Analyse de la variance et régression linéaire simple

1 Analyse de la variance

On a vu en cours et en TD qu'il est nécessaire de faire beaucoup de calculs pour obtenir les résultats des tests. Avec R, c'est bien plus rapide. On commence par créer nos données sous forme d'une `data.frame`. Prenons par exemple l'exercice 1 du cours :

```
> adjuvant = rep(c("sans","alumine","calcium","phosphates"),c(5,6,4,5))
> taux = c(1,3,3,0,1,2,4,5,4,3,6,3,3,4,5,1,4,2,3,3)
> adjuvant = factor(adjuvant)
> anticorps = data.frame(adjuvant,taux)
```

Remarques :

- On peut effectuer le test de Bartlett pour tester l'égalité des variances à l'aide de la fonction `bartlett.test()`.
- Il est nécessaire d'utiliser la commande `options(contrasts=c("contr.sum","contr.poly"))` avant d'effectuer l'analyse de la variance.

1.1 Analyse de la variance à un facteur

Reprenons l'exemple ci-dessus. Les commandes dans ce cas sont :

```
> modele = lm(taux~adjuvant,data=anticorps)
> resultat = anova(modele)
> print(resultat)
Analysis of Variance Table

Response: taux
          Df Sum Sq Mean Sq F value  Pr(>F)
adjuvant   3  18.85  6.2833  3.9973 0.02664 *
Residuals 16  25.15  1.5719
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Remarque : Ici la p-valeur est appelée `Pr(>F)`.

1.2 Analyse de la variance à deux facteurs

En utilisant les notations du cours. Si on veut effectuer les deux tests $\mathcal{H}_{0,A}$ et $\mathcal{H}_{0,B}$, il faut écrire :

`valeurs~facteur1+facteur2`.

Et si on veut effectuer les trois tests $\mathcal{H}_{0,A}$, $\mathcal{H}_{0,B}$ et $\mathcal{H}_{0,AB}$, il faut écrire :

`valeurs~facteur1*facteur2`.

Prenons comme exemple l'exercice 4 du cours. Cela donnerait comme script :

```r
rm(list=ls())
graphics.off()

#Rentrer les données
mode_pres = rep(c("oral","ecrit"),each=12)
nature_mot = rep(c("familier","non familier"),each=6,times=2)
nombre = c(19,16,18,23,14,16,15,13,7,9,8,11,10,12,18,16,17,14,9,16,14,11,12,8)
mode_pres = factor(mode_pres)
nature_mot = factor(nature_mot)
memorisation = data.frame(mode_pres,nature_mot,nombre)

#Commande obligatoire avant l'analyse de la variance
options(contrasts=c("contr.sum","contr.poly"))

#Analyse de la variance avec + :
modele1 = lm(nombre~mode_pres+nature_mot,data=memorisation)
resultat1 = anova(modele1)
print(resultat1)

#Analyse de la variance avec * :
modele2 = lm(nombre~mode_pres*nature_mot,data=memorisation)
resultat2 = anova(modele2)
print(resultat2)
```

Une fois exécuté, on obtient :

```
Analysis of Variance Table

Response: nombre
           Df Sum Sq Mean Sq F value  Pr(>F)
mode_pres   1   6.00   6.000  0.5784 0.45539
nature_mot  1 150.00 150.000 14.4606 0.00104 **
Residuals  21 217.83  10.373
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Analysis of Variance Table

Response: nombre
                     Df  Sum Sq Mean Sq F value    Pr(>F)
mode_pres             1   6.000   6.000  0.6327 0.4357098
nature_mot            1 150.000 150.000 15.8172 0.0007422 ***
mode_pres:nature_mot  1  28.167  28.167  2.9701 0.1002416
Residuals            20 189.667   9.483
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

2 Régression linéaire simple

Voici un tableau qui résume les principales fonctions utilisées en régression linéaire simple :

Fonction	Description
`lm()`	permet de construire la droite de régression
`coeff()`	Donne les coefficients de la droite de régression

Fonction	Description
`summary()`	Donne les coefficients de la droite de régression et les tests associés
`fitted()`	Calcule les valeurs ajustées à partir de la droite de régression pour les observations initiales
`predict()`	Calcule les valeurs ajustées, les intervalles de confiance et les intervalles de précision à partir de la droite de régression pour n'importe quelle valeur
`confint()`	Calcule les intervalles de confiance pour la pente et l'ordonnée à l'origine de la droite de régression
`anova()`	Donne le tableau de l'analyse de la variance
`residuals()`	Calcule les résidus

3 Exercices

Exercice 1 : Des forestiers ont réalisé des plantations d'arbres en trois endroits. Plusieurs années plus tard, ils souhaitent savoir si la hauteur moyenne des arbres est identique dans les trois forêts. Chacune des forêts constitue une population et dans chacune d'entre elles, un échantillon d'arbres est tiré au sort. Puis la hauteur de chaque arbre est mesurée en mètres.

Arbre	Forêt	Hauteur (en m)	Arbre	Forêt	Hauteur (en m)
1	Forêt 1	23.4	4	Forêt 2	22.1
2	Forêt 1	24.4	5	Forêt 2	22.5
3	Forêt 1	24.6	6	Forêt 2	23.5
4	Forêt 1	24.9	1	Forêt 3	22.5
5	Forêt 1	25.0	2	Forêt 3	22.9
6	Forêt 1	26.2	3	Forêt 3	23.7
1	Forêt 2	18.9	4	Forêt 3	24.0
2	Forêt 2	21.1	5	Forêt 3	24.0
3	Forêt 2	21.1	6	Forêt 3	24.5

1. Tester la normalité des échantillons (fonction `shapiro.test()`).
2. Tester l'égalité des variances (fonction `bartlett.test()`).
3. Effectuer une analyse de la variance et conclure quand à l'influence de la forêt sur la taille des arbres.

Exercice 2 :(Exercice 5 du cours d'analyse de la variance)
Pour tester la fiabilité de 4 laboratoires d'analyse, on utilise 4 solutions ayant le même titre de glucose du sérum physiologique additionné de quantités variables de galactose. Chaque laboratoire reçoit un échantillon de chaque solution et fournit le résultat de ses mesures. L'ensemble des résultats, exprimés en grammes de glucose par litre de solution, est regroupé dans le tableau suivant :

Solution / Laboratoire	L1	L2	L3	L4
S1	1.05	1.15	1.08	1.13
S2	1.12	1.15	1.11	1.09
S3	1.02	1.10	1.04	1.05
S4	1.09	1.11	1.07	1.10

1. Au risque 5%, peut-on considérer que le choix du laboratoire a une influence sur la mesure du taux de glucose ?

2. Au risque 5%, peut-on considérer que la quantité de galactose présente dans la solution a une influence sur la mesure du taux de glucose ?

Exercice 3 : Le but de l'exercice est d'appliquer les fonctions pour la régression linéaire simple. On considère le jeu de données `iris` que l'on a déjà rencontré dans un précédent TP. On veut regarder le lien entre les longueurs des pétales (`Petal.Length`) et des sépales (`Sepal.Length`) pour l'espèce (`Species`) "virginica".

1. Extraire les valeurs souhaitées. On notera X les longueurs de sépales pour l'espèce "virginica" et Y les longueurs des pétales. Pour éviter des problèmes pour tracer les courbes, ordonner les valeurs à l'aide de la fonction `order()`. Afficher le nuage de points.

2. Donner les coefficients de la droite des moindres carrés (fonctions `lm()` et `coef()`).

3. Donner les ordonnées $\widehat{y_i}$ calculées par la droite des moindres carrés correspondant aux différentes valeurs des observations x_i.

4. Tracer la droite sur le même graphique (utiliser la fonction `abline()`).

5. A l'aide de la fonction `summary`, donner le coefficient de détermination R^2 ainsi que la valeur de l'estimation de σ^2.

6. Calculer les résidus et faites un test de Shapiro-Wilk sur ceux-ci. Pourquoi fait-on ce test ?

7. Tester $\mathcal{H}_0 : \alpha = 0$ contre $\mathcal{H}_1 : \alpha \neq 0$ ainsi que $\mathcal{H}_0 : \beta = 0$ contre $\mathcal{H}_1 : \beta \neq 0$.

8. A l'aide de la fonction `predict()`, calculer les intervalles de confiance et de prévision. Puis les tracer sur un même graphique (avec le nuage de points) à l'aide de la fonction `matlines()`.