

```
In [ ]: library(readxl)
library(dplyr)
library(zoo)
library("stringr")
library(reshape)
library(countrycode)
```

## Dataset BIS

Importo dataset.

```
In [ ]: full_data <- read.csv("WS_LBS_D_PUB_csv_col.csv",
                             sep=";", header=TRUE)
```

```
In [5]: head(full_data[,1:5])
head(full_data[,6:10])
head(full_data[,10:15])
head(full_data[,30:35])
```

A data.frame: 6 × 5

	FREQ	Frequency	L_MEASURE	Measure	L_POSITION
	<chr>	<chr>	<chr>	<chr>	<chr>
1	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C
2	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C
3	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C
4	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C
5	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C
6	Q	Quarterly	F	FX and break adjusted change (BIS calculated)	C

A data.frame: 6 × 5

	Balance.sheet.position	L_INSTR	Type.of.instruments	L_DENOM	Currency.denomination
	<chr>	<chr>	<chr>	<chr>	<chr>
1	Total claims	A	All instruments	CHF	Swiss franc
2	Total claims	A	All instruments	CHF	Swiss franc
3	Total claims	A	All instruments	CHF	Swiss franc
4	Total claims	A	All instruments	CHF	Swiss franc
5	Total claims	A	All instruments	CHF	Swiss franc
6	Total claims	A	All instruments	CHF	Swiss franc

A data.frame: 6 × 6

	Currency.denomination	L_CURR_TYPE	Currency.type.of.reporting.country	L_PARENT_CTY
	<chr>	<chr>	<chr>	<chr>
1	Swiss franc	A	All currencies (=D+F+U)	5J
2	Swiss franc	A	All currencies (=D+F+U)	5J
3	Swiss franc	A	All currencies (=D+F+U)	5J
4	Swiss franc	A	All currencies (=D+F+U)	5J
5	Swiss franc	A	All currencies (=D+F+U)	5J
6	Swiss franc	A	All currencies (=D+F+U)	5J

A data.frame: 6 × 6

	Organisation.visibility	Series	X1977.Q4	X1978.Q1	X1978.Q2	X1978.
	<lgl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	NA	Q:F:C:A:CHF:A:5J:A:5A:A:TD:N	NA	NA	NA	
2	NA	Q:F:C:A:CHF:A:5J:A:5A:B:TD:N	NA	NA	NA	
3	NA	Q:F:C:A:CHF:A:5J:A:5A:C:TD:N	NA	NA	NA	
4	NA	Q:F:C:A:CHF:A:5J:A:5A:H:TD:N	NA	NA	NA	
5	NA	Q:F:C:A:CHF:A:5J:A:5A:N:TD:N	NA	NA	NA	
6	NA	Q:F:C:A:CHF:A:5J:A:5A:P:TD:N	NA	NA	NA	

Ci sono diverse variabili.

```
In [20]: colnames(full_data)[1:31]
```

```
'FREQ' · 'Frequency' · 'L_MEASURE' · 'Measure' · 'L_POSITION' · 'Balance.sheet.position' ·  
'L_INSTR' · 'Type.of.instruments' · 'L_DENOM' · 'Currency.denomination' · 'L_CURR_TYPE' ·  
'Currency.type.of.reporting.country' · 'L_PARENT_CTY' · 'Parent.country' · 'L_REP_BANK_TYPE' ·  
'Type.of.reporting.institutions' · 'L_REP_CTY' · 'Reporting.country' · 'L_CP_SECTOR' ·  
'Counterparty.sector' · 'L_CP_COUNTRY' · 'Counterparty.country' · 'L_POS_TYPE' · 'Position.type' ·  
'TIME_FORMAT' · 'Time.Format' · 'COLLECTION' · 'Collection.Indicator' · 'ORG_VISIBILITY' ·  
'Organisation.visibility' · 'Series'
```

Seleziono tutti i tipi di strumenti (debt securities long and short term, loans and deposits); tolgo la colonna sulla frequenza temporale (quadrimestrale per tutti i dati); seleziono i dati che prendono in considerazione i flussi finanziari in ogni valuta (poi convertita in USD); considero il singolo paese che dichiara non l'insieme dei paesi; con ogni tipo di valuta, e da ogni tipo di istituzione; anche il debitore può essere qualunque tipo di istituzione; considero solo trasferimenti da paese a paese; considero il singolo paese che riceve o emette il finanziamento non l'insieme dei paesi o le organizzazioni internazionali.

```
In [3]: clean<-full_data[which(full_data$Type.of.instruments=="All instruments"),]  
clean<-clean[,-c(1,2)]  
clean<-clean[which(clean$Currency.denomination=="All currencies"),]  
clean<-clean[which(clean$Reporting.country!="All reporting countries"),]  
clean<-clean[which(clean$Currency.type.of.reporting.country==  
"All currencies (=D+F+U)"),]  
clean<-clean[which(clean$Type.of.reporting.institutions==  
"All reporting banks/institutions  
(domestic, foreign, consortium and unclassified)"),]  
clean<-clean[which(clean$Counterparty.sector=="All sectors"),]  
#qui l'opzione era tra: "FX and break adjusted change (BIS calculated)" e  
"Amounts outstanding / Stocks"  
clean<-clean[which(clean$Measure=="Amounts outstanding / Stocks"),]  
clean<-clean[which(clean$Position.type=="Cross-border"),]  
clean<-clean[which(clean$Counterparty.country!="All countries (total)" &  
clean$Counterparty.country!="International organisations"),]
```

Dopo aver selezionato le righe che mi interessano tolgo le colonne non più utili.

```
In [4]: clean<-subset(clean, select = -c(Type.of.instruments, Currency.denomination,  
Parent.country,  
Currency.type.of.reporting.country,  
Type.of.reporting.institutions,  
Counterparty.sector,Measure,Position.type))  
clean<-subset(clean, select = -c(TIME_FORMAT, Time.Format, COLLECTION,  
Collection.Indicator,ORG_VISIBILITY,  
Organisation.visibility))  
clean<-subset(clean, select = -c(L_MEASURE, L_POSITION, L_INSTR,  
L_DENOM,L_CURR_TYPE,  
L_PARENT_CTY, L_REP_BANK_TYPE,L_CP_SECTOR))  
clean<-subset(clean, select = -c(Series, L_POS_TYPE))
```

Considero separatamente i crediti e i debiti dichiarati.

```
In [5]: tot_claims<-clean[which(clean$Balance.sheet.position=="Total claims"),-1]
tot_liab<-clean[which(clean$Balance.sheet.position=="Total liabilities"),-1]
#han dimensioni diverse:
print(paste0("Il numero delle osservazioni dei debiti è: ", nrow(tot_liab)))
print(paste0("Il numero delle osservazioni dei credito è: ", nrow(tot_claims)))
```

```
[1] "Il numero delle osservazioni dei debiti è: 5928"
```

```
[1] "Il numero delle osservazioni dei credito è: 5469"
```

Preparo per la fusione dei dati per scrivere il dataset in formato panel.

```
In [6]: new_claims<-as.data.frame(tot_claims)
new_claims[is.na(new_claims)] <- 0

new_liab<-as.data.frame(tot_liab)
new_liab[is.na(new_liab)] <- 0

## cambio la dicitura delle date così
colnames(new_liab)<-sub("X", "", colnames(new_liab))
colnames(new_liab)<-sub("\\\\.", "_", colnames(new_liab))

colnames(new_claims) <- sub("X", "", colnames(new_claims))
colnames(new_claims)<-sub("\\\\.", "_", colnames(new_claims))

### melt
resh_tot_liab <- melt(new_liab, id=c(1,2,3,4))
resh_tot_claims <- melt(new_claims, id=c(1,2,3,4))
colnames(resh_tot_claims)[6] <- "value_claim"
colnames(resh_tot_liab)[6] <- "value_liab"
```

## Liabilities (debiti)

Passo da una frequenza quadrimestrale a una frequenza annuale.

```
In [7]: new_liab <- resh_tot_liab %>%
  group_by(year = sub('_', '*', variable), L_REP_CTY, L_CP_COUNTRY)%>%
  summarise(value_liab = mean(value_liab))

colnames(new_liab) <- c("year", "origin", "destination", "value_liab")
new_liab <- new_liab %>%
  select(origin, destination, year, value_liab)
```

`summarise()` has grouped output by 'year', 'L\_REP\_CTY'. You can override using the `groups` argument.

Converto i nomi dei paesi da ISO2c a ISO3c usando la funzione countrycode di R. Escludo quelli che si riferiscono a entità geografiche non più esistenti o troppo generali: 0, 1W, 1Z, AN, C9, CS, DD, PU, SU, YU. Successivamente, cambio il paese creditore e debitore con il corrispondente ISO3c così determinato.

```
In [8]: paesi_dataset <- c(unique(new_liab$origin),setdiff(
  unique(new_liab$destination),unique(new_liab$origin)))
paesi_giusti <- countrycode(paesi_dataset,origin="iso2c",destination="iso3c")
paesi <- as.data.frame(cbind(paesi_dataset,paesi_giusti))

new_liab$origin <- paesi$paesi_giusti[match(as.factor(new_liab$origin),
  as.factor(paesi$paesi_dataset))]
new_liab$destination <- paesi$paesi_giusti[match(as.factor(new_liab$destination),
  as.factor(paesi$paesi_dataset))]

#rimuovo le righe con i NA dei paesi non considerati
new_liab<-na.omit(new_liab)
```

Warning message:

"Some values were not matched unambiguously: 0, 1W, 1Z, AN, C9, CS, DD, PU, SU, YU  
"

## Claims (crediti)

Applico lo stesso procedimento.

```
In [9]: new_claims <- resh_tot_claims %>%
  group_by(year = sub('_', '*', variable), L_REP_CTY, L_CP_COUNTRY)%>%
  summarise(value_claim = mean(value_claim))

colnames(new_claims) <- c("year", "origin", "destination", "value_claim")
new_claims <- new_claims %>%
  select(origin, destination, year, value_claim)

new_claims$origin <- paesi$paesi_giusti[match(as.factor(new_claims$origin),
  as.factor(paesi$paesi_dataset))]
new_claims$destination <- paesi$paesi_giusti[match(
  as.factor(new_claims$destination),
  as.factor(paesi$paesi_dataset))]

new_claims<-na.omit(new_claims)
```

`summarise()` has grouped output by 'year', 'L\_REP\_CTY'. You can override using the  
`.groups` argument.

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

```
In [10]: p <- as.character(na.omit(paesi$paesi_giusti))
y <- unique(new_liab$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_liab")
merge <- rbind(new_liab, tot)
new_liab_tot <- merge %>%
  group_by(origin, destination, year) %>%
```

```

summarise(value_liab = sum(value_liab))
new_liab_tot <- new_liab_tot[order(new_liab_tot$year),]

colnames(tot) <- c("origin","destination","year","value_claim")
merge <- rbind(new_claims,tot)
new_claim_tot <- merge %>%
  group_by(origin,destination,year) %>%
  summarise(value_claim = sum(value_claim))
new_claim_tot <- new_claim_tot[order(new_claim_tot$year),]

```

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

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

Posso unire ora i due dataset: ora riportano gli stessi collegamenti.

```
In [11]: bis <- as.data.frame(cbind(new_claim_tot,new_liab_tot[,4]))
```

## Valori negativi

```
In [12]: print(paste0("Ci sono ", length(which(bis$claim_bis < 0)),
  " valori negativi nel caso dei dati sui crediti e ",
  length(which(bis$claim_liab < 0)),
  " valori negativi nel caso dei dati sui debiti: li elimino"))
bis[which(bis$claim_bis < 0 | bis$claim_liab < 0),c(4:5)] <- 0
```

[1] "Ci sono 56 valori negativi nel caso dei dati sui crediti e 92 valori negativi nel caso dei dati sui debiti: li elimino"

## Da creditore a debitore

Modifico il dataset sui debiti (liab) in modo che l'origin sia il paese creditore e la destination sia il paese debitore, anche se è quest'ultimo il paese dichiarante.

```
In [23]: bis_inv <- bis %>%
  select(destination,origin,year,claim_derived_bis = liab_bis)
colnames(bis_inv)[1:2] <- colnames(bis)[1:2]
bis_inv <- bis_inv[order(bis_inv$origin),]
bis_confronto <- cbind(bis_inv[order(bis_inv$year),],bis$claim_bis)
colnames(bis_confronto) <- c("origin","destination","year","claim_derived_bis", "claim_bis")
```

Prendo i dati che sono non negativi per entrambi: proviamo a confrontarli. Il valore che riporta il paese debitore, quando è quello dichiarante, è sempre minore al valore che riporta il paese creditore; e viceversa.

```
In [24]: sovr_bis <- which(bis_confronto$claim_derived_bis != 0 & bis_confronto$claim_bis != 0)
print(paste0("Il ",
  round(length(sovr_bis)/length(which(bis_confronto$claim_bis != 0))*100,
  "% dei dati sui crediti, con valori diversi da 0 per il creditore,
  son anche diversi da 0 per il debitore"))
print(paste0("Il ",
```

```

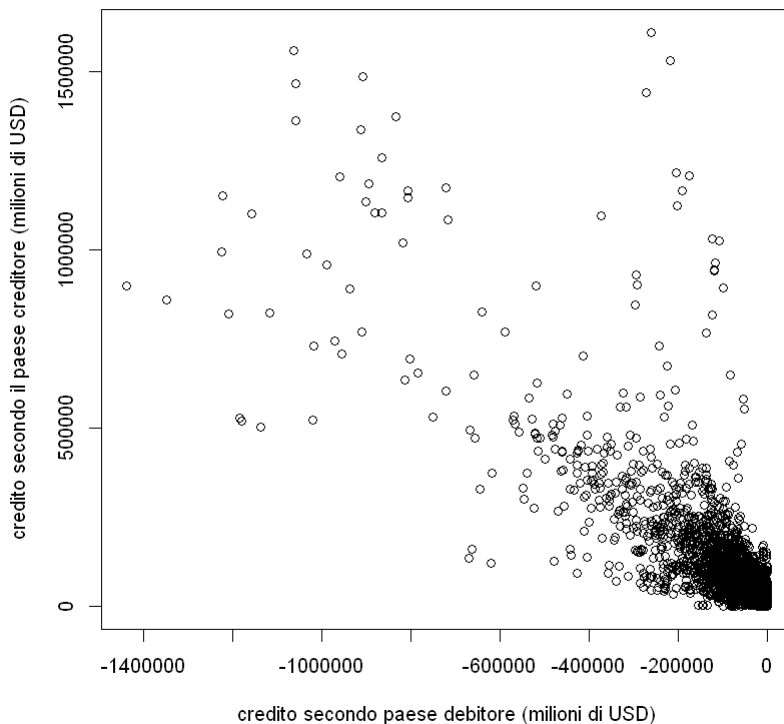
round(length(sovr_bis)/length(which(bis_confronto$claim_derived_bis !=
"% dei dati sui crediti, con valori diversi da 0 per il debitore,
son anche diversi da 0 per il creditore"))
data_bis <- data.frame(y = bis_confronto$claim_bis[sovr_bis],
                      x = (-1)*bis_confronto$claim_derived_bis[sovr_bis])
plot(data_bis$x,data_bis$y,
     xlab = "credito secondo paese debitore (milioni di USD)",
     ylab = "credito secondo il paese creditore (milioni di USD)",
     main = "Confronto tra valori del credito a seconda del paese dichiarante")
plot(data_bis$x,data_bis$y,
     xlab = "credito secondo paese debitore (milioni di USD)",
     ylab = "credito secondo il paese creditore (milioni di USD)",
     main = "Confronto tra valori del credito a seconda del paese dichiarante",
     xlim=c(-200000,0), ylim = c(0,200000))
plot(data_bis$x,data_bis$y,
     xlab = "credito secondo paese debitore (milioni di USD)",
     ylab = "credito secondo il paese creditore (milioni di USD)",
     main = "Confronto tra valori del credito a seconda del paese dichiarante",
     xlim=c(-10000,0), ylim = c(0,10000))
plot(data_bis$x,data_bis$y,
     xlab = "credito secondo paese debitore (milioni di USD)",
     ylab = "credito secondo il paese creditore (milioni di USD)",
     main = "Confronto tra valori del credito a seconda del paese dichiarante",
     xlim=c(-100,0), ylim = c(0,100))

```

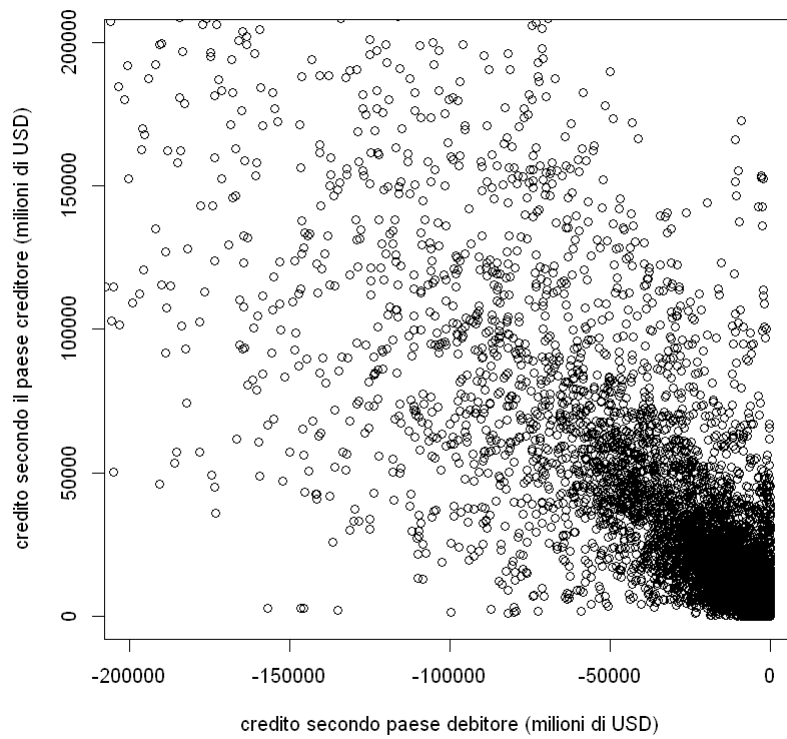
[1] "Il 17.1% dei dati sui crediti, con valori diversi da 0 per il creditore, son anche diversi da 0 per il debitore"

[1] "Il 15% dei dati sui crediti, con valori diversi da 0 per il debitore, son anche diversi da 0 per il creditore"

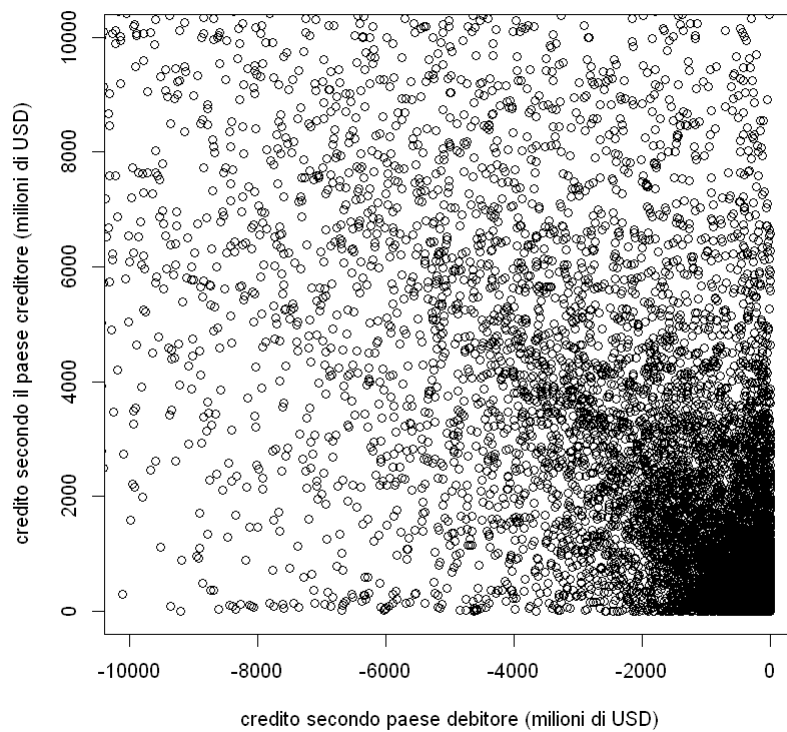
#### Confronto tra valori del credito a seconda del paese dichiarante



**Confronto tra valori del credito a seconda del paese dichiarante**

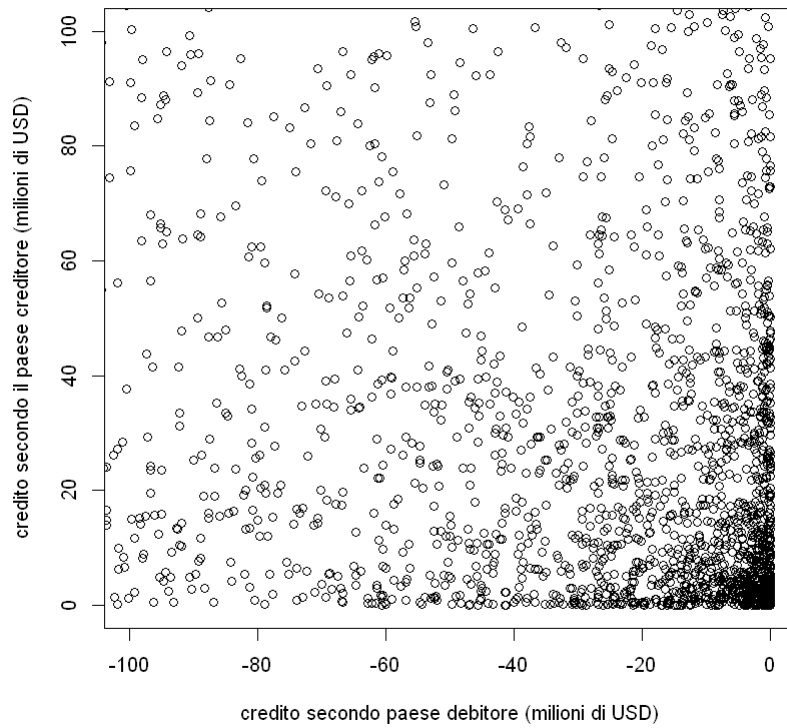


**Confronto tra valori del credito a seconda del paese dichiarante**





### Confronto tra valori del credito a seconda del paese dichiarante



```
In [26]: write.csv(bis_confronto,"bis_fromcredto debt.csv", row.names = FALSE)
write.csv(bis,"bis.csv", row.names = FALSE)
```