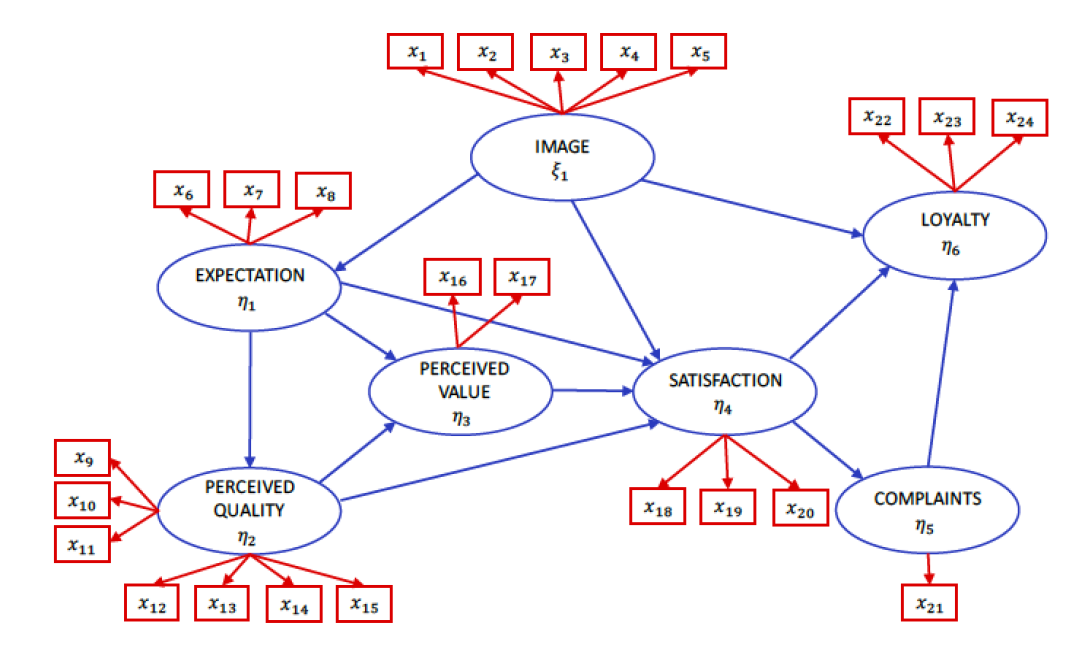
**Assignment 2**

Name: Mousaalreza Dastmard

Course: Advances in data analysis and statistical modelling

**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.



The dataset including 250 observations that has 24 variables (showed as

Endogenous:

Exogenous:

Using “FactoMineR” package, R is able to setup PCA, the first lines of the following code import the dataset to R and also activate the packages that are needed for both PCA + Regression and SEM model. Only the summary results of the first Regression model on LV is shown on the report to avoid repetitive results.

*# Read Dataset and R initiating -------------------------------------------*

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

names(data)

## [1] "ima1" "ima2" "ima3" "ima4" "ima5" "exp1" "exp2" "exp3" "qua1" "qua2"

## [11] "qua3" "qua4" "qua5" "qua6" "qua7" "val1" "val2" "sat1" "sat2" "sat3"

## [21] "comp" "loy1" "loy2" "loy3"

**attach**(data)

**library**(lavaan)

**library**(semPlot)

**library**(FactoMineR)

*# PCA ---------------------------------------------------------------------*

*# ima ~ ima1 + ima2 + ima3 + ima4 + ima5*

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

ima = ima$ind$coord

*# exp ~ exp1 + exp2 + exp3*

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

exp = exp$ind$coord

*# qua ~ qua1 + qua2 + qua3 + qua4 + qua5 + qua6 + qua7*

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

qua = qua$ind$coord

*# val ~ val1 + val2*

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

val = val$ind$coord

*# sat ~ sat1 + sat2 + sat3*

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

sat = sat$ind$coord

*# Comp ~ comp*

Comp = scale(data[,21])

*# loy ~ loy1 + loy2 + loy3*

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

loy = loy$ind$coord

*# Regression Model --------------------------------------------------------*

loy.r = lm(formula = loy ~ ima + sat + comp)

summary(loy.r)

##

## Call:

## lm(formula = loy ~ ima + sat + comp)

##

## Residuals:

## Min 1Q Median 3Q Max

## -5.2211 -0.3811 0.1041 0.5247 3.1745

##

## Coefficients:

## Estimate Std. Error t value Pr(>|t|)

## (Intercept) -0.209259 0.200812 -1.042 0.29840

## ima 0.169904 0.054449 3.120 0.00202 \*\*

## sat 0.395214 0.060163 6.569 2.99e-10 \*\*\*

## comp 0.003104 0.002842 1.092 0.27584

## ---

## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

##

## Residual standard error: 0.9504 on 246 degrees of freedom

## Multiple R-squared: 0.4307, Adjusted R-squared: 0.4237

## F-statistic: 62.03 on 3 and 246 DF, p-value: < 2.2e-16

comp.r = lm(formula = comp ~ sat)

summary(comp.r)

sat.r = lm(formula = sat ~ ima + exp + val + qua)

summary(sat.r)

val.r = lm(formula = val ~ exp + qua)

summary(val.r)

qua.r = lm(formula = qua ~ exp)

summary(qua.r)

exp.r = lm(formula = exp ~ ima)

summary(exp.r)

1. 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.

With the help of the handy packages such lavaan and semPlot, the SEM model can be made and visualized in R. The initial steps are common with the first question and ignored. Based on the figure the arrows show the measurement and structural model. That can be translated to lavaan package as follow:

*# Specify and estimate 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

Comp =~ comp

sat =~ sat1 + sat2 + sat3

loy =~ loy1 + loy2 + loy3

# structural model

loy ~ ima + sat + comp

comp ~ sat

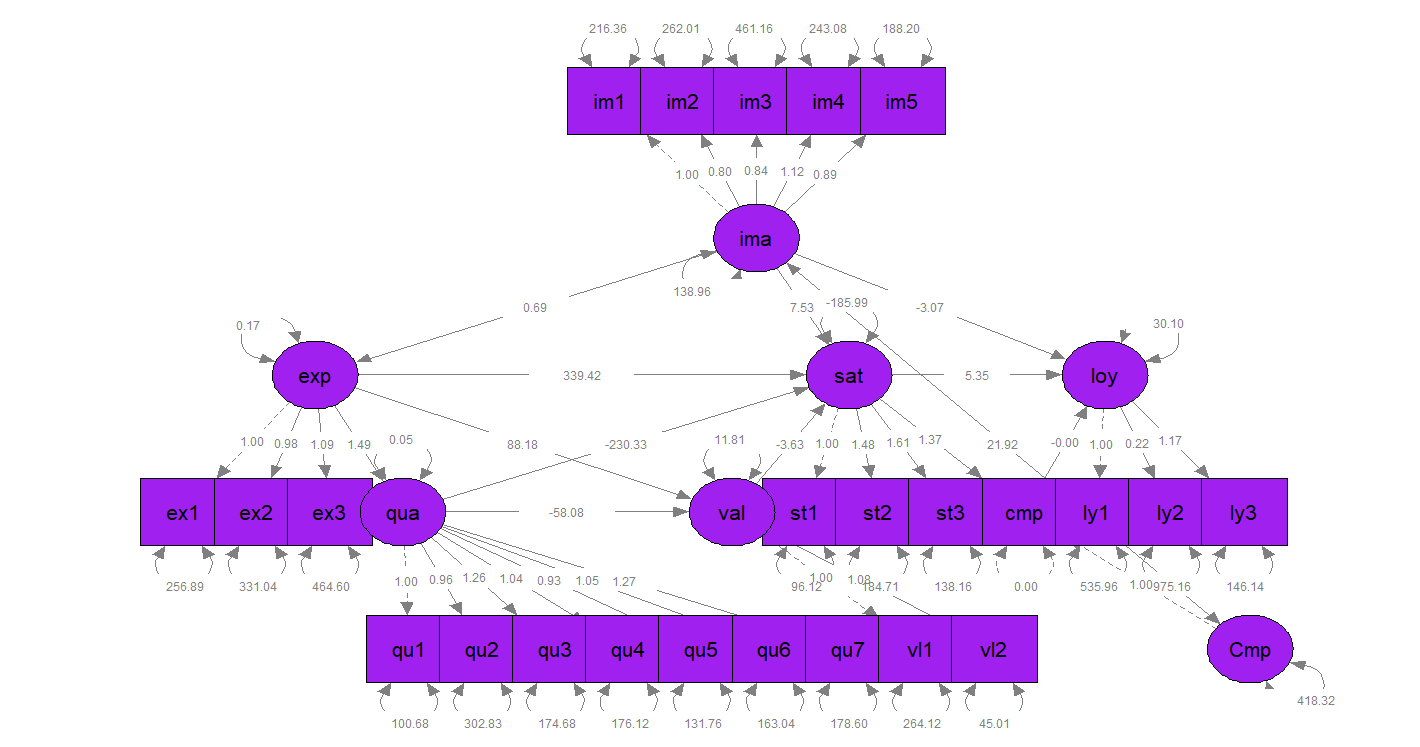
sat ~ ima + exp + val + qua

val ~ exp + qua

qua ~ exp

exp ~ ima

'



**Model Assessment**

gfi rmsea aic bic

0.832 0.075 50912.956 51124.244

Where:

gfi is Goodness of Fit

rmsea is the Root Mean Square Error of Approximation

aic is Akaike's An Information Criterion

bic is Bayesian Information Criterion

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

To leave comment on satisfaction, we first take a look at loadings of satisfaction that are all positive. So, we are sure there is no compensation among manifest variables (sat1, sat2, sat3). So the highest value that satisfaction takes, means the highest arithmetic means of sat1, sat2 and sat3 with respect to positive weights.

The simple command in R shows the person with highest satisfaction.

which.max(satisfaction)

[1] 4

Thus, the 4’th one is the most satisfied person.