Analyse de données : TP1

Avant de commencer, créer dans votre espace de travail, un répertoire `AD` (si ce n'est déjà fait). Lancez le logiciel R, et placez le répertoire courant dans votre répertoire AD (`Fichier -> ...`)

1 Régression linéaire simple

Ouvrez un script (Fichier →Nouveau script) pour pouvoir sauvegarder facilement votre travail. Vous y taperez vos commandes. On peut exécuter tout le script d'un coup (Edition →Exécuter tout), ou seulement la ligne courante (Ctrl+R).

Exercice : Consommation de tabac en France

— On va utiliser les données de l'exercice 1 du TD1, concernant la consommation de tabac en France entre 1985 et 2014. Ces données sont dans le fichier `tabac.txt` sur Teams. Récupérez ce fichier et mettez-le **dans votre répertoire de travail**. Pour importer ce tableau de données dans R, vous pouvez taper la commande suivante :

```
tabac = read.table("tabac.txt", header = T, row.names = 1)
tabac # permet d'afficher le tableau
```

On peut manipuler facilement les colonnes de ce tableau, et calculer quelques statistiques simples :

```
tabac$Vente # permet de récupérer un vecteur avec toutes les valeurs
de la variable Ventes
tabac$Prix # permet de récupérer un vecteur avec toutes les valeurs
de la variable Prix
mean(tabac$Vente) # permet de récupérer la moyenne de la variable
Vente
mean(tabac$Prix) # idem pour Prix
```

— Calculez la corrélation entre les deux variables : `cor(tabac$Prix, tabac$Vente)`

— Tracer le nuage de points par les commandes suivantes

```
plot(tabac$Prix,tabac$Vente)  # tracé simple
plot(tabac$Prix,tabac$Vente, main="Consommation de tabac en France",col="blue",lwd=2,
xlab="Prix relatif de vente (en euros)",ylab="Nombre de cigarettes vendues
(en milliards)")  # tracé plus sophistiqué
```

— Effectuez la régression linéaire de `Vente` en fonction de `Prix`, et examinez l'objet obtenu (vous pouvez l'appeler *reg*) :

```
reg = lm(Vente~Prix, data = tabac)
# Pour utilser la commande lm, on doit indiquer le nom de la variable
# cible (ici Vente),le signe ~, le nom de la variable explicative (ici Prix)
# et le nom du tableau de données (ici tabac)
reg
```

— Que représentent les valeurs 109.4994 et -0.1822 qui sont affichées par R ? Vous pouvez les obtenir via la commande :

```
reg$coefficients
```

— Tracez la droite de régression par dessus le graphique précédent : `abline(reg, col = "red")`

— Tapez la commande `summary(reg)`, et observez ce qui est affiché à l'écran. Nous allons maintenant voir comment retrouver toutes les valeurs qui nous ont servi à faire l'exercice 1 du TD1.

1. les résidus associés au modèle e_i : `reg$residuals`. Pour afficher seulement 3 décimales, vous pouvez utiliser la commande `round` : `round(reg$residuals,3)`. Récupérez le 16ème résidu (année 2000).
2. les valeurs prédites par le modèle $\hat{y}_i$: `reg$fitted.values`
3. SCE_r : à l'aide de la commande `sum()` , qui calcule la somme des éléments d'un vecteur, écrire une commande qui calcule SCE_r
4. SCE_t : à l'aide de la commande `mean()` qui calcule la moyenne d'un vecteur, et de `sum()`, écrire une commande qui calcule SCE_t.
5. SCE_m : idem pour SCE_m
6. Que vaut le coefficient de détermination (R^2) de ce modèle ? Vous pouvez l'obtenir directement par `summary(reg)$r.squared`
7. l'estimation de la variance résiduelle : `summary(reg)$sigma^2`. Vous pouvez aussi la calculer à partir de SCE_r, vérifiez. (La commande `length(v)` permet de récupérer la longueur d'un vecteur `v`)
8. La commande `summary(reg)$coefficients` permet d'obtenir une matrice avec les coefficients du modèle en colonne 1, les écarts-types des estimations des coefficients en colonne 2, les valeurs associées aux tests de signification de Student en colonne 3, et les probabilités critiques de ces tests en colonne 4.
   Récuperez la valeur test pour $H_0 : \beta = 0$ (ligne 2, colonne 3) et comparez-la à la table de Student. Conclusion ?
9. La même conclusion peut être obtenue en regardant la probabilité critique associée (ligne 2, colonne 4). Lorsque cette probabilité est inférieure à 5%, alors on refuse H_0.
10. En utilisant SCE_m et SCE_r (calculés précédemment), determiner la valeur du test de signification du modèle complet (test de Fisher). Elle peut être aussi obtenue directement par `summary(reg)$fstatistic`. Vérifiez, et comparez à la table de Fisher.
11. Pour obtenir une prévision pour l'année 2017 (prix relatif 288.3), ainsi que les intervalles de confiance associés, il faut procéder comme suit :

    ```
    new = data.frame(288.3) # création du nouvel individu avec sa valeur
    de Prix
    colnames(new) = "Prix"  # label de la variable explicative
    predict(reg, new, interval = "confidence") # pour prédiction moyenne
    ```

    Vérifiez les valeurs obtenues en TD.

2 Régression linéaire multiple

Récuperez le fichier `materiau.txt` sur Teams, mettez-le dans votre répertoire de travail, et ouvrez-le. Il contient le tableau de données de l'exercice 2 du TD1. Importez les données :

```
mat<-read.table("materiau.txt",header=T)
mat
      Y X1  X2
1  37.8  4 4.0
2  22.5  4 3.6
3  17.1  3 3.1
...
```

— Effectuez la régression de Y sur X_1 puis de Y sur X_2 et sauvegarder ces modèles dans deux variables différentes.

— Pour effectuer une régression multiple, on utilise toujours la commande `lm`, mais en précisant quelles sont les variables explicatives :

```
regmul<-lm(Y~X1+X2,data=mat)    # on précise le nom des variables explicatives OU
regmulbis<-lm(Y~.,data=mat)     # on prend toutes les variables autres
                                que Y en utilisant .
```

On obtient un objet de type `lm` comme en régression simple. Effectuez la régression de Y sur $X1$ et $X2$, et servez-vous des 3 modèles obtenus pour répondre à toutes les questions du TD1 rapidement.