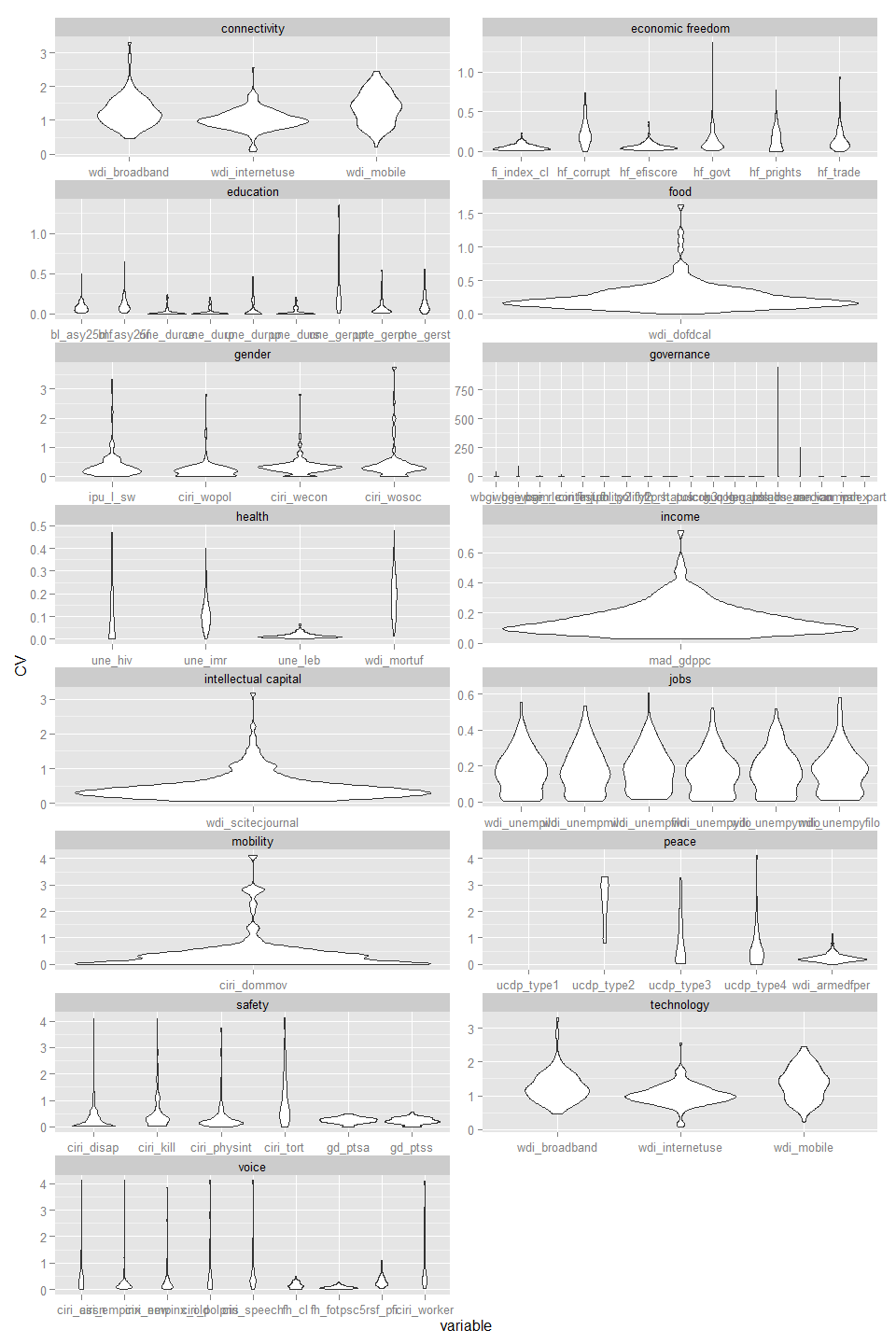
Macro-socio-ecology-variables

Friday, June 26, 2015

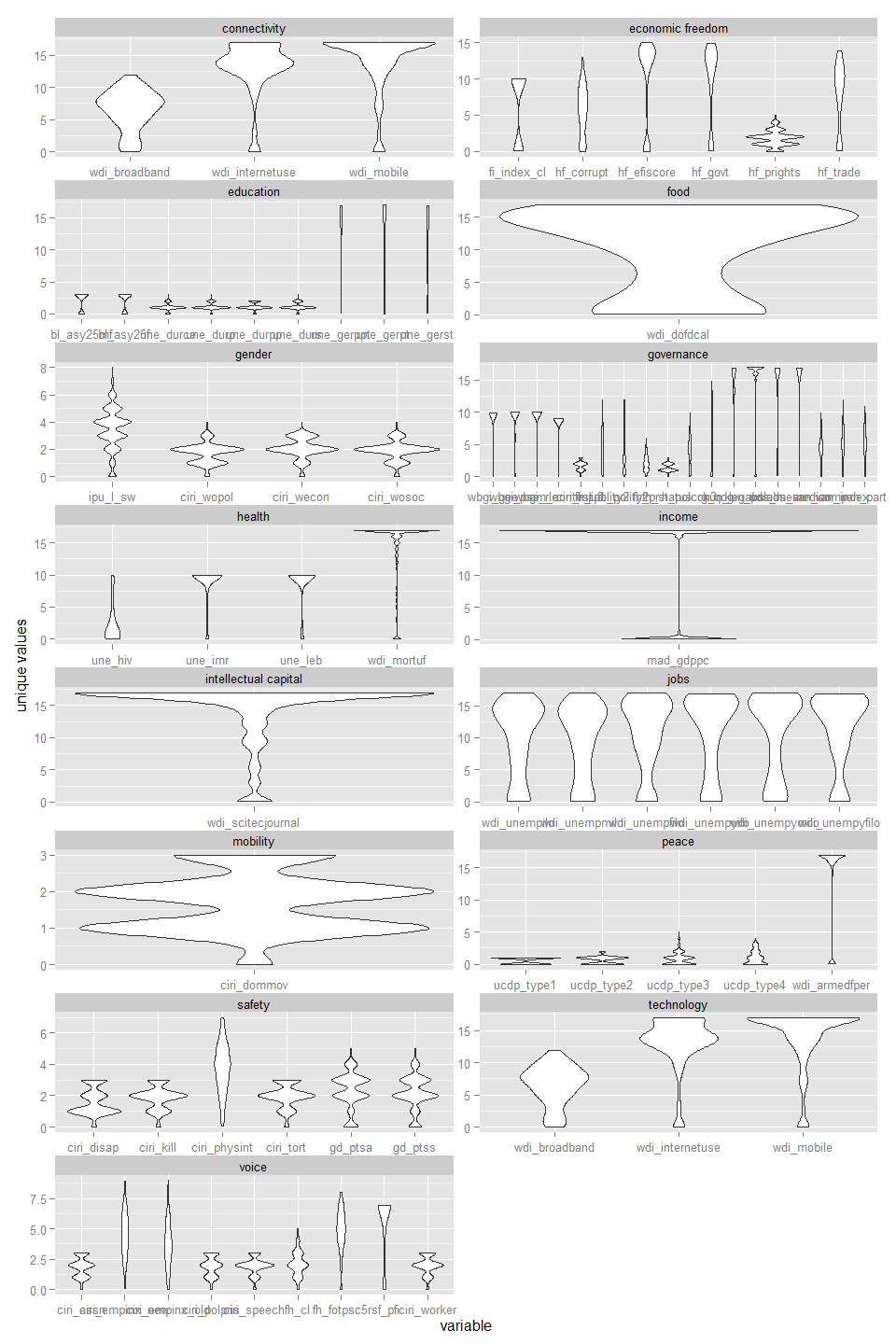
{% include JB/setup %}

### UNIQUE VALUES OF SOCIAL VARIABLES

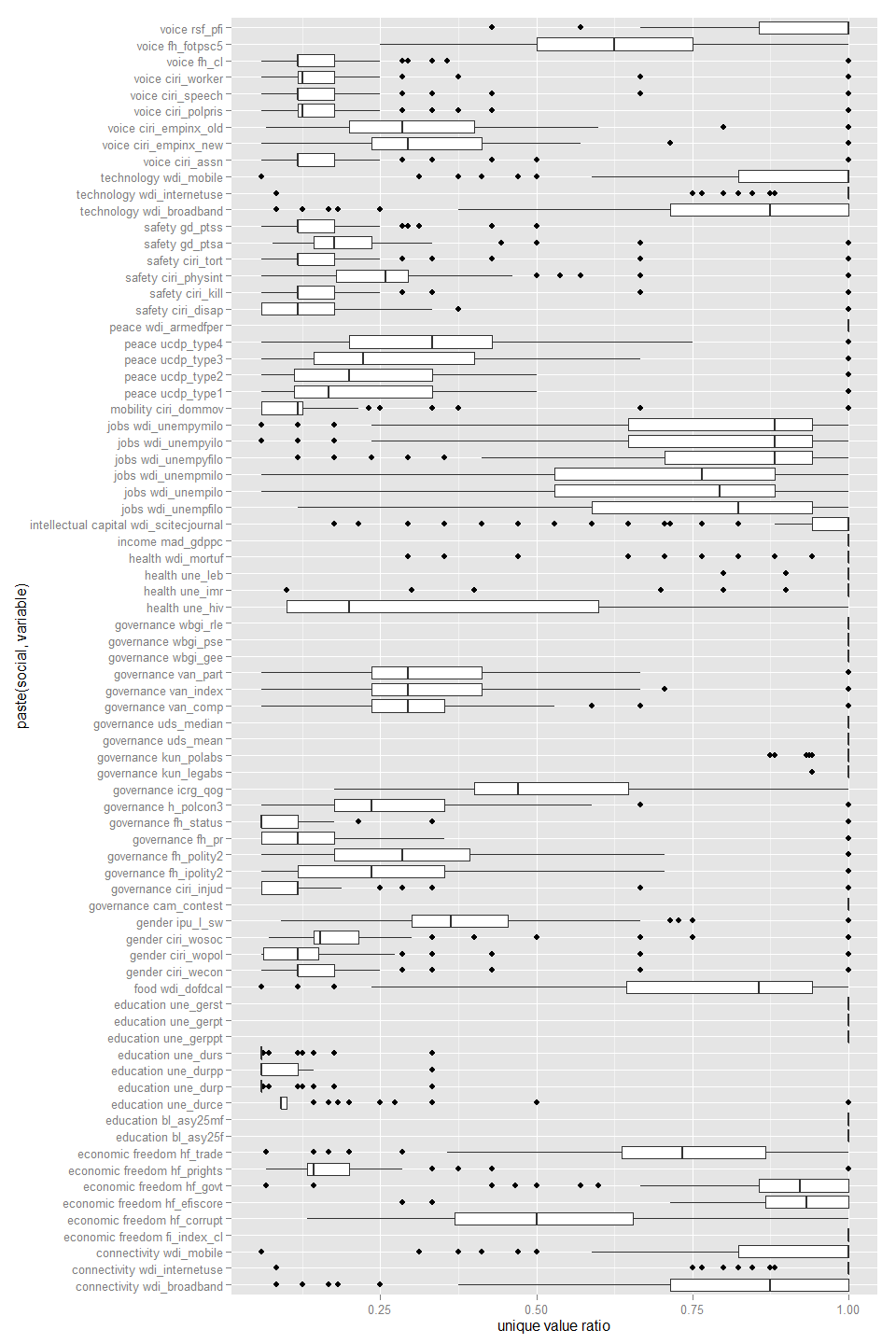
library(ggplot2);library(reshape);library(Hmisc)  
wd<-"G:/Documents/PostDocKVA/Labbook/projects/macro-socio-ecology"  
QoG.wd<-"G:\\Documents\\PostDocKVA\\Data\\QOG"  
WB.poverty.wd<-"G:\\Documents\\PostDocKVA\\Data\\WorldBank\\PovertyEquityDB"  
eora.wd<-"G:/Documents/PostDocKVA/Data/Eora" ### eora data directory  
  
setwd(eora.wd)  
### Reading in .csv file with annual gdp and population sizes  
gdppop.df<-read.csv("gdppop.csv",header=TRUE,skip=1) #skipping the first line which includes a description of the file  
### Reading in .csv file with various regional memberships  
region.df<-read.csv("regionmembership.csv",header=T,skip=1) #skipping the first line which includes a description of the file  
continent.df<-region.df[which(region.df[,"region\_type"]=="Continents"),c("region\_name","name","a3")]  
GDP.df<-region.df[which(region.df[,"region\_type"]=="GDP Regions"),c("region\_name","name","a3")]  
  
#ConsumTrend.df<-merge(ConsumTrend.df,continent.df,by.x="Country",by.y="name",all.z=TRUE)  
  
year.min<-1992  
year.max<-2008  
  
setwd(wd)  
  
var.df<-read.csv("social\_variables.csv",header=T,stringsAsFactors = FALSE,na.strings="")  
  
var.df<-var.df[which(is.na(var.df[,"varcode"])==FALSE),]  
  
data.dirs<-unique(var.df$dir)  
data.files<-unique(var.df$file)  
  
## reading in QoG data  
  
if(QoG.wd %in% data.dirs){  
   
setwd(QoG.wd)   
QoG.basts.df<-read.csv("qog\_bas\_ts\_jan15.csv",header=T,stringsAsFactors = FALSE,na.strings="")   
  
QoG.sel.vars<-unique(var.df[which(var.df$dataset == "QoG"),"varcode"])  
  
  
QoG.var.df<-var.df[which(var.df$dataset == "QoG"),]  
  
  
QoG.sel.df<-QoG.basts.df[,c("ccode","cname","year","ccodealp","cname\_year","ccodealp\_year","ccodecow","ccodewb","version",QoG.sel.vars)]  
  
#c("ccode","cname","year","ccodealp","cname\_year","ccodealp\_year","ccodecow","ccodewb","version",QoG.sel.vars)[c("ccode","cname","year","ccodealp","cname\_year","ccodealp\_year","ccodecow","ccodewb","version",QoG.sel.vars)%in%names(QoG.basts.df)==FALSE]  
  
for(i in 1:length(QoG.sel.vars)){  
  
row.i<-which(QoG.var.df[,"varcode"] == QoG.sel.vars[i])   
  
#print(row.i)  
  
if(is.na(row.i) == FALSE){   
QoG.sel.df[,paste(QoG.sel.vars[i])] <- QoG.sel.df[,QoG.sel.vars[i]]\*QoG.var.df[row.i,"multiplier"]  
}   
  
  
  
  
QoG.sel.df<-QoG.sel.df[which(QoG.sel.df$year %in% c(year.min:year.max)),]  
  
  
}  
  
}  
  
### CALCULATING WITHIN COUNTRY STANDARD DEVIATION  
  
country.df<-data.frame("country"=sort(unique(QoG.sel.df$ccodealp)))  
#country.df[,paste(QoG.sel.vars,"mean",sep=".")]<-NA  
#country.df[,paste(QoG.sel.vars,"median",sep=".")]<-NA  
  
  
QoG.sel.df<-QoG.sel.df[order(QoG.sel.df$ccodealp),]  
  
QoG.sd<-as.data.frame(apply(QoG.sel.df[,QoG.sel.vars],2,function(x,y=QoG.sel.df$ccodealp) tapply(x,y,function(z) abs(sd(z,na.rm=TRUE)/mean(z,na.rm=TRUE)))))  
  
#names(QoG.sd)<-paste(names(QoG.sd),"sd",sep=".")  
  
country.df<-as.data.frame(cbind(country.df,QoG.sd))  
#country.df[,paste(QoG.sel.vars,"sd",sep=".")]<-QoG.sd  
  
country.melt.sd.df<-melt(country.df[,c("country",names(QoG.sd))])  
  
#country.melt.sd.df[,"variable"]<-gsub(".sd","")  
country.melt.sd.df<-merge(country.melt.sd.df,var.df,by.x="variable",by.y="varcode",all.x=TRUE)  
  
#x11()  
ggplot(country.melt.sd.df[-which(country.melt.sd.df[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],aes(x=variable,y=value))+geom\_violin()+facet\_wrap(~social,scales="free",ncol=2)+ylab("CV")



####  
  
### CALCULATING WITHIN COUNTRY UNIQUE VALUES  
  
country.df<-data.frame("country"=sort(unique(QoG.sel.df$ccodealp)))  
#country.df[,paste(QoG.sel.vars,"mean",sep=".")]<-NA  
#country.df[,paste(QoG.sel.vars,"median",sep=".")]<-NA  
  
  
QoG.sel.df<-QoG.sel.df[order(QoG.sel.df$ccodealp),]  
  
QoG.unique<-as.data.frame(apply(QoG.sel.df[,QoG.sel.vars],2,function(x,y=QoG.sel.df$ccodealp) tapply(x,y,function(z) length(which(is.na(unique(z,na.rm=TRUE))==FALSE)))))  
  
#names(QoG.sd)<-paste(names(QoG.sd),"sd",sep=".")  
  
country.df<-as.data.frame(cbind(country.df,QoG.unique))  
#country.df[,paste(QoG.sel.vars,"sd",sep=".")]<-QoG.sd  
  
country.melt.unique.df<-melt(country.df[,c("country",names(QoG.unique))])  
  
#country.melt.sd.df[,"variable"]<-gsub(".sd","")  
country.melt.unique.df<-merge(country.melt.unique.df,var.df,by.x="variable",by.y="varcode",all.x=TRUE)  
  
#x11()  
ggplot(country.melt.unique.df[-which(country.melt.unique.df[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],aes(x=variable,y=value))+geom\_violin()+facet\_wrap(~social,scales="free",ncol=2)+ylab("unique values")

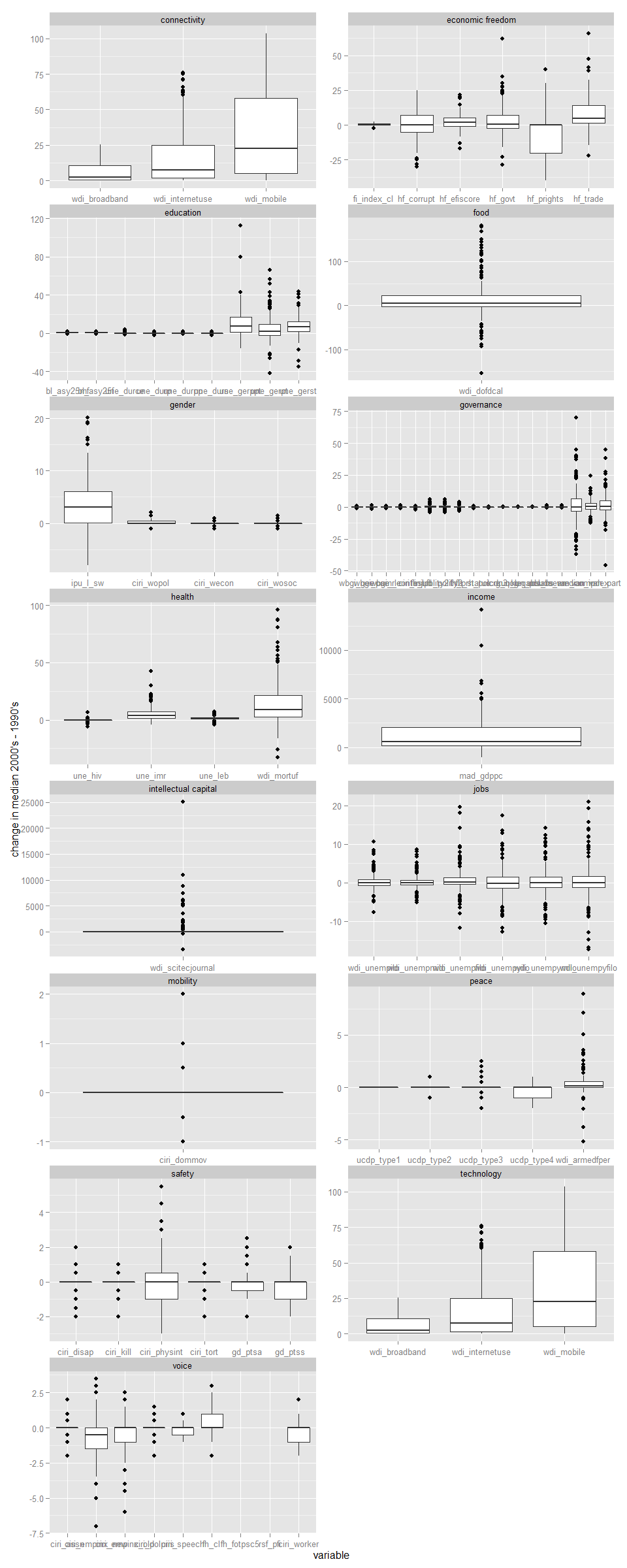


### CALCULATING WITHIN COUNTRY UNIQUE VALUES PER TIME SERIES LENGTH  
  
country.df<-data.frame("country"=sort(unique(QoG.sel.df$ccodealp)))  
  
QoG.sel.df<-QoG.sel.df[order(QoG.sel.df$ccodealp),]  
  
#QoG.relunique<-as.data.frame(apply(QoG.sel.df[,QoG.sel.vars],2,function(x,y=QoG.sel.df$ccodealp) tapply(x,y,function(z) length(which(is.na(unique(z,na.rm=TRUE))==FALSE))/  
# length(which(is.na(z)==FALSE)))))  
  
#QoG.relunique<-as.data.frame(apply(QoG.sel.df[,QoG.sel.vars],2,function(x,y=QoG.sel.df$ccodealp) tapply(x,y,function(z,w=length(which(is.na(unique(z))==FALSE))) ifelse(w>0,  
# length(which(is.na(unique(z))==FALSE))/  
# length(which(is.na(z)==FALSE)),NA)  
# )))  
  
  
QoG.relunique<-as.data.frame(apply(QoG.sel.df[,QoG.sel.vars],2,function(x,y=QoG.sel.df$ccodealp) tapply(x,y,function(z)  
 length(which(is.na(z)==FALSE))  
 )))  
  
#QoG.relunique<-as.data.frame((as.matrix(QoG.unique)-1)/as.matrix(QoG.relunique))  
QoG.relunique<-as.data.frame(as.matrix(QoG.unique)/as.matrix(QoG.relunique))  
  
  
  
#names(QoG.sd)<-paste(names(QoG.sd),"sd",sep=".")  
  
country.df<-as.data.frame(cbind(country.df,QoG.relunique))  
#country.df[,paste(QoG.sel.vars,"sd",sep=".")]<-QoG.sd  
  
country.melt.relunique.df<-melt(country.df[,c("country",names(QoG.relunique))])  
  
#country.melt.sd.df[,"variable"]<-gsub(".sd","")  
country.melt.relunique.df<-merge(country.melt.relunique.df,var.df,by.x="variable",by.y="varcode",all.x=TRUE)  
  
  
#x11()  
ggplot(country.melt.relunique.df[-which(country.melt.relunique.df[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],  
 aes(x=paste(social,variable),y=value))+geom\_boxplot()+ylab("unique value ratio")+coord\_flip()#+facet\_wrap(~social,ncol=2)

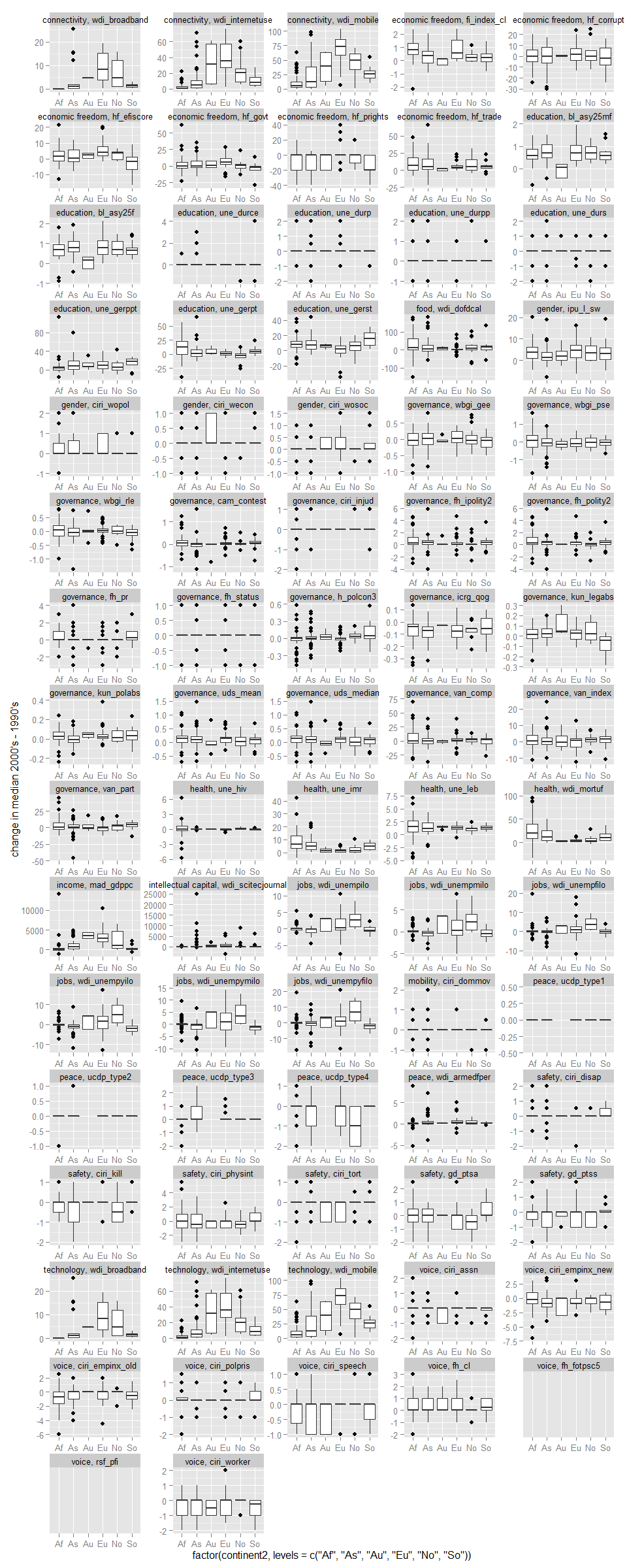


### MEDIAN VALUES IN FIRST AND SECOND TIME PERIOD

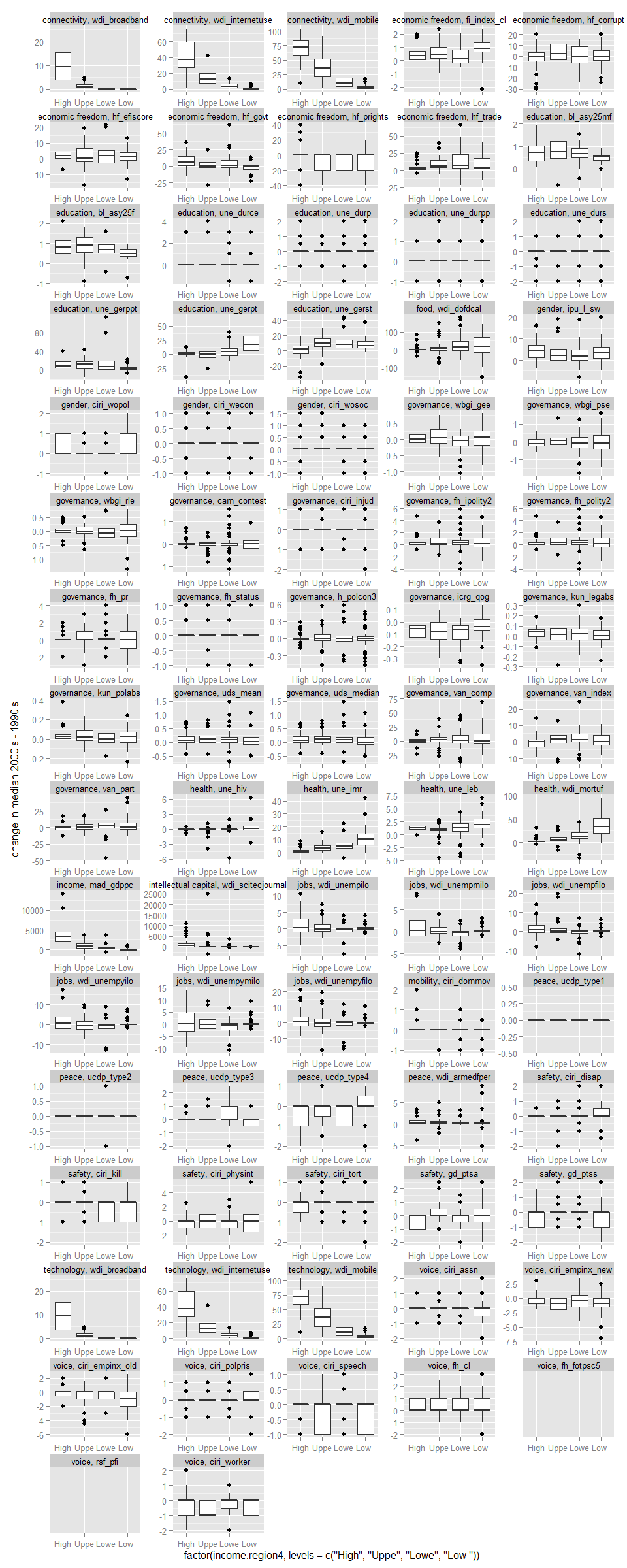
QoG.sel.df[,"period"]<-NA  
QoG.sel.df[,"period"]<-ifelse(QoG.sel.df[,"year"]<2000,"a",ifelse(QoG.sel.df[,"year"]>1999,"b",NA))  
  
  
QoG.sel.df<-QoG.sel.df[order(QoG.sel.df$ccodealp),]  
  
## MEDIAN FIRST PERIOD  
QoG.sd.first<-as.data.frame(apply(QoG.sel.df[which(QoG.sel.df$year %in% c(1992:1999)),QoG.sel.vars],2,function(x,y=QoG.sel.df[which(QoG.sel.df$year %in% c(1992:1999)),"ccodealp"]) tapply(x,y,function(z) median(z,na.rm=TRUE))))  
  
QoG.sd.first[,c("ccodealp","year")]<-cbind(rownames(QoG.sd.first),rep(1990,nrow(QoG.sd.first)))  
QoG.sd.first.melt<-melt(QoG.sd.first,id.var=c("ccodealp","year"))  
names(QoG.sd.first.melt)<-c("ccodealp","first.period","variable","first.value")  
  
## MEDIAN SECOND PERIOD  
QoG.sd.second<-as.data.frame(apply(QoG.sel.df[which(QoG.sel.df$year %in% c(2000:2008)),QoG.sel.vars],2,function(x,y=QoG.sel.df[which(QoG.sel.df$year %in% c(2000:2008)),"ccodealp"]) tapply(x,y,function(z) median(z,na.rm=TRUE))))  
  
QoG.sd.second[,c("ccodealp","year")]<-cbind(rownames(QoG.sd.second),rep(2000,nrow(QoG.sd.second)))  
QoG.sd.second.melt<-melt(QoG.sd.second,id.var=c("ccodealp","year"))  
names(QoG.sd.second.melt)<-c("ccodealp","second.period","variable","second.value")  
  
## MANN WHITNEY TEST  
MUtest.list<-by(QoG.sel.df,QoG.sel.df[,"ccodealp"],function(x) apply(x[,QoG.sel.vars],2,function(y,z=x[,"period"],country=x[1,"ccodealp"],  
 w1=length(which(is.na(y[which(x$year %in% c(1992:1999))])==FALSE)),  
 w2=length(which(is.na(y[which(x$year %in% c(2000:2008))])==FALSE))) #print(c(w1,w2))  
 if(w1 > 2 & w2 > 2){c(unlist(kruskal.test(y, factor(z)))[1:3],country)} else {c(rep(NA,3),country)}#pairwise.wilcox.test  
 ))  
  
MUtest.df<-as.data.frame(t(do.call(cbind,lapply(MUtest.list,function(x) rbind(names(as.data.frame(x)),x)  
 ))))  
  
names(MUtest.df)<-c("varcode","KW.chisq","KW.dfres","KW.pval","ccodealp")  
  
  
###############################  
  
  
  
## MERGING MEDIANS OF FIRST AND SECOND PERIOD AND CALCULATING DIFFERENCE IN MEDIAN VALUE  
QoG.sd.melt<-merge(QoG.sd.first.melt,QoG.sd.second.melt,by=c("ccodealp","variable"),all=TRUE)  
QoG.sd.melt[,"diff.value"]<-QoG.sd.melt[,"second.value"]-QoG.sd.melt[,"first.value"]  
  
## MERGING RESULTS FROM KRUSKAL WALLIS TESTS  
QoG.sd.melt<-merge(QoG.sd.melt,MUtest.df,by.x=c("ccodealp","variable"),by.y=c("ccodealp","varcode"),all.x=TRUE)  
  
  
  
### ADDING INFORMATION ABOUT SOCIAL VARIABLES  
QoG.sd.melt<-merge(QoG.sd.melt,var.df,by.x="variable",by.y="varcode",all.x=TRUE)  
  
### ADDING GEOGROAPHICAL AND ECONOMIC GROUPINGS  
QoG.sd.melt<-merge(QoG.sd.melt,QoG.sel.df[,c("ccode","cname","ccodealp")],by="ccodealp",all.x=TRUE)  
  
QoG.sd.melt<-merge(QoG.sd.melt,continent.df[,c("a3","region\_name")],by.x="ccodealp",by.y="a3",all.x=TRUE)  
names(QoG.sd.melt)[grep("region\_name",names(QoG.sd.melt))]<-"continent"  
  
QoG.sd.melt<-merge(QoG.sd.melt,GDP.df[,c("a3","region\_name")],by.x="ccodealp",by.y="a3",all.x=TRUE)  
names(QoG.sd.melt)[grep("region\_name",names(QoG.sd.melt))]<-"income.region"  
  
  
QoG.sd.melt[,"continent2"]<-strtrim(QoG.sd.melt[,"continent"], 2)  
QoG.sd.melt[,"income.region2"]<-strtrim(QoG.sd.melt[,"income.region"], 2)  
QoG.sd.melt[,"income.region4"]<-strtrim(QoG.sd.melt[,"income.region"], 4)  
  
QoG.sd.melt.plot<-QoG.sd.melt[which(is.na(QoG.sd.melt[,"continent"])==FALSE & is.na(QoG.sd.melt[,"income.region"])==FALSE),]  
# all toghether  
#x11()  
  
ggplot(QoG.sd.melt.plot[-which(QoG.sd.melt.plot[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],aes(x=variable,y=diff.value))+geom\_boxplot()+facet\_wrap(~social,scales="free",ncol=2)+ylab("change in median 2000's - 1990's")



#by continent  
#x11()  
  
ggplot(QoG.sd.melt.plot[-which(QoG.sd.melt.plot[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],  
 aes(x=factor(continent2,levels=c("Af","As","Au","Eu","No","So")),y=diff.value))+geom\_boxplot()+facet\_wrap(~social+variable,scales="free",ncol=5)+ylab("change in median 2000's - 1990's")



#by income region  
#x11()  
  
ggplot(QoG.sd.melt.plot[-which(QoG.sd.melt.plot[,"variable"]%in%c("cam\_inclusive","wbgi\_cce")),],  
 aes(x=factor(income.region4,levels=c("High","Uppe","Lowe","Low ")),y=diff.value))+geom\_boxplot()+facet\_wrap(~social+variable,scales="free",ncol=5)+ylab("change in median 2000's - 1990's")



### READING IN MATERIAL AND ENERGY FOOTPRINT

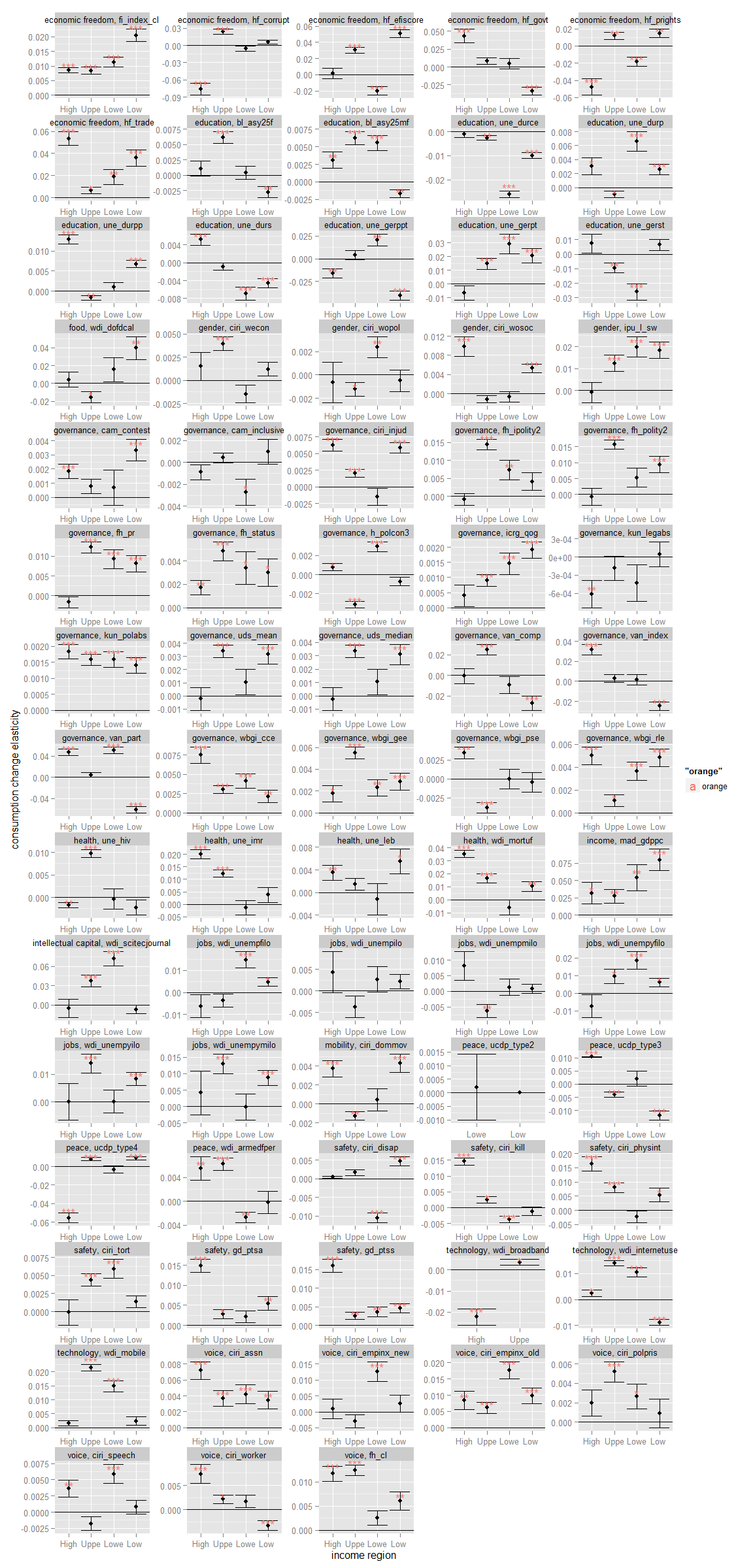
setwd(paste(eora.wd))  
  
## READING IN DATA  
### MATERIAL USE DATA - ENERGY DATASET  
energy.df<-read.csv("TradeBalance\_I-ENERGY.csv",header=TRUE)  
### Reading in .csv file with annual gdp and population sizes  
gdppop.df<-read.csv("gdppop.csv",header=TRUE,skip=1) #skipping the first line which includes a description of the file  
### Reading in .csv file with various regional memberships  
region.df<-read.csv("regionmembership.csv",header=T,skip=1) #skipping the first line which includes a description of the file  
  
  
## REMOVING NEGATIVE AND ZERO CONSUMPTION ENTRIES  
energy.df<-energy.df[which(energy.df[,"Consumption"]>0),]  
  
## REMOVING NEGATIVE AND ZERO CONSUMPTION ENTRIES  
#energy.df<-energy.df[which(as.character(energy.df$Country)!="Former USSR"),]  
  
  
## merging the gdp and population size data onto the energy consumption data frame  
energy.df<-merge(energy.df,gdppop.df,by=c("CountryA3","y","Country"),all.x=TRUE)  
  
  
## To make consumption more comparable let's calculate per capita consumption by associating population data  
  
### calculate per capita consumption and gdp consumption intensity by associating population data  
energy.df[,"Consum.pop.int"]<-energy.df[,"Consumption"]/energy.df[,"val"]  
energy.df[,"Consum.gdp.int"]<-energy.df[,"Consumption"]/energy.df[,"GDP"]  
  
### calculating scaled consumption intensitities with respect to population size and gdp  
energy.df<-energy.df[order(energy.df[,"Country"],energy.df[,"y"]),]  
energy.df[,"Consum.pop.int.scale"]<-unlist(by(energy.df,energy.df[,"Country"], function(x) scale(x[,"Consum.pop.int"],center=TRUE,scale=TRUE)))  
energy.df[,"Consum.gdp.int.scale"]<-unlist(by(energy.df,energy.df[,"Country"], function(x) scale(x[,"Consum.gdp.int"],center=TRUE,scale=TRUE)))  
  
energy.df<-energy.df[-which( energy.df[,"y"] %in% c(1991,2000,2011)),]  
  
  
## Calculating CONSUMPTION BALANCE with respect to EXTRACTION and IMPORTS  
  
### Subtracting exports from imports and domestic extraction in proportion to their size.  
  
### Conceptual code  
#### consum.extract <- extraction - (export \* extraction/(import+extraction))  
#### consum.import <- import - (export \* import/(import+extraction))  
  
### two ways of estimating domestic extraction  
  
#energy.df[,"Extraction"]<- (energy.df[,"TerritorialEmissions"] + energy.df[,"DirectEmissions"]) ## These two categories both refer to domestically extracted resources, i thik. Check up with Wiedmann dataset.  
  
energy.df[,"Extraction"]<- energy.df[,"TerritorialEmissions"]  
  
### consumed resources that come from domestic extraction  
  
energy.df[,"Consum.extract"]<- (energy.df[,"Extraction"]) -   
 (energy.df[,"Exports"] \* ((energy.df[,"Extraction"]) /  
 (energy.df[,"Extraction"] + energy.df[,"Imports"])))  
  
  
### consumed resources that come from imports  
  
energy.df[,"Consum.import"]<- (energy.df[,"Imports"]) -   
 (energy.df[,"Exports"] \* ((energy.df[,"Imports"]) /  
 (energy.df[,"Extraction"] + energy.df[,"Imports"])))  
  
  
### consumption balance index  
  
energy.df[,"Consum.balance"]<-(energy.df[,"Consum.extract"]-energy.df[,"Consum.import"])/energy.df[,"Consumption"]  
energy.df<-energy.df[-which(energy.df[,"Consum.balance"]>1),] ## removing entries that are above 1 (i.e. all resources being domestically extracted)  
  
  
  
### calculating mean gdp and population size  
  
energy.df<-energy.df[order(energy.df[,"Country"],energy.df[,"y"]),]  
energy.df[,"mean.gdp"]<-unlist(tapply(energy.df[,"GDP"],energy.df[,"Country"],function(x) rep(mean(x,na.rm=TRUE),length(x))))  
energy.df[,"mean.val"]<-unlist(tapply(energy.df[,"val"],energy.df[,"Country"],function(x) rep(mean(x,na.rm=TRUE),length(x))))  
energy.df[,"mean.gdp.per.cap"]<-energy.df[,"mean.gdp"]/energy.df[,"mean.val"]  
energy.df[,"log10.mean.gdp.per.cap"]<-log10(energy.df[,"mean.gdp.per.cap"])  
  
### country membership of grouped population size  
  
val.membership.df<-data.frame("mean.val"=unique(energy.df[,"mean.val"]),"val.membership"=NA)  
val.membership.df[,"val.membership"]<-as.numeric(cut2(val.membership.df$mean.val,m=20))  
  
energy.df<-merge(energy.df,val.membership.df,by="mean.val",all.x=TRUE)  
  
  
#####  
  
## CLEANING OF DATASET  
  
### exclude the two sudan's - Sudan, South Sudan  
# exclude Montenegro, Gaza Strip  
  
### include Andorra, Monaco, San Marino, Liechtenstein despite constant -1 index  
  
## Include Czech Republic 1993 - 2010  
## Eritrea 1991-2010, Greenland 1991-2007  
  
# Exclude 2008-2010 for countries where it suddently drops to -1  
  
  
Consum.balance.BA2006.df<-do.call(rbind,by(energy.df,energy.df[,"Country"],function(x) c(mean(x[which(x[,"y"]%in%c(1970:2007)),"Consum.balance"]),  
 mean(x[which(x[,"y"]%in%c(2008:2010)),"Consum.balance"])))  
 )  
Remove.0810.vec<-rownames(Consum.balance.BA2006.df[which(Consum.balance.BA2006.df[,1]>-1 & Consum.balance.BA2006.df[,2]==-1),])  
Remove.country.vec<-c("Sudan","South Sudan","Montenegro","Gaza Strip","Former USSR","Statistical Discrepancies")  
From.1991.vec<-c("Eritrea","Greenland")  
From.1993.vec<-c("Czech Republic")  
  
energy.df<-energy.df[-which( energy.df[,"Country"] %in% Remove.0810.vec & energy.df[,"y"] > 2007 ), ]  
energy.df<-energy.df[ which( energy.df[,"Country"] %in% setdiff( energy.df[,"Country"], Remove.country.vec) ), ]  
energy.df<-energy.df[ -which( energy.df[,"Country"] %in% From.1991.vec & energy.df[,"y"] < 1991), ]  
energy.df<-energy.df[ -which( energy.df[,"Country"] == From.1993.vec & energy.df[,"y"] < 1993), ]  
  
### We still see a lot of strange dynamics in 2008 possibly related to the financial crisis - so for this first analysis let's onlylook at years up to that time.  
energy.df <- energy.df[which(energy.df[,"y"] < 2008), ]  
  
### mean consumption.balance index  
energy.df <- energy.df[order(energy.df[,"Country"],energy.df[,"y"]),]  
energy.df[,"Consum.balance.avg"]<-unlist(by(energy.df, energy.df[,"Country"], function(x) rep(mean(x[,"Consum.balance"]),nrow(x))))  
#### consumption avg - data frame  
energy.ConsumAvg.df<-unique(energy.df[,c("Country","Consum.balance.avg","mean.val","mean.gdp")])  
  
### trend in energy consumption balance  
energy.df <- energy.df[order(energy.df[,"Country"],energy.df[,"y"]),]  
energy.df[,"ConsumBalance.y.trend"]<-unlist(by(energy.df,energy.df[,"Country"],function(x) rep(coef(lm(x[,"Consum.balance"]~x[,"y"]))[2],nrow(x))))  
#### consumption balance trend - data frame  
energy.consumtrend.df<-unique(energy.df[,c("Consum.balance.avg","mean.gdp.per.cap","ConsumBalance.y.trend","log10.mean.gdp.per.cap"),])  
  
  
  
#########################################################################################################################################################################  
#########################################################################################################################################################################  
#########################################################################################################################################################################  
  
  
  
## SETTING DIRECTORY FOR Wiedmann - material footprint - EORA DATA ON LOCAL HARD DRIVE  
setwd(paste(eora.wd,"/Wiedmann",sep=""))  
#dir()  
  
#### We are interested in the following files  
  
#"TradeBalance\_CONSTMA.csv","TradeBalance\_ORES.csv","TradeBalance\_FFUEL.csv","TradeBalance\_BIOMASS.csv","TradeBalance\_I-CMFA-TOT.csv"  
  
### We will read the files we are interested in into a list  
dframes.list<-list()  
  
## FOSSIL FUELS  
dframes.list[["FFUEL"]]<-read.csv("TradeBalance\_FFUEL.csv",header=T)  
dframes.list[["FFUEL"]][,"NAME"]<-"FFUEL"  
## BIOMASS  
dframes.list[["BIOMASS"]]<-read.csv("TradeBalance\_BIOMASS.csv",header=T)  
dframes.list[["BIOMASS"]][,"NAME"]<-"BIOMASS"  
  
## CONSTRUNCTION MATERIALS  
dframes.list[["CONSTMA"]]<-read.csv("TradeBalance\_CONSTMA.csv",header=T)  
dframes.list[["CONSTMA"]][,"NAME"]<-"CONSTMA"  
  
## ORES  
dframes.list[["ORES"]]<-read.csv("TradeBalance\_ORES.csv" ,header=T)  
dframes.list[["ORES"]][,"NAME"]<-"ORES"  
  
## ALL MATERIALS - FOSSIL FUELS, BIOMASS, CONSTRUCTION MATERIALS, ORES  
dframes.list[["TOTAL"]]<-read.csv("TradeBalance\_I-CMFA-TOT.csv",header=T)  
dframes.list[["TOTAL"]][,"NAME"]<-"TOTAL"  
  
#############  
  
## I will attempt a cleaning function to prepare each data frame for analysi  
prepare.dframe.fct<-function(i){  
### cleaning data frame  
i<-i[which(as.character(i$Country)!="Former USSR"),]#!(df1$id %in% idNums1)  
i<-i[-which( i[,"y"] %in% c(1991,2000,2011)),]  
i <- i[which(i[,"y"] < 2008), ]  
i<-i[ which( i[,"Country"] %in% setdiff( i[,"Country"], Remove.country.vec) ), ]  
i<-i[ -which( i[,"Country"] %in% From.1991.vec & i[,"y"] < 1991), ]  
i<-i[ -which( i[,"Country"] == From.1993.vec & i[,"y"] < 1993), ]  
  
### calculating structure of consumption  
#### Territorial emissions as extraction  
i[,"Extraction"]<-i[,"TerritorialEmissions"]  
#### consumed resources that come from domestic extraction  
i[,"Consum.extract"]<- (i[,"Extraction"]) -   
 (i[,"Exports"] \* ((i[,"Extraction"]) /  
 (i[,"Extraction"] + i[,"Imports"])))  
#### consumed resources that come from imports  
i[,"Consum.import"]<- (i[,"Imports"]) -   
 (i[,"Exports"] \* ((i[,"Imports"]) /  
 (i[,"Extraction"] + i[,"Imports"])))  
#### consumption balance index  
i[,"Consum.balance"]<-(i[,"Consum.extract"]-i[,"Consum.import"])/i[,"Consumption"]  
i<-i[which(abs(i[,"Consum.balance"])<1),] ## removing entries that are above 1 (i.e. all resources being domestically extracted or imported)  
  
  
### associating population size and gdp data with data frame - then calculating basic metrics for later analysis  
#### merging the gdp and population size data onto the energy consumption data frame  
i<-merge(i,gdppop.df,by=c("CountryA3","y","Country"),all.x=TRUE)  
  
i <- i[order(i[,"Country"],i[,"y"]),]  
  
i[,"mean.gdp"]<-unlist(tapply(i[,"GDP"],i[,"Country"],function(x) rep(mean(x,na.rm=TRUE),length(x))))  
i[,"mean.val"]<-unlist(tapply(i[,"val"],i[,"Country"],function(x) rep(mean(x,na.rm=TRUE),length(x))))  
i[,"mean.gdp.per.cap"]<-i[,"mean.gdp"]/i[,"mean.val"]  
i[,"log10.mean.gdp.per.cap"]<-log10(i[,"mean.gdp.per.cap"])  
  
val.membership.df<-data.frame("mean.val"=unique(i[,"mean.val"]),"val.membership"=NA)  
val.membership.df[,"val.membership"]<-as.numeric(cut2(val.membership.df$mean.val,m=20))  
  
i<-merge(i,val.membership.df,by="mean.val",all.x=TRUE)  
  
  
### Calculating mean consumption balance  
### mean consumption.balance index  
i <- i[order(i[,"Country"],i[,"y"]),]  
i[,"Consum.balance.avg"]<-unlist(by(i, i[,"Country"], function(x) rep(mean(x[,"Consum.balance"]),nrow(x))))  
  
### Calculating trend in consumption balance  
i <- i[order(i[,"Country"],i[,"y"]),]  
i[,"ConsumBalance.y.trend"]<-unlist(by(i,i[,"Country"],function(x) rep(coef(lm(x[,"Consum.balance"]~x[,"y"]))[2],nrow(x))))  
  
return(i)  
}  
  
  
## apply the data frame preparation function  
dframes.list<-lapply(dframes.list,prepare.dframe.fct)  
  
dframes.list[["ENERGY"]]<-energy.df  
dframes.list[["ENERGY"]][,"NAME"]<-"ENERGY"  
  
  
## Get the calculated averages and trends for each data frame  
#ConsumTrend.df.list<-lapply(dframes.list,function(x) unique(x[,c("Country","Consum.balance.avg","ConsumBalance.y.trend","mean.val","mean.gdp","mean.gdp.per.cap","log10#.mean.gdp.per.cap","NAME")]))

### MEDIAN FOOTPRINT VALUES IN AND CHANGE FROM FIRST AND TO SECOND TIME PERIOD

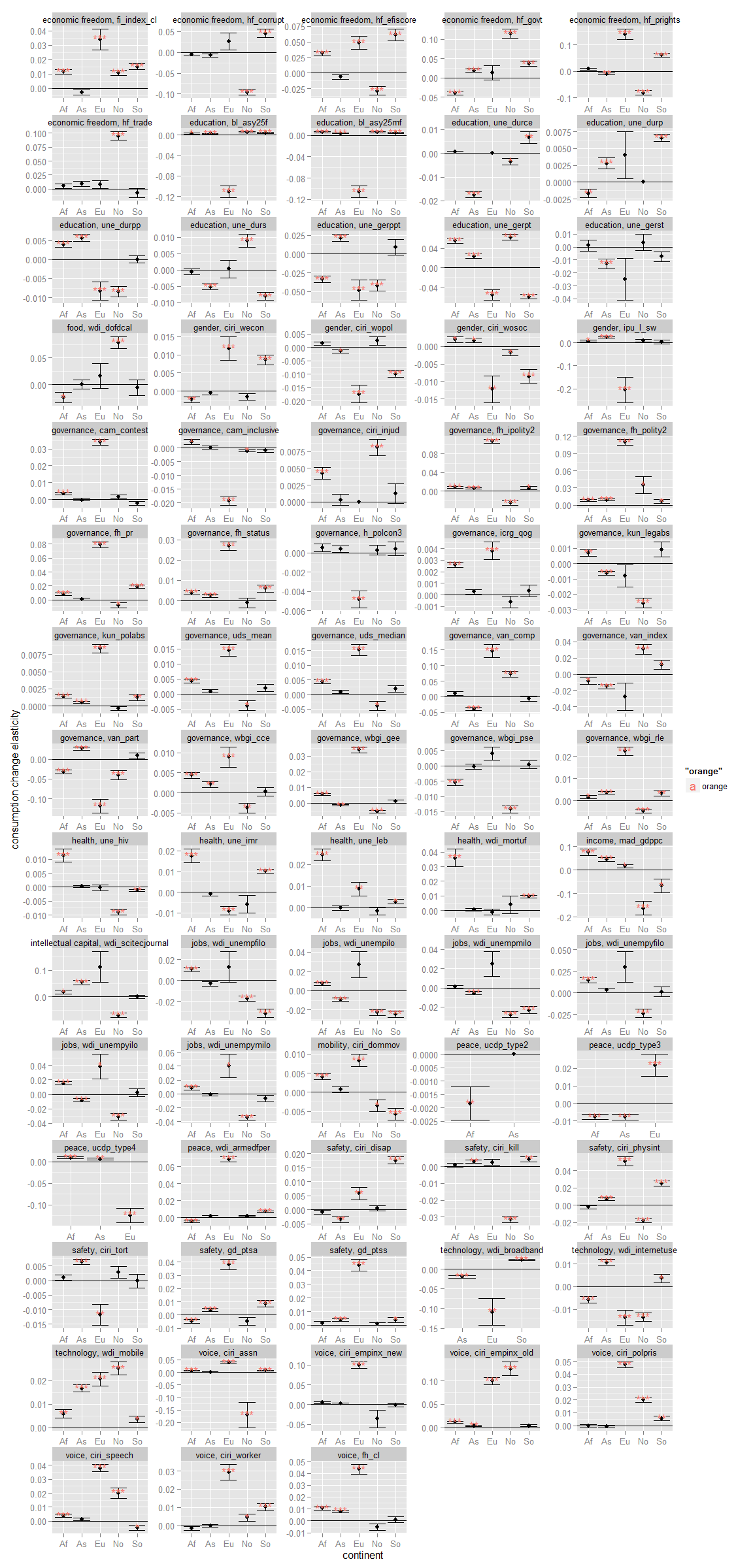
MF.df<-dframes.list[["TOTAL"]]  
  
MF.df[,"period"]<-NA  
MF.df[,"period"]<-ifelse(MF.df[,"y"]<2000,"a",ifelse(MF.df[,"y"]>1999,"b",NA))  
  
MF.vars<-c("TerritorialEmissions","Imports","Exports","DirectEmissions","Consumption","Extraction","Consum.extract","Consum.import","Consum.balance","GDP","val")  
  
MF.df<-MF.df[order(MF.df$CountryA3),]  
  
## MEDIAN FIRST PERIOD  
MF.first<-as.data.frame(apply(MF.df[which(MF.df$y %in% c(1992:1999)),MF.vars],2,function(x,y=MF.df[which(MF.df$y %in% c(1992:1999)),"CountryA3"]) tapply(x,y,function(z) median(z,na.rm=TRUE))))  
  
MF.first[,c("CountryA3","y")]<-cbind(rownames(MF.first),rep(1990,nrow(MF.first)))  
MF.first.melt<-melt(MF.first,id.var=c("CountryA3","y"))  
names(MF.first.melt)<-c("CountryA3","first.period","variable","first.value")  
  
## MEDIAN SECOND PERIOD  
MF.second<-as.data.frame(apply(MF.df[which(MF.df$y %in% c(2000:2008)),MF.vars],2,function(x,y=MF.df[which(MF.df$y %in% c(2000:2008)),"CountryA3"]) tapply(x,y,function(z) median(z,na.rm=TRUE))))  
  
MF.second[,c("CountryA3","y")]<-cbind(rownames(MF.second),rep(2000,nrow(MF.second)))  
MF.second.melt<-melt(MF.second,id.var=c("CountryA3","y"))  
names(MF.second.melt)<-c("CountryA3","second.period","variable","second.value")  
  
## MANN WHITNEY TEST  
MUtest.list<-by(MF.df,MF.df[,"CountryA3"],function(x) apply(x[,MF.vars],2,function(y,z=x[,"period"],country=x[1,"CountryA3"],  
 w1=length(which(is.na(y[which(x$y %in% c(1992:1999))])==FALSE)),  
 w2=length(which(is.na(y[which(x$y %in% c(2000:2008))])==FALSE))) #print(c(w1,w2))  
 if(w1 > 2 & w2 > 2){c(unlist(kruskal.test(y, factor(z)))[1:3],country)} else {c(rep(NA,3),country)}#pairwise.wilcox.test  
 ))  
MUtest.list <- MUtest.list[lapply(MUtest.list,length)>0]  
  
MUtest.df<-as.data.frame(t(do.call(cbind,lapply(MUtest.list,function(x) rbind(names(as.data.frame(x)),x)  
 ))))  
  
  
names(MUtest.df)<-c("varcode","KW.chisq","KW.dfres","KW.pval","CountryA3")  
  
  
###############################  
  
  
  
## MERGING MEDIANS OF FIRST AND SECOND PERIOD AND CALCULATING DIFFERENCE IN MEDIAN VALUE  
MF.melt<-merge(MF.first.melt,MF.second.melt,by=c("CountryA3","variable"),all=TRUE)  
MF.melt[,"diff.value"]<-MF.melt[,"second.value"]-MF.melt[,"first.value"]  
  
## MERGING RESULTS FROM KRUSKAL WALLIS TESTS  
MF.melt<-merge(MF.melt,MUtest.df,by.x=c("CountryA3","variable"),by.y=c("CountryA3","varcode"),all.x=TRUE)

### MERGING MATERIAL FOOTPRINT DATA TO SOCIAL VARIABLES AND CALCULATING A CROSS-COUNTRY ELASTICITY OF SORTS BY GEOGRAPHICAL AND INCOME REGOINS AND PLOTTING IT

## MERGING MF CONSUMPTION DATA AND  
#MF.melt<-merge(MF.melt,MUtest.df,by.x=c("CountryA3","variable"),by.y=c("CountryA3","varcode"),all.x=TRUE)  
#QoG.sd.melt.plot<-QoG.sd.melt[which(is.na(QoG.sd.melt[,"continent"])==FALSE & is.na(QoG.sd.melt[,"income.region"])==FALSE),]  
  
MF.melt.consum<-MF.melt[which(MF.melt[,"variable"]=="Consumption"),]  
  
names(MF.melt.consum)[-c(1:2)]<-paste("consum",c("first.period","first.value","second.period","second.value","diff.value","KW.chisq","KW.dfres","KW.pval"),sep=".")  
  
QoG.MF.df<-merge(QoG.sd.melt.plot,MF.melt.consum,by.x="ccodealp",by.y="CountryA3",all.x=TRUE)  
  
QoG.MF.df.sub<-QoG.MF.df[complete.cases(QoG.MF.df[,c("diff.value","first.value","consum.first.value","consum.diff.value")]),]  
  
## applying log modulus transformation  
QoG.MF.df.sub[,c("diff.value.logmod","first.value.logmod","consum.first.value.logmod","consum.diff.value.logmod")]<-apply(  
 QoG.MF.df.sub[,c("diff.value","first.value","consum.first.value","consum.diff.value")],2, function(x) sign(x)\*log(abs(x)+1))  
  
  
## calculatig partial regression coefficients of initial social value, initial consumption value and change in consumption value on change in social value by income region  
QoG.MF.region.df<-as.data.frame(do.call(rbind,  
   
 by(QoG.MF.df.sub,QoG.MF.df.sub[,"variable.x"], function(x)   
 do.call(rbind,  
   
 by(x,list(x[,"income.region4"]), function(y,z=summary(lm(y[,"diff.value.logmod"]~y[,"first.value.logmod"]+y[,"consum.first.value.logmod"]+y[,"consum.diff.value.logmod"]))[["coefficients"]])   
   
 if(nrow(z)==4){   
 cbind(  
 matrix(c("Int","first.value","consum.first.value","consum.diff.value",rep(c(as.character(y[1,"income.region4"]),as.character(y[1,"variable.x"]),as.character(y[1,"social"])),each=4)),byrow=FALSE,ncol=4)  
 ,z)  
 }  
   
 )))))  
  
  
names(QoG.MF.region.df)<-c("parameter","income.region4","variable.x","social","est","se","tval","pval")  
  
  
## calculatig partial regression coefficients of initial social value, initial consumption value and change in consumption value on change in social value by continent  
QoG.MF.continent.df<-as.data.frame(do.call(rbind,  
   
 by(QoG.MF.df.sub,QoG.MF.df.sub[,"variable.x"], function(x)   
 do.call(rbind,  
   
 by(x,list(x[,"continent2"]), function(y,z=summary(lm(y[,"diff.value.logmod"]~y[,"first.value.logmod"]+y[,"consum.first.value.logmod"]+y[,"consum.diff.value.logmod"]))[["coefficients"]])   
   
 if(nrow(z)==4){   
 cbind(  
 matrix(c("Int","first.value","consum.first.value","consum.diff.value",rep(c(as.character(y[1,"continent2"]),as.character(y[1,"variable.x"]),as.character(y[1,"social"])),each=4)),byrow=FALSE,ncol=4)  
 ,z)  
 }  
   
 )))))  
  
  
names(QoG.MF.continent.df)<-c("parameter","continent2","variable.x","social","est","se","tval","pval")  
  
  
## plotting partial regression coefficients of change in consumption with s.e. as error bars  
  
QoG.MF.continent.df[,"social"]<-gsub("connectivity","technology",QoG.MF.continent.df[,"social"])  
QoG.MF.region.df[,"social"]<-gsub("connectivity","technology",QoG.MF.region.df[,"social"])  
  
  
QoG.MF.continent.df[,c("est","se","tval","pval")] <- apply(QoG.MF.continent.df[,c("est","se","tval","pval")],2,as.numeric)  
QoG.MF.region.df[,c("est","se","tval","pval")] <- apply(QoG.MF.region.df[,c("est","se","tval","pval")],2,as.numeric)  
###  
QoG.MF.continent.df[,"pval.sign"]<-ifelse(QoG.MF.continent.df[,"pval"]<0.001,"\*\*\*",ifelse(QoG.MF.continent.df[,"pval"]<0.01 & QoG.MF.continent.df[,"pval"]>0.001,"\*\*",  
 ifelse(QoG.MF.continent.df[,"pval"]<0.05 & QoG.MF.continent.df[,"pval"]>0.01,"\*","")))  
###  
QoG.MF.region.df[,"pval.sign"]<-ifelse(QoG.MF.region.df[,"pval"]<0.001,"\*\*\*",ifelse(QoG.MF.region.df[,"pval"]<0.01 & QoG.MF.region.df[,"pval"]>0.001,"\*\*",  
 ifelse(QoG.MF.region.df[,"pval"]<0.05 & QoG.MF.region.df[,"pval"]>0.01,"\*","")))  
  
#by income region  
#x11()  
  
ggplot(QoG.MF.region.df[which(QoG.MF.region.df[,"parameter"]=="consum.diff.value"),],  
 aes(x=factor(income.region4,levels=c("High","Uppe","Lowe","Low ")),y=est))+geom\_point()+ geom\_errorbar(aes(ymax = est + se, ymin= est - se)) +  
 facet\_wrap(~social+variable.x,scales="free",ncol=5)+ylab("consumption change elasticity")+xlab("income region")+geom\_hline(x=0) +  
 geom\_text(aes(y=est+(abs(est)\*0.1),label = pval.sign, color="orange"))



#by continent  
#x11()  
  
#limits <- aes(ymax = est + se, ymin=est - se)  
  
ggplot(QoG.MF.continent.df[which(QoG.MF.continent.df[,"parameter"]=="consum.diff.value" & QoG.MF.continent.df[,"continent2"]!="Au"),],  
 aes(x=factor(continent2,levels=c("Af","As","Au","Eu","No","So")),y=est))+geom\_point()+ geom\_errorbar(aes(ymax = est + se, ymin= est - se)) +  
 facet\_wrap(~social+variable.x,scales="free",ncol=5)+ylab("consumption change elasticity")+xlab("continent")+geom\_hline(x=0) +  
 geom\_text(aes(y=est,label = pval.sign, color="orange"))



##### next try using CovOgk(matrix(rnorm(100),byrow=TRUE,ncol=2)) to find a robus estimate of location and covariance

# REFERENCES

The following literature was cited