Exercices de la session 3

Vincent Guillemot & ArthurTenenhaus

Exercice : Faire un rapport en R Markdown

- Créez un rapport en R Markdown basé sur le *template* proposé par RStudio.

Exercice : Faire un tableau dans un document en R Markdown

1. Créez un document R Markdown :

- dont le titre est "Evolution de la moyenne en fonction de la taille de l'échantillon",
- sans auteur,
- qui aura comme *output* un `html_codument`.

2. Créez un *chunk* de code R contenant la génération de trois échantillons $\mathcal{N}(0, 1)$ de tailles 10, 100 et 1000 et le calcul de la moyenne de ces trois échantillons.
3. En vous aidant du guide *Markdown Quick Reference*, créez un tableau à 2 colonnes présentant

- en colonne 1, la taille des trois échantillons,
- en colonne 2, les moyennes calculées.

Modèles linéaires simples

Exercice : Poids et Tailles

Dans une étude de 1990 de Caroline Davis sur l'image de soi parue dans le revue *Appetite*, l'auteur recueille des mesures de poids et de taille sur des individus auxquels on demande également d'estimer sans appareillement leur taille et poids actuels. Les données de cette partie de l'étude se trouve en suivant ce lien.

1. Utilisez la fonction `read.table` pour lire ces données tabulées. Un petit tour sur la page d'aide de la fonction (`?read.table`) aidera certainement.
2. Faites deux graphes côte-à-côte de le taille en fonction du poids mesurés ou reportés.

3. Pouvez-vous identifier un *outlier* ? Pour cela, vous pouvez éventuellement chercher une fonction pour vous aider dans la librairie `car`.
4. Sur le même graphe, représentez la taille en fonction du poids mesurée et reportée. Utilisez la fonction `arrows` pour mieux visualiser les différences.

Exercice : Cerveaux des mammifères (Allison and Cicchetti 1976)

Des mesures du poids du cerveau et du poids total du corps oint été effectuées chez 62 mammifères (Allison and Cicchetti 1976). Nous cherchons à constuire un modèle sans *intercept* (car un animal de poids nul aura certainement un cerveau de poids nul).

1. Lisez les données se trouvant ici avec la fonction `read.table`. L'argument `skip` vous aidera à extraire les données !
2. Renommez les trois colonnes en `ID`, `brain` et `total` avec la fonction `colnames`.

Exercice : Contraste

Exercice : ANOVA

Exercice : Exemple de test *post-hoc*

Exercice : Sélection de variable

Exercice : Comparaison de modèles

La régression simple

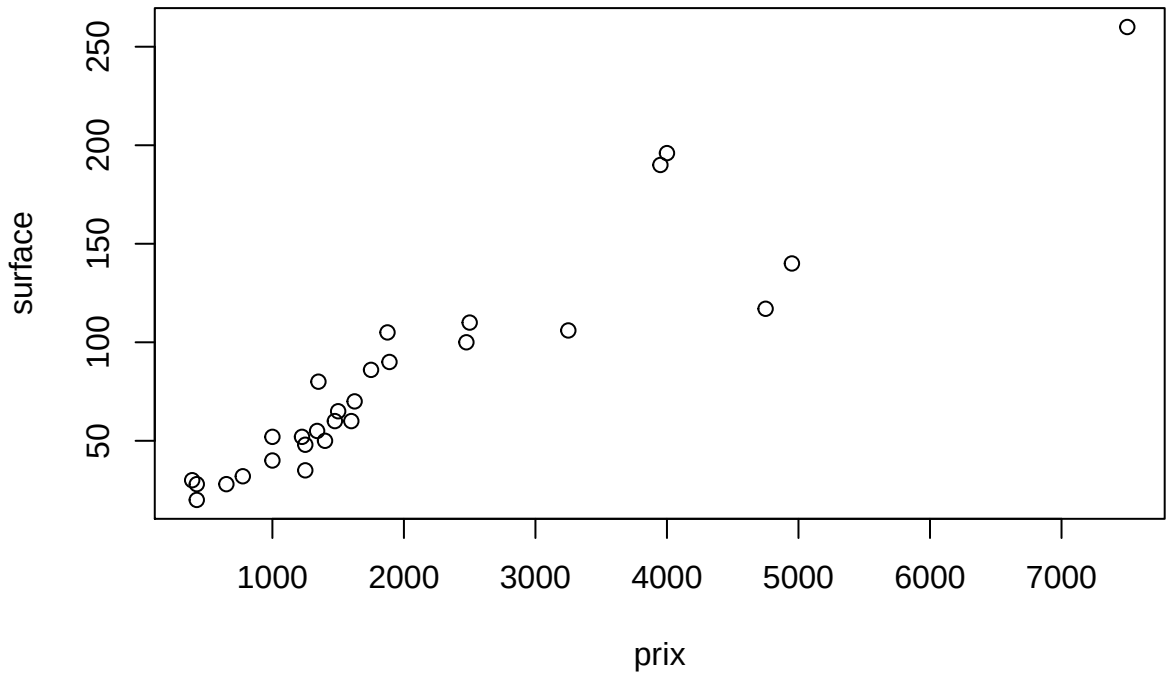
1. Importer dans une variable nommée `X` le jeu de donné nommé `appart.csv`. *Indice : `read.csv`.*
2. Tracer le graphe bivarié de la surface sur le prix. Que constate-t-on ? *Indice : `plot`.*
3. Construire un modèle linéaire reliant le prix à la surface. *Indice : `lm`.*
4. Ajouter au graphe bivarié construit à la question 1, la droite de régression résultant du modèle de la question 2. *Indice : `abline`.*
5. Donner les valeurs prédites par le modèle pour chacune des observations. *Indice : `predict`.*
6. Observer les résidus. *Indice : `...$residuals`.*

Proposition de correction. On peut construire `X` en utilisant la fonction `read.csv` :

```
X <- read.table("appart.txt", header=TRUE, row.names=1)
```

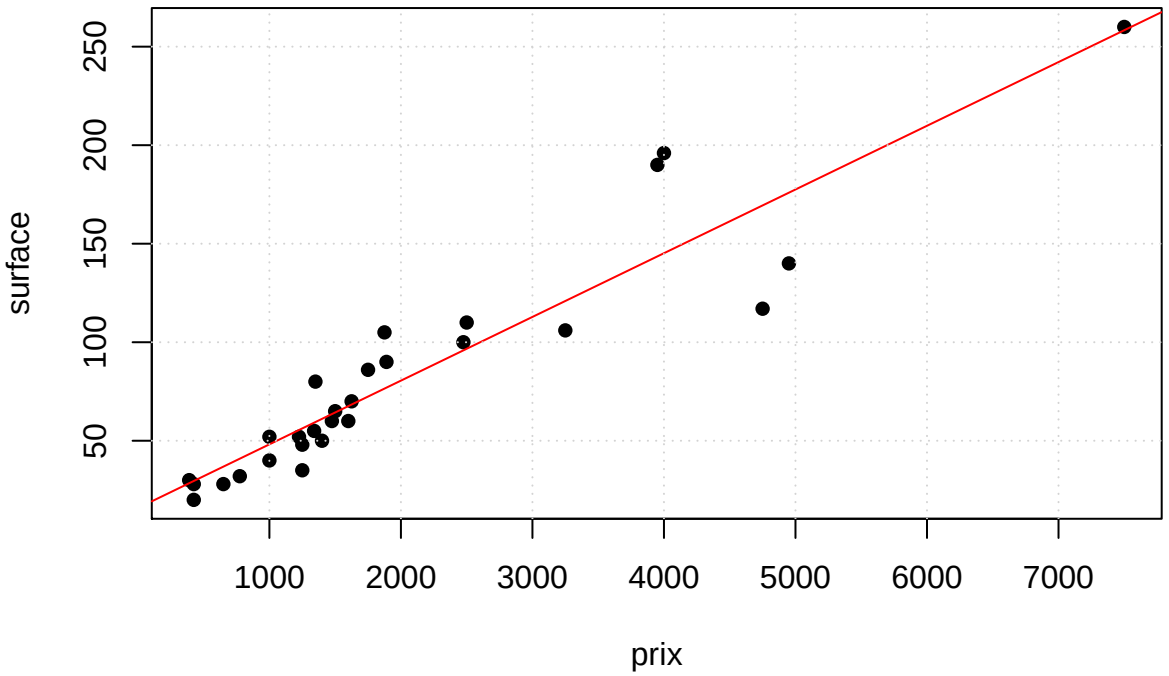
Obtenir le plot bivarié s'obtient très rapidement avec la fonction `plot` :

```
plot(X)
```



On constate qu'il y a un lien fort, comme on pouvait s'y attendre, entre la surface et le prix d'un appartement. En ajoutant le modèle linéaire au graphe, on se rend compte qu'il modélise très bien le lien entre les deux variables.

```
plot(surface~prix, data=X, pch=16)
abline(lm(surface~prix, data=X), col=2)
grid()
```



La prédiction de la surface se fait avec la fonction `predict`. Il est important de réaliser que cette fonction peut s'utiliser sur de nouvelles données ! On peut donc prédire, avec le modèle construit, la surface d'un appartement quand on connaît son prix.

```
res.lm <- lm(surface~prix, data=X)
predict(res.lm, data=X)
```

Enfin, les résidus sont calculés en même temps que le modèle. On peut y accéder avec la commande suivante (quand le modèle a été attribué à une variable s'appelant `res.lm`) :

```
res.lm$residuals
```

La régression multiple et la sélection de variables

Reprenons l'exemple précédent et rajoutons deux variables qui n'ont absolument rien à voir avec le prix et la surface des appartements : une première variable `z1` constituée aléatoirement de 0 et de 1 et une deuxième variable `z2` constituée de valeurs aléatoires réparties uniformément entre 0 et 1.

```
n <- nrow(X)
X$z1 <- sample(0:1, n, replace=TRUE)
X$z2 <- runif(n)
```

Observons maintenant le résultat de l'ajustement d'un modèle linéaire multiple (c'est à dire contenant plus d'une variable) :

```
summary(lm(surface ~ prix + z1 + z2, data=X))
```

```
##
## Call:
## lm(formula = surface ~ prix + z1 + z2, data = X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.367 -11.270  -2.841  10.950  48.796
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.115605   9.333182   1.620    0.118
## prix          0.033146   0.002797  11.850 1.62e-11 ***
## z1          -10.875829   8.699116  -1.250    0.223
## z2            8.976750  16.894653   0.531    0.600
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.39 on 24 degrees of freedom
## Multiple R-squared:  0.8754, Adjusted R-squared:  0.8599
## F-statistic: 56.23 on 3 and 24 DF,  p-value: 5.286e-11
```

On remarque que les coefficients associés aux variables $z1$ et $z2$ ont des valeurs qui ne semblent pas proches de 0, et pourtant leur significativité dans le modèle est faible ! Il est donc important de toujours utiliser la fonction `summary` pour mettre en perspective les valeurs des coefficients.

Enfin, il est possible d'utiliser la fonction `step` pour sélectionner le modèle permettant le mieux de décrire les relations entre les différentes variables :

```
step(lm(surface ~ prix + z1 + z2, data=X))
```

```
## Start:  AIC=175.22
## surface ~ prix + z1 + z2
##
##        Df Sum of Sq   RSS    AIC
## - z2    1       129 11114 173.54
## - z1    1       715 11700 174.98
## <none>              10985 175.22
## - prix  1     64267 75252 227.10
##
## Step:  AIC=173.55
## surface ~ prix + z1
##
##        Df Sum of Sq   RSS    AIC
## - z1    1       754 11868 173.38
## <none>              11114 173.54
## - prix  1     71682 82796 227.77
##
## Step:  AIC=173.38
```

```
## surface ~ prix
##
##        Df Sum of Sq   RSS    AIC
## <none>              11868 173.38
## - prix  1     76322 88190 227.54
```

```
##
## Call:
## lm(formula = surface ~ prix, data = X)
##
## Coefficients:
## (Intercept)         prix
##    15.84153      0.03233
```

Pour cet exemple très simple, les variables `z1` et `z2` ont été éliminées du modèle initial, et on a convergé vers un modèle du type `surface ~ prix`. Il faut faire attention au fait que la méthode utilisée est très sensible : le résultat dépend du nombre de variables, du nombre d'individus, et pose des hypothèses fortes sur la nature statistique des données.

Par exemple, reprenons l'exemple précédent après avoir rajouté 10 variables aléatoire sans rapport avec le prix ou la surface au jeu de données `X`. *(à faire en exercice)*

Exemples

- http://www.ats.ucla.edu/stat/examples/rwg/
- http://www.math.hope.edu/swanson/
- http://www.math.hope.edu/swanson/data/body_temp.txt
- http://www.ats.ucla.edu/stat/r/modules/raw_data.htm
- http://www.statsci.org/datasets.html
- http://www.stat.ufl.edu/~winner/datasets.html

Références

Allison, T, and D. Cicchetti. 1976. "Sleep in Mammals: Ecological and Constitutional Correlates." *Science* 194 (4266): 732–34. doi:10.1126/science.982039.