Université du Havre

Faculté des sciences et techniques

TP : Fouille de données

Régression linaire simple et multiple (Epuisement des matériaux)

Réaliser par :

TOUCHENE Mohamed Amine
ELMI IBRAHIM Youssouf
LO Mouhamadou Moustapha

Enseignant :

Mr : FOURNIER Dominique

Master IWOCS

Année universitaire 2017 / 2018

Introduction

Le TP a été réalisé en vue de faire une étude sur l’épuisement des matériaux et plus précisément sur l’acier. Au moyen du langage **R** et du fichier **DonneesAcier.csv** donné**,** nous allons tenter de donner une estimation sur différentes valeurs.

A notre disposition, un échantillon de résultat qui va nous permettre de mesurer les deux principaux coefficients nécessaire pour l’étude de la ductilité des matériaux en question.

$\mathcal{E}'_f$ *:* Représente le coefficient de ductilité en fatigue ;
Σ'_f *:* Représente le coefficient de résistance en fatigue.

Résultats et tests

Pour débuter, nous allons faire une lecture du fichier de données **DonneesAcier.csv**.

```
R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

> projet=read.csv2("donneesAcier.csv")
> projet
   codeAcier numTest   E BHN RA Sigma_e Sigma_u Epsilon_f Epsilon_prime_f
1       1141       1 216 223 57     457     771      0.84            0.26
2       1141       2 227 277 59     814     925      0.89            0.31
3        220       3 220 199 53     418     695      0.76            0.26
4       1141       4 217 241 54     602     802      0.78            0.36
5       1141       5 214 217 49     450     725      0.67            0.43
6       1141       6 215 252 58     610     797      0.87            0.53
7       1141       7 220 229 47     493     789      0.63            0.60
8       1038       8 201 163 54     331     582      0.78            0.31
9       1038       9 219 185 53     359     652      0.76            0.20
10      1038      10 219 195 67     410     649      1.11            0.23
```

La commande suivante nous donne la matrice de corrélation (notons que les deux premières colonnes sont à omettre, à savoir celles relatives au numéro de test et au code respectivement).

```
> cor(projet[,3:10])
                         E        BHN         RA    Sigma_e    Sigma_u
E              1.000000000 -0.3518188  0.1427242 -0.4542611 -0.3787762
BHN           -0.351818764  1.0000000 -0.5075624  0.9644053  0.9800446
RA             0.142724190 -0.5075624  1.0000000 -0.4403975 -0.5300379
Sigma_e       -0.454261101  0.9644053 -0.4403975  1.0000000  0.9705248
Sigma_u       -0.378776243  0.9800446 -0.5300379  0.9705248  1.0000000
Epsilon_f      0.125542060 -0.5388124  0.9827108 -0.4655494 -0.5535625
Epsilon_prime_f 0.002866055 -0.1239751  0.1962201 -0.1202111 -0.1631696
Sigma_prime_f -0.331031610  0.9361995 -0.5472082  0.8964122  0.9463922
               Epsilon_f Epsilon_prime_f Sigma_prime_f
E              0.1255421     0.002866055    -0.3310316
BHN           -0.5388124    -0.123975139     0.9361995
RA             0.9827108     0.196220131    -0.5472082
Sigma_e       -0.4655494    -0.120211063     0.8964122
Sigma_u       -0.5535625    -0.163169595     0.9463922
Epsilon_f      1.0000000     0.162873180    -0.5660041
Epsilon_prime_f 0.1628732    1.000000000    -0.1259600
Sigma_prime_f -0.5660041    -0.125959958     1.0000000
```

Tests sur Σ'_f et $\mathcal{E}'_f$

Tests sur Σ'_f

Nous avons les variables qui sont fortement corrélé avec Σ'_f :

Σ_e : (0.8964122) ;
BHN : (0.9361995) ;
Σ_u : (0.9463922).

```
> mc [,8]
          E           BHN            RA       Sigma_e       Sigma_u
 -0.3310316     0.9361995    -0.5472082     0.8964122     0.9463922
  Epsilon_f Epsilon_prime_f Sigma_prime_f
 -0.5660041    -0.1259600     1.0000000
```

<u>Test pour Σ_e</u>

```
> summary(lm(Sigma_prime_f~ Sigma_e, projet))

Call:
lm(formula = Sigma_prime_f ~ Sigma_e, data = projet)

Residuals:
    Min      1Q  Median      3Q     Max
-506.38 -166.25  -40.31   92.53  672.30

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 604.35989   57.97588   10.42 5.84e-16 ***
Sigma_e       0.93085    0.05462   17.04  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 251.2 on 71 degrees of freedom
Multiple R-squared:  0.8036,    Adjusted R-squared:  0.8008
F-statistic: 290.4 on 1 and 71 DF,  p-value: < 2.2e-16
```

Remarque :

Le test est pertinent (***) :
F statistique de 290.4
p−value :< 2.2e−16
Moyenne de résidus de 251.2
r1 de 0.8036
r2 de 0.8008

En cas de régression multiple, le chois doit être porté sur une variable ayant une faible corrélation avec $\boldsymbol{\Sigma_e}$ vue que l'ajout d'une variable ayant une forte corrélation avec $\boldsymbol{\Sigma_e}$ ne va pas provoquer un resultat meilleur*.*

```
> mc [,4]
           E          BHN           RA      Sigma_e      Sigma_u
  -0.4542611    0.9644053   -0.4403975    1.0000000    0.9705248
   Epsilon_f Epsilon_prime_f Sigma_prime_f
  -0.4655494   -0.1202111    0.8964122
```

Prenons une variable ayant une corrélation faible mais aussi négative, soit $\boldsymbol{\varepsilon_f}$*.*

```
> summary(lm(Sigma_prime_f~ Sigma_e+Epsilon_f, projet))

Call:
lm(formula = Sigma_prime_f ~ Sigma_e + Epsilon_f, data = projet)

Residuals:
    Min      1Q  Median      3Q     Max
-426.64 -136.79  -40.91   99.70  671.30

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  925.11499  108.07443   8.560  1.7e-12 ***
Sigma_e        0.83908    0.05752  14.588  < 2e-16 ***
Epsilon_f   -332.75903   97.10119  -3.427  0.00103 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 234.1 on 70 degrees of freedom
Multiple R-squared:  0.8318,	Adjusted R-squared:  0.827
F-statistic: 173.1 on 2 and 70 DF,  p-value: < 2.2e-16
```

L'amélioration est considérable (en ce qui concerne r1 et r2) donc cette régression multiple est potentiellement bonne. Or l'ajout d'une variable provoque une baisse de F-statistique.

<u>Test pour *BHN*</u>

```
> mc [,2]
           E          BHN           RA      Sigma_e      Sigma_u
  -0.3518188    1.0000000   -0.5075624    0.9644053    0.9800446
   Epsilon_f Epsilon_prime_f Sigma_prime_f
  -0.5388124   -0.1239751    0.9361995
```

Le test actuel manifeste de meilleurs résultats par rapport au précédent.

```
> summary(lm(Sigma_prime_f~ BHN, projet))

Call:
lm(formula = Sigma_prime_f ~ BHN, data = projet)

Residuals:
    Min      1Q  Median      3Q     Max 
-361.51 -143.24  -16.89   98.31  645.10 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 249.9850    58.5694   4.268 5.99e-05 ***
BHN           4.0384     0.1799  22.445  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 199.2 on 71 degrees of freedom
Multiple R-squared:  0.8765,	Adjusted R-squared:  0.8747 
F-statistic: 503.8 on 1 and 71 DF,  p-value: < 2.2e-16
```

La régression multiple n'apporte pas rien d'intéressant à notre model. Qu'allons nous avoir avec ***Σ'u***

Test pour ***Σ'u***

```
> summary(lm(Sigma_prime_f~ Sigma_u, projet))

Call:
lm(formula = Sigma_prime_f ~ Sigma_u, data = projet)

Residuals:
    Min      1Q  Median      3Q     Max 
-322.72 -135.33  -11.60   92.76  565.55 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 353.0195    49.5474   7.125 6.93e-10 ***
Sigma_u       1.0023     0.0406  24.687  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 183.1 on 71 degrees of freedom
Multiple R-squared:  0.8957,	Adjusted R-squared:  0.8942 
F-statistic: 609.5 on 1 and 71 DF,  p-value: < 2.2e-16
```

La régression simple avec ***Σ'u*** est très pertinente mais qu'aurons nous avec une régression multiple avec une variable à faible corrélation avec ***Σ'u***, soit ***Ɛ'f*** .

```
> summary(lm(Sigma_prime_f~ Sigma_u+Epsilon_prime_f, projet))

Call:
lm(formula = Sigma_prime_f ~ Sigma_u + Epsilon_prime_f, data = projet)

Residuals:
    Min      1Q  Median      3Q     Max 
-313.52 -124.05  -14.87  100.54  561.51 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)     323.58816   63.31780   5.111 2.67e-06 ***
Sigma_u           1.00736    0.04128  24.404  < 2e-16 ***
Epsilon_prime_f  49.73131   66.28855   0.750    0.456    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 183.6 on 70 degrees of freedom
Multiple R-squared:  0.8965,	Adjusted R-squared:  0.8935 
F-statistic: 303.1 on 2 and 70 DF,  p-value: < 2.2e-16
```

On voit bien qu'aucune amélioration n'a été portée.

Le modèle élu

A l'issus de ces résultats, il est claire que la régression linéaire simple avec Σ_u est le modèle le plus adéquat pour une régression linéaire sur Σ'_f.

```
> regression = lm(Sigma_prime_f ~ Sigma_u, projet)
> regression

Call:
lm(formula = Sigma_prime_f ~ Sigma_u, data = projet)

Coefficients:
(Intercept)      Sigma_u
    353.019        1.002
```

La formule du model est la suivante :

$$\boxed{\Sigma'_f = 1.002\ \Sigma_u + 353.019}$$

Tests sur $\mathcal{E}'_f$

```
> mc [,7]
           E            BHN             RA        Sigma_e        Sigma_u
 0.002866055   -0.123975139    0.196220131   -0.120211063   -0.163169595
   Epsilon_f Epsilon_prime_f   Sigma_prime_f
 0.162873180     1.000000000    -0.125959958
```

RA est la variable la plus corrélé avec $\mathcal{E}'_f$ donc la régression linéaire simple va être avec cette variable et on aura :

```
> summary(lm(Epsilon_prime_f~RA, projet))

Call:
lm(formula = Epsilon_prime_f ~ RA, data = projet)

Residuals:
     Min       1Q   Median       3Q      Max
-0.50623 -0.23024 -0.05425  0.16575  1.45575

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.286277   0.121088   2.364   0.0208 *
RA          0.003999   0.002372   1.686   0.0962 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3268 on 71 degrees of freedom
Multiple R-squared:  0.0385,    Adjusted R-squared:  0.02496
F-statistic: 2.843 on 1 and 71 DF,  p-value: 0.09616
```

Conclusion

Une étude a été faite sur l'épuisement des matériaux (l'acier). Grace à un outil puissant (**R)** et un model a été élu pour nos données.

Remarques

La liste illustrée des résultats de lecture du fichier n'est pas exhaustive.

Epsilon_prime_f (le coefficient de ductilité en fatigue) est représenté par $\mathcal{E}'_f$.

Sigma_prime_f (le coefficient de résistance en fatigue) est représenté par Σ'_f **.**

Sigma_e (contrainte ou charge initiale en traction) est représenté par Σ_e**.**

Sigma_u (contrainte ou charge à la rupture en traction) est représenté par Σ_u **.**

Epsilon_f (facteur de ductilité) est représenté par $\mathcal{E}_f$.

BHN (dureté de brinell).

RA (coefficient de striction après rupture).