Code

# Guía de PLS-SEM en R

**Felipe A. López**

julio de 2023

**NOTA:** Se entiende que usted maneja los conceptos básicos de ecuaciones estructurales y que realizó la limpieza y validación de sus datos.

# A. Carga de librería y directorio a trabajar

## A.1 Carga de librerías

Hide

```
# install.packages(pkgs = 'seminr')
# install.packages("xlsx")
#install.packages("genpathmox")
#install.packages("cSEM")
#install.packages("psych")
library(seminr)
library(xlsx)

library(cSEM)
library(genpathmox)

library('psych')
```

## A.2 Carga de datos

### A.2.1 Carga de directorio de trabajo y datos

Reemplace directorio:

Hide

```
getwd()
```

```
## [1] "P:/R_Proyect/PLS-SEM/Proyecto/Rmark"
```

Hide

```
directorio <- "P:/R_Proyect/PLS-SEM/Proyecto" ## Reemplace por su directorio
#setwd(directorio)  # si desea dejar fijo el directorio de trabajo
getwd()
```

```
## [1] "P:/R_Proyect/PLS-SEM/Proyecto/Rmark"
```

En file sustituya por archivo de datos

Hide

```
pls_data <- read.csv(file = "P:/R_Proyect/PLS-SEM/Proyecto/2023.TRI_MGA.csv", header = TRUE, sep = ';')
dim(pls_data)  ## Ver cantidad de filas y columnas
```

```
## [1] 383  54
```

Ver cabecera de los datos y tipos

Hide

```
head(pls_data)  ### Primeros datos
```

| | ï..ID <int> | PE1 <int> | PE2 <int> | PE3 <int> | PE4 <int> | EE1 <int> | EE2 <int> | EE3 <int> | SI1 <int> |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 5 | 4 | 5 | 4 | 3 | 2 | 3 | 5 |
| 2 | 2 | 5 | 5 | 5 | 5 | 3 | 3 | 3 | 4 |
| 3 | 3 | 5 | 5 | 5 | 4 | 4 | 4 | 2 | 5 |
| 4 | 4 | 2 | 2 | 4 | 3 | 2 | 1 | 1 | 3 |
| 5 | 5 | 3 | 4 | 3 | 3 | 2 | 1 | 1 | 4 |
| 6 | 6 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 4 |

6 rows | 1-10 of 55 columns

Hide

```
str(pls_data) ### Tipo de datos
```

```
## 'data.frame':    383 obs. of  54 variables:
##  $ ï..ID     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ PE1       : int  5 5 5 2 3 5 5 4 4 3 ...
##  $ PE2       : int  4 5 5 2 4 5 5 5 4 4 ...
##  $ PE3       : int  5 5 5 4 3 5 5 5 5 5 ...
##  $ PE4       : int  4 5 4 3 3 4 1 1 5 5 ...
##  $ EE1       : int  3 3 4 2 2 5 3 4 4 5 ...
##  $ EE2       : int  2 3 4 1 1 5 5 4 5 5 ...
##  $ EE3       : int  3 3 2 1 1 5 5 5 5 5 ...
##  $ SI1       : int  5 4 5 3 4 4 5 5 5 5 ...
##  $ SI2       : int  4 4 5 3 4 3 4 4 4 5 ...
##  $ SI3       : int  5 4 5 3 4 5 4 4 4 5 ...
##  $ SI4       : int  4 4 5 3 1 4 5 5 4 5 ...
##  $ FC1       : int  3 3 4 2 2 4 5 5 5 5 ...
##  $ FC2       : int  1 3 2 1 1 5 5 5 3 5 ...
##  $ FC3       : int  5 4 5 2 3 5 5 5 5 5 ...
##  $ HM1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ HM2       : int  4 5 5 4 4 5 5 5 5 5 ...
##  $ HM3       : int  4 5 5 3 4 5 5 5 5 5 ...
##  $ HA1       : int  3 4 5 2 2 5 5 5 5 5 ...
##  $ HA2       : int  3 5 5 1 2 5 4 5 5 5 ...
##  $ HA3       : int  2 5 5 1 2 5 5 5 5 5 ...
##  $ HA4       : int  2 4 5 1 2 4 5 5 5 5 ...
##  $ HA5       : int  2 4 3 1 2 5 5 5 4 4 ...
##  $ IU1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ IU2       : int  4 5 5 2 3 3 3 5 5 5 ...
##  $ U1        : int  4 5 5 4 3 5 5 5 5 5 ...
##  $ U2        : int  3 5 5 3 2 5 4 3 5 5 ...
##  $ U3        : int  1 4 4 1 1 5 5 5 5 5 ...
##  $ U4        : int  1 4 4 1 1 2 1 1 1 1 ...
##  $ TRI1      : int  4 4 4 4 3 5 5 5 5 5 ...
##  $ TRI2      : int  4 3 3 3 3 4 5 5 5 5 ...
##  $ TRI3      : int  4 4 2 3 3 4 3 4 5 5 ...
##  $ TRI4      : int  4 4 2 3 4 4 3 3 4 3 ...
##  $ TRI5      : int  2 4 4 2 2 4 1 1 3 4 ...
##  $ TRI6      : int  4 2 1 2 1 2 1 1 2 1 ...
##  $ TRI7      : int  2 2 1 1 1 2 4 4 3 3 ...
##  $ TRI8      : int  1 2 2 3 1 3 3 3 2 3 ...
##  $ TRI9      : int  4 4 4 4 5 2 3 3 2 2 ...
##  $ TRI10     : int  5 3 4 4 4 2 3 3 3 3 ...
##  $ TRI11     : int  5 4 4 4 5 2 5 5 3 2 ...
##  $ TRI12     : int  5 4 4 4 4 2 5 5 2 3 ...
##  $ TRI13     : int  4 2 2 4 4 2 5 5 5 5 ...
##  $ TRI14     : int  3 4 2 4 4 3 5 5 5 5 ...
##  $ TRI15     : int  4 5 4 4 4 4 5 5 5 5 ...
##  $ TRI16     : int  5 3 2 5 5 2 5 5 5 5 ...
##  $ EXP       : int  4 10 10 5 6 17 7 5 2 12 ...
##  $ EDU       : int  3 3 3 3 3 4 2 3 3 4 ...
##  $ SOC       : int  3 2 2 3 2 3 3 3 2 3 ...
##  $ WSTATUS   : chr  "N" "N" "Y" "N" ...
##  $ RETIRED   : chr  "Y" "Y" "N" "Y" ...
##  $ GENDER    : chr  "Female" "Male" "Female" "Female" ...
##  $ BORN      : int  1943 1952 1954 1935 1935 1960 1949 1948 1957 1954 ...
##  $ GENERATION: chr  "Silent generation " "Early Baby boomer " "Early Baby boomer " "Silent generation "
...
##  $ REGION    : chr  "BiobÃo" "BiobÃo" "BiobÃo" "BiobÃo" ...
```

Hide

```
nrow(pls_data) ## numero filas
```

```
## [1] 383
```

Hide

```
ncol(pls_data) ## numero Columnas
```

```
## [1] 54
```

**Crearemos una copia de la tabla en la que haremos los cambios**

Hide

```
pls_data2 <-pls_data
```

## A.2.2 Corrección de datos

Cambiar nombre a una variable

Hide

```
names(pls_data2)
```

```
##  [1] "ï..ID"      "PE1"        "PE2"        "PE3"        "PE4"
##  [6] "EE1"        "EE2"        "EE3"        "SI1"        "SI2"
## [11] "SI3"        "SI4"        "FC1"        "FC2"        "FC3"
## [16] "HM1"        "HM2"        "HM3"        "HA1"        "HA2"
## [21] "HA3"        "HA4"        "HA5"        "IU1"        "IU2"
## [26] "U1"         "U2"         "U3"         "U4"         "TRI1"
## [31] "TRI2"       "TRI3"       "TRI4"       "TRI5"       "TRI6"
## [36] "TRI7"       "TRI8"       "TRI9"       "TRI10"      "TRI11"
## [41] "TRI12"      "TRI13"      "TRI14"      "TRI15"      "TRI16"
## [46] "EXP"        "EDU"        "SOC"        "WSTATUS"    "RETIRED"
## [51] "GENDER"     "BORN"       "GENERATION" "REGION"
```

Hide

```
names(pls_data2)[1] = 'indice'
```

Corregir nombre de la Región Bío-Bío

Hide

```
table(pls_data2[,54])
```

```
##
##  BiobÃo Coquimbo
##      259      124
```

Hide

```
table(pls_data2$REGION)
```

```
##
##  BiobÃo Coquimbo
##      259      124
```

Hide

```
pls_data2$REGION =ifelse(pls_data2$REGION=='BiobÃo', 'Bio-Bio', pls_data2$REGION)
```

Crear o modificar el tipo de dato dejándolo como numérico

Hide

```
pls_data2$AA <- as.integer(pls_data2$BORN)  #AA es el nuevo campo a crear extraído desde BORN.

pls_data2$TRI1 <- as.integer(pls_data2$TRI1) #En este caso estamos modificamos un campo existente sin crear
         uno nuevo.
```

# B. Estadística descriptiva

## B.1 Gráficos tablas

Crear tabla de frecuencia con variable categóricas

Hide

```
table(pls_data2$EDU)
```

```
##
##   1   2   3   4
##   3  26 129 225
```

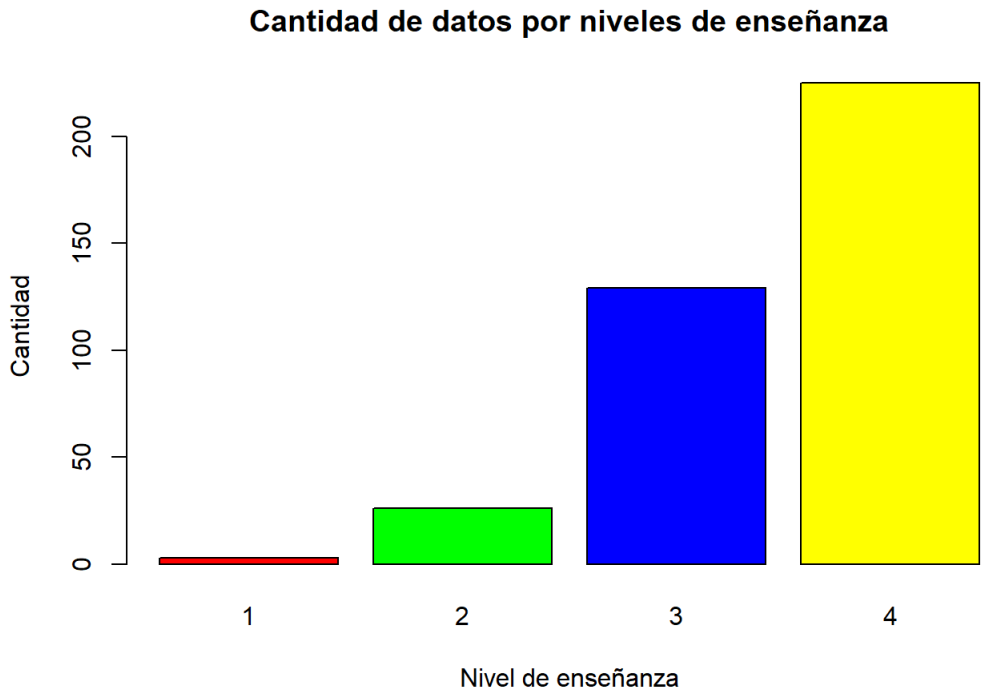
Hide

```
tab1 <- table(pls_data2$EDU)
head(tab1)
```

```
##
##   1   2   3   4
##   3  26 129 225
```

Hide

```
barplot(tab1,
        main = "Cantidad de datos por niveles de enseñanza",
        xlab = "Nivel de enseñanza",
        ylab = "Cantidad",
        col = c("red", "green", "blue", 'yellow'),
)
```




Hide

```
table(pls_data2$SOC)
```

```
##
##   1   2   3   4   5
##   7  55 253  67   1
```

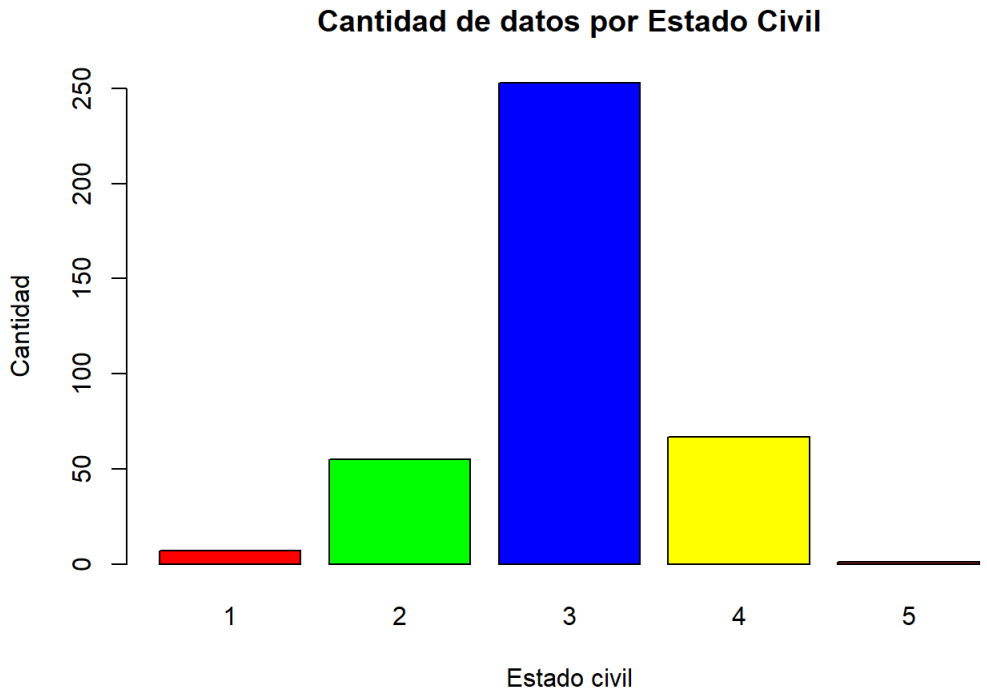
Hide

```
tab2 <- table(pls_data2$SOC)
head(tab2)
```

```
##
##   1   2   3   4   5
##   7  55 253  67   1
```

Hide

```
barplot(tab2,
        main = "Cantidad de datos por Estado Civil",
        xlab = "Estado civil",
        ylab = "Cantidad",
        col = c("red", "green", "blue", 'yellow', 'brown', 'orange'),
)
```




Hide

```
table(pls_data2$GENDER)
```

```
##
## Female   Male
##    213    170
```

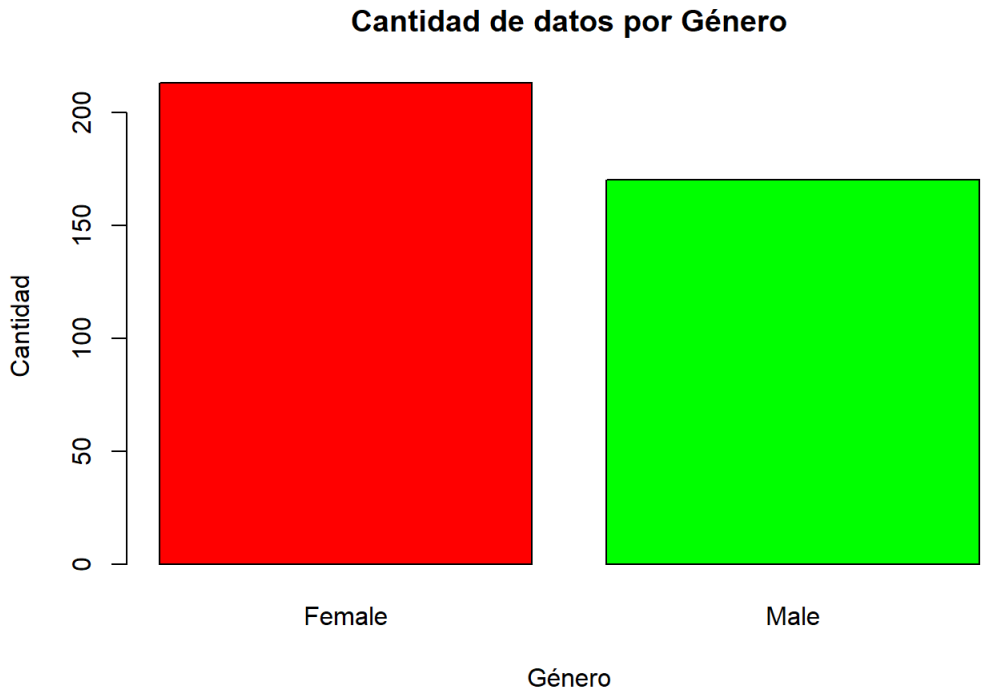
Hide

```
tab3 <- table(pls_data2$GENDER)
head(tab3)
```

```
##
## Female   Male
##    213    170
```

Hide

```
barplot(tab3,
        main = "Cantidad de datos por Género",
        xlab = "Género",
        ylab = "Cantidad",
        col = c("red", "green", "blue", 'yellow', 'brown', 'orange'),
)
```




Hide

```
table(pls_data2$GENERATION)
```

```
##
## Early Baby boomer   Late Baby boomer  Silent generation
##               128               160                95
```

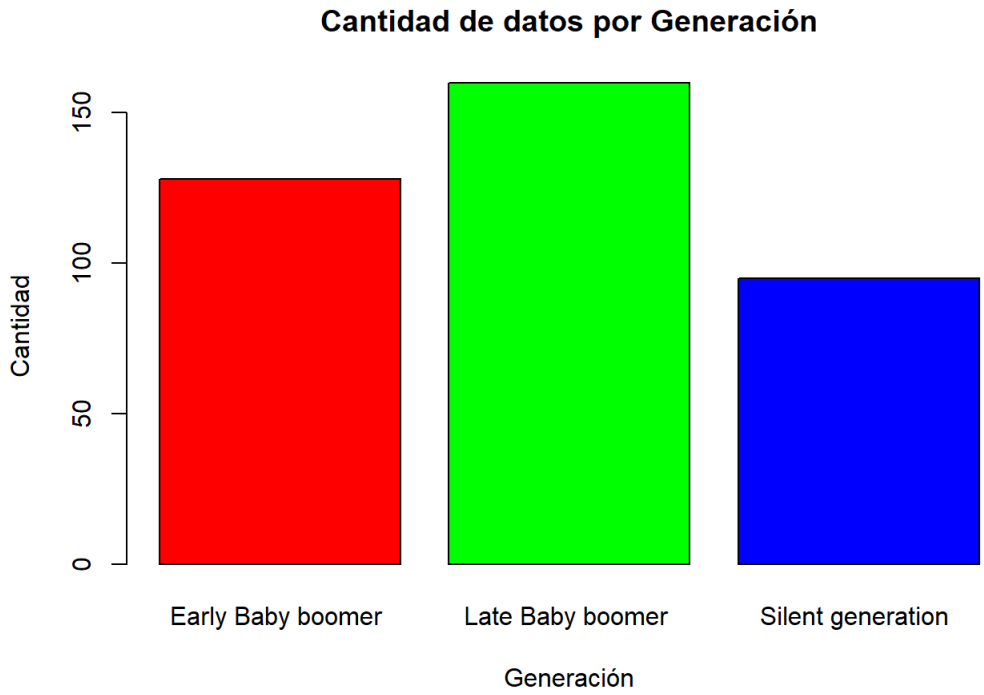
Hide

```
tab4 <- table(pls_data2$GENERATION)
head(tab4)
```

```
##
## Early Baby boomer   Late Baby boomer  Silent generation
##               128               160                95
```

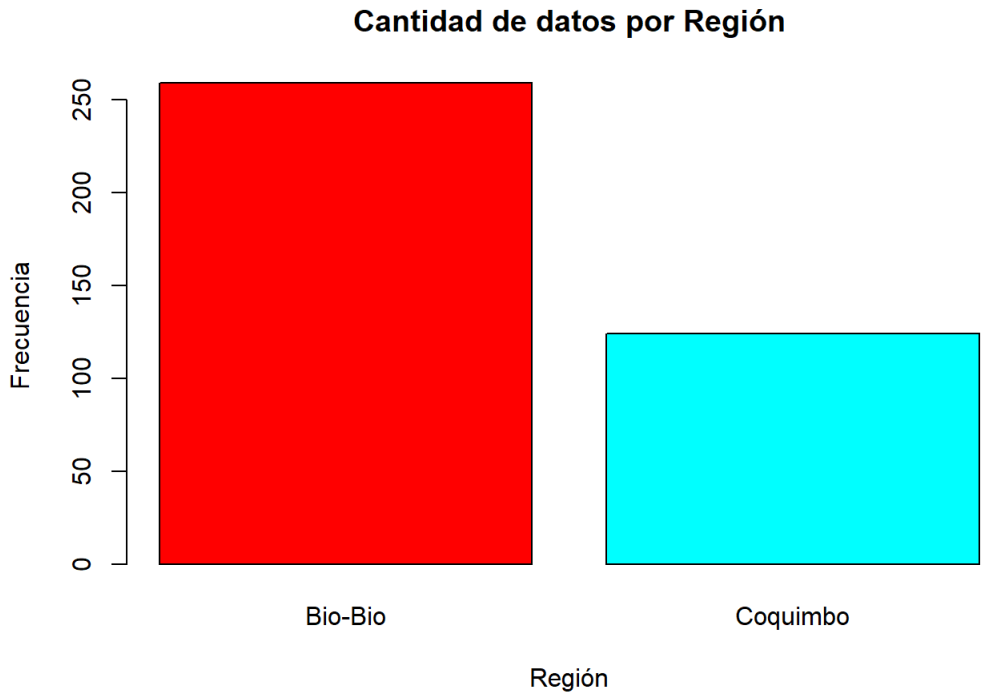
Hide

```
barplot(tab4,
        main = "Cantidad de datos por Generación",
        xlab = "Generación",
        ylab = "Cantidad",
        col = c("red", "green", "blue", 'yellow', 'brown', 'orange'),
)
```



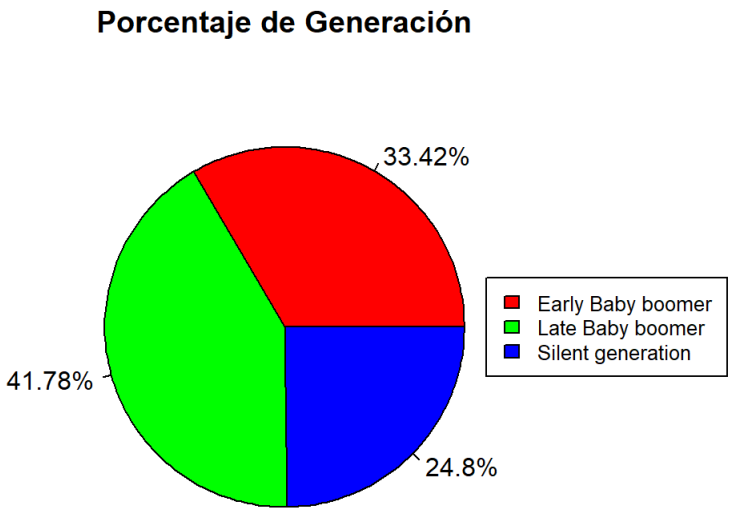

Hide

```
tab4c <- table(pls_data2$REGION)

barplot(tab4c, main = "Cantidad de datos por Región",
     xlab = "Región", ylab = "Frecuencia", col = rainbow(2))
```




Hide

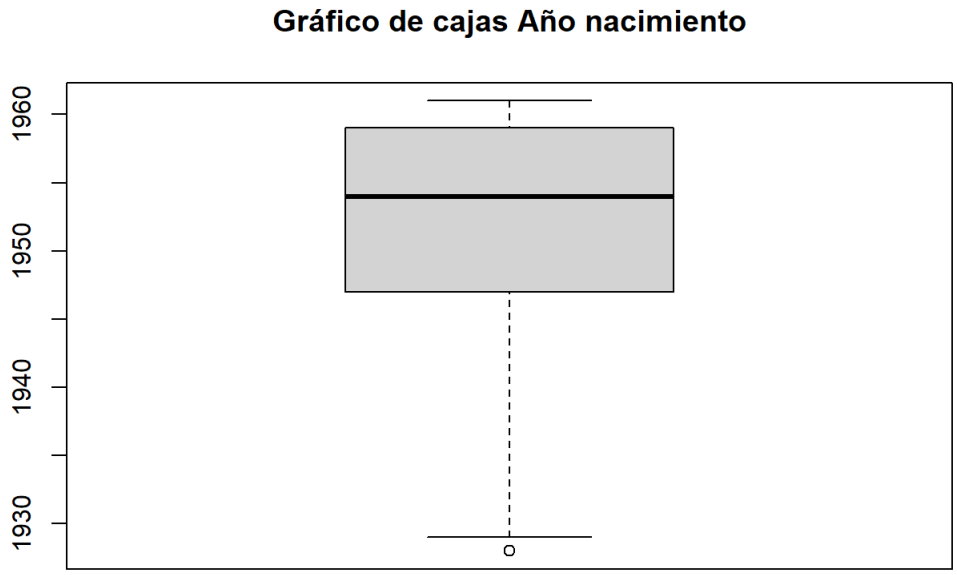
```
porcentaje <- round(tab4 / sum(tab4) * 100, 2)
colores <- rainbow(length(tab4))
pie(porcentaje, labels = paste0(porcentaje, "%"), main = "Porcentaje de Generación", col = colores)
legend("right", legend = names(tab4), cex = 0.8, fill = colores)
```




## B.1.2 Pruebas normalidad

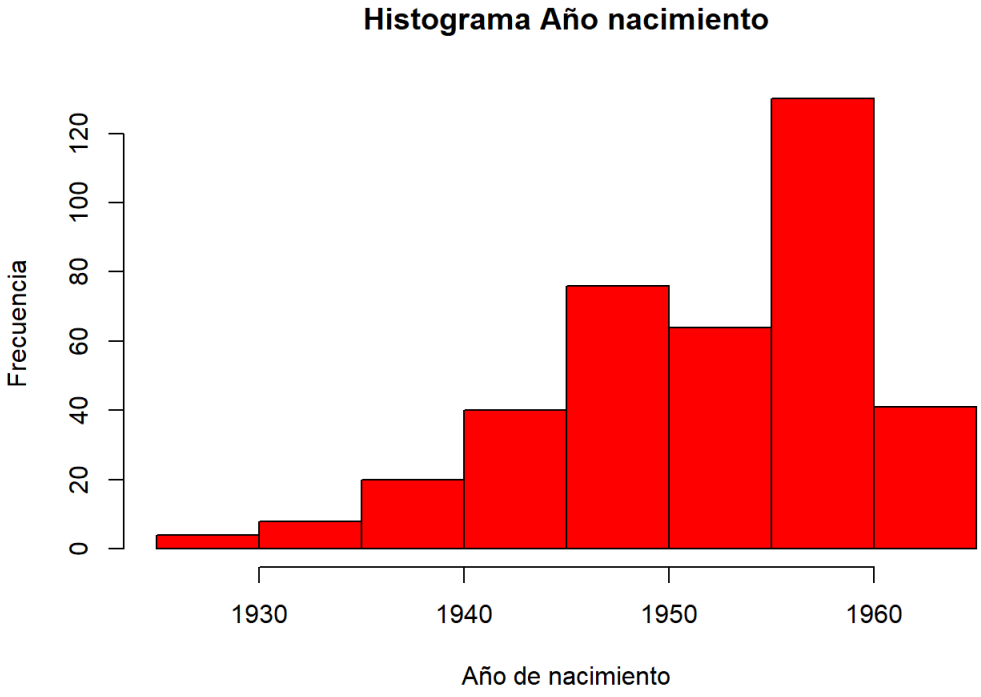
Hide

```
boxplot(pls_data2$BORN, main = "Gráfico de cajas Año nacimiento",
        outline = TRUE)
```



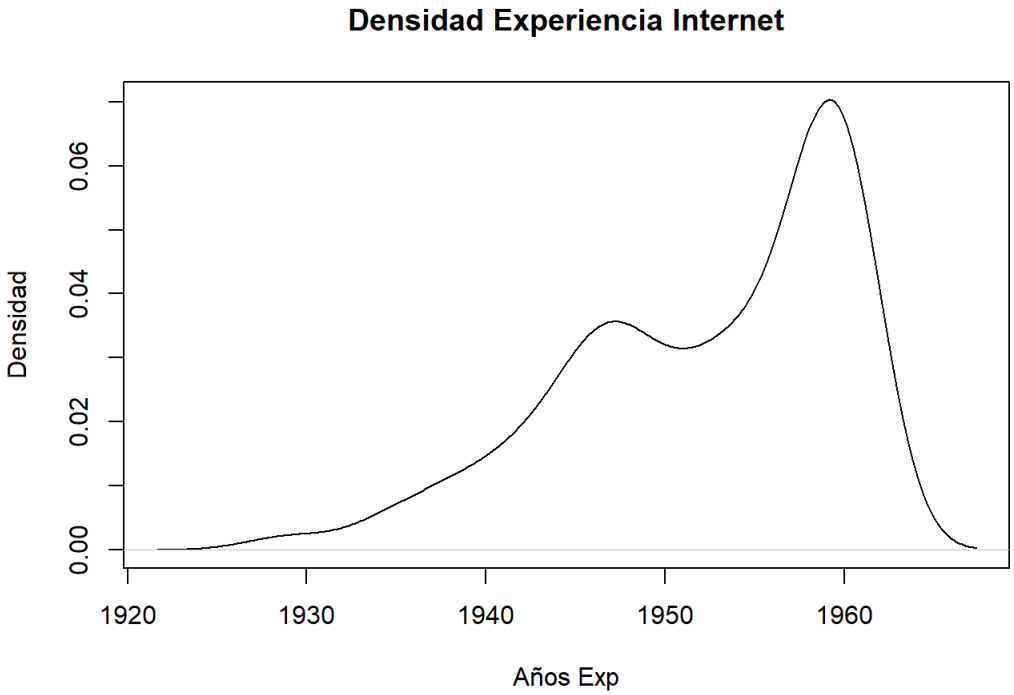

Hide

```
hist(pls_data2$BORN, main = "Histograma Año nacimiento",
     xlab = "Año de nacimiento",
     ylab = "Frecuencia",
     col = "red",
     border = "black")
```




Hide

```
densidad_BORN <- density(pls_data$BORN)
plot(densidad_BORN,
     main = "Densidad Experiencia Internet",
     xlab = "Años Exp",
     ylab = "Densidad")
```




Hide

```
skew(pls_data2$BORN,) # Simetría
```

```
## [1] -0.7942247
```

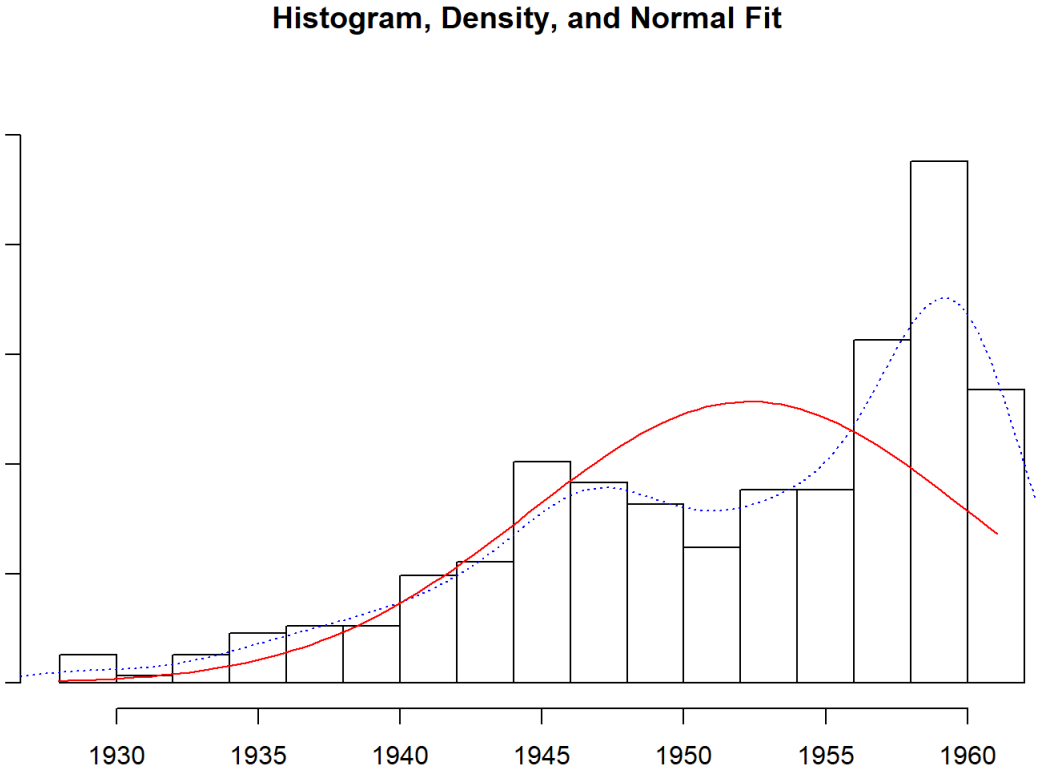
Hide

```
kurtosi(pls_data2$BORN,)
```

```
## [1] -0.1702803
```

Hide

```
multi.hist(pls_data2$BORN,dcol= c("blue","red"),dlty=c("dotted", "solid"))
```




Hide

```
# Test de Shapiro
shapiro.test(pls_data2$BORN)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  pls_data2$BORN
## W = 0.90399, p-value = 7.724e-15
```

Hide

```
# Test de kolmogorov-smirnov
ks.test(pls_data2$BORN, "pnorm", mean(pls_data2$BORN), sd(pls_data2$BORN))
```

```
## Warning in ks.test(pls_data2$BORN, "pnorm", mean(pls_data2$BORN),
## sd(pls_data2$BORN)): ties should not be present for the Kolmogorov-Smirnov test
```

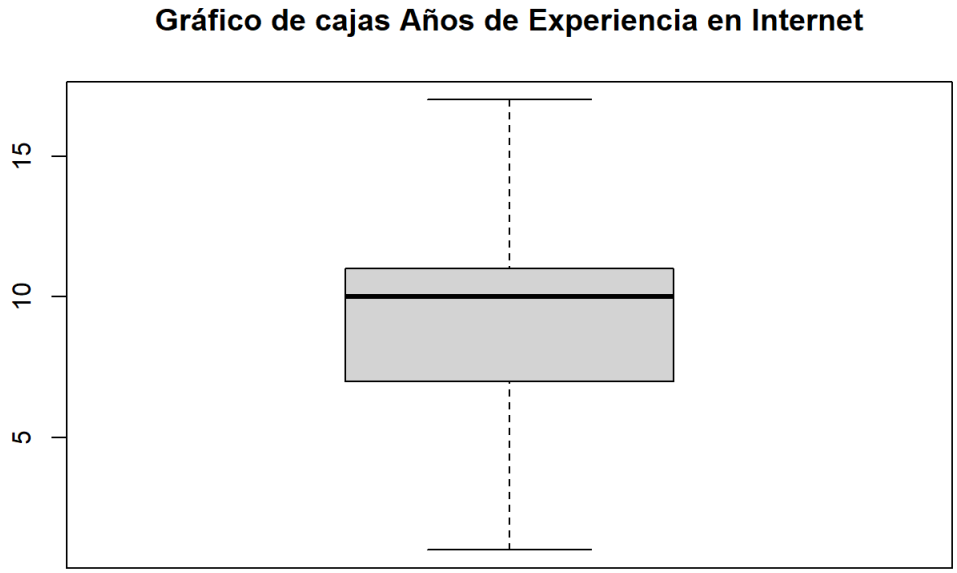
```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  pls_data2$BORN
## D = 0.16525, p-value = 1.645e-09
## alternative hypothesis: two-sided
```

Hide

```
#Con los siguientes comandos se pueden realizar pruebas adicionales de normalidad
#requiere paquete nortest

# require(nortest)
# ad.test(pls_data2$BORN) #test de Anderson-Darling
# cvm.test(pls_data2$BORN) #test de Cramer von mises
# pearson.test(pls_data2$BORN) #Chi cuadrado de pearson
```

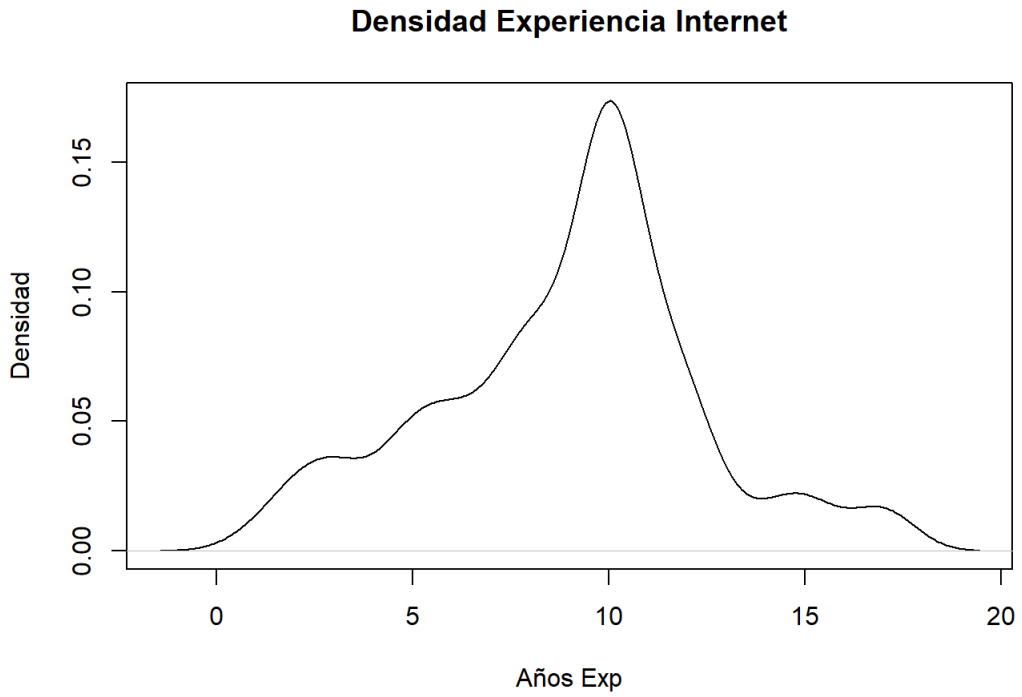
Hide

```
boxplot(pls_data2$EXP, main = "Gráfico de cajas Años de Experiencia en Internet",
        outline = TRUE)
```




Hide

```
densidad_EXP <- density(pls_data$EXP)
plot(densidad_EXP,
     main = "Densidad Experiencia Internet",
     xlab = "Años Exp",
     ylab = "Densidad")
```




Hide

```
skew(pls_data$EXP) # Simetría
```

```
## [1] -0.09126799
```

Hide

```
kurtosi(pls_data$EXP)
```

```
## [1] 0.04841059
```

## B.1.3 Otros gráficos

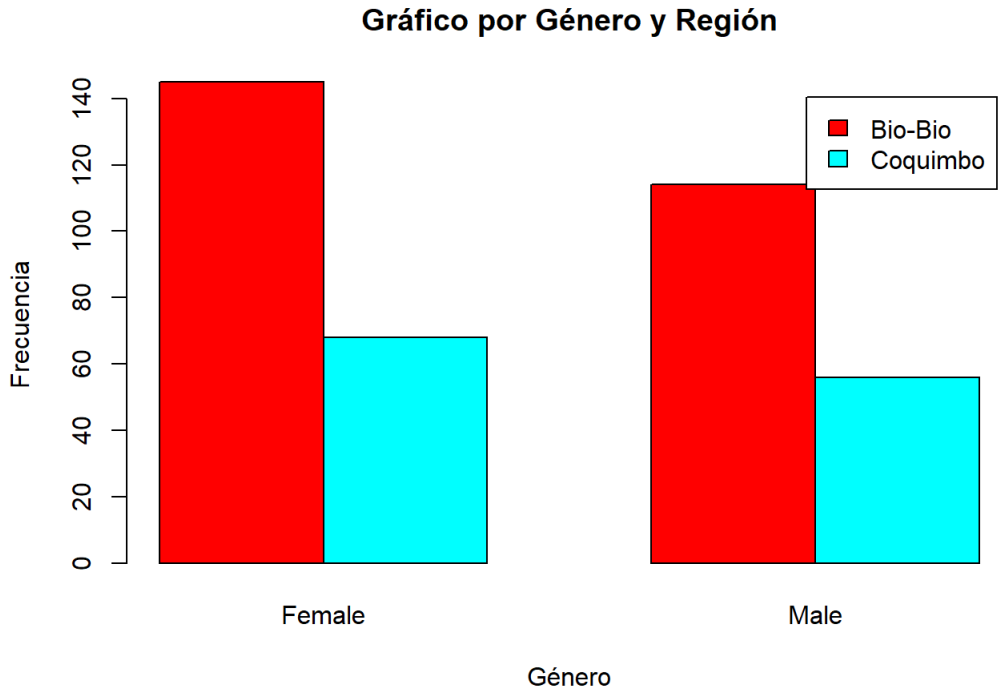
Hide

```
tabla1 <- table(pls_data2$REGION, pls_data2$GENDER)
barplot(tabla1,
        main = "Gráfico por Género y Región",
         xlab = "Género", ylab = "Frecuencia",
         legend.text = rownames(tabla1),
         beside = TRUE,
         col = rainbow(2), label = TRUE)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "label" is not a graphical
## parameter
```

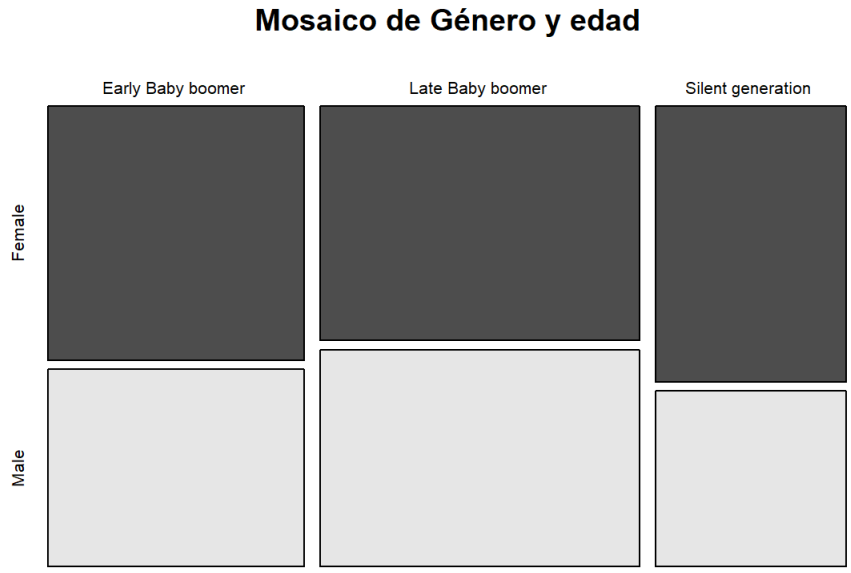
```
## Warning in axis(if (horiz) 2 else 1, at = at.l, labels = names.arg, lty =
## axis.lty, : "label" is not a graphical parameter
```

```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "label"
## is not a graphical parameter
```




Hide

```
tabla2 <- table(pls_data2$GENERATION, pls_data2$GENDER)

mosaicplot(tabla2, main = "Mosaico de Género y edad",
         color = TRUE)
```



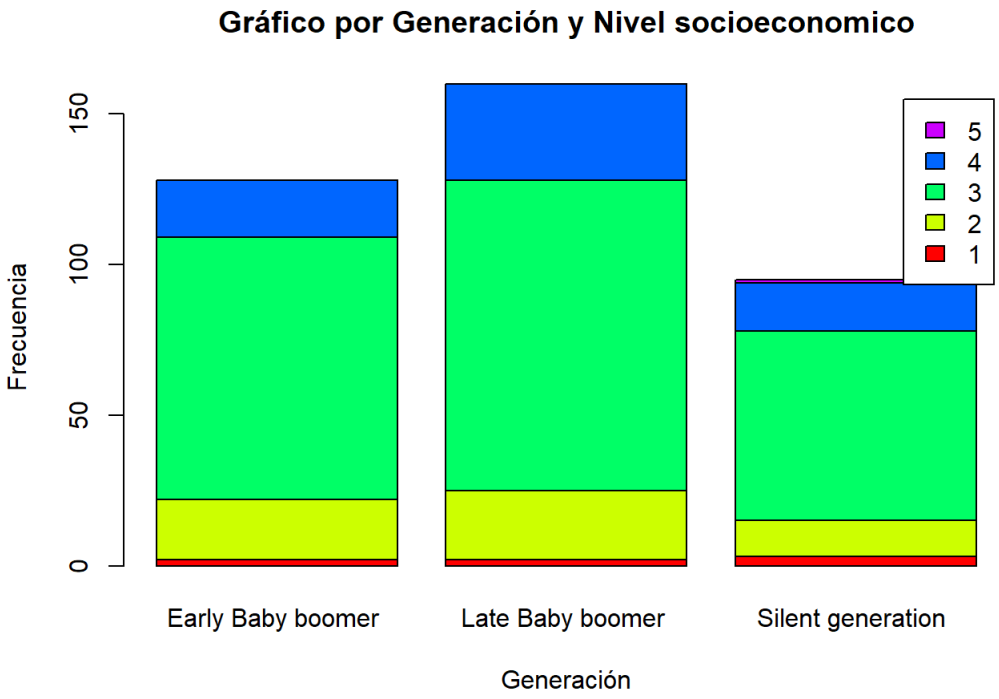
Hide

```
tabla3 <- table(pls_data2$SOC, pls_data2$GENERATION)
barplot(tabla3,
        main = "Gráfico por Generación y Nivel socioeconomico",
        xlab = "Generación", ylab = "Frecuencia",
        legend.text = rownames(tabla3),
        beside = FALSE,
         col = rainbow(5), label = TRUE)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "label" is not a graphical
## parameter
```

```
## Warning in axis(if (horiz) 2 else 1, at = at.l, labels = names.arg, lty =
## axis.lty, : "label" is not a graphical parameter
```

```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "label"
## is not a graphical parameter
```




# B.2 Separar variables categóricas

Hide

```
names(pls_data)  ### Ver los nombres de las columnas
```

```
##  [1] "ï..ID"      "PE1"        "PE2"        "PE3"        "PE4"
##  [6] "EE1"        "EE2"        "EE3"        "SI1"        "SI2"
## [11] "SI3"        "SI4"        "FC1"        "FC2"        "FC3"
## [16] "HM1"        "HM2"        "HM3"        "HA1"        "HA2"
## [21] "HA3"        "HA4"        "HA5"        "IU1"        "IU2"
## [26] "U1"         "U2"         "U3"         "U4"         "TRI1"
## [31] "TRI2"       "TRI3"       "TRI4"       "TRI5"       "TRI6"
## [36] "TRI7"       "TRI8"       "TRI9"       "TRI10"      "TRI11"
## [41] "TRI12"      "TRI13"      "TRI14"      "TRI15"      "TRI16"
## [46] "EXP"        "EDU"        "SOC"        "WSTATUS"    "RETIRED"
## [51] "GENDER"     "BORN"       "GENERATION" "REGION"
```

Hide

```
str(pls_data) ### Tipo de datos
```

```
## 'data.frame':    383 obs. of  54 variables:
##  $ ï..ID     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ PE1       : int  5 5 5 2 3 5 5 4 4 3 ...
##  $ PE2       : int  4 5 5 2 4 5 5 5 4 4 ...
##  $ PE3       : int  5 5 5 4 3 5 5 5 5 5 ...
##  $ PE4       : int  4 5 4 3 3 4 1 1 5 5 ...
##  $ EE1       : int  3 3 4 2 2 5 3 4 4 5 ...
##  $ EE2       : int  2 3 4 1 1 5 5 4 5 5 ...
##  $ EE3       : int  3 3 2 1 1 5 5 5 5 5 ...
##  $ SI1       : int  5 4 5 3 4 4 5 5 5 5 ...
##  $ SI2       : int  4 4 5 3 4 3 4 4 4 5 ...
##  $ SI3       : int  5 4 5 3 4 5 4 4 4 5 ...
##  $ SI4       : int  4 4 5 3 1 4 5 5 4 5 ...
##  $ FC1       : int  3 3 4 2 2 4 5 5 5 5 ...
##  $ FC2       : int  1 3 2 1 1 5 5 5 3 5 ...
##  $ FC3       : int  5 4 5 2 3 5 5 5 5 5 ...
##  $ HM1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ HM2       : int  4 5 5 4 4 5 5 5 5 5 ...
##  $ HM3       : int  4 5 5 3 4 5 5 5 5 5 ...
##  $ HA1       : int  3 4 5 2 2 5 5 5 5 5 ...
##  $ HA2       : int  3 5 5 1 2 5 4 5 5 5 ...
##  $ HA3       : int  2 5 5 1 2 5 5 5 5 5 ...
##  $ HA4       : int  2 4 5 1 2 4 5 5 5 5 ...
##  $ HA5       : int  2 4 3 1 2 5 5 5 4 4 ...
##  $ IU1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ IU2       : int  4 5 5 2 3 3 3 5 5 5 ...
##  $ U1        : int  4 5 5 4 3 5 5 5 5 5 ...
##  $ U2        : int  3 5 5 3 2 5 4 3 5 5 ...
##  $ U3        : int  1 4 4 1 1 5 5 5 5 5 ...
##  $ U4        : int  1 4 4 1 1 2 1 1 1 1 ...
##  $ TRI1      : int  4 4 4 4 3 5 5 5 5 5 ...
##  $ TRI2      : int  4 3 3 3 3 4 5 5 5 5 ...
##  $ TRI3      : int  4 4 2 3 3 4 3 4 5 5 ...
##  $ TRI4      : int  4 4 2 3 4 4 3 3 4 3 ...
##  $ TRI5      : int  2 4 4 2 2 4 1 1 3 4 ...
##  $ TRI6      : int  4 2 1 2 1 2 1 1 2 1 ...
##  $ TRI7      : int  2 2 1 1 1 2 4 4 3 3 ...
##  $ TRI8      : int  1 2 2 3 1 3 3 3 2 3 ...
##  $ TRI9      : int  4 4 4 4 5 2 3 3 2 2 ...
##  $ TRI10     : int  5 3 4 4 4 2 3 3 3 3 ...
##  $ TRI11     : int  5 4 4 4 5 2 5 5 3 2 ...
##  $ TRI12     : int  5 4 4 4 4 2 5 5 2 3 ...
##  $ TRI13     : int  4 2 2 4 4 2 5 5 5 5 ...
##  $ TRI14     : int  3 4 2 4 4 3 5 5 5 5 ...
##  $ TRI15     : int  4 5 4 4 4 4 5 5 5 5 ...
##  $ TRI16     : int  5 3 2 5 5 2 5 5 5 5 ...
##  $ EXP       : int  4 10 10 5 6 17 7 5 2 12 ...
##  $ EDU       : int  3 3 3 3 3 4 2 3 3 4 ...
##  $ SOC       : int  3 2 2 3 2 3 3 3 2 3 ...
##  $ WSTATUS   : chr  "N" "N" "Y" "N" ...
##  $ RETIRED   : chr  "Y" "Y" "N" "Y" ...
##  $ GENDER    : chr  "Female" "Male" "Female" "Female" ...
##  $ BORN      : int  1943 1952 1954 1935 1935 1960 1949 1948 1957 1954 ...
##  $ GENERATION: chr  "Silent generation " "Early Baby boomer " "Early Baby boomer " "Silent generation "
...
##  $ REGION    : chr  "BiobÃo" "BiobÃo" "BiobÃo" "BiobÃo" ...
```

Hide

```
categoricas <- c( 'EDU', 'SOC', 'WSTATUS', 'RETIRED', 'GENDER', 'GENERATION', 'REGION' )
otras <- c("PE1"  ,      "PE2"    ,    "PE3"      ,  "PE4"     ,   "EE1"     ,   "EE2"    ,    "EE3"   ,
         "SI1"   ,     "SI2"  ,      "SI3",
           "SI4"   ,     "FC1"   ,     "FC2"  ,     "FC3" ,     "HM1",       "HM2"  ,    "HM3"  ,    "HA1"
         ,    "HA2"  ,    "HA3"  ,     "HA4",
           "HA5"     ,   "IU1"   ,    "IU2"    ,   "U1"    ,    "U2"  ,     "U3"     ,   "U4"    ,    "TRI1"
         ,    "TRI2" ,     "TRI3",     "TRI4" ,
           "TRI5"     ,  "TRI6" ,     "TRI7"    , "TRI8"    , "TRI9"   ,   "TRI10"    ,"TRI11"    ,"TRI12"
         , "TRI13"   , "TRI14",   "TRI15",
           "TRI16")
```

# B.3 Generar resumen de campos no categóricos

Hide

```
resumen <- summary(pls_data2[,otras])
print(resumen)
```

```
##       PE1             PE2             PE3             PE4             EE1       
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.00  
##  1st Qu.:3.000   1st Qu.:4.000   1st Qu.:4.000   1st Qu.:4.000   1st Qu.:3.00  
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000   Median :4.00  
##  Mean   :3.943   Mean   :4.125   Mean   :4.407   Mean   :4.266   Mean   :3.41  
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:4.00  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.00  
##       EE2             EE3             SI1             SI2       
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:3.000   1st Qu.:4.000   1st Qu.:4.000  
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000  
##  Mean   :3.366   Mean   :3.376   Mean   :4.225   Mean   :4.172  
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:5.000   3rd Qu.:5.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000  
##       SI3             SI4             FC1             FC2             FC3       
##  Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000  
##  1st Qu.:4.000   1st Qu.:3.000   1st Qu.:4.00   1st Qu.:3.000   1st Qu.:4.000  
##  Median :4.000   Median :4.000   Median :4.00   Median :4.000   Median :4.000  
##  Mean   :4.141   Mean   :3.809   Mean   :3.99   Mean   :3.577   Mean   :4.269  
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.00   3rd Qu.:4.000   3rd Qu.:5.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000  
##       HM1             HM2             HM3             HA1             HA2       
##  Min.   :2.000   Min.   :2.000   Min.   :2.000   Min.   :1.000   Min.   :1.00  
##  1st Qu.:4.000   1st Qu.:4.000   1st Qu.:4.000   1st Qu.:3.000   1st Qu.:2.00  
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000   Median :4.00  
##  Mean   :4.021   Mean   :4.052   Mean   :3.984   Mean   :3.543   Mean   :3.36  
##  3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:4.00  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.00  
##       HA3             HA4             HA5             IU1       
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:4.000  
##  Median :4.000   Median :3.000   Median :4.000   Median :4.000  
##  Mean   :3.593   Mean   :3.112   Mean   :3.355   Mean   :4.358  
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:5.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000  
##       IU2              U1              U2             U3              U4        
##  Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.00   1st Qu.:3.000   1st Qu.:1.000  
##  Median :4.000   Median :4.000   Median :4.00   Median :3.000   Median :2.000  
##  Mean   :3.969   Mean   :3.961   Mean   :3.94   Mean   :3.366   Mean   :2.352  
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.00   3rd Qu.:4.000   3rd Qu.:3.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000  
##       TRI1            TRI2            TRI3            TRI4            TRI5      
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.00  
##  1st Qu.:4.000   1st Qu.:4.000   1st Qu.:3.000   1st Qu.:3.000   1st Qu.:2.00  
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000   Median :3.00  
##  Mean   :4.084   Mean   :4.029   Mean   :3.869   Mean   :3.791   Mean   :2.71  
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:4.00  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.00  
##       TRI6            TRI7            TRI8           TRI9           TRI10     
##  Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000  
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000   1st Qu.:3.000  
##  Median :2.000   Median :3.000   Median :3.00   Median :4.000   Median :4.000  
##  Mean   :2.326   Mean   :2.875   Mean   :3.18   Mean   :3.261   Mean   :3.452  
##  3rd Qu.:3.000   3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000  
##      TRI11           TRI12           TRI13           TRI14       
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.500  
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000  
```

```
##  Mean   :3.376   Mean   :3.554   Mean   :3.841   Mean   :3.901
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:5.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000
##      TRI15           TRI16
##  Min.   :1.000   Min.   :1.000
##  1st Qu.:3.000   1st Qu.:2.000
##  Median :4.000   Median :4.000
##  Mean   :3.883   Mean   :3.308
##  3rd Qu.:5.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000
```

Exportar a Excel con datos resumen

Hide

```
write.xlsx2(x=resumen,
            'resumen.xlsx',
            sheetName = "resumen",
            col.names = TRUE,
            row.names = TRUE,
            append = FALSE,
            showNA = TRUE,
            password = NULL)
```

# B.4 Tablas de frecuencias

Hide

```
xtabs(~EDU + GENDER, data =pls_data2) ## Educación y Género
```

```
##    GENDER
## EDU Female Male
##   1      2    1
##   2     17    9
##   3     79   50
##   4    115  110
```

Hide

```
xtabs(~GENDER + WSTATUS, data =pls_data2) ##Género y Estatus Laboral
```

```
##         WSTATUS
## GENDER     N   Y
##   Female 128  85
##   Male    63 107
```

Hide

```
xtabs(~GENDER + RETIRED, data =pls_data2) ##Género y Retirado
```

```
##         RETIRED
## GENDER     N   Y
##   Female  70 143
##   Male    80  90
```

Hide

```
xtabs(~GENDER + GENERATION, data =pls_data2) ## Género y Generación
```

```
##         GENERATION
## GENDER   Early Baby boomer  Late Baby boomer  Silent generation
##   Female                 72                83                 58
##   Male                   56                77                 37
```

Hide

```
xtabs(~GENDER + REGION, data =pls_data2) ##Género y Región
```

```
##         REGION
## GENDER   Bio-Bio Coquimbo
##   Female     145       68
##   Male       114       56
```

Hide

```
xtabs(~REGION + GENERATION, data =pls_data2) ##Género y Región
```

```
##           GENERATION
## REGION     Early Baby boomer  Late Baby boomer  Silent generation
##   Bio-Bio                  87               109                 63
##   Coquimbo                 41                51                 32
```

# C. Datos Faltantes

## C.1 Obtener columnas con datos faltantes

Si aparece list() no hay datos faltantes

Hide

```
nan <- function(df) {
  nulos <- list()
  for (i in 1:length(df)) {
    if (sum(is.na(df[[i]])) != 0) {
      nulos[[length(nulos) + 1]] <- c(names(df)[i], sum(is.na(df[[i]])))
    }
  }
  print(nulos)
}

nan(pls_data)
```

```
## list()
```

## C.2 Reemplazar datos faltantes por -99

Si cambia a un valor distinto, luego al estimar modelo cambiar.

Hide

```
reemp_falt <- function(df) {
  for (i in 1:length(df)) {
    if (sum(is.na(df[[i]])) != 0) {
      df[[i]] <- replace(df[[i]], is.na(df[[i]]), -99)
    }
  }
  return(df)
}

pls_data2 <-reemp_falt(pls_data2)
```

# C.3 Eliminar datos faltantes o con una condición

Eliminar los que se desea

Hide

```
pls_data2$PE1 == "-99"
```

```
##   [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [97] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [193] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [205] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [229] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [241] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [253] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [265] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [277] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [289] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [301] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [313] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [325] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [337] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [349] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [361] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [373] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

Hide

```
pls_data2 <- pls_data2[(pls_data2$PE1 != "-99"),]
```

# D. Crear una variable categórica desde otra variable

# D.1 Convertir en categóricas

Hide

```
str(pls_data2)
```

```
## 'data.frame':    383 obs. of  55 variables:
##  $ indice    : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ PE1       : int  5 5 5 2 3 5 5 4 4 3 ...
##  $ PE2       : int  4 5 5 2 4 5 5 5 4 4 ...
##  $ PE3       : int  5 5 5 4 3 5 5 5 5 5 ...
##  $ PE4       : int  4 5 4 3 3 4 1 1 5 5 ...
##  $ EE1       : int  3 3 4 2 2 5 3 4 4 5 ...
##  $ EE2       : int  2 3 4 1 1 5 5 4 5 5 ...
##  $ EE3       : int  3 3 2 1 1 5 5 5 5 5 ...
##  $ SI1       : int  5 4 5 3 4 4 5 5 5 5 ...
##  $ SI2       : int  4 4 5 3 4 3 4 4 4 5 ...
##  $ SI3       : int  5 4 5 3 4 5 4 4 4 5 ...
##  $ SI4       : int  4 4 5 3 1 4 5 5 4 5 ...
##  $ FC1       : int  3 3 4 2 2 4 5 5 5 5 ...
##  $ FC2       : int  1 3 2 1 1 5 5 5 3 5 ...
##  $ FC3       : int  5 4 5 2 3 5 5 5 5 5 ...
##  $ HM1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ HM2       : int  4 5 5 4 4 5 5 5 5 5 ...
##  $ HM3       : int  4 5 5 3 4 5 5 5 5 5 ...
##  $ HA1       : int  3 4 5 2 2 5 5 5 5 5 ...
##  $ HA2       : int  3 5 5 1 2 5 4 5 5 5 ...
##  $ HA3       : int  2 5 5 1 2 5 5 5 5 5 ...
##  $ HA4       : int  2 4 5 1 2 4 5 5 5 5 ...
##  $ HA5       : int  2 4 3 1 2 5 5 5 4 4 ...
##  $ IU1       : int  4 5 5 3 3 5 5 5 5 5 ...
##  $ IU2       : int  4 5 5 2 3 3 3 5 5 5 ...
##  $ U1        : int  4 5 5 4 3 5 5 5 5 5 ...
##  $ U2        : int  3 5 5 3 2 5 4 3 5 5 ...
##  $ U3        : int  1 4 4 1 1 5 5 5 5 5 ...
##  $ U4        : int  1 4 4 1 1 2 1 1 1 1 ...
##  $ TRI1      : int  4 4 4 4 3 5 5 5 5 5 ...
##  $ TRI2      : int  4 3 3 3 3 4 5 5 5 5 ...
##  $ TRI3      : int  4 4 2 3 3 4 3 4 5 5 ...
##  $ TRI4      : int  4 4 2 3 4 4 3 3 4 3 ...
##  $ TRI5      : int  2 4 4 2 2 4 1 1 3 4 ...
##  $ TRI6      : int  4 2 1 2 1 2 1 1 2 1 ...
##  $ TRI7      : int  2 2 1 1 1 2 4 4 3 3 ...
##  $ TRI8      : int  1 2 2 3 1 3 3 3 2 3 ...
##  $ TRI9      : int  4 4 4 4 5 2 3 3 2 2 ...
##  $ TRI10     : int  5 3 4 4 4 2 3 3 3 3 ...
##  $ TRI11     : int  5 4 4 4 5 2 5 5 3 2 ...
##  $ TRI12     : int  5 4 4 4 4 2 5 5 2 3 ...
##  $ TRI13     : int  4 2 2 4 4 2 5 5 5 5 ...
##  $ TRI14     : int  3 4 2 4 4 3 5 5 5 5 ...
##  $ TRI15     : int  4 5 4 4 4 4 5 5 5 5 ...
##  $ TRI16     : int  5 3 2 5 5 2 5 5 5 5 ...
##  $ EXP       : int  4 10 10 5 6 17 7 5 2 12 ...
##  $ EDU       : int  3 3 3 3 3 4 2 3 3 4 ...
##  $ SOC       : int  3 2 2 3 2 3 3 3 2 3 ...
##  $ WSTATUS   : chr  "N" "N" "Y" "N" ...
##  $ RETIRED   : chr  "Y" "Y" "N" "Y" ...
##  $ GENDER    : chr  "Female" "Male" "Female" "Female" ...
##  $ BORN      : int  1943 1952 1954 1935 1935 1960 1949 1948 1957 1954 ...
##  $ GENERATION: chr  "Silent generation " "Early Baby boomer " "Early Baby boomer " "Silent generation "
...
##  $ REGION    : chr  "Bio-Bio" "Bio-Bio" "Bio-Bio" "Bio-Bio" ...
##  $ AA        : int  1943 1952 1954 1935 1935 1960 1949 1948 1957 1954 ...
```

Hide

```
pls_data2$EDU3=pls_data2$EDU
pls_data2$SOC3= pls_data2$SOC
pls_data2$EXP3= pls_data2$EXP
pls_data2$EDU2= as.factor(pls_data2$EDU)
pls_data2$SOC2= as.factor(pls_data2$SOC)
pls_data2$EXP2= as.factor(pls_data2$EXP)
```

## D.2 Cambiar variables categóricas

Hide

```
pls_data2$GENERO= ifelse(pls_data2$GENDER=='Male', 1, 2)
pls_data2$REGION3= ifelse(pls_data2$REGION=='Coquimbo', 1, 2)
pls_data2$WSTATUS3= ifelse(pls_data2$WSTATUS=='N', 1, 2)
pls_data2$RETIRED3= ifelse(pls_data2$RETIRED=='N', 1, 2)
pls_data2$GENERATION3= ifelse(pls_data2$GENERATION=="Silent generation ", 1,pls_data2$GENERATION)
pls_data2$GENERATION3= ifelse(pls_data2$GENERATION=="Late Baby boomer ", 3, pls_data2$GENERATION3)
pls_data2$GENERATION3= ifelse(pls_data2$GENERATION=="Early Baby boomer ", 2,pls_data2$GENERATION3)
pls_data2$GENERATION3 <- as.numeric(pls_data2$GENERATION3)
#pls_data2$WSTATUS2= as.factor(pls_data2$WSTATUS)
#pls_data2$REGION2= as.factor(pls_data2$REGION)
pls_data2$GENERO3= pls_data2$GENERO
#pls_data2$GENERATION2= as.factor(pls_data2$GENERATION)
#pls_data2$RETIRED2 = as.factor(pls_data2$RETIRED)
```

# E. Modelo de ecuaciones estructurales (semir)

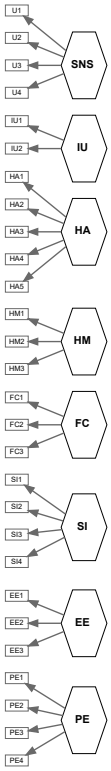
## E.1 Crear el modelo de medida

Por defecto se crean como reflectivo, para crear formativo agregar “weights = mode_B”

### E.1.1 PLS Normal

Hide

```
## Reflectivo = mode_A  (default)
## Formativo  = mode_B  (weights = mode_B)


modelo_medida <- constructs(
  composite('PE', multi_items('PE', 1:4), weights = mode_A),
  composite('EE', multi_items('EE', 1:3)),
  composite('SI', multi_items('SI', 1:4)),
  composite('FC', multi_items('FC', 1:3)),
  composite('HM', multi_items('HM', 1:3)),
  composite('HA', multi_items('HA', 1:5)),
# composite('CUSA', single_item('cusa')),  # En el caso de ser un unico item dejar como single_item
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4))
 )

plot(modelo_medida)
```



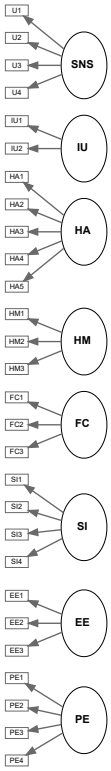

Hide

```
save_plot("modelo_medida.pdf")
```

## E.1.2 PLS Consistente

Hide

```
modelo_medida <- constructs(
  reflective('PE', multi_items('PE', 1:4)),
  reflective('EE', multi_items('EE', 1:3)),
  reflective('SI', multi_items('SI', 1:4)),
  reflective('FC', multi_items('FC', 1:3)),
  reflective('HM', multi_items('HM', 1:3)),
  reflective('HA', multi_items('HA', 1:5)),
# composite('CUSA', single_item('cusa')),  # En el caso de ser un unico item dejar como single_item
  reflective('IU', multi_items('IU', 1:2)),
  reflective('SNS', multi_items('U', 1:4))
 )

plot(modelo_medida)
```




Hide

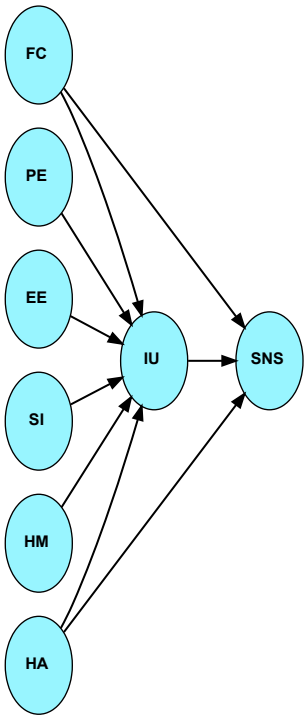
```
save_plot("modelo_medida.pdf")
```

# E.2 Crear Modelo estructural

Hide

```
modelo_estruc <- relationships(
  paths(from = c('PE', 'EE', 'SI', 'FC', 'HM', "HA"), to = c('IU')),
  paths(from = c('FC', 'HA', "IU"), to = c('SNS'))
)

## ----- Generamos el modelo con colores
thm <- seminr_theme_create(plot.rounding = 2,  ## Decimales
                           plot.adj = FALSE,
                           sm.node.fill = "cadetblue1",
                           mm.node.fill = "lightgray",
                           )

seminr_theme_set(thm)
## ----
plot(modelo_estruc, title =  "Fig. 1: Modelo Estructural")
```

Fig. 1: Modelo Estructural




Hide

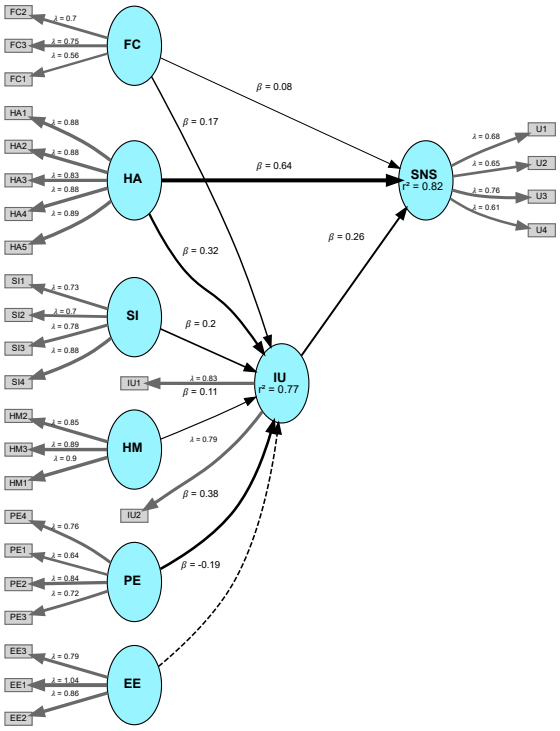
```
save_plot("fig1.Modelo_Estructural.pdf")
```

# E.3 Estimación del Modelo

Hide

```
estimacion_model <- estimate_pls(data = pls_data2,
                                 measurement_model = modelo_medida,  #Constructos
                                 structural_model = modelo_estruc,   # Caminos Path
                                 inner_weights = path_weighting,
                                 # path_weighting para path weighting (default) o path_factorial para f
        actor weighting,
                                 missing = mean_replacement, #Reemplazar los valores perdido mean es de
        fault
                                 missing_value = '-99' ) # Valores perdidos

summary_estimacion_model <- summary(estimacion_model)


plot(estimacion_model, title =  "Fig. 2: Modelo Estimado")
```

Fig. 2: Modelo Estimado




Hide

```
save_plot("fig2.Modelo_Estimado.pdf")
```

# E.4 Reportes modelo

## E.4.1. Valores perdidos y estadísticas de cada variable

Hide

```
summary_estimacion_model$descriptives$statistics  ## Valores perdidos y representación
```

```
## $items
##        No. Missing  Mean Median   Min   Max Std.Dev. Kurtosis Skewness
## PE1  1.000   0.000 3.943  4.000 1.000 5.000    0.969    3.276   -0.853
## PE2  2.000   0.000 4.125  4.000 1.000 5.000    0.877    4.645   -1.178
## PE3  3.000   0.000 4.407  4.000 1.000 5.000    0.652    5.589   -1.157
## PE4  4.000   0.000 4.266  4.000 1.000 5.000    0.692    5.197   -0.976
## EE1  5.000   0.000 3.410  4.000 1.000 5.000    1.119    2.373   -0.508
## EE2  6.000   0.000 3.366  4.000 1.000 5.000    1.105    2.362   -0.455
## EE3  7.000   0.000 3.376  4.000 1.000 5.000    1.095    2.397   -0.473
## SI1  8.000   0.000 4.225  4.000 1.000 5.000    0.757    4.261   -0.978
## SI2  9.000   0.000 4.172  4.000 1.000 5.000    0.750    4.039   -0.852
## SI3 10.000   0.000 4.141  4.000 1.000 5.000    0.803    4.673   -1.079
## SI4 11.000   0.000 3.809  4.000 1.000 5.000    1.118    2.884   -0.780
## FC1 12.000   0.000 3.990  4.000 1.000 5.000    0.850    3.698   -0.877
## FC2 13.000   0.000 3.577  4.000 1.000 5.000    1.043    2.815   -0.711
## FC3 14.000   0.000 4.269  4.000 1.000 5.000    0.677    6.450   -1.196
## HM1 15.000   0.000 4.021  4.000 2.000 5.000    0.765    2.954   -0.491
## HM2 16.000   0.000 4.052  4.000 2.000 5.000    0.703    3.590   -0.570
## HM3 17.000   0.000 3.984  4.000 2.000 5.000    0.812    2.924   -0.559
## HA1 18.000   0.000 3.543  4.000 1.000 5.000    1.113    2.126   -0.422
## HA2 19.000   0.000 3.360  4.000 1.000 5.000    1.213    1.897   -0.214
## HA3 20.000   0.000 3.593  4.000 1.000 5.000    1.098    2.209   -0.462
## HA4 21.000   0.000 3.112  3.000 1.000 5.000    1.264    1.790    0.084
## HA5 22.000   0.000 3.355  4.000 1.000 5.000    1.230    1.931   -0.285
## IU1 23.000   0.000 4.358  4.000 1.000 5.000    0.663    6.294   -1.194
## IU2 24.000   0.000 3.969  4.000 1.000 5.000    0.968    2.978   -0.771
## U1  25.000   0.000 3.961  4.000 1.000 5.000    1.112    3.005   -0.895
## U2  26.000   0.000 3.940  4.000 1.000 5.000    1.002    2.945   -0.661
## U3  27.000   0.000 3.366  3.000 1.000 5.000    1.262    2.244   -0.344
## U4  28.000   0.000 2.352  2.000 1.000 5.000    1.267    2.086    0.468
##
## $constructs
##       No. Missing   Mean Median    Min   Max Std.Dev. Kurtosis Skewness
## PE  1.000   0.000  0.000 -0.159 -3.712 1.239    1.000    3.223   -0.596
## EE  2.000   0.000 -0.000  0.283 -2.302 1.558    1.000    2.491   -0.531
## SI  3.000   0.000  0.000 -0.141 -2.967 1.272    1.000    2.512   -0.309
## FC  4.000   0.000  0.000 -0.005 -4.152 1.514    1.000    4.501   -0.795
## HM  5.000   0.000  0.000 -0.029 -2.893 1.403    1.000    2.912   -0.346
## HA  6.000   0.000 -0.000  0.034 -2.255 1.504    1.000    2.047   -0.119
## IU  7.000   0.000  0.000 -0.286 -4.490 1.116    1.000    4.035   -0.789
## SNS 8.000   0.000  0.000 -0.081 -2.779 1.721    1.000    2.274   -0.105
```

Hide

```
x <- summary_estimacion_model$descriptives$statistics
write.xlsx2(x=x["items"],
            'resumen.xlsx',
            sheetName = "resumen_hor",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)

write.xlsx2(x=x["constructs"],
            'resumen.xlsx',
            sheetName = "resumen_hor_const",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.2. Número de iteraciones

**Nota***:* Si es mayor a 300 significa que no converge

Hide

```
summary_estimacion_model$iterations
```

```
## [1] 5
```

## E.4.3. R^2

**Exógenos**

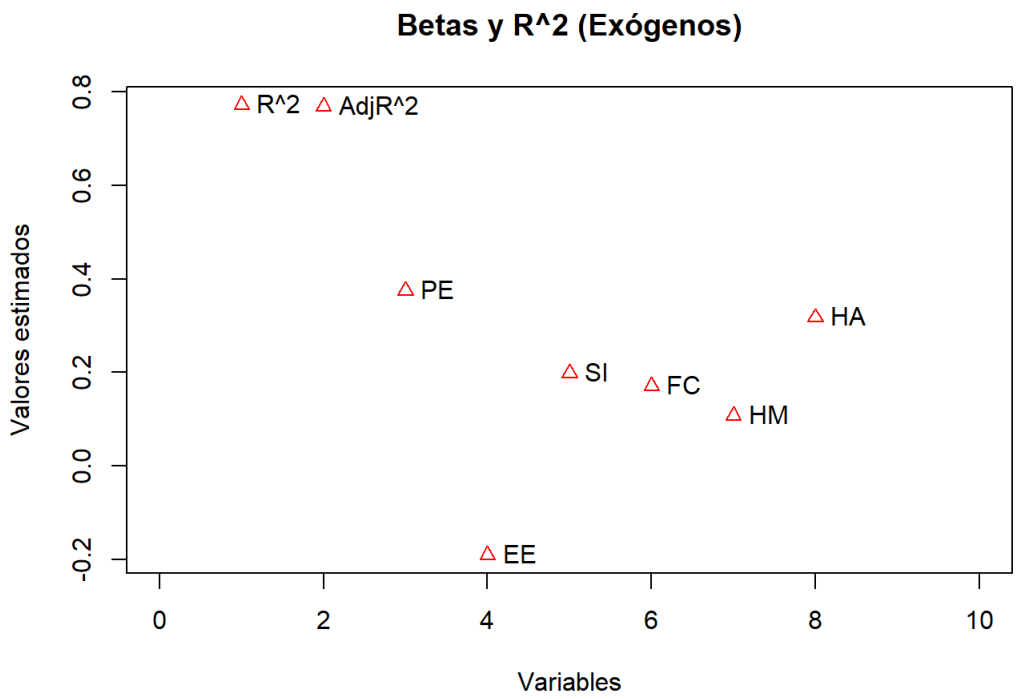
Hide

```
summary_estimacion_model$paths
```

```
##            IU   SNS
## R^2     0.772 0.816
## AdjR^2  0.768 0.814
## PE      0.375     .
## EE     -0.190     .
## SI      0.198     .
## FC      0.171 0.079
## HM      0.108     .
## HA      0.317 0.636
## IU          . 0.257
```

Hide

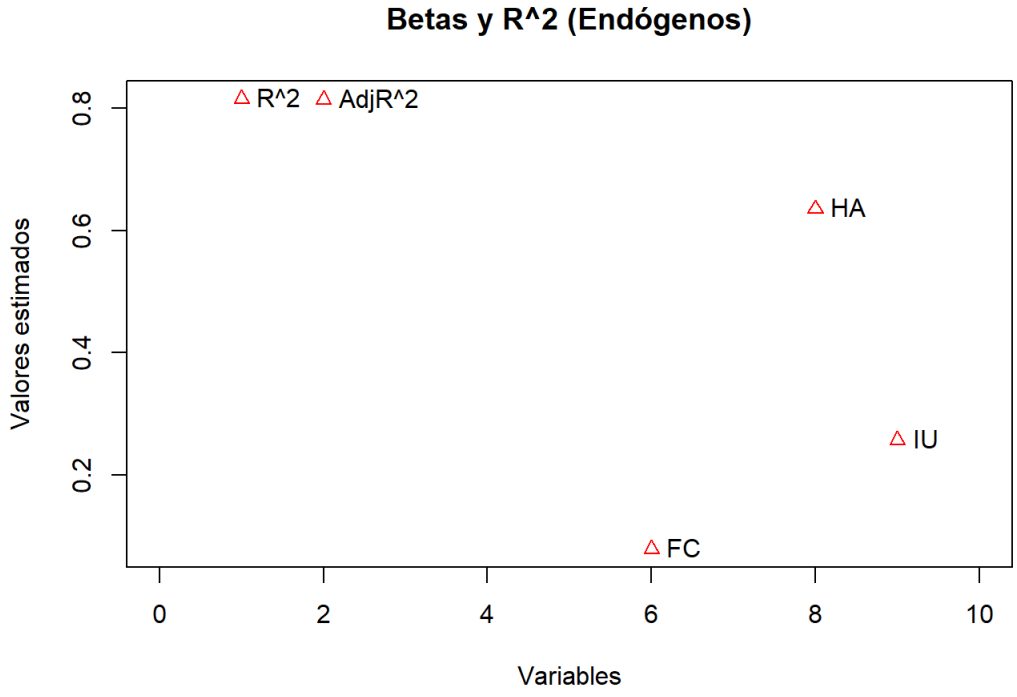
```
plot(summary_estimacion_model$paths[,1], pch = 2, col = "red", main="Betas y R^2 (Exógenos)",
     xlab = "Variables", ylab = "Valores estimados", xlim = c(0,length(row.names(summary_estimacion_model$pa
         ths))+1)
     )
text(summary_estimacion_model$paths[,1],labels = row.names(summary_estimacion_model$paths) , pos = 4)
```




**Endógenos**

Hide

```
plot(summary_estimacion_model$paths[,2], pch = 2, col = "red", main="Betas y R^2 (Endógenos)",
     xlab = "Variables", ylab = "Valores estimados" , xlim = c(0,length(row.names(summary_estimacion_model$p
         aths))+1) )
text(summary_estimacion_model$paths[,2],labels = row.names(summary_estimacion_model$paths) , pos = 4)
```




Exportar Excel

Hide

```
write.xlsx2(x=summary_estimacion_model$paths,
            'resumen.xlsx',
            sheetName = "BetasyR",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.4. Fiabilidad

Cronbach's alpha (alpha), composite reliability (rhoC), average variance extracted (AVE),

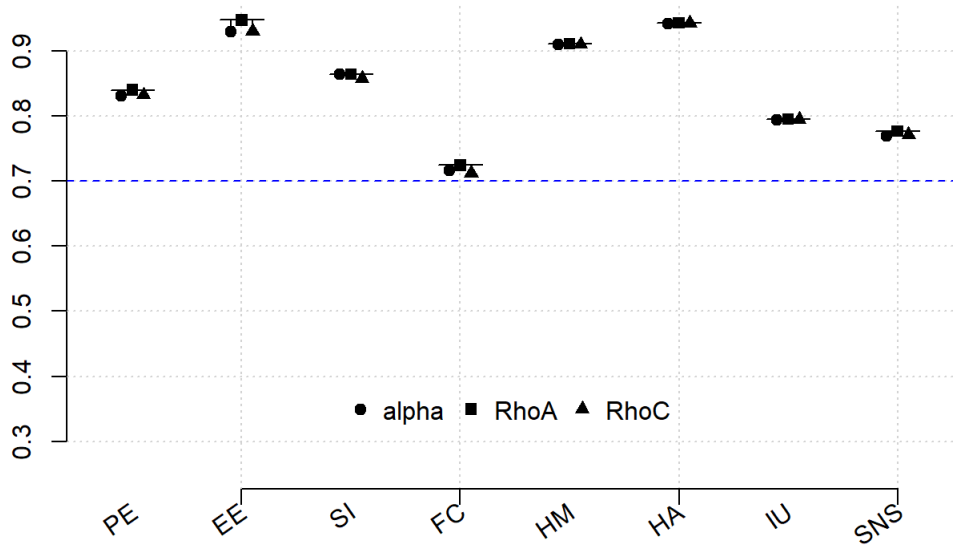
Hide

```
summary_estimacion_model$reliability
```

```
##     alpha  rhoC   AVE  rhoA
## PE  0.831 0.832 0.556 0.840
## EE  0.930 0.930 0.818 0.947
## SI  0.864 0.857 0.602 0.864
## FC  0.716 0.712 0.455 0.724
## HM  0.910 0.910 0.771 0.911
## HA  0.942 0.942 0.765 0.943
## IU  0.794 0.794 0.659 0.795
## SNS 0.769 0.771 0.459 0.776
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5
```

Hide

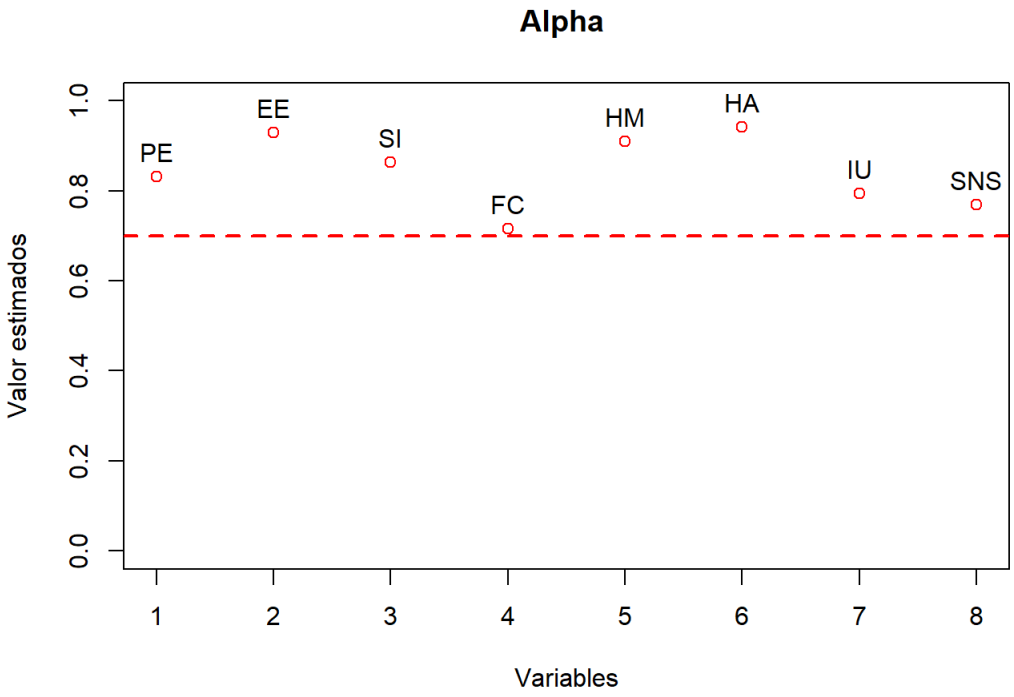
```
plot(summary_estimacion_model$reliability, title =  "Fig. 3: Fiabilidad")
```




Alpha

Hide

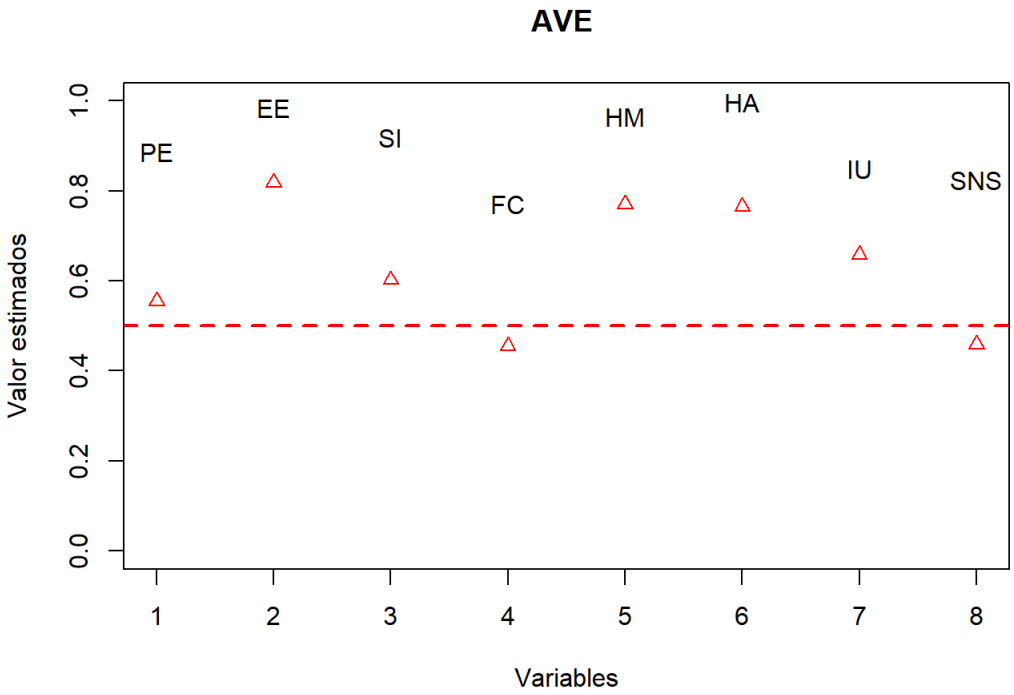
```
plot(summary_estimacion_model$reliability[,1], pch = 1, col = "red", main="Alpha ",
     xlab = "Variables", ylab = "Valor estimados",  ylim = c(0, 1))
text(summary_estimacion_model$reliability[,1],labels = row.names(summary_estimacion_model$reliability) , pos
         = 3)
abline(h=0.7,col="red",lty=2,lwd=2)
```




AVE

Hide

```
plot(summary_estimacion_model$reliability[,3], pch = 2, col = "red", main="AVE ",
     xlab = "Variables", ylab = "Valor estimados",  ylim = c(0, 1))
text(summary_estimacion_model$reliability[,1],labels = row.names(summary_estimacion_model$reliability) , pos
         = 3)
abline(h=0.5,col="red",lty=2,lwd=2)
```




Exportar a Excel

Hide

```
write.xlsx2(x=summary_estimacion_model$reliability,
            'resumen.xlsx',
            sheetName = "reliability",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.5. Cargas

Hide

```
summary_estimacion_model$loadings # Cargas -> reflectivas mayor a 0.70
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE1 0.643 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE2 0.840 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE3 0.720 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE4 0.764 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## EE1 0.000 1.040 0.000 0.000 0.000 0.000 0.000 0.000
## EE2 0.000 0.863 0.000 0.000 0.000 0.000 0.000 0.000
## EE3 0.000 0.793 0.000 0.000 0.000 0.000 0.000 0.000
## SI1 0.000 0.000 0.734 0.000 0.000 0.000 0.000 0.000
## SI2 0.000 0.000 0.702 0.000 0.000 0.000 0.000 0.000
## SI3 0.000 0.000 0.779 0.000 0.000 0.000 0.000 0.000
## SI4 0.000 0.000 0.879 0.000 0.000 0.000 0.000 0.000
## FC1 0.000 0.000 0.000 0.558 0.000 0.000 0.000 0.000
## FC2 0.000 0.000 0.000 0.699 0.000 0.000 0.000 0.000
## FC3 0.000 0.000 0.000 0.752 0.000 0.000 0.000 0.000
## HM1 0.000 0.000 0.000 0.000 0.898 0.000 0.000 0.000
## HM2 0.000 0.000 0.000 0.000 0.846 0.000 0.000 0.000
## HM3 0.000 0.000 0.000 0.000 0.890 0.000 0.000 0.000
## HA1 0.000 0.000 0.000 0.000 0.000 0.883 0.000 0.000
## HA2 0.000 0.000 0.000 0.000 0.000 0.884 0.000 0.000
## HA3 0.000 0.000 0.000 0.000 0.000 0.834 0.000 0.000
## HA4 0.000 0.000 0.000 0.000 0.000 0.880 0.000 0.000
## HA5 0.000 0.000 0.000 0.000 0.000 0.892 0.000 0.000
## IU1 0.000 0.000 0.000 0.000 0.000 0.000 0.830 0.000
## IU2 0.000 0.000 0.000 0.000 0.000 0.000 0.793 0.000
## U1  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.675
## U2  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.651
## U3  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.762
## U4  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.613
```

Hide

```
summary_estimacion_model$loadings^2
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE1 0.413 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE2 0.706 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE3 0.518 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE4 0.584 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## EE1 0.000 1.081 0.000 0.000 0.000 0.000 0.000 0.000
## EE2 0.000 0.745 0.000 0.000 0.000 0.000 0.000 0.000
## EE3 0.000 0.628 0.000 0.000 0.000 0.000 0.000 0.000
## SI1 0.000 0.000 0.538 0.000 0.000 0.000 0.000 0.000
## SI2 0.000 0.000 0.492 0.000 0.000 0.000 0.000 0.000
## SI3 0.000 0.000 0.606 0.000 0.000 0.000 0.000 0.000
## SI4 0.000 0.000 0.772 0.000 0.000 0.000 0.000 0.000
## FC1 0.000 0.000 0.000 0.311 0.000 0.000 0.000 0.000
## FC2 0.000 0.000 0.000 0.489 0.000 0.000 0.000 0.000
## FC3 0.000 0.000 0.000 0.566 0.000 0.000 0.000 0.000
## HM1 0.000 0.000 0.000 0.000 0.806 0.000 0.000 0.000
## HM2 0.000 0.000 0.000 0.000 0.715 0.000 0.000 0.000
## HM3 0.000 0.000 0.000 0.000 0.791 0.000 0.000 0.000
## HA1 0.000 0.000 0.000 0.000 0.000 0.779 0.000 0.000
## HA2 0.000 0.000 0.000 0.000 0.000 0.781 0.000 0.000
## HA3 0.000 0.000 0.000 0.000 0.000 0.696 0.000 0.000
## HA4 0.000 0.000 0.000 0.000 0.000 0.774 0.000 0.000
## HA5 0.000 0.000 0.000 0.000 0.000 0.795 0.000 0.000
## IU1 0.000 0.000 0.000 0.000 0.000 0.000 0.689 0.000
## IU2 0.000 0.000 0.000 0.000 0.000 0.000 0.629 0.000
## U1  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.456
## U2  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.424
## U3  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.580
## U4  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.376
```

Hide

```
summary_estimacion_model$weights  # Pesos -> Formativos
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE1 0.265 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE2 0.347 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE3 0.297 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## PE4 0.315 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## EE1 0.000 0.412 0.000 0.000 0.000 0.000 0.000 0.000
## EE2 0.000 0.342 0.000 0.000 0.000 0.000 0.000 0.000
## EE3 0.000 0.314 0.000 0.000 0.000 0.000 0.000 0.000
## SI1 0.000 0.000 0.283 0.000 0.000 0.000 0.000 0.000
## SI2 0.000 0.000 0.271 0.000 0.000 0.000 0.000 0.000
## SI3 0.000 0.000 0.300 0.000 0.000 0.000 0.000 0.000
## SI4 0.000 0.000 0.339 0.000 0.000 0.000 0.000 0.000
## FC1 0.000 0.000 0.000 0.348 0.000 0.000 0.000 0.000
## FC2 0.000 0.000 0.000 0.436 0.000 0.000 0.000 0.000
## FC3 0.000 0.000 0.000 0.469 0.000 0.000 0.000 0.000
## HM1 0.000 0.000 0.000 0.000 0.370 0.000 0.000 0.000
## HM2 0.000 0.000 0.000 0.000 0.349 0.000 0.000 0.000
## HM3 0.000 0.000 0.000 0.000 0.367 0.000 0.000 0.000
## HA1 0.000 0.000 0.000 0.000 0.000 0.224 0.000 0.000
## HA2 0.000 0.000 0.000 0.000 0.000 0.224 0.000 0.000
## HA3 0.000 0.000 0.000 0.000 0.000 0.212 0.000 0.000
## HA4 0.000 0.000 0.000 0.000 0.000 0.223 0.000 0.000
## HA5 0.000 0.000 0.000 0.000 0.000 0.226 0.000 0.000
## IU1 0.000 0.000 0.000 0.000 0.000 0.000 0.562 0.000
## IU2 0.000 0.000 0.000 0.000 0.000 0.000 0.537 0.000
## U1  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.324
## U2  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.313
## U3  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.366
## U4  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.294
```

Exportar a Excel

Hide

```
write.xlsx2(x=summary_estimacion_model$loadings,
            'resumen.xlsx',
            sheetName = "loadings",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)

write.xlsx2(x=summary_estimacion_model$weights,
            'resumen.xlsx',
            sheetName = "weights",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.6. Cargas Cruzadas

Hide

```
summary_estimacion_model$validity$cross_loadings
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE1 0.763 0.436 0.411 0.412 0.504 0.488 0.456 0.528
## PE2 0.870 0.450 0.457 0.471 0.556 0.499 0.596 0.559
## PE3 0.832 0.333 0.492 0.416 0.536 0.382 0.510 0.396
## PE4 0.789 0.388 0.468 0.465 0.493 0.429 0.542 0.427
## EE1 0.495 0.931 0.314 0.650 0.478 0.598 0.441 0.545
## EE2 0.470 0.953 0.247 0.622 0.441 0.560 0.366 0.475
## EE3 0.406 0.923 0.238 0.595 0.433 0.511 0.336 0.446
## SI1 0.442 0.235 0.860 0.343 0.360 0.328 0.450 0.371
## SI2 0.459 0.201 0.884 0.331 0.357 0.329 0.430 0.375
## SI3 0.480 0.180 0.857 0.360 0.401 0.400 0.477 0.381
## SI4 0.493 0.335 0.767 0.382 0.385 0.422 0.539 0.422
## FC1 0.398 0.478 0.272 0.778 0.467 0.481 0.361 0.341
## FC2 0.388 0.741 0.278 0.806 0.435 0.547 0.393 0.485
## FC3 0.501 0.381 0.445 0.807 0.465 0.431 0.491 0.455
## HM1 0.624 0.455 0.404 0.543 0.919 0.576 0.579 0.549
## HM2 0.565 0.468 0.435 0.521 0.912 0.595 0.545 0.504
## HM3 0.579 0.415 0.404 0.508 0.930 0.595 0.574 0.545
## HA1 0.543 0.573 0.353 0.552 0.628 0.896 0.573 0.684
## HA2 0.528 0.546 0.402 0.524 0.570 0.935 0.557 0.699
## HA3 0.441 0.600 0.376 0.645 0.599 0.855 0.567 0.626
## HA4 0.485 0.492 0.450 0.505 0.560 0.904 0.571 0.682
## HA5 0.483 0.491 0.423 0.519 0.526 0.913 0.568 0.700
## IU1 0.603 0.377 0.511 0.524 0.566 0.580 0.915 0.573
## IU2 0.580 0.374 0.528 0.433 0.555 0.566 0.906 0.534
## U1  0.422 0.411 0.342 0.333 0.431 0.584 0.483 0.755
## U2  0.447 0.373 0.346 0.423 0.487 0.528 0.522 0.766
## U3  0.529 0.422 0.398 0.475 0.499 0.666 0.482 0.831
## U4  0.393 0.418 0.339 0.441 0.357 0.529 0.381 0.722
```

Hide

```
write.xlsx2(x=summary_estimacion_model$validity$cross_loadings,
            'resumen.xlsx',
            sheetName = "cross_loadings",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.7. VIF

Hide

```
summary_estimacion_model$vif_antecedents
```

```
## IU :
##    PE    EE    SI    FC    HM    HA
## 2.184 2.087 1.566 2.283 2.200 2.237
##
## SNS :
##    FC    HA    IU
## 1.678 2.008 1.752
```

Hide

```
summary_estimacion_model$validity$vif_items
```

```
## PE :
##   PE1   PE2   PE3   PE4
## 1.736 2.260 1.995 1.675
##
## EE :
##   EE1   EE2   EE3
## 3.128 5.439 4.161
##
## SI :
##   SI1   SI2   SI3   SI4
## 2.858 3.256 2.271 1.455
##
## FC :
##   FC1   FC2   FC3
## 1.477 1.426 1.336
##
## HM :
##   HM1   HM2   HM3
## 2.950 2.912 3.340
##
## HA :
##   HA1   HA2   HA3   HA4   HA5
## 3.490 5.116 2.560 3.779 3.967
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
```

## E.4.8. Fornell-Larcker

Hide

```
summary_estimacion_model$validity$fl_criteria
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE  0.745     .     .     .     .     .     .     .
## EE  0.493 0.905     .     .     .     .     .     .
## SI  0.561 0.289 0.776     .     .     .     .     .
## FC  0.542 0.668 0.424 0.675     .     .     .     .
## HM  0.641 0.484 0.450 0.570 0.878     .     .     .
## HA  0.551 0.599 0.445 0.608 0.639 0.875     .     .
## IU  0.650 0.413 0.570 0.527 0.616 0.630 0.812     .
## SNS 0.586 0.527 0.464 0.544 0.579 0.753 0.608 0.678
##
## FL Criteria table reports square root of AVE on the diagonal and construct correlations on the lower tria
ngle.
```

Hide

```
write.xlsx2(x=summary_estimacion_model$validity$fl_criteria,
            'resumen.xlsx',
            sheetName = "Fornell-Larcker",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.9. fSquare

Hide

```
summary_estimacion_model$fSquare
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE  0.000 0.000 0.000 0.000 0.000 0.000 0.196 0.000
## EE  0.000 0.000 0.000 0.000 0.000 0.000 0.049 0.000
## SI  0.000 0.000 0.000 0.000 0.000 0.000 0.090 0.000
## FC  0.000 0.000 0.000 0.000 0.000 0.000 0.024 0.013
## HM  0.000 0.000 0.000 0.000 0.000 0.000 0.018 0.000
## HA  0.000 0.000 0.000 0.000 0.000 0.000 0.169 0.782
## IU  0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.149
## SNS 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
```

Hide

```
write.xlsx2(x=summary_estimacion_model$fSquare,
            'resumen.xlsx',
            sheetName = "fSquare",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.10. HTMT

Hide

```
summary_estimacion_model$validity$htmt
```

```
##        PE    EE    SI    FC    HM    HA    IU SNS
## PE      .     .     .     .     .     .     .   .
## EE  0.556     .     .     .     .     .     .   .
## SI  0.657 0.310     .     .     .     .     .   .
## FC  0.695 0.815 0.523     .     .     .     .   .
## HM  0.737 0.524 0.504 0.707     .     .     .   .
## HA  0.624 0.636 0.486 0.744 0.692     .     .   .
## IU  0.795 0.474 0.680 0.688 0.724 0.728     .   .
## SNS 0.730 0.618 0.563 0.721 0.689 0.881 0.777   .
```

Hide

```
write.xlsx2(x=summary_estimacion_model$validity$htmt ,
            'resumen.xlsx',
            sheetName = "htmt",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.11. Tabla de correlaciones

Hide

```
summary_estimacion_model$descriptives$correlations$constructs
```

```
##        PE    EE    SI    FC    HM    HA    IU   SNS
## PE  1.000 0.493 0.561 0.542 0.641 0.551 0.650 0.586
## EE  0.493 1.000 0.289 0.668 0.484 0.599 0.413 0.527
## SI  0.561 0.289 1.000 0.424 0.450 0.445 0.570 0.464
## FC  0.542 0.668 0.424 1.000 0.570 0.608 0.527 0.544
## HM  0.641 0.484 0.450 0.570 1.000 0.639 0.616 0.579
## HA  0.551 0.599 0.445 0.608 0.639 1.000 0.630 0.753
## IU  0.650 0.413 0.570 0.527 0.616 0.630 1.000 0.608
## SNS 0.586 0.527 0.464 0.544 0.579 0.753 0.608 1.000
```

Hide

```
write.xlsx2(x=summary_estimacion_model$descriptives$correlations$constructs  ,
            'resumen.xlsx',
            sheetName = "Correl_constructos",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.4.12. Otros

b) Efectos totales

c) Efectos indirectos

d) Puntuaciones estimadas para los constructos

e) Selección de modelo BIC, AIC

Hide

```
summary_estimacion_model$total_effects             ## b)
```

```
##        PE    EE    SI    FC    HM    HA     IU    SNS
## PE  0.000 0.000 0.000 0.000 0.000 0.000  0.375  0.096
## EE  0.000 0.000 0.000 0.000 0.000 0.000 -0.190 -0.049
## SI  0.000 0.000 0.000 0.000 0.000 0.000  0.198  0.051
## FC  0.000 0.000 0.000 0.000 0.000 0.000  0.171  0.123
## HM  0.000 0.000 0.000 0.000 0.000 0.000  0.108  0.028
## HA  0.000 0.000 0.000 0.000 0.000 0.000  0.317  0.717
## IU  0.000 0.000 0.000 0.000 0.000 0.000  0.000  0.257
## SNS 0.000 0.000 0.000 0.000 0.000 0.000  0.000  0.000
```

Hide

```
summary_estimacion_model$total_indirect_effects     ## c)
```

```
##        PE    EE    SI    FC    HM    HA    IU    SNS
## PE  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.096
## EE  0.000 0.000 0.000 0.000 0.000 0.000 0.000 -0.049
## SI  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.051
## FC  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.044
## HM  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.028
## HA  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.082
## IU  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.000
## SNS 0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.000
```

Hide

```
# summary_estimacion_model$composite_scores           ## d)
summary_estimacion_model$it_criteria                ## e)
```

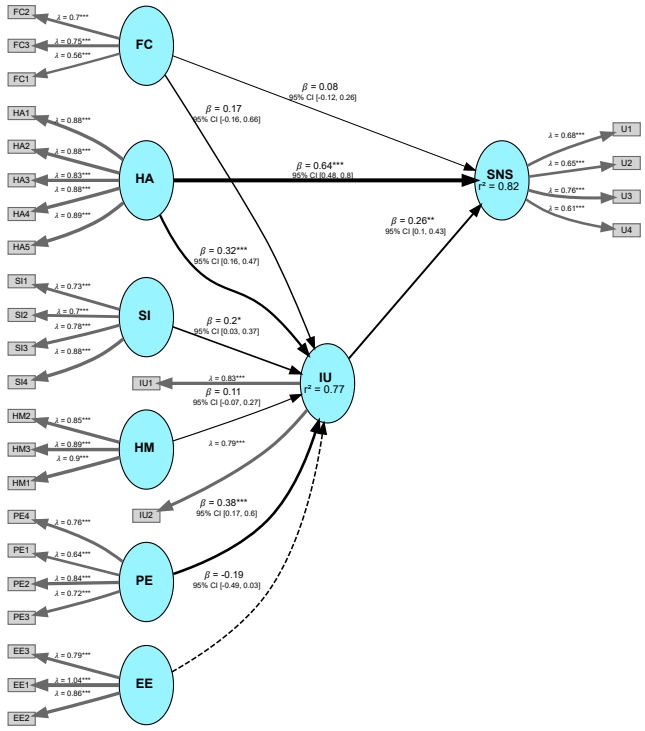
```
##           IU      SNS
## AIC -320.517 -345.750
## BIC -292.881 -329.958
```

# E.5. Estimación Bootstrap

Hide

```
boot_estimacion <- bootstrap_model(seminr_model = estimacion_model , #modelo estimado E.3  estimate_pls()
                nboot = 500,  ### N° Subsamples  >5000
                cores = parallel::detectCores(),        #CPU cores -parallel processing
                seed = 123)     #Semilla inicial


sum_boot <- summary(boot_estimacion,
                    alpha=0.05   ### Intervalo de confianza, en este caso es dos colas 90%
                                 )

plot(boot_estimacion, title = "Fig. 4 Bootstrapped Model")
```

Fig. 4 Bootstrapped Model




Hide

```
save_plot("fig4.Bootstrapped_Modelo.pdf")
```

# E.6. Reportes Bootstrapped

## E.6.1. Paths

Hide

```
sum_boot$bootstrapped_paths
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE  ->  IU          0.375          0.383        0.108   3.479   0.172    0.598
## EE  ->  IU         -0.190         -0.198        0.150  -1.269  -0.493    0.030
## SI  ->  IU          0.198          0.192        0.095   2.079   0.026    0.369
## FC  ->  IU          0.171          0.182        0.250   0.684  -0.156    0.659
## FC  ->  SNS         0.079          0.078        0.093   0.850  -0.115    0.258
## HM  ->  IU          0.108          0.099        0.095   1.144  -0.070    0.265
## HA  ->  IU          0.317          0.319        0.083   3.843   0.161    0.469
## HA  ->  SNS         0.636          0.634        0.085   7.469   0.481    0.803
## IU  ->  SNS         0.257          0.259        0.083   3.109   0.100    0.427
```

Hide

```
write.xlsx2(x=sum_boot$bootstrapped_paths   ,
            'resumen.xlsx',
            sheetName = "bootstrapped_Coef_Path",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

# E.6.2. Cargas, pesos y efectos totales del modelo

Hide

```
sum_boot$bootstrapped_loadings
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE1  ->  PE         0.643          0.642        0.055  11.588   0.526    0.739
## PE2  ->  PE         0.840          0.837        0.039  21.287   0.755    0.908
## PE3  ->  PE         0.720          0.716        0.055  13.153   0.609    0.808
## PE4  ->  PE         0.764          0.770        0.060  12.806   0.657    0.878
## EE1  ->  EE         1.040          1.043        0.051  20.565   0.944    1.147
## EE2  ->  EE         0.863          0.859        0.040  21.536   0.772    0.935
## EE3  ->  EE         0.793          0.791        0.048  16.443   0.692    0.876
## SI1  ->  SI         0.734          0.726        0.063  11.639   0.594    0.833
## SI2  ->  SI         0.702          0.700        0.049  14.383   0.599    0.788
## SI3  ->  SI         0.779          0.778        0.045  17.429   0.692    0.857
## SI4  ->  SI         0.879          0.880        0.061  14.480   0.757    0.984
## FC1  ->  FC         0.558          0.554        0.062   8.964   0.419    0.664
## FC2  ->  FC         0.699          0.695        0.056  12.496   0.578    0.808
## FC3  ->  FC         0.752          0.749        0.042  17.700   0.668    0.827
## HM1  ->  HM         0.898          0.899        0.035  25.629   0.829    0.960
## HM2  ->  HM         0.846          0.842        0.037  22.981   0.766    0.904
## HM3  ->  HM         0.890          0.889        0.028  31.964   0.835    0.949
## HA1  ->  HA         0.883          0.882        0.020  45.192   0.841    0.917
## HA2  ->  HA         0.884          0.884        0.020  45.154   0.843    0.920
## HA3  ->  HA         0.834          0.833        0.026  31.971   0.778    0.883
## HA4  ->  HA         0.880          0.882        0.022  40.056   0.841    0.923
## HA5  ->  HA         0.892          0.893        0.022  39.640   0.847    0.933
## IU1  ->  IU         0.830          0.829        0.023  35.386   0.781    0.871
## IU2  ->  IU         0.793          0.792        0.026  30.782   0.742    0.841
## U1  ->  SNS         0.675          0.676        0.034  19.893   0.603    0.742
## U2  ->  SNS         0.651          0.651        0.037  17.641   0.580    0.719
## U3  ->  SNS         0.762          0.762        0.030  25.317   0.704    0.821
## U4  ->  SNS         0.613          0.614        0.035  17.398   0.541    0.683
```

Hide

```
sum_boot$bootstrapped_weights #bootstrap standard error, t-statistic, and confidence intervals for the indic
        ator weights
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE1  ->  PE         0.265          0.264        0.020  13.363   0.226    0.301
## PE2  ->  PE         0.347          0.345        0.018  18.745   0.310    0.379
## PE3  ->  PE         0.297          0.295        0.022  13.789   0.251    0.336
## PE4  ->  PE         0.315          0.317        0.024  13.178   0.272    0.362
## EE1  ->  EE         0.412          0.414        0.023  17.642   0.374    0.467
## EE2  ->  EE         0.342          0.341        0.014  24.769   0.313    0.367
## EE3  ->  EE         0.314          0.314        0.017  18.921   0.281    0.344
## SI1  ->  SI         0.283          0.281        0.021  13.229   0.238    0.321
## SI2  ->  SI         0.271          0.270        0.014  19.790   0.242    0.295
## SI3  ->  SI         0.300          0.301        0.015  19.496   0.270    0.333
## SI4  ->  SI         0.339          0.341        0.029  11.742   0.289    0.398
## FC1  ->  FC         0.348          0.347        0.026  13.476   0.293    0.393
## FC2  ->  FC         0.436          0.436        0.030  14.407   0.381    0.502
## FC3  ->  FC         0.469          0.471        0.032  14.453   0.406    0.538
## HM1  ->  HM         0.370          0.372        0.015  24.502   0.343    0.402
## HM2  ->  HM         0.349          0.348        0.012  29.951   0.322    0.369
## HM3  ->  HM         0.367          0.367        0.012  30.609   0.346    0.391
## HA1  ->  HA         0.224          0.224        0.005  42.536   0.214    0.235
## HA2  ->  HA         0.224          0.224        0.005  49.239   0.215    0.233
## HA3  ->  HA         0.212          0.211        0.006  34.659   0.200    0.223
## HA4  ->  HA         0.223          0.224        0.005  42.076   0.213    0.234
## HA5  ->  HA         0.226          0.226        0.005  42.781   0.216    0.237
## IU1  ->  IU         0.562          0.562        0.013  44.223   0.537    0.588
## IU2  ->  IU         0.537          0.537        0.011  46.956   0.516    0.561
## U1  ->  SNS         0.324          0.324        0.014  23.155   0.297    0.351
## U2  ->  SNS         0.313          0.312        0.013  23.632   0.289    0.340
## U3  ->  SNS         0.366          0.365        0.016  23.054   0.337    0.395
## U4  ->  SNS         0.294          0.294        0.015  19.186   0.264    0.325
```

Hide

```
sum_boot$bootstrapped_total_paths  #bootstrap standard error, t-statistic, and confidence intervals total ef
        fects
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE  ->  IU          0.375          0.383        0.108   3.479   0.172    0.598
## PE  ->  SNS         0.096          0.099        0.042   2.296   0.027    0.188
## EE  ->  IU         -0.190         -0.198        0.150  -1.269  -0.493    0.030
## EE  ->  SNS        -0.049         -0.050        0.043  -1.127  -0.122    0.007
## SI  ->  IU          0.198          0.192        0.095   2.079   0.026    0.369
## SI  ->  SNS         0.051          0.050        0.030   1.684   0.004    0.110
## FC  ->  IU          0.171          0.182        0.250   0.684  -0.156    0.659
## FC  ->  SNS         0.123          0.124        0.115   1.072  -0.077    0.337
## HM  ->  IU          0.108          0.099        0.095   1.144  -0.070    0.265
## HM  ->  SNS         0.028          0.026        0.027   1.014  -0.018    0.080
## HA  ->  IU          0.317          0.319        0.083   3.843   0.161    0.469
## HA  ->  SNS         0.717          0.717        0.079   9.129   0.576    0.880
## IU  ->  SNS         0.257          0.259        0.083   3.109   0.100    0.427
```

Hide

```
write.xlsx2(x=sum_boot$bootstrapped_loadings    ,
            'resumen.xlsx',
            sheetName = "bootstrapped_loadings",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)

write.xlsx2(x=sum_boot$bootstrapped_weights   ,
            'resumen.xlsx',
            sheetName = "bootstrapped_weights",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)

write.xlsx2(x=sum_boot$bootstrapped_total_paths   ,
            'resumen.xlsx',
            sheetName = "bootstrapped_total_paths",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## E.6.3. HTMT CI

Hide

```
sum_boot$bootstrapped_HTMT
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE  ->  EE          0.556          0.556        0.045  12.385   0.464    0.641
## PE  ->  SI          0.657          0.660        0.048  13.782   0.568    0.753
## PE  ->  FC          0.695          0.698        0.052  13.367   0.594    0.790
## PE  ->  HM          0.737          0.737        0.035  21.122   0.670    0.800
## PE  ->  HA          0.624          0.625        0.040  15.430   0.540    0.698
## PE  ->  IU          0.795          0.796        0.038  20.780   0.723    0.864
## PE  ->  SNS         0.730          0.731        0.048  15.137   0.636    0.817
## EE  ->  SI          0.310          0.311        0.055   5.681   0.198    0.414
## EE  ->  FC          0.815          0.815        0.035  23.482   0.740    0.878
## EE  ->  HM          0.524          0.522        0.045  11.628   0.437    0.600
## EE  ->  HA          0.636          0.637        0.034  18.793   0.575    0.702
## EE  ->  IU          0.474          0.473        0.050   9.402   0.367    0.569
## EE  ->  SNS         0.618          0.617        0.040  15.340   0.534    0.687
## SI  ->  FC          0.523          0.531        0.064   8.182   0.401    0.645
## SI  ->  HM          0.504          0.506        0.049  10.219   0.412    0.601
## SI  ->  HA          0.486          0.484        0.047  10.344   0.385    0.576
## SI  ->  IU          0.680          0.680        0.048  14.067   0.581    0.774
## SI  ->  SNS         0.563          0.563        0.050  11.173   0.461    0.661
## FC  ->  HM          0.707          0.708        0.041  17.348   0.618    0.786
## FC  ->  HA          0.744          0.746        0.043  17.333   0.659    0.825
## FC  ->  IU          0.688          0.689        0.043  16.039   0.609    0.784
## FC  ->  SNS         0.721          0.722        0.041  17.679   0.645    0.794
## HM  ->  HA          0.692          0.691        0.034  20.184   0.620    0.753
## HM  ->  IU          0.724          0.724        0.036  20.304   0.653    0.791
## HM  ->  SNS         0.689          0.689        0.042  16.256   0.607    0.775
## HA  ->  IU          0.728          0.729        0.041  17.779   0.643    0.805
## HA  ->  SNS         0.881          0.880        0.023  37.923   0.833    0.921
## IU  ->  SNS         0.777          0.777        0.042  18.695   0.696    0.858
```

Hide

```
summary_estimacion_model$validity$htmt  ### HTMT modelo estructural ( <0.85 )
```

```
##        PE    EE    SI    FC    HM    HA    IU SNS
## PE      .     .     .     .     .     .     .   .
## EE  0.556     .     .     .     .     .     .   .
## SI  0.657 0.310     .     .     .     .     .   .
## FC  0.695 0.815 0.523     .     .     .     .   .
## HM  0.737 0.524 0.504 0.707     .     .     .   .
## HA  0.624 0.636 0.486 0.744 0.692     .     .   .
## IU  0.795 0.474 0.680 0.688 0.724 0.728     .   .
## SNS 0.730 0.618 0.563 0.721 0.689 0.881 0.777   .
```

Hide

```
write.xlsx2(x=sum_boot$bootstrapped_HTMT   ,
            'resumen.xlsx',
            sheetName = "bootstrapped_HTMT",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

# F. Análisis de Mediación

# F.1 Efectos Indirectos

*Efectos totales indirectos*

Hide

```
summary_estimacion_model$total_indirect_effects
```

```
##        PE    EE    SI    FC    HM    HA    IU    SNS
## PE  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.096
## EE  0.000 0.000 0.000 0.000 0.000 0.000 0.000 -0.049
## SI  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.051
## FC  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.044
## HM  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.028
## HA  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.082
## IU  0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.000
## SNS 0.000 0.000 0.000 0.000 0.000 0.000 0.000  0.000
```

Hide

```
#Evaluación de la significancia de los efectos indirectos. p1 * p2 es significante

specific_effect_significance(boot_estimacion,  ###Boot
                             from ='FC', #Desde
                             through = 'IU',  #Mediador ## podría ser un vector del tipo c('construct1', 'co
        nstruct2')).
                             to = 'SNS',
                             alpha = 0.05)
```

```
##  Original Est. Bootstrap Mean   Bootstrap SD        T Stat.        2.5% CI
##     0.04399887     0.04551061     0.07082786     0.62120854    -0.04255763
##       97.5% CI
##     0.17648413
```

Hide

```
specific_effect_significance(boot_estimacion,  ###Boot
                             from ='HA',
                             through = 'IU',
                             to = 'SNS',
                             alpha = 0.05)
```

```
##  Original Est. Bootstrap Mean   Bootstrap SD        T Stat.        2.5% CI
##     0.08155251     0.08312684     0.03662836     2.22648555     0.02400088
##       97.5% CI
##     0.16578543
```

Hide

```
#FC ==> SNS No significativo ==> Evaluar si p3 es Directo o no efecto
#HA ==> SNS  Significativo  ==> Efecto Complementario/ Competitivo o Indirecto solo
sum_boot$total_indirect_effects
```

```
## NULL
```

# F.2 Efecto directo

## F.2.1 Paso 1: Significancia

Evaluar la significancia y luego para ver si es mediación full o parcial se revisan los path directos.

Hide

```
summary_estimacion_model$paths
```

```
##            IU   SNS
## R^2     0.772 0.816
## AdjR^2  0.768 0.814
## PE      0.375     .
## EE     -0.190     .
## SI      0.198     .
## FC      0.171 0.079
## HM      0.108     .
## HA      0.317 0.636
## IU          . 0.257
```

Hide

```
sum_boot$bootstrapped_paths
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE  ->  IU          0.375          0.383        0.108   3.479   0.172    0.598
## EE  ->  IU         -0.190         -0.198        0.150  -1.269  -0.493    0.030
## SI  ->  IU          0.198          0.192        0.095   2.079   0.026    0.369
## FC  ->  IU          0.171          0.182        0.250   0.684  -0.156    0.659
## FC  ->  SNS         0.079          0.078        0.093   0.850  -0.115    0.258
## HM  ->  IU          0.108          0.099        0.095   1.144  -0.070    0.265
## HA  ->  IU          0.317          0.319        0.083   3.843   0.161    0.469
## HA  ->  SNS         0.636          0.634        0.085   7.469   0.481    0.803
## IU  ->  SNS         0.257          0.259        0.083   3.109   0.100    0.427
```

Hide

```
#FC ==> SNS No significativo ==> No efecto
#HA ==> SNS  Significativo  ==> Evaluar si es complementario (0<) o competitivo (0>)
```

## F.2.2 Paso 2: tipo de mediación

Hide

```
## Calcula el signo de ESE CAMINO p1*p2*p3 complementario (0<) o competitivo (0>)
summary_estimacion_model$paths['HA', 'SNS'] *
  summary_estimacion_model$paths['HA', 'IU'] *
  summary_estimacion_model$paths['IU', 'SNS']
```

```
## [1] 0.05182717
```

Hide

```
summary_estimacion_model$paths['FC', 'SNS'] *
  summary_estimacion_model$paths['FC', 'IU'] *
  summary_estimacion_model$paths['IU', 'SNS']
```

```
## [1] 0.003467245
```

# G. Predict PLS

## G.1. Generar la predicción del modelo

Hide

```
predict_modelo <- predict_pls(
  model = estimacion_model,   ### modelo de medida E.3
  technique = predict_DA,
      # direct antecedent (predict_DA) consideraría tanto el antecedente como el mediador predictor del cons
         tructo
      # earliest antecedent (predict_EA) mediador se excluiría del análisis
  noFolds = 10,                     ### Folds a generar
  reps = 10)                        ### Número de repeticiones CV


sum_predict_modelo <- summary(predict_modelo)
#sum_predict_modelo
```

Comparamos los RMSE de PLS out-of-sample metrics v/s LM out-of-sample metrics. Si PLS<LM Ok

Hide

```
#sum_predict_modelo$PLS_out_of_sample
#sum_predict_modelo$LM_out_of_sample

predict_dif <- sum_predict_modelo$PLS_out_of_sample-sum_predict_modelo$LM_out_of_sample
predict_dif
```

```
##         IU1    IU2    U1    U2    U3     U4
## RMSE  0.008 -0.029 0.019 0.012 0.020 -0.041
## MAE  -0.004 -0.020 0.027 0.026 0.028 -0.020
```

Hide

```
# Si todos los items son negativos ==> Alta predicción (PLS<LM)
# Si es la mayoría ==> Baja predicción
# Si ninguno ==> No poder de predicción
```

Hide

```
write.xlsx2(x=predict_dif,
            'resumen.xlsx',
            sheetName = "Predict_dif (PLS-LM)",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

Hide

```
sum_predict_modelo$prediction_error
```

| | IU1 <dbl> | IU2 <dbl> | U1 <dbl> | U2 <dbl> | U3 <dbl> |
|---|---|---|---|---|---|
| 1 | -3.292677e-01 | 6.981588e-02 | 0.605063768 | -0.4675924214 | -1.683447206 |
| 2 | 8.315474e-02 | 2.687975e-01 | 0.445801527 | 0.5451121091 | -0.127688184 |
| 3 | 6.911516e-02 | 2.437044e-01 | 0.281919452 | 0.4202103174 | -0.324320623 |
| 4 | -1.248332e-01 | -2.841356e-01 | 1.760336334 | 0.5619169077 | -0.280634887 |
| 5 | -6.254800e-01 | 7.478602e-02 | 0.151908163 | -0.9599985251 | -0.957142181 |
| 6 | 1.772859e-01 | -1.601363e+00 | 0.362440064 | 0.4749212635 | 0.762474541 |
| 7 | 2.821945e-01 | -1.501784e+00 | 0.346112099 | -0.5494927518 | 0.750333818 |
| 8 | 4.245908e-01 | 7.277176e-01 | 0.047016903 | -1.8498735040 | 0.330899840 |
| 9 | 2.587692e-01 | 4.975236e-01 | 0.189317998 | 0.2735825626 | 0.509694001 |
| 10 | 1.661894e-01 | 3.690041e-01 | 0.134965248 | 0.2264277838 | 0.441402508 |

1-10 of 383 rows | 1-6 of 7 columns Previous **1** 2 3 4 5 6 … 39 Next

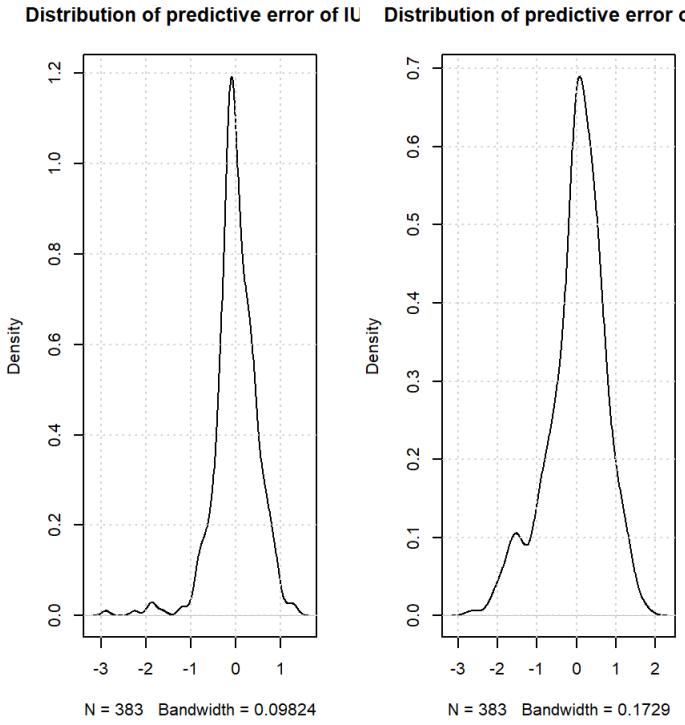
Hide

```
write.xlsx2(x=sum_predict_modelo$prediction_error,
            'resumen.xlsx',
            sheetName = "Predict_erro",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

# G.2. Analizar la distribución del error (indicador en específico)

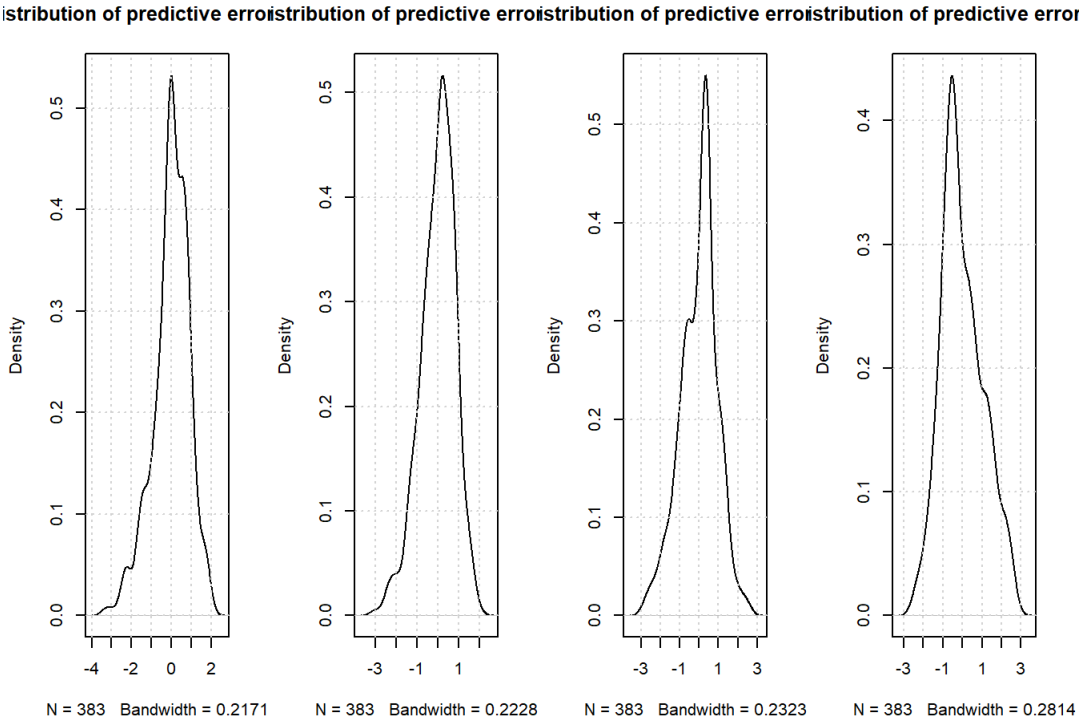
Hide

```
par(mfrow=c(1,3))
plot(sum_predict_modelo,
     indicator = 'IU1')
plot(sum_predict_modelo,
     indicator = 'IU2')
par(mfrow=c(1,1))
```




Hide

```
par(mfrow=c(1,4))
plot(sum_predict_modelo,
     indicator = 'U1')
plot(sum_predict_modelo,
     indicator = 'U2')
plot(sum_predict_modelo,
     indicator = 'U3')
plot(sum_predict_modelo,
     indicator = 'U4')
```



Hide

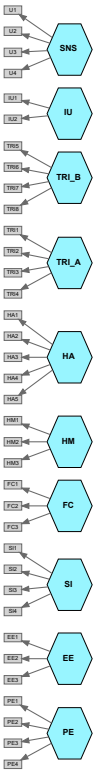
```
par(mfrow=c(1,1))
```

# H. Análisis de Moderadores

## H.1. Modelo de medida Moderadores

Hide

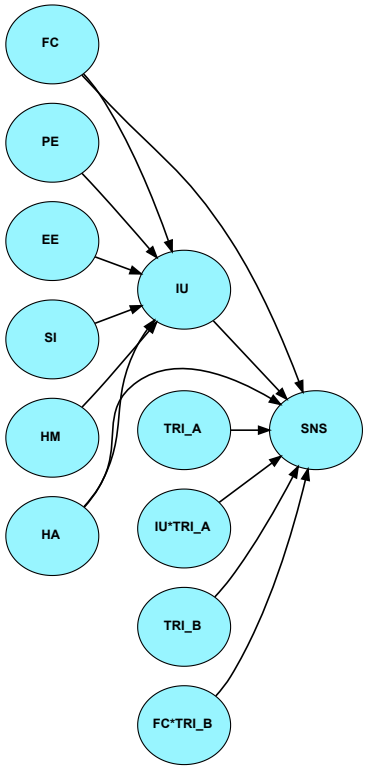
```
modelo_medida_mod <- constructs(
  composite('PE', multi_items('PE', 1:4), weights = mode_A),
  composite('EE', multi_items('EE', 1:3)),
  composite('SI', multi_items('SI', 1:4)),
  composite('FC', multi_items('FC', 1:3)),
  composite('HM', multi_items('HM', 1:3)),
  composite('HA', multi_items('HA', 1:5)),
# composite('CUSA', single_item('cusa')),  # En el caso de ser un unico item dejar como single_item
  composite('TRI_A', multi_items('TRI', 1:4)),
  composite('TRI_B', multi_items('TRI', 5:8)),
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4)),
  interaction_term(iv = 'IU', moderator = 'TRI_A', method = two_stage),  #Moderador method = orthogonal o me
         thod = two_stage
  interaction_term(iv = 'FC', moderator = 'TRI_B', method = two_stage)  #Moderador method = orthogonal o met
         hod = two_stage
)

plot(modelo_medida_mod)
```




## H.2. Modelo Estructural Moderadores

Hide

```
modelo_estruc_mod <- relationships(
  paths(from = c('PE', 'EE', 'SI', 'HM','FC', "HA"), to = c('IU')),
  paths(from = c('HA'), to = c('SNS')),
  paths(from = c('IU', 'TRI_A', 'IU*TRI_A'), to = c('SNS')),
  paths(from = c('FC', 'TRI_B', 'FC*TRI_B'), to = c('SNS'))
)

 plot(modelo_estruc_mod)
```

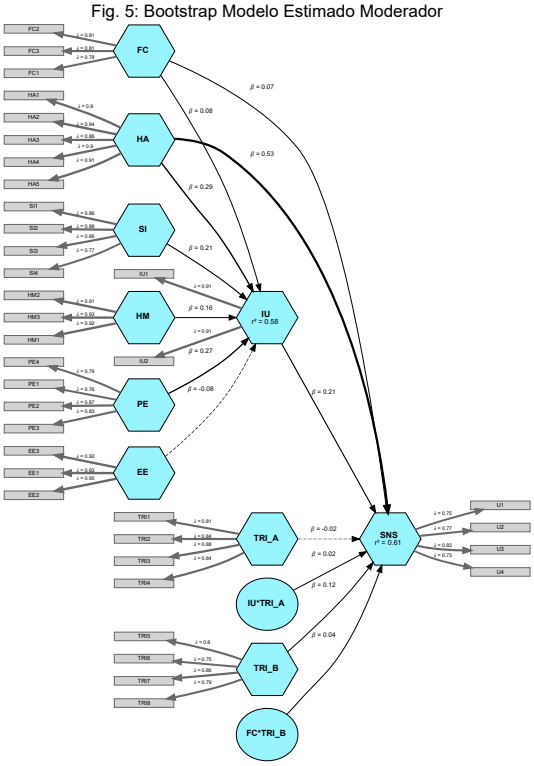



# H.3. Ejecución modelo

Hide

```
pls_model_mod_med <- estimate_pls(data = pls_data2,
                                  measurement_model = modelo_medida_mod,
                                  structural_model = modelo_estruc_mod,
                                  missing = mean_replacement,
                                  missing_value = '-99'
                                  )

boot_pls_model_mod_med <- bootstrap_model(seminr_model = pls_model_mod_med,
                                          nboot = 500)   #Cambiar al menos a 5000

sum_pls_model_mod_med <- summary(pls_model_mod_med)
sum_boot_pls_model_mod <- summary(boot_pls_model_mod_med, alpha = 0.05)

plot(pls_model_mod_med, title =  "Fig. 5: Bootstrap Modelo Estimado Moderador")
```

Fig. 5: Bootstrap Modelo Estimado Moderador




Hide

```
save_plot("fig 5.Bootstrap Modelo Estimado Moderador.pdf")
```

# H.4. Evaluar el modelo Moderador

## H.4.1. R^2 Exógenos

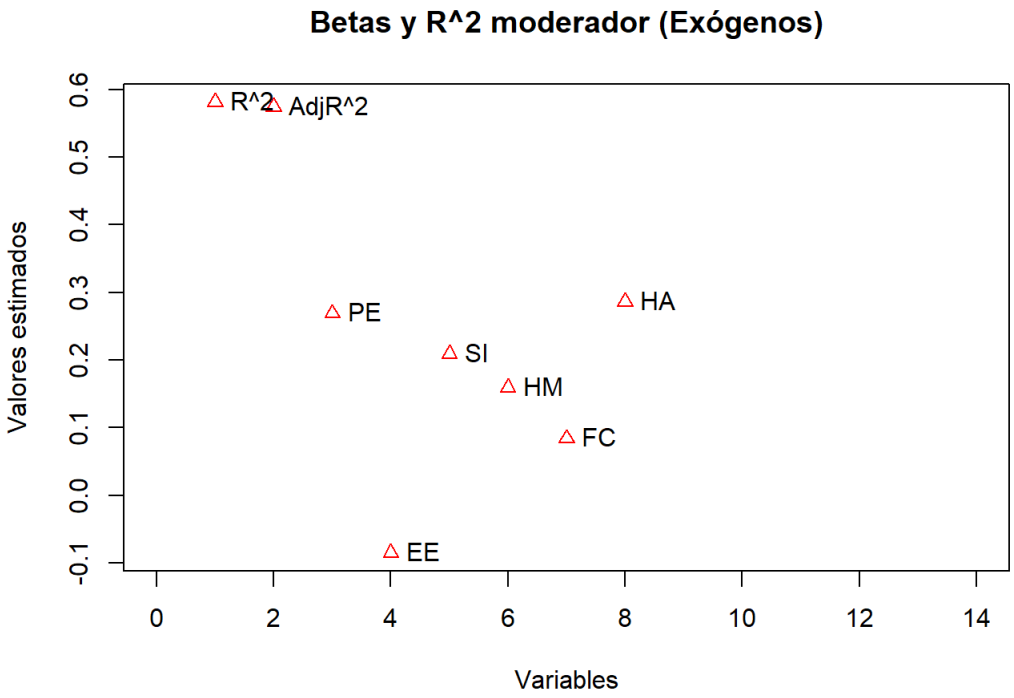
Hide

```
sum_pls_model_mod_med$paths
```

```
##                IU    SNS
## R^2         0.581  0.613
## AdjR^2      0.575  0.606
## PE          0.269      .
## EE         -0.085      .
## SI          0.209      .
## HM          0.159      .
## FC          0.084  0.066
## HA          0.286  0.531
## IU              .  0.205
## TRI_A           . -0.017
## IU*TRI_A        .  0.021
## TRI_B           .  0.119
## FC*TRI_B        .  0.040
```

Hide

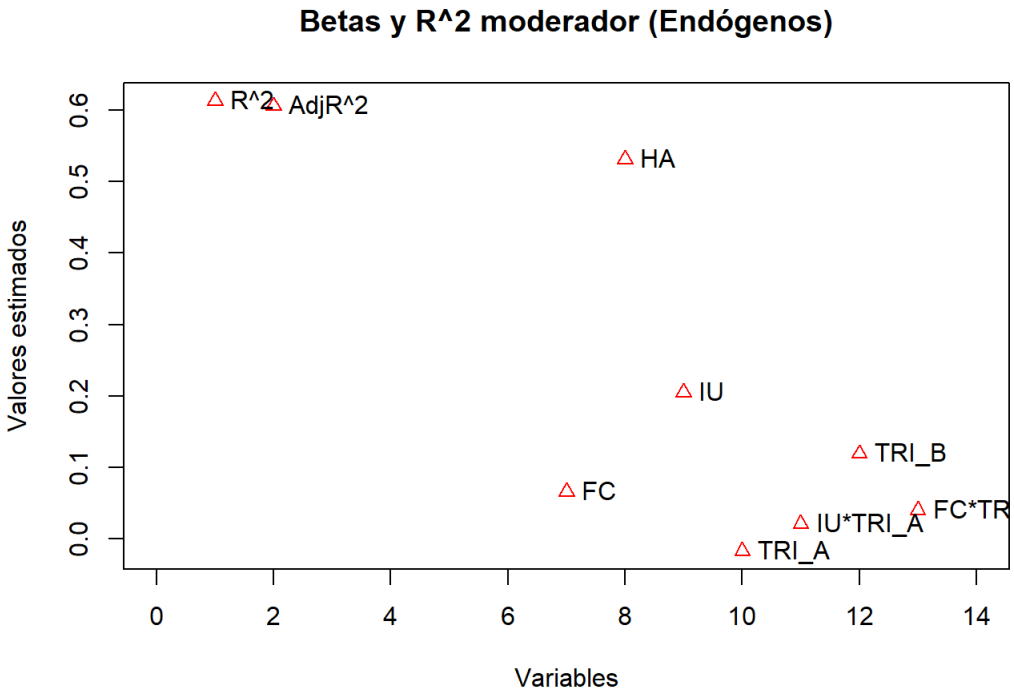
```
plot(sum_pls_model_mod_med$paths[,1], pch = 2, col = "red", main="Betas y R^2 moderador (Exógenos)",
     xlab = "Variables", ylab = "Valores estimados", xlim = c(0,length(row.names(sum_pls_model_mod_med$path
         s))+1)
     )
text(sum_pls_model_mod_med$paths[,1],labels = row.names(sum_pls_model_mod_med$paths) , pos = 4)
```




Endógenos

Hide

```
plot(sum_pls_model_mod_med$paths[,2], pch = 2, col = "red", main="Betas y R^2 moderador (Endógenos)",
     xlab = "Variables", ylab = "Valores estimados" , xlim = c(0,length(row.names(sum_pls_model_mod_med$path
         s))+1) )
text(sum_pls_model_mod_med$paths[,2],labels = row.names(sum_pls_model_mod_med$paths) , pos = 4)
```




Exportar Excel

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$paths,
            'resumen.xlsx',
            sheetName = "BetasyR_Moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.2. Fiabilidad Cronbach's alpha (alpha), composite reliability (rhoC), average variance extracted (AVE),

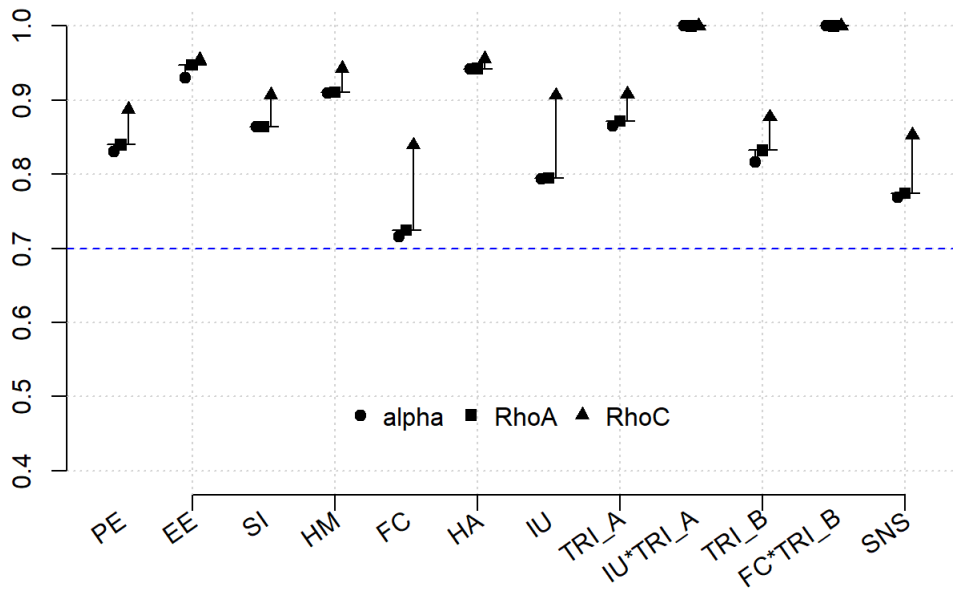
Hide

```
sum_pls_model_mod_med$reliability
```

```
##          alpha  rhoC   AVE  rhoA
## PE       0.831 0.887 0.664 0.840
## EE       0.930 0.955 0.876 0.947
## SI       0.864 0.907 0.711 0.864
## HM       0.910 0.943 0.847 0.911
## FC       0.716 0.839 0.635 0.724
## HA       0.942 0.956 0.812 0.943
## IU       0.794 0.906 0.829 0.795
## TRI_A    0.866 0.908 0.712 0.872
## IU*TRI_A 1.000 1.000 1.000 1.000
## TRI_B    0.816 0.877 0.642 0.832
## FC*TRI_B 1.000 1.000 1.000 1.000
## SNS      0.769 0.853 0.592 0.774
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5
```

Hide

```
plot(sum_pls_model_mod_med$reliability)
```




Exportar a Excel

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$reliability,
            'resumen.xlsx',
            sheetName = "reliability_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.3. Cargas

Hide

```
sum_pls_model_mod_med$loadings # Cargas -> reflectivas mayor a 0.70
```

```
##                    PE     EE    SI     HM     FC     HA     IU  TRI_A IU*TRI_A
## PE1             0.763  0.000 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## PE2             0.870  0.000 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## PE3             0.832  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## PE4             0.789  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## EE1             0.000  0.931 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## EE2             0.000  0.953 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## EE3             0.000  0.923 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## SI1             0.000  0.000 0.860  0.000  0.000  0.000  0.000  0.000    0.000
## SI2             0.000  0.000 0.884  0.000  0.000  0.000  0.000  0.000    0.000
## SI3             0.000  0.000 0.857  0.000  0.000  0.000  0.000  0.000    0.000
## SI4             0.000  0.000 0.767  0.000  0.000  0.000  0.000  0.000    0.000
## FC1             0.000  0.000 0.000  0.000  0.778  0.000  0.000  0.000    0.000
## FC2             0.000  0.000 0.000  0.000  0.806  0.000  0.000  0.000   -0.000
## FC3             0.000  0.000 0.000  0.000  0.807  0.000  0.000  0.000    0.000
## HM1             0.000  0.000 0.000  0.919  0.000  0.000  0.000  0.000    0.000
## HM2             0.000  0.000 0.000  0.912  0.000  0.000  0.000  0.000    0.000
## HM3             0.000  0.000 0.000  0.930  0.000  0.000  0.000  0.000    0.000
## HA1             0.000  0.000 0.000  0.000  0.000  0.896  0.000  0.000    0.000
## HA2             0.000  0.000 0.000  0.000  0.000  0.935  0.000  0.000    0.000
## HA3             0.000  0.000 0.000  0.000  0.000  0.855  0.000  0.000    0.000
## HA4             0.000  0.000 0.000  0.000  0.000  0.904  0.000  0.000    0.000
## HA5             0.000  0.000 0.000  0.000  0.000  0.913  0.000  0.000    0.000
## TRI1            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.810    0.000
## TRI2            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.844    0.000
## TRI3            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.884    0.000
## TRI4            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.836    0.000
## TRI5            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## TRI6            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## TRI7            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## TRI8            0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## IU1             0.000  0.000 0.000  0.000  0.000  0.000  0.915  0.000   -0.000
## IU2             0.000  0.000 0.000  0.000  0.000  0.000  0.906  0.000   -0.000
## U1              0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## U2              0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000   -0.000
## U3              0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## U4              0.000  0.000 0.000  0.000  0.000  0.000  0.000  0.000    0.000
## IU*TRI_A_intxn -0.000 -0.000 0.000  0.000  0.000  0.000 -0.000  0.000    0.888
## FC*TRI_B_intxn -0.000 -0.000 0.000 -0.000 -0.000 -0.000 -0.000 -0.000    0.000
##                 TRI_B FC*TRI_B    SNS
## PE1             0.000   -0.000  0.000
## PE2             0.000   -0.000  0.000
## PE3             0.000   -0.000  0.000
## PE4             0.000   -0.000  0.000
## EE1             0.000   -0.000  0.000
## EE2             0.000   -0.000  0.000
## EE3             0.000   -0.000  0.000
## SI1             0.000    0.000  0.000
## SI2             0.000    0.000  0.000
## SI3             0.000    0.000  0.000
## SI4             0.000    0.000  0.000
## FC1             0.000   -0.000  0.000
## FC2             0.000   -0.000  0.000
## FC3             0.000   -0.000  0.000
## HM1             0.000   -0.000  0.000
## HM2             0.000   -0.000  0.000
## HM3             0.000   -0.000  0.000
## HA1             0.000   -0.000  0.000
## HA2             0.000   -0.000  0.000
## HA3             0.000   -0.000  0.000
```

```
## HA4             0.000   -0.000  0.000
## HA5             0.000   -0.000  0.000
## TRI1            0.000    0.000  0.000
## TRI2            0.000   -0.000  0.000
## TRI3            0.000   -0.000  0.000
## TRI4            0.000   -0.000  0.000
## TRI5            0.800   -0.000  0.000
## TRI6            0.753   -0.000  0.000
## TRI7            0.856   -0.000  0.000
## TRI8            0.792   -0.000  0.000
## IU1             0.000   -0.000  0.000
## IU2             0.000   -0.000  0.000
## U1              0.000   -0.000  0.752
## U2              0.000   -0.000  0.765
## U3              0.000   -0.000  0.828
## U4              0.000   -0.000  0.729
## IU*TRI_A_intxn  0.000    0.000  0.000
## FC*TRI_B_intxn -0.000    1.132 -0.000
```

Hide

```
sum_pls_model_mod_med$weights  # Pesos -> Formativos
```

```
##                    PE    EE    SI    HM    FC    HA    IU TRI_A IU*TRI_A TRI_B
## PE1             0.265 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## PE2             0.347 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## PE3             0.297 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## PE4             0.315 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## EE1             0.000 0.412 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## EE2             0.000 0.342 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## EE3             0.000 0.314 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SI1             0.000 0.000 0.283 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SI2             0.000 0.000 0.271 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SI3             0.000 0.000 0.300 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SI4             0.000 0.000 0.339 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## FC1             0.000 0.000 0.000 0.000 0.348 0.000 0.000 0.000    0.000 0.000
## FC2             0.000 0.000 0.000 0.000 0.436 0.000 0.000 0.000    0.000 0.000
## FC3             0.000 0.000 0.000 0.000 0.468 0.000 0.000 0.000    0.000 0.000
## HM1             0.000 0.000 0.000 0.370 0.000 0.000 0.000 0.000    0.000 0.000
## HM2             0.000 0.000 0.000 0.349 0.000 0.000 0.000 0.000    0.000 0.000
## HM3             0.000 0.000 0.000 0.367 0.000 0.000 0.000 0.000    0.000 0.000
## HA1             0.000 0.000 0.000 0.000 0.000 0.224 0.000 0.000    0.000 0.000
## HA2             0.000 0.000 0.000 0.000 0.000 0.224 0.000 0.000    0.000 0.000
## HA3             0.000 0.000 0.000 0.000 0.000 0.212 0.000 0.000    0.000 0.000
## HA4             0.000 0.000 0.000 0.000 0.000 0.223 0.000 0.000    0.000 0.000
## HA5             0.000 0.000 0.000 0.000 0.000 0.226 0.000 0.000    0.000 0.000
## TRI1            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.248    0.000 0.000
## TRI2            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.312    0.000 0.000
## TRI3            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.311    0.000 0.000
## TRI4            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.311    0.000 0.000
## TRI5            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.328
## TRI6            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.213
## TRI7            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.362
## TRI8            0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.338
## IU1             0.000 0.000 0.000 0.000 0.000 0.000 0.562 0.000    0.000 0.000
## IU2             0.000 0.000 0.000 0.000 0.000 0.000 0.537 0.000    0.000 0.000
## U1              0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## U2              0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## U3              0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## U4              0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## IU*TRI_A_intxn 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    1.000 0.000
## FC*TRI_B_intxn 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
##                FC*TRI_B   SNS
## PE1               0.000 0.000
## PE2               0.000 0.000
## PE3               0.000 0.000
## PE4               0.000 0.000
## EE1               0.000 0.000
## EE2               0.000 0.000
## EE3               0.000 0.000
## SI1               0.000 0.000
## SI2               0.000 0.000
## SI3               0.000 0.000
## SI4               0.000 0.000
## FC1               0.000 0.000
## FC2               0.000 0.000
## FC3               0.000 0.000
## HM1               0.000 0.000
## HM2               0.000 0.000
## HM3               0.000 0.000
## HA1               0.000 0.000
## HA2               0.000 0.000
## HA3               0.000 0.000
```

```
## HA4               0.000 0.000
## HA5               0.000 0.000
## TRI1              0.000 0.000
## TRI2              0.000 0.000
## TRI3              0.000 0.000
## TRI4              0.000 0.000
## TRI5              0.000 0.000
## TRI6              0.000 0.000
## TRI7              0.000 0.000
## TRI8              0.000 0.000
## IU1               0.000 0.000
## IU2               0.000 0.000
## U1                0.000 0.322
## U2                0.000 0.313
## U3                0.000 0.358
## U4                0.000 0.305
## IU*TRI_A_intxn    0.000 0.000
## FC*TRI_B_intxn    1.000 0.000
```

Exportar a Excel

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$loadings,
            'resumen.xlsx',
            sheetName = "loadings_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)

write.xlsx2(x=sum_pls_model_mod_med$weights,
            'resumen.xlsx',
            sheetName = "weights_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.4. Cargas Cruzadas

Hide

```
sum_pls_model_mod_med$validity$cross_loadings
```

```
##                     PE     EE    SI     HM     FC     HA     IU  TRI_A IU*TRI_A
## PE1              0.763  0.436 0.411  0.504  0.412  0.488  0.456  0.509   -0.044
## PE2              0.870  0.450 0.457  0.556  0.471  0.499  0.596  0.532   -0.097
## PE3              0.832  0.333 0.492  0.536  0.416  0.382  0.510  0.440    0.001
## PE4              0.789  0.388 0.468  0.493  0.465  0.429  0.542  0.475    0.088
## EE1              0.495  0.931 0.314  0.478  0.650  0.598  0.441  0.421   -0.026
## EE2              0.470  0.953 0.247  0.441  0.622  0.560  0.366  0.402   -0.057
## EE3              0.406  0.923 0.238  0.433  0.595  0.511  0.336  0.399   -0.039
## SI1              0.442  0.235 0.860  0.360  0.343  0.328  0.450  0.346    0.139
## SI2              0.459  0.201 0.884  0.357  0.331  0.329  0.430  0.346    0.203
## SI3              0.480  0.180 0.857  0.401  0.359  0.400  0.477  0.381    0.166
## SI4              0.493  0.335 0.767  0.385  0.382  0.422  0.539  0.448    0.065
## FC1              0.398  0.478 0.272  0.467  0.778  0.481  0.361  0.367    0.105
## FC2              0.388  0.741 0.278  0.435  0.806  0.547  0.393  0.400   -0.021
## FC3              0.501  0.381 0.445  0.465  0.807  0.431  0.491  0.411    0.058
## HM1              0.624  0.455 0.404  0.919  0.543  0.576  0.579  0.556    0.027
## HM2              0.565  0.468 0.435  0.912  0.521  0.595  0.545  0.551    0.049
## HM3              0.579  0.415 0.404  0.930  0.508  0.595  0.574  0.539    0.028
## HA1              0.543  0.573 0.353  0.628  0.552  0.896  0.573  0.587    0.046
## HA2              0.528  0.546 0.402  0.570  0.524  0.935  0.557  0.589    0.101
## HA3              0.441  0.600 0.376  0.599  0.645  0.855  0.567  0.487    0.032
## HA4              0.485  0.492 0.450  0.560  0.505  0.904  0.571  0.611    0.135
## HA5              0.483  0.491 0.423  0.526  0.519  0.913  0.568  0.576    0.124
## TRI1             0.430  0.305 0.379  0.446  0.353  0.465  0.432  0.810    0.159
## TRI2             0.483  0.359 0.382  0.479  0.461  0.563  0.477  0.844    0.129
## TRI3             0.518  0.351 0.383  0.538  0.418  0.544  0.515  0.884    0.125
## TRI4             0.582  0.449 0.403  0.541  0.426  0.556  0.553  0.836    0.056
## TRI5             0.346  0.470 0.210  0.361  0.431  0.466  0.327  0.307   -0.009
## TRI6             0.287  0.409 0.159  0.241  0.334  0.288  0.233  0.229   -0.086
## TRI7             0.403  0.589 0.321  0.432  0.566  0.529  0.385  0.418    0.035
## TRI8             0.295  0.480 0.167  0.338  0.491  0.480  0.297  0.381    0.125
## IU1              0.603  0.377 0.511  0.566  0.524  0.580  0.915  0.506   -0.090
## IU2              0.580  0.374 0.528  0.555  0.433  0.566  0.906  0.568   -0.052
## U1               0.422  0.411 0.342  0.431  0.333  0.584  0.483  0.405    0.007
## U2               0.447  0.373 0.346  0.487  0.423  0.528  0.522  0.399   -0.035
## U3               0.529  0.422 0.398  0.499  0.475  0.666  0.482  0.496    0.120
## U4               0.393  0.418 0.339  0.357  0.441  0.529  0.381  0.307    0.123
## IU*TRI_A_intxn -0.017 -0.043 0.166  0.037  0.055  0.098 -0.078  0.136    1.000
## FC*TRI_B_intxn -0.084 -0.223 0.165 -0.130 -0.348 -0.158 -0.069 -0.021    0.259
##                  TRI_B FC*TRI_B    SNS
## PE1              0.423   -0.093  0.527
## PE2              0.365   -0.109  0.558
## PE3              0.268   -0.062  0.395
## PE4              0.324   -0.009  0.428
## EE1              0.602   -0.209  0.545
## EE2              0.582   -0.221  0.475
## EE3              0.538   -0.195  0.446
## SI1              0.229    0.149  0.371
## SI2              0.225    0.189  0.375
## SI3              0.218    0.144  0.381
## SI4              0.247    0.083  0.422
## FC1              0.445   -0.306  0.342
## FC2              0.615   -0.292  0.486
## FC3              0.342   -0.243  0.455
## HM1              0.406   -0.123  0.548
## HM2              0.428   -0.146  0.503
## HM3              0.383   -0.091  0.545
## HA1              0.512   -0.169  0.683
## HA2              0.508   -0.128  0.699
## HA3              0.579   -0.244  0.626
```

```
## HA4             0.495   -0.045  0.682
## HA5             0.467   -0.130  0.700
## TRI1            0.281    0.052  0.370
## TRI2            0.339   -0.026  0.466
## TRI3            0.368   -0.041  0.465
## TRI4            0.448   -0.041  0.464
## TRI5            0.800   -0.019  0.437
## TRI6            0.753   -0.061  0.283
## TRI7            0.856   -0.099  0.482
## TRI8            0.792   -0.135  0.450
## IU1             0.354   -0.089  0.572
## IU2             0.369   -0.036  0.533
## U1              0.362   -0.038  0.752
## U2              0.418   -0.113  0.765
## U3              0.370   -0.067  0.828
## U4              0.493   -0.035  0.729
## IU*TRI_A_intxn  0.034    0.259  0.072
## FC*TRI_B_intxn -0.101    1.000 -0.082
```

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$validity$cross_loadings,
            'resumen.xlsx',
            sheetName = "cross_loadings_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.5. VIF

Hide

```
sum_pls_model_mod_med$vif_antecedents
```

```
## IU :
##    PE    EE    SI    HM    FC    HA
## 2.184 2.087 1.566 2.200 2.284 2.237
##
## SNS :
##       HA       IU    TRI_A IU*TRI_A       FC    TRI_B FC*TRI_B
##    2.529    2.067    1.953    1.193    2.304    1.742    1.334
```

Hide

```
sum_pls_model_mod_med$validity$vif_items
```

```
## PE :
##   PE1   PE2   PE3   PE4
## 1.736 2.260 1.995 1.675
##
## EE :
##   EE1   EE2   EE3
## 3.128 5.439 4.161
##
## SI :
##   SI1   SI2   SI3   SI4
## 2.858 3.256 2.271 1.455
##
## HM :
##   HM1   HM2   HM3
## 2.950 2.912 3.340
##
## FC :
##   FC1   FC2   FC3
## 1.477 1.426 1.336
##
## HA :
##   HA1   HA2   HA3   HA4   HA5
## 3.490 5.116 2.560 3.779 3.967
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## IU*TRI_A :
## IU*TRI_A_intxn
##              1
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## FC*TRI_B :
## FC*TRI_B_intxn
##              1
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
```

## H.4.6. Fornell-Larcker

Hide

```
sum_pls_model_mod_med$validity$fl_criteria
```

```
##              PE     EE    SI     HM     FC     HA     IU  TRI_A IU*TRI_A  TRI_B
## PE        0.815      .     .      .      .      .      .      .        .      .
## EE        0.493  0.936     .      .      .      .      .      .        .      .
## SI        0.561  0.289 0.843      .      .      .      .      .        .      .
## HM        0.641  0.484 0.450  0.920      .      .      .      .        .      .
## FC        0.542  0.668 0.424  0.570  0.797      .      .      .        .      .
## HA        0.551  0.599 0.445  0.639  0.608  0.901      .      .        .      .
## IU        0.650  0.413 0.570  0.616  0.527  0.630  0.910      .        .      .
## TRI_A     0.600  0.437 0.458  0.596  0.494  0.633  0.589  0.844        .      .
## IU*TRI_A -0.017 -0.043 0.166  0.037  0.055  0.098 -0.078  0.136    1.000      .
## TRI_B     0.420  0.617 0.275  0.440  0.583  0.568  0.397  0.429    0.034  0.801
## FC*TRI_B -0.084 -0.223 0.165 -0.130 -0.348 -0.158 -0.069 -0.021    0.259 -0.101
## SNS       0.585  0.528 0.464  0.579  0.544  0.753  0.608  0.526    0.072  0.530
##          FC*TRI_B   SNS
## PE              .     .
## EE              .     .
## SI              .     .
## HM              .     .
## FC              .     .
## HA              .     .
## IU              .     .
## TRI_A           .     .
## IU*TRI_A        .     .
## TRI_B           .     .
## FC*TRI_B    1.000     .
## SNS        -0.082 0.769
##
## FL Criteria table reports square root of AVE on the diagonal and construct correlations on the lower tria
ngle.
```

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$validity$fl_criteria,
            'resumen.xlsx',
            sheetName = "Fornell-Larcker_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.7. fSquare

Hide

```
sum_pls_model_mod_med$fSquare
```

```
##             PE    EE    SI    HM    FC    HA    IU TRI_A IU*TRI_A TRI_B
## PE       0.000 0.000 0.000 0.000 0.000 0.000 0.079 0.000    0.000 0.000
## EE       0.000 0.000 0.000 0.000 0.000 0.000 0.008 0.000    0.000 0.000
## SI       0.000 0.000 0.000 0.000 0.000 0.000 0.066 0.000    0.000 0.000
## HM       0.000 0.000 0.000 0.000 0.000 0.000 0.028 0.000    0.000 0.000
## FC       0.000 0.000 0.000 0.000 0.000 0.000 0.007 0.000    0.000 0.000
## HA       0.000 0.000 0.000 0.000 0.000 0.000 0.087 0.000    0.000 0.000
## IU       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## TRI_A    0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## IU*TRI_A 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## TRI_B    0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## FC*TRI_B 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SNS      0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
##          FC*TRI_B   SNS
## PE          0.000 0.000
## EE          0.000 0.000
## SI          0.000 0.000
## HM          0.000 0.000
## FC          0.000     .
## HA          0.000 0.285
## IU          0.000     .
## TRI_A       0.000     .
## IU*TRI_A    0.000 0.001
## TRI_B       0.000     .
## FC*TRI_B    0.000 0.004
## SNS         0.000 0.000
##
## The fSquare for certain relationships cannot be calculated as the model contains an interaction term and
omitting either the antecedent or moderator in the interaction term will cause model estimation to fail
```

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$fSquare,
            'resumen.xlsx',
            sheetName = "fSquare_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.8. HTMT

Hide

```
sum_pls_model_mod_med$validity$htmt
```

```
##             PE    EE    SI    HM    FC    HA    IU TRI_A IU*TRI_A TRI_B
## PE           .     .     .     .     .     .     .     .        .     .
## EE       0.556     .     .     .     .     .     .     .        .     .
## SI       0.657 0.310     .     .     .     .     .     .        .     .
## HM       0.737 0.524 0.504     .     .     .     .     .        .     .
## FC       0.695 0.815 0.523 0.707     .     .     .     .        .     .
## HA       0.624 0.636 0.486 0.692 0.744     .     .     .        .     .
## IU       0.795 0.474 0.680 0.724 0.688 0.728     .     .        .     .
## TRI_A    0.704 0.482 0.522 0.669 0.620 0.697 0.707     .        .     .
## IU*TRI_A 0.078 0.045 0.183 0.039 0.091 0.100 0.087 0.149        .     .
## TRI_B    0.508 0.692 0.315 0.496 0.744 0.627 0.481 0.489    0.088     .
## FC*TRI_B 0.092 0.231 0.180 0.137 0.415 0.164 0.077 0.051    0.259 0.108
## SNS      0.730 0.618 0.563 0.689 0.721 0.881 0.777 0.636    0.106 0.655
##          FC*TRI_B SNS
## PE              .   .
## EE              .   .
## SI              .   .
## HM              .   .
## FC              .   .
## HA              .   .
## IU              .   .
## TRI_A           .   .
## IU*TRI_A        .   .
## TRI_B           .   .
## FC*TRI_B        .   .
## SNS         0.094   .
```

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$validity$htmt ,
            'resumen.xlsx',
            sheetName = "htmt_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.9. Tabla de correlaciones

Hide

```
sum_pls_model_mod_med$descriptives$correlations$constructs
```

```
##              PE     EE    SI     HM     FC     HA     IU  TRI_A IU*TRI_A  TRI_B
## PE        1.000  0.493 0.561  0.641  0.542  0.551  0.650  0.600   -0.017  0.420
## EE        0.493  1.000 0.289  0.484  0.668  0.599  0.413  0.437   -0.043  0.617
## SI        0.561  0.289 1.000  0.450  0.424  0.445  0.570  0.458    0.166  0.275
## HM        0.641  0.484 0.450  1.000  0.570  0.639  0.616  0.596    0.037  0.440
## FC        0.542  0.668 0.424  0.570  1.000  0.608  0.527  0.494    0.055  0.583
## HA        0.551  0.599 0.445  0.639  0.608  1.000  0.630  0.633    0.098  0.568
## IU        0.650  0.413 0.570  0.616  0.527  0.630  1.000  0.589   -0.078  0.397
## TRI_A     0.600  0.437 0.458  0.596  0.494  0.633  0.589  1.000    0.136  0.429
## IU*TRI_A -0.017 -0.043 0.166  0.037  0.055  0.098 -0.078  0.136    1.000  0.034
## TRI_B     0.420  0.617 0.275  0.440  0.583  0.568  0.397  0.429    0.034  1.000
## FC*TRI_B -0.084 -0.223 0.165 -0.130 -0.348 -0.158 -0.069 -0.021    0.259 -0.101
## SNS       0.585  0.528 0.464  0.579  0.544  0.753  0.608  0.526    0.072  0.530
##          FC*TRI_B    SNS
## PE         -0.084  0.585
## EE         -0.223  0.528
## SI          0.165  0.464
## HM         -0.130  0.579
## FC         -0.348  0.544
## HA         -0.158  0.753
## IU         -0.069  0.608
## TRI_A      -0.021  0.526
## IU*TRI_A    0.259  0.072
## TRI_B      -0.101  0.530
## FC*TRI_B    1.000 -0.082
## SNS        -0.082  1.000
```

Hide

```
write.xlsx2(x=sum_pls_model_mod_med$descriptives$correlations$constructs  ,
            'resumen.xlsx',
            sheetName = "Correl_constructos_moderador",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

## H.4.10. Otros

b) Efectos totales

c) Efectos indirectos

d) Puntuaciones estimadas para los constructos

e) seleccion de modelo BIC, AIC

Hide

```
sum_pls_model_mod_med$total_effects             ## b)
```

```
##             PE    EE    SI    HM    FC    HA     IU TRI_A IU*TRI_A TRI_B
## PE       0.000 0.000 0.000 0.000 0.000 0.000  0.269 0.000    0.000 0.000
## EE       0.000 0.000 0.000 0.000 0.000 0.000 -0.085 0.000    0.000 0.000
## SI       0.000 0.000 0.000 0.000 0.000 0.000  0.209 0.000    0.000 0.000
## HM       0.000 0.000 0.000 0.000 0.000 0.000  0.159 0.000    0.000 0.000
## FC       0.000 0.000 0.000 0.000 0.000 0.000  0.084 0.000    0.000 0.000
## HA       0.000 0.000 0.000 0.000 0.000 0.000  0.286 0.000    0.000 0.000
## IU       0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
## TRI_A    0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
## IU*TRI_A 0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
## TRI_B    0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
## FC*TRI_B 0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
## SNS      0.000 0.000 0.000 0.000 0.000 0.000  0.000 0.000    0.000 0.000
##          FC*TRI_B    SNS
## PE          0.000  0.055
## EE          0.000 -0.017
## SI          0.000  0.043
## HM          0.000  0.033
## FC          0.000  0.084
## HA          0.000  0.590
## IU          0.000  0.205
## TRI_A       0.000 -0.017
## IU*TRI_A    0.000  0.021
## TRI_B       0.000  0.119
## FC*TRI_B    0.000  0.040
## SNS         0.000  0.000
```

Hide

```
sum_pls_model_mod_med$total_indirect_effects     ## c)
```

```
##             PE    EE    SI    HM    FC    HA    IU TRI_A IU*TRI_A TRI_B
## PE       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## EE       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SI       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## HM       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## FC       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## HA       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## IU       0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## TRI_A    0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## IU*TRI_A 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## TRI_B    0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## FC*TRI_B 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
## SNS      0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000    0.000 0.000
##          FC*TRI_B    SNS
## PE          0.000  0.055
## EE          0.000 -0.017
## SI          0.000  0.043
## HM          0.000  0.033
## FC          0.000  0.017
## HA          0.000  0.059
## IU          0.000  0.000
## TRI_A       0.000  0.000
## IU*TRI_A    0.000  0.000
## TRI_B       0.000  0.000
## FC*TRI_B    0.000  0.000
## SNS         0.000  0.000
```

Hide

```
# sum_pls_model_mod_med$composite_scores          ## d)
sum_pls_model_mod_med$it_criteria               ## e)
```

```
##           IU      SNS
## AIC -320.516 -348.341
## BIC -292.880 -316.757
```

# H.5. Evaluar Boot Moderador

Hide

```
sum_boot_pls_model_mod$bootstrapped_paths
```

```
##                   Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI
## PE  ->  IU               0.269          0.269        0.059   4.571   0.162
## EE  ->  IU              -0.085         -0.084        0.051  -1.673  -0.180
## SI  ->  IU               0.209          0.210        0.049   4.288   0.122
## HM  ->  IU               0.159          0.156        0.049   3.284   0.057
## FC  ->  IU               0.084          0.087        0.054   1.549  -0.015
## FC  ->  SNS              0.066          0.064        0.049   1.344  -0.021
## HA  ->  IU               0.286          0.285        0.049   5.792   0.185
## HA  ->  SNS              0.531          0.532        0.051  10.484   0.434
## IU  ->  SNS              0.205          0.208        0.046   4.486   0.121
## TRI_A  ->  SNS          -0.017         -0.019        0.048  -0.346  -0.113
## IU*TRI_A  ->  SNS        0.021          0.024        0.042   0.516  -0.051
## TRI_B  ->  SNS           0.119          0.120        0.043   2.790   0.044
## FC*TRI_B  ->  SNS        0.040          0.042        0.034   1.185  -0.022
##                   97.5% CI
## PE  ->  IU           0.387
## EE  ->  IU           0.017
## SI  ->  IU           0.300
## HM  ->  IU           0.250
## FC  ->  IU           0.198
## FC  ->  SNS          0.160
## HA  ->  IU           0.383
## HA  ->  SNS          0.627
## IU  ->  SNS          0.292
## TRI_A  ->  SNS       0.067
## IU*TRI_A  ->  SNS    0.110
## TRI_B  ->  SNS       0.198
## FC*TRI_B  ->  SNS    0.103
```

Hide

```
sum_boot$bootstrapped_HTMT
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## PE  ->  EE          0.556          0.556        0.045  12.385   0.464    0.641
## PE  ->  SI          0.657          0.660        0.048  13.782   0.568    0.753
## PE  ->  FC          0.695          0.698        0.052  13.367   0.594    0.790
## PE  ->  HM          0.737          0.737        0.035  21.122   0.670    0.800
## PE  ->  HA          0.624          0.625        0.040  15.430   0.540    0.698
## PE  ->  IU          0.795          0.796        0.038  20.780   0.723    0.864
## PE  ->  SNS         0.730          0.731        0.048  15.137   0.636    0.817
## EE  ->  SI          0.310          0.311        0.055   5.681   0.198    0.414
## EE  ->  FC          0.815          0.815        0.035  23.482   0.740    0.878
## EE  ->  HM          0.524          0.522        0.045  11.628   0.437    0.600
## EE  ->  HA          0.636          0.637        0.034  18.793   0.575    0.702
## EE  ->  IU          0.474          0.473        0.050   9.402   0.367    0.569
## EE  ->  SNS         0.618          0.617        0.040  15.340   0.534    0.687
## SI  ->  FC          0.523          0.531        0.064   8.182   0.401    0.645
## SI  ->  HM          0.504          0.506        0.049  10.219   0.412    0.601
## SI  ->  HA          0.486          0.484        0.047  10.344   0.385    0.576
## SI  ->  IU          0.680          0.680        0.048  14.067   0.581    0.774
## SI  ->  SNS         0.563          0.563        0.050  11.173   0.461    0.661
## FC  ->  HM          0.707          0.708        0.041  17.348   0.618    0.786
## FC  ->  HA          0.744          0.746        0.043  17.333   0.659    0.825
## FC  ->  IU          0.688          0.689        0.043  16.039   0.609    0.784
## FC  ->  SNS         0.721          0.722        0.041  17.679   0.645    0.794
## HM  ->  HA          0.692          0.691        0.034  20.184   0.620    0.753
## HM  ->  IU          0.724          0.724        0.036  20.304   0.653    0.791
## HM  ->  SNS         0.689          0.689        0.042  16.256   0.607    0.775
## HA  ->  IU          0.728          0.729        0.041  17.779   0.643    0.805
## HA  ->  SNS         0.881          0.880        0.023  37.923   0.833    0.921
## IU  ->  SNS         0.777          0.777        0.042  18.695   0.696    0.858
```

Hide

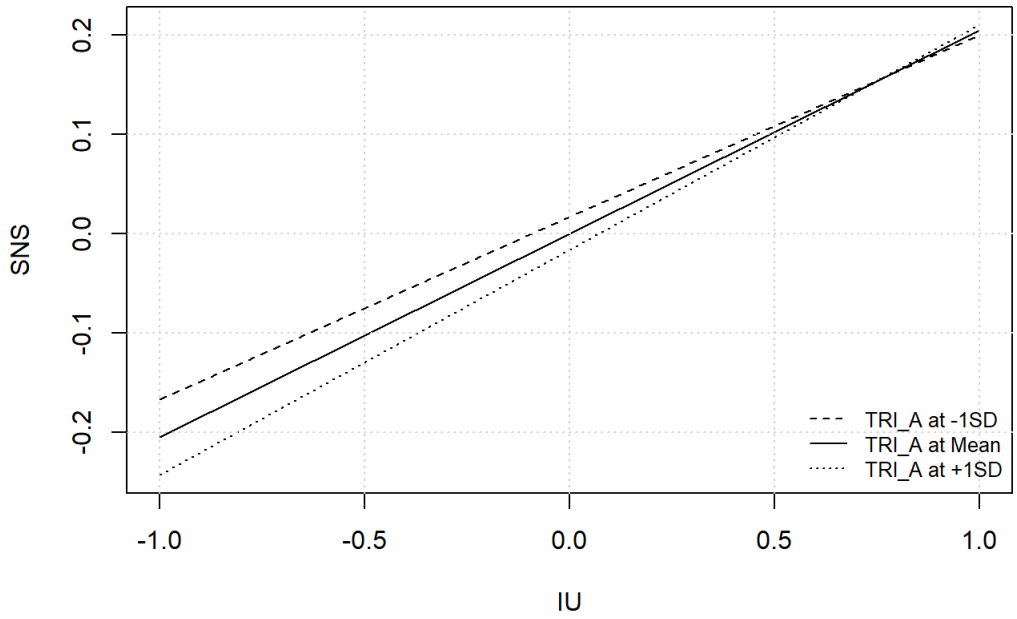
```
summary_estimacion_model$validity$htmt
```

```
##        PE    EE    SI    FC    HM    HA    IU SNS
## PE      .     .     .     .     .     .     .   .
## EE  0.556     .     .     .     .     .     .   .
## SI  0.657 0.310     .     .     .     .     .   .
## FC  0.695 0.815 0.523     .     .     .     .   .
## HM  0.737 0.524 0.504 0.707     .     .     .   .
## HA  0.624 0.636 0.486 0.744 0.692     .     .   .
## IU  0.795 0.474 0.680 0.688 0.724 0.728     .   .
## SNS 0.730 0.618 0.563 0.721 0.689 0.881 0.777   .
```

# H.6. Simple slope analysis plot

Hide

```
slope_analysis(
moderated_model = pls_model_mod_med,
dv = 'SNS',
moderator = 'TRI_A',
iv = 'IU',
leg_place = 'bottomright')
```

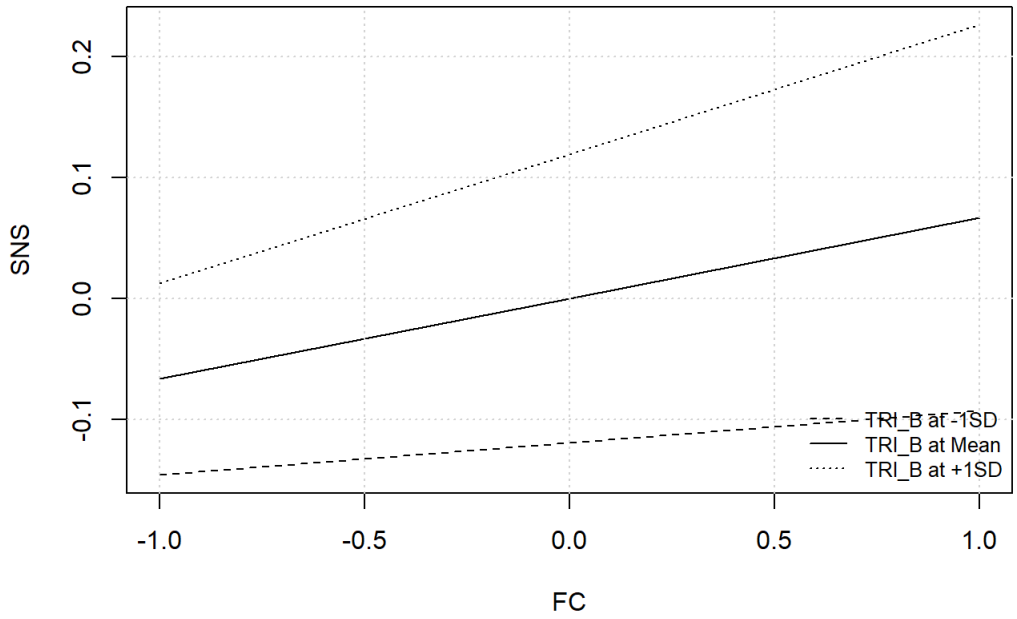



Hide

```
#plot_interaction(pls_model_mod_med, 'IU*TRI_A', 'SNS')
```

Hide

```
slope_analysis(
moderated_model = pls_model_mod_med,
dv = 'SNS',
moderator = 'TRI_B',
iv = 'FC',
leg_place = 'bottomright')
```




Hide

```
#plot_interaction(pls_model_mod_med, 'FC*TRI_B', 'SNS')
```

# I. Comparación con otros modelos

## I.1 Creamos los modelos adicionales.

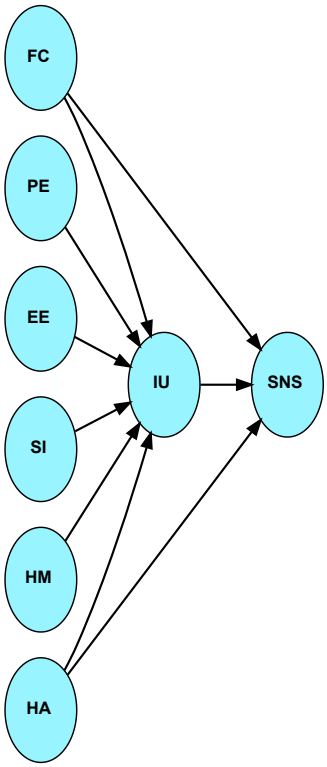
*Nota:* No se modificará el modelo de medida. Comparación es a nivel de modelo estructural

Hide

```
#modelo 0 #modelo evaluado creado en E.2
#modelo_estruc <- relationships(
#  paths(from = c('PE', 'EE', 'SI', 'FC', 'HM', "HA"), to = c('IU')),
#  paths(from = c('FC', 'HA', "IU"), to = c('SNS'))
# )


# Modelo 1
structural_model1 <- relationships(
  paths(from = c('PE', 'EE', 'SI', 'FC', 'HM', "HA"), to = c('IU')),
  paths(from = c('HA', "IU"), to = c('SNS'))
)

# Modelo 2
structural_model2 <- relationships(
  paths(from = c('PE', 'EE', 'SI', 'FC', 'HM', "HA"), to = c('IU')),
  paths(from = c("IU"), to = c('SNS'))
)

# Modelo 3
structural_model3 <- relationships(
  paths(from = c('PE', 'EE', 'SI',  'HM' ), to = c('IU')),
  paths(from = c( 'HA', 'FC','IU'), to = c('SNS'))
)
```

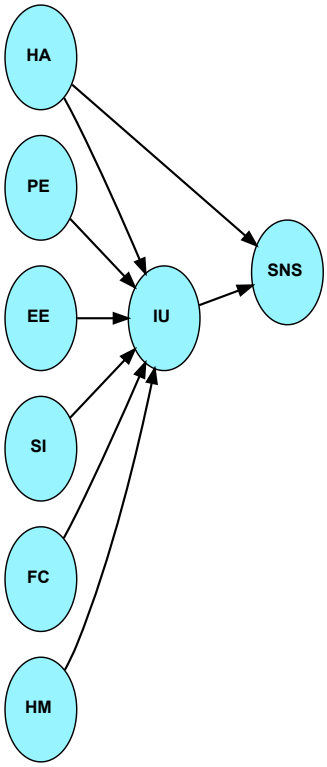
Hide

```
plot(modelo_estruc) # Modelo inicial
```



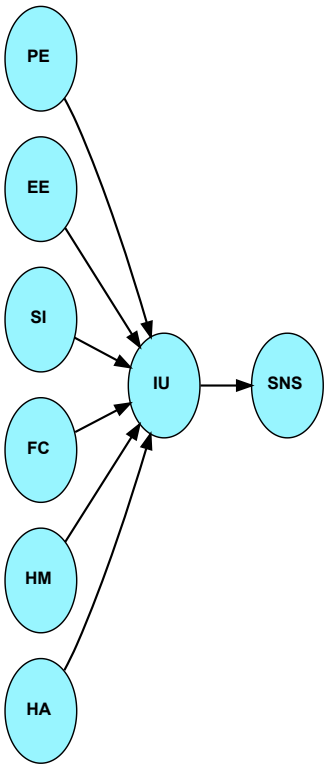

Hide

```
plot(structural_model1)
```



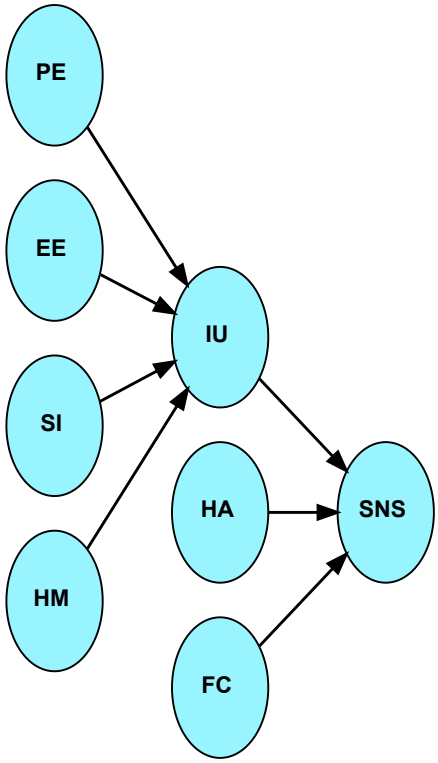

Hide

```
plot(structural_model2)
```




Hide

```
plot(structural_model3)
```




# I.2 Generamos los modelos

Hide

```
pls_model1 <- estimate_pls(
  data = pls_data2,
  measurement_model = modelo_medida,
  structural_model = structural_model1,
  missing_value = '-99'
)
sum_model1 <- summary(pls_model1)

pls_model2 <- estimate_pls(
  data = pls_data2,
  measurement_model = modelo_medida,
  structural_model = structural_model2,
  missing_value = '-99'
)
sum_model2 <- summary(pls_model2)

pls_model3 <- estimate_pls(
 data = pls_data2,
  measurement_model = modelo_medida,
  structural_model = structural_model3,
  missing_value = '-99'
)
sum_model3 <- summary(pls_model3)
```

# I.3 Comparamos los modelos

Hide

```
summary_estimacion_model$it_criteria
```

```
##           IU      SNS
## AIC -320.517 -345.750
## BIC -292.881 -329.958
```

Hide

```
sum_model1$it_criteria
```

```
##           IU      SNS
## AIC -320.542 -343.473
## BIC -292.906 -331.629
```

Hide

```
sum_model2$it_criteria
```

```
##           IU      SNS
## AIC -320.758 -177.370
## BIC -293.122 -169.474
```

Hide

```
sum_model3$it_criteria
```

```
##           IU      SNS
## AIC -286.121 -346.233
## BIC -266.381 -330.441
```

Hide

```
# Menor BIC  tiene mejor poder predictivo
```

Hide

```
# Recogemos los valores BIC de cada modelo.
#Nos centramos en este ya que es el que intermedia, el que está cambiando los modelos

itcriteria_vector <- c(summary_estimacion_model$it_criteria['BIC', 'IU'],
                       sum_model1$it_criteria['BIC', 'IU'],
                       sum_model2$it_criteria['BIC', 'IU'],
                       sum_model3$it_criteria['BIC', 'IU'])

itcriteria_vector2 <- c(summary_estimacion_model$it_criteria['BIC', 'SNS'],
                       sum_model1$it_criteria['BIC', 'SNS'],
                       sum_model2$it_criteria['BIC', 'SNS'],
                       sum_model3$it_criteria['BIC', 'SNS'])
# Asignamos los nombres de los modelos a IT Criteria vector
names(itcriteria_vector) <- c('Original','Model1', 'Model2', 'Model3')
names(itcriteria_vector2) <- c('Original','Model1', 'Model2', 'Model3')
```

Hide

```
# Valores BIC por modelos # El menor BIC seleccionamos - IU
itcriteria_vector
```

```
##  Original    Model1    Model2    Model3
## -292.8812 -292.9058 -293.1221 -266.3810
```

Hide

```
# Calcula BIC Akaike # Mayor implica  mejor poder predictivo - IU
compute_itcriteria_weights(itcriteria_vector)
```

```
##     Original       Model1       Model2       Model3
## 3.184311e-01 3.223746e-01 3.591937e-01 5.605111e-07
```

Hide

```
# Valores BIC para SNS en distintos modelos - SNS
itcriteria_vector2
```

```
##  Original    Model1    Model2    Model3
## -329.9579 -331.6285 -169.4738 -330.4410
```

Hide

```
# Calcula BIC Akaike # Mayor implica  mejor poder predictivo -SNS
compute_itcriteria_weights(itcriteria_vector2)
```

```
##     Original       Model1       Model2       Model3
## 2.183965e-01 5.035313e-01 3.094420e-36 2.780722e-01
```

# J. Análisis Multigrupo

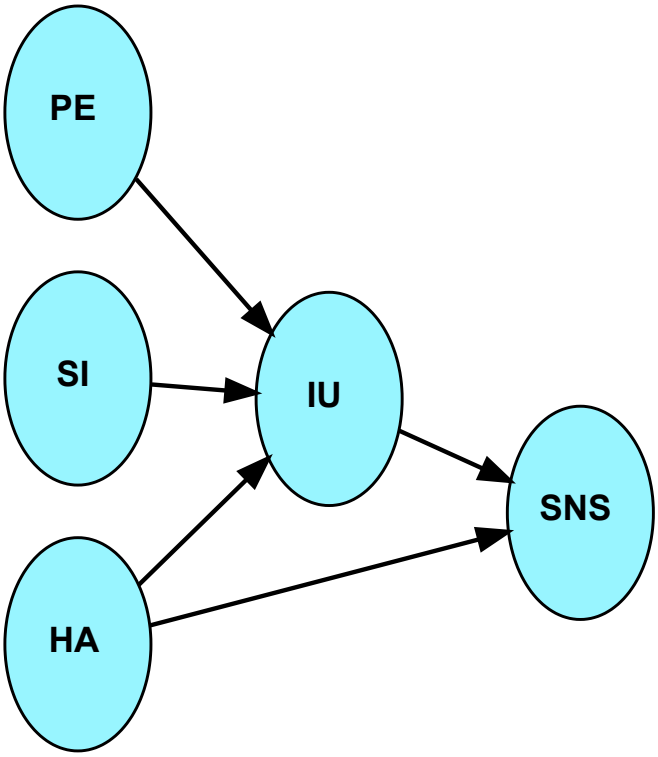
Asumiremos que se desea crear multigrupo con la variable género.

*NOTA:* Solo se puede hacer multigrupo de 2 grupos. Más grupos no es posible en esta versión.

*NOTA2: Cambiaremos el modelo estructural para que MGA sea significativo*

Hide

```
modelo_estruc_mga <- relationships(
  paths(from = c('PE', 'SI',  "HA"), to = c('IU')),
  paths(from = c('HA', "IU"), to = c('SNS'))
)

plot(modelo_estruc_mga)
```




Hide

```
mga_esti <- estimate_pls(data = pls_data2,
                         measurement_model = modelo_medida, #E1
                         structural_model = modelo_estruc_mga,
                         missing = mean_replacement,
                         missing_value = -99)
```

# J.1. Preparación de variable

En caso que no se haya convertido en D.2

Hide

```
#pls_data2$GENDER
#pls_data2$GENERO = ifelse(pls_data2$GENDER=='Male', 1, 2)
#pls_data2$GENDER
#pls_data2$REGION3= ifelse(pls_data2$REGION=='Coquimbo', 1, 2)  #59
```

Hide

```
sum(pls_data2$GENERO==1) #Male
```

```
## [1] 170
```

Hide

```
sum(pls_data2$GENERO==2)
```

```
## [1] 213
```

# J.2. Generamos el multigrupo

En este caso probaremos 2 MGA uno con el Género y otro con la región

Hide

```
pls_mga <- estimate_pls_mga(mga_esti,
                            pls_data2$GENERO == 1,
                            nboot=500) ## sobre 5000
```

Hide

```
pls_mga_region <- estimate_pls_mga(mga_esti,
          pls_data2$REGION3 == 1,
          nboot=500) ## sobre 5000
```

# J.3. Análisis del Multigrupo

Hide

```
Desde <- pls_mga$source
Hasta <- pls_mga$target
Grupo_1 <- pls_mga$group1_beta
Grupo_2 <- pls_mga$group2_beta
p_value <- pls_mga$pls_mga_p

mga_1 <- data.frame(Desde, Hasta, "B Grupo1" = Grupo_1, "B Grupo2" = Grupo_2, p_value)
mga_1
```

| **Desde** <chr> | **Hasta** <chr> | **B.Grupo1** <dbl> | **B.Grupo2** <dbl> | **p_value** <dbl> |
|---|---|---|---|---|
| PE | IU | 0.3114443 | 0.5108957 | 0.868380 |
| SI | IU | 0.2771152 | 0.2064978 | 0.340768 |
| HA | IU | 0.4270876 | 0.2927959 | 0.147260 |
| HA | SNS | 0.6030178 | 0.7135530 | 0.819468 |
| IU | SNS | 0.3719146 | 0.2347841 | 0.162896 |

5 rows

Hide

```
# p-values <0.05 significa que hay diferencia significativa, entre los grupos por cada Hipo.
```

Hide

```
Desde <- pls_mga_region$source
Hasta <- pls_mga_region$target
Grupo_1 <- pls_mga_region$group1_beta
Grupo_2 <- pls_mga_region$group2_beta
p_value <- pls_mga_region$pls_mga_p

mga_2 <- data.frame(Desde, Hasta, "B Grupo1" = Grupo_1, "B Grupo2" = Grupo_2, p_value)
mga_2
```

| Desde <chr> | Hasta <chr> | B.Grupo1 <dbl> | B.Grupo2 <dbl> | p_value <dbl> |
|---|---|---|---|---|
| PE | IU | 0.3315039 | 0.4513070 | 0.676172 |
| SI | IU | 0.2424379 | 0.2701690 | 0.575760 |
| HA | IU | 0.4742478 | 0.2709948 | 0.090972 |
| HA | SNS | 0.5882664 | 0.7340213 | 0.840268 |
| IU | SNS | 0.3528072 | 0.2446508 | 0.219704 |

5 rows

Hide

```
# p-values <0.05 significa que hay diferencia significativa, entre los grupos por cada Hipo.
```

Hide

```
write.xlsx2(x=pls_mga  ,
            'resumen.xlsx',
            sheetName = "MGA",
            col.names = TRUE,
            row.names = TRUE,
            append = TRUE,
            showNA = TRUE,
            password = NULL)
```

# J.4. Análisis MICOM

NOTA: Utilizaremos paquete cSEM y sentencias en Lavaan

Modelo de medida y estructural se crean en conjunto entre “”

Hide

```
cSmodel2 <- "
# modelo estructural
SNS  ~ IU +  HA
IU  ~  HA + SI + PE
# modelo de medida
PE =~ PE1 + PE2 + PE3 + PE4
SI =~ SI1 + SI2 + SI3 + SI4
HA =~ HA1 + HA2 + HA3 + HA4 + HA5
IU =~ IU1 + IU2
SNS =~ U1 + U2+ U3 + U4
"
```

Generamos data y probamos los modelos

Hide

```
#1 Data género
g11 <- pls_data2[(pls_data2$GENERO==1),]
g12 <- pls_data2[(pls_data2$GENERO!=1 ),]

#2 Data región
g21 <- pls_data2[(pls_data2$REGION=='Coquimbo'),]
g22 <- pls_data2[(pls_data2$REGION!='Coquimbo' ),]

csem_results1 <- csem(.data = g11, cSmodel2)
csem_results2 <- csem(.data = g12, cSmodel2)


## Análisis con cSEM
csem_results1 <- csem(.data = g11, cSmodel2)
csem_results2 <- csem(.data = g12, cSmodel2)

#Si en Status da "not Ok", no se puede usar para MGA
verify(csem_results1)
```

```
## ________________________________________________________________________________
##
## Verify admissibility:
##
##   admissible
##
## Details:
##
##   Code   Status    Description
##   1      ok        Convergence achieved
##   2      ok        All absolute standardized loading estimates <= 1
##   3      ok        Construct VCV is positive semi-definite
##   4      ok        All reliability estimates <= 1
##   5      ok        Model-implied indicator VCV is positive semi-definite
## ________________________________________________________________________________
```

Hide

```
verify(csem_results2)
```

```
## ________________________________________________________________________________
##
## Verify admissibility:
##
##   admissible
##
## Details:
##
##   Code   Status    Description
##   1      ok        Convergence achieved
##   2      ok        All absolute standardized loading estimates <= 1
##   3      ok        Construct VCV is positive semi-definite
##   4      ok        All reliability estimates <= 1
##   5      ok        Model-implied indicator VCV is positive semi-definite
## ________________________________________________________________________________
```

Hide

```
## Análisis con cSEM
csem_results1 <- csem(.data = g21, cSmodel2)
csem_results2 <- csem(.data = g22, cSmodel2)

#Si en Status da "not Ok", no se puede usar para MGA
verify(csem_results1)
```

```
## ________________________________________________________________________
##
## Verify admissibility:
##
##   inadmissible
##
## Details:
##
##   Code   Status    Description
##   1      ok        Convergence achieved
##   2      not ok    All absolute standardized loading estimates <= 1
##   3      ok        Construct VCV is positive semi-definite
##   4      ok        All reliability estimates <= 1
##   5      ok        Model-implied indicator VCV is positive semi-definite
## ________________________________________________________________________
```

Hide

```
verify(csem_results2)
```

```
## ________________________________________________________________________
##
## Verify admissibility:
##
##   admissible
##
## Details:
##
##   Code   Status    Description
##   1      ok        Convergence achieved
##   2      ok        All absolute standardized loading estimates <= 1
##   3      ok        Construct VCV is positive semi-definite
##   4      ok        All reliability estimates <= 1
##   5      ok        Model-implied indicator VCV is positive semi-definite
## ________________________________________________________________________
```

Test MICOM

Hide

```
csem_results <- csem(.data = list("group1" = g11, "group2" = g12), # Data creada por grupo
                     cSmodel2, .resample_method = "bootstrap",
                    .R = 500) ##Subir número


testMICOM(csem_results,
          .R = 500)  ##Subir número
```

```
## ________ Test for measurement invariance based on Henseler et al (2016) ________
```

```
## ________________________________________________________________________________
## -------- Test for measurement invariance based on Henseler et al (2016) --------
## ======================== Step 1 - Configural invariance ========================
##
##  Configural invariance is a precondition for step 2 and 3.
##  Do not proceed to interpret results unless
##  configural invariance has been established.
##
## ======================== Step 2 - Compositional invariance =====================
##
## Null hypothesis:
##
##        +-----------------------------------------------------------------+
##        |                                                                 |
##        |   H0: Compositional measurement invariance of the constructs.   |
##        |                                                                 |
##        +-----------------------------------------------------------------+
##
## Test statistic and p-value:
##
##   Compared groups: group1_group2
##                               p-value by adjustment
##  Construct   Test statistic      none
##  HA              1.0000        0.9087
##  SI              0.9980        0.2527
##  PE              0.9995        0.7983
##  IU              1.0000        0.6391
##  SNS             0.9995        0.5796
##
##
## ================= Step 3 - Equality of the means and variances =================
##
## Null hypothesis:
##
##           +-------------------------------------------------------------+
##           |                                                             |
##           |   1. H0: Difference between group means is zero             |
##           |   2. H0: Log of the ratio of the group variances is zero    |
##           |                                                             |
##           +-------------------------------------------------------------+
##
## Test statistic and critical values:
##
##   Compared groups: group1_group2
##
##  Mean
##                               p-value by adjustment
##  Construct   Test statistic      none
##  HA              0.0153        0.8740
##  SI              -0.0230       0.8100
##  PE              0.1527        0.1440
##  IU              -0.0701       0.4700
##  SNS             0.0132        0.9180
##
##  Var
##                               p-value by adjustment
##  Construct   Test statistic      none
##  HA              -0.1683       0.1180
##  SI              -0.0284       0.8060
##  PE              -0.2563       0.0880
```

```
##  IU              -0.1839       0.3260
##  SNS             -0.0355       0.7420
##
##
## Additional information:
##
##  Out of 500 permutation runs, 471 where admissible.
##  See ?verify() for what constitutes an inadmissible result.
##
##  The seed used was: -272468639
##
##  Number of observations per group:
##
##  Group       No. observations
##  group1      170
##  group2      213
## ________________________________________________________________________________
```

Test de comparacion MGA

Hide

```
testmgd <- testMGD(csem_results, .parameters_to_compare = NULL,
                   .alpha = 0.05,
        .approach_p_adjust = c("none", "bonferroni"),   ## Tipo de ajuste a los p
        .R_permutation         = 60,
        .R_bootstrap = 60,  #Subir número
        .saturated             = FALSE,
        .approach_mgd = "all", #test a aplicar
        .output_type           = "complete", #"c("complete", "structured"),
        .eval_plan             = c("sequential", "multicore", "multisession"),
        .verbose = FALSE)
```

```
## Warning: The following warning occured in the testMGD() function:
## Currently, there is no p-value adjustment possible for the approach suggested by
## Henseler (2007), CI_para, and CI_overlap. Adjustment is ignored for these approaches.
```

Hide

```
###Test no rechazarán sus respectivas H0, los grupos son prácticamente idénticos.

testmgd
```

```
## ___________________________________________________________________________
## ---------------------------------- Overview ----------------------------------
##
##  Total permutation runs            = 62
##  Admissible permutation results    = 60
##  Permutation seed                  = -130546490
##
##  Total bootstrap runs              = 500
##  Admissible bootstrap results:
##
##  Group         Admissibles
##  group1            380
##  group2            482
##
##  Bootstrap seed:
##
##  Group            Seed
##  group1          -873491064
##  group2           628296840
##
##  Number of observations per group:
##
##  Group        No. Obs.
##  group1          170
##  group2          213
##
##  Overall decision (based on alpha = 5%):
##
##               p_adjust = 'none'p_adjust = 'bonferroni'
##  Sarstedt                reject              reject
##  Chin             Do not reject       Do not reject
##  Keil             Do not reject       Do not reject
##  Nitzl            Do not reject       Do not reject
##
##  For details on a particular approach type:
##
##      - `print(<object-name>, .approach_mgd = '<approach-name>')`
##
## ___________________________________________________________________________
```

# K. Análisis Segundo Orden

Data contiene TRI el cual está conformado por 4 constructos, asumiremos que corresponde a un constructo de segundo orden el que afecta a IU

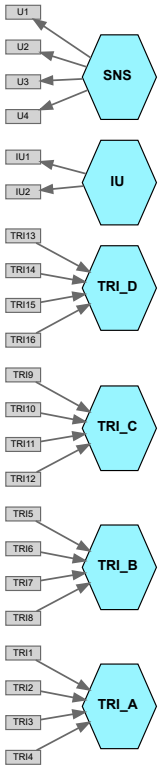
# K.1. Evaluar constructos de orden inferior

## K.1.1. Modelo de medida

*K.1.1.a Modelo de medida Formativo*

Hide

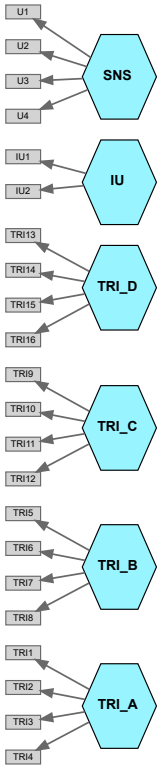
```
m_medida_1 <- constructs(
  composite('TRI_A', multi_items('TRI', 1:4), weights = mode_B ), #Formativo de ejemplo
  composite('TRI_B', multi_items('TRI', 5:8), weights = mode_B ),
  composite('TRI_C', multi_items('TRI', 9:12), weights = mode_B ),
  composite('TRI_D', multi_items('TRI', 13:16), weights = mode_B ),
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4))
 )
plot(m_medida_1)
```




*K.1.2. Modelo de medida Reflectivo*

Hide

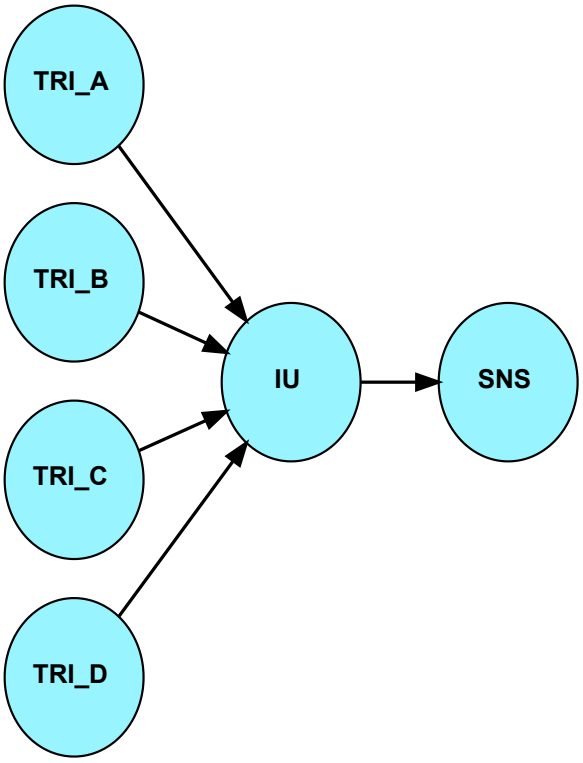
```
m_medida_2 <- constructs(
  composite('TRI_A', multi_items('TRI', 1:4)),
  composite('TRI_B', multi_items('TRI', 5:8)),
  composite('TRI_C', multi_items('TRI', 9:12)),
  composite('TRI_D', multi_items('TRI', 13:16) ),
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4))
 )

plot(m_medida_2)
```




## K.1.2. Modelo estructural

Hide

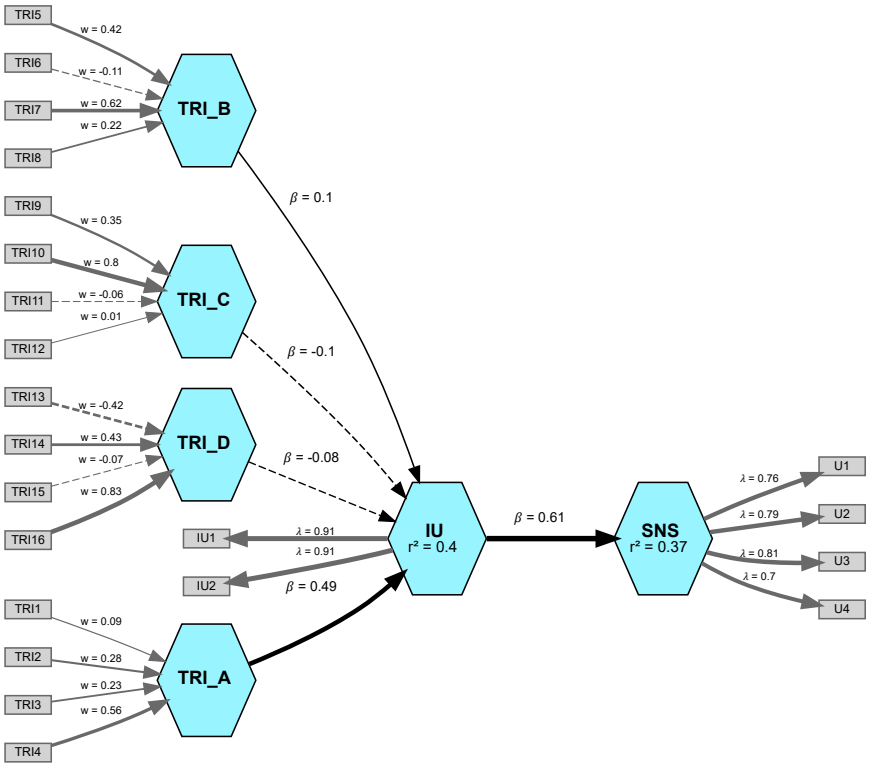
```
m_estruc_1  <- relationships(
  paths(from = c('TRI_A', 'TRI_B', 'TRI_C', 'TRI_D'), to = c('IU')),
  paths(from = c("IU"), to = c('SNS'))
)
plot(m_estruc_1)
```



## K.1.3. Estimación modelo

Hide

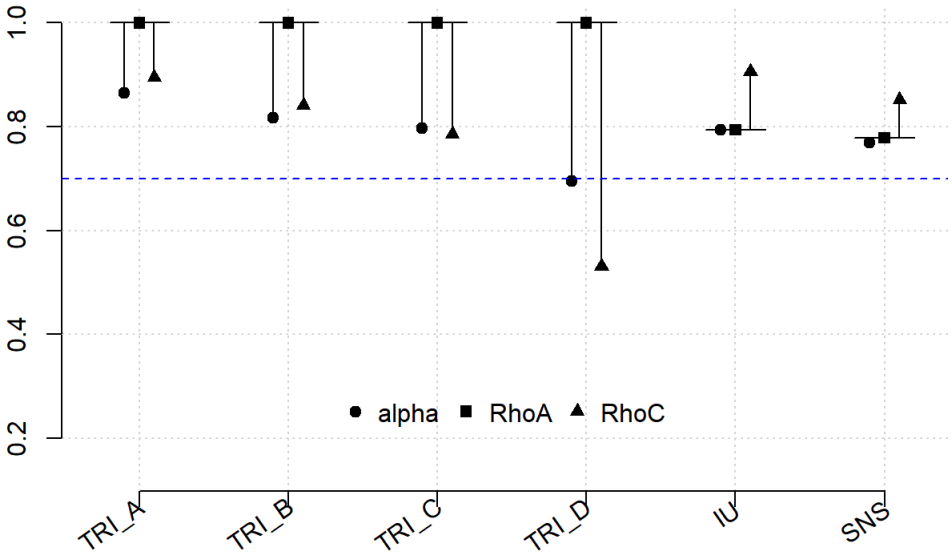
```
estimacion_model_1 <- estimate_pls(data = pls_data2,
                                   measurement_model = m_medida_1,  #K.1.1. - modelo de medida
                                   structural_model = m_estruc_1,   #K.1.2.  - modelo estructural
                                   inner_weights = path_weighting,
                                   # path_weighting para path weighting (default) o path_factorial para f
        actor weighting,
                                   missing = mean_replacement,
                                   missing_value = '-99' )

summary_m_1 = summary(estimacion_model_1)

plot(estimacion_model_1)
```




## K.1.4. Evaluación del modelo de orden inferior

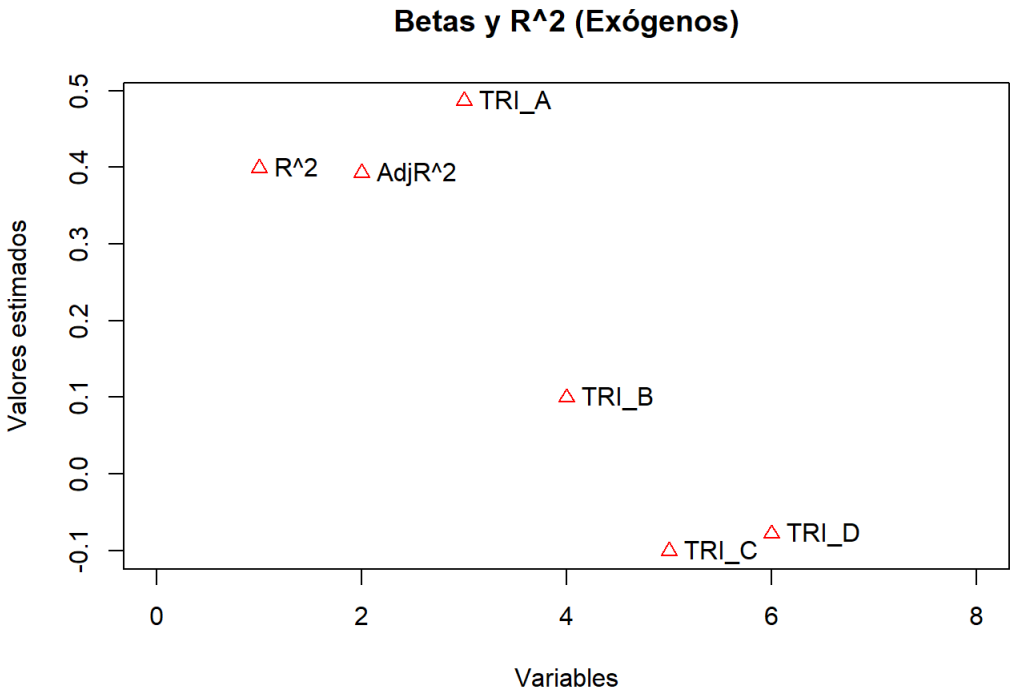
Hide

```
plot(summary_m_1$reliability, title =  "Fig. : Fiabilidad orden inferior")
```



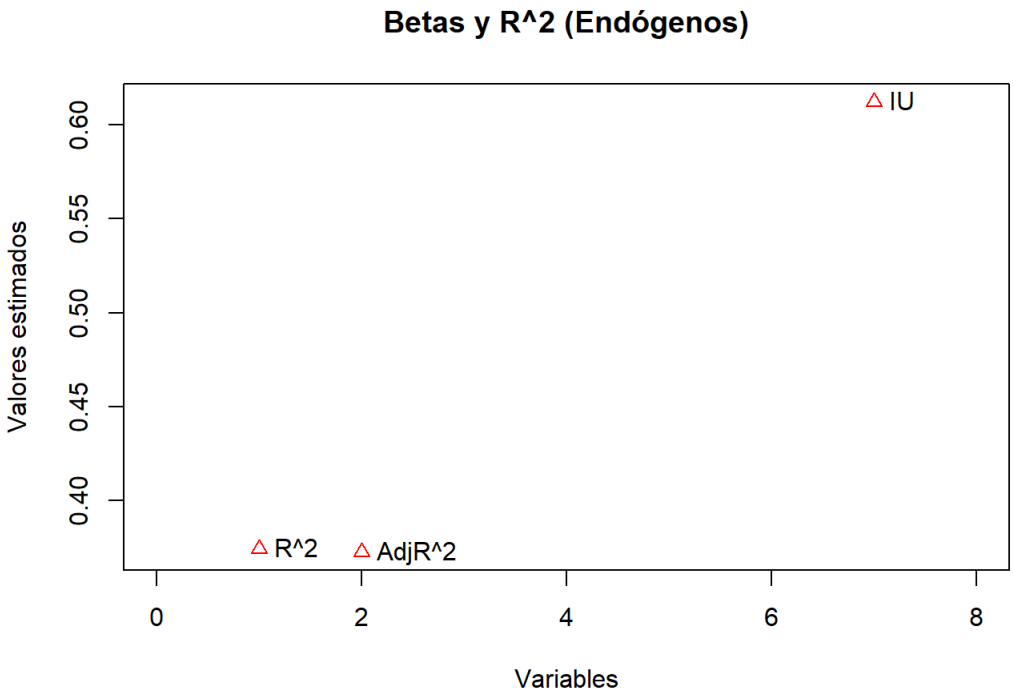

Hide

```
plot(summary_m_1$paths[,1], pch = 2, col = "red", main="Betas y R^2 (Exógenos)",
     xlab = "Variables", ylab = "Valores estimados", xlim = c(0,length(row.names(summary_m_1$paths))+1)
     )
text(summary_m_1$paths[,1],labels = row.names(summary_m_1$paths) , pos = 4)
```




Hide

```
plot(summary_m_1$paths[,2], pch = 2, col = "red", main="Betas y R^2 (Endógenos)",
     xlab = "Variables", ylab = "Valores estimados" , xlim = c(0,length(row.names(summary_m_1$paths))+1) )
text(summary_m_1$paths[,2],labels = row.names(summary_m_1$paths) , pos = 4)
```




Hide

```
summary_m_1$reliability
```

```
##       alpha  rhoC   AVE  rhoA
## TRI_A 0.866 0.895 0.683 1.000
## TRI_B 0.816 0.841 0.577 1.000
## TRI_C 0.797 0.786 0.497 1.000
## TRI_D 0.695 0.532 0.333 1.000
## IU    0.794 0.906 0.829 0.794
## SNS   0.769 0.852 0.591 0.778
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5
```

Hide

```
summary_m_1$loading
```

```
##        TRI_A  TRI_B  TRI_C  TRI_D     IU    SNS
## TRI1   0.718  0.000 -0.000 -0.000  0.000  0.000
## TRI2   0.794  0.000 -0.000 -0.000  0.000  0.000
## TRI3   0.858  0.000 -0.000 -0.000  0.000  0.000
## TRI4   0.923  0.000 -0.000 -0.000  0.000  0.000
## TRI5   0.000  0.792 -0.000 -0.000  0.000  0.000
## TRI6   0.000  0.562 -0.000 -0.000  0.000  0.000
## TRI7   0.000  0.926 -0.000 -0.000  0.000  0.000
## TRI8   0.000  0.713 -0.000 -0.000  0.000  0.000
## TRI9  -0.000 -0.000  0.736  0.000 -0.000 -0.000
## TRI10 -0.000 -0.000  0.955  0.000 -0.000 -0.000
## TRI11 -0.000 -0.000  0.449  0.000 -0.000 -0.000
## TRI12 -0.000 -0.000  0.576  0.000 -0.000 -0.000
## TRI13  0.000  0.000  0.000 -0.157  0.000  0.000
## TRI14 -0.000 -0.000  0.000  0.499 -0.000 -0.000
## TRI15 -0.000 -0.000  0.000  0.494 -0.000 -0.000
## TRI16 -0.000 -0.000  0.000  0.904 -0.000 -0.000
## IU1    0.000  0.000 -0.000 -0.000  0.907  0.000
## IU2    0.000  0.000 -0.000 -0.000  0.913  0.000
## U1     0.000  0.000 -0.000 -0.000  0.000  0.763
## U2     0.000  0.000 -0.000 -0.000  0.000  0.793
## U3     0.000  0.000 -0.000 -0.000  0.000  0.814
## U4     0.000  0.000 -0.000 -0.000  0.000  0.700
```

Hide

```
summary_m_1$validity$fl_criteria
```

```
##        TRI_A  TRI_B  TRI_C  TRI_D    IU   SNS
## TRI_A  0.827      .      .      .     .     .
## TRI_B  0.475  0.760      .      .     .     .
## TRI_C -0.335 -0.471  0.705      .     .     .
## TRI_D -0.441 -0.483  0.468  0.577     .     .
## IU     0.602  0.415 -0.347 -0.388 0.910     .
## SNS    0.531  0.547 -0.372 -0.453 0.612 0.769
##
## FL Criteria table reports square root of AVE on the diagonal and construct correlations on the lower tria
ngle.
```

Hide

```
summary_m_1$validity$htmt
```

```
##       TRI_A TRI_B TRI_C TRI_D    IU SNS
## TRI_A     .     .     .     .     .   .
## TRI_B 0.489     .     .     .     .   .
## TRI_C 0.303 0.610     .     .     .   .
## TRI_D 0.406 0.550 0.558     .     .   .
## IU    0.707 0.481 0.375 0.370     .   .
## SNS   0.636 0.655 0.460 0.485 0.777   .
```

Hide

```
summary_m_1$validity$vif_items
```

```
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## TRI_C :
##  TRI9 TRI10 TRI11 TRI12
## 1.402 1.733 1.767 2.054
##
## TRI_D :
## TRI13 TRI14 TRI15 TRI16
## 1.239 1.663 1.690 1.375
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
```

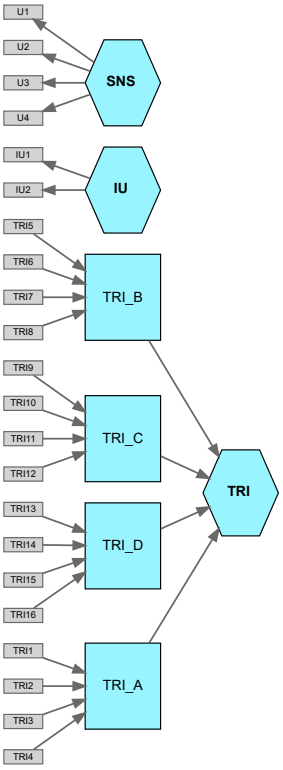
# K.2. Constructo de orden superior

## K.2.1. Modelo de medida

*a. Modelo de medida Formativo*

Hide

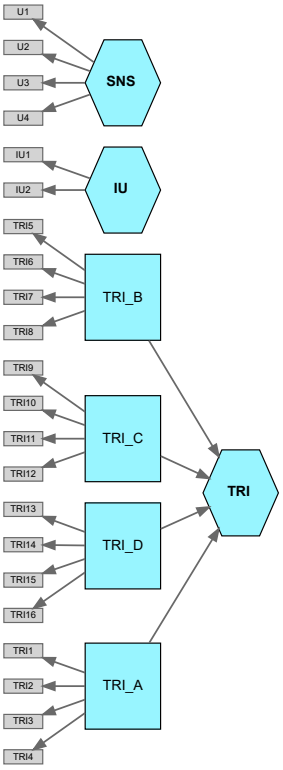
```
m_medida_3 <- constructs(
  composite('TRI_A', multi_items('TRI', 1:4), weights = mode_B),
  composite('TRI_B', multi_items('TRI', 5:8), weights = mode_B),
  composite('TRI_C', multi_items('TRI', 9:12), weights = mode_B),
  composite('TRI_D', multi_items('TRI', 13:16), weights = mode_B),
  higher_composite('TRI', c('TRI_A', 'TRI_B', 'TRI_C', 'TRI_D'), method ='two stage', weights = mode_B),
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4))
 )
plot(m_medida_3)
```




*b. Modelo de medida Reflectivo*

Hide

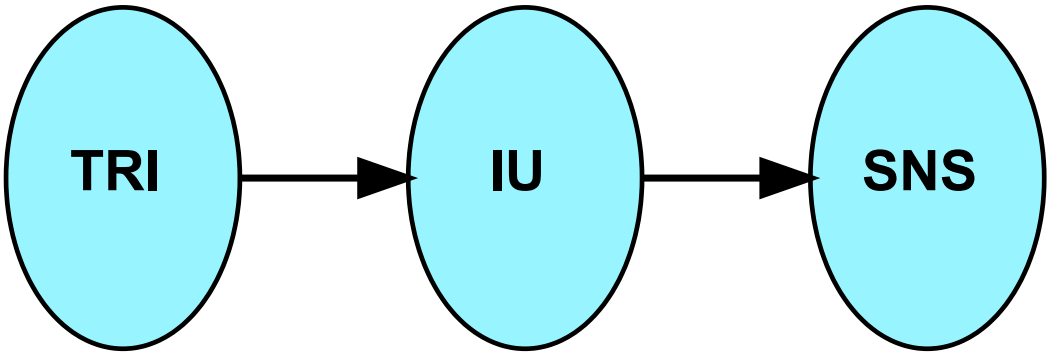
```
m_medida_4 <- constructs(
  composite('TRI_A', multi_items('TRI', 1:4)),
  composite('TRI_B', multi_items('TRI', 5:8)),
  composite('TRI_C', multi_items('TRI', 9:12)),
  composite('TRI_D', multi_items('TRI', 13:16)),
  higher_composite('TRI', c('TRI_A', 'TRI_B', 'TRI_C', 'TRI_D'), method ='two stage', weights = mode_B),
  composite('IU', multi_items('IU', 1:2)),
  composite('SNS', multi_items('U', 1:4))
 )
plot(m_medida_4)
```




## K.2.2. Modelo estructural

Hide

```
m_estruc_2 <- relationships(
  paths(from = 'TRI', to = 'IU'),
  paths(from = c("IU"), to = c('SNS')))
plot(m_estruc_2)
```

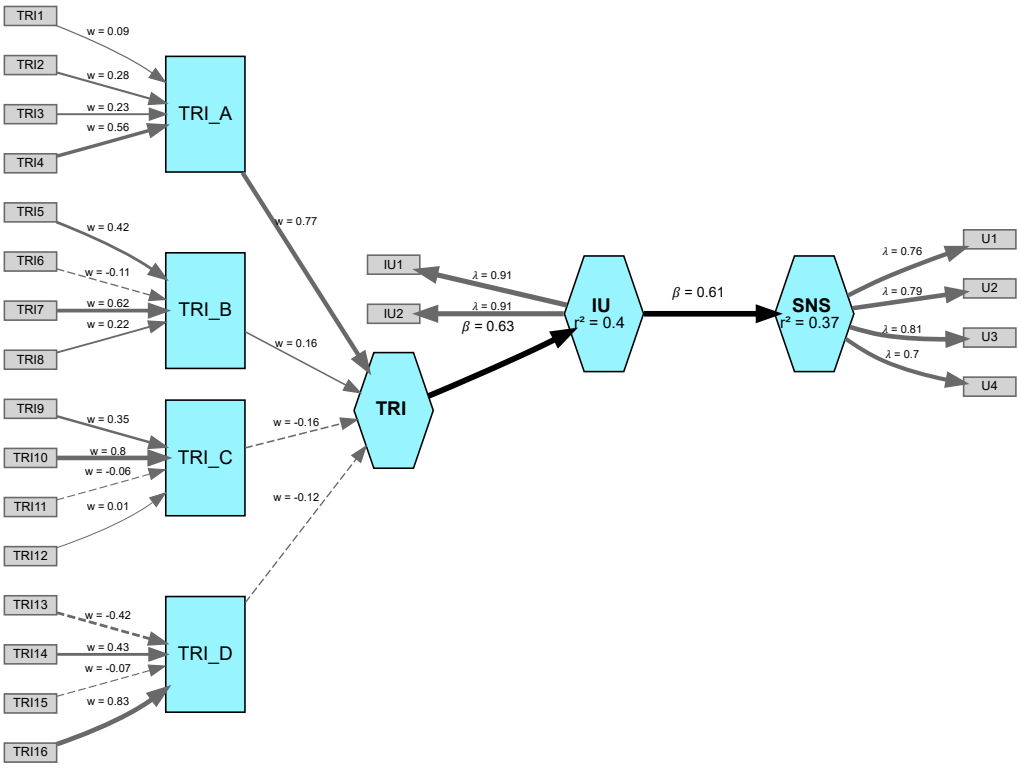


# K.2.3. Estimación modelo

*a. Estimación modelo Formativo*

Hide

```
estimacion_model_2 <- estimate_pls(data = pls_data2,
                                   measurement_model = m_medida_3,  #K.2.1. a
                                   structural_model = m_estruc_2,   # K.2.2.
                                   inner_weights = path_weighting,
                                   # path_weighting para path weighting (default) o path_factorial para f
        actor weighting,
                                   missing = mean_replacement, #Reemplazar los valores perdidos por defec
        to mean
                                   missing_value = '-99' ) #indicador de valores perdidos

plot(estimacion_model_2)
```



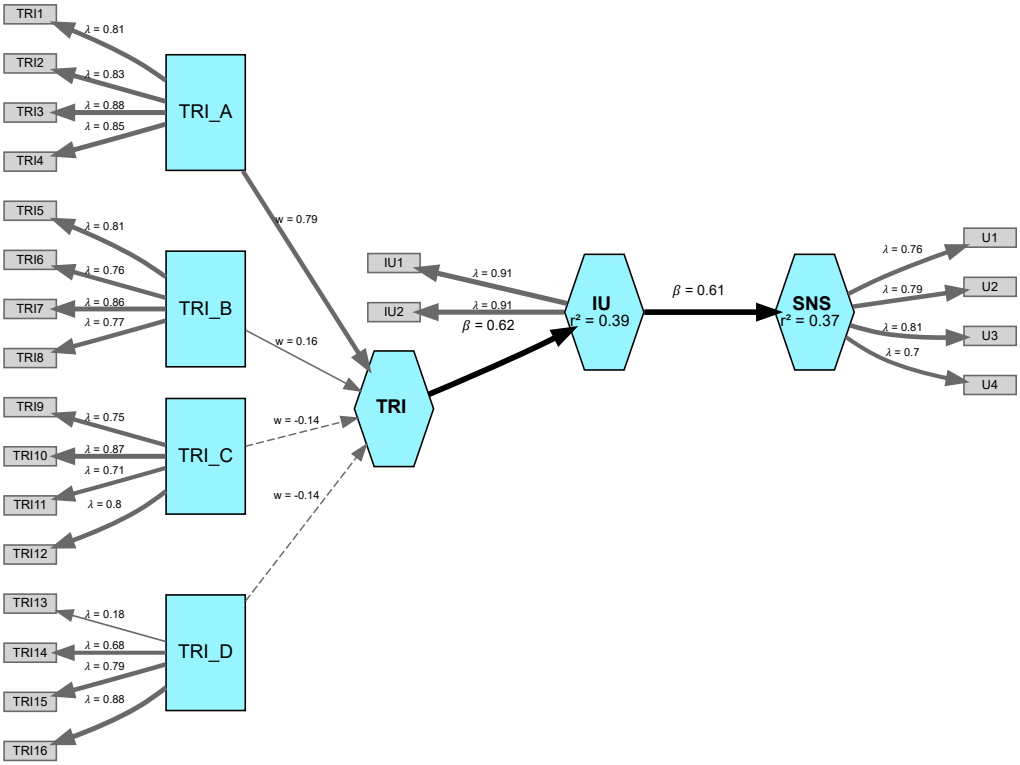

Hide

```
summary_m_2 = summary(estimacion_model_2)
```

*b. Estimación modelo Reflectivo*

Hide

```
estimacion_model_3 <- estimate_pls(data = pls_data2,
                                   measurement_model = m_medida_4,  #K.2.1. b
                                   structural_model = m_estruc_2,   # K.2.2.
                                   inner_weights = path_weighting,
                                   # path_weighting para path weighting (default) o path_factorial para f
        actor weighting,
                                   missing = mean_replacement, #Reemplazar los valores perdidos por defec
        to mean
                                   missing_value = '-99' ) #indicador de valores perdidos

plot(estimacion_model_3)
```



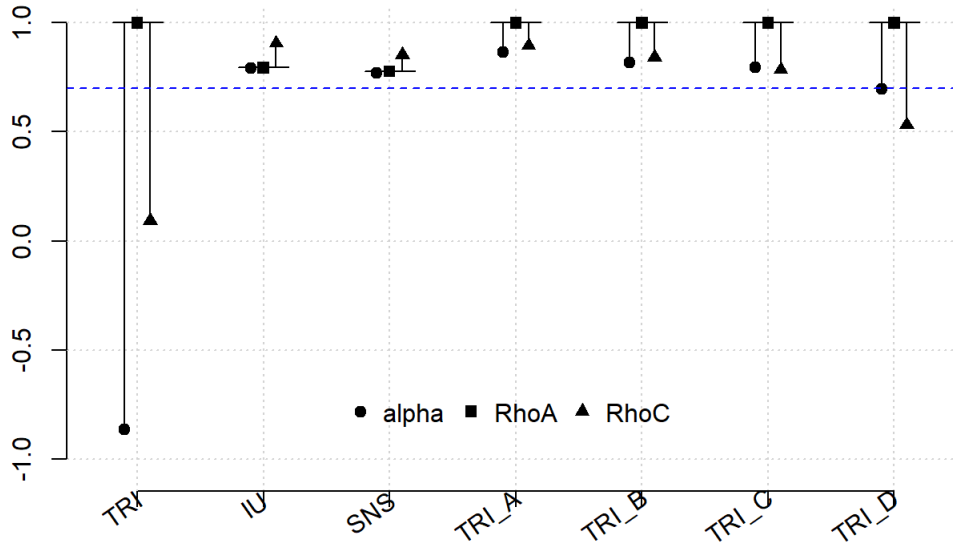

Hide

```
summary_m_3 = summary(estimacion_model_3)
```

## K.2.4. Evaluación modelo de 2do orden Formativo

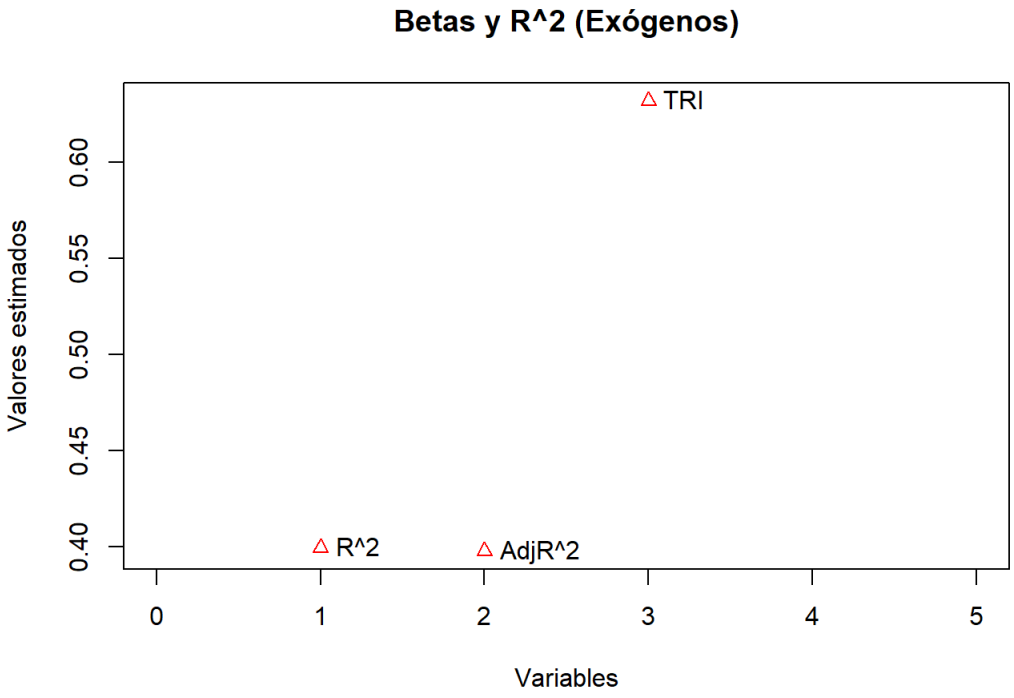
Hide

```
plot(summary_m_2$reliability, title =  "Fig. : Fiabilidad orden inferior")
```



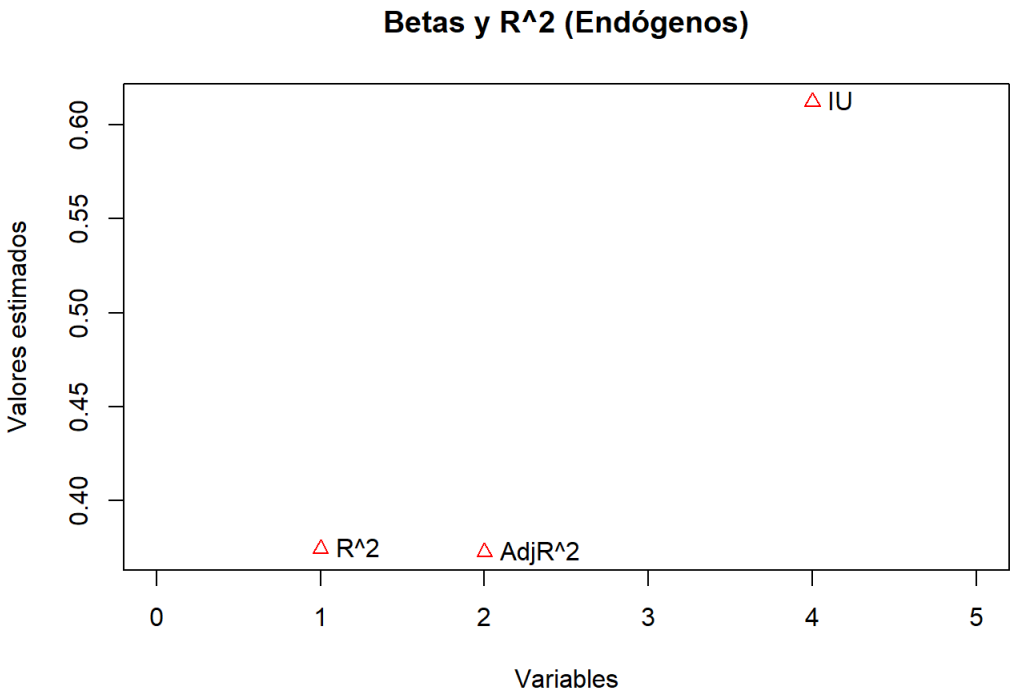

Hide

```
plot(summary_m_2$paths[,1], pch = 2, col = "red", main="Betas y R^2 (Exógenos)",
     xlab = "Variables", ylab = "Valores estimados", xlim = c(0,length(row.names(summary_m_2$paths))+1)
     )
text(summary_m_2$paths[,1],labels = row.names(summary_m_2$paths) , pos = 4)
```




Hide

```
plot(summary_m_2$paths[,2], pch = 2, col = "red", main="Betas y R^2 (Endógenos)",
     xlab = "Variables", ylab = "Valores estimados" , xlim = c(0,length(row.names(summary_m_2$paths))+1) )
text(summary_m_2$paths[,2],labels = row.names(summary_m_2$paths) , pos = 4)
```




Hide

```
summary_m_2$reliability
```

```
##        alpha  rhoC   AVE  rhoA
## TRI   -0.864 0.092 0.504 1.000
## IU     0.794 0.906 0.829 0.794
## SNS    0.769 0.852 0.591 0.778
## TRI_A  0.866 0.895 0.683 1.000
## TRI_B  0.816 0.841 0.577 1.000
## TRI_C  0.797 0.786 0.497 1.000
## TRI_D  0.695 0.532 0.333 1.000
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5
```

Hide

```
summary_m_2$loading
```

```
##          TRI     IU    SNS  TRI_A  TRI_B  TRI_C  TRI_D
## TRI_A  0.953  0.000  0.000  0.000  0.000  0.000  0.000
## TRI_B  0.657  0.000  0.000  0.000  0.000  0.000  0.000
## TRI_C -0.549 -0.000 -0.000  0.000  0.000  0.000  0.000
## TRI_D -0.613 -0.000 -0.000  0.000  0.000  0.000  0.000
## IU1    0.000  0.907  0.000  0.000  0.000  0.000  0.000
## IU2    0.000  0.913  0.000  0.000  0.000  0.000  0.000
## U1     0.000  0.000  0.763  0.000  0.000  0.000  0.000
## U2     0.000  0.000  0.793  0.000  0.000  0.000  0.000
## U3     0.000  0.000  0.814  0.000  0.000  0.000  0.000
## U4     0.000  0.000  0.700  0.000  0.000  0.000  0.000
## TRI1   0.000  0.000  0.000  0.718  0.000 -0.000 -0.000
## TRI2   0.000  0.000  0.000  0.794  0.000 -0.000 -0.000
## TRI3   0.000  0.000  0.000  0.858  0.000 -0.000 -0.000
## TRI4   0.000  0.000  0.000  0.923  0.000 -0.000 -0.000
## TRI5   0.000  0.000  0.000  0.000  0.792 -0.000 -0.000
## TRI6   0.000  0.000  0.000  0.000  0.562 -0.000 -0.000
## TRI7   0.000  0.000  0.000  0.000  0.926 -0.000 -0.000
## TRI8   0.000  0.000  0.000  0.000  0.713 -0.000 -0.000
## TRI9   0.000  0.000  0.000 -0.000 -0.000  0.736  0.000
## TRI10  0.000  0.000  0.000 -0.000 -0.000  0.955  0.000
## TRI11  0.000  0.000  0.000 -0.000 -0.000  0.449  0.000
## TRI12  0.000  0.000  0.000 -0.000 -0.000  0.576  0.000
## TRI13  0.000  0.000  0.000  0.000  0.000  0.000 -0.157
## TRI14  0.000  0.000  0.000 -0.000 -0.000  0.000  0.499
## TRI15  0.000  0.000  0.000 -0.000 -0.000  0.000  0.494
## TRI16  0.000  0.000  0.000 -0.000 -0.000  0.000  0.904
```

Hide

```
summary_m_2$validity$fl_criteria
```

```
##          TRI     IU    SNS  TRI_A  TRI_B TRI_C TRI_D
## TRI    0.710      .      .      .      .     .     .
## IU     0.632  0.910      .      .      .     .     .
## SNS    0.610  0.612  0.769      .      .     .     .
## TRI_A  0.953  0.602  0.531  0.827      .     .     .
## TRI_B  0.657  0.415  0.547  0.475  0.760     .     .
## TRI_C -0.549 -0.347 -0.372 -0.335 -0.471 0.705     .
## TRI_D -0.613 -0.388 -0.453 -0.441 -0.483 0.468 0.577
##
## FL Criteria table reports square root of AVE on the diagonal and construct correlations on the lower tria
ngle.
```

Hide

```
summary_m_2$validity$htmt
```

```
##         TRI    IU   SNS TRI_A TRI_B TRI_C TRI_D
## TRI       .     .     .     .     .     .     .
## IU    0.736     .     .     .     .     .     .
## SNS   0.816 0.777     .     .     .     .     .
## TRI_A 0.870 0.707 0.636     .     .     .     .
## TRI_B 0.943 0.481 0.655 0.489     .     .     .
## TRI_C 0.899 0.375 0.460 0.303 0.610     .     .
## TRI_D 0.796 0.370 0.485 0.406 0.550 0.558     .
```

Hide

```
summary_m_2$validity$vif_items
```

```
## TRI :
## TRI_A TRI_B TRI_C TRI_D
## 1.402 1.593 1.429 1.536
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
##
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## TRI_C :
##  TRI9 TRI10 TRI11 TRI12
## 1.402 1.733 1.767 2.054
##
## TRI_D :
## TRI13 TRI14 TRI15 TRI16
## 1.239 1.663 1.690 1.375
```

Hide

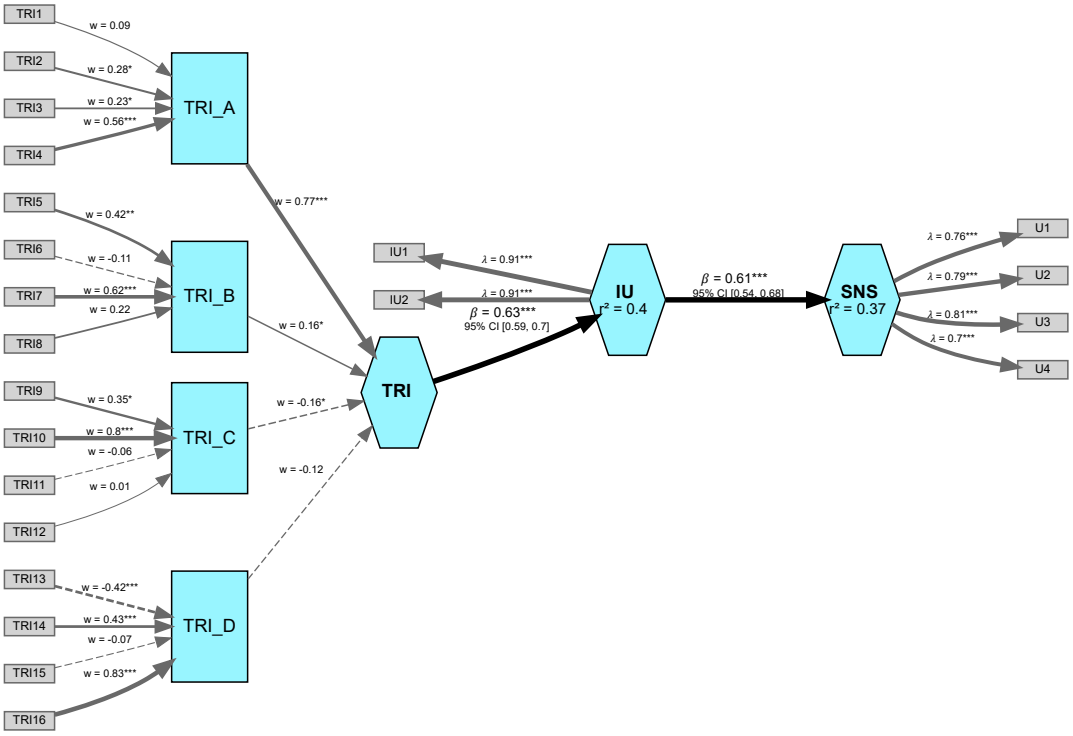
```
summary_m_2$validity$cross_loadings
```

```
##          TRI     IU    SNS  TRI_A  TRI_B  TRI_C  TRI_D
## TRI1   0.665  0.432  0.373  0.718  0.304 -0.190 -0.282
## TRI2   0.758  0.478  0.462  0.794  0.355 -0.298 -0.354
## TRI3   0.813  0.517  0.467  0.858  0.357 -0.320 -0.363
## TRI4   0.883  0.555  0.467  0.923  0.476 -0.288 -0.417
## TRI5   0.474  0.329  0.435  0.339  0.792 -0.309 -0.319
## TRI6   0.361  0.234  0.282  0.254  0.562 -0.247 -0.302
## TRI7   0.624  0.385  0.483  0.449  0.926 -0.457 -0.480
## TRI8   0.511  0.296  0.441  0.375  0.713 -0.393 -0.389
## TRI9  -0.366 -0.255 -0.306 -0.168 -0.425  0.736  0.426
## TRI10 -0.542 -0.331 -0.344 -0.351 -0.430  0.955  0.418
## TRI11 -0.271 -0.156 -0.212 -0.125 -0.413  0.449  0.314
## TRI12 -0.356 -0.200 -0.267 -0.203 -0.374  0.576  0.398
## TRI13  0.205  0.061  0.021  0.244  0.014  0.028 -0.157
## TRI14 -0.207 -0.193 -0.228 -0.106 -0.229  0.176  0.499
## TRI15 -0.339 -0.191 -0.312 -0.220 -0.368  0.319  0.494
## TRI16 -0.553 -0.350 -0.441 -0.369 -0.483  0.510  0.904
## TRI_A  0.953  0.602  0.531  1.000  0.475 -0.335 -0.441
## TRI_B  0.657  0.415  0.547  0.475  1.000 -0.471 -0.483
## TRI_C -0.549 -0.347 -0.372 -0.335 -0.471  1.000  0.468
## TRI_D -0.613 -0.388 -0.453 -0.441 -0.483  0.468  1.000
## IU1    0.538  0.907  0.578  0.505  0.367 -0.310 -0.335
## IU2    0.612  0.913  0.537  0.589  0.389 -0.321 -0.371
## U1     0.463  0.482  0.763  0.417  0.381 -0.248 -0.343
## U2     0.468  0.522  0.793  0.400  0.444 -0.315 -0.329
## U3     0.533  0.482  0.814  0.494  0.393 -0.300 -0.345
## U4     0.407  0.381  0.700  0.308  0.482 -0.282 -0.395
```

## K.2.5. Bootstrap modelo de 2do orden Formativo

Hide

```
boot_m_2 <- bootstrap_model(seminr_model = estimacion_model_2 , #K.2.3. a
                nboot = 500,  ### N° Subsamples  5000<
                cores = parallel::detectCores(),                    #CPU cores -parallel processing
                seed = 123)    #Fijar la semilla
plot(boot_m_2)
```




Hide

```
sum_boot_m_2 <- summary(boot_m_2, alpha=0.05 )  ### Intervalo de confianza, en este caso es dos colas 90%
```

## K.2.6. Evaluación Bootstrap modelo de 2do orden Formativo

Hide

```
sum_boot_m_2$bootstrapped_weights
```

```
##                 Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI
## TRI_A  ->  TRI           0.770          0.758        0.060  12.829   0.631
## TRI_B  ->  TRI           0.158          0.166        0.071   2.231   0.026
## TRI_C  ->  TRI          -0.159         -0.163        0.071  -2.223  -0.296
## TRI_D  ->  TRI          -0.123         -0.122        0.080  -1.549  -0.281
## IU1  ->  IU              0.540          0.541        0.014  39.603   0.515
## IU2  ->  IU              0.558          0.558        0.013  41.846   0.534
## U1  ->  SNS              0.335          0.334        0.021  15.915   0.293
## U2  ->  SNS              0.362          0.362        0.023  15.697   0.320
## U3  ->  SNS              0.334          0.334        0.024  14.196   0.289
## U4  ->  SNS              0.264          0.265        0.019  13.847   0.226
## TRI1  ->  TRI_A          0.093          0.092        0.128   0.723  -0.148
## TRI2  ->  TRI_A          0.277          0.276        0.119   2.322   0.036
## TRI3  ->  TRI_A          0.232          0.244        0.119   1.949   0.000
## TRI4  ->  TRI_A          0.558          0.536        0.112   4.970   0.324
## TRI5  ->  TRI_B          0.423          0.412        0.142   2.971   0.103
## TRI6  ->  TRI_B         -0.113         -0.115        0.134  -0.843  -0.352
## TRI7  ->  TRI_B          0.621          0.606        0.142   4.364   0.322
## TRI8  ->  TRI_B          0.216          0.221        0.168   1.284  -0.096
## TRI9  ->  TRI_C          0.348          0.323        0.166   2.100   0.001
## TRI10  ->  TRI_C         0.800          0.792        0.151   5.316   0.461
## TRI11  ->  TRI_C        -0.064         -0.075        0.182  -0.351  -0.375
## TRI12  ->  TRI_C         0.014          0.018        0.185   0.076  -0.344
## TRI13  ->  TRI_D        -0.419         -0.423        0.125  -3.355  -0.658
## TRI14  ->  TRI_D         0.425          0.426        0.131   3.248   0.182
## TRI15  ->  TRI_D        -0.065         -0.059        0.131  -0.495  -0.307
## TRI16  ->  TRI_D         0.835          0.806        0.102   8.170   0.588
##                 97.5% CI
## TRI_A  ->  TRI      0.858
## TRI_B  ->  TRI      0.299
## TRI_C  ->  TRI     -0.010
## TRI_D  ->  TRI      0.033
## IU1  ->  IU         0.568
## IU2  ->  IU         0.585
## U1  ->  SNS         0.374
## U2  ->  SNS         0.410
## U3  ->  SNS         0.383
## U4  ->  SNS         0.299
## TRI1  ->  TRI_A     0.344
## TRI2  ->  TRI_A     0.487
## TRI3  ->  TRI_A     0.468
## TRI4  ->  TRI_A     0.761
## TRI5  ->  TRI_B     0.684
## TRI6  ->  TRI_B     0.133
## TRI7  ->  TRI_B     0.851
## TRI8  ->  TRI_B     0.525
## TRI9  ->  TRI_C     0.671
## TRI10  ->  TRI_C    1.048
## TRI11  ->  TRI_C    0.315
## TRI12  ->  TRI_C    0.355
## TRI13  ->  TRI_D   -0.166
## TRI14  ->  TRI_D    0.662
## TRI15  ->  TRI_D    0.187
## TRI16  ->  TRI_D    0.966
```

Hide

```
summary_m_2$validity$vif_items
```

```
## TRI :
## TRI_A TRI_B TRI_C TRI_D
## 1.402 1.593 1.429 1.536
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
##
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## TRI_C :
##  TRI9 TRI10 TRI11 TRI12
## 1.402 1.733 1.767 2.054
##
## TRI_D :
## TRI13 TRI14 TRI15 TRI16
## 1.239 1.663 1.690 1.375
```

Hide

```
sum_boot_m_2$bootstrapped_loadings
```

```
##                 Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI
## TRI_A  ->  TRI           0.953          0.945        0.022  42.457   0.893
## TRI_B  ->  TRI           0.657          0.653        0.055  11.912   0.548
## TRI_C  ->  TRI          -0.549         -0.554        0.065  -8.490  -0.672
## TRI_D  ->  TRI          -0.613         -0.611        0.059 -10.340  -0.722
## IU1  ->  IU              0.907          0.907        0.010  94.655   0.887
## IU2  ->  IU              0.913          0.913        0.009  98.926   0.894
## U1  ->  SNS              0.763          0.764        0.025  30.934   0.715
## U2  ->  SNS              0.793          0.793        0.021  37.472   0.751
## U3  ->  SNS              0.814          0.813        0.023  35.072   0.765
## U4  ->  SNS              0.700          0.701        0.031  22.364   0.636
## TRI1  ->  TRI_A          0.718          0.711        0.070  10.267   0.572
## TRI2  ->  TRI_A          0.794          0.789        0.056  14.077   0.671
## TRI3  ->  TRI_A          0.858          0.853        0.044  19.554   0.761
## TRI4  ->  TRI_A          0.923          0.909        0.036  25.666   0.836
## TRI5  ->  TRI_B          0.792          0.774        0.072  11.030   0.607
## TRI6  ->  TRI_B          0.562          0.551        0.093   6.026   0.363
## TRI7  ->  TRI_B          0.926          0.907        0.047  19.560   0.794
## TRI8  ->  TRI_B          0.713          0.705        0.097   7.377   0.481
## TRI9  ->  TRI_C          0.736          0.703        0.104   7.091   0.487
## TRI10  ->  TRI_C         0.955          0.930        0.051  18.890   0.799
## TRI11  ->  TRI_C         0.449          0.428        0.137   3.274   0.153
## TRI12  ->  TRI_C         0.576          0.557        0.117   4.909   0.316
## TRI13  ->  TRI_D        -0.157         -0.165        0.141  -1.112  -0.417
## TRI14  ->  TRI_D         0.499          0.491        0.114   4.387   0.251
## TRI15  ->  TRI_D         0.494          0.483        0.103   4.779   0.258
## TRI16  ->  TRI_D         0.904          0.880        0.054  16.591   0.756
##                 97.5% CI
## TRI_A  ->  TRI      0.978
## TRI_B  ->  TRI      0.759
## TRI_C  ->  TRI     -0.422
## TRI_D  ->  TRI     -0.492
## IU1  ->  IU         0.924
## IU2  ->  IU         0.930
## U1  ->  SNS         0.808
## U2  ->  SNS         0.833
## U3  ->  SNS         0.854
## U4  ->  SNS         0.756
## TRI1  ->  TRI_A     0.833
## TRI2  ->  TRI_A     0.882
## TRI3  ->  TRI_A     0.932
## TRI4  ->  TRI_A     0.973
## TRI5  ->  TRI_B     0.895
## TRI6  ->  TRI_B     0.730
## TRI7  ->  TRI_B     0.979
## TRI8  ->  TRI_B     0.859
## TRI9  ->  TRI_C     0.890
## TRI10  ->  TRI_C    0.992
## TRI11  ->  TRI_C    0.687
## TRI12  ->  TRI_C    0.761
## TRI13  ->  TRI_D    0.130
## TRI14  ->  TRI_D    0.705
## TRI15  ->  TRI_D    0.671
## TRI16  ->  TRI_D    0.963
```

*Significancia modelo segundo orden*

Hide

```
specific_effect_significance(boot_seminr_model = boot_m_2,
                             from = 'TRI',
                             through = 'IU',
                             to = 'SNS',
                             alpha = 0.05)
```

```
##  Original Est. Bootstrap Mean   Bootstrap SD        T Stat.        2.5% CI
##     0.38674111     0.39751212     0.03353316    11.53309594     0.32980491
##       97.5% CI
##     0.46136011
```

Hide

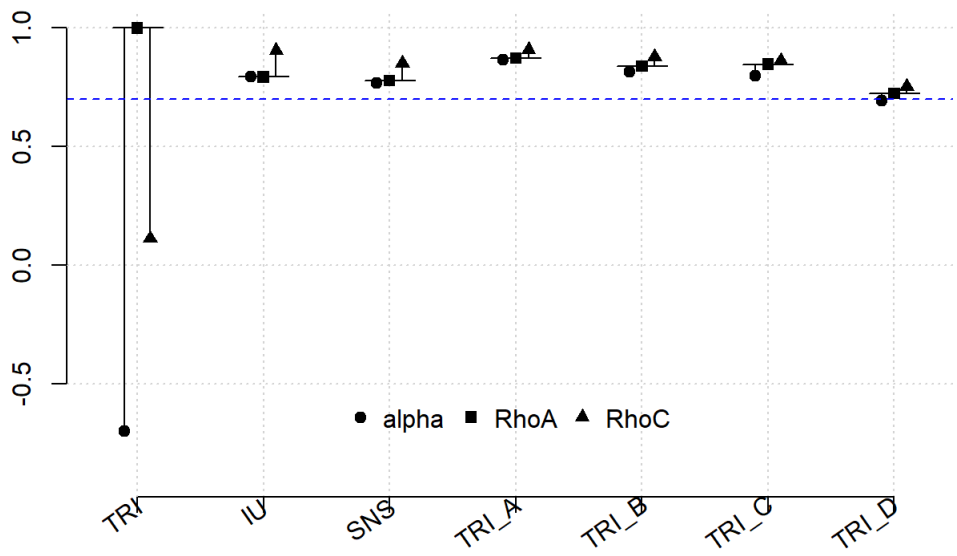
```
sum_boot_m_2$bootstrapped_paths
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## TRI  ->  IU         0.632          0.647        0.030  20.838   0.586    0.704
## IU  ->  SNS         0.612          0.614        0.034  17.943   0.544    0.682
```

## K.2.7. Evaluación modelo de 2do orden Reflectivo

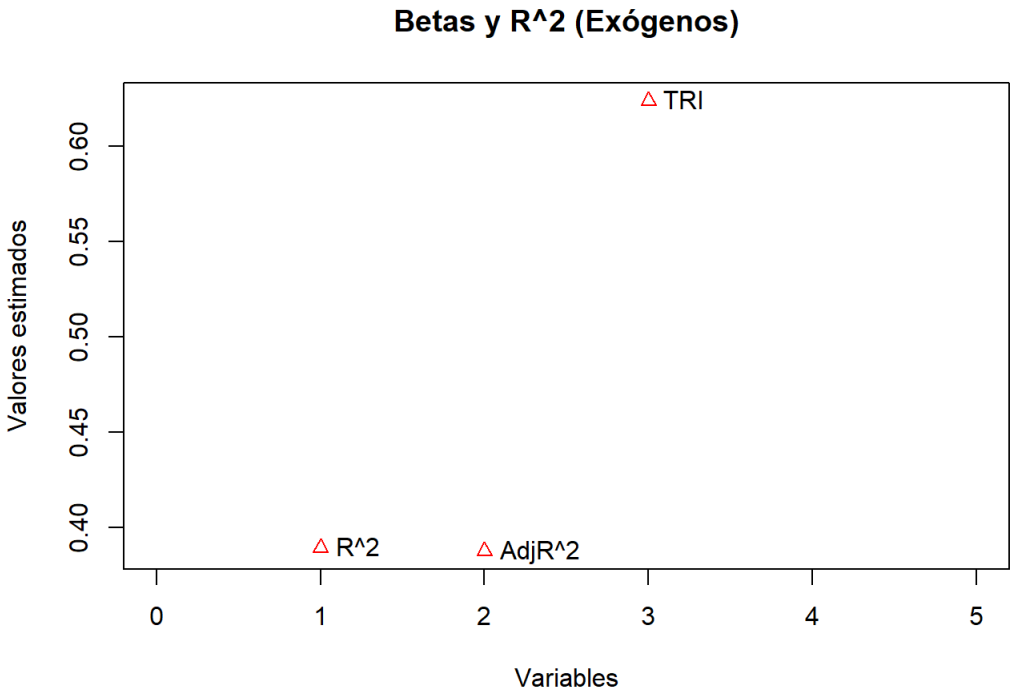
Hide

```
plot(summary_m_3$reliability, title =  "Fig. : Fiabilidad orden inferior")
```



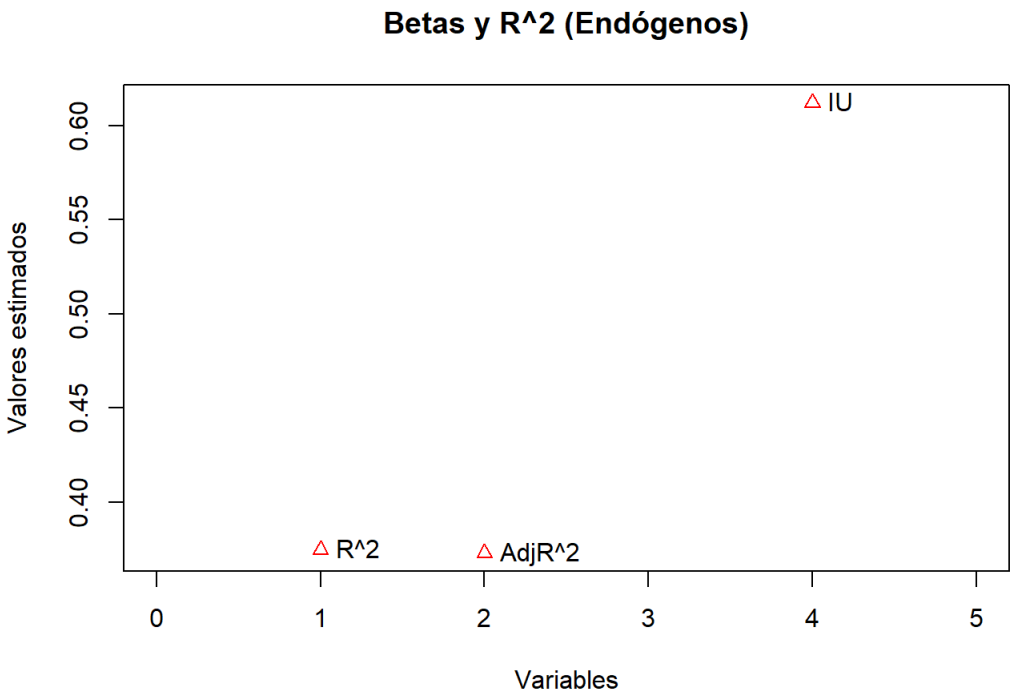

Hide

```
plot(summary_m_3$paths[,1], pch = 2, col = "red", main="Betas y R^2 (Exógenos)",
     xlab = "Variables", ylab = "Valores estimados", xlim = c(0,length(row.names(summary_m_3$paths))+1)
     )
text(summary_m_3$paths[,1],labels = row.names(summary_m_3$paths) , pos = 4)
```




Hide

```
plot(summary_m_3$paths[,2], pch = 2, col = "red", main="Betas y R^2 (Endógenos)",
     xlab = "Variables", ylab = "Valores estimados" , xlim = c(0,length(row.names(summary_m_3$paths))+1) )
text(summary_m_3$paths[,2],labels = row.names(summary_m_3$paths) , pos = 4)
```




Hide

```
summary_m_3$reliability
```

```
##        alpha  rhoC   AVE  rhoA
## TRI   -0.701 0.112 0.468 1.000
## IU     0.794 0.907 0.829 0.794
## SNS    0.769 0.852 0.591 0.778
## TRI_A  0.866 0.908 0.712 0.872
## TRI_B  0.816 0.878 0.643 0.838
## TRI_C  0.797 0.864 0.614 0.847
## TRI_D  0.695 0.753 0.474 0.725
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5
```

Hide

```
summary_m_3$loading
```

```
##          TRI     IU    SNS  TRI_A  TRI_B  TRI_C  TRI_D
## TRI_A  0.947  0.000  0.000  0.000  0.000  0.000  0.000
## TRI_B  0.636  0.000  0.000  0.000  0.000  0.000  0.000
## TRI_C -0.511 -0.000 -0.000  0.000  0.000  0.000  0.000
## TRI_D -0.554 -0.000 -0.000  0.000  0.000  0.000  0.000
## IU1    0.000  0.909  0.000  0.000  0.000  0.000  0.000
## IU2    0.000  0.912  0.000  0.000  0.000  0.000  0.000
## U1     0.000  0.000  0.763  0.000  0.000  0.000  0.000
## U2     0.000  0.000  0.793  0.000  0.000  0.000  0.000
## U3     0.000  0.000  0.814  0.000  0.000  0.000  0.000
## U4     0.000  0.000  0.700  0.000  0.000  0.000  0.000
## TRI1   0.000  0.000  0.000  0.812  0.000 -0.000 -0.000
## TRI2   0.000  0.000  0.000  0.833  0.000 -0.000 -0.000
## TRI3   0.000  0.000  0.000  0.884  0.000 -0.000 -0.000
## TRI4   0.000  0.000  0.000  0.845  0.000 -0.000 -0.000
## TRI5   0.000  0.000  0.000  0.000  0.805 -0.000 -0.000
## TRI6   0.000  0.000  0.000  0.000  0.765 -0.000 -0.000
## TRI7   0.000  0.000  0.000  0.000  0.862 -0.000 -0.000
## TRI8   0.000  0.000  0.000  0.000  0.771 -0.000 -0.000
## TRI9   0.000  0.000  0.000 -0.000 -0.000  0.753  0.000
## TRI10  0.000  0.000  0.000 -0.000 -0.000  0.866  0.000
## TRI11  0.000  0.000  0.000 -0.000 -0.000  0.713  0.000
## TRI12  0.000  0.000  0.000 -0.000 -0.000  0.796  0.000
## TRI13  0.000  0.000  0.000  0.000 -0.000  0.000  0.184
## TRI14  0.000  0.000  0.000 -0.000 -0.000  0.000  0.676
## TRI15  0.000  0.000  0.000 -0.000 -0.000  0.000  0.795
## TRI16  0.000  0.000  0.000 -0.000 -0.000  0.000  0.879
```

Hide

```
summary_m_3$validity$fl_criteria
```

```
##          TRI     IU    SNS  TRI_A  TRI_B TRI_C TRI_D
## TRI    0.684      .      .      .      .     .     .
## IU     0.624  0.910      .      .      .     .     .
## SNS    0.614  0.612  0.769      .      .     .     .
## TRI_A  0.947  0.591  0.527  0.844      .     .     .
## TRI_B  0.636  0.397  0.524  0.428  0.802     .     .
## TRI_C -0.511 -0.319 -0.372 -0.282 -0.502 0.784     .
## TRI_D -0.554 -0.346 -0.448 -0.337 -0.502 0.505 0.688
##
## FL Criteria table reports square root of AVE on the diagonal and construct correlations on the lower tria
ngle.
```

Hide

```
summary_m_3$validity$htmt
```

```
##         TRI    IU   SNS TRI_A TRI_B TRI_C TRI_D
## TRI       .     .     .     .     .     .     .
## IU    0.710     .     .     .     .     .     .
## SNS   0.821 0.777     .     .     .     .     .
## TRI_A 0.836 0.707 0.636     .     .     .     .
## TRI_B 1.018 0.481 0.655 0.489     .     .     .
## TRI_C 0.965 0.375 0.460 0.303 0.610     .     .
## TRI_D 0.922 0.370 0.485 0.406 0.550 0.558     .
```

Hide

```
summary_m_3$validity$vif_items
```

```
## TRI :
## TRI_A TRI_B TRI_C TRI_D
## 1.257 1.644 1.512 1.537
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
##
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## TRI_C :
##  TRI9 TRI10 TRI11 TRI12
## 1.402 1.733 1.767 2.054
##
## TRI_D :
## TRI13 TRI14 TRI15 TRI16
## 1.239 1.663 1.690 1.375
```

Hide

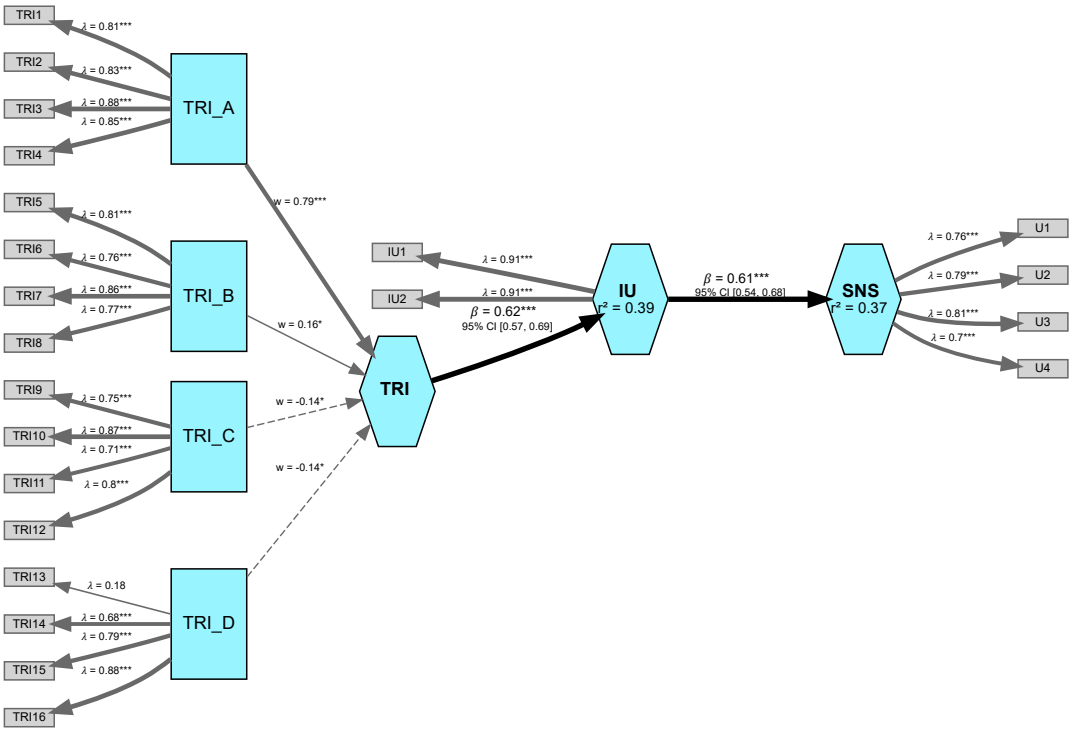
```
summary_m_3$validity$cross_loadings
```

```
##          TRI     IU    SNS  TRI_A  TRI_B  TRI_C  TRI_D
## TRI1   0.740  0.432  0.373  0.812  0.279 -0.164 -0.206
## TRI2   0.786  0.478  0.462  0.833  0.332 -0.227 -0.296
## TRI3   0.836  0.516  0.467  0.884  0.365 -0.275 -0.276
## TRI4   0.827  0.555  0.467  0.845  0.450 -0.271 -0.344
## TRI5   0.471  0.329  0.435  0.310  0.805 -0.353 -0.353
## TRI6   0.396  0.234  0.282  0.231  0.765 -0.295 -0.369
## TRI7   0.606  0.385  0.483  0.420  0.862 -0.506 -0.482
## TRI8   0.534  0.296  0.441  0.379  0.771 -0.420 -0.390
## TRI9  -0.360 -0.255 -0.306 -0.162 -0.405  0.753  0.460
## TRI10 -0.515 -0.331 -0.344 -0.345 -0.412  0.866  0.405
## TRI11 -0.294 -0.156 -0.212 -0.108 -0.414  0.713  0.320
## TRI12 -0.374 -0.200 -0.267 -0.194 -0.359  0.796  0.384
## TRI13  0.174  0.061  0.021  0.273 -0.037  0.083  0.184
## TRI14 -0.231 -0.193 -0.228 -0.081 -0.240  0.253  0.676
## TRI15 -0.386 -0.191 -0.312 -0.216 -0.385  0.316  0.795
## TRI16 -0.549 -0.350 -0.441 -0.346 -0.494  0.538  0.879
## TRI_A  0.947  0.591  0.527  1.000  0.428 -0.282 -0.337
## TRI_B  0.636  0.397  0.524  0.428  1.000 -0.502 -0.502
## TRI_C -0.511 -0.319 -0.372 -0.282 -0.502  1.000  0.505
## TRI_D -0.554 -0.346 -0.448 -0.337 -0.502  0.505  1.000
## IU1    0.538  0.909  0.578  0.506  0.353 -0.284 -0.298
## IU2    0.598  0.912  0.537  0.570  0.370 -0.297 -0.332
## U1     0.453  0.482  0.763  0.407  0.362 -0.225 -0.300
## U2     0.475  0.522  0.793  0.400  0.419 -0.316 -0.344
## U3     0.540  0.482  0.814  0.493  0.364 -0.305 -0.354
## U4     0.416  0.381  0.700  0.305  0.489 -0.303 -0.395
```

## K.2.8. Bootstrap modelo de 2do orden Reflectivo

Hide

```
boot_m_3 <- bootstrap_model(seminr_model = estimacion_model_3 , #K.2.3. b
               nboot = 500,  ### N° Subsamples  5000<
               cores = parallel::detectCores(),                    #CPU cores -parallel processing
               seed = 123)
plot(boot_m_3)
```




Hide

```
sum_boot_m_3 <- summary(boot_m_3, alpha=0.05 )  ### Intervalo de confianza, en este caso es dos colas 90%
```

## K.2.9. Evaluación Bootstrap modelo de 2do orden Reflectivo

Hide

```
sum_boot_m_3$bootstrapped_weights
```

```
##                 Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI
## TRI_A  ->  TRI           0.794          0.790        0.051  15.435   0.688
## TRI_B  ->  TRI           0.157          0.154        0.075   2.083   0.004
## TRI_C  ->  TRI          -0.139         -0.138        0.079  -1.758  -0.283
## TRI_D  ->  TRI          -0.138         -0.136        0.083  -1.649  -0.299
## IU1  ->  IU              0.544          0.544        0.014  39.431   0.518
## IU2  ->  IU              0.554          0.555        0.013  41.414   0.530
## U1  ->  SNS              0.335          0.334        0.021  15.928   0.293
## U2  ->  SNS              0.362          0.362        0.023  15.701   0.320
## U3  ->  SNS              0.334          0.334        0.024  14.185   0.289
## U4  ->  SNS              0.264          0.264        0.019  13.845   0.226
## TRI1  ->  TRI_A          0.258          0.258        0.020  12.672   0.221
## TRI2  ->  TRI_A          0.285          0.286        0.018  16.086   0.251
## TRI3  ->  TRI_A          0.308          0.309        0.016  19.294   0.279
## TRI4  ->  TRI_A          0.332          0.330        0.020  16.736   0.292
## TRI5  ->  TRI_B          0.328          0.326        0.030  10.901   0.262
## TRI6  ->  TRI_B          0.233          0.231        0.031   7.441   0.169
## TRI7  ->  TRI_B          0.383          0.383        0.028  13.593   0.332
## TRI8  ->  TRI_B          0.295          0.297        0.039   7.575   0.218
## TRI9  ->  TRI_C          0.341          0.338        0.053   6.402   0.238
## TRI10  ->  TRI_C         0.442          0.449        0.049   9.066   0.358
## TRI11  ->  TRI_C         0.208          0.201        0.052   3.997   0.084
## TRI12  ->  TRI_C         0.267          0.265        0.040   6.650   0.181
## TRI13  ->  TRI_D        -0.105         -0.122        0.109  -0.958  -0.363
## TRI14  ->  TRI_D         0.334          0.330        0.056   5.925   0.212
## TRI15  ->  TRI_D         0.330          0.324        0.046   7.227   0.226
## TRI16  ->  TRI_D         0.604          0.603        0.067   8.957   0.479
##                 97.5% CI
## TRI_A  ->  TRI      0.885
## TRI_B  ->  TRI      0.299
## TRI_C  ->  TRI      0.026
## TRI_D  ->  TRI      0.033
## IU1  ->  IU         0.571
## IU2  ->  IU         0.584
## U1  ->  SNS         0.374
## U2  ->  SNS         0.410
## U3  ->  SNS         0.383
## U4  ->  SNS         0.299
## TRI1  ->  TRI_A     0.295
## TRI2  ->  TRI_A     0.318
## TRI3  ->  TRI_A     0.344
## TRI4  ->  TRI_A     0.371
## TRI5  ->  TRI_B     0.384
## TRI6  ->  TRI_B     0.287
## TRI7  ->  TRI_B     0.438
## TRI8  ->  TRI_B     0.370
## TRI9  ->  TRI_C     0.462
## TRI10  ->  TRI_C    0.551
## TRI11  ->  TRI_C    0.294
## TRI12  ->  TRI_C    0.337
## TRI13  ->  TRI_D    0.075
## TRI14  ->  TRI_D    0.430
## TRI15  ->  TRI_D    0.402
## TRI16  ->  TRI_D    0.746
```

Hide

```
summary_m_3$validity$vif_items
```

```
## TRI :
## TRI_A TRI_B TRI_C TRI_D
## 1.257 1.644 1.512 1.537
##
## IU :
##   IU1   IU2
## 1.764 1.764
##
## SNS :
##    U1    U2    U3    U4
## 1.459 1.501 1.708 1.422
##
## TRI_A :
##  TRI1  TRI2  TRI3  TRI4
## 1.967 2.076 2.563 2.065
##
## TRI_B :
##  TRI5  TRI6  TRI7  TRI8
## 1.785 1.746 1.930 1.636
##
## TRI_C :
##  TRI9 TRI10 TRI11 TRI12
## 1.402 1.733 1.767 2.054
##
## TRI_D :
## TRI13 TRI14 TRI15 TRI16
## 1.239 1.663 1.690 1.375
```

Hide

```
sum_boot_m_3$bootstrapped_loadings
```

```
##                 Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI
## TRI_A  ->  TRI           0.947          0.942        0.022  43.098   0.893
## TRI_B  ->  TRI           0.636          0.631        0.056  11.368   0.526
## TRI_C  ->  TRI          -0.511         -0.513        0.062  -8.297  -0.624
## TRI_D  ->  TRI          -0.554         -0.559        0.057  -9.752  -0.662
## IU1  ->  IU              0.909          0.908        0.009  96.057   0.889
## IU2  ->  IU              0.912          0.912        0.009  96.936   0.892
## U1  ->  SNS              0.763          0.764        0.025  30.964   0.715
## U2  ->  SNS              0.793          0.793        0.021  37.454   0.751
## U3  ->  SNS              0.814          0.813        0.023  35.054   0.765
## U4  ->  SNS              0.700          0.701        0.031  22.359   0.636
## TRI1  ->  TRI_A          0.812          0.811        0.025  31.984   0.754
## TRI2  ->  TRI_A          0.833          0.833        0.021  39.232   0.788
## TRI3  ->  TRI_A          0.884          0.884        0.021  43.093   0.836
## TRI4  ->  TRI_A          0.845          0.845        0.019  43.361   0.805
## TRI5  ->  TRI_B          0.805          0.804        0.027  29.470   0.740
## TRI6  ->  TRI_B          0.765          0.764        0.036  21.410   0.686
## TRI7  ->  TRI_B          0.862          0.862        0.016  53.413   0.828
## TRI8  ->  TRI_B          0.771          0.773        0.037  21.082   0.695
## TRI9  ->  TRI_C          0.753          0.748        0.044  16.962   0.654
## TRI10  ->  TRI_C         0.866          0.867        0.023  37.847   0.819
## TRI11  ->  TRI_C         0.713          0.706        0.056  12.645   0.564
## TRI12  ->  TRI_C         0.796          0.790        0.038  21.080   0.704
## TRI13  ->  TRI_D         0.184          0.158        0.146   1.261  -0.136
## TRI14  ->  TRI_D         0.676          0.660        0.084   8.040   0.446
## TRI15  ->  TRI_D         0.795          0.780        0.050  15.794   0.663
## TRI16  ->  TRI_D         0.879          0.873        0.029  30.111   0.811
##                 97.5% CI
## TRI_A  ->  TRI      0.977
## TRI_B  ->  TRI      0.731
## TRI_C  ->  TRI     -0.389
## TRI_D  ->  TRI     -0.443
## IU1  ->  IU         0.925
## IU2  ->  IU         0.928
## U1  ->  SNS         0.808
## U2  ->  SNS         0.833
## U3  ->  SNS         0.854
## U4  ->  SNS         0.756
## TRI1  ->  TRI_A     0.855
## TRI2  ->  TRI_A     0.868
## TRI3  ->  TRI_A     0.918
## TRI4  ->  TRI_A     0.879
## TRI5  ->  TRI_B     0.852
## TRI6  ->  TRI_B     0.823
## TRI7  ->  TRI_B     0.891
## TRI8  ->  TRI_B     0.831
## TRI9  ->  TRI_C     0.823
## TRI10  ->  TRI_C    0.908
## TRI11  ->  TRI_C    0.793
## TRI12  ->  TRI_C    0.851
## TRI13  ->  TRI_D    0.424
## TRI14  ->  TRI_D    0.787
## TRI15  ->  TRI_D    0.846
## TRI16  ->  TRI_D    0.926
```

*Significancia modelo segundo orden*

Hide

```
specific_effect_significance(boot_seminr_model = boot_m_3,
                             from = 'TRI',
                             through = 'IU',
                             to = 'SNS',
                             alpha = 0.05)
```

```
##  Original Est. Bootstrap Mean   Bootstrap SD        T Stat.        2.5% CI
##      0.3819306      0.3888027      0.0319770     11.9439172      0.3243270
##       97.5% CI
##      0.4515041
```

Hide

```
sum_boot_m_3$bootstrapped_paths
```

```
##             Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
## TRI  ->  IU         0.624          0.632        0.028  22.111   0.572    0.685
## IU  ->  SNS         0.612          0.614        0.034  17.943   0.544    0.682
```

# L. Análisis Pathmox

## L.1. Definición modelo

Definir modelo usando laavan syntax.

Hide

```
cSmodel <- "
# Structural model
SNS  ~ IU + FC + HA
IU  ~ FC + HA + SI + HM + PE + EE
#modelo de medida
PE =~ PE1 + PE2 + PE3 + PE4
EE =~ EE1 + EE2 + EE3
SI =~ SI1 + SI2 + SI3 + SI4
FC =~ FC1 + FC2 + FC3
HM =~ HM1 + HM2 + HM3
HA =~ HA1 + HA2 + HA3 + HA4 + HA5
IU =~ IU1 + IU2
SNS =~ U1 + U2+ U3 + U4
"
```

## L.2. Análisis con cSEM

Hide

```
est_model <- csem(.data = pls_data2, .model = cSmodel)
bootstrap<- csem(.data = pls_data2, .model = cSmodel, .resample_method = "bootstrap", .R = 1000)
#summarize(bootstrap)
#summarize(est_model)
#valides <- assess(est_model)
#infer(est_model)
#predict(est_model)
#verify(est_model)
```

# L.3. Configurando las variables

*Nota* Variables deben estar como factor y no deben contener puntos "."

Hide

```
pls_data2$GENDER2= as.factor(pls_data2$GENDER)
pls_data2$EDU2= as.factor(pls_data2$EDU)
pls_data2$RETIRED2= as.factor(pls_data2$RETIRED)
pls_data2$WSTATUS2= as.factor(pls_data2$WSTATUS)
pls_data2$GENERATION2= as.factor(pls_data2$GENERATION)
pls_data2$REGION2= as.factor(pls_data2$REGION)
pls_data2$EXP2= as.factor(pls_data2$EXP)
pls_data2$SOC2= as.factor(pls_data2$SOC)
```

En este caso creamos una variable de ejemplo en la cual se asigna a un tipo de TRI de acuerdo con el mayor valor presentado.

Hide

```
#Sumarizamos por TRI
pls_data2$TRI_A = pls_data2$TRI1 + pls_data2$TRI2 + pls_data2$TRI3 + pls_data2$TRI4
pls_data2$TRI_B = pls_data2$TRI5 + pls_data2$TRI6 + pls_data2$TRI7 + pls_data2$TRI8
pls_data2$TRI_C = pls_data2$TRI9 + pls_data2$TRI10 + pls_data2$TRI11 + pls_data2$TRI12
pls_data2$TRI_D = pls_data2$TRI13 + pls_data2$TRI14 + pls_data2$TRI15 + pls_data2$TRI16

#Asignamos a un tipo
pls_data2$TRI_T <- ifelse(pls_data2$TRI_B <= pls_data2$TRI_A & pls_data2$TRI_C <= pls_data2$TRI_A
        & pls_data2$TRI_D <= pls_data2$TRI_A, 1,
        ifelse (pls_data2$TRI_A <= pls_data2$TRI_B & pls_data2$TRI_C <= pls_data2$TRI_B
          & pls_data2$TRI_D <= pls_data2$TRI_B, 2,
          ifelse (pls_data2$TRI_A < pls_data2$TRI_C & pls_data2$TRI_B <= pls_data2$TRI_C
            & pls_data2$TRI_D <= pls_data2$TRI_C, 3, 4)))

#cambiamos el tipo a factor
pls_data2$TRI_T2= as.factor(pls_data2$TRI_T)
```

Genero grupo de categóricas

Hide

```
categoricas2 <- c( #"EXP2",
  "EDU2", "SOC2" , "WSTATUS2", "RETIRED2"   ,"GENDER2" ,
  "GENERATION2",  "REGION2",
  "TRI_T2")
```

Conjunto de datos con categóricas

Hide

```
CSIcatvar <- pls_data2[, categoricas2]
```

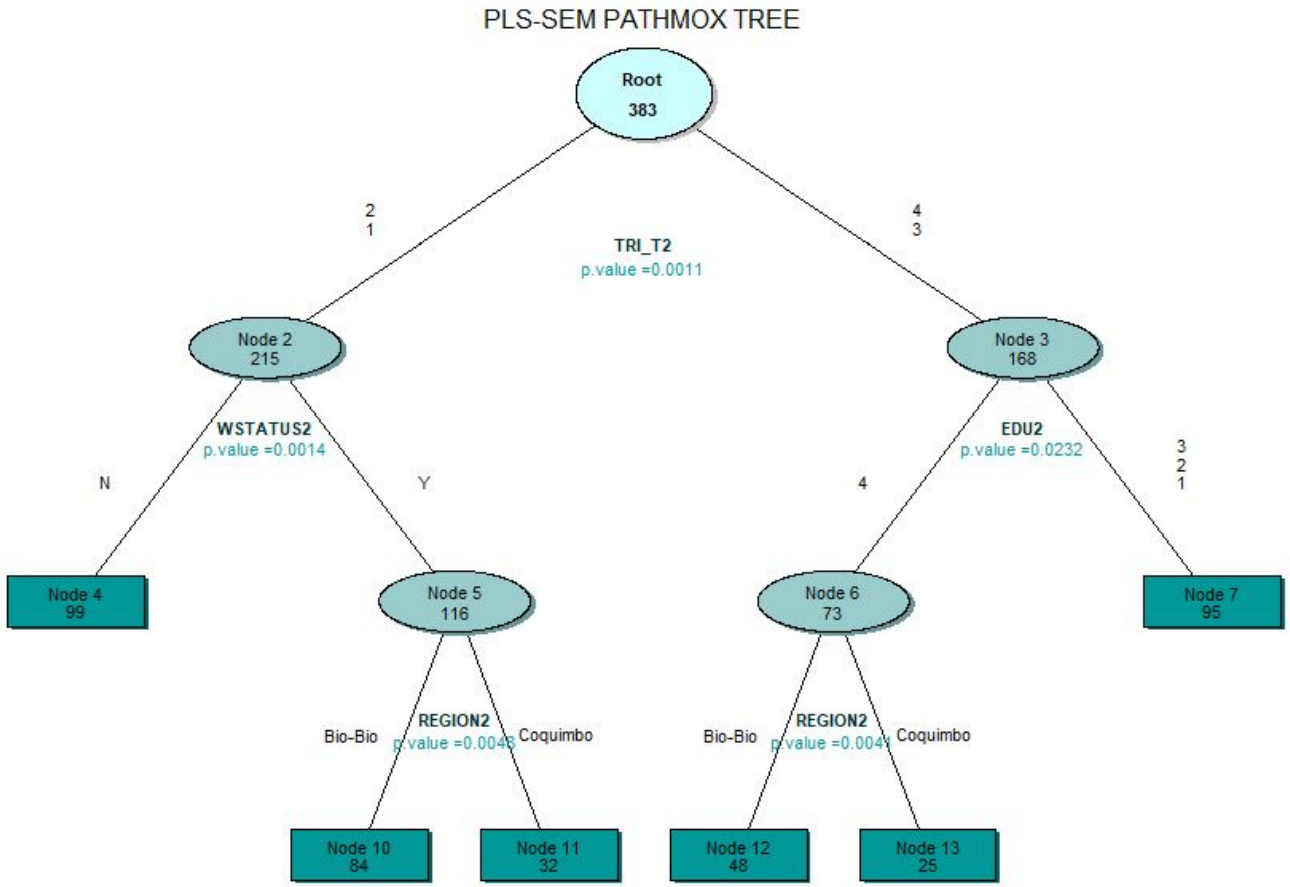
# L.4. Generación modelo y resultado

Ejecutar análisis Phatmox (ver Lamberti et al., 2016; 2017)

Hide

```
csi.pathmox = pls.pathmox(
  .model = cSmodel ,
  .data  = pls_data2,
  .catvar= CSIcatvar,  ## Variables categóricas a ser utilizadas
  # .scheme= 'centroid', 'factorial', 'path' defecto Tipo de esquema de ponderación interna
  .size = 0.10, #mínimo de observaciones en porcentaje
  .size_candidate = 15, #mínimo de observaciones en cantidad  por defecto es 50
  # .consistent = TRUE, #defecto es TRUE
  .alpha = 0.05,   ### umbral mínimo de importancia por defecto 0.05
  .deep = 8        ### Máxima profundidad del arbol
  )
```

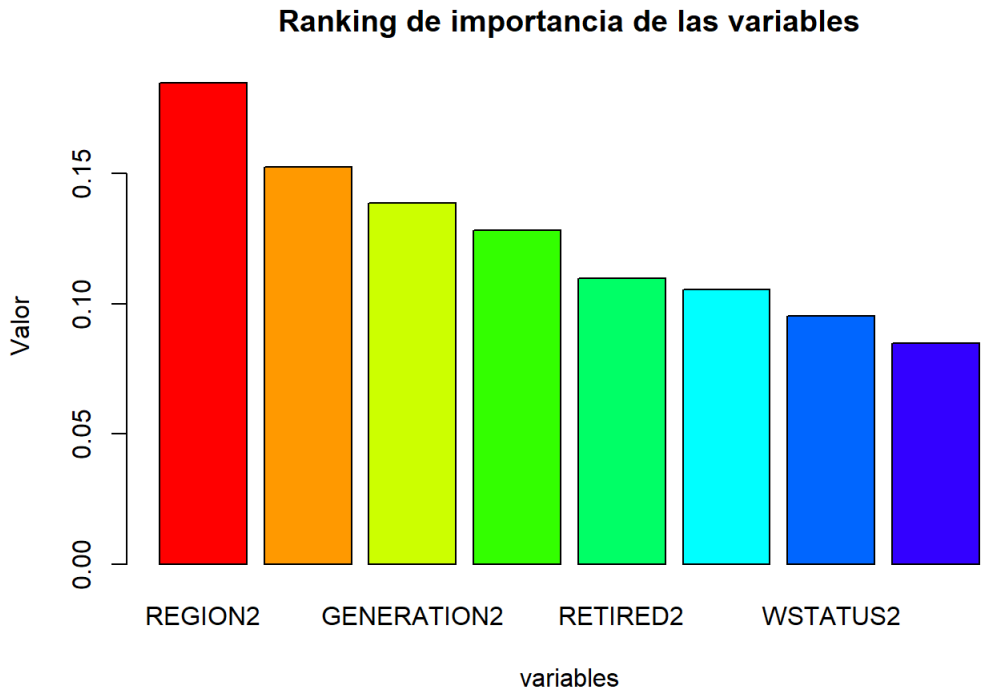
```
##
## PLS-SEM PATHMOX ANALYSIS
##
## ------------------------------------------
## Info parameters algorithm
##   parameters algorithm value
## 1    threshold signif.  0.05
## 2   node size limit(%)  0.10
## 3     tree depth level  8.00
##
## ------------------------------------------
## Info segmentation variables
##             nlevels ordered treatment
## EDU2              4   FALSE   nominal
## SOC2              5   FALSE   nominal
## WSTATUS2          2   FALSE    binary
## RETIRED2          2   FALSE    binary
## GENDER2           2   FALSE    binary
## GENERATION2       3   FALSE   nominal
## REGION2           2   FALSE    binary
## TRI_T2            4   FALSE   nominal
```




Ranking de importancia de las variables

Hide

```
variables <-csi.pathmox[["var_imp"]][["variable"]]
ranking <- csi.pathmox[["var_imp"]][["ranking"]]

barplot(ranking, main = "Ranking de importancia de las variables",
      xlab = "variables",
       ylab = "Valor",
       col = rainbow(10),
       names.arg = variables
      )
```




Resultados

Hide

```
summary(csi.pathmox)
```

```
##
## PLS-SEM PATHMOX ANALYSIS
##
## --------------------------------------------
## Info parameters algorithm:
##   parameters algorithm value
## 1     threshold signif  0.05
## 2   node size limit(%)  0.10
## 3     tree depth level  8.00
## --------------------------------------------
## Info tree:
##         parameters tree value
## 1             deep tree     3
## 2 number terminal nodes     6
## --------------------------------------------
## Info nodes:
##    node parent depth  type terminal size      % variable category
## 1     1      0     0  root       no  383 100.00     <NA>     <NA>
## 2     2      1     1  node       no  215  56.14   TRI_T2      1/2
## 3     3      1     1  node       no  168  43.86   TRI_T2      3/4
## 4     4      2     2 least      yes   99  25.85 WSTATUS2        N
## 5     5      2     2  node       no  116  30.29 WSTATUS2        Y
## 6     6      3     2  node       no   73  19.06     EDU2        4
## 7     7      3     2 least      yes   95  24.80     EDU2    1/2/3
## 8    10      5     3 least      yes   84  21.93  REGION2  Bio-Bio
## 9    11      5     3 least      yes   32   8.36  REGION2 Coquimbo
## 10   12      6     3 least      yes   48  12.53  REGION2  Bio-Bio
## 11   13      6     3 least      yes   25   6.53  REGION2 Coquimbo
## --------------------------------------------
## Info splits:
##
## Variable:
##   node variable  g1.mod   g2.mod
## 1    1   TRI_T2     1/2      3/4
## 2    2 WSTATUS2       N        Y
## 3    3     EDU2       4    1/2/3
## 4    5  REGION2 Bio-Bio Coquimbo
## 5    6  REGION2 Bio-Bio Coquimbo
##
## Info F-global test results (global differences):
##      node F value Pr(>F)
## [1,]    1  2.8561 0.0011 **
## [2,]    2  2.8227 0.0014 **
## [3,]    3  2.0571 0.0232 *
## [4,]    5  2.5468 0.0048 **
## [5,]    6  2.6742 0.0041 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Info F-coefficient test results (coefficent differences) :
##
## Node 1 :
##           F value Pr(>F)
## FC -> IU   0.7652 0.3820
## HA -> IU   9.6143 0.0020 **
## SI -> IU   0.3196 0.5720
## HM -> IU   0.9399 0.3326
## PE -> IU   2.1189 0.1459
## EE -> IU   1.9001 0.1685
## FC -> SNS  0.3249 0.5688
```

```
## HA -> SNS  1.0578 0.3041
## IU -> SNS  4.0827 0.0437 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Node 2 :
##           F value Pr(>F)
## FC -> IU   1.6655 0.1976
## HA -> IU   1.6850 0.1950
## SI -> IU   8.1413 0.0045 **
## HM -> IU   2.7871 0.0958 .
## PE -> IU   4.3042 0.0386 *
## EE -> IU   0.1956 0.6586
## FC -> SNS  1.4242 0.2334
## HA -> SNS  5.5389 0.0191 *
## IU -> SNS  0.6483 0.4212
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Node 3 :
##           F value Pr(>F)
## FC -> IU   5.1724 0.0236 *
## HA -> IU   1.5674 0.2115
## SI -> IU   0.2740 0.6010
## HM -> IU   3.3059 0.0700 .
## PE -> IU   4.5573 0.0336 *
## EE -> IU   2.2010 0.1389
## FC -> SNS  5.3666 0.0212 *
## HA -> SNS  1.9446 0.1642
## IU -> SNS  0.7523 0.3864
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Node 5 :
##           F value Pr(>F)
## FC -> IU   1.0129 0.3154
## HA -> IU   1.7484 0.1875
## SI -> IU   0.0173 0.8954
## HM -> IU   4.4443 0.0362 *
## PE -> IU   1.7347 0.1893
## EE -> IU   0.3251 0.5691
## FC -> SNS  0.0646 0.7996
## HA -> SNS  0.0412 0.8393
## IU -> SNS 14.9921 0.0001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Node 6 :
##           F value Pr(>F)
## FC -> IU   0.2793 0.5981
## HA -> IU   0.4374 0.5096
## SI -> IU   6.0320 0.0154 *
## HM -> IU   6.1007 0.0149 *
## PE -> IU   2.9845 0.0866 .
## EE -> IU   9.7832 0.0022 **
## FC -> SNS  1.5643 0.2134
## HA -> SNS  0.0001 0.9913
## IU -> SNS  0.1357 0.7132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## -----------------------------------------------
```

```
## Info variable importance ranking:
##      variable    ranking
## 4     REGION2 0.18493153
## 6        SOC2 0.15264481
## 3 GENERATION2 0.13868048
## 1        EDU2 0.12815501
## 5    RETIRED2 0.10978720
## 7      TRI_T2 0.10557265
## 8    WSTATUS2 0.09538325
## 2     GENDER2 0.08484506
##
## ------------------------------------------
## Info terminal nodes PLS-SEM models (path coeff. & R^2):
##          node 4  node 7 node 10 node 11 node 12 node 13
## FC->IU   0.0961 -0.0113  0.0246 -0.1171  0.2891  0.1823
## HA->IU  -0.0431  0.0370  0.1790  0.1676  0.4235  0.5707
## SI->IU   0.1656  0.3557 -0.1303  0.2338  0.3636  0.3064
## HM->IU   0.6981  0.6388  0.4671  0.2535  0.4998  0.3373
## PE->IU   0.4734  0.3033  0.1713  0.1733  0.2198  0.5503
## EE->IU   0.1068  0.2746  0.5072 -0.0048  0.1027 -0.2770
## FC->SNS  0.0698  0.1508  0.2720  0.5807  0.5361  0.0742
## HA->SNS -0.0553 -0.0641 -0.0752  0.0281 -0.3989  0.2895
## IU->SNS  0.2289  0.0542  0.1260  0.5494  0.0090  0.0227
## R^2 IU   0.5540  0.5474  0.5253  0.6699  0.7541  0.6425
## R^2 SNS  0.6710  0.4799  0.4476  0.6798  0.5936  0.5917
```