

```
*CORRIGE DE L'EXERCICE CHAPITRE 5 - EXERCICE 6
. *Dalila Chenaf-Nicet  Université de Bordeaux.
.
. * Tout d'abord s'assurer d'avoir ouvert le fichier de données stata C5EX6 avant d'ouvrir
le do.fileC5EX6 qui est fichier programme.
.
. * Une fois le fichier de données ouvert ainsi que le do-file appuyer sur Run (Execute en
haut à droite de la barre de menu du dofile)
. * pour démarrer
.
.
. * Toutefois il est possible en sélectionnant les parties du programme de l'exécuter pas à
pas afin de voir apparaître pas à pas les
. * différents résultats.
```

```
. *
.                                     CHAPITRE 5 EXERCICE 6
.                                     * Prévvision de part de marché
.
.
. * Dans un premier temps il s'agit de faire la régression de pm (part de marché) par
rapport à dn (l'univers de vente).
. * Mais les données sont en log dans le modèle. Il donc faut créer les variables en log
.
. gen logpm=log(pm)
.
. gen logdn=log(dn)
```

```
. * Puis faire la régression avec les nouvelles variables
. regress logpm logdn
```

Source	SS	df	MS	Number of obs	=	41
Model	2.27947929	1	2.27947929	F(1, 39)	=	46.67
Residual	1.90470363	39	.048838555	Prob > F	=	0.0000
				R-squared	=	0.5448
				Adj R-squared	=	0.5331
Total	4.18418293	40	.104604573	Root MSE	=	.22099

logpm	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
logdn	1.029819	.1507387	6.83	0.000	.7249217 1.334717
_cons	2.911733	.0600075	48.52	0.000	2.790356 3.03311

```
. * Le coefficient a1 = 1.029 représente une élasticité car le modèle est sous la forme
log-log.
```

```
. * Si on observe le t de student de a1 (t=6.83), le coefficient est significativement
différent de 0 car le t est largement supérieur
```

```
. * au t de la table =1.96
```

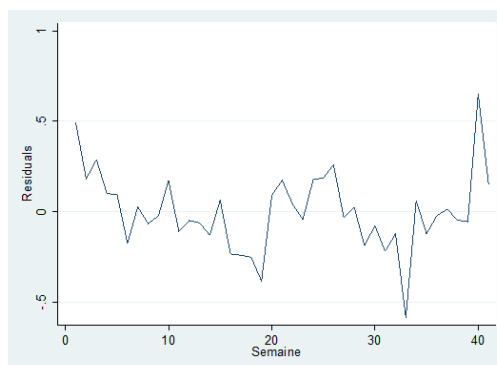
```
. * Il s'agit de récupérer dans un deuxième temps les résidus de la relation
```

```
. predict residu, re
```

```
. * Il faut ensuite faire le graphe des résidus, en précisant que semaine est notre variable
du temps
```

```
. tsset semaine
      time variable:  semaine, 1 to 41
      delta: 1 unit
```

```
. twoway(tslines residu)
```



. * Le graphique laisse supposer une autocorrélation des erreurs. Il faut faire ici le test de Durbin watson.

. * stata indique directement le test avec la commande suivante qui donne la fois les résultats de la régression et la statistique

. regdw logpm logdn

Source	SS	df	MS	Number of obs	=	41
Model	2.27947929	1	2.27947929	F(1, 39)	=	46.67
Residual	1.90470363	39	.048838555	Prob > F	=	0.0000
				R-squared	=	0.5448
				Adj R-squared	=	0.5331
Total	4.18418293	40	.104604573	Root MSE	=	.22099

logpm	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
logdn	1.029819	.1507387	6.83	0.000	.7249217 1.334717
_cons	2.911733	.0600075	48.52	0.000	2.790356 3.03311

Durbin-Watson Statistic = 1.290181

. * Dw = 1.29 < 1.44 (valeur de la table). Il existe une présomption d'autocorrélation des erreurs d'ordre 1.

. * Il s'agit d'introduire une nouvelle variable dans l'analyse qui est IP, l'indice des prix

. gen logip=log(ip)

. regdw logpm logdn logip

Source	SS	df	MS	Number of obs	=	41
Model	3.19515434	2	1.59757717	F(2, 38)	=	61.38
Residual	.989028582	38	.026027068	Prob > F	=	0.0000
				R-squared	=	0.7636
				Adj R-squared	=	0.7512
Total	4.18418293	40	.104604573	Root MSE	=	.16133

logpm	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
logdn	1.285666	.1181933	10.88	0.000	1.046396 1.524936
logip	-4.716588	.795188	-5.93	0.000	-6.326362 -3.106814
_cons	25.22141	3.761531	6.71	0.000	17.60659 32.83623

Durbin-Watson Statistic = 1.828477

. * Concernant les résultats on peut faire les commentaires suivant:

. * - La statistique de Fisher F=61.38 indique que le modèle est globalement significatif.

. * - La statistique de DW = 1.828 ne laisse plus présager d'autocorrélation.

. * Le modèle est donc valide sur le plan statistique.

. * Connaissant pour la semaine 42 les nouvelles valeurs de dn=0.60 et ip=100, il est demandé de calculer la nouvelle valeur de pm.

. * On doit d'abord calculer les données en log

. gen logdn42=log(0.60)

. gen logip42=log(100)

```

.
. * On peut avec les coefficients estimés calculer la nouvelle valeur de logpm42
. gen logpm42 = 1.285*logdn42 - 4.716*logip42 + 25.221

. display logpm42
2.8466063

.
. * Il faut calculer l'exponentielle de cette valeur pour obtenir la grandeur réelle de pm42
. gen pm42=exp(logpm42)

. display pm42
17.229212

.
. * La part de marché prévue est de 17,2%.

. * Il faut calculer l'écart type de prévision. Il faut construire  $(\text{inv}(X'X)) = Z$ 
.
. gen b=1

. mkmat b logdn logip, mat(X)

.
. matrix Z=(inv(X'*X))

. mat list Z

symmetric Z[3,3]
               b          logdn          logip
b      543.63076
logdn  6.3850043   .53673557
logip -114.91581  -1.3178505   24.294862

.
. * Il faut calculer la variance estimée des résidus de la régression avec les deux
variables explicatives
. regress logpm logdn logip

-----+-----
Source |           SS           df           MS       Number of obs   =          41
-----+-----
Model  |    3.19515434           2    1.59757717       F(2, 38)         =          61.38
Residual |    .989028582          38    .026027068       Prob > F          =          0.0000
-----+-----
Total  |    4.18418293          40    .104604573       R-squared         =          0.7636
                                           Adj R-squared     =          0.7512
                                           Root MSE         =          .16133

-----+-----
logpm |           Coef.      Std. Err.      t    P>|t|     [95% Conf. Interval]
-----+-----
logdn |    1.285666      .1181933     10.88   0.000     1.046396     1.524936
logip |   -4.716588      .795188     -5.93   0.000    -6.326362    -3.106814
_cons |   25.22141      3.761531      6.71   0.000    17.60659     32.83623
-----+-----

. predict résidu2,re

.
. egen zz=sum(résidu2^2)

.
. gen sigma2=zz/(_N-2-1)

. display sigma2
.02602707

.
. * IL faut calculer  $X't+h * (\text{inv}(X'X)*Xt+h)$  en définissant au préalable xt+h avec les
données de l'exercice

.
. matrix Xth=(1\ -0.51\ 4.605)

```

```
. matlist Xth
```

	c1
r1	1
r2	-.51
r3	4.605

```
. matrix uu= Xth'*(Z)*Xth
```

```
. matlist uu
```

	c1
c1	.2705919

```
. svmat uu
```

```
. gen sigma2th=sigma2*(1+uu)  
(40 missing values generated)
```

```
. display sigma2th  
.03306978
```

```
.  
. * Les bornes de l'intervalle de prévision sont données par 2.84(+)*1.96*sqrt(0.033)
```

```
. gen bornebasse= 2.84-1.96*(sqrt(sigma2th))  
(40 missing values generated)
```

```
. display bornebasse  
2.483572
```

```
. gen bornehaute= 2.84+1.96*(sqrt(sigma2th))  
(40 missing values generated)
```

```
. display bornehaute  
3.1964278
```

```
.  
end of do-file
```

```
. exit, clear
```