

Fiche TD avec le logiciel  : tdr25

---

## Courbes de granulométrie

D. Chessel, A.B. Dufour & J.R. Lobry

---

Ecrire une fonction utilitaire

### Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Implanter un jeu d'essai</b>	<b>2</b>
<b>3</b>	<b>Définir l'objectif de la fonction</b>	<b>3</b>
<b>4</b>	<b>Vérifier les résultats</b>	<b>4</b>

### 1 Introduction

L'existence du biométricien est parsemée de message du genre :

Bonjour,  
à toutes fins utiles et si cela a une quelconque utilité, je signale que les versions PC et Mac (ADE4\_2001) de l'option "RowRepartF" du module EcolTool ne donnent pas les mêmes résultats. Je joins le jeu de données sur lequel j'ai fait le test. Peut-être y a-t-il l'équivalent dans R ?

A plus,

Le but de l'exercice est de refaire une fonction R pour pallier le désagrément de l'utilisateur sympathique. La documentation dit :

### EcolTools : RowRepartF

Édition d'informations élémentaires sur un tableau de distribution de poids par lignes.

Dans un fichier binaire, on possède n (lignes) distributions de poids en p (colonnes) catégories. L'exemple type est celui des courbes granulométriques. On a de plus dans une autre fichier à p+1 lignes et 1 colonne les bornes des classes définissant les catégories (par exemple 9 classes granulométriques et 10 bornes millimétriques, .3, .5, 1, 2, 4, 8, 16, 32, 64, 128). Le programme fabrique 3 fichiers :

- .hist qui contient par lignes les distributions de fréquences (somme 1 par ligne) ;
- .cumu qui contient les fonctions de répartition cumulée par ligne ;
- .para qui contient 8 paramètres descriptifs élémentaires des distributions à savoir 5 percentiles (.05, .25=1<sup>o</sup> quartile, .50=médiane, .75=3<sup>o</sup> quartile, et .95), l'intervalle inter-quartile, la moyenne, la variance et l'écart-type.

L'option utilise une seule fenêtre de dialogue :

RowRepartF

Input file (See info)

Input file (See info)

Quit

OK

- Nom du fichier d'entrée binaire des distributions.
- Nom du fichier d'entrée binaire des bornes des classes.

Utiliser la carte Granulo de la pile ADE•Data.

```

N°|alpha .05|alpha .25|alpha .50|alpha .75|alpha .95|a.75-a.25| mean | variance|
---|-----|-----|-----|-----|-----|-----|-----|-----|
 1| 7.487e+00|1.719e+01|2.798e+01|4.433e+01|6.007e+01|2.714e+01|3.074e+01|2.316e+02|
 2| 4.457e+00|1.138e+01|2.276e+01|4.376e+01|1.009e+02|3.238e+01|3.196e+01|7.555e+02|
 3| 5.220e+00|1.142e+01|2.046e+01|2.899e+01|5.867e+01|1.758e+01|2.381e+01|3.028e+02|
***
48| 6.714e+00|1.596e+01|2.459e+01|3.595e+01|5.839e+01|1.999e+01|2.739e+01|2.022e+02|
49| 6.344e+00|1.311e+01|2.139e+01|3.002e+01|5.570e+01|1.691e+01|2.399e+01|1.758e+02|
-----
Result stored in output file: Granulo.para
ROWS: 49 Col: 8

```

## 2 Implanter un jeu d'essai

Lire dans R les fichiers de référence `granu.txt` et `granuborn.txt`. Les données sont issues de Gaschignard-Fossati, O. 1986. *Répartition spatiale des macroinvertébrés benthiques d'un bras vif du Rhône. Rôle des crues et dynamique saisonnière*. Thèse de doctorat, Université Lyon 1.

```
granuborn <- readLines("http://pbil.univ-lyon1.fr/R/donnees/granuborn.txt")
(granuborn <- as.numeric(granuborn))
```

```
[1] 0.3 0.5 1.0 2.0 4.0 8.0 16.0 32.0 64.0 128.0
```

```
granu <- read.table("http://pbil.univ-lyon1.fr/R/donnees/granu.txt")
granu[c(1:3, 25:27), ]
```

```

V1 V2 V3 V4 V5 V6 V7 V8 V9
1  0  2  1 15 260 810 1815 1990  0
2  0  0 12 340 1220 2095 2950 2050 1160

```

```

3  0  0  0  12 1505 1960 4415 1265 260
25 0  1 25 170 230 260 1250 1250 1760
26 0  1  6 310 240 1920 2725 2260 395
27 0  0  1 120 2510 5020 9350 2630  0

```

Chaque ligne est un prélèvement faisant l'objet d'une étude granulométrique au laboratoire. L'échantillon passe sur une colonne de tamis et le poids (en grammes) de chaque refus de tamis est enregistré. Les mailles des tamis définissent les classes de diamètres dans lesquelles les grains sont répartis. Dans le premier échantillon 1815 grammes de particules de sédiments ont été enregistrés dans la classe 16 mm - 32 mm (graviers grossiers).

### 3 Définir l'objectif de la fonction

La fonction doit lire le tableau des poids et les bornes des classes et renvoyer une liste qui contiendra :

\$hist le tableau des fréquences des poids par ligne (somme 1 par ligne)

```
granu[1, ]
```

```

V1 V2 V3 V4 V5 V6 V7 V8 V9
1  0  2  1 15 260 810 1815 1990  0

```

```
w <- unlist(granu[1, ])
w
```

```

V1 V2 V3 V4 V5 V6 V7 V8 V9
0  2  1 15 260 810 1815 1990  0

```

```
w/sum(w)
```

```

V1 V2 V3 V4 V5 V6
0.0000000000 0.0004087472 0.0002043736 0.0030656039 0.0531371347 0.1655426119
V7 V8 V9
0.3709380748 0.4067034539 0.0000000000

```

\$cumu le tableau des fréquences cumulées qui a autant de colonnes que de bornes

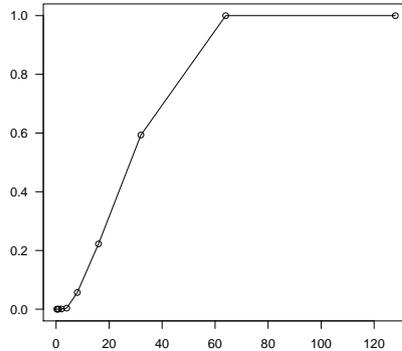
```
w1 <- c(0, cumsum(w/sum(w)))
w1
```

```

V1 V2 V3 V4 V5
0.0000000000 0.0000000000 0.0004087472 0.0006131208 0.0036787247 0.0568158594
V6 V7 V8 V9
0.2223584713 0.5932965461 1.0000000000 1.0000000000

```

```
plot(granuborn, w1, type = "o", las = 1, xlab = "", ylab = "")
```



\$para le tableau des paramètres de chaque profil granulométrique. On veut la moyenne, la variance et l'écart-type des centres des classes pondérés par le poids de grain dans chaque classe, les quantiles 0.05, 0.25, 0.50, 0.75, 0.95 et la longueur de l'intervalle inter-quartile. Le quartile alpha est la valeur en x (la plus petite si il y en a plusieurs) pour laquelle la courbe de poids cumulée (ci-dessus) vaut alpha. L'intervalle est la différence entre le quartile 0.75 et le quartile 0.25.

\$born les bornes des classes.

\$cent les centres des classes.

## 4 Vérifier les résultats

Quand la fonction est au point, vous devez obtenir les résultats suivants :

```
options(digits = 4)
rowrepartf(granu[1:3, ], granuborn)

$hist
 Cla1 Cla2 Cla3 Cla4 Cla5 Cla6 Cla7 Cla8 Cla9
1 0 0.0004087 0.0002044 0.003066 0.05314 0.1655 0.3709 0.4067 0.00000
2 0 0.0000000 0.0012211 0.034599 0.12415 0.2132 0.3002 0.2086 0.11804
3 0 0.0000000 0.0000000 0.001274 0.15982 0.2081 0.4688 0.1343 0.02761

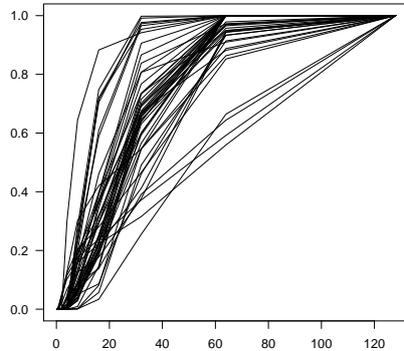
$cumu
 Born0 Born1 Born2 Born3 Born4 Born5 Born6 Born7 Born8 Born9
1 0 0 0.0004087 0.0006131 0.003679 0.05682 0.2224 0.5933 1.0000 1
2 0 0 0.0000000 0.0012211 0.035820 0.15997 0.3732 0.6733 0.8820 1
3 0 0 0.0000000 0.0000000 0.001274 0.16109 0.3692 0.8381 0.9724 1

$para
 moy var et a0.05 a0.25 a0.5 a0.75 a0.95 iq
1 30.74 231.6 15.22 7.487 17.19 27.98 44.33 60.07 27.14
2 31.96 755.5 27.49 4.457 11.38 22.76 43.76 100.89 32.38
3 23.81 302.8 17.40 5.220 11.42 20.46 28.99 58.67 17.58

$born
 [1] 0.3 0.5 1.0 2.0 4.0 8.0 16.0 32.0 64.0 128.0

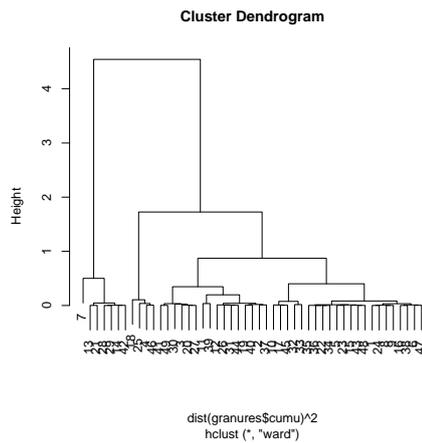
$cent
 [1] 0.40 0.75 1.50 3.00 6.00 12.00 24.00 48.00 96.00
```

Vérifier enfin le résultat global (*test and test and test again*), par exemple tracer toutes les courbes cumulées :



Plus tard, faire l'ACP centrée du tableau `$hist`, faire une classification sur le tableau `$cumu` et reporter la classification sur la carte factorielle de l'ACP :

```
plot(hc1 <- hclust(dist(granures$cumu)^2, "ward"))
```



```
fac <- factor(cutree(hc1, 4))
library(ade4)
d1 <- dudi.pca(granures$hist, scal = F, scan = F)
scatter(d1)
s.class(d1$li, fac, add.p = T, cpoi = 0)
```

