**Appendix 2. Code listing**

################################################################################

############# Header of code file. Adding all needed libraries #################

################################################################################

#install.packages("tictoc")

library(readr)

library(dplyr) # for data cleaning

library("stringr", lib.loc="/Library/Frameworks/R.framework/Versions/3.5/Resources/library")

library("plotly", lib.loc="/Library/Frameworks/R.framework/Versions/3.5/Resources/library")

library(naniar)

library(VIM)

library(FactoMineR)

library(missMDA)

library(cluster) # for gower similarity and pam

library(Rtsne) # for t-SNE plot

library(ggplot2) # for visualization

library(tictoc)

################################################################################

######################## Import educational dataset ###########################

################################################################################

education\_UKR\_2012\_2018 <- read\_csv("education\_UKR\_2012-2018.csv")

class(education\_UKR\_2012\_2018) #check the class of data we get

education\_UKR\_2012\_2018 = as.data.frame(education\_UKR\_2012\_2018) #make all data to be a data.frame

class(education\_UKR\_2012\_2018)

################################################################################

############################ Data understanding ###############################

################################################################################

# first I want to understand what are the unique statistical units of each of 30 columns that I can analyse.

list\_of\_unique\_values<- list(1:numb\_cols)

list\_of\_unique\_values

for (name in 1:numb\_cols) {

unique\_value<-education\_UKR\_2012\_2018[name]%>%

unique()

list\_of\_unique\_values[[name]]<- unique\_value

}

View(list\_of\_unique\_values)

# After checking the unique values we can conclude that columns "Level of educational attainment",

# "School subject", "Teaching experience", "Type of contract", "Reference area" and "Time period" are redundant, so we can drop them

education\_UKR\_2012\_2018 <- select (education\_UKR\_2012\_2018,-c(`Level of educational attainment`,

`School subject`,

`Teaching experience`,

`Type of contract`,

`Reference area`,

`Time Period`))

View(education\_UKR\_2012\_2018)

################################################################################

######################### Deal with missing values ############################

################################################################################

numb\_rows <- nrow(education\_UKR\_2012\_2018)

numb\_complete\_rows <- sum(complete.cases(education\_UKR\_2012\_2018))

numb\_rows # there are 3318 rows in the dataset

numb\_complete\_rows # but only 47 rows with no NA values!

1 - numb\_complete\_rows/numb\_rows # so, if we are going to drop all missing values we'll loose 98% of information, that is not possible

numb\_cols <-ncol(education\_UKR\_2012\_2018)

numb\_cols

################################################################################

#### Question 1: Discover attendance patterns for urban and rural locations ####

################################################################################

# we want get information that is assosiated with Rural and urban location grouped by wealth level

urban\_rural\_area <- filter(education\_UKR\_2012\_2018, (

((`Location` == "RUR:Rural") | (`Location` == "URB:Urban"))

# & (`Unit of measure` == "PT:Percentage")

# & (`Sex` !="\_T:Total")

# & (`Grade` !="\_T:Total")

# & (`Wealth quintile` !="\_T:Total")

)

)

View(urban\_rural\_area)

##################################################

########### Males in Rural area ################

##################################################

# Here I select data about males in Rural area

rural\_area\_male <-urban\_rural\_area %>%

filter(`Sex`=="M:Male")%>%

filter(`Statistical unit`=="NAR:Net attendance rate")%>%

filter(str\_detect(`Unit of measure`, "Percentage"))%>%

filter(str\_detect(`Location`, "Rural"))%>%

select\_if(~ length(unique(.)) > 1)

# Create a general variable - concatanation of education and wealth

rural\_area\_male$`Educ\_wealth\_male` <- paste(rural\_area\_male$`Level of education`,"\_",rural\_area\_male$`Wealth quintile`)

# delete variables education and wealth

rural\_area\_male <- select(rural\_area\_male,-c(1,2))

# Order variables

rural\_area\_male <- rural\_area\_male[order(rural\_area\_male$`Educ\_wealth\_male`),]

#Swap data

rural\_area\_male <-rural\_area\_male[ ,c(2,1)]

##################################################

########### Females in Rural area ################

##################################################

# Here I select data about females in Rural area

rural\_area\_female <-urban\_rural\_area %>%

filter(`Sex`=="F:Female")%>%

filter(`Statistical unit`=="NAR:Net attendance rate")%>%

filter(str\_detect(`Unit of measure`, "Percentage"))%>%

filter(str\_detect(`Location`, "Rural"))%>%

select\_if(~ length(unique(.)) > 1)

# Create a general variable - concatanation of education and wealth

rural\_area\_female$`Educ\_wealth\_female` <- paste(rural\_area\_female$`Level of education`,"\_",rural\_area\_female$`Wealth quintile`)

# delete variables education and wealth

rural\_area\_female <-select(rural\_area\_female,-c(1,2))

# Order variables

rural\_area\_female <- rural\_area\_female[order(rural\_area\_female$`Educ\_wealth\_female`),]

#Swap data

rural\_area\_female <-rural\_area\_female[,c(2,1)]

rural\_area\_female

##################################################

########### Males in Urban area ################

##################################################

# Here I select data about males in Rural area

urban\_area\_male <-urban\_rural\_area %>%

filter(`Sex`=="M:Male")%>%

filter(`Statistical unit`=="NAR:Net attendance rate")%>%

filter(str\_detect(`Unit of measure`, "Percentage"))%>%

filter(str\_detect(`Location`, "Urban"))%>%

select\_if(~ length(unique(.)) > 1)

# Create a general variable - concatanation of education and wealth

urban\_area\_male$`Educ\_wealth\_male` <- paste(urban\_area\_male$`Level of education`,"\_",urban\_area\_male$`Wealth quintile`)

# delete variables education and wealth

urban\_area\_male <- select(urban\_area\_male,-c(1,2))

# Order variables

urban\_area\_male <- urban\_area\_male[order(urban\_area\_male$`Educ\_wealth\_male`),]

#Swap data

urban\_area\_male <-urban\_area\_male[,c(2,1)]

urban\_area\_male

##################################################

########### Females in Urban area ################

##################################################

# Here I select data about females in Rural area

urban\_area\_female <-urban\_rural\_area %>%

filter(`Sex`=="F:Female")%>%

filter(`Statistical unit`=="NAR:Net attendance rate")%>%

filter(str\_detect(`Unit of measure`, "Percentage"))%>%

filter(str\_detect(`Location`, "Urban"))%>%

select\_if(~ length(unique(.)) > 1)

# Create a general variable - concatanation of education and wealth

urban\_area\_female$`Educ\_wealth\_female` <- paste(urban\_area\_female$`Level of education`,"\_",urban\_area\_female$`Wealth quintile`)

# delete variables education and wealth

urban\_area\_female <-select(urban\_area\_female,-c(1,2))

# Order variables

urban\_area\_female <- urban\_area\_female[order(urban\_area\_female$`Educ\_wealth\_female`),]

#Swap data

urban\_area\_female <-urban\_area\_female[,c(2,1)]

urban\_area\_female

# Create new data frame that combine male and female in urban and rural data

urban\_rural\_area\_male\_female <-urban\_area\_male

urban\_rural\_area\_male\_female <-cbind(urban\_rural\_area\_male\_female, rural\_area\_male[,2])

urban\_rural\_area\_male\_female <-cbind(urban\_rural\_area\_male\_female, urban\_area\_female[,2])

urban\_rural\_area\_male\_female <-cbind(urban\_rural\_area\_male\_female, rural\_area\_female[,2])

urban\_rural\_area\_male\_female

colnames(urban\_rural\_area\_male\_female) <- c("Educ\_wealth", "2013-male-urban", "2013-male-rural","2013-female-urban" , "2013-female-rural")

View(urban\_rural\_area\_male\_female)

# Plot male in urban and rural data

p <- plot\_ly(urban\_rural\_area\_male\_female, x = ~`Educ\_wealth`, y = ~`2013-male-urban`, type = 'bar', name = 'Males in urban area') %>%

add\_trace(y = ~`2013-male-rural`, name = 'Males in rural area') %>%

layout(xaxis= list(title = 'Wealth-Education level'), yaxis = list(title = 'Attendance rate in %'), barmode = 'group')

p

# Plot female in urban and rural data

p\_fem <- plot\_ly(urban\_rural\_area\_male\_female, x = ~`Educ\_wealth`, y = ~`2013-female-urban`, type = 'bar', name = 'Females in urban area', marker = list(color = 'rgb(102, 0, 255)')) %>%

add\_trace(y = ~`2013-female-rural`, name = 'Females in rural area', marker = list(color = 'rgb(204, 0, 153)')) %>%

layout(xaxis= list(title = 'Wealth-Education level'), yaxis = list(title = 'Attendance rate in %'), barmode = 'group')

p\_fem

# Plot female in urban and rural data

p\_gen <- plot\_ly(urban\_rural\_area\_male\_female, x = ~`Educ\_wealth`, y = ~`2013-male-urban`, type = 'bar', name = 'Males in urban area', marker = list(color = 'rgb(51, 153, 255)')) %>%

add\_trace(y = ~`2013-male-rural`, name = 'Males in rural area', marker = list(color = 'rgb(0, 102, 204)')) %>%

add\_trace(y = ~`2013-female-urban`, name = 'Females in urban area', marker = list(color = 'rgb(255, 204, 204)')) %>%

add\_trace(y = ~`2013-female-rural`, name = 'Females in rural area', marker = list(color = 'rgb(255, 102, 102)')) %>%

layout(xaxis= list(title = 'Wealth-Education level'), yaxis = list(title = 'Attendance rate in %'), barmode = 'group')

p\_gen

################ Mean check urban males #######################

#mean attendancy value for primary education of urban males for all classes

m\_male\_urban <- urban\_rural\_area\_male\_female[1:5,2]

mean(m\_male\_urban[!sapply(m\_male\_urban, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for lower-secondary education of urban males for all classes

m\_male\_urban <- urban\_rural\_area\_male\_female[7:11,2]

mean(m\_male\_urban[!sapply(m\_male\_urban, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for upper-secondary education of urban males for all classes

m\_male\_urban <- urban\_rural\_area\_male\_female[13:17,2]

mean(m\_male\_urban[!sapply(m\_male\_urban, function(x)isTRUE(all.equal(x, 0)))])

################ Mean check rural males #######################

#mean attendancy value for primary education of rural males for all classes

m\_male\_rural <- urban\_rural\_area\_male\_female[1:5,3]

mean(m\_male\_rural[!sapply(m\_male\_rural, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for lower-secondary education of rural males for all classes

m\_male\_rural <- urban\_rural\_area\_male\_female[7:11,3]

mean(m\_male\_rural[!sapply(m\_male\_rural, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for upper-secondary education of rural males for all classes

m\_male\_rural<- urban\_rural\_area\_male\_female[13:17,3]

mean(m\_male\_rural[!sapply(m\_male\_rural, function(x)isTRUE(all.equal(x, 0)))])

################ Mean check urban females #######################

#mean attendancy value for primary education of urban females for all classes

m\_fem\_urban <- urban\_rural\_area\_male\_female[1:5,4]

mean(m\_fem\_urban[!sapply(m\_fem\_urban, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for lower-secondary education of urban females for all classes

m\_fem\_urban <- urban\_rural\_area\_male\_female[7:11,4]

mean(m\_fem\_urban[!sapply(m\_fem\_urban, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for upper-secondary education of urban females for all classes

m\_fem\_urban <- urban\_rural\_area\_male\_female[13:17,4]

mean(m\_fem\_urban[!sapply(m\_fem\_urban, function(x)isTRUE(all.equal(x, 0)))])

################ Mean check rural females #######################

#mean attendancy value for primary education of rural females for all classes

m\_fem\_rur <- urban\_rural\_area\_male\_female[1:5,5]

mean(m\_fem\_rur[!sapply(m\_fem\_rur, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for lower-secondary education of rural females for all classes

m\_fem\_rur <- urban\_rural\_area\_male\_female[7:11,5]

mean(m\_fem\_rur[!sapply(m\_fem\_rur, function(x)isTRUE(all.equal(x, 0)))])

#mean attendancy value for upper-secondary education of rural females for all classes

m\_fem\_rur <- urban\_rural\_area\_male\_female[13:17,5]

mean(m\_fem\_rur[!sapply(m\_fem\_rur, function(x)isTRUE(all.equal(x, 0)))])

################################################################################

#### Question 2: Make an analysis of teachers’ qualification over the years ####

############# depending on sex and educational institutions level. #############

################################################################################

teachers\_data <- education\_UKR\_2012\_2018 %>%

filter(str\_detect(`Statistical unit`, "teacher")) %>%

select\_if(~ length(unique(.)) > 1)

View(teachers\_data)

# check what unique statistical units we can use for data exploration

unique\_value\_teachers\_data\_units<-teachers\_data[1]%>%

unique()

unique\_value\_teachers\_data\_units

# select only the data in %

teachers\_data\_perc <- teachers\_data%>%

filter(str\_detect(`Unit of measure`, "PT:Percentage"))

#select\_if(~ length(unique(.)) > 1)

View(teachers\_data\_perc)

# try to get information about what rows has less missing values

teachers\_data\_no\_missing\_vals <- teachers\_data[which.max(rowSums(!is.na(teachers\_data))),]

View(teachers\_data\_no\_missing\_vals)

# we've discovered that for statistical unit : "PTR:Pupil-teacher ratio" we have the smallest number

# of missing values, try to get all connected relevant data

pupil\_teacher\_ratio<- teachers\_data %>%

filter(`Statistical unit`=="PTR:Pupil-teacher ratio")%>%

filter(`Orientation`=="\_T:Total")%>%

select\_if(~ length(unique(.)) > 1)

pupil\_teacher\_ratio

# Order variables

pupil\_teacher\_ratio <- pupil\_teacher\_ratio[order(pupil\_teacher\_ratio$`Level of education`),]

pupil\_teacher\_ratio

# number of missing variables

gg\_miss\_var(pupil\_teacher\_ratio)

# aggr calculates and represents the number of missing entries in each variable

# and for certain combinations of variables (which tend to be missing simultaneously)

res<-summary(aggr(pupil\_teacher\_ratio, sortVar=TRUE))$combinations

matrixplot(pupil\_teacher\_ratio, sortby = 1)

# omit rows where more than 55% of data is missing, select data about all institutions

pupil\_teacher\_ratio <- pupil\_teacher\_ratio[is.na(pupil\_teacher\_ratio)%\*%rep(1,ncol(pupil\_teacher\_ratio))<=ncol(pupil\_teacher\_ratio)\*0.55,]

summary(pupil\_teacher\_ratio)

pupil\_teacher\_ratio\_all\_institutions<-pupil\_teacher\_ratio %>%

filter(`Type of institution`=="INST\_T:All institutions")%>%

select\_if(~ length(unique(.)) > 1)

pupil\_teacher\_ratio\_all\_institutions

#change all NA values to 0 values

pupil\_teacher\_ratio\_all\_institutions[is.na(pupil\_teacher\_ratio\_all\_institutions)] <- 0

# re-structure data

final\_df <- t(pupil\_teacher\_ratio\_all\_institutions)

final\_df <- final\_df[-c(1), ]

years <-c(2012:2017)

final\_df <-data.frame(years,final\_df)

colnames(final\_df) <-c("years","L0:Early childhood education", "L1:Primary education", "L2\_3:Secondary education", "L5T8:Tertiary education")

final\_df

p <- plot\_ly(final\_df, type = 'scatter', mode = 'markers') %>%

add\_trace(x = ~`years`,y = ~`L0:Early childhood education`, name = 'L0:Early childhood education') %>%

add\_trace(x = ~`years`,y = ~`L1:Primary education`, name = 'L1:Primary education') %>%

add\_trace(x = ~`years`,y = ~`L2\_3:Secondary education`, name = 'L2\_3:Secondary education') %>%

add\_trace(x = ~`years`,y = ~`L5T8:Tertiary education`, name = 'L5T8:Tertiary education') %>%

layout(

title = " Students teachers Ratio",

yaxis = list(title = "Ratio"))

p

################################################################################

##### Question 3: Make cluster analysis of the countries where Ukrainian ######

############# student will preferably go depending on educational level #######

################ (bac, license, master) over the years 2012-2018. ##############

################################################################################

students\_to\_country <- education\_UKR\_2012\_2018 %>%

filter(`Destination region` != "W00:All countries") %>%

select\_if(~ length(unique(.)) > 1)

# check what unique statistical units we can use for data exploration

unique\_value\_students\_to\_country<-students\_to\_country[1]%>%

unique()

unique\_value\_students\_to\_country

# I want to check dependencies for "OE:Outbound internationally mobile students"

students\_to\_country\_mobile <- students\_to\_country %>%

filter(`Statistical unit` == "OE:Outbound internationally mobile students") %>%

select\_if(~ length(unique(.)) > 1)

students\_to\_country\_mobile

scaled.dat <- scale(t(students\_to\_country\_mobile[,-1]))

scaled.dat

# so, it will be level of similarity to go to specific country over the years. But why? languages, cultural stuf, money, any kind of educational programs and diplomas

euroclust<-hclust(dist(t(scaled.dat)))

euroclust

plot(euroclust, labels=students\_to\_country\_mobile$`Destination region`)

# re-structure data

final\_df\_students\_mobile <- t(students\_to\_country\_mobile)

final\_df\_students\_mobile

final\_df\_students\_mobile <- apply(final\_df\_students\_mobile[-c(1), ],1,function(x) log(as.numeric(x)))

final\_df\_students\_mobile <- t(final\_df\_students\_mobile)

final\_df\_students\_mobile[(final\_df\_students\_mobile == -Inf)] <- 0

final\_df\_students\_mobile

years <-c(2012:2017)

final\_df\_students\_mobile <-data.frame(years,final\_df\_students\_mobile)

final\_df\_students\_mobile

colnames(final\_df\_students\_mobile) <-c("years","Sub\_Saharan\_Africa", "South\_and\_West\_Asia", "Oceania", "Central\_and\_Eastern\_Europe", "Central\_Asia", "East\_Asia", "Latin\_America", "North\_America\_and\_Western\_Europe", "East\_Asia\_and\_the\_Pacific", "Arab\_States","Latin\_America\_and\_the\_Caribbean")

final\_df\_students\_mobile

p <- plot\_ly(final\_df\_students\_mobile, type = 'scatter', mode = 'lines') %>%

add\_trace(x = ~`years`,y = ~`Sub\_Saharan\_Africa`, name = 'Sub-Saharan Africa') %>%

add\_trace(x = ~`years`,y = ~`South\_and\_West\_Asia`, name = 'South and West Asia') %>%

add\_trace(x = ~`years`,y = ~`Oceania`, name = 'Oceania') %>%

add\_trace(x = ~`years`,y = ~`Central\_and\_Eastern\_Europe`, name = 'Central and Eastern Europe') %>%

add\_trace(x = ~`years`,y = ~`Central\_Asia`, name = 'Central Asia') %>%

add\_trace(x = ~`years`,y = ~`East\_Asia`, name = 'East Asia') %>%

add\_trace(x = ~`years`,y = ~`Latin\_America`, name = 'Latin America') %>%

add\_trace(x = ~`years`,y = ~`North\_America\_and\_Western\_Europe`, name = 'North America and Western Europe') %>%

add\_trace(x = ~`years`,y = ~`East\_Asia\_and\_the\_Pacific`, name = 'East Asia and the Pacific') %>%

add\_trace(x = ~`years`,y = ~`Arab\_States`, name = 'Arab States') %>%

add\_trace(x = ~`years`,y = ~`Latin\_America\_and\_the\_Caribbean`, name = 'Latin America and the Caribbean') %>%

layout(

title = "International mobility",

yaxis = list(title = "Number of departing people (log scaled)"))

p

# regression

students\_to\_country\_mobile

students\_to\_country\_mobile\_norm <-students\_to\_country\_mobile[-c(1)]

students\_to\_country\_mobile\_norm

students\_to\_country\_mobile\_norm <- t(apply(students\_to\_country\_mobile\_norm,1,function(x) log(as.numeric(x))))

students\_to\_country\_mobile\_norm

scatter.smooth(x=students\_to\_country\_mobile\_norm[,1], y=students\_to\_country\_mobile\_norm[,2], main="Latin America ~ Western Europe")

# has no meaning

students\_to\_country\_mobile\_norm <- as.data.frame(students\_to\_country\_mobile\_norm)

students\_to\_country\_mobile\_norm

final\_df\_students\_mobile

library("ggpubr")

require(gridExtra)

plot1 <- ggscatter(final\_df\_students\_mobile, x="years", y = "Oceania",

add = "reg.line", conf.int = TRUE,

cor.coef = TRUE, cor.method = "pearson",

xlab = "years", ylab = "Oceania")

plot2 <- ggscatter(final\_df\_students\_mobile, x="years", y = "North\_America\_and\_Western\_Europe",

add = "reg.line", conf.int = TRUE,

cor.coef = TRUE, cor.method = "pearson",

xlab = "years", ylab = "North\_America\_and\_Western\_Europe")

plot3 <- ggscatter(final\_df\_students\_mobile, x="years", y = "Arab\_States",

add = "reg.line", conf.int = TRUE,

cor.coef = TRUE, cor.method = "pearson",

xlab = "years", ylab = "East\_Asia")

grid.arrange(plot1, plot2, plot3)

################################################################################

### Question 4: Discover patterns about preferred fields of studies for a ######

############# male/female student over the years (2012-2018) and make a #######

################ prognosis about 2019.##############

################################################################################

educ\_patters <- education\_UKR\_2012\_2018 %>%

filter(`Field of education` != "\_T:Total") %>%

filter(`Field of education` != "\_X:Unspecified") %>%

filter(`Field of education` != "\_Z:Not applicable") %>%

filter(`Sex` != "\_T:Total") %>%

select\_if(~ length(unique(.)) > 1)

View(educ\_patters)

educ\_patterns\_tertiary <- educ\_patters %>%

filter(`Statistical unit` == "FOSEP:Distribution of students in tertiary education by field of education") %>%

select\_if(~ length(unique(.)) > 1)

View(educ\_patterns\_tertiary)

# check what unique statistical units we can use for data exploration

unique\_unit\_educ\_patterns\_tertiary<-educ\_patterns\_tertiary[1]%>%

unique()

unique\_unit\_educ\_patterns\_tertiary

unique\_field\_educ\_patterns\_tertiary<-educ\_patterns\_tertiary[3]%>%

unique()

unique\_field\_educ\_patterns\_tertiary

# change char values to numbers

rows\_patterns <- nrow(educ\_patterns\_tertiary)

rows\_patterns

#Swap data

educ\_patterns\_tertiary <-educ\_patterns\_tertiary[ ,c(3,1,2,4,5,6)]

educ\_patterns\_tertiary

# Order variables

educ\_patterns\_tertiary <- educ\_patterns\_tertiary[order(educ\_patterns\_tertiary$`Field of education`),]

educ\_patterns\_tertiary

#append extra variable

Value <-c(1:40)

educ\_patterns\_tertiary <-data.frame(Value,educ\_patterns\_tertiary)

educ\_patterns\_tertiary

glimpse(educ\_patterns\_tertiary)

educ\_patterns\_tertiary <- educ\_patterns\_tertiary%>%

mutate(Field.of.education = factor(Field.of.education)) %>%

mutate(Level.of.education = factor(Level.of.education)) %>%

mutate(Sex = factor(Sex))

educ\_patterns\_tertiary

glimpse(educ\_patterns\_tertiary)

gower\_dist <- daisy(educ\_patterns\_tertiary,

metric = "gower")

summary(gower\_dist)

gower\_mat <- as.matrix(gower\_dist)

gower\_mat

# Output most similar pair

educ\_patterns\_tertiary[

which(gower\_mat == min(gower\_mat[gower\_mat != min(gower\_mat)]),

arr.ind = TRUE)[1, ], ]

# Output most dissimilar pair

educ\_patterns\_tertiary[

which(gower\_mat == max(gower\_mat[gower\_mat != max(gower\_mat)]),

arr.ind = TRUE)[1, ], ]

# Calculate silhouette width for many k using PAM

sil\_width <- c(NA)

for(i in 2:10){

pam\_fit <- pam(gower\_dist,

diss = TRUE,

k = i)

sil\_width[i] <- pam\_fit$silinfo$avg.width

}

# Plot sihouette width (higher is better)

plot(1:10, sil\_width,

xlab = "Number of clusters",

ylab = "Silhouette Width")

lines(1:10, sil\_width)

tic("run clustering")

pam\_fit <- pam(gower\_dist, diss = TRUE, k = 2)

pam\_results <- educ\_patterns\_tertiary%>%

mutate(cluster = pam\_fit$clustering) %>%

group\_by(cluster) %>%

do(the\_summary = summary(.))

toc()

pam\_results$the\_summary

educ\_patterns\_tertiary[pam\_fit$medoids,]

tsne\_obj <- Rtsne(gower\_dist, perplexity = 1.5 ,is\_distance = TRUE)

tsne\_data <- tsne\_obj$Y %>%

data.frame() %>%

setNames(c("X", "Y")) %>%

mutate(cluster = factor(pam\_fit$clustering),

Field.of.education = educ\_patterns\_tertiary$Field.of.education)

ggplot(aes(x = X, y = Y), data = tsne\_data) +

geom\_point(aes(color = cluster))