

```
In [22]: library(magrittr)
library(dplyr)
library(reshape)
library(countrycode)
```

## Dataset THOMSON REUTERS

Importo la lista del dataset costituito da matrici le cui righe rappresentano i direct investor (DI) residenti in un paese e le cui colonne rappresentano le imprese di diretto investimento (DIENT) residenti in un altro paese. Importo anche lista dei paese e degli anni.

```
In [30]: list_of_files <- list.files(path = "C:dataset/", recursive = TRUE,
                                   pattern = "\\.",
                                   full.names = TRUE)
paesi <- read.csv("nodes.txt",
                 sep=",", header=FALSE)
paesi_dataset <- c(paesi$V1)

year <- read.csv("years.txt",
                 sep=",", header=FALSE)
year <- c(year$V1)

print(paste0("ANNO ",year[1]))
dat <- read.table(list_of_files[1],sep=',', header=FALSE)
colnames(dat) <- rownames(dat) <- paesi_dataset
head(dat)
```

[1] "ANNO 1995"

A data.frame: 6 × 195

	ABW	AFG	AGO	AIA	ALB	AND	ANT	ARE	ARG	ARM	...	VI
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	...	<db
<b>ABW</b>	0	0	0	0	0	0	0	0	0	0	...	
<b>AFG</b>	0	0	0	0	0	0	0	0	0	0	...	
<b>AGO</b>	0	0	0	0	0	0	0	0	0	0	...	
<b>AIA</b>	0	0	0	0	0	0	0	0	0	0	...	
<b>ALB</b>	0	0	0	0	0	0	0	0	0	0	...	
<b>AND</b>	0	0	0	0	0	0	0	0	0	0	...	

```
In [25]: full_dat <- data.frame()
#metto i dati in panel data
for(i in 1:length(year)){
  dat<-read.table(list_of_files[i],sep=',', header=FALSE)
  new_dat <- cbind(rep(paesi_dataset, each = length(paesi_dataset)),
                  rep(paesi_dataset,length(paesi_dataset)),
                  rep(year[i],length(paesi_dataset)),c(t(dat)))
```

```

colnames(new_dat) <- c("origin","destination","year","value_ts")
new_dat <- as.data.frame(new_dat)
new_new_data <- new_dat %>%
  filter(origin != destination)
full_dat <- rbind(full_dat,new_new_data)
}

full_dat$value_ts <- as.numeric(full_dat$value_ts)

```

Utilizzo la funzione della library "countrycode" per trasformare i nomi dei paesi in codici ISO3c. Non vengono riconosciuti dei paesi che non sono scritti in modo corretto: li inserisco a mano; altri paesi, invece, van tolti, in quanto si riferiscono a entità geografiche non più esistenti in quei confini. Successivamente, sostituisco i nomi dei paesi presenti nel dataset con i rispettivi ISO3c così determinati.

```

In [26]: paesi_giusti <- countrycode(paesi_dataset,origin="iso3c",destination="iso3c")
paesi <- as.data.frame(cbind(paesi_dataset,paesi_giusti))
#Some values were not matched unambiguously: ANT, PAL, ROM, TMP, YUG, ZAR
paesi$paesi_giusti[which(paesi_dataset == "PAL")] <- "PSE"
paesi$paesi_giusti[which(paesi_dataset == "ROM")] <- "ROU"
paesi$paesi_giusti[which(paesi_dataset == "TMP")] <- "TLS"

full_dat$origin <- paesi$paesi_giusti
  [match(as.factor(full_dat$origin),
        as.factor(paesi$paesi_dataset))]
full_dat$destination <- paesi$paesi_giusti[
  match(as.factor(full_dat$destination),
        as.factor(paesi$paesi_dataset))]
#rimuovo le righe con i NA dei paesi non considerati
fdi_ts <- na.omit(full_dat)

```

Warning message:

"Some values were not matched unambiguously: ANT, PAL, ROM, TMP, YUG, ZAR  
"

Creo un dataset completo con tutte le possibili combinazioni di *origin*, *destination* e *year* (tra quelli presenti nel dataset originale).

```

In [27]: p <- as.character(na.omit(paesi$paesi_giusti))
y <- unique(fdi_ts$year)
tot <- as.data.frame(cbind(rep(p, each = length(p)*length(y)),
                          rep(rep(p,times=length(p)),each=length(y)),
                          rep(y, times = length(p)^2)))
colnames(tot) <- c("origin","destination","year")
tot <- tot[-c(which(tot$origin == tot$destination)),]
tot <- cbind(tot,rep(0,nrow(tot)))
colnames(tot) <- c("origin","destination","year","value_ts")
tot$year <- as.numeric(tot$year)
merge <- rbind(fdi_ts,tot)
fdi_ts_tot <- merge %>%
  group_by(origin,destination,year) %>%
  summarise(value_ts = sum(value_ts))

```

``summarise()`` has grouped output by 'origin', 'destination'. You can override using the `` .groups`` argument.

```
In [28]: fdi_ts_tot <- fdi_ts_tot[order(fdi_ts_tot$year),]  
write.csv(fdi_ts_tot, "fdi_ts.csv", row.names = FALSE)
```