Projet Obe Maghreb

Ecole thématique gestion et analyse de données
20 au 29 avril 2010

Gestion et analyse de données d'enquêtes épidémiologiques

Gestion de données
Mise en œuvre avec EpiData Analysis

Pierre Traissac
UMR 204 « Prévention des malnutritions et pathologies associées »
IRD, Montpellier, France

Institut de recherche pour le développement

Gestion de données

1- Accès aux données

2- Gestion des données

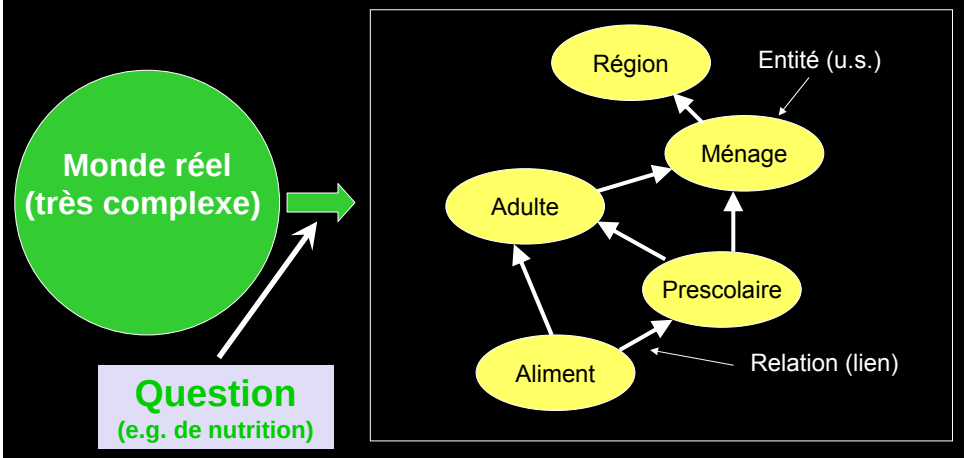
3- Analyse des données

4- Présentation des données

- Conservation
  - sécurité
  - sauvegardes
- Préparation avant analyse
  - sélections (individus, variables)
  - mise en relation, fusion de fichiers
  - calcul de nouvelles variables
  - recodages
  - documentation (versions fichiers, dictionnaires)
- Outil logiciel
  - SGBD (e.g. MS- Access, Oracle, …)
  - gestion de données dans logiciels statistiques (SAS, Stata, SPSS) ou généralistes (EpiData)
- Data manager ⇔ spécialiste discipline

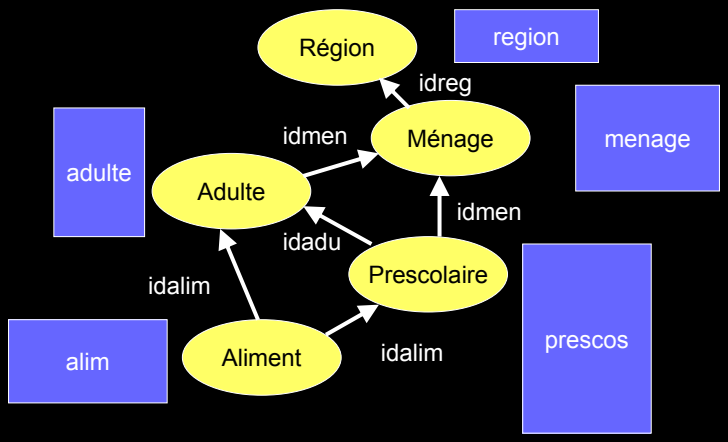
Modèle de données

- Représentation schématique monde réel



Modèle entités-relations

- Base de donnée relationnelle

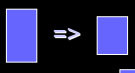
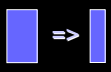
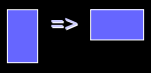


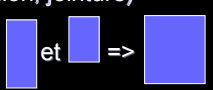
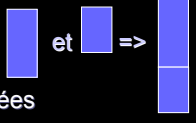
Gestion de données

- **Langage de gestion de données**
  - mise en forme des données avant analyse
  - création de nouvelles tables
    (opérateurs de sélection de lignes, colonnes, fusion de tables)
  - création de nouvelles variables
    (recodage, calcul d'indices, de scores, …)
  - documentation des données
    (labels et notes pour tables et variables)

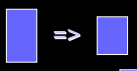
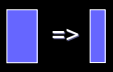
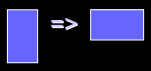
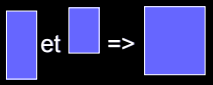
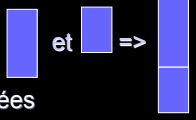
- **Concepts vs. mise en œuvre pratique**
  - concepts, opérateurs : invariants
  - mise en œuvre informatique: dépend du logiciel
    Oracle, Ms-Access, SAS, SPSS, Stata, EpiData Analysis, …
  - SQL (Standard Query Language) +/- standard (pas dans Epidata Analysis)
  - Menus vs. langage de commande / programmes
    (documentation, traçabilité)

Gestion de données

- **Opérateurs sur une seule table**
  - sélection (sélection de lignes) 
  - projection (sélection de colonnes/variables) 
  - changement d'unité statistique
    (table large devient table longue ou inversement) 

- **Opérateurs sur 2 tables et +**
  - langage relationnel (intersection, réunion, jointure)
  - fusion horizontale (mise en relation) : 
  - fusion verticale : 
  - requêtes imbriquées

Gestion de données

- **Opérateurs sur une seule table**
  - sélection (sélection de lignes)  =>
  - projection (sélection de colonnes/variables)  =>
  - changement d'unité statistique (table large devient table longue ou inversement)  =>
- **Opérateurs sur 2 tables et +**
  - langage relationnel (intersection, réunion, jointure)
  - fusion horizontale (mise en relation) :  et =>
  - fusion verticale :  et =>
  - requêtes imbriquées

Gestion de données

- **Sélection de colonnes**

- But : créer table « anthro » à partir de la table « adultes »
  sélection colonnes : sexe, gross, taille, poids, tt, th

idadu	idreg	sexe	age	csp	nivsco	gross	taille	poids	tt	th

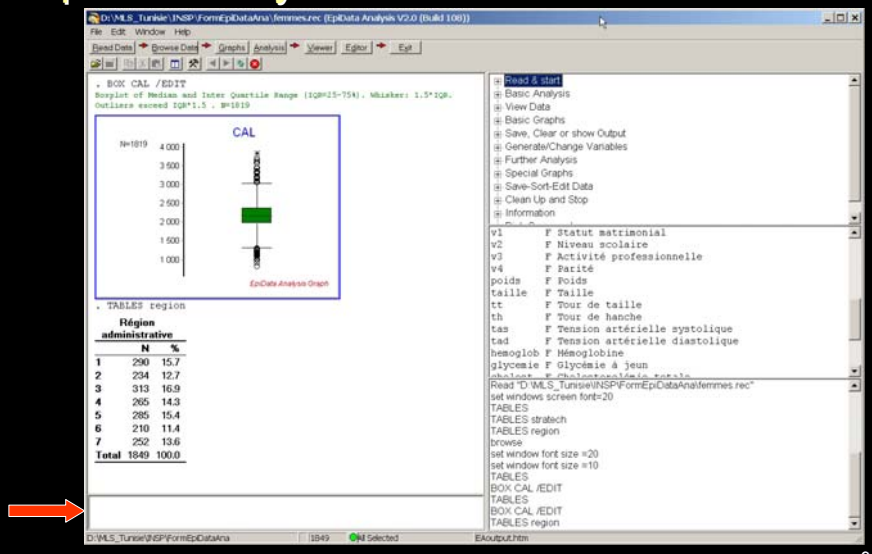
- Epidata Analysis

```
read "adultes.rec"  /close    (possible aussi de lire format .dbf)
keep idadu sexe gross taille poids tt th
ou  drop idreg age csp nivsco
savedata "anthro.rec"  /replace (création table anthro)

read "adultes.rec"  /close
list (ou browse) idadu sexe gross taille poids tt th (liste variables à afficher à l'écran)
                    !!! IDENTIFIANT(S) !!!
```

Gestion de données

- Epidata Analysis



Gestion de données

- Sélection de lignes

- But : créer table « hommes » à partir de la table « adultes »
  sélection des lignes sexe masculin

idadu	idreq	sexe	age	csp	nivsco	gross	taille	poids	tt	th
		1								
		2								
		1								

- Epidata

```
read "adultes.rec"  /close
select if sexe=1 (sélection temporaire)
 count (donne le nombre de lignes sélectionnées)
list (ou browse) (affichage écran de la sélection)

read "adultes.rec"  /close
select if sexe=1
savedata "hommes.rec"  /replace  (création table hommes)
```

Gestion de données

- **Sélection de lignes**

  - sexe=1 condition logique (résultat=0 (faux) ou 1 (vrai))
  si vrai : ligne sélectionnée,
  si faux : ligne non sélectionnée
  - Exemples de conditions
  age<=24 poidsnai<2500 region<>3 obese=«oui»
  doivent respecter les domaines
  - Combinaison de conditions avec and, or, not
  sexe=1 and csp=5 (et logique)
  sexe=1 or csp=5 (ou inclusif)
  malade=1 and not(csp=5)
  sexe=2 and gross=2 and (csp=3 OR nivsco>=2)

  - Annulation sélection temporaire : select (sans rien …) sinon sélections s'aditionnent (équivalent de AND)

Gestion de données

- **Sélection de lignes et colonnes**

  - But : créer table « anthroF » à partir de la table « adultes »
  sélection colonnes : sexe, taille, poids, tt, th
  sélection lignes : femmes non enceintes

  - Epidata Analysis
  read "adultes.rec" /close
  select if sexe=2 and gross=2
  keep idadu sexe taille poids tt th
  savedata " anthroF.rec" / replace (création table anthroF)
  !!! IDENTIFIANT(S) !!!
  ou
  read "adultes.rec" /close
  select if sexe=2 and gross=2
  list (ou browse) idadu sexe taille poids tt th (liste de variables à afficher)

Gestion de données

- **Changement d'unité statistique**

- But : créer une variable nombre d'adultes par ménage (nbadult) ou somme des revenus par ménage (revtotal) à partir de la table « adultes »

u.s. : adulte

idadu	idmen	sexe	revenus	...
100010201	1000102	1	900	
100010202	1000102	2	800	
100010203	1000102	2	0	
100010301	1000103	1	1100	
100010302	1000103	2	600	
100010303	1000103	1	0	
100010304	1000103	1	400	
100010401	1000104	1	800	
100010402	1000104	2	0	

u.s. : ménage

idmen	nbadult	revtotal
1000102	3	1700
1000103	4	2100
1000104	2	800

Gestion de données

- **Changement d'unité statistique**

- Epidata Analysis

read "adultes.rec" /close

aggregate idmen /sum="revenus" /close (possible aussi /save="….")

idmen	N	Nrevenus	SUMrevenus
1000102	3	3	1700
1000103	4	3	2100
1000104	2	2	800

drop nrevenus

rename n to nbadult

rename sumrevenus to revtotal

savedata "men_rev.rec" /replace

Gestion de données

- **Changement d'unité statistique**

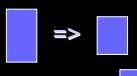
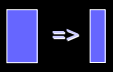
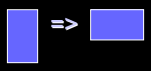
u.s. : adulte

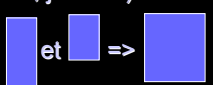
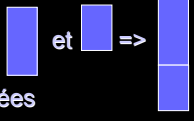
idadu	idmen	sexe	revenus	...
100010201	1000102	1	900	
100010202	1000102	2	800	
100010203	1000102	2	0	
100010301	1000103	1	1100	
100010302	1000103	2	600	
100010303	1000103	1	0	
100010304	1000103	1	400	
100010401	1000104	1	800	
100010402	1000104	2	0	

u.s. : ménage

idmen	nbadult	revtotal
1000102	3	1700
1000103	4	2100
1000104	2	800

Gestion de données

- **Opérateurs sur une seule table**
  - sélection (sélection de lignes) 
  - projection (sélection de colonnes/variables) 
  - changement d'unité statistique (table large devient table longue ou inversement) 

- **Opérateurs sur 2 tables et +**
  - langage relationnel (intersection, réunion, jointure)
  - fusion horizontale (mise en relation) : 
  - fusion verticale : 
  - requêtes imbriquées

Gestion de données

- Fusion horizontale - Relation 1:1

- But

table : adusoec(idadu, idmen, sexe, age, statmat, csp, nivsco) **n x 7**

table : adunutr(idadu, idmen, cal, lip, glu, pro) **n x 6**

création table :

adutot(idadu, sexe, age, statmat, csp, nivsco, cal, lip, glu, pro) **n x 12**

!!! Identifiant commun idadu !!!

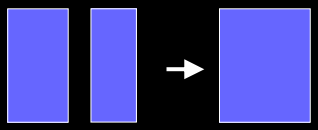
- Epidata Analysis

read "adusoec.rec" /close

merge idadu /file= "adunutr.rec" /table

browse ou list (affichage écran)

savedata "adutot.rec" /replace (création table adutot)



- Si les n sont différents (pb de référence) ?

Gestion de données

- Fusion horizontale (mise en relation)- Relation 1:n

- But

table : adunutr(idadu, idmen, cal, lip, glu, pro) **n x 6**

table : menage(idmen, datenqm, nbpers, revenus) **m x 4** **m<=n**

création table :

adunutrme(idadu, idmen, cal, lip, glu, pro, datenq, nbpers, revenus) **n x 9**

!!! Identifiant commun idmen !!!

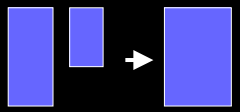
- Epidata Analysis

read "adunutr.rec" /close

merge idmen /file="menage.rec" /table

browse ou list (affichage écran)

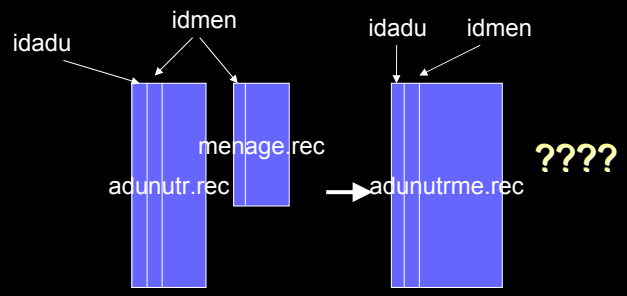
savedata " adunutrme.rec" /replace (création table adunutrme)



- Si pb de référence (idmen présent dans adunutr mais pas dans menage) ?

Gestion de données

- Fusion horizontale (mise en relation)- Relation 1:n



- Duplication (redondance) des données ménage dans table résultante :
  . non unicité de idmen
  . tous adultes d'un même ménage ont même valeur des variables ménage

Gestion de données

- Relation 1: n - Duplication données ménage

- table adunutrme(idadu, idmen, cal, lip, glu, pro, datenqm, nbpers, revenus) n x 9

idadu	idmen	cal	lip	glu	pro	datenqm	nbpers	revenus
100010203	1000102					24/10/2000	5	10000
100010206	1000102					24/10/2000	5	10000
100020702	1000207					31/10/2000	7	8500
100020703	1000207					31/10/2000	7	8500
100020704	1000207					31/10/2000	7	8500
100031706	1000317					04/11/2000	2	12000
100031803	1000318					12/11/2000	3	9000
100031806	1000318					12/11/2000	3	9000
100010203	1000102					12/11/2000	4	5000

Gestion de données

- **Fusion 1:n (e.g u.s. menage : u.s. personne)**

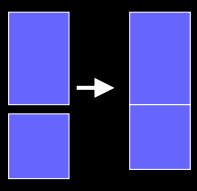
- Nécessaire pour certaines analyses
  (étude conjointe variables ménage et personne)

- Ne pas utiliser pour analyses u.s. ménage
  !!! Redondance données ménages !!!

Gestion de données

- **Fusion verticale**

- But :
  table Tunisie : adutun(idadu, idmen, sexe, age, statmat, csp, nivsco) n_1 x 7
  table Maroc : adumar(idadu, idmen, sexe, age, statmat, csp, nivsco) n_2 x 7
  création table :
  adumag(idadu, idmen, sexe, age, statmat, csp, nivsco) $(n_1 + n_2)$ x 7
  mêmes variables, lignes des 2 tables



- Epidata Analysis
  read "adutun.rec" /close
  append "adumar.rec"
  browse ou list (affichage écran)
  savedata "adumag.rec" /replace (création table adumag)

- ATTENTION : unicité de idadu dans adumag ???? Code pays ????

- Si les variables sont différentes ? se passe +/- bien suivant logiciel.

Gestion de données

- **Requête imbriquée**

  - sélection de lignes d'une table
  sur une condition utilisant une variable d'une autre table

  - adunutr(idadu, idmen, cal, lip, glu, pro) n x 6

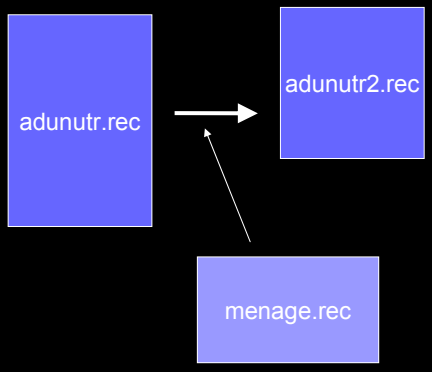
  - sélectionner dans cette table les individus dont le ménage comprend plus de 2 adultes

  - ? variable nbadult dans : menage(idmen, datenqm, nbadult, revtotal)

  - certains langages (e.g. SQL, mais pas Epidata Analysis) : requêtes imbriquées directes sans fusion des tables

Gestion de données

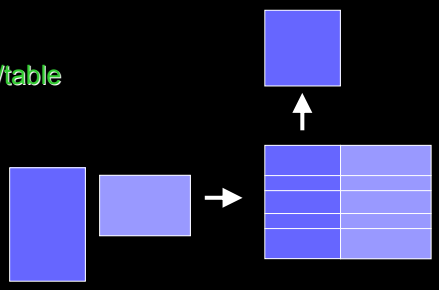
- **Requête imbriquée**



Gestion de données

- (pseudo) requête imbriquée Epidata
  - mise en relation (fusion temporaire) tables adunutr et menage
  - sélection de lignes
  - création table adunutr2 contenant seulement les variables adunutr pour les u.s. sélectionnées
  - Epidata Analysis

```
read "adunutr.rec" /close
merge idmen /file="menage.rec" /table
select if nbadult>=2
keep idadu idmen cal  lip glu pro
save "adunutr2.rec" /replace
```



Gestion de données

- Sélection lignes, colonnes
- Fusion verticale, horizontale
- !!!! Vérifier la table résultante !!!
  - nombre de lignes ?
  - nombre de colonnes ?
  - mise en relation correcte ?

Gestion de données

1- Accès aux données

2- Gestion des données

3- Analyse des données

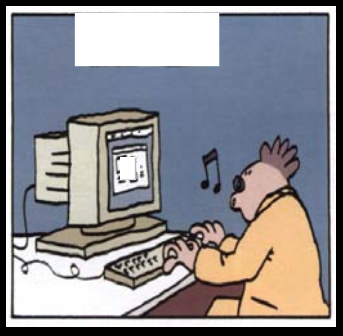
4- Présentation des données

- Conservation
  - sécurité
  - sauvegardes
- Préparation avant analyse
  - sélections (individus, variables)
  - mise en relation, fusion de fichiers
  - calcul de nouvelles variables
  - recodages
  - documentation (versions fichiers, dictionnaires)
- Outil logiciel
  - SGBD (e.g. MS- Access, Oracle, …)
  - gestion de données dans logiciels statistiques (SAS, Stata, SPSS) ou généralistes (EpiData)
- Data manager ⇔ spécialiste discipline

Gestion de données

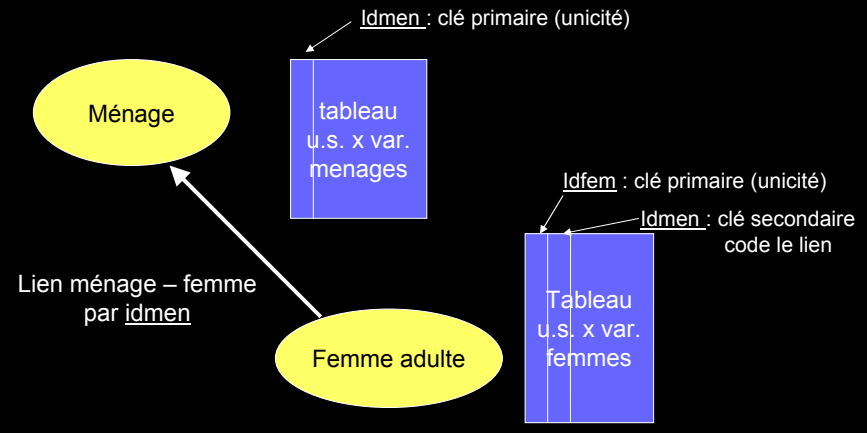
Exercices pratiques
gestion de données (1)

- lecture de fichiers, visualisation
- vérifications (tris, stat. élémentaires)
- sélection de lignes, colonnes
- changement d'unité statistique
- fusion de tables
- sauvegarde des nouvelles tables
- documentation



Gestion de données

- **Modèle de données pour les exercices**



Gestion de données

1- Accès aux données

2- Gestion des données

3- Analyse des données

4- Présentation des données

- **Conservation**
  - sécurité
  - sauvegardes
- **Préparation avant analyse**
  - sélections (individus, variables)
  - mise en relation, fusion de fichiers
  - calcul de nouvelles variables
  - recodages
  - documentation (versions fichiers, dictionnaires)
- **Outil logiciel**
  - SGBD (e.g. MS- Access, Oracle, …)
  - gestion de données dans logiciels statistiques (SAS, Stata, SPSS) ou généralistes (EpiData)
- **Data manager ⇔ spécialiste discipline**

Gestion de données

- **Différents types statistiques de variables**

- Quantitative . continue (taille en cm : 175,6 ; ingéré : 2456 kcal)
  . discrète (nombre de personnes dans ménage : 12)

  Valeurs dans domaine (intervalle)

- Qualitative . ordonnée (état habitat : bon , moyen, mauvais)
  . quelconque (statut matrimonial :
  célibataire, marié, veuf, divorcé, autre)
  . dichotomique (sexe : F/M, fumeur : oui/non)

  Modalités exclusives, exhaustives (catégorie « autre »)

  Modalités dans domaine (liste de modalités)

Gestion de données

- **Différents types informatiques de variables**
  - numériques (entiers, réels)
  - caractère
  (- date : type numérique, nombre de jours depuis date de référence , souvent le 1er janvier 1960)

- **Notion de codage**

Gestion de données

- **Variables numériques**
  - entiers : 0, 1, 15, 345
  - réels : 3.56, 100, -2.5
  - longueur, décimales (#, ###, ####.##)
  - opérations algébriques : +, x, -, /, log, ^2, ….
  - comparaisons : < >, <=, =, <>
  - valeur manquante : .

- Création de variable dans Epidata Analysis (e.g. variable réelle)

```
define caltot  ####.##
caltot=calprot+calglu+callip+calalc
```

ou

```
gen caltot=calprot+calglu+callip+calalc
label caltot « Energie totale (calories) »
```

possible préciser le type (i :entier, f: réel)

```
gen i  var= …..
gen f  var=…..
```


Gestion de données

- **Variables caractère (alphanumériques)**
  - contenu : obèse, oui, non, c2, 145, 4.5, pierre
  - longueur
  - pas d'opérations algébriques
  - comparaison =
  - fonctions spécifiques (sous chaînes, …)
  - valeur manquante : chaîne vide

- Création de variable dans Epidata Analysis

```
define nom _______   (nom, type caractère, longueur 8)
nom=expression
```

ou gen s nom=expression

label nom «Nom de famille de la personne»

- Epidémiologie / statistique : peu utilisées en pratique

Gestion de données

- **Variables date**
  - type numérique, nombre de jours depuis le 1er janvier 1960
  - différents formats
  - opérateurs, comparaisons
    (~ idem variables numériques)
  - valeur manquante : .

- Création de variable dans Epidata
  define datvis <dd/mm/yyyy>
  datvis=dmy(jourvis,moisvis, anvis) (par exemple) datvis=today
  ou
  gen d datvis=dmy(jourvis,moisvis, anvis)

  label datvis «Date de visite»

Gestion de données

- **Notion de codage**
  - variable quantitative : unités
    taille : 1.756 m, 175.6 cm, 1756 mm

  - variable qualitative : codes (beaucoup de choix possibles)
  Satut matrimonial :

Valeurs	---------- Différents codages possibles ----------					
Célibataire	Célibataire	CEL	C	1	5	4
Marié(e)	Marié	MAR	M	2	4	5
Veuf(ve)	Veuf	VEU	V	3	3	6
Divorcé(e)	Divorcé	DIV	D	4	2	7
Autre	Autre	AUT	A	5	1	8

  - !!! codes ≠ valeurs de la variable (cf. analyse) !!!

  - documentation : unités, codes (variables de base ET calculées)

Gestion de données

Codage valeurs manquantes

- u.s. : ménage (définition …)
- variable : « nombre de personnes dans le ménage »
  type statistique : quantitative discrète (1 à 20)
  informatique : variable nbpers numérique

- Valeurs manquantes ??
  statistique /épidémiologie : on n'a pas la donnée pour ce ménage
  informatique : on n'a pas affecté de valeur pour ce ménage (vide)

Choix possibles :
- codage(s) spécifique(s) manquants (e.g. code 9 ou 99 ou 9999)
  => manquant « statistique » différent de « manquant informatique »
  on peut imaginer avoir à la fois des . (i.e. on n'a rien saisi)
  et des 99 (on a saisi code manquant)
- codage par non affectation de valeur (e.g. pas de saisie) si pas la donnée
  => les deux notions se confondent

Gestion de données

Variables transformées (recodages)

- Taille en m (1,756) => Taille en cm (175,6)
  quantitatif => quantitatif

- Catégories d'âge
  18 <= âge < 25 : catégorie 1
  25<=âge < 40 : catégorie 2
  40<= âge : catégorie 3
  quantitatif => qualitatif

- Anémie :
  hb <110 g /L : oui
  hb>=110 g/L : non
  quantitatif => dichotomique

Gestion de données

- Variables transformées (recodages)

- Marié (oui/non) :
  - statut matrimonial = marié : oui
  - statut matrimonial = célibataire,veuf, divorcé, autre : non
  - qualitatif « quelconque » => dichotomique
  - dichotomique : codage en 1/2 ou en 1/0

- Variable qualitative => indicatrices (pour les modèles)
  - une nouvelle variable en 0/1 (=indicatrice) pour chaque modalité de la variable qualitative
  - statut matrimonial : 5 modalités => 5 indicatrices

Gestion de données

- Variables calculées (indices, scores)

A réfléchir en fonction de l'étape « Analyse des données »

- IMC = poids/taille^2 kg/m^2 (quantitatif => quantitatif)

- Equipement du ménage
  - WC (oui/non), électricité (oui/non), eau (oui/non), égout (oui/non)

=> score d'équipement : 0 à 4 (quantitative discrète)

- questionnaire de fréquence activité physique

=> dépense énergétique totale

- questionnaire échelle d'attitudes

=> attitude vis-à-vis de l'obésité

Gestion de données

- Variables transformées (unités)

taille	=>	taille2
taille en m		taille en cm
1.534		153.4

```
gen taille2=taille*100
label taille2 «Taille en cm»
```

agejour	=>	agemois
âge en jours		âge en mois décimaux
567		18.62

```
gen agemois=agejour/30.4375
label agemois «Age en mois»
```

Vérifier les recodages (e.g. describe taille taille2)

Gestion de données

- Variables transformées (recodages)

- qualitatif => qualitatif

Statmat : type numérique valeurs (1,2,3,4,5)
(célibataire, marie, veuf, divorcé, autre)

⇒ statmat3 : type numérique valeurs (1,2,3)
(célibataire, marié, autre)

```
define statmat3 #
recode statmat to statmat3 1=1  2=2  3,4,5=3
label statmat3 «Statut matrimonial en 3 classes»
```

(possible aussi d'attribuer des « labelsvalues » (exercices)

- Vérifier le recodage (tables statmat statmat3)
- Mettre à jour le dictionnaire de variables

Gestion de données

▪ Variables transformées (recodages)

- qualitatif => qualitatif dichotomique

Statmat : type numérique valeurs (1,2,3,4,5)
(célibataire, marié, veuf, divorcé, autre)
⇒ marie : type numérique valeurs (1,0)
(marié, autre)

```
define marie #
recode statmat to marie (2=1) (1,3,4,5=0)
ou
select  statmat<>.  (sinon les manquants sont recodés en 0)
gen i marie=(statmat=2)
select
label marie «Statut matrimonial en 2 classes : marié ou non)»
```

- Vérifier le recodage (e.g. freq statmat marie)
- Mettre à jour le dictionnaire de variables

Gestion de données

▪ Variables transformées (recodages)

- Quantitatif => qualitatif
  age (âge en années) : type numérique valeurs 18 à 79
  => catage3 : type numérique 1: 18 à 24, 2: 25 à 39, 3: 40 et +

```
define catage3 #
recode age to catage3 lo-24.999=1  25-39.999=2  40-hi=3
label catage3 «Age en 3 classes»
```

- Vérifier recodage (min et max de age par catégorie de catage3)

```
means age catage3
```

ou bien

```
sort age
browse age catage3
```

- Mettre à jour le dictionnaire de variables

Gestion de données

Variables transformées (recodages)

- Variable qualitative => indicatrices

Statmat : type numérique valeurs (1,2,3,4,5) (célibataire, marié, veuf, divorcé, autre)

=> 5 indicatrices : type numérique valeurs (1,0)

idadu	statmat	mat1	mat2	mat3	mat4	mat5
100010203	3	0	0	1	0	0
100010206	1	1	0	0	0	0
100020702	1	1	0	0	0	0
100020703	4	0	0	0	1	0
100020704	2	0	1	0	0	0
100031706	5	0	0	0	0	1
100031803	5	0	0	0	0	1

Gestion de données

Variables transformées (recodages)

- Variable qualitative => indicatrices

select statmat<>. (sinon les manquants sont recodés en 0)
gen i mat1=(statmat=1) (si statmat=1 mat1=1 et 0 sinon)
gen i mat2=(statmat=2) (si statmat=2 mat2=1 et 0 sinon)
gen i mat3=(statmat=3) (si statmat=3 mat3=1 et 0 sinon)
gen i mat4=(statmat=4) (si statmat=4 mat4=1 et 0 sinon)
gen i mat5=(statmat=5) (si statmat=5 mat5=1 et 0 sinon)
select

Vérifier les nouvelles variables
freq statmat
freq mat1 mat2 mat3 mat4 mat5
(freq de mat1=1 : fréquence de modalité 1 de statmat, etc..)

- Utile pour codage des variables qualitatives dans les modèles
  e.g. regress imc mat1 mat2 mat3 mat4 (à voir + tard…)

Gestion de données

- **Calcul nouvelles variables (indices, scores)**

- Calcul âge :
  à partir de datenq, datnai : type date

```
gen agean= (datenq – datnai)/365.25
label agean «Age en années»
describe agean
```

- Calcul IMC (indice)
  à partir de poids (poids en kg), taille (taille en m)

```
gen  imc= poids / (taille^2)
label imc «Indice de masse corporelle en kg/m2»
describe imc
```

- Exemple : recodage IMC >=30kg/m^2 (obésité)

```
select imc<>.
generate obese=(imc>=30)
select
label obese «Personne obèse (1:oui, 0:non) »
means imc obese
```

- Vérifier les nouvelles variables - Mettre à jour le dictionnaire de variables

Gestion de données

- **Calcul nouvelles variables (indices, scores)**

- e.g. score de biens possédés par le ménage
  à partir de frigo, malav, cuisi, tv, voitu, para, ordi, intern
  toutes de type #, 1:oui 0:non

```
gen i nbiens=frigo+malav+ cuisi + tv+ voitu+para+ordi+intern
```

=> nbiens : domaine (valeurs) de 0 à 8

Remarque : possible pondérer (valeur monétaire, autre …)

```
gen i val_biens=400*frigo+500*malav+ 250*cuisi +250* tv
                + 5000*voitu+300*para+1000*ordi+250*intern
```

- Examen distribution :

```
freq nbiens /m
bar nbiens
```

Recodage e.g. 3 catégories (voir signification en fonction répartition)

```
define nbiensc3 #
recode nbiens to nbiensc3  0,1,2=1  3,4,5=2  6,7,8=3
freq nbiens nbiensc3
```

- Vérifier les nouvelles variables - Mettre à jour le dictionnaire de variables

Gestion de données

- Variables transformées (recodages)
- Variables calculées (indices, scores)
- Rajouter variables dans nouvelle(s) table(s)

```
read "table_entree.rec" /close
        calculs nouvelle(s) variable(s)
        recodages  …
        labels …
savedata "table_sortie.rec" /replace
```

Mettre à jour le dictionnaire de variables

Gestion de données

1- Accès aux données

2- Gestion des données

3- Analyse des données

4- Présentation des données

- Conservation
  - sécurité
  - sauvegardes
- Préparation avant analyse
  - sélections (individus, variables)
  - mise en relation, fusion de fichiers
  - calcul de nouvelles variables
  - recodages
  - documentation (versions fichiers, dictionnaires)
- Outil logiciel
  - SGBD (e.g. MS- Access, Oracle, …)
  - gestion de données dans logiciels statistiques (SAS, Stata, SPSS) ou généralistes (EpiData)
- Data manager ⇔ spécialiste discipline

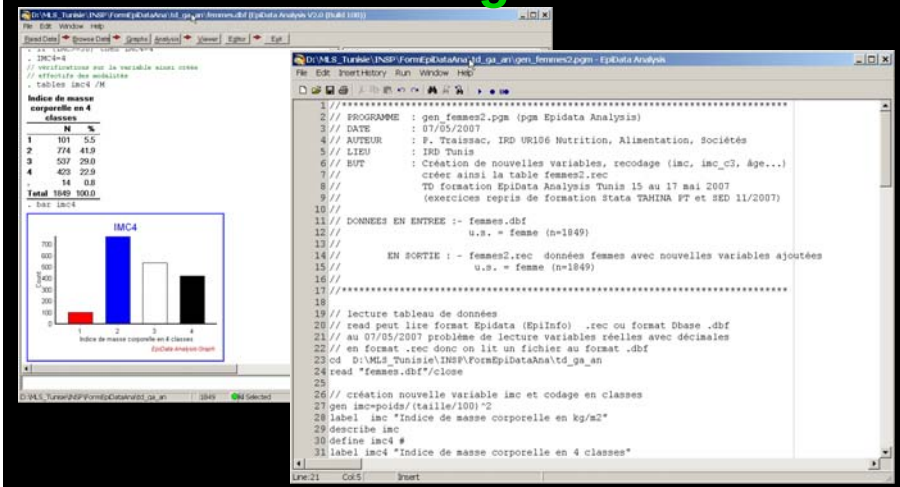
Gestion de données

Qualité ⟷ Traçabilité ⟷ Documentation

- Documenter les opérations sur les fichiers (fusions, création de variables)
- Mise à jour des dictionnaires de variables
- Programmation vs. mode interactif
- Entêtes / commentaires dans les programmes (//)

Traçabilité : gestion et analyse

Programmes : oui !



Traçabilité : gestion et analyse

- **Entête des programmes (e.g. Epidata Analysis)**

```
//*************************************************************************************************
// Nom du programme : gen_femmes2.pgm
// Type de programme : Epidata Analysis
// Auteur : PT, IRD, UR 106 Nutrition, Alimentation, Sociétés
// Date : 08/05/2007
// Lieu : IRD, Montpellier – IRD, Tunis
// But :   Calculs et recodages données femmes en préalable à analyse
//           facteurs de risque de l'obésité
//
// Données en entrée :
//        femmes.rec (u.s. : femme adulte, n=1849)
//
// Données en sortie :  femmes2.rec (n=1763)
//
// Remarques diverses : pgm reprix de exercices Stata SED et PT
//*************************************************************************************************
À partir d'ici les instructions de programmation proprement dites
```


Traçabilité : gestion et analyse

- **Commentaires dans les programmes**

```
read "femmes.rec "   /close
//          création nouvelle variable imc
gen imc=poids/(taille/100)^2
label  imc "Indice de masse corporelle en kg/m2 "
//          imc en 4 classes (bornes OMS : maigreur, surpoids, obésité)
define imc4 #
label imc4 "Indice de masse corporelle en 4 classes"
//           à ce stade elle n'a que des valeurs manquantes
recode imc to imc4 lo-18.4999=1 18.50-24.9999=2 25.0-29.9999=3 30-hi=4
//          on peut assigner des labels aux codes ainsi créés
labelvalue imc4 /1="maigre" /2="normal" /3="surpoids" /4="obèse"
//          codage obesité
define obesite #
recode imc to obesite lo-29.999=0 30-hi=1

//           sauvegarde nouvelles variables dans table femmes2
savedata "femmes2.rec"   /replace
```

Gestion de données

Exercices pratiques
gestion de données (2)

- création de nouvelles variables, indices
- recodages
- sauvegarde des nouvelles variables
- écriture de programmes EpiData Analysis
- mise à jour dictionnaire de variables