code

Cory Moore

2022-05-04

R Code  
Data Cleaning and Preparation  
#libraries   
library(dplyr)  
library(careless)  
library(psych)  
library(lessR)  
library(mirt)  
library(Hmisc)  
library(factoextra)  
library(sjmisc)  
  
#import data  
vi <- read.delim("data/data.csv", header=TRUE)  
  
###### CLEAN/PREP DATA: ########  
# remove cases that completed RIASEC and demographic items in <3 sec / items &  
# remove that are not US respondents (avoid cultural confounding)  
vi <- vi %>%   
 filter(country %in% c("MX","US"),  
 testelapse > 110,   
 surveyelapse > 65,  
 age < 100) #should I be more stringent?  
  
# delete longstring responders  
longstring <- longstring(vi[,1:48], avg=TRUE)  
boxplot(longstring)  
longstring  
ls <- longstring$longstr  
vi$ls <- ls #bind longstring values to df   
names(vi)  
  
vi <- vi %>% #deletes cases with longstring >= 16  
 filter(ls < 16)  
  
rm(longstring)  
  
#subset dataframe - only variables I need  
vi <- vi %>%  
 select(c(R1,R2,R3,R4,R5,R6,R7,R8,  
 I1,I2,I3,I4,I5,I6,I7,I8,  
 education,  
 gender,  
 engnat,  
 age,  
 race,  
 country))  
  
#recode categorical variables  
vi$education <- dplyr::recode(vi$education,  
 '1' = "Less than high school",  
 '2' = "High school",  
 '3' = "University degree",  
 '4' = "Graduate degree")  
  
  
vi$gender <- dplyr::recode(vi$gender,  
 '1' = "Male",  
 '2' = "Female",  
 '3' = "Other")  
  
vi$engnat <- dplyr::recode(vi$engnat,  
 '1' = "Yes",  
 '2' = "No")  
  
vi$race <- dplyr::recode(vi$race,  
 '1' = "Asian",  
 '2' = "Arab",  
 '3' = "Black",  
 '4' = "Indigenous Australian / Native American / White",  
 '5' = "Other")  
  
# remova all missing cases (ie, 0's)  
vi[vi==0] <- NA  
vi <- vi[complete.cases(vi),]  
  
#further clean US sample to remove cases where eng was not first language (assuming they are not native US)  
vi <- vi %>% filter(!(country=="US" & engnat=="No"))  
  
#subset further to specific interest dimensions   
I <- vi %>%  
 select(c(I1,I2,I3,I4,I5,I6,I7,I8))  
  
R <- vi %>%  
 select(c(R1,R2,R3,R4,R5,R6,R7,R8))  
  
demo <- vi %>%  
 select(c(education,  
 gender,  
 engnat,  
 age,  
 race,  
 country))  
  
### Inspect the data a bit ###  
summary(vi)  
hist.data.frame(vi)  
boxplot(I)  
boxplot(R)  
boxplot(vi$age)  
summary(vi$age)  
apply(vi, 2, table)  
table(vi$country)  
  
IRT (Check assumptions, Run GRM, Assess Model Fit)  
# internal reliability  
I.alpha <- psych::alpha(I)  
I.alpha  
  
R.alpha <- psych::alpha(R)  
R.alpha  
  
#scree plots   
fa.parallel(I)  
fa.parallel(R)  
  
#oblique rotation  
efa.I <- fa(I, nfactors=1, rotate="oblimin") #Investigative dimension  
print(efa.I, sort=TRUE)  
efa.I$values #EFA eigenvalues - unidimensional (factor two eigenvalue <.1)  
efa.I$loadings  
fa.diagram(efa.I)  
  
efa.R <- fa(R, nfactors=1, rotate="oblimin")  
print(efa.R, sort=TRUE)  
efa.R$values #EFA eigenvalues - unidimensional (factor two eigenvalue <.1)  
efa.R$loadings  
fa.diagram(efa.R)  
  
#double-check 2-factor solutions  
efa.I2 <- fa(I, nfactors=2, rotate="oblimin")  
print(efa.I2, sort=TRUE)  
efa.I2$values #EFA eigenvalues. REPORT THESE  
efa.I2$loadings  
fa.diagram(efa.I2)  
  
efa.R2 <- fa(R, nfactors=2, rotate="oblimin")  
print(efa.R2, sort=TRUE)  
efa.R2$values #EFA eigenvalues. REPORT THESE  
efa.R2$loadings  
fa.diagram(efa.R2)  
  
rm(efa.I,efa.I2,efa.R,efa.R2)  
  
  
"  
### RUN GRMs ###  
"  
# first create data subsets of US and MX   
vi.us <- filter(vi, country=='US')  
I.us <- vi.us %>%  
 select(c(I1,I2,I3,I4,I5,I6,I7,I8))  
R.us <- vi.us %>%  
 select(c(R1,R2,R3,R4,R5,R6,R7,R8))  
  
vi.mx <- filter(vi, country=="MX")  
I.mx <- vi.mx %>%  
 select(c(I1,I2,I3,I4,I5,I6,I7,I8))  
R.mx <- vi.mx %>%  
 select(c(R1,R2,R3,R4,R5,R6,R7,R8))  
  
  
# combined models  
grm.I.out <- mirt(I, model=1, itemtype="graded", SE=TRUE)  
grm.I.out  
  
grm.R.out <- mirt(R, model=1, itemtype="graded", SE=TRUE)  
grm.R.out  
  
# split group models  
grm.I.us <- mirt(I.us, model=1, itemtype="graded", SE=TRUE)  
grm.I.us  
  
grm.R.us <- mirt(R.us, model=1, itemtype="graded", SE=TRUE)  
grm.R.us  
  
grm.I.mx <- mirt(I.mx, model=1, itemtype="graded", SE=TRUE)  
grm.I.mx  
  
grm.R.mx <- mirt(R.mx, model=1, itemtype="graded", SE=TRUE)  
grm.R.mx  
  
  
"  
### Assess model fit ###  
"  
mirtCluster(4)  
### SPLIT BY GROUP (US(reference) and MX(focal))   
#model fit  
coef(grm.I.us)  
M2(grm.I.us, type="C2")  
  
coef(grm.R.us)  
M2(grm.R.us, type="C2")  
  
coef(grm.I.mx)  
M2(grm.I.mx, type="C2")  
  
coef(grm.R.mx)  
M2(grm.R.mx, type="C2")  
  
(grm.I.us.item.fit <- itemfit(grm.I.us))  
(grm.R.us.item.fit <- itemfit(grm.R.us))  
(grm.I.mx.item.fit <- itemfit(grm.I.mx))  
(grm.R.mx.item.fit <- itemfit(grm.R.mx))  
  
  
"  
### look at model probability functions and parameters ###  
"  
#### All of the below analyses are done with groups combined ####  
  
## Investigative ##   
plot(grm.I.out) #expected test scores  
plot(grm.I.out, type="info") #test info  
plot(grm.I.out, type="infoSE") #item info  
plot(grm.I.out, type="trace") #item CRCs  
  
plots.I.2 <- list()  
for(i in 1:length(I)){  
 plots.I.2[[i]]<-itemplot(grm.I.out,i)  
}  
plots.I.2 #prob functions not bad   
  
#item parameters ## remember difficulty (b) = -d/a   
(coef.table.I <- coef(grm.I.out, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.I <- coef(grm.I.out, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.I  
### NOTE you get DIFFERENT SEs for IRT parameters than default  
### RECALL that b1 = response 1 and 2, b2 = response 2 and 3, b3 = 3 and 4, b4 = 4 and 5  
  
  
## Realistic ##  
plot(grm.R.out) #expected test scores  
plot(grm.R.out, type="info") #test info  
plot(grm.R.out, type="infoSE") #item info  
plot(grm.R.out, type="trace") #item CRCs  
  
plots.R.2 <- list()  
for(i in 1:length(I)){  
 plots.R.2[[i]]<-itemplot(grm.R.out,i)  
}  
plots.R.2 ##prob functions not bad   
  
#item parameters (combined groups for R and I)  
(coef.table.R <- coef(grm.R.out, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.R <- coef(grm.R.out, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.R  
  
(coef.table.I <- coef(grm.I.out, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.I <- coef(grm.I.out, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.I  
  
  
#items paramets by group by interest  
(coef.table.R.us <- coef(grm.R.us, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.R.us <- coef(grm.R.us, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.R.us  
  
(coef.table.I.us <- coef(grm.I.us, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.I.us <- coef(grm.I.us, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.I.us  
  
(coef.table.R.mx <- coef(grm.R.mx, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.R.mx <- coef(grm.R.umx, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.R.us  
  
(coef.table.I.mx <- coef(grm.I.mx, simplify = TRUE, IRTpars = TRUE)[[1]])  
par.SE.I.mx <- coef(grm.I.mx, IRTpars=TRUE, printSE=TRUE) #print SE from non-IRT format  
par.SE.I.mx  
  
Demographic data  
#libraries   
library(dplyr)  
library(xlsx)  
  
#import data  
demo <- read.csv("data/demo.csv", header=T)  
  
  
##### change variable classes  
demo <- demo %>%  
 mutate(education=as.factor(education),  
 gender=as.factor(gender),  
 engnat=as.factor(engnat),  
 race=as.factor(race),  
 country=as.factor(country))  
  
## subset data by US and MX  
demo.us <- filter(demo, country=='US')  
demo.mx <- filter(demo, country=="MX")  
  
  
"##### demographics with groups combined #####"  
# Gender  
gender <- demo %>%  
 group\_by(gender) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
gender  
  
# to excel output  
write.xlsx(as.data.frame(gender), file="output/output.xlsx",sheetName="gender",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# Education  
education <- demo %>%  
 group\_by(education) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
education  
  
# to excel output  
write.xlsx(as.data.frame(education), file="output/output.xlsx",sheetName="education",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
# Country  
country <- demo %>%  
 group\_by(country) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
country  
  
# to excel output  
write.xlsx(as.data.frame(country), file="output/output.xlsx",sheetName="country",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
# Race  
race <- demo %>%  
 group\_by(race) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
race  
  
# to excel output  
write.xlsx(as.data.frame(race), file="output/output.xlsx",sheetName="race",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
# English  
english <- demo %>%  
 group\_by(engnat) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
english  
  
# to excel output  
write.xlsx(as.data.frame(english), file="output/output.xlsx",sheetName="english",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
# AGE  
#mean, min/max, and SD for age'  
age <- psych::describe(demo$age)   
age  
  
# to excel output  
write.xlsx(as.data.frame(age), file="output/output.xlsx",sheetName="age",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
"#### demographics for the US ####"  
# Gender  
gender.us <- demo.us %>%  
 group\_by(gender) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
gender.us  
  
# to excel output  
write.xlsx(as.data.frame(gender.us), file="output/output.xlsx",sheetName="gender.us",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# Education  
education.us <- demo.us %>%  
 group\_by(education) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
education.us  
  
# to excel output  
write.xlsx(as.data.frame(education.us), file="output/output.xlsx",sheetName="education.us",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# Race  
race.us <- demo.us %>%  
 group\_by(race) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
race.us  
  
# to excel output  
write.xlsx(as.data.frame(race.us), file="output/output.xlsx",sheetName="race.us",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
# AGE  
#mean, min/max, and SD for age'  
age.us <- psych::describe(demo.us$age)   
age.us  
  
# to excel output  
write.xlsx(as.data.frame(age.us), file="output/output.xlsx",sheetName="age.us",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
  
"#### demographics for MX ####"  
# Gender  
gender.mx <- demo.mx %>%  
 group\_by(gender) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
gender.mx  
  
# to excel output  
write.xlsx(as.data.frame(gender.mx), file="output/output.xlsx",sheetName="gender.mx",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# Education  
education.mx <- demo.mx %>%  
 group\_by(education) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
education.mx  
  
# to excel output  
write.xlsx(as.data.frame(education.mx), file="output/output.xlsx",sheetName="education.mx",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# Race  
race.mx <- demo.mx %>%  
 group\_by(race) %>%  
 summarise(n = n()) %>%  
 mutate(freq = round(n / sum(n),3)) %>%  
 arrange(desc(freq))  
race.mx  
  
# to excel output  
write.xlsx(as.data.frame(race.mx), file="output/output.xlsx",sheetName="race.mx",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
# AGE  
#mean, min/max, and SD for age'  
age.mx <- psych::describe(demo.mx$age)   
age.mx  
  
# to excel output  
write.xlsx(as.data.frame(age.mx), file="output/output.xlsx",sheetName="age.mx",col.names=TRUE,row.names=FALSE,append=TRUE)  
  
DIF Analysis  
#libraries   
library(dplyr)  
library(careless)  
library(psych)  
library(lessR)  
library(mirt)  
library(Hmisc)  
library(factoextra)  
library(sjmisc)  
  
"  
######## create get.dif.items function #########  
"  
  
get.dif.items <- function(f.data,p.val=.05,parms){  
 r.warnings = ""  
 keep.vars <- c("X2", "df", "p") # just keep these variables  
 f.data <- f.data[keep.vars]  
 f.data$p = round(f.data$p,3)  
 if(missing(f.data)) return('Missing model output out.list')  
 f.data$sig <- ifelse(f.data$p < p.val,'dif','no\_dif')  
 if(!missing(parms)){  
 if(nrow(f.data) == nrow(parms)){  
 f.data <- cbind(f.data,parms)   
 }else{  
 r.warnings = "There number of item parameters doesn't match the number of items "  
 r.warnings = paste(r.warnings,"given to get.dif.items. Item parameters omitted.")  
 }  
 }  
 dif.items <- subset(f.data, sig == 'dif')  
 no.dif.items <- subset(f.data, sig == 'no\_dif')  
 if(!missing(parms) && nrow(f.data) == nrow(parms)){  
 if(nrow(no.dif.items)>1){  
 no.dif.items <- no.dif.items[order(-no.dif.items$a1),]   
 }  
 }  
   
 r.list <- list(dif\_items = dif.items, no\_dif = no.dif.items, warnings = r.warnings)  
 return(r.list)  
}  
  
  
"  
###### import data and group by gender #####  
"  
vi <- read.csv("data/vi.csv", header=TRUE)  
R <- read.csv("data/realistic.csv", header=TRUE)   
I <- read.csv("data/investigative.csv", header=TRUE)   
  
  
##appends 'country' to each interest df  
R.ethn <- cbind(R, vi[c("country")])   
I.ethn <- cbind(I, vi[c("country")])  
  
# look at distributions of samples  
table(R.ethn$country)  
table(I.ethn$country)  
  
  
# creates vectors of gender variable values for later analyses   
group.R <- as.character(R.ethn$country)  
group.I <- as.character(I.ethn$country)  
  
  
################  
"############  
 ### IRT ####  
 ############"  
################  
mirtCluster(4) #speeds up processing  
  
"  
##### constrained baseline models ####  
"  
## Investigative dimension  
model.constrained.I <- multipleGroup(I, 1, group.I, invariance = c(colnames(I), 'free\_means', 'free\_var'), technical = list(NCYCLES = 2000))  
  
coef(model.constrained.I, simplify = T) #different group latent means  
constrained.parameters.I <- coef(model.constrained.I, simplify = T)[[1]][[1]]  
constrained.parameters.I #write this out to excel and put it in the manuscript  
  
# to excel output  
write.xlsx(as.data.frame(constrained.parameters.I), file="output/DIF.xlsx",sheetName="constrained.base.I",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
  
## Realistic dimension  
model.constrained.R <- multipleGroup(R, 1, group.R,invariance = c(colnames(R),'free\_means', 'free\_var'), technical = list(NCYCLES = 2000))  
  
coef(model.constrained.R, simplify = T) #different group latent means  
constrained.parameters.R <- coef(model.constrained.R, simplify = T)[[1]][[1]]  
constrained.parameters.R #write this out to excel and put it in the manuscript  
  
# to excel output  
write.xlsx(as.data.frame(constrained.parameters.R), file="output/DIF.xlsx",sheetName="constrained.base.R",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
  
"  
##### first round of DIF analyses - All Others As Anchors #####  
"  
## investigative ##  
(dif.drop.I <- DIF(model.constrained.I, c('a1','d1','d2','d3','d4'), scheme="drop", seq\_stat = 0.5, technical = list(NCYCLES = 2000))) #gives us chi-square - compares constrained baseline model to model where items in ? have been freed up across groups  
  
# Adam's function that tables the output --> all items have DIF  
get.dif.items(f.data=dif.drop.I, p.val=.05, parms=constrained.parameters.I)  
  
## realistic ##  
(dif.drop.R <- DIF(model.constrained.R, c('a1','d1','d2','d3','d4'), scheme="drop", seq\_stat = 0.5, technical = list(NCYCLES = 2000)))  
  
# Adam's function that tables the output --> all items have DIF  
get.dif.items(f.data=dif.drop.R, p.val=.05, parms=constrained.parameters.R)  
  
  
"  
#### Run an anchor-item model ####  
"  
## Investigative  
itemnames.I <- colnames(I)  
anc.items.names.I <- itemnames.I[c(3,7)]   
test.items.I <- c(1,5,6,8)  
model\_anchor.I <- multipleGroup(I, model = 1, group = group.I,  
 invariance = c(anc.items.names.I, 'free\_means', 'free\_var'),   
 technical = list(NCYCLES = 2000))  
(anchor.parms.I <-coef(model\_anchor.I,simplify = TRUE)[[1]][[1]])  
  
# to excel output  
write.xlsx(as.data.frame(anchor.parms.I), file="output/DIF.xlsx",sheetName="anchor.model.parms.I",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
## Realistic  
itemnames.R <- colnames(R)  
anc.items.names.R <- itemnames.R[c(2,6)]   
test.items.R <- c(3,4,5,8)  
model\_anchor.R <- multipleGroup(R, model = 1, group = group.R,  
 invariance = c(anc.items.names.R, 'free\_means', 'free\_var'),  
 technical = list(NCYCLES = 2000))  
(anchor.parms.R <-coef(model\_anchor.R,simplify = TRUE)[[1]][[1]])  
  
# to excel output  
write.xlsx(as.data.frame(anchor.parms.R), file="output/DIF.xlsx",sheetName="anchor.model.parms.R",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
"  
#### Final round of DIF (A5?) ####  
"  
## Investigative  
(dif.anchor.I <- DIF(model\_anchor.I, c('a1','d1','d2','d3'), items2test = test.items.I, plotdif = TRUE, technical = list(NCYCLES = 2000)))  
dif.anchor.I  
## use the optional function to table the output  
get.dif.items(f.data=dif.anchor.I, p.val=.05, parms = anchor.parms.I)  
  
## Realistic  
(dif.anchor.R <- DIF(model\_anchor.R, c('a1','d1','d2','d3'), items2test = test.items.R, plotdif = TRUE,technical = list(NCYCLES = 2000)))  
dif.an  
## use the optional function to table the output  
get.dif.items(f.data=dif.anchor.R, p.val=.05, parms = anchor.parms.R)  
  
  
"  
#### compute effect sizes ####  
"  
## Investigative  
ES.test.lvl.I <- empirical\_ES(model\_anchor.I, DIF=FALSE) # test level stats  
ES.item.lvl.I <- empirical\_ES(model\_anchor.I) # item level stats  
expected.test.plot.I <- empirical\_ES(model\_anchor.I, DIF=FALSE, plot=TRUE) # expected test score plots  
expected.test.plot.I  
expected.item.plots.I <- empirical\_ES(model\_anchor.I, plot=TRUE) # expected item score plots  
expected.item.plots.I  
itemplot(model\_anchor.I, 8) # further investigate item with DF  
  
# to excel output  
write.xlsx(as.data.frame(ES.test.lvl.I), file="output/DIF.xlsx",sheetName="ES.test.lvl.I",col.names=TRUE,row.names=TRUE,append=TRUE)  
write.xlsx(as.data.frame(ES.item.lvl.I), file="output/DIF.xlsx",sheetName="ES.item.lvl.I",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
  
## Realistic  
ES.test.lvl.R <-empirical\_ES(model\_anchor.R, DIF=FALSE) # test level stats  
ES.item.lvl.R <-empirical\_ES(model\_anchor.R) # item level stats  
expected.test.plot.R <- empirical\_ES(model\_anchor.R, DIF=FALSE, plot=TRUE) # expected test score plots  
expected.test.plot.R  
expected.item.plots.R <- empirical\_ES(model\_anchor.R, plot=TRUE) # expected item score plots  
expected.item.plots.R  
itemplot(model\_anchor.R, 8) # further investigate item with DF  
  
# to excel output  
write.xlsx(as.data.frame(ES.test.lvl.R), file="output/DIF.xlsx",sheetName="ES.test.lvl.R",col.names=TRUE,row.names=TRUE,append=TRUE)  
write.xlsx(as.data.frame(ES.item.lvl.R), file="output/DIF.xlsx",sheetName="ES.item.lvl.R",col.names=TRUE,row.names=TRUE,append=TRUE)  
  
  
"  
#### lattice graph plots ####  
"  
## Investigative  
expected.test.plot.I$main <- "ETS for Reference and Focal Groups"  
expected.test.plot.I$legend$top$args$key$text[[8]] <- c('Focal', 'Reference')  
expected.test.plot.I  
  
## Investigative  
expected.test.plot.R$main <- "ETS for Reference and Focal Groups"  
expected.test.plots.R$legend$top$args$key$text[[8]] <- c('Focal', 'Reference')  
expected.test.plot.R  
  
  
mirtCluster(remove=TRUE) #turns off extra processors