Correction épreuve informatique Master AEM 2019

Exercice 1

1. Fonction nommé fonc1 qui retourne la valeur $y1$ de f dans le cas $f(x) = x.\ln(x)$.

```r
fonc1 <- function(x){
    y1 <- x*log(x)
    return(y1)
}
```

2. Les variables $i, n\ et\ y1$ sont des variables locales. Car ils sont déclarés à l'intérieur d'une fonction, c'est-à-dire hors de l'environnement de travail.
3. La fonction fonc2 qui permet de calculer $s = \sum_{i=1}^{n} f(x)$.

```r
fonc2 <- function(n){
    s <- 0
    for(i in 1:n){
        s <- s+fonc1(i)
    }
    return(s)
}
```

L'exécution de la fonction pour $n = 200$ donne le résultat suivant

```
> fonc2(200)
[1] 96496.87
>
```

4. La fonction fonc3 qui utilise la fonction fonc2 et qui retourne la première valeur de n dont $s > 1157$.

```
fonc3 <- function(){
     s <- 0
     i <- 1
     while(s <= 1157){
          s <-fonc2(i)
          i <- i + 1
     }
     return(i)
}
```

L'exécution de la fonction donne le résultat suivant :

```
> fonc3()
[1] 29
>
```

Exercice 2

Soit le modèle de régression linéaire multiple suivant : $Y = a_0 + a_1X_1 + a_2X_2 + a_3X_3 + \varepsilon$.

Sur les données suivantes :

Y	X_1	X_2	X_3
4	1	0.3	11
2.6	0.1	0.4	12
5.5	1.2	0.6	11.5
6.1	2	0.32	13
4.37	0.9	0.45	12.6
6.95	2.1	0.5	10.8
6.81	2	0.53	13.5

A) La méthode personnelle

D'abords on fait entrer les données et on crée la matrice X qui nous permettra d'effectuer les calculs.

```
Y <- c(4,2.6,5.5,6.1,4.37,6.95,6.81)
X1 <- c(1,0.1,1.2,2,0.9,2.1,2)
X2 <- c(0.3,0.4,0.6, 0.32,0.45,0.5,0.53)
X3 <- c(11,12,11.5,13,12.6,10.8,13.5)

X <- matrix(c(rep(1, length(Y)),X1,X2,X3), length(Y), 4)
```

1. Ecrire une fonction *estim* et retourne a_0, a_1, a_2 et a_3. Pour le calcul on applique la relation d'estimation économétrique $\hat{a} = (X^t \times X)^{-1} \times X^t \times Y$.

```
estim <- function(Y, X){
    a <- solve(t(X)%*%X)%*%t(X)%*%Y
    return(a)
}
```

L'exécution de la fonction donne le résultat suivant, chaque valeur représente respectivement $\widehat{a_0}$, $\widehat{a_1}, \widehat{a_2}, \widehat{a_3}$.

```
> estim(Y,X)
             [,1]
[1,]  1.005649858
[2,]  1.990946684
[3,]  3.589779593
[4,] -0.004190402
>
```

2. La fonction *testFisher* qui retourne est ce que le modèle est significatif ou non.

Le principe de cette fonction est le suivant : Dans un premier temps on calcul les coefficients du modèle avec la fonction *estim()* qu'on a créer dans la première question (*on peut aussi réeffectuer le calcul une deuxième fois).* Ensuite on détermine le *Y_estimee,* avec le quelle on peut calculer l'erreur de lestimation $e = y - \hat{y}$.

Ce qui suit est le calcul des sommes de carrés résiduels, expliqué et total. En appliquant la formule $R^2 = 1 - \frac{SCR}{SCT}$ on obtient le coefficient de détermination.

Avec la formule $\frac{R^2/_k}{(1-R^2)/_{n-k-1}}$ on obtient le coefficient de Fisher empirique. Et par la fonction prédéfinit *qf()* on détermine le coefficient de Fisher théorique, elle prend comme argument respectivement ***0.95*** *(qui correspond à* $1 - \alpha$*, avec* $\alpha = 0.05$*),* ***3*** (qui correspond au degrés de liberté du numérateur) et ***3*** (le degré de liberté du dénominateur).

Finalement avec une condition *if* on vérifie si l'hypothèse H_1 est vérifier, dans ce cas on stock l'interprétation suivante *« le modèle est globalement significative »* dans une variable mot. Sinon on stock *« le modèle n'est PAS globalement significative ».* Cette variable est retournée.

Le code de la fonction est la suivant, il peut sembler compliqué au début mais c'est tout simplement la formulation des étapes précédentes sous le langage R.

On doit préciser que c'est n'est pas la seule façon pour coder cette fonction, une multitude de façon existe. Le plus important que ça donne le même résultat.

```r
testFisher <- function(Y, X){
    # Dabords on estime les parametres du modele
    a <- estim(Y, X)
    # On calcul le Y estimée.
    Y_estimee <- X%*%a

    # Erreur de l'estimation.
    Erreur_estimation <- Y - Y_estimee

    # Somme des carrees residuels SCR.
    SCR <- as.numeric(t(Erreur_estimation)%*%Erreur_estimation)

    # Somme des carrees expliques SCE.
    SCE <- sum((Y_estimee - mean(Y))^2)

    # Somme des carree total SCT = SCE + SCR.
    SCT = round(SCR + SCE,3)

    # Coefficient determination
    R2 <- 1 - (SCR/SCT)

    # On calcule le coefficient de Fisher
    coeff_Fisher_empirique <- ( R2/(ncol(X)-1) ) / ( (1-R2)/(length(Y)-length(a)) )

    # on genere le coefficient de fisher theorique
    coeff_Fisher_theorique <- qf(0.95, ncol(X)-1, length(Y)-length(a))

    if(coeff_Fisher_empirique > coeff_Fisher_theorique){
        mot <- " Le modéle est globalement significative"
    }else{
        mot <- " Le modéle n'est PAS globalement significative"
    }
    return(mot)
}
```

Ensuite on exécute notre fonction en lui donnant comme argument le vecteur Y et la matrice X.

```r
> testFisher(Y, X)
[1] " Le modéle est globalement significative"
>
```

Comme on peut le voir le test donne que le modèle est globalement significatif.

3. Fonction testStudent

```
testStudent <- function(Y,X){
    a <- estim(Y,X)

    var_erreur <- as.numeric(t(Y-X%*%a) %*% (Y-X%*%a))/(length(Y)-length(a))

    mat_var_cov <- var_erreur*solve(t(X)%*%X)
    var_estim <- diag(mat_var_cov)

    ts <- c()
    for(i in 1:length(a)){
        ts[i] <- abs(a[i])/sqrt(var_estim[i])
    }

    mot <- c()
    for( i in 1:length(a)){
        if(ts[i]>4.303){
            mot[i] <- "le coefficient est different de 0"
        }
        else{
        mot[i] <- "le coefficient n'est pas different de 0"
        }
    }
    return(mot)
}
```

Cette fonction qui permet d'effectuer le test de Student sur les paramètres de la fonction. Elle prend comme argument le vecteur Y et la matrice X. Dans un premier temps à l'aide de la fonction *estim*() on calcul les coefficients du modèle, ensuite on détermine la variance de l'erreur (on doit convertir le résultat de calcul avec *as.numeric()* pour qu'on puisse travailler avec). L'étape suivante consiste à déterminer la matrice de variance-covariance et extrait ensuite la diagonale principale qui contient la variance des estimateurs.

Ensuite avec une boucle ***for*** on calcule le t de Student de chaque coefficient. Et avec une deuxième boucle ***for*** on vérifie avec une condition ***if*** si l'hypothèse est vérifiée. Finalement une liste contenant l'interprétation de chaque coefficient est retournée, comme on peut le voir dans l'exécution suivante.

4.

```
> testStudent(Y,X)
[1] "le coefficient est different de 0"
[2] "le coefficient est different de 0"
[3] "le coefficient est different de 0"
[4] "le coefficient n'est pas different de 0"
>
>
```

Comme on peut le voir durant l'exécution, seul le coefficient $\widehat{a_3}$ n'est pas significativement différent de 0.

B) <u>La méthode automatique</u>

1. Pour le calcul des coefficients on utilise la fonction prédéfinit dans T ***lm().*** Elle prend comme argument la variable à explique et la somme des variables explicatives comme on peut le voir dans l'execution suivante :

```
> lm(Y~X1+X2+X3)

Call:
lm(formula = Y ~ X1 + X2 + X3)

Coefficients:
(Intercept)           X1           X2           X3
    1.00565      1.99095      3.58978     -0.00419

>
```

On obtient les mêmes résultats que dans la première partie.

2. Pour le test de Fisher on utilise ***summary()*** dans la quelle on donne comme argument ***lm().*** Comme on peut voir dans l'exécution suivante :

```
> summary(lm(Y~X1+X2+X3))

Call:
lm(formula = Y ~ X1 + X2 + X3)

Residuals:
        1         2         3         4         5         6         7 
-0.027436  0.009628 -0.000464  0.018203  0.009896  0.013729 -0.023556 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.00565    0.13251   7.590  0.00475 ** 
X1           1.99095    0.01482 134.360 9.09e-07 ***
X2           3.58978    0.09798  36.639 4.47e-05 ***
X3          -0.00419    0.01056  -0.397  0.71802    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.02594 on 3 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9997 
F-statistic:  7650 on 3 and 3 DF,  p-value: 2.536e-06
```

On remarque que la ***p-value*** est inférieur à 5% donc le modèle est globalement significatif.

De même on peut remarquer que les p-value des coefficients du modèle dont tous inférieur à 5%, sauf pour $\hat{a}_3$ *qui n'est pas significativement différent de 0 ce qui confirme les résultats trouver précédemment.*