GaelMillot

Jc Bartier

29/10/2014

Ce document est élaboré à partir de l'ouvrage eponyme **Comprendre et réaliser les tests statistiques à l'aide de R. Manuel pour les débutants.** De Boeck editeur (https://perso.curie.fr/Gael.Millot/Telecharger.htm).

R est un logiciel libre et gratuit exrêmement puissant dont le principal inconvénient est la difficulté d'apprentissage. Il existe une version pour Linux, Mac et Windows. Pour le trouver avec un moteur de recherche utilisez le mot clé **CRAN R** (CRAN pour Comprehensive R Archives Network).

Télécharger R à l'adresse http://www.r-project.org/. R s'utilise en ligne de commandes à partir d'une console, ce qui peut être assez déroutant pour un premier contact. Il est recommandé d'installer ensuite le logiciel **R Studio** qui crée une interface beaucoup plus conviviale pour utiliser R, et met à disposition une palette d'outil pour faciliter la production de documents et s'inscrire dans la démarche de **recherche reproductible**. Ce document a été créé avec l'option *Latex* de RStudio.

Le livre est basé sur les données du fichier *mais.txt*

Structure de la table MAIS

```
mais <- read.table("../GaelMillot/mais.txt", header=TRUE, sep="\t")
names(mais)
```

```
##  [1] "Individu"         "Hauteur"          "Masse"
##  [4] "Nb.grains"        "Masse.grains"     "Couleur"
##  [7] "Germination.epi"  "Enracinement"     "Verse"
## [10] "Attaque"          "Parcelle"         "Hauteur.J7"
## [13] "Verse.Traitement" "Nb.jours.attaque" "Censure.droite"
```

```
str(mais)
```

```
## 'data.frame':    100 obs. of  15 variables:
##  $ Individu        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Hauteur         : int  NA 199 205 173 233 206 261 155 214 174 ...
##  $ Masse           : int  NA 1431 1468 1398 1622 1428 1574 1215 1457 1368 ...
##  $ Nb.grains       : int  NA 320 290 147 138 166 151 293 345 234 ...
##  $ Masse.grains    : num  NA 92.1 89.4 42.6 43.2 ...
##  $ Couleur         : Factor w/ 3 levels "Jaune","Jaune.rouge",..: NA 3 1 1 3 1 1 2 2 3 ...
##  $ Germination.epi : Factor w/ 2 levels "Non","Oui": NA 1 1 1 1 1 1 1 1 1 ...
##  $ Enracinement    : Factor w/ 4 levels "Faible","Fort",..: 1 3 3 1 4 3 3 1 3 2 ...
##  $ Verse           : Factor w/ 2 levels "Non","Oui": NA 1 2 2 2 2 2 1 1 1 ...
##  $ Attaque         : Factor w/ 2 levels "Non","Oui": 2 1 1 1 1 1 1 1 1 1 ...
##  $ Parcelle        : Factor w/ 4 levels "Est","Nord","Ouest",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Hauteur.J7      : int  171 196 198 176 230 200 266 176 220 182 ...
##  $ Verse.Traitement: Factor w/ 2 levels "Non","Oui": NA 2 2 2 2 1 2 2 2 2 ...
##  $ Nb.jours.attaque: int  NA NA NA NA NA NA NA NA NA NA ...
##  $ Censure.droite  : int  NA NA NA NA NA NA NA NA NA NA ...
```

Test de normalité

article: http://eric.univ-lyon2.fr/~ricco/cours/cours/Test_Normalite.pdf

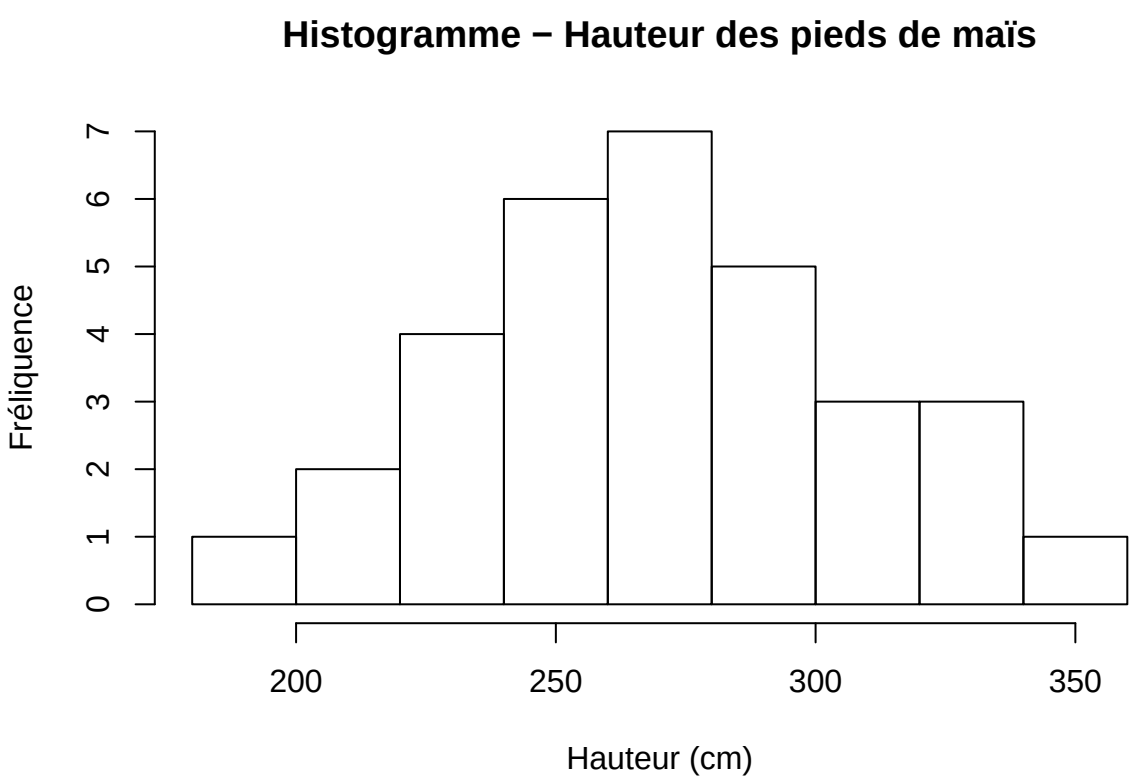
Beaucoup de tests statistiques usuels recquièrent que les données soient distribuées selon une Loi normale. Un test de normalité devrait être un préalable dans ces situations.

On s'intéresse à la normalité de la hauteur des pieds de maïs dans la parcelle Est.

Aspect graphique On résupère le fchier des données et on sélectionne la colonne 1:

```
obs1 <- data.frame(mais[which(mais$Parcelle == "Est"), "Hauteur"])
d <- obs1[,1]

hist(as.numeric(d), main="Histogramme - Hauteur des pieds de maïs", xlab="Hauteur (cm)", ylab="Fréliquer
```



- Hauteur moyenne 272.40625
- écart-type 37.570458

Test de Shapiro-Wilk

Peut-on dire que la variable *hauteur des pieds de maïs* se distribue selon une loi normale ?

Pour répondre à la question on utilise le test de Shapiro-Wilk:

- hypothèse nulle (H0): la hauteur des pieds de maïs se distribue selon une Loi normale. On accepte cette hypothèse si la p-value résultant du test est plus grande (supérieure à) que 0.05
- hypothèse alternative (H1): ce n'et pas une Loi normale

```r
shapiro.test(d)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  d
## W = 0.9903, p-value = 0.9907
```

La p-value est très supérieure à 0.05. On accepte l'hypothèse nulle, la distribution de hauteur des pieds de maïs est normale.

Test de Kolmogorov-Smirnov

On teste si la distribution est conforme à une Loi normale dont on connait **à priori** les paramètres (moyenne, écart-type).

Peut-on dire que la variable hauteur des pieds de maïs se distribue selon une loi normale de moyenne 270 et d'écart-type 35 ?

Pour répondre à la question on utilise le test de Kolmogorov-Smirnov:

- hypothèse nulle (H0): la hauteur des pieds de maïs se distribue selon une Loi normale de moyenne 270 cm avec un écart-type de 35 cm. On accepte cette hypothèse si la p-value résultant du test est plus grande (supérieure à) que 0.05
- alternative (H1): ce n'est pas une Loi normale (si p-value est inférieure à 0.05)

```r
ks.test(d, "pnorm", 270,35)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  d
## D = 0.098, p-value = 0.8888
## alternative hypothesis: two-sided
```

Conclusion: on ne peut pas rejeter l'hypothèse nulle. Donc on accepte que la distribution de la hauteur des pieds de maïs s'ajuste à une Loi normale N(270,35).

Test de Lilliefors

Wikipédia: http://www.wikiwand.com/fr/Test_de_Lilliefors

Le test de Lilliefors est une adaptation du test de Kolmogorov-Smirnov permettant de tester l'hypothèse nulle sur une distribution normale quand les paramètres de la loi normale ne sont pas connus, c'est-à-dire quand ni l'espérance mu ni l'écart type sigma ne sont connus. Ce test fut proposé par Hubert Lilliefors qui était professeur de statistique à l'Université George Washington.

Principe du test

1. Estimer la moyenne et la variance de la distribution en se basant sur les données.
2. Trouver le maximum de variance entre la fonction de répartition empirique et la fonction de répartition de la distribution normale d'espérance et de variance estimée en 1, comme dans le test de Kolmogorov-Smirnov.
3. Enfin, estimer si le maximum de variance est assez grand pour être statistiquement significatif ce qui entraînerait le rejet de l'hypothèse nulle en fonction de la distribution de Lilliefors.

Avec **R** il faut charger la librairie *nortest*. La formule est simple: **lillie.test(x)** où x est le vecteur des valeursdont on veut tester la normalité.

```
    Test de normalité de Lilliefors (Kolmogorov-Smirnov)

data:  d
D = 0.0666, p-value = 0.974

[1] "On accepte l'hypothèse de normalité de données"
```