

# Pondérations du panel ELIPSS - Guide utilisateur

Blazej Palat

▶ **To cite this version:**

Blazej Palat. Pondérations du panel ELIPSS - Guide utilisateur. 2021. hal-03999484

**HAL Id: hal-03999484**
**https://sciencespo.hal.science/hal-03999484**

Submitted on 21 Feb 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Pondérations du panel ELIPSS

**Guide utilisateur**

## Pondérations mises à disposition dans le jeu de données

### Poids des répondants à l'enquête – *seconde phase de correction de la non-réponse totale*

***PDSPLT1_[NOM DE l'ENQUETE]***

Panel « pilote » (recrutement 2012). A utiliser principalement pour analyser les données de l'enquête avec appariemments de variables provenant d'enquêtes réalisées avant le premier rafraîchissement (2016). Une telle analyse porterait forcément sur un groupe de participants à l'enquête limité aux panélistes recrutés en 2012...

***PDSPLT2_[NOM DE l'ENQUETE]***

Panel rafraîchi en 2016. A utiliser principalement pour analyser les données de l'enquête avec appariemments de variables provenant d'enquêtes réalisées avant le rafraîchissement de 2020.

***POIDS_[NOM DE l'ENQUETE]***

Panel rafraîchi en 2020. A utiliser pour l'enquête en question y compris avec appariemment de variables d'enquêtes réalisées après 2020.

Les poids correspondant à la pondération enquête servent à pondérer les résultats. *PDSPLT1_* et *PDSPLT2_* contiennent des valeurs manquantes (panélistes recrutés lors de vagues de rafraîchissement postérieures, non concernés par ces pondérations).

### Poids initiaux – *première phase de correction de la non-réponse totale*

***PDSPLT_INIT1***

Panel « pilote » (recrutement 2012).

***PDSPLT_INIT2***

Panel rafraîchi en 2016.

***POIDS_INIT***

Panel rafraîchi en 2020.

Les poids initiaux ne servent pas à pondérer les résultats. Ils sont fournis pour les utilisateurs qui souhaitent réaliser eux-mêmes les calages (et donc prendre en charge eux-mêmes la seconde phase de correction de la non-réponse). Le code `R` (R Core Team, 2021), faisant appel aux fonctionnalités du package `Icarus` (Rebecq, 2016), utile dans ce but peut être trouvé dans la section *Calage avec `Icarus`* du présent guide. Comme dans le cas des poids propres à l'enquête, les poids *PDSPLT_* contiennent des valeurs manquantes.

# Variables de calage [1]

## A utiliser avec les poids initiaux *PDSPLT_*

***cal_AGE1***

Age décennal, à inclure dans les calages des poids *PDSPLT_INIT1*.

***cal_AGE2***

Idem, à inclure dans les calages des poids *PDSPLT_INIT2*.

***cal_DIPL2***

Niveau de diplôme

***cal_NAT***

Nationalité

## A utiliser avec les poids initiaux *POIDS_*

***cal_AGE***

Age décennal

***cal_DIPL***

Niveau de diplôme

***cal_TUU***

Tranche d'unité urbaine

## A utiliser dans tous les cas

***cal_SEXE***

Sexe

***cal_ZEAT***

Zone d'étude et d'aménagement du territoire

1. Pour le codage des modalités, voir la documentation de l'enquête.

# Exemples de calage avec **Icarus**

## Quelques fonctions utiles

### `tri_levels()` – Ordonner les modalités des variables de calage

```
tri_levels = function(variable){
  return(factor(variable
                ,levels(as.factor(variable))[order(levels(as.factor(variable)))]))
}
```

### `statsDescrPoids()` – Statistiques descriptives pour évaluer la dispersion des poids

```
# Rapport max/min
maxMin<-function(x){
    return(max(x,na.rm=T)/min(x,na.rm=T))
  }
# Rapport Q3/Q1
q3q1<-function(x){
    return(quantile(x,0.75,na.rm=T)/quantile(x,0.25,na.rm=T))
  }
# Rapport C95/C5
c95c5<-function(x){
    return(quantile(x,0.95,na.rm=T)/quantile(x,0.05,na.rm=T))
  }

# Tableau avec statistiques descriptives
statsDescrPoids<-function(x){
  if(
    (length(table(is.na(x)))>1 & names(table(is.na(x)))[2]=="TRUE") |
    (length(table(is.na(x)))==1 & names(table(is.na(x)))[1]=="TRUE")
  ) warning("NAs in x !")
  return(
  matrix(
    c(
  round(mean(x,na.rm=T),3)
  ,round(sd(x,na.rm=T),3)
  ,round(median(x,na.rm=T),3)
  ,round(q3q1(x),3)
  ,round(c95c5(x),3)
  ,round(min(x,na.rm=T),3)
  ,round(max(x,na.rm=T),3)
  ,round(maxMin(x),3)
      )
        ,nrow=8,ncol=1
        ,dimnames=list(c("Moy.","EC","Med."
                         ,"Rapp. Q3/Q1","Rapp. C95/C5"
                         ,"Min.","Max.","Rapp. max./min.")
                       ,NULL)
        )
        )
                              }
```

### Calage des poids *PDSPLT*

```
# Dans l'exemple ci-dessous, "data" est le jeu de données,
# objet de classe "data.frame"

# Marges issues de l'EAR 2014
MARGES_EAR.2014 <- rbind(
c("cal_AGE1","6","0.1099","0.162","0.175","0.1956","0.172","0.1855","0","0"),
#c("cal_AGE2","7","0.0806","0.1914","0.175","0.1956","0.172","0.1464", "0.039", "0"),
#commentez la première ligne [#c("cal_AGE1",...] et décommentez la ligne ci-dessus
#[c("cal_AGE2",...] pour caler les poids PDSPLT_INIT2
c("cal_DIPL2","4","0.2783","0.2335","0.3364","0.1518","0","0","0","0"),
c("cal_SEXE","2","0.4902","0.5098","0","0","0","0","0","0"),
c("cal_NAT","3","0.8828","0.0564","0.0608","0","0","0","0","0"),
c("cal_ZEAT","8","0.1886","0.1632","0.0642","0.0862","0.1385","0.1142","0.1217"
  ,"0.1234"))

# Total de la population estimé, ou total de l'échantillon suivant l'utilisation
# de la pondération
N = nrow(data) # soit 45300000 pour les poids non normalisés

# Ordonner les modalités des facteurs
for (i in c("cal_AGE2","cal_DIPL2","cal_SEXE","cal_NAT","cal_ZEAT"))
  {data[,i] = tri_levels(data[,i])}

# Calage sur marges - création d'un vecteur de poids calés "poids"
poids <- calibration(
data = data,
colWeights = "PDSPLT_INIT1", # mettez "PDSPLT_INIT2" à la place pour
                             # caler les poids des répondants recrutés avant 2020
marginMatrix = MARGES_EAR.2014,
pct = TRUE,
popTotal = N,
method = "raking")

# pour obtenir quelques statistiques descriptives afin d'évaluer la dispersion
statsDescrPoids(poids)
```

### Calage des poids *POIDS*

```
# Dans l'exemple ci-dessous, "data" est le jeu de donnée,
# objet de classe "data.frame"

# marges EEC 2019
MARGES_EEC.2019 <- rbind(
c("cal_AGE","7","0.115659690667959","0.150702768262258"
  ,"0.159322593353643","0.171946583904539","0.160767332277733","0.137167898613123"
  ,"0.104433132920745","0"),
c("cal_DIPL","4","0.191872078218975","0.316098600118058"
  ,"0.287584662627322","0.204444659035645","0","0","0","0"),
c("cal_SEXE","2","0.476515516960587","0.523484483039413","0","0","0","0","0","0"),
c("cal_TUU","5","0.235947681307464","0.169952391173349","0.117779311542026"
  ,"0.310656810283793","0.165663805693368","0","0","0"),
c("cal_ZEAT","8","0.186362889148176","0.16597386630283","0.0614300521959146"
  ,"0.0849281638313224","0.138529351473008","0.114843273759216","0.124035058216417"
  ,"0.123897345073116")
)

# Total de la population estimé, ou total de l'échantillon suivant l'utilisation
# de la pondération
N = nrow(data) # soit 49748338 pour les poids non normalisés
```

```
# Ordonner les modalités des facteurs
for (i in c("cal_AGE","cal_DIPL","cal_SEXE","cal_TUU","cal_ZEAT"))
  {data[,i] = tri_levels(data[,i])}

# Calage sur marges - création d'un vecteur de poids calés "poids"
poids <- calibration(
data = data,
colWeights = "POIDS_INIT",
marginMatrix = MARGES_EEC.2019,
pct = TRUE,
popTotal = N,
method = "raking"
)

# pour obtenir quelques statistiques descriptives afin d'évaluer la dispersion
statsDescrPoids(poids)
```

*

* *

**Qu'il s'agisse de poids enquête, fournis avec le jeu de données, ou de poids calés par les soins de l'utilisateur à partir des poids initiaux, nous recommandons vivement de surveiller la dispersion des poids ainsi que, si cela est jugé approprié, d'effectuer des troncatures (selon le cas, à droite et/ou à gauche de la distribution des poids).**

# Références

R Core Team. (2021). *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/

Rebecq, A. (2016, octobre). *Icarus, un package R pour le calage sur marges et ses variantes*. 9e Colloque International Francophone sur les Sondages, Gatineau, Canada. http://paperssondages16.sfds.asso.fr/submission_54.pdf