0.028  
## cooXncc 0.022 0.019 1.135 0.256 0.022 0.058  
## hallXcXncc 0.024 0.013 1.853 0.064 0.024 0.093  
## DM02\_01 21.595 0.293 73.818 0.000 21.595 3.697  
## DM01 1.847 0.018 102.418 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## ncc\_mc (b) 0.287 0.022 12.752 0.000 0.287 1.000  
## .pp\_cake 221.151 16.004 13.819 0.000 221.151 0.944  
## halal 0.250 0.001 269.981 0.000 0.250 1.000  
## coo 0.250 0.001 271.251 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 16.216 0.000 0.186 1.000  
## halalXncc 0.139 0.016 8.471 0.000 0.139 1.000  
## cooXncc 0.147 0.019 7.752 0.000 0.147 1.000  
## hallXcXncc 0.066 0.012 5.366 0.000 0.066 1.000  
## DM02\_01 34.111 5.433 6.279 0.000 34.111 1.000  
## DM01 0.130 0.012 10.399 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 6.438 3.236 1.989 0.047 6.438 0.264  
## cond2 -1.653 3.225 -0.513 0.608 -1.653 -0.104  
## cond3 6.831 3.311 2.063 0.039 6.831 0.275  
## cond4 -1.260 2.836 -0.444 0.657 -1.260 -0.093  
## cond5 3.651 3.725 0.980 0.327 3.651 0.173  
## cond6 -4.440 4.067 -1.092 0.275 -4.440 -0.195  
## cond7 4.044 2.283 1.772 0.076 4.044 0.184  
## cond8 -4.047 2.243 -1.804 0.071 -4.047 -0.184

This is the result of path analysis (using lavaan, when the dv is **product perception of beverage** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

# dv=prod perception \*\*BEVERAGE\*\* iv=halal, coo, ncc  
  
model2ncc <- '  
# Model  
pp\_bev ~ b1\*halal + b2\*coo + b3\*ncc\_mc + b4\*halalXcoo + b5\*halalXncc + b6\*cooXncc + b7\*halalXcooXncc + DM02\_01 + DM01  
  
  
# NCC Mean and Variances  
ncc\_mc ~ a\*1  
ncc\_mc ~~ b\*ncc\_mc   
  
# Conditional effect (Halal)  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high NCC  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turley, low NCC  
cond3:= b1 + b5\*(a+sqrt(b)) # England, high NCC  
cond4:= b1 + b5\*(a-sqrt(b)) # England, low NCC  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high NCC  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low NCC  
cond7:= b5\*(a+sqrt(b)) # England, high NCC  
cond8 := b5\*(a-sqrt(b)) # England, low NCC  
  
# Residual  
  
pp\_bev ~~ pp\_bev  
  
'  
  
fitmod2ncc <- sem(model2ncc, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = model2ncc, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

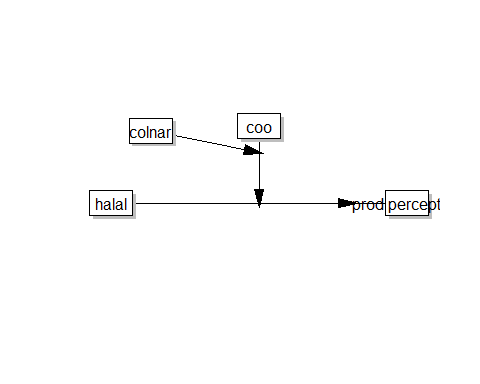
summary(fitmod2ncc, standardized=T)

## lavaan 0.6-3 ended normally after 210 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 569.712  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_bev ~   
## halal (b1) 1.383 2.577 0.537 0.592 1.383 0.040  
## coo (b2) -0.222 2.515 -0.088 0.930 -0.222 -0.006  
## ncc\_mc (b3) 1.937 4.016 0.482 0.630 1.937 0.060  
## halalXcoo (b4) 5.115 3.445 1.485 0.138 5.115 0.128  
## halalXncc (b5) 2.923 5.310 0.550 0.582 2.923 0.064  
## cooXncc (b6) 4.582 5.070 0.904 0.366 4.582 0.102  
## hllXcXncc (b7) -3.847 7.193 -0.535 0.593 -3.847 -0.057  
## DM02\_01 0.079 0.159 0.498 0.619 0.079 0.027  
## DM01 -1.302 2.657 -0.490 0.624 -1.302 -0.027  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.013 -0.336 0.737 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.235 0.000 0.124 0.577  
## halalXncc 0.005 0.009 0.578 0.564 0.005 0.029  
## cooXncc 0.013 0.010 1.332 0.183 0.013 0.067  
## halalXcooXncc 0.012 0.007 1.818 0.069 0.012 0.094  
## DM02\_01 0.030 0.151 0.196 0.845 0.030 0.010  
## DM01 -0.002 0.010 -0.198 0.843 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.249 0.000 0.122 0.565  
## halalXncc 0.019 0.010 1.929 0.054 0.019 0.099  
## cooXncc 0.011 0.010 1.106 0.269 0.011 0.057  
## halalXcooXncc 0.012 0.007 1.808 0.071 0.012 0.092  
## DM02\_01 -0.044 0.148 -0.298 0.766 -0.044 -0.015  
## DM01 -0.003 0.009 -0.327 0.744 -0.003 -0.016  
## halalXcoo ~~   
## halalXncc 0.021 0.011 2.008 0.045 0.021 0.132  
## cooXncc 0.018 0.010 1.766 0.077 0.018 0.111  
## halalXcooXncc 0.018 0.010 1.820 0.069 0.018 0.163  
## DM02\_01 -0.043 0.124 -0.345 0.730 -0.043 -0.017  
## DM01 -0.004 0.008 -0.449 0.653 -0.004 -0.024  
## halalXncc ~~   
## cooXncc 0.066 0.013 5.069 0.000 0.066 0.461  
## halalXcooXncc 0.066 0.013 5.073 0.000 0.066 0.690  
## DM02\_01 -0.039 0.132 -0.294 0.769 -0.039 -0.018  
## DM01 0.005 0.009 0.559 0.576 0.005 0.035  
## cooXncc ~~   
## halalXcooXncc 0.066 0.013 5.078 0.000 0.066 0.668  
## DM02\_01 0.020 0.126 0.162 0.871 0.020 0.009  
## DM01 0.002 0.008 0.196 0.845 0.002 0.011  
## halalXcooXncc ~~   
## DM02\_01 -0.020 0.099 -0.198 0.843 -0.020 -0.013  
## DM01 0.001 0.006 0.109 0.914 0.001 0.007  
## DM02\_01 ~~   
## DM01 -0.382 0.140 -2.716 0.007 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## ncc\_mc (a) -0.000 0.027 -0.005 0.996 -0.000 -0.000  
## .pp\_bev -1.180 6.760 -0.175 0.861 -1.180 -0.069  
## halal 0.496 0.024 20.312 0.000 0.496 0.992  
## coo 0.506 0.026 19.386 0.000 0.506 1.013  
## halalXcoo 0.247 0.022 11.053 0.000 0.247 0.572  
## halalXncc 0.011 0.018 0.574 0.566 0.011 0.028  
## cooXncc 0.022 0.020 1.104 0.270 0.022 0.058  
## hallXcXncc 0.024 0.013 1.801 0.072 0.024 0.093  
## DM02\_01 21.595 0.306 70.575 0.000 21.595 3.697  
## DM01 1.847 0.018 100.629 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## ncc\_mc (b) 0.287 0.024 12.181 0.000 0.287 1.000  
## .pp\_bev 282.313 18.136 15.567 0.000 282.313 0.958  
## halal 0.250 0.001 302.360 0.000 0.250 1.000  
## coo 0.250 0.001 251.296 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 16.508 0.000 0.186 1.000  
## halalXncc 0.139 0.017 8.150 0.000 0.139 1.000  
## cooXncc 0.147 0.020 7.538 0.000 0.147 1.000  
## hallXcXncc 0.066 0.013 5.078 0.000 0.066 1.000  
## DM02\_01 34.111 5.628 6.061 0.000 34.111 1.000  
## DM01 0.130 0.013 10.181 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 8.062 3.639 2.215 0.027 8.062 0.232  
## cond2 4.933 3.669 1.345 0.179 4.933 0.105  
## cond3 2.947 4.133 0.713 0.476 2.947 0.104  
## cond4 -0.182 3.509 -0.052 0.959 -0.182 -0.023  
## cond5 6.679 4.181 1.598 0.110 6.679 0.192  
## cond6 3.550 4.712 0.753 0.451 3.550 0.065  
## cond7 1.564 2.850 0.549 0.583 1.564 0.064  
## cond8 -1.565 2.806 -0.558 0.577 -1.565 -0.064

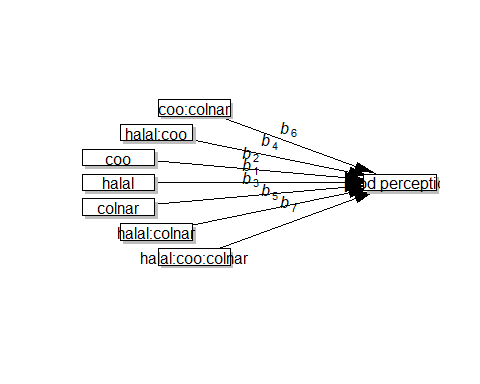
## Collective Narcissism

Here is the plot

### Plot  
labels=list(X="halal", W="coo", Z="colnar", Y="prod perception")  
pmacroModel(3,labels=labels)



statisticalDiagram(3,labels=labels)



This is the result of path analysis (using lavaan, when the dv is **product perception of cake** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

model1cn <- '  
# Model  
pp\_cake ~ b1\*halal + b2\*coo + b3\*cn\_mc + b4\*halalXcoo + b5\*halalXcn + b6\*cooXcn + b7\*halalXcooXcn + DM02\_01 + DM01  
  
  
# CN Mean and Variances  
cn\_mc ~ a\*1  
cn\_mc ~~ b\*cn\_mc   
  
## Three Way Interaction  
# Conditional effect  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond3:= b1 + b5\*(a+sqrt(b)) # English, high CN  
cond4:= b1 + b5\*(a-sqrt(b)) # English, low CN  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond7:= b5\*(a+sqrt(b)) # England, high CN  
cond8 := b5\*(a-sqrt(b)) # England, low CN  
  
'  
  
fitmod1cn <- sem(model1cn, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = model1cn, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

summary(fitmod1cn, standardized=T)

## lavaan 0.6-3 ended normally after 190 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 553.187  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_cake ~   
## halal (b1) 2.691 2.124 1.267 0.205 2.691 0.083  
## coo (b2) 3.534 2.273 1.555 0.120 3.534 0.109  
## cn\_mc (b3) -4.848 2.468 -1.965 0.049 -4.848 -0.199  
## halalXcoo (b4) -0.592 3.145 -0.188 0.851 -0.592 -0.016  
## halalXcn (b5) 10.809 3.310 3.266 0.001 10.809 0.310  
## cooXcn (b6) 6.636 3.557 1.866 0.062 6.636 0.195  
## hallXcXcn (b7) -7.288 4.962 -1.469 0.142 -7.288 -0.147  
## DM02\_01 -0.072 0.152 -0.471 0.638 -0.072 -0.026  
## DM01 1.410 2.545 0.554 0.580 1.410 0.031  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.013 -0.353 0.724 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.468 0.000 0.124 0.577  
## halalXcn 0.009 0.012 0.717 0.473 0.009 0.038  
## cooXcn 0.025 0.012 2.081 0.037 0.025 0.106  
## halalXcooXcn 0.015 0.008 1.848 0.065 0.015 0.094  
## DM02\_01 0.030 0.151 0.197 0.844 0.030 0.010  
## DM01 -0.002 0.009 -0.203 0.839 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.695 0.000 0.122 0.565  
## halalXcn 0.022 0.011 1.895 0.058 0.022 0.093  
## cooXcn 0.005 0.012 0.439 0.660 0.005 0.022  
## halalXcooXcn 0.015 0.008 1.866 0.062 0.015 0.092  
## DM02\_01 -0.044 0.147 -0.299 0.765 -0.044 -0.015  
## DM01 -0.003 0.009 -0.312 0.755 -0.003 -0.016  
## halalXcoo ~~   
## halalXcn 0.026 0.013 2.045 0.041 0.026 0.131  
## cooXcn 0.028 0.013 2.122 0.034 0.028 0.135  
## halalXcooXcn 0.023 0.012 1.866 0.062 0.023 0.162  
## DM02\_01 -0.043 0.121 -0.354 0.723 -0.043 -0.017  
## DM01 -0.004 0.008 -0.470 0.638 -0.004 -0.024  
## halalXcn ~~   
## cooXcn 0.107 0.020 5.456 0.000 0.107 0.487  
## halalXcooXcn 0.107 0.020 5.468 0.000 0.107 0.708  
## DM02\_01 0.097 0.134 0.722 0.470 0.097 0.036  
## DM01 -0.003 0.008 -0.431 0.666 -0.003 -0.021  
## cooXcn ~~   
## halalXcooXcn 0.107 0.020 5.472 0.000 0.107 0.690  
## DM02\_01 0.366 0.143 2.564 0.010 0.366 0.132  
## DM01 -0.006 0.010 -0.544 0.587 -0.006 -0.033  
## halalXcooXcn ~~   
## DM02\_01 0.134 0.092 1.453 0.146 0.134 0.070  
## DM01 -0.003 0.006 -0.415 0.678 -0.003 -0.022  
## DM02\_01 ~~   
## DM01 -0.382 0.141 -2.698 0.007 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cn\_mc (a) -0.002 0.034 -0.052 0.959 -0.002 -0.003  
## .pp\_cake -4.078 6.779 -0.602 0.548 -4.078 -0.252  
## halal 0.496 0.025 19.569 0.000 0.496 0.992  
## coo 0.506 0.026 19.794 0.000 0.506 1.013  
## halalXcoo 0.247 0.022 11.059 0.000 0.247 0.572  
## halalXcn 0.017 0.024 0.714 0.475 0.017 0.037  
## cooXcn 0.011 0.024 0.438 0.662 0.011 0.022  
## halalXcXcn 0.030 0.016 1.843 0.065 0.030 0.093  
## DM02\_01 21.595 0.302 71.579 0.000 21.595 3.697  
## DM01 1.847 0.019 99.425 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cn\_mc (b) 0.439 0.034 12.821 0.000 0.439 1.000  
## .pp\_cake 216.160 16.281 13.277 0.000 216.160 0.829  
## halal 0.250 0.001 285.354 0.000 0.250 1.000  
## coo 0.250 0.001 254.820 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 16.499 0.000 0.186 1.000  
## halalXcn 0.214 0.028 7.598 0.000 0.214 1.000  
## cooXcn 0.226 0.026 8.719 0.000 0.226 1.000  
## halalXcXcn 0.106 0.019 5.468 0.000 0.106 1.000  
## DM02\_01 34.111 5.659 6.028 0.000 34.111 1.000  
## DM01 0.130 0.013 10.143 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 9.243 3.261 2.835 0.005 9.243 0.376  
## cond2 -5.082 3.177 -1.600 0.110 -5.082 -0.243  
## cond3 9.835 2.757 3.568 0.000 9.835 0.392  
## cond4 -4.490 3.414 -1.315 0.188 -4.490 -0.227  
## cond5 6.552 4.147 1.580 0.114 6.552 0.293  
## cond6 -7.773 3.567 -2.179 0.029 -7.773 -0.326  
## cond7 7.143 2.245 3.182 0.001 7.143 0.309  
## cond8 -7.181 2.272 -3.160 0.002 -7.181 -0.311

This is the result of path analysis (using lavaan, when the dv is **product perception of beverage** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

# dv=prod perception \*\*BEVERAGE\*\* iv=halal, coo, collective narcissism  
  
model2cn <- '  
# Model  
pp\_bev ~ b1\*halal + b2\*coo + b3\*cn\_mc + b4\*halalXcoo + b5\*halalXcn + b6\*cooXcn + b7\*halalXcooXcn + DM02\_01 + DM01  
  
  
# CN Mean and Variances  
cn\_mc ~ a\*1  
cn\_mc ~~ b\*cn\_mc   
  
## Three Way Interaction  
# Conditional effect  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond3:= b1 + b5\*(a+sqrt(b)) # English, high CN  
cond4:= b1 + b5\*(a-sqrt(b)) # English, low CN  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond7:= b5\*(a+sqrt(b)) # England, high CN  
cond8 := b5\*(a-sqrt(b)) # England, low CN  
  
'  
  
fitmod2cn <- sem(model2cn, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = model2cn, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

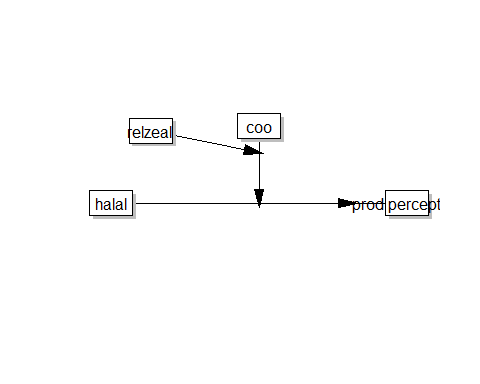
summary(fitmod2cn, standardized=T)

## lavaan 0.6-3 ended normally after 183 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 553.187  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_bev ~   
## halal (b1) 1.148 2.797 0.410 0.681 1.148 0.033  
## coo (b2) -0.070 2.585 -0.027 0.978 -0.070 -0.002  
## cn\_mc (b3) -3.242 3.214 -1.008 0.313 -3.242 -0.122  
## halalXcoo (b4) 5.165 3.575 1.445 0.149 5.165 0.126  
## halalXcn (b5) 2.698 4.603 0.586 0.558 2.698 0.071  
## cooXcn (b6) 5.144 4.334 1.187 0.235 5.144 0.139  
## hallXcXcn (b7) 0.534 6.111 0.087 0.930 0.534 0.010  
## DM02\_01 0.066 0.169 0.393 0.694 0.066 0.022  
## DM01 -0.992 2.644 -0.375 0.707 -0.992 -0.020  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.013 -0.350 0.726 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.661 0.000 0.124 0.577  
## halalXcn 0.009 0.012 0.748 0.454 0.009 0.038  
## cooXcn 0.025 0.012 2.039 0.041 0.025 0.106  
## halalXcooXcn 0.015 0.008 1.910 0.056 0.015 0.094  
## DM02\_01 0.030 0.148 0.199 0.842 0.030 0.010  
## DM01 -0.002 0.009 -0.211 0.833 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.775 0.000 0.122 0.565  
## halalXcn 0.022 0.011 1.901 0.057 0.022 0.093  
## cooXcn 0.005 0.012 0.447 0.655 0.005 0.022  
## halalXcooXcn 0.015 0.008 1.880 0.060 0.015 0.092  
## DM02\_01 -0.044 0.144 -0.306 0.759 -0.044 -0.015  
## DM01 -0.003 0.009 -0.310 0.756 -0.003 -0.016  
## halalXcoo ~~   
## halalXcn 0.026 0.013 2.068 0.039 0.026 0.131  
## cooXcn 0.028 0.013 2.121 0.034 0.028 0.135  
## halalXcooXcn 0.023 0.012 1.903 0.057 0.023 0.162  
## DM02\_01 -0.043 0.118 -0.362 0.717 -0.043 -0.017  
## DM01 -0.004 0.008 -0.471 0.638 -0.004 -0.024  
## halalXcn ~~   
## cooXcn 0.107 0.019 5.505 0.000 0.107 0.487  
## halalXcooXcn 0.107 0.019 5.521 0.000 0.107 0.708  
## DM02\_01 0.097 0.133 0.728 0.466 0.097 0.036  
## DM01 -0.003 0.008 -0.448 0.654 -0.003 -0.021  
## cooXcn ~~   
## halalXcooXcn 0.107 0.019 5.515 0.000 0.107 0.690  
## DM02\_01 0.366 0.141 2.603 0.009 0.366 0.132  
## DM01 -0.006 0.010 -0.550 0.583 -0.006 -0.033  
## halalXcooXcn ~~   
## DM02\_01 0.134 0.087 1.535 0.125 0.134 0.070  
## DM01 -0.003 0.006 -0.416 0.678 -0.003 -0.022  
## DM02\_01 ~~   
## DM01 -0.382 0.138 -2.773 0.006 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cn\_mc (a) -0.002 0.033 -0.054 0.957 -0.002 -0.003  
## .pp\_bev -1.533 7.230 -0.212 0.832 -1.533 -0.087  
## halal 0.496 0.026 18.936 0.000 0.496 0.992  
## coo 0.506 0.026 19.631 0.000 0.506 1.013  
## halalXcoo 0.247 0.022 11.435 0.000 0.247 0.572  
## halalXcn 0.017 0.023 0.746 0.456 0.017 0.037  
## cooXcn 0.011 0.024 0.447 0.655 0.011 0.022  
## halalXcXcn 0.030 0.016 1.887 0.059 0.030 0.093  
## DM02\_01 21.595 0.297 72.662 0.000 21.595 3.697  
## DM01 1.847 0.018 100.545 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cn\_mc (b) 0.439 0.033 13.426 0.000 0.439 1.000  
## .pp\_bev 285.441 17.326 16.475 0.000 285.441 0.916  
## halal 0.250 0.001 258.082 0.000 0.250 1.000  
## coo 0.250 0.001 238.906 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 17.128 0.000 0.186 1.000  
## halalXcn 0.214 0.027 8.080 0.000 0.214 1.000  
## cooXcn 0.226 0.027 8.471 0.000 0.226 1.000  
## halalXcXcn 0.106 0.019 5.523 0.000 0.106 1.000  
## DM02\_01 34.111 5.504 6.198 0.000 34.111 1.000  
## DM01 0.130 0.013 10.220 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 8.096 3.920 2.065 0.039 8.096 0.229  
## cond2 4.521 3.843 1.176 0.239 4.521 0.088  
## cond3 2.931 4.105 0.714 0.475 2.931 0.103  
## cond4 -0.644 4.148 -0.155 0.877 -0.644 -0.038  
## cond5 6.948 4.758 1.460 0.144 6.948 0.197  
## cond6 3.373 4.641 0.727 0.467 3.373 0.055  
## cond7 1.783 3.053 0.584 0.559 1.783 0.071  
## cond8 -1.792 3.030 -0.592 0.554 -1.792 -0.071

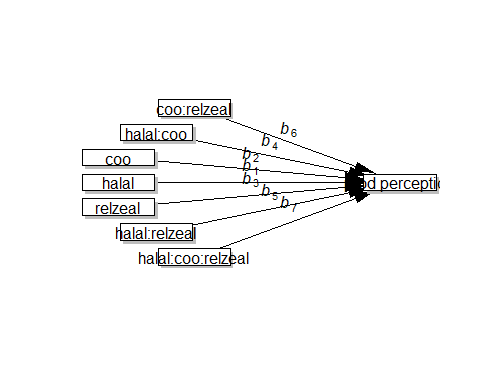
## Religious Zeal

Here is the plot

### Plot  
labels=list(X="halal", W="coo", Z="relzeal", Y="prod perception")  
pmacroModel(3,labels=labels)



statisticalDiagram(3,labels=labels)



This is the result of path analysis (using lavaan, when the dv is **product perception of cake** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

model1rz <- '  
# Model  
pp\_cake ~ b1\*halal + b2\*coo + b3\*rz\_mc + b4\*halalXcoo + b5\*halalXrz + b6\*cooXrz + b7\*halalXcooXrz + DM02\_01 + DM01  
  
  
# CN Mean and Variances  
rz\_mc ~ a\*1  
rz\_mc ~~ b\*rz\_mc   
  
## Three Way Interaction  
# Conditional effect  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high RZ  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turkey, low RZ  
cond3:= b1 + b5\*(a+sqrt(b)) # English, high RZ  
cond4:= b1 + b5\*(a-sqrt(b)) # English, low RZ  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high RZ  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low RZ  
cond7:= b5\*(a+sqrt(b)) # England, high RZ  
cond8 := b5\*(a-sqrt(b)) # England, low RZ  
  
'  
  
fitmod1rz <- sem(model1rz, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = model1rz, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

summary(fitmod1rz, standardized=T)

## lavaan 0.6-3 ended normally after 196 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 514.309  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_cake ~   
## halal (b1) 2.289 1.964 1.165 0.244 2.289 0.065  
## coo (b2) 2.449 2.207 1.110 0.267 2.449 0.070  
## rz\_mc (b3) -8.526 1.964 -4.340 0.000 -8.526 -0.368  
## halalXcoo (b4) 0.099 2.992 0.033 0.974 0.099 0.002  
## halalXrz (b5) 13.497 2.718 4.966 0.000 13.497 0.398  
## cooXrz (b6) 6.440 2.815 2.288 0.022 6.440 0.195  
## hallXcXrz (b7) -6.671 3.989 -1.673 0.094 -6.671 -0.137  
## DM02\_01 0.008 0.146 0.055 0.956 0.008 0.003  
## DM01 1.312 2.198 0.597 0.551 1.312 0.027  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.013 -0.355 0.722 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.231 0.000 0.124 0.577  
## halalXrz 0.004 0.014 0.305 0.760 0.004 0.016  
## cooXrz 0.036 0.014 2.616 0.009 0.036 0.134  
## halalXcooXrz 0.020 0.009 2.190 0.029 0.020 0.110  
## DM02\_01 0.030 0.146 0.202 0.840 0.030 0.010  
## DM01 -0.002 0.009 -0.204 0.838 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.274 0.000 0.122 0.565  
## halalXrz 0.036 0.013 2.693 0.007 0.036 0.137  
## cooXrz 0.004 0.013 0.306 0.759 0.004 0.015  
## halalXcooXrz 0.020 0.009 2.191 0.028 0.020 0.108  
## DM02\_01 -0.044 0.147 -0.300 0.764 -0.044 -0.015  
## DM01 -0.003 0.009 -0.328 0.743 -0.003 -0.016  
## halalXcoo ~~   
## halalXrz 0.038 0.014 2.611 0.009 0.038 0.168  
## cooXrz 0.038 0.015 2.569 0.010 0.038 0.164  
## halalXcooXrz 0.030 0.014 2.202 0.028 0.030 0.191  
## DM02\_01 -0.043 0.122 -0.353 0.724 -0.043 -0.017  
## DM01 -0.004 0.008 -0.457 0.648 -0.004 -0.024  
## halalXrz ~~   
## cooXrz 0.133 0.021 6.279 0.000 0.133 0.481  
## halalXcooXrz 0.133 0.021 6.371 0.000 0.133 0.705  
## DM02\_01 0.109 0.167 0.652 0.514 0.109 0.036  
## DM01 -0.004 0.010 -0.370 0.711 -0.004 -0.020  
## cooXrz ~~   
## halalXcooXrz 0.133 0.021 6.333 0.000 0.133 0.688  
## DM02\_01 0.508 0.167 3.049 0.002 0.508 0.163  
## DM01 -0.003 0.011 -0.252 0.801 -0.003 -0.015  
## halalXcooXrz ~~   
## DM02\_01 0.212 0.120 1.773 0.076 0.212 0.100  
## DM01 -0.003 0.008 -0.413 0.679 -0.003 -0.025  
## DM02\_01 ~~   
## DM01 -0.382 0.137 -2.780 0.005 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## rz\_mc (a) -0.003 0.039 -0.077 0.939 -0.003 -0.004  
## .pp\_cake -4.919 5.822 -0.845 0.398 -4.919 -0.279  
## halal 0.496 0.027 18.622 0.000 0.496 0.992  
## coo 0.506 0.026 19.825 0.000 0.506 1.013  
## halalXcoo 0.247 0.023 10.844 0.000 0.247 0.572  
## halalXrz 0.008 0.027 0.307 0.759 0.008 0.016  
## cooXrz 0.008 0.027 0.306 0.759 0.008 0.015  
## halalXcXrz 0.040 0.018 2.175 0.030 0.040 0.109  
## DM02\_01 21.595 0.295 73.251 0.000 21.595 3.697  
## DM01 1.847 0.018 103.469 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## rz\_mc (b) 0.576 0.038 14.999 0.000 0.576 1.000  
## .pp\_cake 207.914 14.997 13.864 0.000 207.914 0.671  
## halal 0.250 0.001 237.324 0.000 0.250 1.000  
## coo 0.250 0.001 246.811 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 16.202 0.000 0.186 1.000  
## halalXrz 0.270 0.029 9.448 0.000 0.270 1.000  
## cooXrz 0.283 0.032 8.790 0.000 0.283 1.000  
## halalXcXrz 0.132 0.021 6.386 0.000 0.132 1.000  
## DM02\_01 34.111 5.596 6.095 0.000 34.111 1.000  
## DM01 0.130 0.012 10.491 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 12.596 3.143 4.008 0.000 12.596 0.464  
## cond2 -7.900 3.203 -2.466 0.014 -7.900 -0.332  
## cond3 12.496 2.894 4.319 0.000 12.496 0.462  
## cond4 -8.000 2.913 -2.746 0.006 -8.000 -0.335  
## cond5 10.307 3.694 2.790 0.005 10.307 0.399  
## cond6 -10.189 3.724 -2.736 0.006 -10.189 -0.397  
## cond7 10.207 2.172 4.699 0.000 10.207 0.397  
## cond8 -10.289 2.161 -4.760 0.000 -10.289 -0.400

This is the result of path analysis (using lavaan, when the dv is **product perception of beverage** and with two control variables (**gender** (DM01) and **age** (DM02\_01)).

# dv=prod perception \*\*BEVERAGE\*\* iv=halal, coo, religious zeal  
  
model2rz <- '  
# Model  
pp\_bev ~ b1\*halal + b2\*coo + b3\*rz\_mc + b4\*halalXcoo + b5\*halalXrz + b6\*cooXrz + b7\*halalXcooXrz + DM02\_01 + DM01  
  
  
# CN Mean and Variances  
rz\_mc ~ a\*1  
rz\_mc ~~ b\*rz\_mc   
  
## Three Way Interaction  
# Conditional effect  
cond1 := b1 + b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond2:= b1 + b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond3:= b1 + b5\*(a+sqrt(b)) # English, high CN  
cond4:= b1 + b5\*(a-sqrt(b)) # English, low CN  
  
# Conditional effect (no-Halal)  
cond5 := b4 + b5\*(a+sqrt(b)) # Turkey, high CN  
cond6:= b4 + b5\*(a-sqrt(b)) # Turkey, low CN  
cond7:= b5\*(a+sqrt(b)) # England, high CN  
cond8 := b5\*(a-sqrt(b)) # England, low CN  
  
'  
  
fitmod2rz <- sem(model2rz, data=ds, se="bootstrap", bootstrap=1000)

## Warning in lavaan::lavaan(model = model2rz, data = ds, se = "bootstrap", :  
## lavaan WARNING: syntax contains parameters involving exogenous covariates;  
## switching to fixed.x = FALSE

summary(fitmod2rz, standardized=T)

## lavaan 0.6-3 ended normally after 204 iterations  
##   
## Optimization method NLMINB  
## Number of free parameters 57  
##   
## Used Total  
## Number of observations 385 386  
##   
## Estimator ML  
## Model Fit Test Statistic 514.309  
## Degrees of freedom 8  
## P-value (Chi-square) 0.000  
##   
## Parameter Estimates:  
##   
## Standard Errors Bootstrap  
## Number of requested bootstrap draws 1000  
## Number of successful bootstrap draws 1000  
##   
## Regressions:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## pp\_bev ~   
## halal (b1) 0.954 2.433 0.392 0.695 0.954 0.025  
## coo (b2) -0.806 2.669 -0.302 0.763 -0.806 -0.021  
## rz\_mc (b3) -7.338 2.672 -2.746 0.006 -7.338 -0.294  
## halalXcoo (b4) 5.851 3.552 1.647 0.099 5.851 0.133  
## halalXrz (b5) 10.552 3.721 2.836 0.005 10.552 0.289  
## cooXrz (b6) 7.461 3.779 1.974 0.048 7.461 0.210  
## hallXcXrz (b7) -8.934 5.164 -1.730 0.084 -8.934 -0.171  
## DM02\_01 0.146 0.156 0.937 0.349 0.146 0.045  
## DM01 -1.082 2.662 -0.406 0.685 -1.082 -0.021  
##   
## Covariances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## halal ~~   
## coo -0.005 0.012 -0.366 0.714 -0.005 -0.018  
## halalXcoo 0.124 0.009 13.804 0.000 0.124 0.577  
## halalXrz 0.004 0.013 0.307 0.759 0.004 0.016  
## cooXrz 0.036 0.013 2.679 0.007 0.036 0.134  
## halalXcooXrz 0.020 0.009 2.192 0.028 0.020 0.110  
## DM02\_01 0.030 0.147 0.201 0.841 0.030 0.010  
## DM01 -0.002 0.009 -0.213 0.831 -0.002 -0.011  
## coo ~~   
## halalXcoo 0.122 0.009 13.636 0.000 0.122 0.565  
## halalXrz 0.036 0.013 2.764 0.006 0.036 0.137  
## cooXrz 0.004 0.013 0.305 0.761 0.004 0.015  
## halalXcooXrz 0.020 0.009 2.204 0.028 0.020 0.108  
## DM02\_01 -0.044 0.150 -0.295 0.768 -0.044 -0.015  
## DM01 -0.003 0.009 -0.315 0.752 -0.003 -0.016  
## halalXcoo ~~   
## halalXrz 0.038 0.014 2.636 0.008 0.038 0.168  
## cooXrz 0.038 0.015 2.599 0.009 0.038 0.164  
## halalXcooXrz 0.030 0.014 2.204 0.027 0.030 0.191  
## DM02\_01 -0.043 0.125 -0.344 0.731 -0.043 -0.017  
## DM01 -0.004 0.008 -0.464 0.643 -0.004 -0.024  
## halalXrz ~~   
## cooXrz 0.133 0.020 6.534 0.000 0.133 0.481  
## halalXcooXrz 0.133 0.020 6.631 0.000 0.133 0.705  
## DM02\_01 0.109 0.162 0.673 0.501 0.109 0.036  
## DM01 -0.004 0.010 -0.386 0.700 -0.004 -0.020  
## cooXrz ~~   
## halalXcooXrz 0.133 0.020 6.622 0.000 0.133 0.688  
## DM02\_01 0.508 0.161 3.155 0.002 0.508 0.163  
## DM01 -0.003 0.011 -0.267 0.789 -0.003 -0.015  
## halalXcooXrz ~~   
## DM02\_01 0.212 0.118 1.790 0.073 0.212 0.100  
## DM01 -0.003 0.008 -0.429 0.668 -0.003 -0.025  
## DM02\_01 ~~   
## DM01 -0.382 0.141 -2.709 0.007 -0.382 -0.181  
##   
## Intercepts:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## rz\_mc (a) -0.003 0.039 -0.076 0.939 -0.003 -0.004  
## .pp\_bev -2.474 6.769 -0.365 0.715 -2.474 -0.131  
## halal 0.496 0.025 20.238 0.000 0.496 0.992  
## coo 0.506 0.025 19.924 0.000 0.506 1.013  
## halalXcoo 0.247 0.022 11.186 0.000 0.247 0.572  
## halalXrz 0.008 0.027 0.306 0.759 0.008 0.016  
## cooXrz 0.008 0.027 0.304 0.761 0.008 0.015  
## halalXcXrz 0.040 0.018 2.170 0.030 0.040 0.109  
## DM02\_01 21.595 0.314 68.752 0.000 21.595 3.697  
## DM01 1.847 0.018 100.250 0.000 1.847 5.127  
##   
## Variances:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## rz\_mc (b) 0.576 0.039 14.884 0.000 0.576 1.000  
## .pp\_bev 279.787 17.888 15.641 0.000 279.787 0.779  
## halal 0.250 0.001 291.004 0.000 0.250 1.000  
## coo 0.250 0.001 266.128 0.000 0.250 1.000  
## halalXcoo 0.186 0.011 16.668 0.000 0.186 1.000  
## halalXrz 0.270 0.028 9.503 0.000 0.270 1.000  
## cooXrz 0.283 0.031 9.061 0.000 0.283 1.000  
## halalXcXrz 0.132 0.020 6.652 0.000 0.132 1.000  
## DM02\_01 34.111 5.699 5.986 0.000 34.111 1.000  
## DM01 0.130 0.013 10.163 0.000 0.130 1.000  
##   
## Defined Parameters:  
## Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
## cond1 14.785 3.813 3.877 0.000 14.785 0.447  
## cond2 -1.239 3.774 -0.328 0.743 -1.239 -0.132  
## cond3 8.934 3.748 2.384 0.017 8.934 0.313  
## cond4 -7.090 3.783 -1.874 0.061 -7.090 -0.265  
## cond5 13.832 4.604 3.004 0.003 13.832 0.421  
## cond6 -2.193 4.511 -0.486 0.627 -2.193 -0.157  
## cond7 7.980 2.893 2.758 0.006 7.980 0.288  
## cond8 -8.044 2.843 -2.829 0.005 -8.044 -0.291

# Reliability analysis

To estimate reliability, I performed McDonald’s omega (with confidence interval) with MBESS package.

library(dplyr)  
library(MBESS)

## Warning: package 'MBESS' was built under R version 3.5.3

##   
## Attaching package: 'MBESS'

## The following object is masked from 'package:lavaan':  
##   
## cor2cov

rz <- select(ds, 36:51)  
ncc <- select(ds, 16:30)  
cn <- select(ds, 31:35)  
prodperc\_bev <- select(ds, 78,80,82,84,86,88)  
prodperc\_cake <- select(ds, 79,81,83,85,87,89)

…and here is the result…

# reliability (omega)  
ci.reliability(data=rz, type="omega", interval.type="bca", B=1000)

## $est  
## [1] 0.8956031  
##   
## $se  
## [1] 0.008192329  
##   
## $ci.lower  
## [1] 0.8785484  
##   
## $ci.upper  
## [1] 0.9095459  
##   
## $conf.level  
## [1] 0.95  
##   
## $type  
## [1] "omega"  
##   
## $interval.type  
## [1] "bca bootstrap"

ci.reliability(data=ncc, type="omega", interval.type="bca", B=1000)

## $est  
## [1] 0.7407914  
##   
## $se  
## [1] 0.04481911  
##   
## $ci.lower  
## [1] 0.6763228  
##   
## $ci.upper  
## [1] 0.7940351  
##   
## $conf.level  
## [1] 0.95  
##   
## $type  
## [1] "omega"  
##   
## $interval.type  
## [1] "bca bootstrap"

ci.reliability(data=cn, type="omega", interval.type="bca", B=1000)

## $est  
## [1] 0.7507226  
##   
## $se  
## [1] 0.02349959  
##   
## $ci.lower  
## [1] 0.7005773  
##   
## $ci.upper  
## [1] 0.7901337  
##   
## $conf.level  
## [1] 0.95  
##   
## $type  
## [1] "omega"  
##   
## $interval.type  
## [1] "bca bootstrap"

ci.reliability(data=prodperc\_bev, type="omega", interval.type="bca", B=1000)

## $est  
## [1] 0.925967  
##   
## $se  
## [1] 0.006372094  
##   
## $ci.lower  
## [1] 0.913036  
##   
## $ci.upper  
## [1] 0.9377648  
##   
## $conf.level  
## [1] 0.95  
##   
## $type  
## [1] "omega"  
##   
## $interval.type  
## [1] "bca bootstrap"

ci.reliability(data=prodperc\_cake, type="omega", interval.type="bca", B=1000)

## $est  
## [1] 0.9134072  
##   
## $se  
## [1] 0.008432604  
##   
## $ci.lower  
## [1] 0.8951692  
##   
## $ci.upper  
## [1] 0.9286659  
##   
## $conf.level  
## [1] 0.95  
##   
## $type  
## [1] "omega"  
##   
## $interval.type  
## [1] "bca bootstrap"

It looks like the reliability of CN and NCC scales are slightly unsatisfying (~0.7), while RZ scale has the best reliability coefficient.