R Notebook

This is an [R Markdown](http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

library(ExPosition)

## Loading required package: prettyGraphs

library(InPosition)  
# use the last version of PTCA  
devtools::install\_github('HerveAbdi/PTCA4CATA')

## Skipping install of 'PTCA4CATA' from a github remote, the SHA1 (87581916) has not changed since last install.  
## Use `force = TRUE` to force installation

library(PTCA4CATA)

##   
## Attaching package: 'PTCA4CATA'

## The following object is masked from 'package:InPosition':  
##   
## boot.ratio.test

library(corrplot)

## corrplot 0.84 loaded

library(ggplot2)  
# install.packages('gplots')  
# also install data4PCCAR  
devtools::install\_github('HerveAbdi/data4PCCAR')

## Skipping install of 'data4PCCAR' from a github remote, the SHA1 (e52e1cb7) has not changed since last install.  
## Use `force = TRUE` to force installation

library(data4PCCAR)  
library(corrplot)

WE <- read.csv('WeeklyEarningsbyRace.csv', row.names=1)  
head(WE)

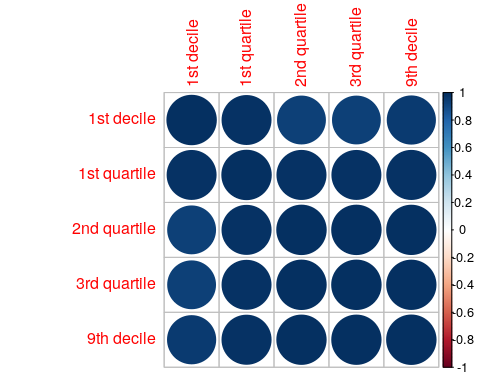
## White.men White.women Black.men Black.Women  
## 1st decile 412 374 361 331  
## 1st quartile 594 506 483 423  
## 2nd quartile 920 743 680 615  
## 3rd quartile 1467 1140 1046 935  
## 9th decile 2278 1726 1551 1453  
## Total people (in thousands) 48746 36698 6445 7142  
## Asian.Men Asian.Women Hispanic.Men  
## 1st decile 420 385 358  
## 1st quartile 648 551 451  
## 2nd quartile 1129 877 631  
## 3rd quartile 1860 1411 979  
## 9th decile 2699 2024 1498  
## Total people (in thousands) 3684 2954 11142  
## Hispanic.Women  
## 1st decile 320  
## 1st quartile 404  
## 2nd quartile 566  
## 3rd quartile 830  
## 9th decile 1266  
## Total people (in thousands) 7168

WE\_data <- t(WE[-6,])  
  
head(WE\_data)

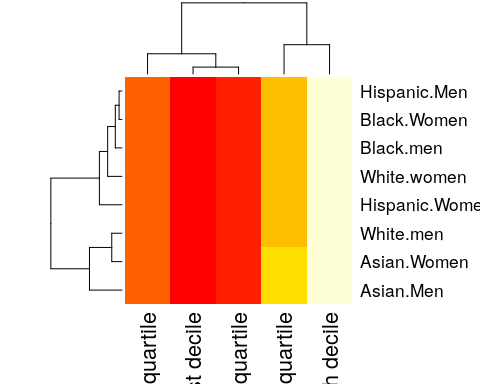
## 1st decile 1st quartile 2nd quartile 3rd quartile 9th decile  
## White.men 412 594 920 1467 2278  
## White.women 374 506 743 1140 1726  
## Black.men 361 483 680 1046 1551  
## Black.Women 331 423 615 935 1453  
## Asian.Men 420 648 1129 1860 2699  
## Asian.Women 385 551 877 1411 2024

WE\_DESIGN\_gender <- rep(c("Men", "Women"),4)  
WE\_DESIGN\_race <- rep(c("White", "Black", "Asian", "Hispanic"), each=2)

corr\_result = cor(WE\_data)  
corrplot(corr\_result)#,order = 'hclust', addrect = 7)



heatmap(WE\_data)

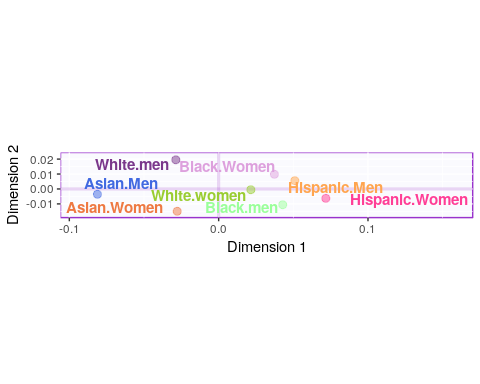


Weekly\_Earnings <- WE\_data  
resCA.sym <- epCA(Weekly\_Earnings, DESIGN = WE\_DESIGN\_gender, make\_design\_nominal = TRUE, symmetric = TRUE)

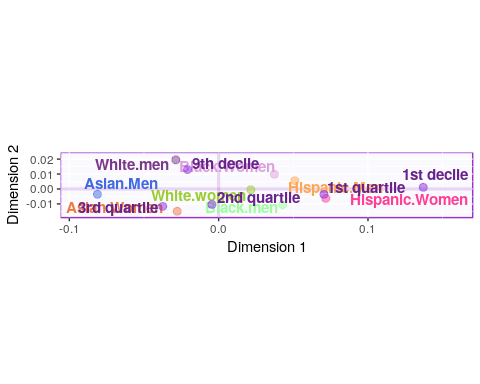
# to run a plain CA but asymetric  
resCA.asym <- epCA(Weekly\_Earnings, DESIGN = WE\_DESIGN\_gender, make\_design\_nominal = TRUE, symmetric = FALSE)

Fj.a <- resCA.asym$ExPosition.Data$fj  
Fi <- resCA.sym$ExPosition.Data$fi  
Fj <- resCA.sym$ExPosition.Data$fj  
constraints.sym <- minmaxHelper(mat1 = Fi, mat2 = Fj)  
constraints.asym <- minmaxHelper(mat1 = Fi, mat2 = Fj.a)

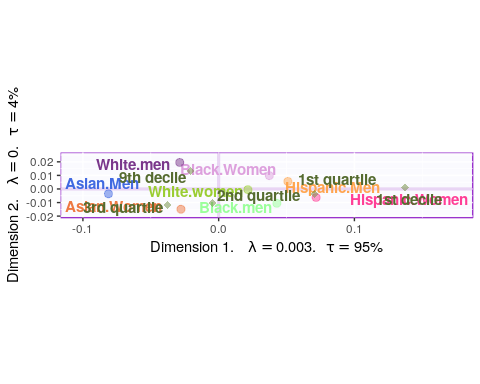
# Get some colors ----  
color4Authors <-prettyGraphsColorSelection(n.colors = nrow(Fi))  
# baseMaps ----  
baseMap.i <- createFactorMap(Fi, constraints = constraints.sym,  
 col.points = color4Authors,  
 col.labels = color4Authors)  
print(baseMap.i$zeMap)



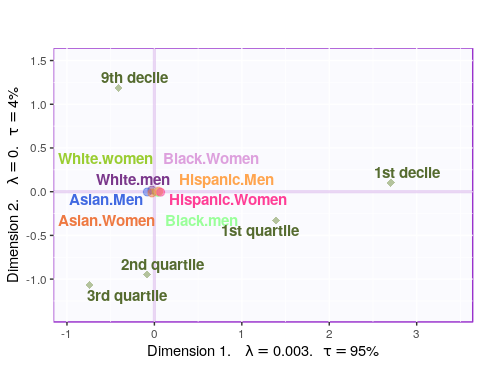
baseMap.j <- createFactorMap(Fj, constraints = constraints.sym,  
 color.points = 'darkorchid4')  
print(baseMap.i$zeMap + baseMap.j$zeMap\_dots + baseMap.j$zeMap\_text)



symMap <- createFactorMapIJ(Fi,Fj,  
 col.points.i = color4Authors,  
 col.labels.i = color4Authors)  
  
asymMap <- createFactorMapIJ(Fi,Fj.a,  
 col.points.i = color4Authors,  
 col.labels.i = color4Authors)  
  
labels4CA <- createxyLabels(resCA = resCA.sym)  
  
# draw the maps ----  
map.IJ.sym <- symMap$baseMap + symMap$I\_labels + symMap$I\_points +  
 symMap$J\_labels + symMap$J\_points + labels4CA  
print(map.IJ.sym)



map.IJ.asym <- asymMap$baseMap + asymMap$I\_labels +   
 asymMap$I\_points + asymMap$J\_labels +   
 asymMap$J\_points + labels4CA  
print(map.IJ.asym)



we\_data\_inf <- epCA.inference.battery(WE\_data, DESIGN = WE\_DESIGN\_gender, make\_design\_nominal = TRUE)

## [1] "It is estimated that your iterations will take 0.02 minutes."  
## [1] "R is not in interactive() mode. Resample-based tests will be conducted. Please take note of the progress bar."  
## ===========================================================================

PTCA4CATA::PlotScree(ev = resCA.sym$ExPosition.Data$eigs,  
 p.ev = we\_data\_inf$Inference.Data$components$p.vals,  
 title = 'SCREE Plot',  
 plotKaiser = TRUE  
)

