Cours 04
Quantiles d'une loi normale
empirique, théorique, simulé

1 Quantiles, déciles $k/10$, fractiles k/d

Nous avons vu la définition des quartiles Q_1, Q_2, et Q_3 d'un échantillon ; c'est un cas particulier de quantile q d'une proportion $p = k/d$ (ou d-fractile) : on cherche un nombre q tel qu'une proportion p des individus de l'échantillon qui sont tels que $x_i \leq p$. Lorsque $p = 10\%$, 20%, ..., 90%, les quantiles q_1, q_2, ..., q_9 correspondants s'appellent des `déciles`. Ceci n'est généralement possible qu'approximativement (par exemple si la taille de l'échantillon n'est pas un multiple de d). En pratique, pour $p = k/d$ on choisira [1] q_k =`sort(x)[n*k/d]`

2 Quantiles d'une loi normale

Désignons par $F_{\mu,\sigma}$ la fonction de répartition d'une loi normale $\mathcal{N}(\mu, \sigma$ (ou gaussienne). En d'autres termes $F_{\mu,\sigma} = \int_{-\infty}^{x} \frac{1}{2\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}}\, dt$; c'est pas définition la *probabilité* d'être inférieure à x d'une grandeur aléatoire qui suit une loi $\mathcal{N}(\mu, \sigma$. Ici il est facile de définir le quantile de $p_k = k/d$ de la loi $\mathcal{N}(\mu, \sigma)$: c'est le nombre [2] q_k tel que $F_{\mu,\sigma}(q_k) = p_k (= k/d)$. Il existe bien et est unique pourvu que $0 < p_k < 1$, puisque $F_{\mu,\sigma}$ est continue, strictement croissante, et de valeurs $F_{\mu,\sigma}(]-\infty, +\infty) =]0, 1[$. Sa valeur est donnée, par `R`, par la fonction `qnorme(p,mu,sigma)`, si `p`= p_k, `mu`= μ, et `sigma`= σ. En d'autres termes, c'est le nombre q_k tel que la probabilité d'être inférieure à ce nombre est égale à p_k. Nous voyons donc ici qu'on à juste remplacé "proportion" par "probabilité" en passant d'empirique [3] à théorique. En théorie des probabilités, c'est la Loi des Grands Nombres qui motive cette relation entre "proportion" et "probabilité".

3 Comparaison des quantiles empiriques et théorique

Nous avons déjà vu qu'en statistique un échantillon `x` présentant un histogramme en cloche suggère un modèle gaussien $\mathcal{N}(\mu, \sigma)$, avec μ =`mean(x)` et σ =`sd(x)`, qu'on peut "tester" visuellement, en superposant à l'histogramme (en densité ou proportions e_k/n, par `freq=F`)) la courbe du graphe de la fonction de densité de $\mathcal{N}(\mu, \sigma)$. Une meilleure façon de tester si un modèle gaussien pour `x` est pertinent est de comparer les fractiles q_k de $p_k = k/d$ de l'échantillon et les fractiles théoriques $F^{-1}_{\mu,\sigma}(p_k)$, pour $k = 1, \ldots, d-1$. Pour l'échantillon `x=survey.cc$WrHnd`, nous trouvons μ =`mean(x)`= 18.80238 et σ =`sd(x)`= 1.919205. Nous obtenons, pour $d = 4$, les quartiles empiriques et théoriques suivants :

empirique	17.5	18.5	20.0
théorique	17.50790	18.80238	20.09687

Pour $d = 10$, nous obtenons les déciles empiriques et théoriques suivants :

empirique	16.5	17.5	17.6	18.0	18.5	18.9	19.5	20.5	21.5
théorique	16.34282	17.18714	17.79595	18.31616	18.80238	19.28861	19.80881	20.41762	21.26194

4 Exercice : outlayers d'un échantillon théorique

Considérons la loi gaussienne $\mathcal{N}(\mu, \sigma)$ associée à un échantillon `x`. La boite à moustaches théorique est alors donnée par Q_1 =`Q1=qnorm(0.25,mu,sigma)`= 17.50790, Q_2 =`Q2=qnorm(0.50,mu,sigma)`= 18.80238, Q_3 =`Q3=qnorm(0.75,mu,sigma)`= 20.09687, et la longueur maximale des moustaches est L =`1.5*(Q3-Q1)`= 3.883452.

1. `R` accepte des numéros non entiers : il utilise alors la partie entière du nombre passé comme numéro.
2. ou encore $q_k = F^{-1}_{\mu,\sigma}(p_k)$, où $F^{-1}_{\mu,\sigma}$ *fonction réciproque* de $F_{\mu,\sigma}$, qui est définie pour tout $p \in]0, 1[$.
3. empirique : qui (ne) s'appuie (que sur) l'expérience ; nous utilisons ici le mot "empirique" pour désigner ce qui est relatif à un échantillon issue de la mesure d'un caractère pour une famille d'individus, comme ceux fournis par le bibliothèque (en anglais : library) `MASS` de `R`.

Si `mu`= $\mu = 0$ et si `sigma`= $\sigma = 1$, l'interquartile vaut $Q_3 - Q_1 = 1.34898$ et la longueur maximale des moustaches vaut $1.5(Q_1 - Q_3) = 2.023469$. Les valeur exceptionnelles de `x` sont donc celles qui sont inférieures à $q_{min} = Q_1 - L = -2.697959$ ou supérieures à $q_{max} = Q_3 + L = 2.697959$. La probabilité p_{min} d'être inférieur à q_{min} est alors de 0.003488302, et la probabilité p_{max} d'être supérieur à q_{max} est alors de 0.003488302

```
#        Cours de Stat Mass 04
#                Quantiles
rnorm(1,0,1) # vérification que cela commence toujours pareil ou non
set.seed(2015) # on s'assure que cela commencera toujours pareil
###################### quelques fonctions liées à la loi normale
# dnorm(x, mean = 0, sd = 1, log = FALSE)
# pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
# qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
# rnorm(n, mean = 0, sd = 1)
#############################################################
# initialisation des variables "Tailles" et "x"
library(MASS)  # MASS = Modern Applied Statistics with S
data(survey)   #lecture des données de  survey
survey.cc<-survey[complete.cases(survey),] #nettoyage de  survey
x<-survey.cc$Wr.Hnd;x  #x  prends la valeur de la première colonne de  Tailles.cc
hist(x)        #histogramme des valeurs de  x
hist(x,freq=F)
n=length(x);n

################# quartile ############################
d=4
d;n/d  # affichage du fractile et de la taille (idéale) des classes
(1:d)*n/d   # utile pour les indices de quartiles
indicesbornesx=floor((1:d)*n/d);indicesbornesx # on arrondi à l'entier inférieur
sort(x)   #affichage trié
bornesx=1:(d+1);bornesx  #creation du vecteur
bornesx[1]=sort(x)[1];bornesx[1]   #on commence par le plus petit
bornesx[2:(d+1)]=sort(x)[indicesbornesx]
bornesx  # affichage des quantiles et des élements extrèmes
hist(x,breaks=bornesx)
hist(x,breaks=bornesx,freq=T) #

################  deciles ######################################
d=10
d;n/d  # affichage du fractile et de la taille (idéale) des classes
(1:d)*n/d
indicesbornesx=floor((1:d)*n/d);indicesbornesx # on arrondi à l'entier inférieur
sort(x)   #affichage trié
bornesx=1:(d+1);bornesx  #creation du vecteur
bornesx[1]=sort(x)[1];bornesx[1]   #on commence par le plus petit
bornesx[2:(d+1)]=sort(x)[indicesbornesx]
bornesx  # affichage des quantiles et des élements extrèmes
hist(x,breaks=bornesx)
hist(x,breaks=bornesx,freq=T) # il y a énormément de 17.5: ils vont tous dans la classe ]16.5,17.
hist(x,breaks=bornesx)  # retour à l'histogramme en densité

######################" comparaison avec une loi "normale" de même caractéristiques
mu=mean(x);mu
sigma=sd(x);sigma  # sd vient de "standard deviation"
magaussienne=function(x){return(dnorm(x,mu,sigma))}
xmin=min(x);xmax=max(x)
plot(magaussienne,xmin,xmax,col="red",add=TRUE) # tracé du graphe de la fonction entre  xmin et x

#################   deciles théoriques
d=10
bornesthx=qnorm((0:d)/d,mu,sigma);bornesthx
bornesthxBis=bornesthx
bornesthxBis[1]=xmin # on triche: on remplace -Inf par xmin
bornesthxBis[11]=xmax # on triche: on remplace Inf par xmax
hist(x,breaks=bornesthxBis,col="red",add=T) #
hist(x,breaks=bornesx,col="yellow",add=T)
bornesx;bornesthx
qqplot(bornesx,bornesthx)
x.norm=rnorm(length(x),mean(x),sd(x))
qqplot(x.norm,x)

##################  Quartiles théorique
boxplot(x)
mu=mean(x);mu
sigma=sd(x);sigma  # sd vient de "standard deviation"
Q=qnorm(0.25*(1:3),mu,sigma);Q
Q[3]-Q[1] # interquartile
L=1.5*(Q[3]-Q[1]);L # longueur maximale des moustache:
sort(x) #affichage trié
```

```
# Tout caractère inférieur qmin ou supérieur à qmax est exceptionnel (outliers)
qmin=Q[1]-L;qmin
qmax=Q[3]+L;qmax
pmin=pnorm(qmin,mu,sigma);pmin # proba théorique d'être inférieur à qmin
# déciles théorique
D=qnorm(0.1*(1:9),mu,sigma);D
d=10
indicesbornesx=floor((1:d)*n/d);indicesbornesx # c'est la première classe qui est trop petite

###################### échantillons gaussiens centré réduit
set.seed(1515)  # choix de la "graine" d'initiation des nombres aléatoires
n=1000
u=rnorm(n,0,1)
hist(u)
h=hist(u,freq=F)
magaussienne=function(x){return(dnorm(x,0,1))}
plot(magaussienne,min(u),max(u),col="red",add=T) # tracé du graphe de la fonction entre  xmin et
plot(h,add=T,freq=F)
d=4;(0:d)/d
u.sort=sort(u);u.sort
bornes=1:(d+1);bornes # creation de d; sinon "objet 'bornes' introuvable" à la suivante
bornes[1]=u.sort[1];bornes[1]
(1:d)*n/d
u.sort[(1:d)*n/d]
bornes[2:(d+1)]=u.sort[(1:d)*n/d]
bornes
h=hist(u,breaks=bornes,freq=T)
h$xname <- "u normal (effectifs)"
plot(h,freq=T)
hist(u,breaks=bornes,freq=F)
hist(u,breaks=bornes)
magaussienne=function(x){return(dnorm(x,0,1))}
plot(magaussienne,u.sort[1],u.sort[n],col="red",add=TRUE) # tracé du graphe de la fonction entre
```