Introduction au Logiciel STATA

Dr. Ir. Epiphane SODJINOU
Agroéconomiste, Biostatisticien

Contenu

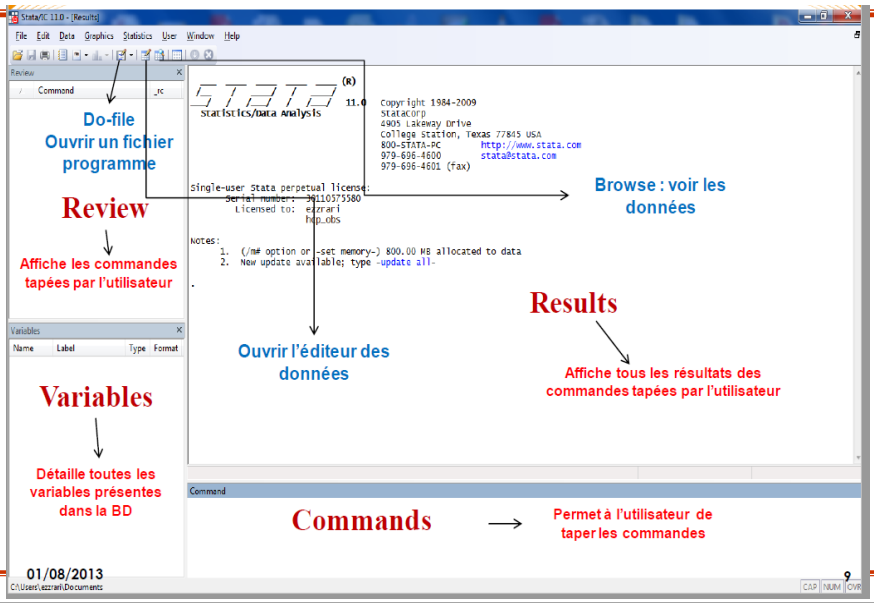
1. Introduction
2. Présentation du logiciel
3. Gestion de bases de données
4. Gestion des variables
5. Statistiques descriptives
6. Graphiques dans Stata
7. Régressions dans stata
8. Autres opérations avec Stata

1. Introduction

- Stata
  - Logiciel complet permettant l'analyse statistique et économétrique
  - Logiciel développé par Stata Corporation.
  - logiciel particulièrement utilisé en épidémiologie et en économie
  - Existe pour tous les systèmes d'exploitation (Windows, Linux, Mc, etc.)
- Contrairement à d'autres logiciels (SAS, R, etc.),
  - Stata a des problèmes pour gérer de très grosses bases de données
- Mode de fonctionnement :
  - Mode commande interactif
  - Mode Menu
  - Mode de programmation (fichiers .do)

2. Présentation du logiciel

2.1. Présentation de l'interface



2. Présentation du logiciel

2.1. Présentation de l'interface

LES DIFFÉRENTES FENÊTRES STATA

- **Fenêtre COMMAND** (*bandeau du bas*) : permet de taper les commandes, qui peuvent être exécutées par la touche « ENTREE »
- **Fenêtre RESULTS** (*plus grande fenêtre*) : décrit les résultats des commandes et indique le cas échéant pourquoi STATA n'a pas pu réaliser la commande
- **Fenêtre REVIEW** (*en haut à gauche*) : liste l'ensemble des commandes réalisées ; en rouge apparaissent celles qui ont échoué.

On peut cliquer sur une des commandes listées dans cette fenêtre pour qu'elle apparaisse à nouveau dans la fenêtre Command

- **Fenêtre VARIABLES** (*en haut à droite*) : donne la liste des variables de la base chargée par STATA, avec leur **label** (ce qu'elles veulent dire)
- On peut cliquer sur une variable listée dans cette fenêtre pour qu'elle apparaisse directement dans la fenêtre Command
- **Fenêtre PROPERTIES** (*en bas à droite*) : indique les propriétés des variables et de la base de données

2. Présentation du logiciel

2.2. Extensions de fichiers Stata

- **Fichier données** :
  - Fichier de données sous format stata avec l'extension **.dta** (les variables sont en colonnes et les individus sont en ligne).
- **Fichier programme** :
  - Fichier de commandes au format ASCII et permet à l'utilisateur de :
    - lancer plusieurs commandes Stata en une seule opération;
    - Garder une trace des commandes exécutées.
  - Extension de ce programme est **.do**
  - On peut appeler un fichier do-file à partir du menu (do-file Editor) ou bien taper **doedit** dans la partie réservée aux commandes.

2. Présentation du logiciel

2.3. Quelques commandes

- **Quelques commandes pour manipuler les fichiers Stata**

cd c:\formation_stata	/*spécifier le répertoire de travail*/
clear all	/*Effacer les fichiers existants et vider la mémoire*/
set memory 800m, permanent	/*permet d'augmenter la mémoire disponible*/
delimit	/*utile pour les commandes très longues et on souhaite revenir à la ligne*/
log using	/*ouvre un fichier résultat*/
cmdlog using	/*ouvre un fichier pour sauvegarder les commandes utilisées*/
log close	/*ferme le fichier résultat*/
cmdlog close	/*ferme le fichier des commandes*/

2. Présentation du logiciel

2.3. Quelques commandes

Fichier résultats :

- Fichier permettant de stocker toutes les commandes exécutées ainsi que les résultats obtenus. Il y a deux types de fichiers :
  - un fichier en format **smcl** ouvrable uniquement sur le logiciel Stata
  - un fichier **log** ou **txt** ouvrable avec n'importe quel éditeur
- Commande utilisée:
  - **log using** "nom_fichier ", **replace append** (smcl)
  - **log using** "nom_fichier.log", **replace append** (fichier texte)
- **log close** (fermer) ; **log off** (suspendre) ; **log on** (reprendre)
- **cmdlog using** : commande permettant de sauvegarder uniquement les commandes exécutées sans résultats

2. Présentation du logiciel

2.4. Problème de mémoire

- Dans certains cas, la **mémoire vive** dégagée par défaut par STATA est **insuffisante**
  - Du rouge apparaît dans votre fenêtre RESULTS
- **Deux solutions**, selon le moment où apparaît le problème **:**
- **Avant d'ouvrir la base** :
  - Commande **set memory** Xm
  - Commande **set maxvar** Y
  - Lorsque la base à charger est grosse
- **Une fois la base ouverte :**
  - Commande **compress**
  - Utile lorsque certains traitements statistiques demandent beaucoup de mémoire (puissance)
  - Inutile si c'est l'ouverture de la base elle-même qui pose problème !

2. Présentation du logiciel

2.5. Opérations dans Stata

Fonctions et expressions

1. Opérateurs arithmétiques		2. Opérateurs de relation		3. Opérateurs logiques	
Addition	+	Supérieur Inférieur	> <	OU (alt gr + 6)	|
Soustraction	-	Supérieur ou égal	>=	ET	&
Multiplication	*	Inférieur ou égal	<=		
Division	/	Egal Egal (s'il y a if)	= ==		
Exposant	^	Différent	~= !=		

2. Présentation du logiciel

2.5. Opérations dans Stata

Fonctions et expressions

4. Fonctions		5. Expressions by, if et in
Racine carrée	sqrt	by : permet de répéter une commande pour chaque valeur (ou modalité) d'une variable donnée. Syntaxe générale pour by est : by variables : commande ...
Exponentielle	exp	
Logarithme	log ln	if : permet de spécifier les conditions dans lesquelles une commande doit être exécutée. Syntaxe générale pour if est : commande .... if condition
Valeur Absolue	abs	
Partie entière	int	in : permet de spécifier les observations auxquelles s'applique une commande. Syntaxe générale pour in est : commande .... in intervalle

3. Gestion de bases de données

3.1. Importation de bases de données dans Stata

Pour un fichier qui est déjà au format Stata, pour l'ouvrir il faut taper :

- **use** "Chemin_du_fichier\ nom_fichier.dta ", **clear** (ouvrir la totalité du fichier)
- OU dans la barre d'outils, cliquer sur l'onglet « ouvrir » et trouver le fichier
- **use** var1 var2 var3 …. **using** "nom_fichier.dta", **clear** (n'ouvrir le fichier qu'avec les variables mentionnées var1 var2 var3…)
- **clear** Permet d'effacer le fichier de données déjà utilisé par Stata

3. Gestion de bases de données

3.1. Importation de bases de données dans Stata

- Pour les fichiers qui ne sont pas au format Stata :
  - Utiliser le ***Stat Transfer*** : c'est un logiciel qui permet de convertir les données d'un autre format (Excel, SAS, R, Limdep, SPSS, etc.) vers le format Stata
- Stata peut lire les données également sous format ASCII. Dans ce cas on utilise souvent les trois commandes suivantes :
  - **infile**
  - **insheet**
  - **import**

3. Gestion de bases de données

3.1. Importation de bases de données dans Stata

- **infile** s'utilise si les données sauvegardées dans un fichier sont séparées par un espace, pour lire les données on utilise :

infile var1 var2 var3 ….. **using** "exercice1.prn" , **clear**

- **insheet** s'utilise si les données sauvegardées dans un fichier sont séparées par des tabulations, pour lire les données on utilise

insheet var1 var2 var3 ….. **using** "exercice1.txt" , **clear** (le fichier ne contient pas les noms des variables)

insheet using "exercice1.txt" , **clear** (fichier contient les noms des variables)

3. Gestion de bases de données

3.1. Importation de bases de données dans Stata

- **import excel** : Permet d'importer un fichier Excel (.xls ou .xlsx) ou CSV (.csv)
- **Import excel** « Chemin_du_fichier\Nom_du_fichier.xls », **clear sheet**(« nom de la feuille contenant les données à importer") **firstrow**
- **firstrow** : Permet d'indiquer que la première ligne de la feuille contient le nom des variables
- **sheet** : Permet de préciser la feuille contenant les données à importer

Exemple

import excel "C:\Users\SODJINOU\Desktop\A_Base complette Syprobio Benin 2014.xlsx", **clear sheet**("Feuil1") **firstrow**;

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables

- L'objet est de fusionner deux bases de données contenant des individus en commun et des variables différentes.
- Supposons qu'on dispose de deux bases de données (carte_démogaraphique) et (carte_emploi) de six individus et qu'on veut fusionner ces deux bases.

  1- il faut s'assurer que les individus ont un identifiant unique dans les deux bases

  2- Trier les deux bases selon cet identifiant

  3- utiliser la commande merge dans stata pour la fusion

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables (one to one)

cartedemog.dta

Idmen	idind	sexe	age
01	0101	1	44
01	0102	2	38
01	0103	1	15
02	0201	2	36
02	0202	2	5
02	0203	1	8

emploi.dta

idmen	idind	sitac
01	0101	AO
01	0102	FF
01	0103	EE
02	0201	AO
02	0203	EE

```
cd c:\formation_stata
use "emploi.dta", clear
sort idind
save "emploi.dta", replace
use "cartedemog.dta", clear
sort idind
merge 1:1 idind using "emploi.dta"
```

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables (one to one)

```
Result                      # of obs.
-----------------------------------------
not matched                         1
from master                         1    (_merge==1)
from using                          0    (_merge==2)

matched                             5    (_merge==3)
-----------------------------------------
```

idmen	idind	sexe	age	sitac	_merge
01	0101	1	44	AO	matched (3)
01	0102	2	38	FF	matched (3)
01	0103	1	15	EE	matched (3)
02	0201	2	36	AO	matched (3)
02	**0202**	**2**	**5**		**master only (1)**
02	0203	1	8	EE	matched (3)

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables (many to one) or (one to many)

cartedemog.dta

idmen	idind	sexe	age
01	0101	1	44
01	0102	2	38
01	0103	1	15
02	0201	2	36
02	0202	2	5
02	0203	1	8

logement.dta

idmen	typehab
01	Apprt
02	MM

```
cd c:\formation_stata
use "logement.dta", clear
sort idmen
save "logement.dta", replace
use "cartedemog.dta", clear
sort idmen
merge m:1 idmen using "legement.dta"
```

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables (many to one) or (one to many)

Result	# of obs.	
not matched	0	
matched	6	(_merge==3)

idmen	idind	sexe	age	typehab	_merge
01	0101	1	44	Apprt	matched (3)
01	0102	2	38	Apprt	matched (3)
01	0103	1	15	Apprt	matched (3)
02	0201	2	36	MM	matched (3)
02	0202	2	5	MM	matched (3)
02	0103	1	8	MM	matched (3)

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des variables

- la commande **mmerge** est une extension de la commande merge et permet de faire la fusion des bases de données sans passer par le tri.
- Exemple :
  - **cd** c:\formation_stata
  - **use** “cartedemog.dta”, clear
  - **mmerge** idind **using** “emploi.dta” `/*ajouter toutes les varaibles du fichier emploi*/`
- **mmerge** idind **using** “emploi.dta”, **table ukeep**(var1 var2) `/*n'ajouter que les variables var1 et var2 du fichier emploi*/`

3. Gestion de bases de données

3.2. Fusion des bases des données

Ajout des observations

- Supposons qu'on dispose de deux bases de données, l'une pour le milieu urbain et l'autre pour le milieu rural et on veut les fusionner en une seule base. Il s'agit là d'ajout d'observations et la commande qu'on utilse dans Stata est **append**.

cd c:\formation_stata

use “fichier_urbain”, clear

append using “fichier_rural”

save “fichier_national”, replace

3. Gestion de bases de données

3.2. Fusion des bases des données

Agréger des variables

- Il s'agit de passer d'une base de données désagrégées à une base de données agrégées. En d'autres termes, il s'agit de remplacer la base de données utilisée par une base de statistiques descriptives

- Supposons qu'on dispose des données par ménage sur la pauvreté et les niveaux de vie et nous voulons agréger les indicateurs de pauvreté et des niveaux de vie au niveau régional, la commande qu'on utilise dans Stata est : **collapse**

```
cd c:\formation_stata
use "ennvm07", clear
preserve                                    /*garder le fichier existant*/
collapse pauvreté deptotp (sum) pop=taille (count) men=up, by(c_region)
save "pauvreté_région", replace
restore                                     /*récupérer le fichier*/
```

3. Gestion de bases de données

3.3. Gestion et Manipulation des variables

- **rename** : permet de renommer la variable

  rename anc_var new_var

- **drop** : permet de supprimer une ou plusieurs variables

  drop var1 var2 …. in, if

- **keep** : permet de conserver dans le fichier les variables choisies

  keep var1 var2 var3 … in, if

- **sort** : permet de trier le fichier selon des clés choisis

  sort idt_men n_ordre

3. Gestion de bases de données
3.3. Gestion et Manipulation des variables

- Identification des observations dupliquées
  - **sort** "list des variable"
  - **quietly by** "list des variable": **gen** dup = **cond**(_N==1,0,_n)
  - **tabulate** dup
- Suppression des observations dupliquées en se basant sur une variable spécifiques ou sur un identifiant
  - **drop if** dup>1
- Identification d'observations dupliquées en se basant sur toutes les variables
  - **unab vlist** : _all
  - **sort** `vlist'
  - **quietly by** `vlist': **gen** dup = **cond**(_N==1,0,_n)

3. Gestion de bases de données
3.3. Gestion et Manipulation des variables

- **order** : sert à ordonner les variables de la base
  - **order** idt_men n_ordre region province
- **aorder** : sert à ordonner les variables de la base par ordre alphabétique
- **destring** : permet de transformer une variable alphanumérique en variable numérique
  - **destring** region, g(c_region)
- **tostring** : transformer variable numérique en une variable alphanumérique
  - **tostring** c_region, g(region)
- **encode** : transformer variable alphanumérique en une variable numérique dont les modalités sont labelisées avec des chaines de caractère
  - **encode** region, g(c_region)

4. Gestion des variables
4.1. Description des données

- **edit** : voir la base de données et permet de la modifier à la main

  edit ou **edit variables**

- **browse** : voir la base de données et ne permet pas de la modifier à la main

  browse ou **browse variables**

- **describe** : Permet de décrire les données de façon générale (format de la variable, label des modalités de la variable, label de la variable)

  describe : décrit toute la base

  describe variables : ne décrit que les variables indiquées

4. Gestion des variables
4.1. Description des données

- **list** : permet d'afficher la base de données ou un extrait de cette base dans la fenêtre des résultats

  list ou **list variables**

- **insheet using** "c:\formation_stata\exercice1.txt", clear

- **list** in 1/6 , voir la base de données pour uniquement les 6 premières observations

4. Gestion des variables

4.1. Description des données

- **codebook** : permet de créer un dictionnaire des variables indiquant le nom de la variable, son label, son format, l'intervalle de ses valeurs, sa moyenne, son écart type, des quantiles (variable continue), fréquences des modalités et leurs labels (variable discrète) , etc.

  codebook sexe revenu

- **lookfor** : Commande utilisée pour chercher les variables d'une grande base de données à partir des libellés des variables.

  lookfor eau

4. Gestion des variables

4.2. Etiquetage des variables et des modalités

- Pour une meilleure description et une meilleure lecture des fichiers de données on affecte un label à chaque variable et à chaque modalité
- *Label des variables :*
- **label var** var1 "nom de la variable"
- *Label des modalités :*
- **label define** var1 1 "label1" 2 "label2" 3 "label3" …
- **label values** var1 var1
- **label values** var2 var1 (affecter les labels de la variable var1 à la variable var2)

4. Gestion des variables

4.3. Création d'une nouvelle variable

- Les principale commandes de création de variables sont : **generate** et **egen**.
- La commande **egen** est une extension de la commande **generate**, elle est utilisée pour créer des variables avec des fonctions spécifiques.

Exemples :

```
gen var3=var1+var2                    /*addition*/
gen var4=5*var1                       /*multiplication*/
gen var6=var2/var1                    /*division*/
gen logvar=log(var)                   /*logarithme*/
gen region=int(identifiant/10000)     /*partie entière*/
```

4. Gestion des variables

4.3. Création d'une nouvelle variable

- Exemples :
  - **use** "ennvm07 ", **clear**
  - **gen** pauvrete=1 **if** (deptotp<=3834&milieu==1)|(deptotp<=3569&milieu==2)
  - **replace** pauvrete=0 if pauvrete==. (. = missing)
  - ou **gen** pauvrete=(deptotp<=3834&milieu==1) | (deptotp<=3569&milieu==2)
- Création des variables dymmies (dichotomiques)
  - **gen** var1=var2==1 /*(var1 est une variable dichotomique prenant la valeur 1 si var2 est égale à 1, 0 sinon)*/
  - **gen** urbain=milieu==1

Ou bien

- **tabulate** var2, **gen**(var) /* créer des variables dichotomiques pour chaque modalité de la var2*/
- **tabulate** nivscol2, **gen**(niv_scolaire)

4. Gestion des variables

4.3. Création d'une nouvelle variable

- **egen** var7=sum(var1) /*somme de la variable1*/
- **egen** var8=sd(var2) /*écart type de la variable2*/
- **egen** var10=rsum(var1 var2 var3) /*somme des variables 1, 2 et 3*/
- **egen** damm=rsum(alim habit habillement sante transport enseignement …) /*la dépense totale du ménage est la somme des différents groupes de dépenses*/
- création des percentiles :
- **xtile** quintile=deptotp, **nq**(**5**) /*quintile à l'échelle nationale*/
- **xtile** quint_urb=deptotp **if** milieu==1, **nq**(**5**) /*quintile au niveau urbain*/
- **xtile** decile=deptotp, **nq**(**10**) /*décile à l'échelle nationale*/
- **xtile** decile_rur=deptotp **if** milieu==2, nq(**10**) /*décile au niveau rural*/
- **xtile** percentile=deptotp, **nq**(**100**) /*centile à l'échelle nationale*/

4. Gestion des variables

4.4. Transformation d'une variable

- Transformations des variables initiales à d'autres formes de variables

Exemples :

- transformer l'âge en groupe d'âge;
  - **recode** age (0/14=1) (15/59=2) (60/max=3), **g**(groupe_âge) /*créer une autre variable*/

  Ou
  - **gen** groupe_age = age
  - **recode** groupe_age (0/14=1) (15/59=2) (60/max=3)
- transformer le type d'activité à 11 modalités (type d'activité détaillé) à un type d'activité à 3 ou 2 modalités (type d'activité agrégé).
  - **recode** typeact (1=1) (2=2) (3/max=3), **g**(type_act_agr) /*créer une autre variable*/

4. Gestion des variables

4.4. Transformation d'une variable

- Remplacement de données manquantes par une valeur X

  recode variable (mis = X)

- Recodification des données alphanumériques en code numériques **sans** ajout des labels

  egen nouvelle_variable = **group**(**variable**)

- Recodification des données alphanumériques en code numériques **avec** ajout des labels

  encode variable, **gen**(nouvelle_variable)

- Conversion de variables CARACTERE en variable numérique

  gen nouvelle_variable = **real**(variable)

4. Gestion des variables

4.5. Les boucles

- Les boucles sont des programmes qui permettent de faire une seule manipulation des variables au lieu de plusieurs. Il y a deux commandes principales de boucles : **forvalues**, **foreach**

- **forvalues** : on l'utilise si les variables contiennent des chiffres

Exemple : créer plusieurs fichiers (ENNVM07) relatifs aux 16 régions

```
forvalues i=1/16 {
use "ennvm07" if c_region==`i', clear
save "ennvm07_reg`i'", replace
}
```

- **foreach** : on l'utilise pour toute autre variable

```
foreach var in sexe age etatmatr act_occ chomeur inactif lirecrir sans_niv f1 f2 second superir {
rename `var' `var'_cm
}
```

5. Statistiques descriptives

5.1. Cas d'une variable quantitiave

- **summarize** var1 /* N, mean, sd, min, max*/
- **summarize** var1, **detail** /* N, mean, sd, min, max, variance,
- **skewness**, kurtosis, percentiles*/
- **tabstat** var1 /* seulement la moyenne*/
- **tabstat** var1, **stat**(n, mean, median, sd, var , min, max) /*plusieurs statistiques*/
- **tabstat** var1, **stat**(mean, median) **by**(sexe) /*plusieurs statistiques ventilées par une variable catégorielle*/

- Exemple:
  - **cd** c:\formation_stata
  - **use** "ennvm07", clear
  - **sum** deptotp deptotm taille age [fw=coef_ind]
  - **sum** deptotp deptotm taille age [fw=coef_ind], **detail**
  - **tabstat** deptotp [fw=coef_ind]
  - **tabstat** deptotp [fw=coef_ind], **stat**(mean median min max N)
  - **tabstat** deptotp [fw=coef_ind], **stat**(mean median) by(milieu)

5. Statistiques descriptives

5.2. Tableaux de croisement

- **tab** var1 var2 /*les n seulement d'un tableau de croisement*/
- **tab** var1 var2, **row** /*les n+% lignes*/
- **tab** var1 var2, **row col** /*les n+%lignes+%colonnes*/
- **tab** var1 var2, **nofreq row col** /*%lignes+%colonnes*/
- **table** var1 var2 var3 /*les n seulement d'une table à 3 entrées*/

- Exemple:
  - **cd** c:\formation_stata
  - **use** "ennvm07", clear
  - **tab** sexe milieu [fw=coef_ind], col row
  - **table** sexe milieu etatmatr [fw=coef_ind], col row scol format(%12.0g)

5. Statistiques descriptives

5.3. Liaison variables qualitatives et variables quantitatives

- **tab** var1 var2, **sum**(var3) nofreq /*moyenne de v_quant vs 2 var qual*/
- **table** var1 var2 var3, c (**mean** var4 **median** var4) row col scol
  /*les stat des d'une variable quantitative en fonction de 3 var qualitatives*/

- Exemple:
- **cd** c:\formation_stata
- **use** "ennvm07", clear
- tab sexe milieu [fw=coef_ind], **sum**(deptotp) nofreq means
- **table** milieu sexe etatmatr [fw=coef_ind], **contents**(**mean** p0_mon **median** p0_mon) row col scol

5. Statistiques descriptives

5.3. Tests usuels

- Test d'indépendance entre deux varaibles qualitatives
  - table var1 var2, **chi2** /*relation d'indépendance entre deux variables*/
- Test de corrélation de Pearson
  - **corr** var1 var2 var3 /*coefficient de corrélation entre les variables*/
  - **pwcorr** var1 var2 var3 …, sig /*coef. corrél. entre les variables + degré de sig*/
- Test de différences de moyennes
  - **ttest** var1=var2 /*comparaison de la moyenne de 2 échantillons*/
  - **ttest** var1=valeur /*comparaison de la moyenne d'une variable*/
  - **ttest** var1, by(var2) /*comparaison de la moyenne de deux groupes*/

5. Statistiques descriptives

5.4. Exporter les tableaux statistiques

- **logout** permet d'exporter un tableau de résultats au format excel, word ou text.
  - **logout**, **save**(table1) **excel replace** : table var1 var2 … /*format Excel*/
  - **logout**, **save**(table1) **word replace** : table var1 var2 … /*format Word*/
  - **logout**, **save**(table1) **tex replace** : table var1 var2 … /*format texte*/
  - **logout**, **save**(table1) **excel word tex replace** : table var1 var2 … /*tous les formats*/

Exemple

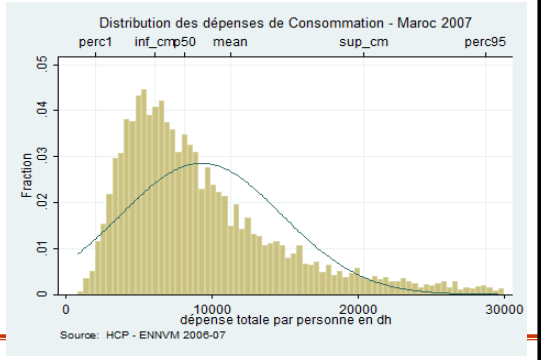
- **cd** c:\formation_stata
- **use** "ennvm07", clear
- **logout**, **save**(table1) **excel word replace** : tabstat deptotp alimsstabp habillement_p [fw=coef_ind], by(milieu)

6. Graphiques dans Stata

6.1. Histogramme

- **hist** var1, width(20) start(50) fraction=freq
- **hist** var1, width(20) start(50) fraction=freq **normal** /*Ajouter la courbe de distribution normale*/
- Exemple
  - cd c:\formation_stata
  - **use** "ennvm07", clear
  - **hist** deptotp [fw=coef_ind], width(2000) start(800) **fraction normal**

histogram deptotp [fw=coef_ind] **if** deptotp<=30000, **fraction normal** **xaxis**(1 2) **ylabel**(0(0.01)0.05, **grid**) **xlabel**(8095 "p50" 2034 "perc1" 6071 "inf_cm" 11233 "mean" 20388 "sup_cm" 28688 "perc95" , **axis**(2) grid gmax) **xtitle**("", axis(2)) subtitle ("Distribution des dépenses de Consommation - Maroc 2007") **note**("Source: HCP - ENNVM 2006-07")



6. Graphiques dans Stata

6.2. Diagramme en barre

- **graph bar** : Commande généralement utilisée pour faire des graphiques en barre
- **graph bar** var1 , over(var2) /*donne graphe de moyenne de la var1 en fonction de var2 (qualitative)*/ (vertical)
- **graph bar** (median) var1, over(var2) /*donne graphe de la médiane de la var1 en fonction de var2 (qualitative)*/ (vertical)
- **graph hbar** var1 , over(var2) /*donne graphe de moyenne de la var1 en fonction de var2 (qualitative)*/ (horizontal)
- Exemple

```
cd c:\formation_stata
use "ennvm07", clear
graph bar deptotp [fw=coef_ind], over(nivscol2) ytitle(deptotp)
title("les dépenses de consommation selon le niveau scolaire du CM",
size(median)) note("Source: HCP – ENNVM 2006-07")
```

6. Graphiques dans Stata

6.3. Diagramme en secteur

- **graph pie** : Commande utilisée pour faire des Diagrammes en secteurs
- **graph pie** var1 , over(var2) /*donne graph en secteur de la var1 selon la var2*/
- **graph pie** var1, over(var2) by(var3 total /*donne graph ventilé par la var3 de la var1 en fonction de var2 (qualitative)*/
- Exemple

```
cd c:\formation_stata
use "ennvm07", clear
graph pie deptotp [fw=coef_ind], over(nivscol2) by(, title(Dépenses de
consommation selon le milieur de résidence)) by(, note(Source: HCP –
ENNVM 2006-07)) by(milieu, total)
```

6. Graphiques dans Stata

6.4. Nuage de points

- **twoway scatter** : Commande pour le nuage de points.

Nuage de points est utilisé lorsqu'on veut voir la liaison entre deux variables quantitatives

- **twoway** (**scatter** var1 var2) /*donne graph de la var1 en fonct de la var var2*/
- **graph** (**scatter** var1 var2) **||** (**lfit** var1 var2) /*donne en plus du graph de la var1 en fonction de la var2, la droite de régression */

Exemple

```
cd c:\formation_stata
use "ennvm07", clear
twoway (scatter coeff_budg_alim_pc deptotp) || (lfit
coeff_budg_alim_pc deptotp), title("la relation entre le niveau de vie")
subtitle("et le coefficient budgétaire de l'alimentaire") xlabel(50000 "5"
100000 "10" 150000 "15" 200000 "20")
```

7. Régressions dans stata

7.1. Régression linéaire (par MCO)

- Commande pour faire des régressions en MCO : **regress** (**reg**) suivie de la variable dépendante, des variables indépendantes et le cas échéant aux options.
- Syntaxe générale :
  - **reg** var_dep var_explicatives (**if**, **in**), options

Pour le cas de l'existence des variables qualitatives parmi les variables explicatives, il faut :

- soit créer des variables ***dummy*** à partir de cette variable et introduire l'ensemble des variables créées sauf une (référence)
- soit utiliser la commande ci-dessous :
  - **xi** : **reg** dep_var var1 var2 **i.**var3 **i.**var4 /*var3 et var4 étant des variables qualitatives*/
  - **reg** dep_var var1 var2 i.var3 i.var4

7. Régressions dans stata

7.1. Régression linéaire (par MCO)

- **predict** : permet d'obtenir la valeur prédite (estimée) de la variable dépendante ainsi que les résidus de la régression
- **predict** yhat, **xb**
- **predict** residu, **re**

7. Régressions dans stata

7.2. Economie de la production

Efficacité technique par la frontier stochastique

- **frontier** depvar [indepvars]
  - depvar = La variable dépendante
  - indepvars Liste des variables indépendantes
- Note
  - Exécuter ainsi, le modèle utilize la distribution semi-normale (**hnormal**) par défaut. Pour modifier cette distribution, on utilize l'option **distribution**(distname), comme suit:
  - **frontier** depvar [indepvars], **distribution**(distname)
  - Les distribution disponible sont : la distribution exponentielle (**exponential**): la distribution normale tronquée (**tnormal**) et la distribution semi-normal (**hnormal**)
- L'extraction des efficacités techniques (TechEff) se fait avec la command
  - **predict** TechEff, **te**

7. Régressions dans stata

7.2. Economie de la production

Efficacité économique

- **frontier** depvar [indepvars], cost
- L'extraction des efficacités économiques (EffEco) se fait avec la command
  - **predict** EffEco, **te**

7. Régressions dans stata

7.2. Economie de la production

Déterminant de l'efficacité technique

- Méthode à deux étapes
  - **frontier** depvar [indepvars]
  - **predict** TechEff, **te**
  - **tobit** TechEff [variables explicatives de l'efficacité], **ul**(1) **ll**(0)
- On peut aussi utilizer la regression linéaire en lieu et place du tobit mais en faisant d'abord une transformation de la variable TechEff.
  - **frontier** depvar [indepvars]
  - **predict** TechEff, **te**
  - **gen** NewTechEff = 1/TechEff
  - **reg** TechEff [liste des variables explicatives de l'efficacité]

7. Régressions dans stata

7.2. Economie de la production

Déterminant de l'efficacité technique

- Méthode à une seule étape
  - **frontier** depvar [indepvars], **uhet**(variables explicatives de l'éfficacité)
- L'extraction des efficacités techniques (TechEff) se fait avec la command
  - **predict** TechEff, **te**

7. Régressions dans stata

7.2. Données de Panel

- **Xtreg** : Commande utilisée pour faire des régressions en panel
- *Attention* : il faut déclarer que vous disposez des données en panel en mentionnant la variable individus et la variable temps.
- **xtset** id tps
- **xtreg** var_dep var_explicatives, **fe** /*modèles à effets fixes*/
- **xtreg** var_dep var_explicatives, **re** /*modèles à effets aléatoires*/

Le test d'Hausman permet de choisir entre les deux modèles.

7. Régressions dans stata

7.3. Econométrie des variables qualitatives

- Le modèle à utiliser dépend de la nature des variables dépendante.
- Quelques cas possibles :
  - Variables dichotomiques ou variables à deux modalités : Exemple: genre, adoption ou non d'une innovation
  - Variables polytomiques ou variables discrètes à plus de 2 modalités:
    - Variables ordonnées (classes des dépenses, classes des revenus, degré de satisfaction, etc.)
    - Variables non ordonnées (catégories socioprofessionnelles, le lieu de consultation médicale, le personnel consulté, etc.)
    - variables séquentielles (le niveau de diplôme, etc.)
  - Variables censurées ou ayant des limites

7. Régressions dans stata

7.3. Econométrie des variables qualitatives

Cas d'une seule Variables dichotomiques ou variables à deux modalités

- Deux commandes sont disponibles:
  - **logit** ou **probit**
- **logit** var_dep var_explicatives
- **logit** var_dep var_explicatives, **or** /*pour avoir les odd ratio*/
- **logistic** var_dep var_explicatives /*pour avoir les odd ratio*/

- Les commandes de post-estimation sont les suivantes :
  - **predict pscore**, xb /*les valeurs prédites*/
  - **compute mfx**, **dydx** /*les effets marginaux*/
  - **compute mfx**, **eyex** /*les élasticités*/
  - **lstat** /*le seuil pris par défaut est 0.5*/
  - **lstat**, **cutoff(pr.)** /*possibilité de choisir un autre seuil*/

7. Régressions dans stata

7.3. Econométrie des variables qualitatives

Cas où la variable expliquée est polytomique ordonnée

- On utilise soit le modèle **logit** ordonné ou le modèle **probit** ordonné
  - **oprobit** var_dep var_explicatives, **robust**
  - **ologit** var_dep var_explicatives

- Les commandes de post-estimation sont les suivantes :
  - **predict pscore**, **xb** /*les valeurs valeurs prédites de l'estimation*/
  - **predict** mod1 mod2 … /*les probabilités prédites de chaque modalité*/
  - **mfx**, **predict** (**p outcome(0)**) /*les effets marginaux de chaque modalité*/

7. Régressions dans stata

7.3. Econométrie des variables qualitatives

Cas où la variable expliquée est polytomique non ordonnée

- On utilise soit le modèle logit conditionnel ou le modèle logit multinomial dit indépendant, ce dernier est le plus souvent utilisé
  - **clogit** var_dep var_explicatives /*pour le logit conditionnel*/
  - **mlogit** var_dep var_explicatives /*pour le logit multinomianl*/
  - **mlogit** var_dep var_explicatives, **base(i)** /*permet de choisir la modalité de référence*/
- Les commandes de post-estimation sont les suivantes :
  - **mfx**, **predict** (**p outcome(1)**) /*les effets marginaux de chaque modalité*/

7. Régressions dans stata

7.3. Econométrie des variables qualitatives

Cas où la variable expliquée censurée

- **tobit** var_dep var_explicatives, **ul**(.) **ll**(.)

Cas des méthodes d'estimation en deux étapes (heckman ou hechprob)

- **heckman** var_dep var_explicatives, **select**(variables_explicatives de l'équation de sélection) /*si la variable censurée est quantitative*/
- **heckprob** var_dep var_explicatives, **select**(variables_explicatives de l'équation de sélection) /*si la variable censurée est dichotomique*/

8. Autres opérations avec Stata

8.1. Matrices dans Stata

Création d'une matrice

- Pour créer une matrice dans Stata, on utilise la commande **matrix input**
  - **matrix input** A = (a,b,c,d\e,f,g,h\.....)
  - **matrix input** X = (1,2\3,4)

- Pour créer une matrice dans Stata à partir d'autres matrices, on utilise la commande **matrix define**
  - **matrix define** X = A + B + C

  Ou
  - **matrix** X = A + B + C

8. Autres opérations avec Stata

8.1. Matrices dans Stata

Transformation d'une matrice

- Pour transférer une base de données en une matrice, on utilise la commande **mkmat**
  - **mkmat** A B C D E, matrix(X) /* créer une matrice X contenant les quatre variables de la base de données A, B, C et D*/
  - **mkmat** revenu /* créer une matrice ligne revenu contenant la variable revenu */
- Pour transférer une matrice en une base de données on utilise la commande **svmat**
  - **svmat** X /* transférer la matrice X en une base de données stata avec les lignes comme observations et les colonnes comme des variables*/

8. Autres opérations avec Stata

8.1. Matrices dans Stata

Calcul matriciel et d'autres utilisations

- Le calcul matriciel dans stata se fait par des opérateurs arithmétiques tels +, - ou *, etc et par les fonctions matricielles de type inverse ou transposé.
- Exemple : dégager le vecteur de régression d'une variable Y en fonction d'un vecteur X:
  - **matrix** B = **inv(X'X)X'Y** /***inv** signifie l'inverse d'une matrice et ' est le symbole de la transposée d'une matrice*/
- Autres utilsations
  - **matrix dir** /*voir les différentes matrices utilisées dans le fichier de travail*/
  - **matrix list** /*lister les matrices*/
  - **matrix rename** /*renommer une matrice*/
  - **matrix drop** /*supprimer une matrice*/

8. Autres opérations avec Stata

8.2. Cartographie des indicateurs dans Stata

- Pour faire des cartes dans stata, il faut d'abord aller au site http://www.diva-gis.org/gdata et télécharger les données SIG de votre pays ou atres pays
- Convertir les données sharpe en données stata par la commande **shp2dta**
- **shp2dta** using XXX_adm1, database(region) coordinates(map) genid(id) gencentroids(center)
- Pour faire la cartographie on utilise la commande :
- **spmap** tx_pauvreté **using** map.dta, id(id) label(data("region.dta") label(VARNAME_1) xcoord(x_center) ycoord(y_center)) fcolor(Reds) title("Taux de pauvreté par région")

8. Autres opérations avec Stata

8.3. Ajout de nouveaux modules Stata

- Pour installer des application récemment développées on utilse la commande ssc install.
  - **ssc install logout** /*permet d'exporter des tableaux au format excel, word ou texte*/
  - **ssc install mmerge** /*permet de fusionner des bases sans passer par le tri*/
  - **ssc install sumdist** /*permet d'avoir les statistiques détaillées de la distribution d'une variable quantitaitive selon les percentiles (distribution des dépenses)*/
  - **ssc install spmap**
  - **ssc install shp2dta**
  - **ssc install mif2dta**

  /*consiste à installer les commandes de la cartographie*/

8. Autres opérations avec Stata

8.4. Le DO-FILE

- Pour travailler de manière efficace sur STATA, il faut utiliser un fichier .do (appelé **do-file**)
  - Permet de conserver en mémoire les commandes faites et de pouvoir retrouver les mêmes résultats à chaque fois
- **Pour ouvrir un do-file** :
  - Icone « **do-file editor** » dans la barre d'outils
  - Ouvre une nouvelle fenêtre, **l'éditeur de do-files**, dans lequel vous pouvez ouvrir vos do-files sauvegardés ou en composer un nouveau
- **Pour utiliser un do-file** :
  - On sélectionne les instructions qu'on veut réaliser et on clique sur l'icone « Execute selection (do) » (*flèche à côté d'une feuille*) dans l'éditeur de do-files
  - Permet d'avoir sur un fichier l'ensemble des commandes qu'on a réalisées, ou qu'on veut réaliser → Constitue le « **programme** »
  - Permet de sauvegarder et de réutiliser plus tard son programme

8. Autres opérations avec Stata

8.6. Recherche de l'aide

- Vous pouvez (allez) oublier certaines commandes. **L'essentiel avec STATA c'est de savoir comment trouver l'information.** Plusieurs possibilités :
- Chercher dans les **manuels**
- Chercher sur **Internet**
- Dans STATA :
  - Le logiciel intègre une version abrégée du *User Manual*
  - Pour y accéder il suffit de taper la commande **help**, suivi du nom de la commande sur laquelle on veut avoir des informations
    - Ex : **help regress**
  - Vous pouvez aussi lancer une recherche sur Internet à partir de STATA, avec la commande **findit** :
    - Ex : **findit regress**