Chapitre 6 Premiers traitements statistiques

6.1 La fonction `summary`

La fonction `summary` peut s'appliquer à une table entière, et, dans ce cas, donner les statistiques principales sur chacun des variables, en s'adaptant au type de celles-ci (numerique ou texte). On peut également les appliquer à un vecteur seul.

```
library (tidyverse)
base <- read.csv ("Data/Base_synth_territoires.csv",
                  header = T, sep=";", dec=",")
base_extrait <- select (base, 1, 3, 5, 7:12)
summary (base_extrait)
```

```
##      CODGEO           REG
##  01001  :    1   Min.   : 1.00
##  01002  :    1   1st Qu.:28.00
##  01004  :    1   Median :44.00
##  01005  :    1   Mean   :52.05
##  01006  :    1   3rd Qu.:76.00
##  01007  :    1   Max.   :94.00
##  (Other):36683
##                                                ZAU           P14_P(
##  112 - Couronne d'un grand pôle                  :12297   Min.   :
##  400 - Commune isolée hors influence des pôles  : 7383   1st Qu.:
##  300 - Autre commune multipolarisée              : 7021   Median :
##  120 - Multipolarisée des grandes aires urbaines: 3962   Mean   :
##  111 - Grand pôle (plus de 10 000 emplois)       : 3285   3rd Qu.:
##  221 - Petit pôle (de 1 500 à 5 000 emplois)     :  888   Max.   :2
##  (Other)                                         : 1853   NA's   :8
##     P09_POP             SUPERF             NAIS0914
##  Min.   :      0   Min.   :    0.04   Min.   :     0.0
##  1st Qu.:    193   1st Qu.:    6.44   1st Qu.:     9.0
##  Median :    431   Median :   10.81   Median :    23.0
##  Mean   :   1793   Mean   :   17.64   Mean   :   114.4
##  3rd Qu.:   1072   3rd Qu.:   18.58   3rd Qu.:    60.0
##  Max.   :2234105   Max.   :18360.00   Max.   :150843.0
##  NA's   :821       NA's   :821        NA's   :821
##     DECE0914           P14_MEN
##  Min.   :    0.00   Min.   :      0.0
##  1st Qu.:    8.00   1st Qu.:     83.8
##  Median :   17.00   Median :    183.2
##  Mean   :   77.35   Mean   :    802.0
##  3rd Qu.:   43.00   3rd Qu.:    454.9
##  Max.   :69907.00   Max.   :1147990.9
##  NA's   :821        NA's   :821
```

- Les variables quantitatives

```
summary (pull (base_extrait, NAIS0914))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
##      0.0      9.0     23.0    114.4     60.0 150843.0      821
```

- Les variables qualitatives

```
summary (pull (base_extrait, ZAU))
```

```
##       111 - Grand pôle (plus de 10 000 emplois)
##                                            3285
##                  112 - Couronne d'un grand pôle
##                                           12297
## 120 - Multipolarisée des grandes aires urbaines
##                                            3962
##       211 - Moyen pôle (5 000 à 10 000 emplois)
##                                             456
##                  212 - Couronne d'un moyen pôle
##                                             815
##     221 - Petit pôle (de 1 500 à 5 000 emplois)
##                                             888
##                  222 - Couronne d'un petit pôle
##                                             582
##              300 - Autre commune multipolarisée
##                                            7021
##   400 - Commune isolée hors influence des pôles
##                                            7383
```

6.2 Calculer des statistiques spécifiques

```
sum (pull (base_extrait, P14_POP), na.rm = T)
```

```
## [1] 65907160
```

```
mean (pull (base_extrait, P14_POP), na.rm = T)
```

```
## [1] 1837.492
```

```
median (pull (base_extrait, P14_POP), na.rm = T)
```

```
## [1] 444
```

```
quantile (pull (base_extrait, P14_POP), probs = c (.25, .5, .75), na.
```

```
##  25%  50%  75%
##  197  444 1110
```

6.3 Agréger des données selon un facteur

La fonction `summarise()` couplée à `group_by()` permet de calculer des statistiques pour chaque modalité d'une variable qualitative. Avec `group_by()`, on précise les variables qui formeront des groupes, sur lesquels on appliquera une fonction :

```
TableauGroupes <- group_by (TableEnEntree, Variable1, ..., VariableN)
summarise (TableauGroupes, NomVariableAgregee = Fonction (NomVariable
```

Par exemple, si on veut avoir la médiane de la variable P14_POP, pour chaque ZAU et chaque région :

```
base_reg_ann <- group_by (base_extrait, ZAU, REG) %>%
  summarise (population_med = median (P14_POP, na.rm = T))
```

6.4 Une autre manière de sélectionner une variable : $

On peut aussi accéder aux variables d'un data frame grâce au symbole `$` :

```
Dataframe$Variable
```

Si on veut sélectionner la variable ZAU, on peut donc écrire, de manière équivalente :

```
pull (base_extrait, ZAU) # méthode "tidyverse"
base_extrait$ZAU # méthode "base"
```

6.5 Tableau de contingence

La fonction table calcule les effectifs d'un tableau croisé :

```
t <- table (base_extrait$ZAU, base_extrait$REG)
print (t)
```

```
##
##                                                1    2    3
##   111 - Grand pôle (plus de 10 000 emplois)   17   16    3
##   112 - Couronne d'un grand pôle               6    0    3
##   120 - Multipolarisée des grandes aires urbaines  1    4    0
##   211 - Moyen pôle (5 000 à 10 000 emplois)    0    3    2
##   212 - Couronne d'un moyen pôle               0    0    0
##   221 - Petit pôle (de 1 500 à 5 000 emplois)  3    2    1
##   222 - Couronne d'un petit pôle               0    0    0
##   300 - Autre commune multipolarisée           2    4    0
##   400 - Commune isolée hors influence des pôles  3    5   13
##
##                                               24   27   28
##   111 - Grand pôle (plus de 10 000 emplois)  103  140  216
##   112 - Couronne d'un grand pôle             734 1299 1126 1
##   120 - Multipolarisée des grandes aires urbaines  188  336  488
##   211 - Moyen pôle (5 000 à 10 000 emplois)   30   31   34
##   212 - Couronne d'un moyen pôle              72  122  104
##   221 - Petit pôle (de 1 500 à 5 000 emplois)  46   60   78
##   222 - Couronne d'un petit pôle              19  137   29
##   300 - Autre commune multipolarisée         375  737  762
##   400 - Commune isolée hors influence des pôles  275  969  396
##
##                                               52   53   75
##   111 - Grand pôle (plus de 10 000 emplois)  108   89  333
##   112 - Couronne d'un grand pôle             535  415 1161 1
##   120 - Multipolarisée des grandes aires urbaines  186  152  254
##   211 - Moyen pôle (5 000 à 10 000 emplois)   23   47   51
##   212 - Couronne d'un moyen pôle              44   18   81
##   221 - Petit pôle (de 1 500 à 5 000 emplois)  69   42  163
##   222 - Couronne d'un petit pôle               8    1   81
##   300 - Autre commune multipolarisée         386  325 1002
##   400 - Commune isolée hors influence des pôles  143  181 1379 1
##
##                                               93   94
##   111 - Grand pôle (plus de 10 000 emplois)  220    8
##   112 - Couronne d'un grand pôle             229   99
##   120 - Multipolarisée des grandes aires urbaines  69    3
##   211 - Moyen pôle (5 000 à 10 000 emplois)   20    1
##   212 - Couronne d'un moyen pôle              11    2
##   221 - Petit pôle (de 1 500 à 5 000 emplois)  32   13
##   222 - Couronne d'un petit pôle               6   29
##   300 - Autre commune multipolarisée         107   53
##   400 - Commune isolée hors influence des pôles  269  152
```

6.6 Tableau de proportions

La fonction `prop.table` prend en entrée un objet `table` (tableau de contingence avec les effectifs) et calcule les pourcentages (total, ligne, colonne) associés → `?prop.table`

```
round (100 * prop.table (t), digits = 1)
```

```
##
##                                                 1   2   3   4
##   111 - Grand pôle (plus de 10 000 emplois)       0.0 0.0 0.0 0.0
##   112 - Couronne d'un grand pôle                  0.0 0.0 0.0 0.0
##   120 - Multipolarisée des grandes aires urbaines 0.0 0.0 0.0 0.0
##   211 - Moyen pôle (5 000 à 10 000 emplois)       0.0 0.0 0.0 0.0
##   212 - Couronne d'un moyen pôle                  0.0 0.0 0.0 0.0
##   221 - Petit pôle (de 1 500 à 5 000 emplois)     0.0 0.0 0.0 0.0
##   222 - Couronne d'un petit pôle                  0.0 0.0 0.0 0.0
##   300 - Autre commune multipolarisée              0.0 0.0 0.0 0.0
##   400 - Commune isolée hors influence des pôles   0.0 0.0 0.0 0.0
##
##                                                  27  28  32  44
##   111 - Grand pôle (plus de 10 000 emplois)       0.4 0.6 1.3 0.9
##   112 - Couronne d'un grand pôle                  3.5 3.1 4.1 4.7
##   120 - Multipolarisée des grandes aires urbaines 0.9 1.3 2.0 2.2
##   211 - Moyen pôle (5 000 à 10 000 emplois)       0.1 0.1 0.1 0.1
##   212 - Couronne d'un moyen pôle                  0.3 0.3 0.0 0.3
##   221 - Petit pôle (de 1 500 à 5 000 emplois)     0.2 0.2 0.1 0.2
##   222 - Couronne d'un petit pôle                  0.4 0.1 0.0 0.3
##   300 - Autre commune multipolarisée              2.0 2.1 1.9 3.1
##   400 - Commune isolée hors influence des pôles   2.6 1.1 0.8 2.2
##
##                                                  75  76  84  93
##   111 - Grand pôle (plus de 10 000 emplois)       0.9 0.7 1.5 0.6
##   112 - Couronne d'un grand pôle                  3.2 3.1 4.0 0.6
##   120 - Multipolarisée des grandes aires urbaines 0.7 0.9 1.1 0.2
##   211 - Moyen pôle (5 000 à 10 000 emplois)       0.1 0.2 0.1 0.1
##   212 - Couronne d'un moyen pôle                  0.2 0.4 0.2 0.0
##   221 - Petit pôle (de 1 500 à 5 000 emplois)     0.4 0.3 0.3 0.1
##   222 - Couronne d'un petit pôle                  0.2 0.2 0.1 0.0
##   300 - Autre commune multipolarisée              2.7 2.4 1.4 0.3
##   400 - Commune isolée hors influence des pôles   3.8 4.2 2.6 0.7
```

```
print (chisq.test (t))
```

```
##
##  Pearson's Chi-squared test
##
## data:  t
## X-squared = 6100.4, df = 128, p-value < 2.2e-16
```

6.7 Exercice : calcul de statistiques

- Utilisez la fonction `summary` pour obtenir un résumé de l'ensemble des variables de la table df
- Calculez maintenant les moyenne, médiane, écart-type et variance de la variable de densité de population. Que constatez-vous ?
- Utilisez le paramètre `na.rm=T` pour gérer les valeurs manquantes
- Calulez à présent les quartiles puis déciles de cette variables
- Calculez la version centrée réduite de la variable de densité. Rappel : on calcule la version centrée réduite d'une variable X en lui appliquant la transformation suivante :

$$STD_X = \frac{X - \bar{X}}{\sigma_X}$$

où $\bar{X}$ est la moyenne empirique de X et σ_X son écart-type

Tableaux croisés :

- Calculer le nombre de communes par type d'espace à l'aide de la fonction `table` , et le pourcentage associé
- Calculer le nombre de communes par région et type d'espace, et les pourcentages associés

```
df <- base %>%
  select (1:24) %>%
  mutate (densite = P14_POP / SUPERF,
          tx_natal = 1000 * NAISD15 / P14_POP,
          tx_mort = DECESD15 / P14_POP,
          ZAU2 = as.factor (substr (ZAU, 1, 3))) # Parce que la varia

summary (df)
```

```
##      CODGEO                LIBGEO          REG             DEP
##  01001  :    1   Sainte-Colombe:   13   Min.   : 1.00   62     :
##  01002  :    1   Beaulieu      :   11   1st Qu.:28.00   02     :
##  01004  :    1   Saint-Sauveur :   11   Median :44.00   80     :
##  01005  :    1   Sainte-Marie  :   11   Mean   :52.05   76     :
##  01006  :    1   Le Pin        :   10   3rd Qu.:76.00   57     :
##  01007  :    1   Saint-Aubin   :   10   Max.   :94.00   14     :
##  (Other):36683   (Other)       :36623                   (Other):32
##                                                ZAU
##  112 - Couronne d'un grand pôle                :12297
##  400 - Commune isolée hors influence des pôles  : 7383
##  300 - Autre commune multipolarisée             : 7021
##  120 - Multipolarisée des grandes aires urbaines: 3962
##  111 - Grand pôle (plus de 10 000 emplois)      : 3285
##  221 - Petit pôle (de 1 500 à 5 000 emplois)    :  888
##  (Other)                                        : 1853
##                        ZE           P14_POP           P09_POP
##  0061 - Toulouse        :  717   Min.   :      0   Min.   :      0
##  2307 - Rouen           :  501   1st Qu.:    197   1st Qu.:    193
##  2210 - Amiens          :  479   Median :    444   Median :    431
##  7310 - Tarbes - Lourdes:  455   Mean   :   1838   Mean   :   1793
##  2102 - Troyes          :  452   3rd Qu.:   1110   3rd Qu.:   1072
##  2603 - Dijon           :  448   Max.   :2220445   Max.   :2234105
##  (Other)                :33637   NA's   :821       NA's   :821
##      SUPERF            NAIS0914          DECE0914
##  Min.   :    0.04   Min.   :     0.0   Min.   :    0.00
##  1st Qu.:    6.44   1st Qu.:     9.0   1st Qu.:    8.00
##  Median :   10.81   Median :    23.0   Median :   17.00
##  Mean   :   17.64   Mean   :   114.4   Mean   :   77.35
##  3rd Qu.:   18.58   3rd Qu.:    60.0   3rd Qu.:   43.00
##  Max.   :18360.00   Max.   :150843.0   Max.   :69907.00
##  NA's   :821        NA's   :821        NA's   :821
##     P14_MEN             NAISD15           DECESD15
##  Min.   :      0.0   Min.   :    0.00   Min.   :    0.00
##  1st Qu.:     83.8   1st Qu.:    1.00   1st Qu.:    1.00
##  Median :    183.2   Median :    4.00   Median :    3.00
##  Mean   :    802.0   Mean   :   21.96   Mean   :   16.47
##  3rd Qu.:    454.9   3rd Qu.:   11.00   3rd Qu.:    9.00
##  Max.   :1147990.9   Max.   :28267.00   Max.   :13997.00
##  NA's   :821         NA's   :821        NA's   :821
##     P14_LOG              P14_RP          P14_RSECOCC
##  Min.   :      0.0   Min.   :      0.0   Min.   :     0.00
##  1st Qu.:    115.0   1st Qu.:     83.8   1st Qu.:     7.00
##  Median :    239.1   Median :    183.2   Median :    19.00
##  Mean   :    970.2   Mean   :    802.0   Mean   :    91.63
##  3rd Qu.:    565.0   3rd Qu.:    454.9   3rd Qu.:    49.29
```

```
##  Max.   :1362181.9   Max.   :1147990.9   Max.   :107061.99
##  NA's   :821         NA's   :821         NA's   :821
##    P14_LOGVAC         P14_RP_PROP        NBMENFISC13
##  Min.   :     0.00   Min.   :     0.0   Min.   :     32.0
##  1st Qu.:     8.00   1st Qu.:    68.4   1st Qu.:    102.0
##  Median :    18.00   Median :   148.0   Median :    205.0
##  Mean   :    76.60   Mean   :   462.2   Mean   :    809.7
##  3rd Qu.:    43.75   3rd Qu.:   349.2   3rd Qu.:    484.0
##  Max.   :107129.02   Max.   :381934.3   Max.   :1038789.0
##  NA's   :821         NA's   :821        NA's   :3793
##      PIMP13          MED13           TP6013        P14_EMPLT
##  Min.   :24.46   Min.   :10021   Min.   : 5.00   Min.   :      0.(
##  1st Qu.:50.57   1st Qu.:18452   1st Qu.: 8.73   1st Qu.:     26.(
##  Median :58.40   Median :19844   Median :11.97   Median :     66.8
##  Mean   :58.98   Mean   :20250   Mean   :13.35   Mean   :    733.9
##  3rd Qu.:67.19   3rd Qu.:21563   3rd Qu.:16.80   3rd Qu.:    229.5
##  Max.   :89.38   Max.   :46251   Max.   :44.84   Max.   :1801865.8
##  NA's   :31598   NA's   :3793    NA's   :32531   NA's   :821
##     densite            tx_natal          tx_mort            ZAU2
##  Min.   :    0.00   Min.   :  0.000   Min.   :0.0000   112    :122
##  1st Qu.:   18.59   1st Qu.:  5.679   1st Qu.:0.0044   400    : 73
##  Median :   40.35   Median :  9.264   Median :0.0079   300    : 76
##  Mean   :  160.15   Mean   :  9.699   Mean   :0.0093   120    : 39
##  3rd Qu.:   94.57   3rd Qu.: 12.931   3rd Qu.:0.0124   111    : 32
##  Max.   :27126.14   Max.   :111.111   Max.   :0.1577   221    :  8
##  NA's   :821        NA's   :827       NA's   :827      (Other): 18
```

```
mean (df$densite)
sd (df$densite)
median (df$densite)
var (df$densite)
```

On a des NA car les valeurs manquantes sont absorbantes !

```
mean (df$densite, na.rm = T)
sd (df$densite, na.rm = T)
median (df$densite, na.rm = T)
var (df$densite, na.rm = T)
```

```
df <- mutate (df, std_dens = (densite - mean (densite, na.rm = T)) /
mean (df$std_dens, na.rm = T)
```

```
## [1] 9.482279e-18
```

```
sd (df$std_dens, na.rm = T)
```

```
## [1] 1
```

Avantage des variables centrées réduites : on élimine les effets d'unité (d'ordre de grandeur), et on peut donc comparer les distributions de deux variables qui ont des unités différentes (voir module 3)

```
quantile (df$densite, na.rm = T)
```

```
##          0%         25%         50%         75%        100%
##     0.00000    18.59047    40.35457    94.57430 27126.14108
```

```
quantile (df$densite,probs = seq (0,1,.1), na.rm = T)
```

```
##          0%         10%         20%         30%         40%
##     0.00000    10.03439    15.65357    21.84208    29.76144    40.
##         60%         70%         80%         90%        100%
##    54.82089    77.65199   119.08740   240.40789 27126.14108
```

```
t <- table(df$ZAU2)
t
```

```
##
##   111   112   120   211   212   221   222   300   400
##  3285 12297  3962   456   815   888   582  7021  7383
```

```
100 * prop.table(t) %>% round(digits = 4)
```

```
##
##   111   112   120   211   212   221   222   300   400
##  8.95 33.52 10.80  1.24  2.22  2.42  1.59 19.14 20.12
```

- Deux variables

```
t <- table (df$REG, df$ZAU2)
t
```

```
##
##       111  112  120  211  212  221  222  300  400
##   1    17    6    1    0    0    3    0    2    3
##   2    16    0    4    3    0    2    0    4    5
##   3     3    3    0    2    0    1    0    0   13
##   4    10    3    4    2    0    0    0    1    4
##   11  413  853    3    3    2    0    0    7    0
##   24  103  734  188   30   72   46   19  375  275
##   27  140 1299  336   31  122   60  137  737  969
##   28  216 1126  488   34  104   78   29  762  396
##   32  481 1505  729   33   18   54   18  711  289
##   44  322 1721  822   54  102   89  118 1155  815
##   52  108  535  186   23   44   69    8  386  143
##   53   89  415  152   47   18   42    1  325  181
##   75  333 1161  254   51   81  163   81 1002 1379
##   76  258 1124  333   79  155  123   89  877 1527
##   84  548 1484  390   43   84  113   47  517  963
##   93  220  229   69   20   11   32    6  107  269
##   94    8   99    3    1    2   13   29   53  152
```

```
100 * prop.table (t) %>% round (digits = 4)
```

```
##
##       111  112  120  211  212  221  222  300  400
##   1  0.05 0.02 0.00 0.00 0.00 0.01 0.00 0.01 0.01
##   2  0.04 0.00 0.01 0.01 0.00 0.01 0.00 0.01 0.01
##   3  0.01 0.01 0.00 0.01 0.00 0.00 0.00 0.00 0.04
##   4  0.03 0.01 0.01 0.01 0.00 0.00 0.00 0.00 0.01
##   11 1.13 2.32 0.01 0.01 0.01 0.00 0.00 0.02 0.00
##   24 0.28 2.00 0.51 0.08 0.20 0.13 0.05 1.02 0.75
##   27 0.38 3.54 0.92 0.08 0.33 0.16 0.37 2.01 2.64
##   28 0.59 3.07 1.33 0.09 0.28 0.21 0.08 2.08 1.08
##   32 1.31 4.10 1.99 0.09 0.05 0.15 0.05 1.94 0.79
##   44 0.88 4.69 2.24 0.15 0.28 0.24 0.32 3.15 2.22
##   52 0.29 1.46 0.51 0.06 0.12 0.19 0.02 1.05 0.39
##   53 0.24 1.13 0.41 0.13 0.05 0.11 0.00 0.89 0.49
##   75 0.91 3.16 0.69 0.14 0.22 0.44 0.22 2.73 3.76
##   76 0.70 3.06 0.91 0.22 0.42 0.34 0.24 2.39 4.16
##   84 1.49 4.04 1.06 0.12 0.23 0.31 0.13 1.41 2.62
##   93 0.60 0.62 0.19 0.05 0.03 0.09 0.02 0.29 0.73
##   94 0.02 0.27 0.01 0.00 0.01 0.04 0.08 0.14 0.41
```

Pour aller plus loin et ajouter des variables de pondération, calculer les profils-ligne ou profils-colonne, rendez-vous au module 3, ou demandez à

votre GF (Gentil Formateur)