

IMPORTATION, CRÉATION, MANIPULATION, EXPORTATION DE DONNÉES STATISTIQUES

Bernard Dupont

Bernard.Dupont@univ-lille1.fr

ahm.bahah@yahoo.fr

Bien que l'analyse et le traitement des données ne soient pas sa vocation première, Maple se révèle un auxiliaire précieux dans ce domaine. Encore faut-il savoir dominer les techniques permettant de stocker intelligemment le contenu statistique (les données) dans le contenant adéquat (tableau, listes, listes de listes, vecteurs, matrices). La première partie est consacrée à la question non triviale de l'importation de données statistiques provenant de fichiers externes. La seconde partie donne des indications sur le création de données dans une feuille de travail Maple. La troisième partie porte sur la manipulation des données dans une feuille de travail Maple. La quatrième montre comment exporter une base de données créée dans un worksheet.

Ce chapitre est essentiellement dû aux efforts constants et enthousiastes de M. Ahmed OULD, étudiant en master d'économétrie appliquée à la faculté de sciences économiques de l'université de Lille 1 qui a effectué son stage au sein du laboratoire EQUIPPE sur le thème des potentialités de Maple dans le domaine de la statistique descriptive et de l'économétrie.

▼ Importation de données statistiques

[Soit deux fichiers de données statistiques : abc.xls et Durbin_Watson_5%_6_100.txt.

[Le fichier abc.xls stocke des données sur une feuille Excel dans un tableau de 1205 lignes et 30 colonnes. En voici un aperçu partiel :

Microsoft Excel - abc

Fichier Edition Affichage Insertion Format Outils Données Fenêtre ? Adobe PDF

A1 trimestre

	A	B	C	D	E	F	G	H	I
1	trimestre	code OP	SOR	ouv/mois	SAPRE	SAPRIS	SAPRIS/mois	SANPRIS	ST
2	T1 2002	BERE	132	44	5135	5015	1672	120	
3	T1 2002	DABA	97	32	4058	3913	1304	145	
4	T1 2002	DADE	85	28	4178	4001	1334	177	
5	T1 2002	FRVA	80	40	2870	2828	1414	42	
6	T1 2002	JEND	112	56	3263	3242	1621	21	
7	T1 2002	LACZ	146	49	6012	5877	1959	135	
8	T1 2002	LIBA	124	41	4348	4247	1416	101	
9	T1 2002	MCHE	175	58	4146	4106	1369	40	
10	T1 2002	PHPO	108	36	3899	3882	1294	17	
11	T1 2002	RAUB	55	28	4314	4272	2136	42	
12	T1 2002	SEDE	116	39	4548	4306	1435	242	
13	T1 2002	SOMA	92	31	4203	3746	1249	457	
14	T1 2002	STBE	48	48	2744	2744	2744	0	
15	T1 2002	STER	89	30	5046	4894	1631	152	
16	T1 2002	SYWI	76	25	3129	3063	1021	66	
17	T1 2002	VEPR	143	48	5079	5074	1691	5	
18	T1 2002	VIME	44	44	1930	1880	1880	50	
19	T1 2002	VINC	96	32	2950	2934	978	16	
20	T1 2002	AULA	30	30	1508	1463	1463	45	
21	T1 2002	AURE	84	42	3563	3557	1779	6	
22	T1 2002	CAFA	115	38	4657	4643	1548	14	
23	T1 2002	CAP	22	22	1314	1216	1216	98	
24	T1 2002	CAPO	27	27	855	824	824	31	
25	T1 2002	CJ	108	36	5301	5168	1723	133	
26	T1 2002	DATA	143	48	5052	4828	1609	224	
27	T1 2002	NADJ	219	73	5716	5705	1902	11	
28	T1 2002	OLIM	134	45	4470	4453	1484	17	

Feuil1 Feuil2 Feuil3

Prêt NUM

Le fichier Durbin_Watson_5%_6_100.txt est un document texte contenant des données sur 1710 lignes et 4 colonnes et qui s'ouvre avec le bloc-notes :

T	K	dL	dU
6.	2.	0.61018	1.40015
7.	2.	0.69955	1.35635
7.	3.	0.46723	1.89636
8.	2.	0.76290	1.33238
8.	3.	0.55907	1.77711
8.	4.	0.36744	2.28664
9.	2.	0.82428	1.31988
9.	3.	0.62910	1.69926
9.	4.	0.45476	2.12816
9.	5.	0.29571	2.58810
10.	2.	0.87913	1.31971
10.	3.	0.69715	1.64134
10.	4.	0.52534	2.01632
10.	5.	0.37602	2.41365
10.	6.	0.24269	2.82165
11.	2.	0.92733	1.32409
11.	3.	0.75798	1.60439
11.	4.	0.59477	1.92802
11.	5.	0.44406	2.28327
11.	6.	0.31549	2.64456
11.	7.	0.20253	3.00447
12.	2.	0.97076	1.33137
12.	3.	0.81221	1.57935
12.	4.	0.65765	1.86397
12.	5.	0.51198	2.17662
12.	6.	0.37956	2.50609
12.	7.	0.26813	2.83196
12.	8.	0.17144	3.14940
13.	2.	1.00973	1.34040
13.	3.	0.86124	1.56212
13.	4.	0.71465	1.81593
13.	5.	0.57446	2.09428
13.	6.	0.44448	2.38967
13.	7.	0.32775	2.69204
13.	8.	0.23049	2.98506
13.	9.	0.14693	3.26577

On veut importer les données dans un document de travail Maple.

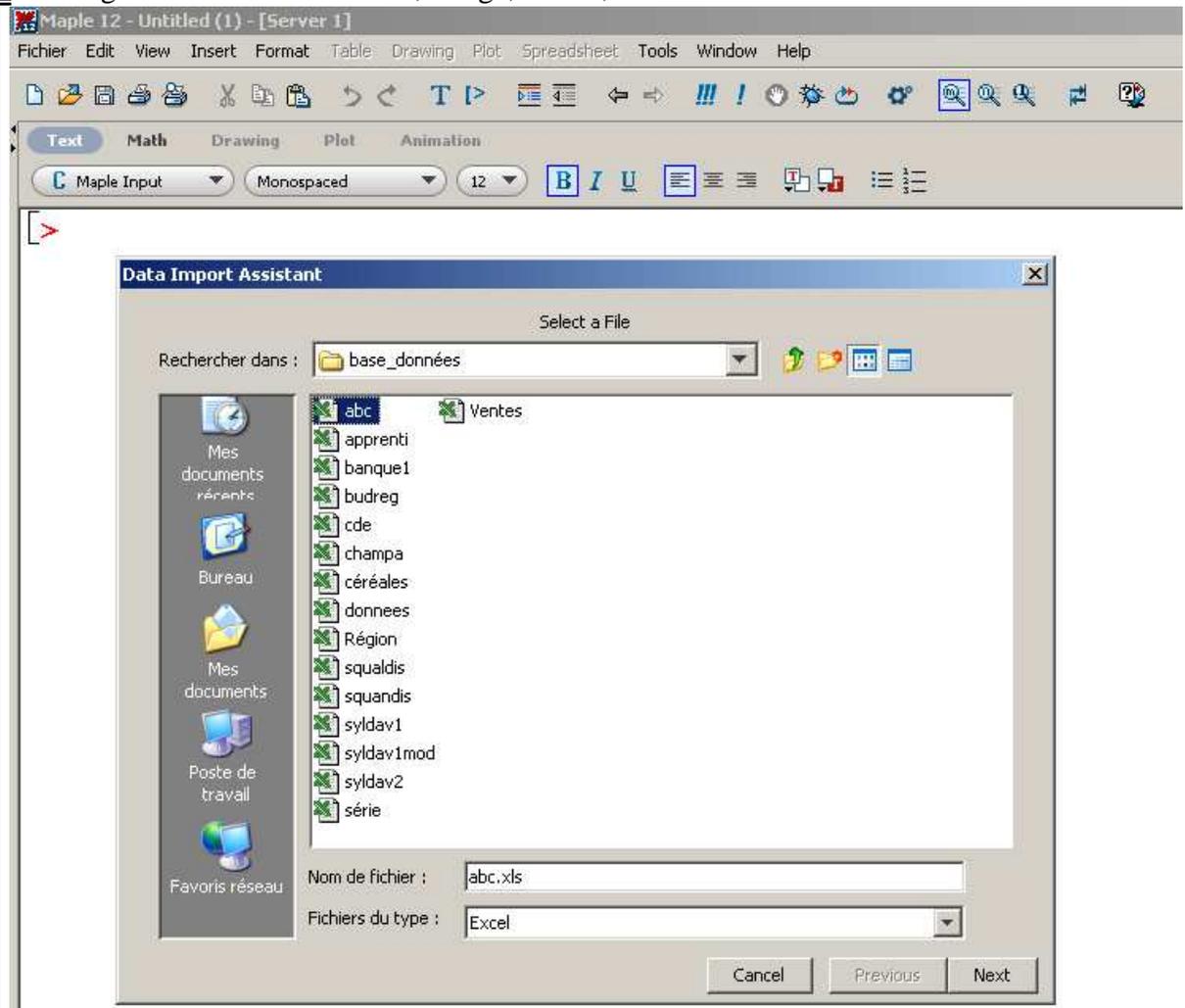
Données en .xls

Le cas le plus fréquent est celui où les données sont au format Excel. Maple a développé deux outils (parfaitement équivalents) pour importer la base dans le worksheet.

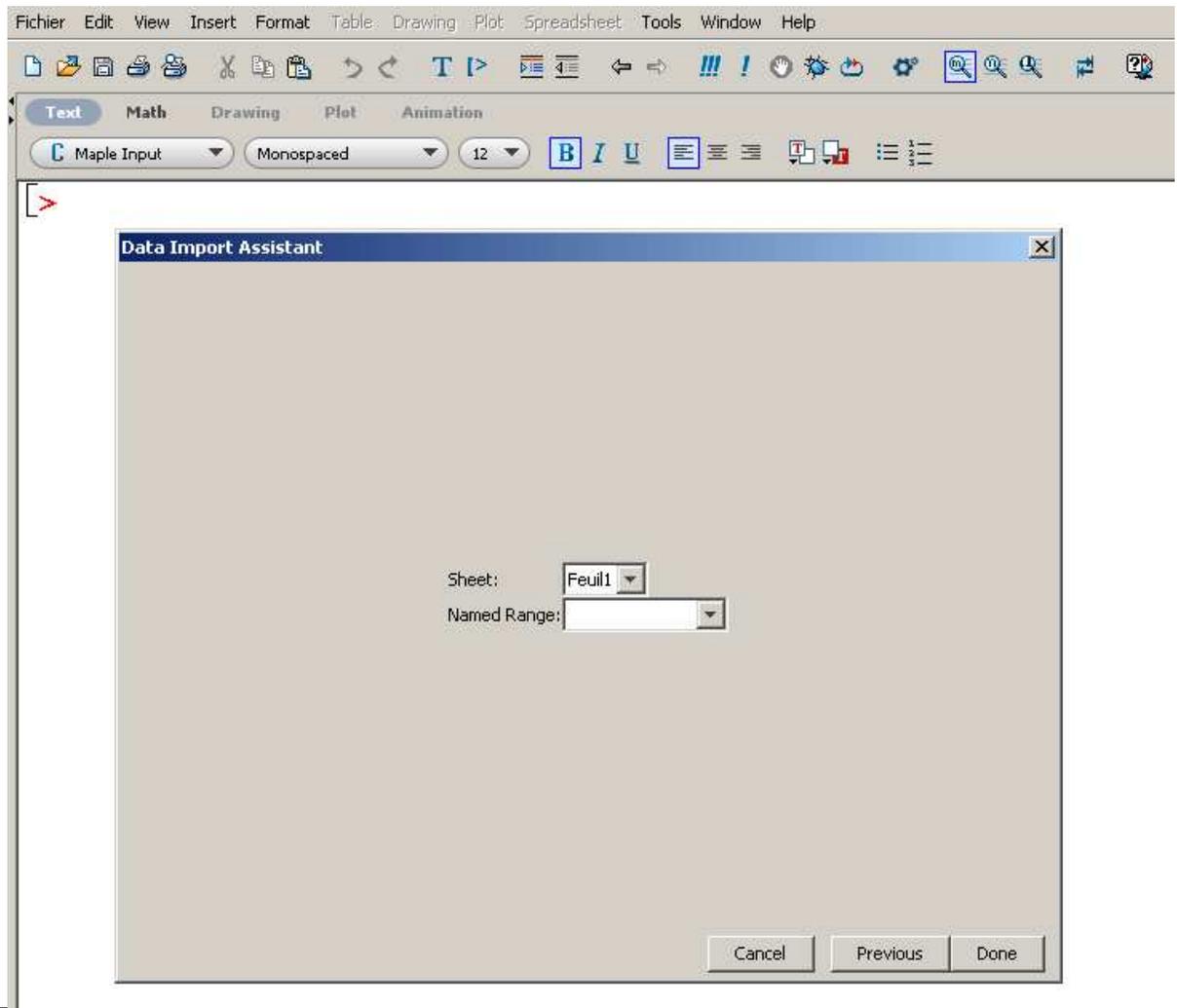
Importation à l'aide de l'assistant Import data

Le plus simple est d'utiliser Tools→Assistants→Import Data qui ouvre une boîte de dialogue Data Import Assistant. On remplit les champs Rechercher dans (il va sans dire que l'utilisateur

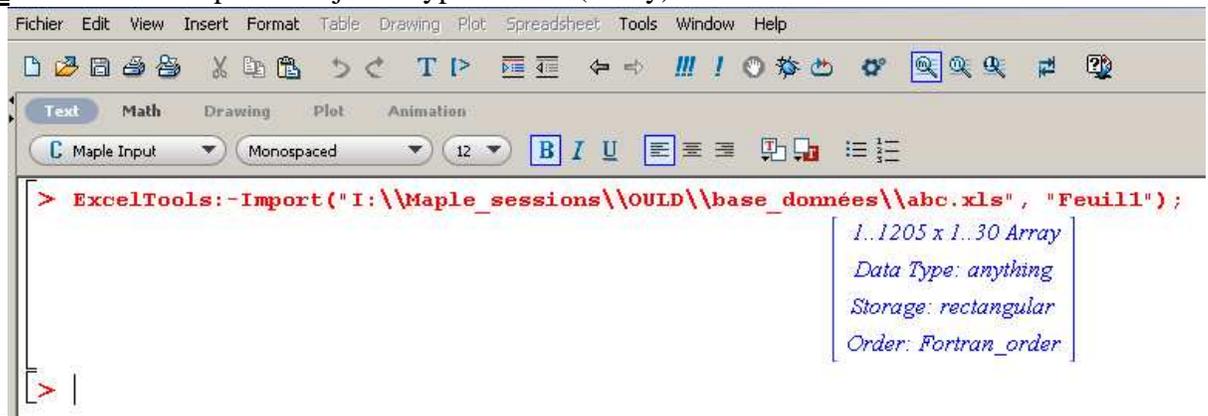
sait où se trouve exactement le fichier sur son disque dur!), Nom de fichier et Fichiers du type. Notons que les fichiers attendus ne sont pas seulement du type Excel. Maple prend aussi en charge les formats MATLAB, Image, Audio, Matrix Market et Delimited.



Une fois le fichier repéré et sélectionné, un clic sur Next amène à la page de choix d'une feuille de travail dans le fichier Excel. On peut aussi sélectionner et extraire des cellules particulières dans le champ Named Range.



En cliquant sur Done, Maple génère automatiquement une ligne de commande d'importation et affiche en output un objet du type tableau (Array) .



Importation à l'aide du paquetage Exceltools

L'assistant d'importation est en fait un raccourci de la commande **Import** disponible dans le paquetage **ExcelTools**. Sa syntaxe est **Import(fichier, feuille, cellules)** où :

1. **fichier** est une chaîne de caractères (string) donnant, entre guillemets, le chemin informatique complet menant au fichier .xls sur le disque dur. Par exemple :

The screenshot shows the Maple software interface. The command window contains the following command:

```
> ExcelTools:-Import("I:\\Maple_sessions\\OULD\\base_données\\abc.xls");
```

To the right of the command, a tooltip provides the following information:

- 1..1205 x 1..30 Array
- Data Type: anything
- Storage: rectangular
- Order: Fortran_order

The 'Browse Matrix' dialog box is open, showing a table with 15 rows and 7 columns. The columns are labeled 1 through 7. The data in the table is as follows:

	1	2	3	4	5	6	7
1	"trimestre"	"code OP "	"SOR"	"ouv/mois"	"SAPRE"	"SAPRIS"	"SAPRIS/
2	"T1 2002"	"BERE"	132.0	44.0	5135.0	5015.0	1671.666
3	"T1 2002"	"DABA"	97.0	32.33333...	4058.0	3913.0	1304.333
4	"T1 2002"	"DADE"	85.0	28.33333...	4178.0	4001.0	1333.666
5	"T1 2002"	"FRVA"	80.0	40.0	2870.0	2828.0	1414.0
6	"T1 2002"	"JEND"	112.0	56.0	3263.0	3242.0	1621.0
7	"T1 2002"	"LACZ"	146.0	48.66666...	6012.0	5877.0	1959.0
8	"T1 2002"	"LIBA"	124.0	41.33333...	4348.0	4247.0	1415.666
9	"T1 2002"	"MCHE"	175.0	58.33333...	4146.0	4106.0	1368.666
10	"T1 2002"	"PHPO"	108.0	36.0	3899.0	3882.0	1294.0
11	"T1 2002"	"RAUB"	55.0	27.5	4314.0	4272.0	2136.0
12	"T1 2002"	"SEDE"	116.0	38.66666...	4548.0	4306.0	1435.333
13	"T1 2002"	"SOMA"	92.0	30.66666...	4203.0	3746.0	1248.666
14	"T1 2002"	"STBE"	48.0	48.0	2744.0	2744.0	2744.0
15	"T1 2002"	"STED"	89.0	29.66666...	5046.0	4894.0	1631.333

Données en .txt

Il arrive que les données à traiter soient stockées sur le disque dur au format .txt. Dans ce cas, l'importation se fait via la fonction **readdata** qui requiert 3 arguments : le chemin d'accès au fichier, le format (integer, float ou string) et le nombre de colonnes à lire. Notons que seul le premier argument est obligatoire mais si le nombre de colonnes n'est pas précisé, Maple ne renvoie que la première colonne. Si le nombre de colonnes est spécifié, il transforme chaque ligne des données de départ en une liste, ce qui veut dire que le résultat de l'importation de données .txt est une liste de listes. L'exemple suivant le montre. Cette première requête correspond au cas où le nombre de colonnes n'est pas spécifiée. L'output est une liste de 1710 nombres (qui sont les T de la table de Durbin et Watson). Vue sa longueur, on termine la requête par **:** pour bloquer l'édition de la liste et on montre une partie de l'output dans un graphique.

```
> readdata
("I:\\Maple_sessions\\document_cours_maple_2009\\Durbin_Wats
```



```

1..1710 x 1..4 Array
Data Type: anything
Storage: rectangular
Order: Fortran_order

```

(1.2.1)

▼ Création de données statistiques

Avant d'éditer ses données "à la main", l'utilisateur doit parfaitement anticiper la tâche qu'il va effectuer car c'est de ce plan de travail que dépend le **type** Maple des données à saisir (éléments d'une liste, d'un tableau, d'une matrice, d'un vecteur, etc). En effet, les commandes statistiques Maple exigent chacune des arguments d'un type bien particulier et ces types sont variables d'une commande à l'autre. D'où le conseil : pour éviter des manipulations fastidieuses : il est impératif de choisir dès le départ le type adéquat pour stocker les données statistiques. En pratique, on constate que les commandes statistiques attendent le plus souvent des arguments du type liste ou liste de listes.

▼ Manipulation de données statistiques

Une fois que l'utilisateur dispose de ses données sous Maple, il doit, dans la quasi totalité des cas, se livrer à une gymnastique de manipulation des données. La première porte sur la duplication de cette base. La seconde sur l'extraction de lignes ou de colonnes. La troisième sur la concaténation.

▼ Duplication d'une base de données

Avant même de commencer son travail statistique, l'utilisateur a toujours intérêt à assigner sa base de données. Ceci fait, il peut juger utile de ne pas travailler directement sur sa base de données. Deux possibilités existent : soit il fait une copie de sa base initiale grâce à la commande **copy** (noter le "c" minuscule) suivie du nom affecté préalablement à la base entre parenthèses, soit il assigne un nom au nom déjà affecté à la base. Voyons cela sur un exemple :

```

> base_initiale:=ExcelTools:-Import
  ("I:\\Maple_sessions\\OULD\\base_données\\abc.xls",
  "Feuill1");#importation puis assignation de la base de
données

```

```

base_initiale :=
1..1205 x 1..30 Array
Data Type: anything
Storage: rectangular
Order: Fortran_order

```

(3.1.1)

```

> copie_base_initiale:=copy(base_initiale);# première
possibilité : copier de la base initiale

```

```

copie_base_initiale :=
1..1205 x 1..30 Array
Data Type: anything
Storage: rectangular
Order: Fortran_order

```

(3.1.2)

> **base_bis:=base_initiale;# deuxième possibilité: on assigne un nom au nom déjà affecté à la base initiale**

<i>base_bis :=</i>	<i>1..1205 x 1..30 Array</i> <i>Data Type: anything</i> <i>Storage: rectangular</i> <i>Order: Fortran_order</i>	(3.1.3)
--------------------	--	---------

Malgré les apparences, les deux procédés sont très différents. Dans le cas de la copie, les bases de données sont initialement identiques mais deviennent par la suite indépendantes : la copie peut être modifiée sans que la base originale ne soit affectée. Dans le deuxième cas, les deux bases restent liées : si un élément du tableau initial est modifié, l'élément correspondant du nouveau tableau se modifie aussi.

Extraction de données

Une analyse statistique exige souvent l'extraction de données correspondant à des variables particulières. Il existe plusieurs méthodes d'extraction de données. On en détaille deux ici : la première porte sur des tableaux, tandis que la seconde concerne les matrices.

Extraction de données dans un tableau

On suppose dans cette section que la base de données est du type Array. Pour récupérer une variable, il est nécessaire de savoir si elle se trouve en ligne ou en colonne.

Dans le cas où la variable se trouve en colonne, il faut préalablement créer et assigner un vecteur colonne de format $(n, 1)$ où n correspond au nombre de lignes du tableau de départ. Rappelons que la taille du tableau se lit directement dans l'output relatif à l'importation des données ou encore en demandant explicitement à Maple la dimension du tableau à l'aide de la commande **rtable_dims** suivie du nom du tableau entre parenthèses. Dans un second temps, on extrait la variable souhaitée en utilisant la commande **Copy** (noter le "C" majuscule) qui se trouve dans le paquetage **ArrayTools**. Les arguments requis par cette commande sont, dans l'ordre : la base de données initiale (**et non sa copie**), le numéro de la colonne contenant la variable diminué de 1, le tout entre parenthèses, multiplié par le nombre de lignes du tableau de départ, enfin le nom du vecteur colonne construit préalablement. Pour afficher le contenu, il faut appeler le vecteur colonne par son nom. Pour ensuite isoler la variable extraite, un nom peut être assigné au résultat.

Dans l'exemple qui suit, on cherche à extraire la variable "code OP" du tableau base_initiale. Elle correspond à la deuxième colonne du tableau.

> **A1:=Vector[column](1205,datatype=anything);#création d'un vecteur-colonne à 1205 éléments nuls**

<i>A1 :=</i>	<i>1 .. 1205 Vector_{column}</i> <i>Data Type: anything</i> <i>Storage: rectangular</i> <i>Order: Fortran_order</i>	(3.2.1.1)
--------------	--	-----------

> **with(ArrayTools):#chargement du paquetage
Copy(base_initiale,(2-1)*1205,A1);#extraction de la variable "code OP"**

```
Var_code:=A1;#assignation
```

```
Var_code := [ 1 .. 1205 Vectorcolumn
              Data Type: anything
              Storage: rectangular
              Order: Fortran_order ] (3.2.1.2)
```

Passons au cas où la variable désirée se trouve en ligne. On commence par générer un vecteur ligne de format $(1, k)$ où k est le nombre de colonnes du tableau de données. L'extraction se fait toujours avec la commande **Copy** et la syntaxe est quasi identique : nom de la base initiale, numéro de la ligne contenant la variable moins un, le nombre de lignes de la base initiale et le nom du vecteur ligne. Pour illustrer cela, extrayons la quatrième ligne de la base nommée `base_initiale` :

```
> A2:=Vector[row](30,datatype=anything);#création d'un
vecteur ligne à 30 éléments nuls
```

```
A2 := [ 1 .. 30 Vectorrow
        Data Type: anything
        Storage: rectangular
        Order: Fortran_order ] (3.2.1.3)
```

```
> Copy(base_initiale,4-1,1205,A2);#extraction de la
quatrième ligne du tableau base_initiale
ligne_4:=A2;#assignation
```

```
ligne_4 := [ 1 .. 30 Vectorrow
             Data Type: anything
             Storage: rectangular
             Order: Fortran_order ] (3.2.1.4)
```

Extraction de données dans une matrice

Un autre moyen de sélectionner la ou les variable(s) souhaitée(s) est de travailler sur des matrices. Comme son nom l'indique, l'instruction **convert** (**nom_du_tableau_initial, Matrix**) permet de convertir le tableau de données en une matrice. La suppression de colonnes ou de lignes se fait commodément avec les commandes **DeleteColumn** et **DeleteRow** du paquetage **LinearAlgebra**. Ces deux fonctions attendent deux arguments : une matrice puis la liste des colonnes ou de lignes à supprimer. Pour montrer qu'on obtient le même résultat qu'avec la fonction **Copy**, nous allons conserver dans l'exemple ci-dessous uniquement la variable "code OP" qui se trouve dans la deuxième colonne. Pour ce faire, il faudrait supprimer toutes les autres variables. La génération d'une séquence reprenant les numéros des colonnes à éliminer permet de réaliser un gain de temps.

```
> var_supp:=[1,seq(3..30)];#liste des colonnes à supprimer
var_supp := [1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30] (3.2.2.1)
```

```
> base_matrice:=convert(base_initiale,Matrix);#conversion
du tableau en matrice
```

```
base_matrice :=  $\left[ \begin{array}{l} 1205 \times 30 \text{ Matrix} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array} \right]$  (3.2.2.2)
```

```
> with(LinearAlgebra):
var_code_bis:=DeleteColumn(base_matrice,var_supp);
```

```
var_code_bis :=  $\left[ \begin{array}{l} 1205 \times 1 \text{ Matrix} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array} \right]$  (3.2.2.3)
```

La démarche est identique pour sélectionner une variable en ligne.

```
> lignes_sup:=[1,2,3,seq(5..1205)];#liste des lignes à
supprimer. L'affichage est bloqué.
```

```
> ligne_4_bis:=DeleteRow(base_matrice,lignes_sup);
```

```
ligne_4_bis :=  $\left[ \begin{array}{l} 1 \times 30 \text{ Matrix} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array} \right]$  (3.2.2.4)
```

Concaténation de données

On peut parfois vouloir créer une nouvelle base de données à partir de bases déjà existantes, par exemple en ajoutant des individus, des variables, etc. Maple donne la possibilité de concaténer ou de fusionner des bases de données.

La concaténation verticale consiste à mettre les fichiers les uns en dessous des autres. Cette manœuvre peut être utile lorsque les fichiers n'ont pas été saisis par les mêmes individus et que l'on souhaite créer une base unique. Pour que l'opération soit possible, il faut évidemment que les fichiers de départ contiennent le même nombre de variables en colonnes. La concaténation verticale se fait à l'aide de la commande **Concatenate** du paquetage **ArrayTools**.

Ses arguments sont : **1** (signalant une demande de concaténation verticale) puis les noms des fichiers dans l'ordre dans lequel on souhaite les superposer.

A titre d'exemple, chargeons deux bases de données placés dans la même feuille Excel intitulée cde.xls mais dans deux feuilles différentes.

```
> fichier1:=ExcelTools:-Import
("I:\\Maple_sessions\\OULD\\base_données\\cde.xls",
"Feuill1");
fichier2:=ExcelTools:-Import
("I:\\Maple_sessions\\OULD\\base_données\\cde.xls",
```

```
"Feuil2");
```

$$\begin{aligned} \text{fichier1} &:= \begin{bmatrix} 23.0 & 12.0 & 45.0 & 36.0 \\ 7.0 & 23.0 & 9.0 & 10.0 \\ 54.0 & 76.0 & 50.0 & 32.0 \\ 77.0 & 21.0 & 80.0 & 22.0 \end{bmatrix} \\ \text{fichier2} &:= \begin{bmatrix} 1234.0 & 546.0 & 778.0 & 990.0 \\ 765.0 & 234.0 & 988.0 & 434.0 \\ 465.0 & 786.0 & 543.0 & 231.0 \end{bmatrix} \end{aligned} \quad (3.3.1)$$

Leur concaténation verticale se fait très simplement :

```
> with(ArrayTools):  
Con_ver:=Concatenate(1,fichier1,fichier2);
```

$$\text{Con_ver} := \begin{bmatrix} 23.0 & 12.0 & 45.0 & 36.0 \\ 7.0 & 23.0 & 9.0 & 10.0 \\ 54.0 & 76.0 & 50.0 & 32.0 \\ 77.0 & 21.0 & 80.0 & 22.0 \\ 1234.0 & 546.0 & 778.0 & 990.0 \\ 765.0 & 234.0 & 988.0 & 434.0 \\ 465.0 & 786.0 & 543.0 & 231.0 \end{bmatrix} \quad (3.3.2)$$

Dans le cas où les fichiers ne contiennent pas le même nombre de colonnes, Maple refuse l'opération et renvoie un message d'erreur :

```
> fichier3:=ExcelTools:-Import  
("I:\\Maple_sessions\\OULD\\base_données\\cde.xls",  
"Feuil3");  
Concatenate(1,fichier1,fichier3);
```

$$\text{fichier3} := \begin{bmatrix} 24.0 & 68.0 & 98.0 & 12.0 & 65.0 & 187.0 \\ 90.0 & 12.0 & 54.0 & 89.0 & 32.0 & 43.0 \\ 35.0 & 93.0 & 67.0 & 47.0 & 44.0 & 64.0 \end{bmatrix}$$

Error, (in ArrayTools:-Concatenate) number of columns must match

La concaténation horizontale consiste à mettre côte à côte des fichiers existants, étant entendu qu'ils contiennent les mêmes individus mais des variables différentes. Elle se fait encore avec la commande **Concatenate**, à ceci près que le 1 est remplacé par 2.

```
> fichier4:=ExcelTools:-Import  
("I:\\Maple_sessions\\OULD\\base_données\\cde.xls",  
"Feuil4");  
Con_hor:=Concatenate(2,fichier1,fichier4);
```

$$\begin{aligned}
 \text{fichier4} &:= \begin{bmatrix} 20.0 & 40.0 & 70.0 & 43.0 & 98.0 & 12.0 \\ 66.0 & 56.0 & 89.0 & 76.0 & 43.0 & 112.0 \\ 23.0 & 55.0 & 99.0 & 87.0 & 56.0 & 20.0 \\ 44.0 & 30.0 & 50.0 & 21.0 & 78.0 & 99.0 \end{bmatrix} \\
 \text{Con_hor} &:= \begin{bmatrix} 23.0 & 12.0 & 45.0 & 36.0 & 20.0 & 40.0 & 70.0 & 43.0 & 98.0 & 12.0 \\ 7.0 & 23.0 & 9.0 & 10.0 & 66.0 & 56.0 & 89.0 & 76.0 & 43.0 & 112.0 \\ 54.0 & 76.0 & 50.0 & 32.0 & 23.0 & 55.0 & 99.0 & 87.0 & 56.0 & 20.0 \\ 77.0 & 21.0 & 80.0 & 22.0 & 44.0 & 30.0 & 50.0 & 21.0 & 78.0 & 99.0 \end{bmatrix}
 \end{aligned}
 \tag{3.3.3}$$

Par contre, lorsque les fichiers d'origine ne contiennent pas le même nombre de lignes, la concaténation horizontale n'est pas réalisée.

```

> Concatenate(2,fichier1,fichier2);
Error, (in ArrayTools:-Concatenate) number of rows must
match

```

▼ Exportation de données statistiques

L'exportation des données vers Excel est facile à réaliser. Un clic droit sur l'output et la sélection de Browse ouvrent la fenêtre Browse Matrix, qui permet d'accéder au bouton Export. Dans la seconde fenêtre : Export to Excel, l'utilisateur doit informer les champs de destination, de nom de la feuille Excel, etc. Un clic sur OK puis sur Done terminent l'opération d'exportation. L'autre possibilité, équivalente, fait appel à la commande **Export** du paquetage **ExcelTools**.