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

DATASET MPI

Importo il dataset sul flusso di migrazione scolastica attraverso i dati bibliometrici su larga scala di OpenAlex e Scopus.

In [1]:
```
openalex <-  read.csv("openalex_scholarlymigration.csv",
                          sep=",", header=TRUE)
scopus <-  read.csv("scopus_scholarlymigration.csv",
                          sep=",", header=TRUE)
```

In [2]:
```
print(paste0("I paesi presente in OpenAlex sono ",
             length(unique(openalex$migrationfrom))))
print(paste0("I paesi presente in Scopus sono ",
             length(unique(scopus$migrationfrom))))
```

```
[1] "I paesi presente in OpenAlex sono 219"
[1] "I paesi presente in Scopus sono 228"
```

Decido di tenere il dataset di Scopus che considera più paesi. Seleziono la seconda colonna che corrisponde al paese di origine, la terza colonna che corrisponde al paese di destinazione, la quarta che corrisponde all'anno dell'osservazione e la quinta che corrisponde al numero di studenti che emigrano. Successivamente verifico che il codice ISO3c utilizzato nel dataset per i paesi siano un codice esistente anche per la libreria countrycode di R che ho utilizzato per tutti gli altri dataset.

In [5]:
```
head(scopus[,1:6])
head(scopus[,7:10])
colnames(scopus)
```

A data.frame: 6 × 6

	X	migrationfrom	migrationto	year	n_migrations	n_migrations_back
	<int>	<chr>	<chr>	<dbl>	<int>	<int>
1	0	ITA	ESP	2010	190	89
2	1	USA	MEX	2014	166	186
3	2	BRA	AUS	2013	34	18
4	3	RUS	DEU	2010	126	70
5	4	TUN	FRA	2009	79	92
6	5	FRA	DEU	2010	324	265

A data.frame: 6 × 4

	netmigrations	avgauthorage	avgauthorage_back	avgyears_in_previous_country
	<int>	<dbl>	<dbl>	<dbl>
1	101	6.336842	6.022472	4.726316
2	-20	7.457831	6.150538	4.620482
3	16	6.147059	6.888889	5.617647
4	56	7.055556	8.200000	5.341270
5	-13	4.341772	5.380435	3.405063
6	59	5.623457	6.396226	3.962963

'X' · 'migrationfrom' · 'migrationto' · 'year' · 'n_migrations' · 'n_migrations_back' · 'netmigrations' · 'avgauthorage' · 'avgauthorage_back' · 'avgyears_in_previous_country' · 'avgyears_in_previous_country_back'

In [30]:
```
full_data <- scopus
full_data <- full_data[,2:5]

colnames(full_data) <- c("origin","destination", "year","n_migrations")

paesi_nomi <- c(unique(full_data$origin),
                setdiff(unique(full_data$destination),unique(full_data$origin)))
paesi_iso3c <- countrycode(paesi_nomi,origin="iso3c",destination="iso3c")
```

```
Warning message:
"Some values were not matched unambiguously: , EUE
"
```

L'unico codice che non risulta giusto è quello associata all'Unione Europea. Elimino i dati il cui paese di origine o di destinazione è quest'ultimo. Elimino anche i dati relativi all'anno 2024 e 2025 (?) e le righe dove la casella del paese di origine o di destinazione è vuota.

In [75]:
```
mpi <- full_data %>%
  group_by(origin,destination,year) %>%
  summarize(value = sum(n_migrations))

mpi <- mpi[-which(mpi$origin == "EUE" | mpi$destination == "EUE"),]
mpi <- mpi[-which(mpi$year == 2024 | mpi$year == 2025),]
mpi <- mpi[-which(mpi$origin == "" & mpi$destination == ""),]
```

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

Creo dataset con tutte le combinazioni possibili tra origin, destination e year (tra quelli presenti in quello originale).

In [76]:
```
p <- as.character(na.omit(paesi_iso3c))
y <- c(1994:2023)
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) <- colnames(mpi)
tot$year <- as.numeric(tot$year)
merge <- rbind(mpi,tot)
mpi_tot <- merge %>%
  group_by(origin,destination,year) %>%
  summarise(value = sum(value))
```

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

In [78]:
```
mpi_tot <- mpi_tot[order(mpi_tot$year),]
write.csv(mpi_tot,"mpi.csv", row.names = FALSE)
```