Statistiques Inférentielles II

Jairo Cugliari & Guillaume Metzler

2/5/2022

Exercice 1

On considère le fichier de données "SOCELL.csv." A l'aide de R, tester si le courant de court-circuit μ (ISC) au temps t_1 est significativement inférieur à $\mu_0 = 4$ ampères. On effectuera un test de student.

Pour rappel, il faudra donc déterminer la moyenne $\bar{x}$ des courants de court-circuit x_i ainsi que l'écart-type s de ce même quantité au temps t_1, i.e.

$$\bar{x} = \sum_{i=1}^{n} x_i \quad \text{et} \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

En outre, nous allons effectuer un test de Student car nous ne connaissons pas la variance associée à la distribution de l'ampérage au temps t_1. Nous testons les hypothèses suivantes :

$$H_O : \mu \geq \mu_0 = 4 \quad \text{v.s.} \quad H_1 : \mu > \mu_0 = 4.$$

La statistique de test de student, se présente sous la forme

$$\frac{\bar{x} - \mu_0}{\sqrt{\frac{s^2}{n}}}.$$

Faisons ce test à l'aide de la fonction "t.test" de R. Commençons par charger le jeu de données (je suppose que le fichier de données se trouve dans le même répertoire que votre code.)

```
# Chargement des données
SOCELL <- read.csv("~/Documents/SOCELL.csv")

# Estimations
x_m = mean(SOCELL$t1)  # Moyenne
s = sd(SOCELL$t1)   # Ecart-type
n = length(SOCELL$t1)  # taille échantillon

# Calcul de la statistique de test sous H_0
stat.t <- (x_m - 4)/(s/sqrt(n))

# On compare cette valeur là à une valeur critique (on prendra un seuil de 5%)
print(paste("L'hypothèse H_0 peut être rejetée ", stat.t > qt(0.95,n-1)))
```

```
## [1] "L'hypothèse H_0 peut être rejetée  FALSE"
```

Nous aurions aussi pu faire cela avec la ligne de commande suivante

```
# Test de student
t.test(SOCELL$t1, mu = 4, alternative = "greater")
```

```
##
##  One Sample t-test
##
## data:  SOCELL$t1
## t = -0.40303, df = 15, p-value = 0.6537
## alternative hypothesis: true mean is greater than 4
## 95 percent confidence interval:
##  3.76261     Inf
## sample estimates:
## mean of x
##  3.955625
```

La p-value de $0.65 > 0.05$ ne permet pas de rejeter l'hypothèse selon laquelle le courant de circuit est, en moyenne, significativement inférieur à 4.

Exercice 2

Toujours avec le même fichier de données qu'à l'exercice précédent, déterminer si le courant de court-circuit au temps t_2 est significativement supérieur à 4 ampères.

On procède exactement de la même façon que précédemment, nous effectuerons tout de suite le test avec la fonction appropriée.

```
# Chargement des données
SOCELL <- read.csv("~/Documents/SOCELL.csv")

# Test de student
t.test(SOCELL$t2, mu = 4, alternative = "less")
```

```
##
##  One Sample t-test
##
## data:  SOCELL$t2
## t = 2.0056, df = 15, p-value = 0.9684
## alternative hypothesis: true mean is less than 4
## 95 percent confidence interval:
##      -Inf 4.392381
## sample estimates:
## mean of x
##  4.209375
```

On ne peut pas rejeter l'hypothèse selon laquelle la moyenne des courants de court-circuit au temps t_2 est supérieure à 4.

Exercice 3 :

Un échantillon de taille $n = 20$ est tiré aléatoirement selon une loi normale et donne les valeurs suivantes

```
x <- c(20.74, 20.85, 20.54, 20.05, 20.08, 22.55, 19.61,
       19.72, 20.34, 20.37, 22.69, 20.79, 21.76, 21.94,
       20.31, 21.38, 20.42, 20.86, 18.80, 21.41)
```

Intervalle de confiance sur la moyenne μ On rappelle que cet intervalle $I_{1-\alpha}$, étant donné que notre échantillon est gaussien de variance inconnue, est de la forme

$$I_{1-\alpha} = \left[\bar{x} - t_{1-\alpha/2,n-1}\frac{s}{\sqrt{n}};\ \bar{x} + t_{1-\alpha/2,n-1}\frac{s}{\sqrt{n}}\right],$$

où $(x_i)_{i=1}^n$ désignent les valeurs prises par notre échantillon de taille n, $\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i$ et $s^2 = \frac{1}{n-1}\sum_{i=1}^n (x_i - \bar{x})^2$. Les bornes de cet intervalle sont données par

```
# Erreur
alpha = 0.01

# Taille échantillon
n <- length(x)

# Bornes de l'intervalle
lower_bound <- mean(x) - qt(1-alpha/2,n-1)*sd(x)/sqrt(n)
upper_bound <- mean(x) + qt(1-alpha/2,n-1)*sd(x)/sqrt(n)
```

Intervalle de confiance sur la variance σ^2 L'intervalle de confiance sur la variance se construit à l'aide de la loi du $\mathcal{X}^2$, distribution de probabilité suivie par la variable aléatoire

$$\frac{(n-1)s^2}{\sigma^2} \sim \mathcal{X}^2_{n-1}.$$

On en déduit que notre intervalle de confiance $I_{1-\alpha}$ sur la variance σ^2 est donné par

$$I_{1-\alpha} = \left[\frac{(n-1)s^2}{\mathcal{X}^2_{1-\alpha/2,n-1}};\ \frac{(n-1)s^2}{\mathcal{X}^2_{\alpha/2,n-1}}\right].$$

Les bornes de cet intervalle sont données par

```
# Erreur
alpha = 0.01

# Taille échantillon
n <- length(x)

# Bornes de l'intervalle
lower_bound <- ((n-1)*sd(x)^2)/qchisq(1-alpha/2,n-1)
upper_bound <- ((n-1)*sd(x)^2)/qchisq(alpha/2,n-1)
```

Déterminer un intervalle de confiance sur l'écart-type σ Il suffit de reprendre ce qui précède en considérant la racinne carrée des bornes supérieure et inférieure de notre intervalle de confiance

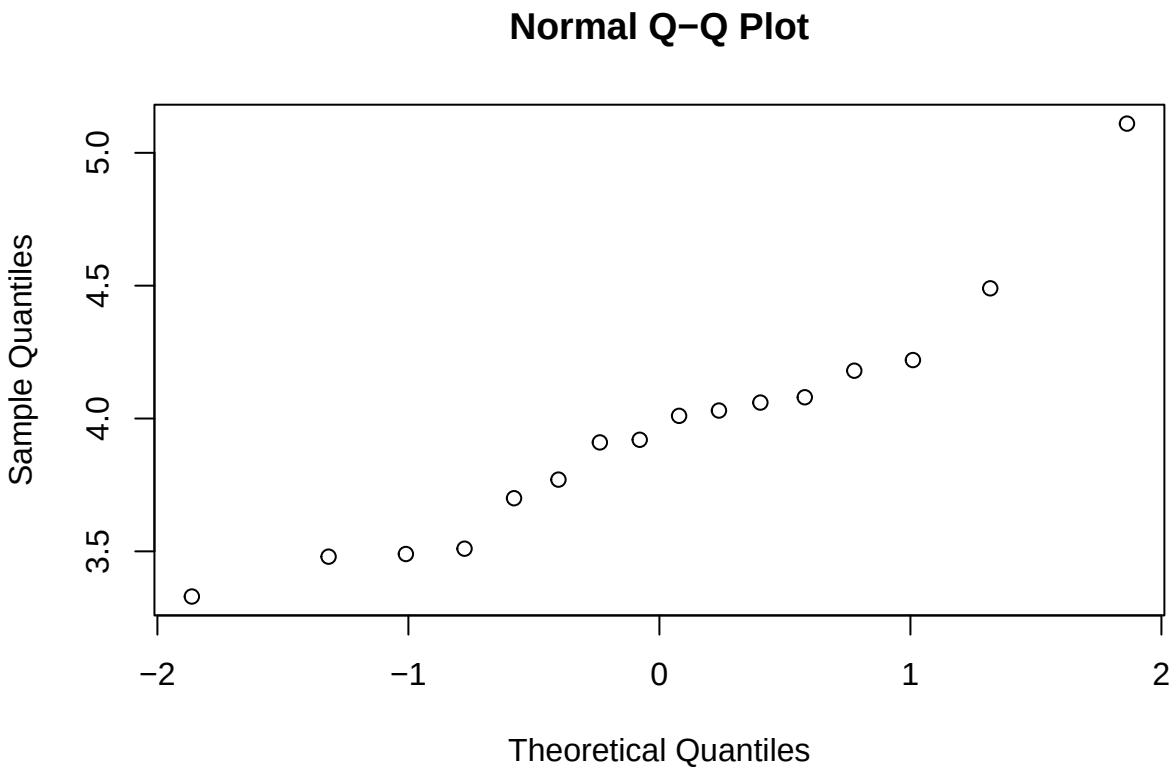
Exercice 4

Effectuer un plot normal quantile-quantile afin de tester si, graphiquement, le courant de court-circuit au temps t_1 est normalement distribué ou non.

On utilisera pour cela la fonction “qqnorm”.

```
# Chargement des données
SOCELL <- read.csv("~/Documents/SOCELL.csv")

# Graphe quantile-quantile
qqnorm(SOCELL$t1)
```



Obtention du graphique On retrouve, en abscisse les quantiles théoriques de la loi normale centrée réduite, et en ordonnée les quantiles empirique de notre échantillon, i.e. l'ordonnée des points représentent les valeurs de notre échantillon.

La fonction “qqnorm” va associer, à notre échantillon ordonné, son quantile théorique de la loi normale centrée réduite.

Notons $(x_1, x_2, \ldots, x_n)$ notre n-échantillon et $(x_{(1)}, x_{(2)}, \ldots, x_{(n)})$ sa version ordonnée. A partir de cet échantillon ordonné, il est possible d'associer la probabilité qu'une autre valeur de cet échantillon soit inférieure à celle observée, i.e. pour tout i, on peut calculer la probabilité $\mathbb{P}[X \leq x_{(i)}]$ pour tout $i = 1, \ldots, n$. Comme notre échantillon est ordonnée, cette probabilité est donnée par

$$\mathbb{P}[X < x_{(i)}] = \frac{i}{n}.$$

Cette probabilité i/n est ensuite utilisée pour calculer le quantile théorique d'ordre i/n, noté $\Phi(i/n) = z_{i/n}$, de la loi normale centrée réduite, où Φ désigne la fonction de répartition de la loi normale centrée réduite. Ce qui nous permet d'obtenir la valeur en abscisse pour chaque valeur de notre échantillon ordonné. Pour résumer :

sample	$\mathbb{P}[X \leq x]$	$\Phi(p) = z_p$
$x_{(1)}$	$1/n$	$\Phi(1/n) = z_{1/n}$
$x_{(2)}$	$1/n$	$\Phi(2/n) = z_{2/n}$
$\vdots$	$\vdots$	$\vdots$
$x_{(n)}$	$1/n$	$\Phi(n/n) = z_{n/n}$

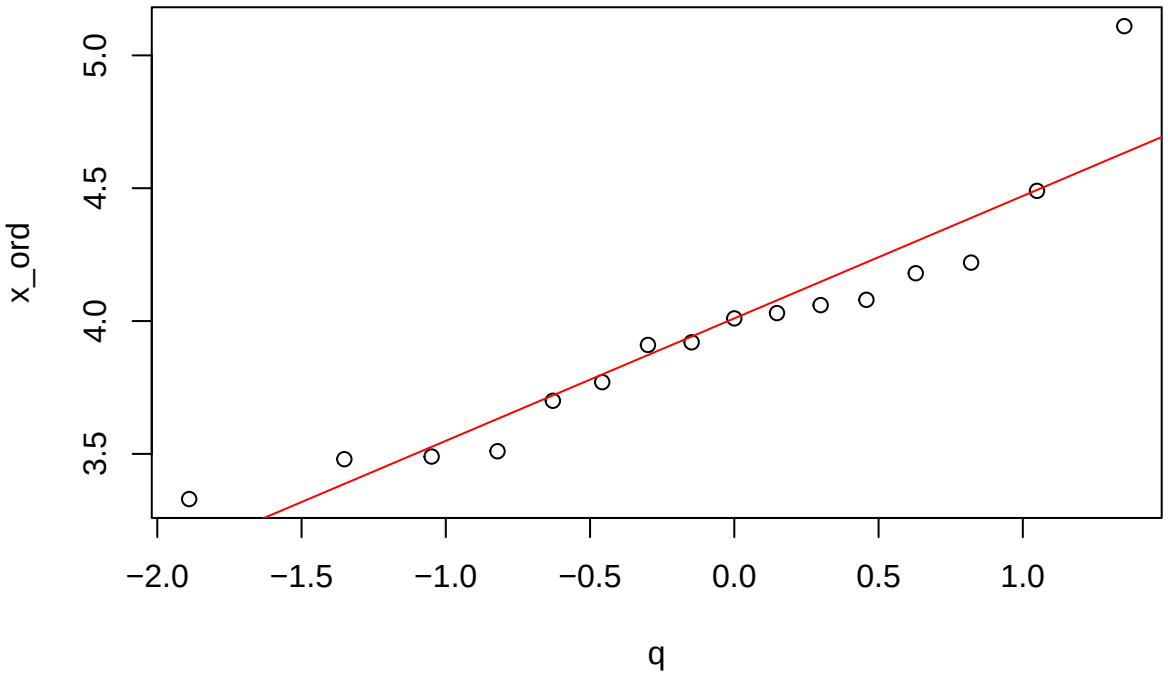
Dans la pratique, on va plutôt considérer des probabilité de la forme $\frac{i-0.5}{n}$ au lieu de $\frac{i}{n}$.

Droite de régression L'objectif est ici de déterminer les paramètres de la loi normale (à supposer que notre distribution est bien gaussienne). On peut le faire en effectuant une régression linéaire sur notre précédent graphe. On va donc recalculer les coordonnées des différents points.

```r
# Echantillon ordonné
x_ord <- sort(SOCELL$t1)
n <- length(x_ord)
# Probabilité
p = (c(1:n)-0.5)/(n+1)
# Quantiles de la loi normale
q = qnorm(p)

# On effectue la régression linéaire
mymodel <- lm(x_ord~q)

# Graphes
plot(q,x_ord)
abline(mymodel$coefficients, col = "red")
```



Les coefficients de la régression sont donnés par

```
mymodel$coefficients
```

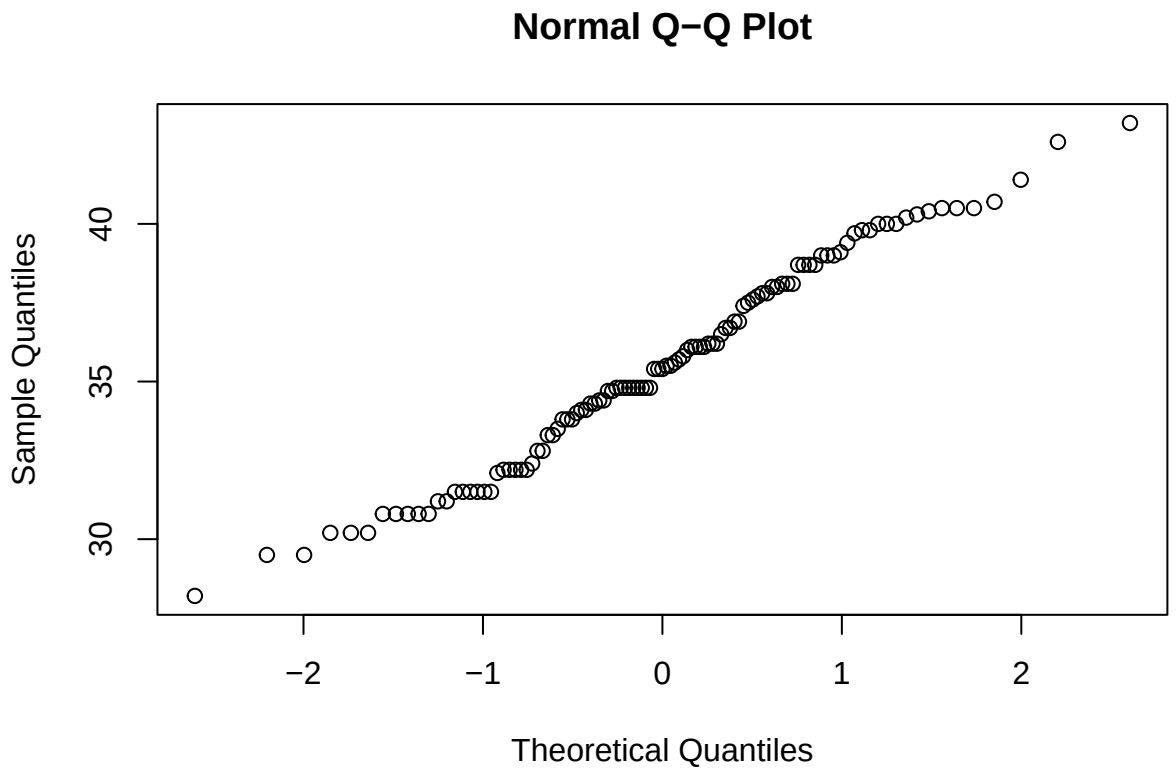
```
## (Intercept)           q
##    4.010046    0.460830
```

Et permettent de lire directement les paramètres de la loi, à savoir une moyenne μ égale à 4.01 et un écart-type de 0.46.
Cela reste bien évidemment uen estimation qui est valable uniquement si notre ajustement est correct.

Exercice 5 A l'aide R et de l'exercice précédent, et en utilisant le fichier "CAR.csv".

(i) Tester graphiquement le fait que le diamètre de rotation est normalement distribué

```
# Chargement des données
CAR <- read.csv("~/Documents/CAR.csv")
# Graphe quantile-quantile
qqnorm(CAR$turn)
```



On pourait à nouveau représenter la droite de régression comme cela a été fait dans l'exercice précédent, en utilisant la même méthode. Cela nous permettrait également d'estimer les paramètres de la loi normale, à supposer que notre échantillon est bien distribué selon une loi gaussienne.

```
# Test de Shapiro
shapiro.test(CAR$turn)
```

```
##
##  Shapiro-Wilk normality test
##
```

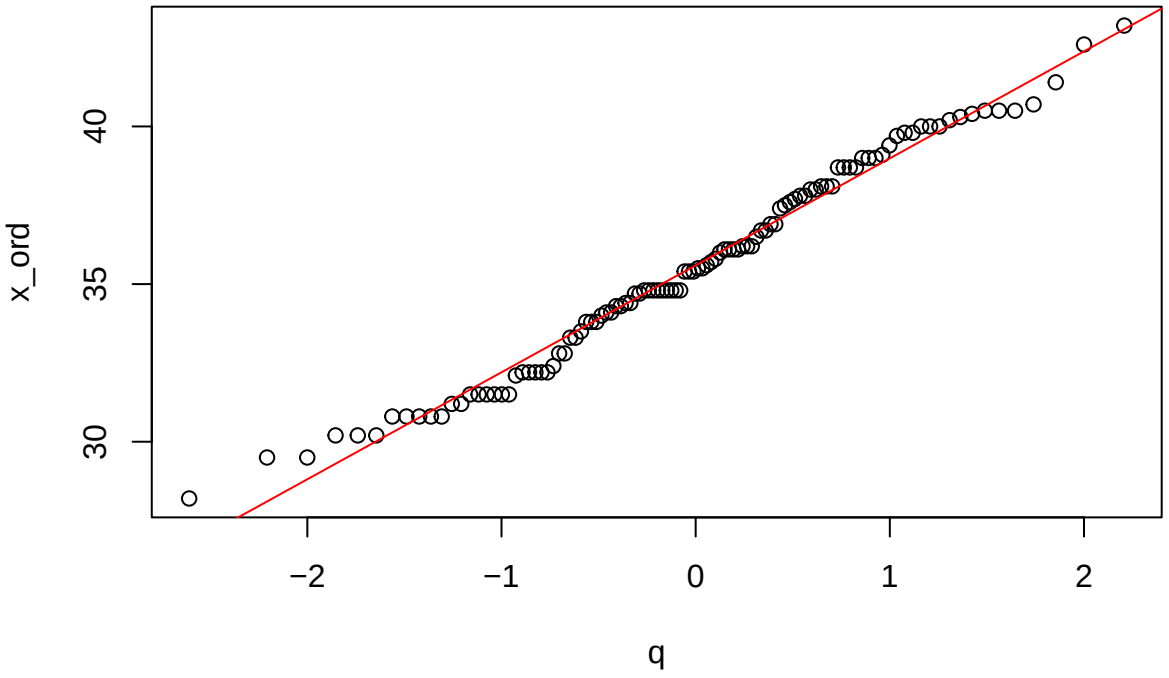
```
## data:  CAR$turn
## W = 0.98154, p-value = 0.1355
```

Ici, il n'y rien dans les données qui contredit l'hypothèse de normalité.

```
# Echantillon ordonné
x_ord <- sort(CAR$turn)
n <- length(x_ord)
# Probabilité
p = (c(1:n)-0.5)/(n+1)
# Quantiles de la loi normale
q = qnorm(p)

# On effectue la régression linéaire
mymodel <- lm(x_ord~q)

# Graphes
plot(q,x_ord)
abline(mymodel$coefficients, col = "red")
```



Et enfin, les paramètres de notre loi

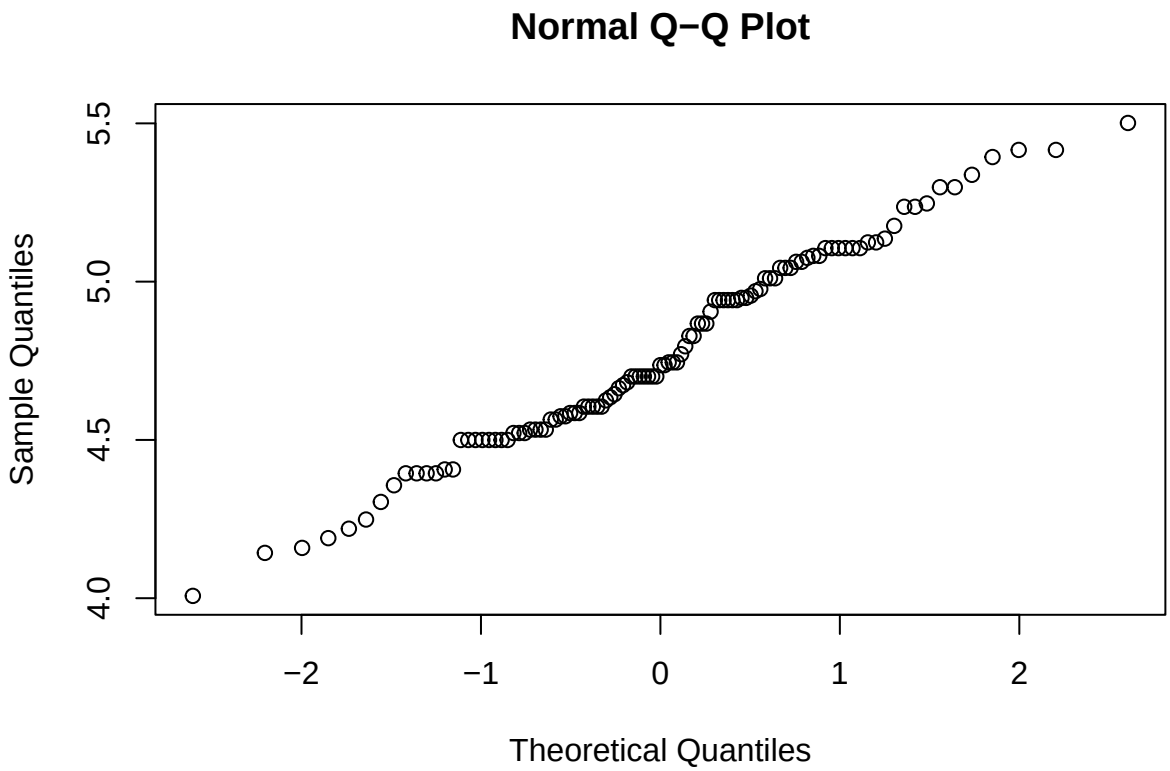
```
mymodel$coefficients
```

```
## (Intercept)           q
##   35.594974    3.393449
```

(ii) Tester graphiquement le fait que le log du nombre de chevaux de la voiture est normalement distribué.

On va faire la même chose avec le log de la puissance de la voiture

```r
# Chargement des données
CAR <- read.csv("~/Documents/CAR.csv")
# Graphe quantile-quantile
qqnorm(log(CAR$hp))
```



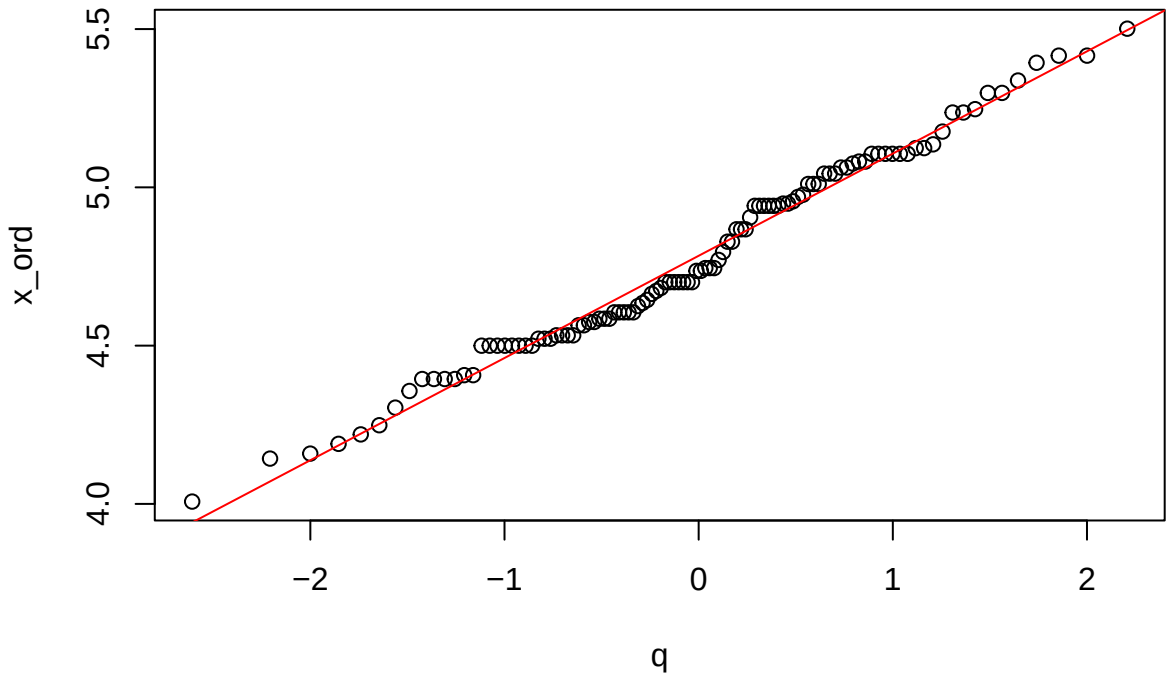

```r
# Test de Shapiro
shapiro.test(log(CAR$hp))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(CAR$hp)
## W = 0.98298, p-value = 0.1789
```

```r
# Echantillon ordonné
x_ord <- sort(log(CAR$hp))
n <- length(x_ord)
# Probabilité
p = (c(1:n)-0.5)/(n+1)
# Quantiles de la loi normale
q = qnorm(p)

# On effectue la régression linéaire
mymodel <- lm(x_ord~q)

# Graphes
plot(q,x_ord)
abline(mymodel$coefficients, col = "red")
```




```
mymodel$coefficients
```

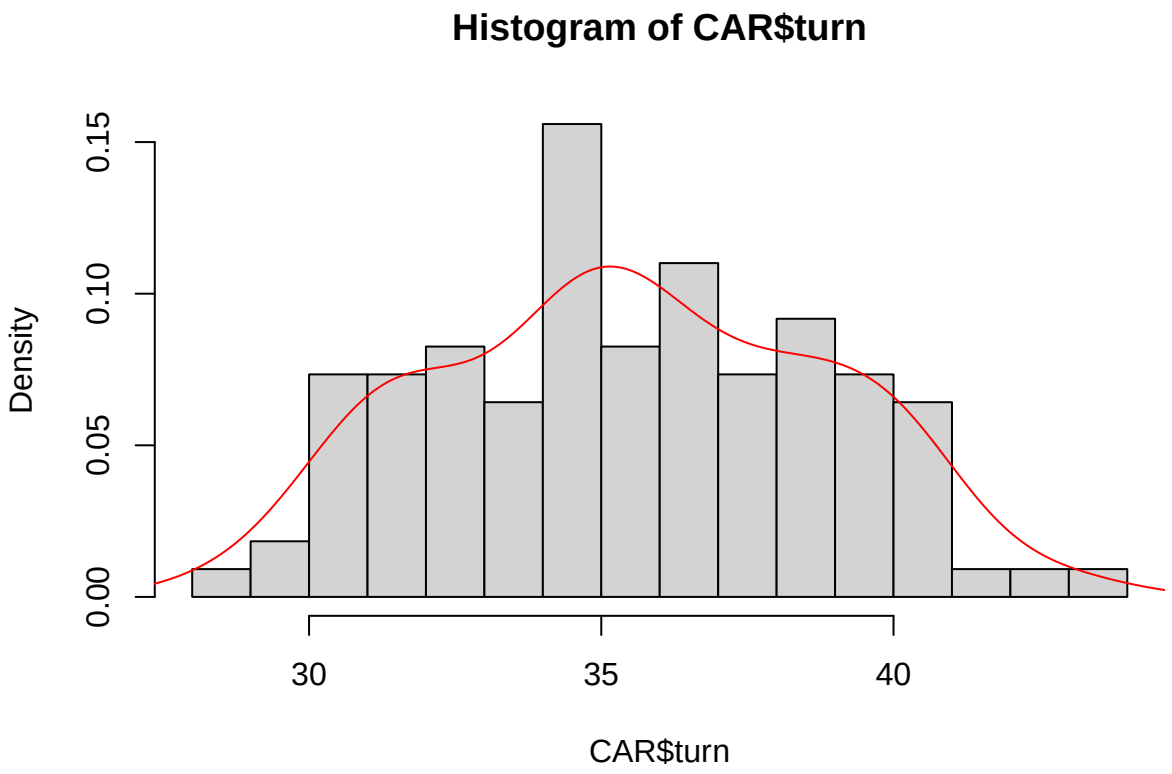
```
## (Intercept)           q
##   4.7837651   0.3229355
```

Exercice 6

En utilisant le fichier "CAR.csv", tracer un histogramme de la distribution du diamètre de rotation, en représentant 11 intervalles. Estimer la distribution normal sur ce diagramme et effectuer un test d'ajustement du $\mathcal{X}^2$ pour tester l'adéquation.

Histogramme Commençons par représenter nos données sous la forme d'un histogramme.

```
# Chargement des données
CAR <- read.csv("~/Documents/CAR.csv")
# Graphe quantile-quantile
hist(CAR$turn, breaks = 11, probability = TRUE)
lines(density(CAR$turn), col = "red")
```



Test d'ajustement du Chi-deux L'énoncé suggère de séparer les données en 11 classes, on va donc séparer notre échantillon ordonné en 11 classes distinctes.

On va ensuite comparer l'effectif de ces classes auxquelles on aura associé une probabilité, l'effectif théorique de cette classe sous l'hypothèse d'ajustement.

Création des classes, pas forcément le même effectif ! On commence par regarder la valeur min et max et on divise cela en 11 intervalles (on déterminer les bornes de ces intervalles).

On souhaite faire un ajustement à une loi normale ... mais laquelle ? Comme on ne le sait pas, on va faire un test d'ajustement à une loi normale centrée et réduite. Pour cela, on va donc centrer et réduire notre jeu de données !

```
# Normalisation des données
data<- as.data.frame(CAR$turn)
names(data) ="turn"
data$turn <- (data$turn-mean(data$turn))/sd(data$turn)
```

On va amintenant pouvoir s'occuper de la mise en classe.

```
# Bornes des classes
bornes_classes <- seq(min(data$turn), max(data$turn), length = 12)
# Regroupement dans des classes
data$classe = 0
for (i in 1:length(data$turn)){
  data$classe[i] <- max(which((data$turn[i]-bornes_classes)>=0))
}
data$classe[data$classe == 12] = 11
```

On peut maintenant extraire les effectifs de chaque classe et aussi les probabilités de chaque classe (les probabilités sont inutiles pour la suite !).

```
eff_emp <- table(data$classe)
prob_emp <- table(data$classe)/sum(eff_emp)
```

on peut ensuite déterminer les effectifs théoriques sous l'hypothèse que noys ayons à faire à une loi normale centrée réduite.
Pour cela il faut repartir de la définition de nos classes, donc repartir du vecteur qui contient les bornes des classes, pour déterminer les probabilités théoriques et après les effectifs théoriques.

```
p <- pnorm(bornes_classes)
# On rassemble les résultats de la première et deuxième borne, qui représente la même classe
p[2] = p[2]+p[1]
# On supprime alors la première valeur
p <- p[-1]
# Ensuite on utilise le fait que P(a<X<b) = P(x<b) - P(X<a).
for(i in length(p):2){
  p[i] = p[i] - p[i-1]
}
# La dernière valeur est ensuite ajustée pour que l'on ait la valeur 1
p[length(p)] = p[length(p)] + (1-sum(p))

# Calcul des effectifs théoriques
p_th <- p
eff_th <- sum(eff_emp)*p
```

La statistique de test du $\mathcal{X}^2$ est définie par

$$x = \sum_{j=1}^{n} \frac{(n_j - np_j)^2}{np_j},$$

où n est l'effectif total, p_j représente la proportion théorique de la classe. Elle est distribuée selon une loi du $\mathcal{X}^2$ à K-1 degrés de liberté, où K représente le nombre de classes, donc ici une loi à 10 degrés de liberté.

```
# Statistique de test
x = sum(((eff_emp - eff_th)^2)/eff_th)
# Test
x > qchisq(0.95,10)
```

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

On ne peut donc pas affirmer que notre échantillon est distribué selon une loi normale centrée et réduite.