FSML-2 Exam

Moussa GRICHE, DS-S22 Student

2022-10-21

Exercice n°:1

X and Y 2 IRV

$X \rightsquigarrow Exp(\mu) : f_X(x) = \mu e^{-\mu x}$

$Y \rightsquigarrow Exp(\lambda) : f_Y(y) = \lambda e^{-\lambda y}$

- Computing $P(X > Y)$

let's compute first the CDF of Y:

$$F_Y(y) = P(Y < y) = \int_{-\infty}^{y} f_Y(t)\,dt = \int_0^y f_Y(t)\,dt$$

$$= \int_0^y \lambda e^{-\lambda t}\,dt$$

$$= \left[-e^{-\lambda t}\right]_0^y$$

$$= -e^{-\lambda y} - (-e^{0})$$

$$= 1 - e^{-\lambda y}$$

$$P(X > Y) = \int_0^{+\infty}\int_y^{+\infty} f_X(x)\, f_Y(y)\,dx\,dy$$

$$= \int_0^{+\infty}\int_y^{+\infty} \lambda e^{-\lambda y} \mu e^{-\mu x}\,dx\,dy$$

$$= \int_0^{+\infty} \lambda e^{-\lambda y}\left[-e^{-\mu x}\right]_y^{+\infty} dy$$

$$= \int_0^{+\infty} \lambda e^{-\lambda y} \cdot e^{-\lambda y}\,dy$$

$$= \lambda \int_0^{+\infty} e^{-(\lambda+\mu)y}\, dy$$

$$= \lambda \left[\frac{-1}{\lambda+\mu} e^{-(\lambda+\mu)y} \right]_0^{+\infty}$$

$$= \lambda \cdot \left(\frac{1}{\lambda+\mu} \right)$$

$$\Rightarrow P(X > Y) = \frac{\lambda}{\lambda+\mu}$$

Exercice n°:2

let F the function definied by

$$\forall x \in \mathbb{R},\ F(x) = \frac{1}{1+\exp(-x)}$$

1.1/ Showing that F is a distributed function :

a) F must be an increasing function:

$$F(x) = \frac{1}{1+\exp(-x)} \Rightarrow F'(x) = \frac{\exp(-x)}{(1+\exp(-x))^2}$$

$F'(x) > 0$ because $\exp(-x) > 0$ and $(1+\exp(-x))^2 > 0$

$\Rightarrow F(x)$ is an increasing function

b) $\lim\limits_{x \to -\infty} F(x)$ must be equal to 0:

$$\lim_{x \to -\infty} F(x) = \lim_{x \to -\infty} \frac{1}{1+\exp(-x)} = \frac{1}{1+\exp(+\infty)} = 0$$

c) $\lim\limits_{x \to +\infty} F(x)$ must be equal to 1:

$$\lim_{x \to +\infty} F(x) = \lim_{x \to +\infty} \frac{1}{1+\exp(-x)} = \frac{1}{1+0} = 1$$

d) $F(x)$ must be a right continuous function :

$F(x) = \frac{1}{1+\exp(-x)}$ is continuous in $\mathbb{R}$, so it is right continuous

The four conditions hold, so $F(x)$ is a distributed function.

1.2/ The density function of X:

$$F_X(x) = \frac{1}{1+\exp(-x)}$$

$$f_X(x) = \frac{\partial F_X(x)}{\partial x} = \frac{\partial}{\partial x} \frac{1}{1+\exp(-x)} = \frac{\exp(-x)}{(1+\exp(-x))^2}$$

So, $$f_X(x) = \frac{\exp(-x)}{(1+\exp(-x))^2}$$

2/ $Y \sim F_Y(y)$ i RV
$X \sim F_X(x)$

let $Z = \min(X, Y)$

Determine the density function of Z:

$$f_Z(z) = \frac{\partial F_Z(z)}{\partial z}$$

let compute $F_Z(z)$

$$\begin{aligned}
F_Z(z) = P(Z \leq z) &= P(\min(X,Y) \leq z) \\
&= 1 - P(\min(X,Y) > z) \\
&= 1 - P(X > z \text{ and } Y > z) \\
&= 1 - P(X > z) \cdot P(Y > z) \quad \text{because } X \text{ and } Y \text{ are independant} \\
&= 1 - \big((1 - P(X \leq z)) \cdot (1 - P(Y \leq z))\big) \\
&= 1 - \big((1 - F_X(z))(1 - F_Y(z))\big) \\
&= 1 - \big(1 - F_Y(z) - F_X(z) + F_X(z) F_Y(z)\big) \\
&= F_Y(z) + F_X(z) - F_X(z) F_Y(z) \\
&= \frac{1}{1+\exp(-z)} + \frac{1}{1+\exp(-z)} - \frac{1}{1+\exp(-z)} \cdot \frac{1}{1+\exp(-z)}
\end{aligned}$$

$$F_Z(z) = \frac{2}{1+\exp(-z)} - \frac{1}{(1+\exp(-z))^2}$$

$$f_Z(z) = \frac{\partial F_Z(z)}{\partial z} = \frac{\partial}{\partial z}\left(\frac{2}{1+\exp(-z)} - \frac{1}{(1+\exp(-z))^2}\right)$$

$$f_Z(z) = \frac{2\exp(-z)}{(1+\exp(-z))^2} - \frac{2\exp(-z)(1+\exp(-z))}{(1+\exp(-z))^4}$$

$$f_Z(z) = \frac{2\exp(-z)}{(1+\exp(-z))^2} - \frac{2\exp(-z)}{(1+\exp(-z))^3}$$

$$f_Z(z) = \frac{2\exp(-z)(1+\exp(-z)-1)}{(1+\exp(-z))^3}$$

$$f_Z(z) = \frac{2\exp(-2z)}{(1+\exp(-z))^3}$$

Exercice n°:3

Upload Data

```
Fund <- read.table(file="C:/Users/lahor/Desktop/DSTI/Advenced Statistic/Project/VSURF_Toys-and-Fund-data
```

```
head(Fund)
```

```
##           V1         V2
## 1 -3.554288 11.1514000
## 2 -1.088076  5.2602900
## 3 -6.594419 19.6295100
## 4  1.774103 -4.5525900
## 5  1.498014 -0.9290205
## 6 -2.229028 10.7375000
```

```
library(dplyr, quietly = TRUE)
```

```
## Warning: le package 'dplyr' a été compilé avec la version R 4.2.1

##
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':
##
##     filter, lag

## Les objets suivants sont masqués depuis 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
Fund <- Fund %>% dplyr::rename(Y = V1, X = V2)
```

```
head(Fund)
```

```
##            Y          X
## 1 -3.554288 11.1514000
## 2 -1.088076  5.2602900
## 3 -6.594419 19.6295100
## 4  1.774103 -4.5525900
## 5  1.498014 -0.9290205
## 6 -2.229028 10.7375000
```

```
summary(Fund$Y)
```

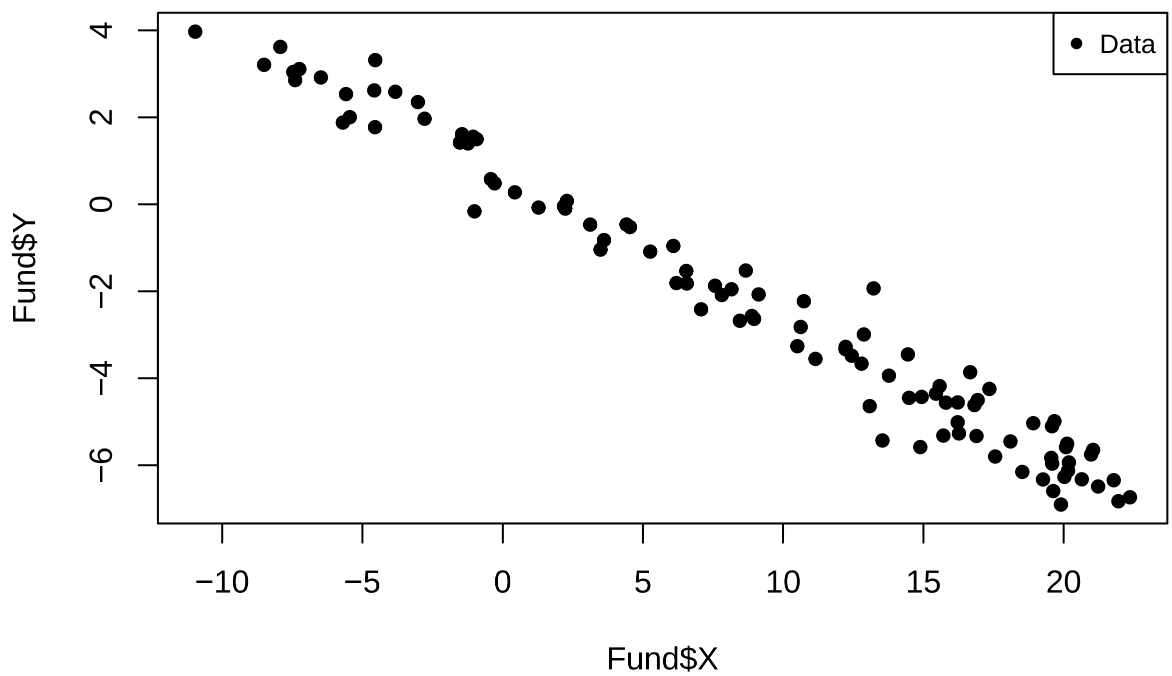
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -6.90428 -5.14459 -2.74931 -2.34046 -0.01529  3.96989
```

```
summary(Fund$X)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -10.962    1.067   10.681    9.038   16.903   22.366
```

Plotting the data

```
plot(Fund$Y~Fund$X, pch = 16)
legend("topright", legend=c("Data" ),col=c("black"),pch=16, cex=0.8)
```



1.1 Computing the estimators of the linear model:

The regression model that explain the relationship between the response variable (Y) and the explanatory variable (X) is writing as following:

$$Y = \beta_0 + \beta_1 X + \varepsilon_i$$

where:

$$\varepsilon_i \; is \; the \; nose \; and \; assume \; that \; respects \; the \; following \; assumptions$$

1-

$$\forall i \in \{1, ..., n\}\,,\; E\left[\varepsilon_i\right] = 0$$

2-

$$\forall i \in \{1, ..., n\}\,,\; V\left[\varepsilon_i\right] = \sigma^2$$

3-

$$\forall i \in \{1, ..., n\}\,, \forall k \in \{1, ..., n\}\,, if\;\; i \neq k \Rightarrow\;\; Cov\left(\varepsilon_i, \varepsilon_k\right) = 0$$

4-

$$\forall i \in \{1, ..., n\}\,,\; \varepsilon_i \sim N(0, \sigma^2)$$

To fit the model, we should estimate the parameters B0 and B1.

We could estimate B0 and B1 with two main methods:

1- Ordinary Least Square (OLS)

2- Maximum Likelihood

In our case, I will use the OLS method to estimate these parameters. Using OLS, the formula to estimate B0 and B1 is done as following:

$$\hat{\beta_1} = \frac{\sum_{i=1}^{n}(y_i - \bar{y_n})(x_i - \bar{x_n})}{\sum_{i=1}^{n}(x_i - \bar{x_n})^2}$$
$$\hat{\beta_0} = \bar{y_n} - \hat{\beta_1}\bar{x_n}$$

Where:

$$\bar{X_n} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\bar{Y_n} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

```
n <- length(Fund$X)
X_n.barre <- (1/n) * sum(Fund$X)
Y_n.barre <- (1/n) * sum(Fund$Y)
```

Paramaters' estimation:

```
B_1.hate = sum((Fund$Y-Y_n.barre) *(Fund$X-X_n.barre)) / sum((Fund$X-X_n.barre)^2)
B_0.hate = Y_n.barre - B_1.hate*X_n.barre

cat("Coefficients:
               Estimate \n",
    "(Intercepte) ", B_0.hate, "\n",
    "X            ", B_1.hate)
```

```
## Coefficients:
##                  Estimate
##  (Intercepte)  0.6606614
##  X            -0.3320686
```

So, our model can be writing as following:

$$Y = 0.6606614 - 0.3320686X + \varepsilon_i$$

Computing the fitted values of Y:

```
Pred.Y <- 0.6606614 -0.3320686*Fund$X
```

Mean Squared Error:

```
MSE <- sum((Fund$Y-Pred.Y)^2)/n

cat("The Mean Squared Error of the fitted model equal to:",MSE)
```

```
## The Mean Squared Error of the fitted model equal to: 0.2835037
```

Parameters' Confidence Intervals:

$$[\hat{\beta_0} - \hat{\sigma_n}\sqrt{(\frac{1}{n} + \frac{\bar{X_n}^2}{\sum(x_i - \bar{X_n})^2})}t_{1-\frac{\alpha}{2},n-2} \ , \ \hat{\beta_0} + \hat{\sigma_n}\sqrt{(\frac{1}{n} + \frac{\bar{X_n}^2}{\sum(x_i - \bar{X_n})^2})}t_{1-\frac{\alpha}{2},n-2}]$$

Now, we should calculate the confidence interval for B1 in order to know if we could assume that B1 could take the value 0, so there is no linear relationship between Y and X.

The confidence interval of B1 can be writing as following:

$$[\hat{\beta_1} - \frac{\hat{\sigma_n}}{\sqrt{\sum(x_i - \bar{X_n})^2}}t_{1-\frac{\alpha}{2},n-2}, \hat{\beta_1} + \frac{\hat{\sigma_n}}{\sqrt{\sum(x_i - \bar{X_n})^2}}t_{1-\frac{\alpha}{2},n-2}]$$

Where:

$$\hat{\sigma_n} = \sqrt{\frac{\sum \hat{\varepsilon_i}^2}{(n-2)}} t_{1-\frac{\alpha}{2},n-2} \ is \ the \ student \ test \ with \ 1 - \frac{\alpha}{2} \ and \ n-2 \ degree \ of \ fredom$$

Calculating sigma_n.hate

```
sigma_n.hate <- sqrt(sum((Fund$Y-Pred.Y)^2)/98)
cat("The unbiased estimator of sigma_n is equal to:",sigma_n.hate)
```

```
## The unbiased estimator of sigma_n is equal to: 0.5378564
```

Calculating the confidence interval for B0

```
CI_B0.Lower <- B_0.hate - sigma_n.hate * sqrt((1/n) + ((X_n.barre)^2/ sum((Fund$X-X_n.barre)^2))) *
  qt(0.975, (n-2))
CI_B0.Upper <- B_0.hate + sigma_n.hate * sqrt((1/n) + ((X_n.barre)^2/ sum((Fund$X-X_n.barre)^2))) *
  qt(0.975, (n-2))
cat("The confidence interval for B0 is : (", CI_B0.Lower , ",", CI_B0.Upper, ")" )
```

```
## The confidence interval for B0 is : ( 0.5120103 , 0.8093125 )
```

Calculating the confidence interval for B1

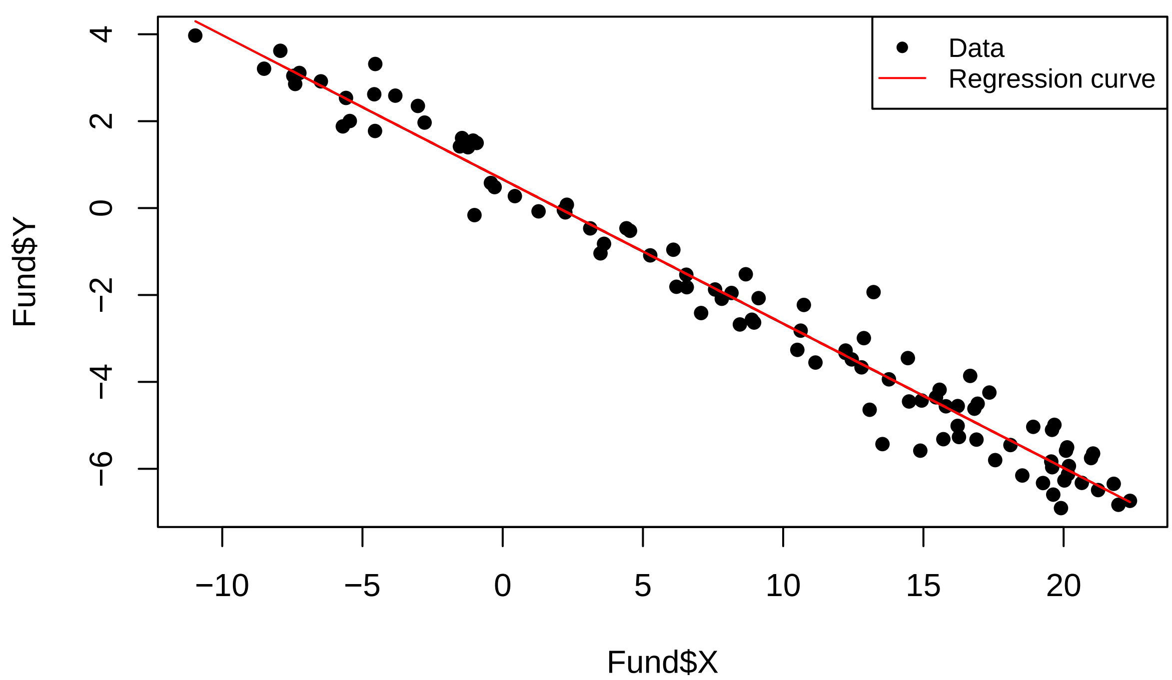
```
CI_B1.Lower <- B_1.hate - sigma_n.hate / sqrt(sum((Fund$X-X_n.barre)^2)) *
  qt(0.975, (n-2))
CI_B1.Upper <- B_1.hate + sigma_n.hate / sqrt(sum((Fund$X-X_n.barre)^2)) *
  qt(0.975, (n-2))
cat("The confidence interval for B1 is : (", CI_B1.Lower , ",", CI_B1.Upper, ")" )
```

```
## The confidence interval for B1 is : ( -0.3435166 , -0.3206207 )
```

1.2- The representation of the cloud of points and of the regression curve:

Plotting the fitted model:

```
plot(Fund$Y~Fund$X, pch = 16)
lines(Fund$X, Pred.Y, type = "l", col="red")
legend("topright", legend=c("Data","Regression curve" ),col=c("black","red"),
       lty = c(NA,1),pch=c(16, NA), cex=0.8)
```



1.3- The curves associated to the confidence interval for the prediction:

Calculating the confidence interval for the predicted values :

The confidence interval of the predicted values could be calculated using the following formula:

$$[\hat{y}_h - \sqrt{MSE \times (1 + \frac{1}{n} + \frac{(x_h - \bar{X_n})^2}{\sum(x_i - \bar{X_n})^2})}\ t_{1-\frac{\alpha}{2},n-2}\ , \hat{y}_h + \sqrt{MSE \times (1 + \frac{1}{n} + \frac{(x_h - \bar{X_n})^2}{\sum(x_i - \bar{X_n})^2})}\ t_{1-\frac{\alpha}{2},n-2}]$$

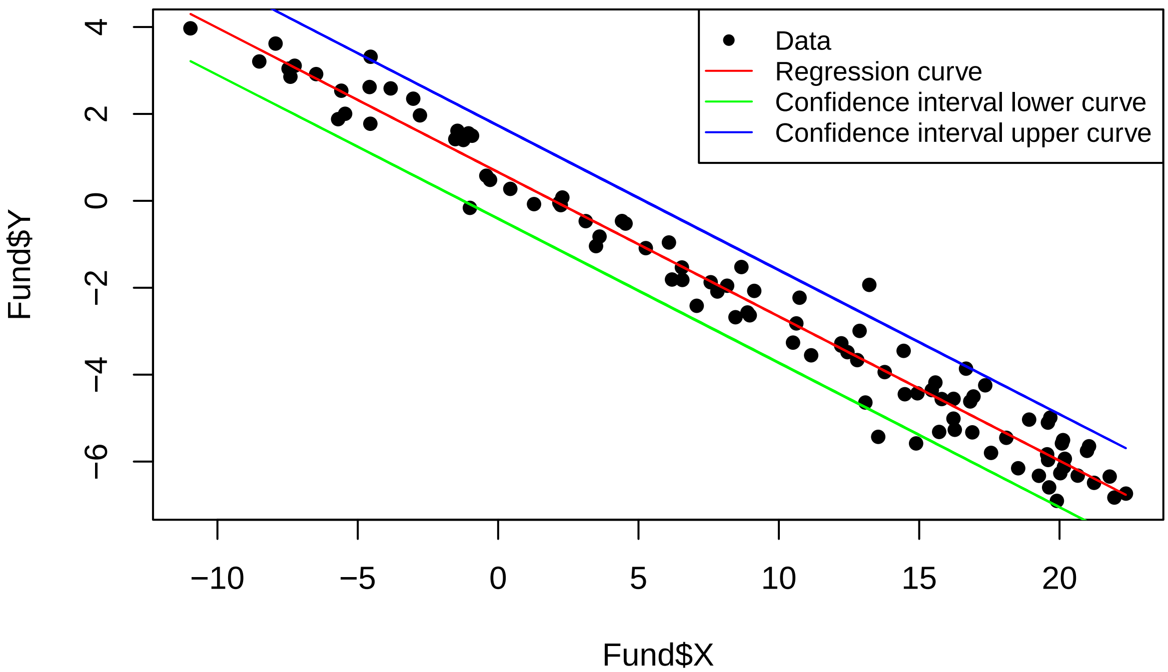
Where:

```
Pred.Y.CI_Lower <- Pred.Y - sqrt ( MSE * (1 + (1/n) +
                                        ((Fund$X - X_n.barre)^2/
                                          sum((Fund$X-X_n.barre)^2)))) * qt(0.975, (n-2))
Pred.Y.CI_Upper <- Pred.Y + sqrt ( MSE * (1 + (1/n) +
                                        ((Fund$X - X_n.barre)^2/
                                          sum((Fund$X-X_n.barre)^2)))) * qt(0.975, (n-2))


Pred.Y.CI <- matrix(nrow = n, ncol = 3, dimnames =
                      list(c(1:n),c("Pred.Y","Pred.Y.CI_Lower","Pred.Y.CI_Upper")))
Pred.Y.CI[,1] <- Pred.Y
Pred.Y.CI[,2] <- Pred.Y.CI_Lower
Pred.Y.CI[,3] <- Pred.Y.CI_Upper

head(Pred.Y.CI)
```

```
##       Pred.Y Pred.Y.CI_Lower Pred.Y.CI_Upper
## 1 -3.0423684     -4.10453962     -1.98019716
## 2 -1.0861157     -2.14887933     -0.02335214
## 3 -5.8576825     -6.92634649     -4.78901852
## 4  2.1724336      1.09942136      3.24544582
## 5  0.9691599     -0.09873143      2.03705131
## 6 -2.9049252     -3.96700099     -1.84284940
```

Plotting the curves associated to the confidence interval for the prediction:

```
plot(Fund$Y~Fund$X, pch = 16)
lines(Fund$X, Pred.Y, type = "l", col="red")
lines(Fund$X, Pred.Y.CI_Lower, type = "l" , col="green")
lines(Fund$X, Pred.Y.CI_Upper, type = "l" , col="blue")
legend("topright", legend=c("Data","Regression curve", "Confidence interval lower curve",
                            "Confidence interval upper curve" ),col=c("black","red","green",
                                                                       "blue"),lty=c(NA,1,1,1),
       pch=c(16, NA, NA, NA), cex=0.8)
```



2- The Fisher test:

```
LSE_F = sum((Fund$Y-Pred.Y)^2)     #Least squared error of the fitted model
LSE_R = sum((Fund$Y-Y_n.barre)^2) #Least squared error of the reduced model (yi = B0 + epsi)

MSR = (LSE_R - LSE_F)/((n-1)-(n-2))
MSE = LSE_F/(n-2)

F_value = MSR/MSE
P_value = df(F_value, (n-1)-(n-2), (n-2))
cat("The P-Value is equal to:", P_value)
```

```
## The P-Value is equal to: 3.346974e-79
```

The P-value is to small (<0.05), so we could reject the null hypothesis (h0) and accept the alternative hypothesis (H1). Thus, this means that exist an influence of the explanatory variable on the response.