**Mohadeseh Najafi Semnani**

**Assignment 2**

**Supervisor: Dr. Maurizio Vichi**

**Dr. Mario Fordellone**

**SEM analysis for European Consumer Satisfaction Index (ECSI)**

SEM has the property to estimate the multiple and interrelated dependencies in a single analysis by combining factor analysis and multivariate regression analysis. SEM has been used in many different fields, as in economics and social sciences, in

Marketing for example to assess customer satisfaction.

**Path analysis and Structural Equation Modelling**

1) Define a sequential structural equation models on ECSI dataset included in the folder. In particular, apply 7 different principal component model (one for each block) and take only the first principal component obtained by each block. Subsequently, through the application of regression models, use these obtained component scores to construct a path model as shown in the figure below.

In this report a SEM method analysis (structural equation model) has been done on a dataset including 250 observations of 24 variables related to

A Commercial consumer factors model which will lead to satisfaction and loyal approach of consumers.

Name of factors that has been studied in this case and are the latent variables are given as follows:

Image : ima , Expectation : exp , perceived value : val , perceived quality : qua , satisfaction : sat , complaints : comp , loyalty : loy



LV is represented by an ellipse or circle

MV is represented by rectangle or a square

By installing packages lavaan and sempath we are going to draw the semplot for model of this dataset in Rstudio.

> # Import and explore data -------------------------------------------------

> library(lavaan)

> library(SemPath)

## # the application of command library : Loading/Attaching and Listing of Packages

> data <- read.table("data.txt", header=T)

> names(data)

[1] "ima1" "ima2" "ima3" "ima4" "ima5" "exp1"

[7] "exp2" "exp3" "qua1" "qua2" "qua3" "qua4"

[13] "qua5" "qua6" "qua7" "val1" "val2" "sat1"

[19] "sat2" "sat3" "comp" "loy1" "loy2" "loy3"

> # Model -------------------------------------------------------------------

>

> model <- '

+ # measurement model

+ ima =~ ima1 + ima2 + ima3 + ima4 + ima5

+ exp =~ exp1 + exp2 + exp3

+ qua =~ qua1 + qua2 + qua3 + qua4 + qua5 + qua6 + qua7

+ val =~ val1 + val2

+ sat =~ sat1 + sat2 + sat3

+ Comp =~ comp

+ loy =~ loy1 + loy2 + loy3

+ # structural model

+ exp ~ ima

+ qua ~ exp

+ val ~ exp + qua

+ sat ~ ima + exp + qua + val

+ comp ~ sat

+ loy ~ ima + sat + comp

+ '

2) Through “lavaan” R package, apply the structural equation modelling (SEM) on the same dataset and using the same theoretical path model. Use the “semPath” R package to represent your path model.

data <- read.table(file = "data.txt", header = T)

head(data)

dim(data)

names(data)

library("FactoMineR")

Factor1 = PCA(X = data[,c(1:5)], ncp = 1)

Factor1 = Factor1$ind$coord

Factor2 = PCA(X = data[,c(6:8)], ncp = 1)

Factor2 = Factor2$ind$coord

Factor3 = PCA(X = data[,c(9:15)], ncp = 1)

Factor3 = Factor3$ind$coord

Factor4 = PCA(X = data[,c(16:17)], ncp = 1)

Factor4 = Factor4$ind$coord

Factor5 = PCA(X = data[,c(18:20)], ncp = 1)

Factor5 = Factor5$ind$coord

Factor6 = scale(data[,21])

Factor7 = PCA(X = data[,c(22:24)], ncp = 1)

Factor7 = Factor7$ind$coord

Factor7r = lm(formula = Factor7 ~ Factor6 + Factor5 + Factor1)

summary(Factor7r)

Factor6r = lm(formula = Factor6 ~ Factor5)

summary(Factor6r)

Factor5r = lm(formula = Factor5 ~ Factor1 + Factor2 + Factor3 + Factor4)

summary(Factor5r)

Factor4r = lm(formula = Factor4 ~ Factor2 + Factor3)

summary(Factor4r)

Factor3r = lm(formula = Factor3 ~ Factor2)

summary(Factor3r)

Factor2r = lm(formula = Factor2 ~ Factor1)

summary(Factor2r)

> # Estimation model --------------------------------------------------------

>

> fit <- sem(model, data)

> summary(fit)

In this step by applying command “sem” measurement and structural models could be built regarding correlation coefficient between LV and regression between ….

lavaan 0.6-3 ended normally after 914 iterations and “fit” is created as a large lavaan object in database:

the output result for estimation are as follows:

Latent Variables:

Estimate Std.Err z-value P(>|z|)

ima =~

ima1 1.000

ima2 0.798 NA

ima3 0.841 NA

ima4 1.122 NA

ima5 0.891 NA

exp =~

exp1 1.000

exp2 0.970 NA

exp3 1.076 NA

qua =~

qua1 1.000

qua2 0.964 NA

qua3 1.259 NA

qua4 1.040 NA

qua5 0.932 NA

qua6 1.054 NA

qua7 1.274 NA

val =~

val1 1.000

val2 1.076 NA

sat =~

sat1 1.000

sat2 1.477 NA

sat3 1.614 NA

Comp =~

comp 1.000

loy =~

loy1 1.000

loy2 0.219 NA

loy3 1.174 NA

Regressions:

Estimate Std.Err z-value P(>|z|)

exp ~

ima 0.703 NA

qua ~

exp 1.469 NA

val ~

exp 71.633 NA

qua -47.728 NA

sat ~

ima 17.961 NA

exp 263.318 NA

qua -192.050 NA

val -3.702 NA

comp ~

sat 1.399 NA

loy ~

ima -3.137 NA

sat 5.430 NA

comp -0.005 NA

Covariances:

Estimate Std.Err z-value P(>|z|)

ima ~~

Comp 18.276 NA

Variances:

Estimate Std.Err z-value P(>|z|)

.ima1 216.777 NA

.ima2 262.150 NA

.ima3 461.232 NA

.ima4 243.272 NA

.ima5 188.453 NA

.exp1 256.867 NA

.exp2 331.036 NA

.exp3 464.722 NA

.qua1 100.716 NA

.qua2 302.820 NA

.qua3 174.648 NA

.qua4 176.092 NA

.qua5 131.780 NA

.qua6 163.038 NA

.qua7 178.604 NA

.val1 264.214 NA

.val2 44.932 NA

.sat1 96.170 NA

.sat2 184.776 NA

.sat3 138.347 NA

.comp 0.000

.loy1 535.894 NA

.loy2 975.144 NA

.loy3 146.148 NA

ima 138.137 NA

.exp 0.068 NA

.qua 0.069 NA

.val 10.953 NA

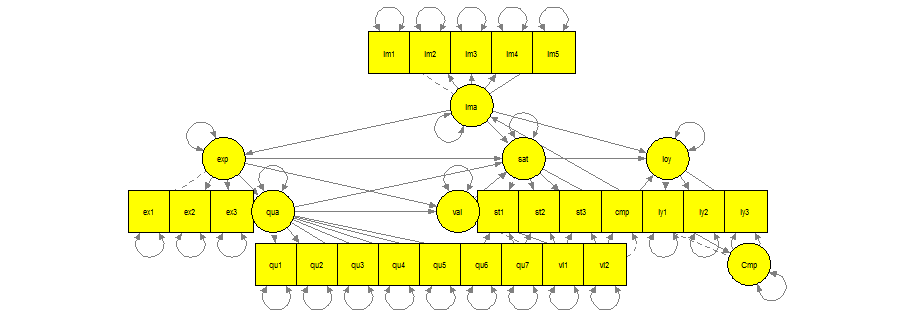
.sat -203.265 NA

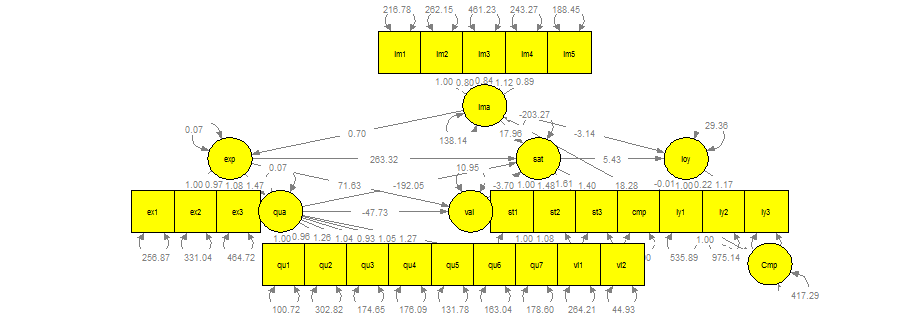
Comp 417.293 NA

.loy 29.362 NA

semPaths(fit, col = "yellow", whatLabels = "est" , edge.label.cex=1, layout = "tree2", curvePivot = TRUE, rotation = 1, sizeMan = 6, sizeLat = 6)

the basic SEM model regardless of estimation of covariance matrix and weights is the below graph:





**Reversal arrows in this diagram are referring to variance for each LV or MV variable regarding to itself,**

**Rstudio labled all estimations with low rounding on the edges. One side arrows are covariances between latent variables, as well.**

**Fitmeasures function:**

(This function computes a variety of fit measures to assess the global fit of a latent variable model)

> fitMeasures(object = fit, c("gfi", "rmsea", "aic", "bic"))

gfi rmsea aic bic

0.832 0.075 50913.056 51124.344

Measures considered for fitting this model in above vector are as follows:

# Goodnes of Fit (GFI)

# Root Mean Square Error of Approximation (RMSEA)

# Akaike's An Information Criterion (AIC)

# Bayesian Information Criterion (BIC)

3) Create a ranking with respect to “satisfaction” latent scores obtained in point 1) and point 2). Who is the most satisfied?

In this segment, to have a specific analysis in order to comment on satisfaction latent scores, a new variable containing sat1+sat2+sat3 columns has been defined and added to the main database. Hence, this new database named data2 has 25 variables; data2 <- (data + satisfaction)

data2=cbind(data, satisfaction)

u=anyDuplicated(which.max(satisfaction))

this command has been used to check out; if there are more than 1 maximum existing numbers of customers for satisfaction, in this current case the value for object “u” equals to 0, means that only 1 maximum exists. As highlighted in blue, it is also obvious that the most strong regression coefficient boosting the satisfaction, is the expectation with 263.318, that is highly considerable and is illustrating a positive relation between these two factors.

which.max(satisfaction)

[1] 4

Thus, the 4’th customer has been the most satisfied one.