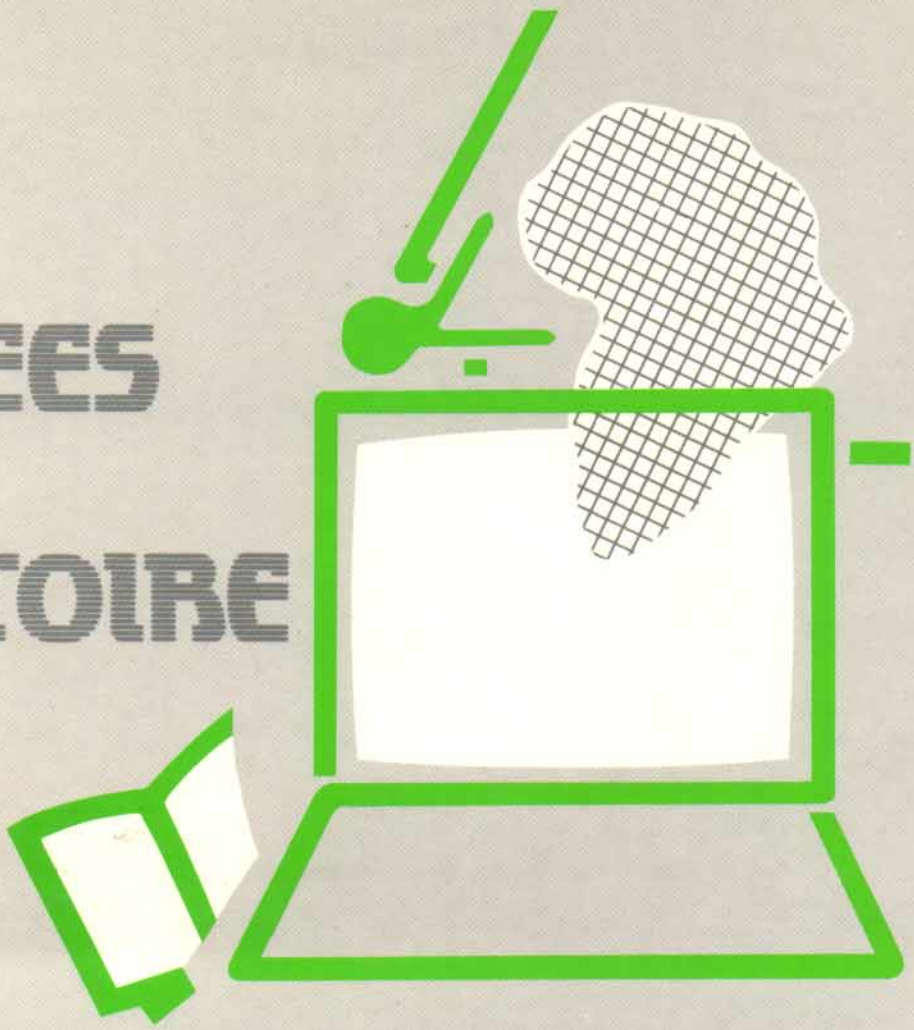
LES DONNEES ET LE TERRITOIRE



INITIATION AU TRAITEMENT INFORMATIQUE DES DONNEES SPATIALISEES.

Ph. WANIEZ

ORSTOM
RECLUS

Éditions de l'ORSTOM

INSTITUT FRANÇAIS DE RECHERCHE SCIENTIFIQUE POUR LE DÉVELOPPEMENT EN COOPÉRATION

Philippe WANIEZ
Attaché de recherche à l'ORSTOM
Département Indépendance alimentaire
U.R. 502

LES DONNÉES ET LE TERRITOIRE

INITIATION AU TRAITEMENT INFORMATIQUE DES DONNÉES SPATIALISÉES

avec la collaboration de

Gérard DANDOY
Chargé de recherche à l'ORSTOM

et

Violette CABOS
Ingénieur cartographe au CNRS

Préface de Jacques CHAMPAUD
Directeur de recherche à l'ORSTOM

Éditions de l'ORSTOM
INSTITUT FRANÇAIS DE RECHERCHE SCIENTIFIQUE POUR LE DÉVELOPPEMENT EN COOPÉRATION
Collection **INITIATIONS - DOCUMENTATIONS TECHNIQUES** n° 67
PARIS 1986

ISSN : 0071-9021
ISBN : 2-7099-0815-8

Philippe WANIEZ
Attaché de recherche à l'ORSTOM
Département Indépendance alimentaire
U.R. 502

LES DONNÉES ET LE TERRITOIRE

INITIATION AU TRAITEMENT INFORMATIQUE DES DONNÉES SPATIALISÉES

avec la collaboration de

Gérard DANDOY
Chargé de recherche à l'ORSTOM

et

Violette CABOS
Ingénieur cartographe au CNRS

Préface de Jacques CHAMPAUD
Directeur de recherche à l'ORSTOM

Coproduction ORSTOM – GIP RECLUS

Institut Français de Recherche Scientifique pour le Développement en Coopération.

Groupement d'Intérêt Public Réseau d'Étude des Changements dans les Localisations et les Unités Spatiales

Préface

Le traitement des données spatialisées est une préoccupation ancienne des géographes, mais aussi de tous ceux qui s'intéressent à l'espace et veulent traduire en cartes les résultats d'enquêtes quantitatives ou d'observations d'ordre qualitatif. Critiquer les données, les classer, les corréler font partie du travail de base de nombreux chercheurs. Il en est de même de la cartographie. Ce sont des tâches qui, cependant, demandent un investissement en temps considérable et imposent de trier entre les cartes pour ne montrer que les plus significatives.

L'avènement de l'informatique permet de traiter plus vite et mieux un plus grand nombre de données chiffrées. Mais surtout il facilite la traduction cartographique de ces résultats. Dès lors que l'on peut éviter le travail souvent fastidieux du dessin cartographique, on hésitera moins à multiplier les cartes, à tester des corrélations nouvelles, à élaborer des synthèses partielles. L'infographie n'est certes qu'une technique parmi d'autres. Elle ne remplace ni la critique des chiffres à traiter ni la réflexion sur les commentaires à établir. Elle est cependant de nature à bouleverser bien des habitudes et la rigueur est plus que jamais nécessaire.

Le présent ouvrage est un manuel d'introduction au traitement statistique et cartographique des données spatialisées. Il s'attache à montrer le parti que l'on peut tirer d'un logiciel qui parait bien adapté, SAS. Celui-ci n'est pas le seul sur le marché ; aussi l'intérêt de ce livre est-il moins de montrer l'usage qui peut être fait de ce logiciel particulier que de traiter le problème dans son ensemble. Dans cette perspective, c'est plus en amont, dès la phase de collecte des données, qu'il faut se préoccuper de l'exploitation informatique.

Ce travail est le résultat pour partie de l'enseignement donné par Philippe Waniez à l'Université de Paris I et aussi de mises en commun et de discussions avec l'équipe de la Maison de la Géographie de Montpellier. Il montre aussi la continuité du travail commencé par l'ORSTOM et dont il est rendu compte par ailleurs (1). Il met l'accent sur la formation nécessaire des chercheurs dont il ne constitue qu'un élément.

Jacques Champaud

(1) DANDOY (Gérard) (éd.), 1986, *Traitement des données spatialisées, l'infographie à l'ORSTOM.* Paris, ORSTOM, coll. Colloques et séminaires, 300 p.

Table des matières

Liste des figures

Liste des tableaux

Liste des cartes

Introduction

Le présent ouvrage est un manuel pratique s'adressant à tous ceux qui ont le souci d'aborder l'exploitation des données statistiques spatialisées en ayant une certaine autonomie vis à vis des informaticiens. Après plusieurs années d'expérience du service informatique en milieu universitaire, il me semble réaliste de guider les pas du néophyte tout en lui évitant de se perdre dans une documentation difficile d'accès, volumineuse, et le plus souvent, en anglais. Initialement, il s'agissait d'un simple support de cours réalisé à l'intention des chercheurs de l'ORSTOM inscrits au stage d'Introduction au Traitement Informatique des Données Spatialisées organisé à la Maison de la Géographie de Montpellier. A la suite de nombreuses demandes, il est apparu très utile de proposer un volume plus complet, couvrant non seulement le domaine limité de l'organisation des données, mais aussi celui de leur traitement statistique et cartographique. On a cherché par là à combler une lacune difficilement compréhensible en raison de l'extraordinaire essor de l'usage de l'informatique en sciences humaines. Ce manuel voudrait être pour l'informatique ce qu'est, à la statistique, l'excellent livre du Groupe Chadule « Initiation aux méthodes statistiques en géographie » dont on attend la prochaine réédition. La complémentarité de ces deux manuels est très grande, notamment pour tout ce qui concerne l'analyse des matrices d'information spatiale. Le lecteur ne trouvera donc pas ici l'exposé des méthodes statistiques, mais seulement celui des techniques informatiques nécessaires à leur mise en oeuvre dans le cadre spécifique des systèmes informatiques SAS et ADDAD.

Ce livre s'adresse en premier lieu aux chercheurs et ingénieurs des établissements de recherche en sciences humaines désireux de profiter de toute la puissance de l'informatique pour organiser et traiter leur information statistique. Les étudiants des second et troisième cycles universitaires de géographie et de sociologie, mais peut-être aussi ceux de mathématiques appliquées aux sciences sociales profiteront de cette lecture pour complèter leur apprentissage du traitement des données. Enfin, tous les « analystes de données » cherchant à faire connaissance avec le Statistical Analysis System (SAS) touveront ici le guide de leurs premiers pas dans l'univers SAS.

D'une manière très schématique, l'usage qui est habituellement fait de l'outil informatique en sciences humaines peut revêtir trois formes par ailleurs non exclusives : la programmation en vue d'une application particulière, le recours à une bibliothèque de programmes plus divers, l'emploi d'un progiciel assurant de multiples fonctions. Cette dernière pratique semble être dans la majorité des cas la solution d'avenir, celle à laquelle il est indispensable de se préparer dès aujourd'hui : mis à part des situations très particulières, la programmation en langage évolué du genre FORTRAN ou BASIC verra sa place s'amenuiser pendant que les bibliothèques de programmes s'intègreront petit à petit aux grands progiciels dont elles ne constitueront à terme que des modules supplémentaires.

Pour montrer précisément comment il faut procéder pour obenir des résultats (des « outputs »), l'ensemble du texte s'appuie sur la réalisation de programmes dans le langage du Statistical Analysis System (SAS). Celui-ci a été choisi parmi d'autres (SPSS notamment) en raison de la grande variété de ses fonctions ; de plus, son usage en cours de généralisation à tous les domaines d'application de la statistique, tant dans les gros centres informatiques que sur les micro-ordinateurs de type IBM PC, lui assure le statut d'un standard de fait pour les années à venir.

Le plan adopté ventile les 14 chapitres en trois parties à peu près égales : la prépartion des données, l'analyse statistique, la cartographie automatique.

Par préparation des données, on entend la réalisation d'une base de données à partir des tableaux statistiques initiaux. Cette étape est indispensable à qui veut rationaliser son rapport à l'informatique pour limiter le temps et le coût du traitement. A cette fonction première, il faut adjoindre la construction d'indicateurs élémentaires (rapports, pourcentages...). Autant dire que la première partie s'adresse à tous ceux qui ont affaire avec les statistiques. Cinq chapitres conduisent,

à partir de l'examen des caractéristiques des matrices d'information spatiale (chapitre 1) et du système SAS (chapitre 2), aux premiers contacts avec l'informatique (chapitre 3) pour construire une base de données (chapitre 4). Le cinquième et dernier chapitre de la première partie montre comment procéder à l'élaboration d'un tableau de travail en vue de son traitement statistique, à partir des tableaux initiaux figurant dans la base de données.

Toute la seconde partie est axée sur l'analyse statistique descriptive univariée et multivariée sur les tableaux élaborés précédemment. Après un brève présentation des procédures SAS (chapitre 6), les deux principales procédures de statistique descriptive univariée font l'objet d'un examen approfondi (chapitre 7). La régression et la corrélation sont traitées dans le chapitre n° 8. Enfin, les techniques de l'analyse des données proposées par la bibliothèque de l'ADDAD, sous SAS, constituent le neuvième chapitre. Pour conclure cette seconde partie, plusieurs points de programmation sont précisés ; le macro langage et le langage matriciel font également l'objet d'un bref exposé.

En définitive, seule la troisième partie, consacrée à l'informatique graphique et à la cartographie automatique apparaît comme plus spécifique à la géographie ; elle ne peut cependant se dispenser des deux précédentes. Les chapitres 11 à 14 présentent successivement les unités informatiques graphiques, les fonctions principales de SAS/GRAPH, la constitution des fonds de carte et les projections cartographiques, la réalisation des cartes choroplèthes.

Pour renforcer la cohérence de l'ensemble, tous les exemples s'appuient sur des données communes relatives à l'endettement des pays les moins avancés. Enfin, lorsque cela est apparu indispensable, on s'est efforcé d'indiquer une ou deux références bibliographiques pour que le lecteur puisse complèter son information.

En achevant cette introduction, je tiens à remercier l'ORSTOM et plus particulièrement la sous-commission scientifique de géographie : J. Champaud et G. Dandoy ont rendu possible l'organisation des stages « Traitement des données spatialisées » ainsi que la publication du présent ouvrage. Durant les quatre mois qu'a nécessité la rédaction, j'ai pu bénéficier de l'ensemble des moyens informatiques disponibles à la Maison de la Géographie de Montpellier ; à son direceur, R. Brunet, ainsi qu'à tous les techniciens, ingénieurs et chercheurs, j'adresse mes plus vifs remerciements pour leurs conseils, critiques et informations. Enfin, ce travail n'aurait pas sa forme actuelle sans les conseils de - presque - tous les jours que m'ont prodigués G. Dandoy et V. Cabos.

1. Elaborer une matrice d'information spatiale

Organiser ses données en une matrice d'information spatiale est une étape essentielle , délicate, conditionnant en grande partie la qualité du travail scientifique. Il apparaît donc très utile de préciser la forme et le contenu de ces tableaux de données.

1.1. La mesure comme action et comme produit

La mesure vue comme l'action de mesurer est le lien essentiel entre l'abstraction conceptuelle et ce qui est directement observable. Les concepts sont souvent des généralisations échappant à l'analyse statistique. Il est alors indispensable d'en donner une définition opérationnelle spécifiant la manière de procéder aux mesures (fig. 1.1). Ici se pose le problème de l'adéquation de la mesure au concept : avons-nous bien mesuré ce que nous voulions mesurer ?

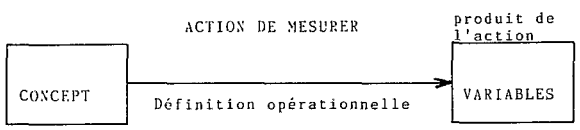


Fig. 1.1. – *Du concept aux variables.*

Conceptualisation et observation constituent deux étapes très importantes du travail de recherche scientifique. Leur produit s'appuie sur quatre niveaux d'échelles. La plus rudimentaire, l'échelle nominale, représente un phénomène sous forme de modalités ; les classifications régionales en constituent un excellent exemple. Les modalités utilisées pour construire une échelle nominale doivent être exhaustives, c'est-à-dire couvrir l'ensemble des situations ; de plus, il est nécessaire qu'elles s'excluent mutuellement : une observation doit posséder une et une seule modalité. Une échelle binaire est un cas particulier à deux modalités seulement. Pour représenter ces modalités, il est préférable d'employer des chaînes de caractères afin de les distinguer des valeurs numériques des autres échelles de mesure.

Une échelle ordinale est établie lorsque toutes les modalités sont représentées par leur rang dans l'ordre croissant ou décroissant de leurs valeurs. On parle d'ordre partiel lorsque le critère de rangement s'énonce sous la forme « plus petit que » ou « plus grand que », ou plus précisément, lorsque plusieurs observations peuvent prendre le même rang ; à l'inverse, cet ordre est total si un rang ne peut être affecté qu'à une observation et une seule. Même dans ce dernier cas, la différence quantitative entre observations successives reste inconnue. La représentation numérique des rangs fait appel à l'ensemble des nombres entiers privé du zéro.

Dans une échelle d'intervalles, qui ne comporte pas de zéro naturel,rien ne s'oppose au calcul d'une valeur numérique exprimant la différence entre deux observations. On préfère, en général, utiliser des échelles de rapport possédant une vraie valeur zéro et rapportant les différences entre observations à une même unité.

La plupart des échelles d'intervalles ou de rapport sont continues, c'est-à-dire qu'elles prennent n'importe quelle valeur à l'intérieur de certaines limites. A l'inverse, les échelles discrètes n'ont que des valeurs entières.

1.2. Le conditionnement de l'information dans des matrices d'information spatiale

Une matrice d'information spatiale résulte de la mise en correspondance de deux ensembles : celui des unités

spatiales et celui des attributs géographiques. Les unités spatiales peuvent être notamment des points ou des surfaces. Il s'agit d'unités d'observation ou, plus simplement, d'observations. Si ce sont des points, les observations seront disjointes ; on parlera d'un semis dans l'espace comme les semis des villes. Dans le cas de surfaces, les observations seront des mailles irrégulières comme cela est le cas pour les maillages administratifs ; leurs limites ne devront pas se recouper : il s'agira donc d'une partition de l'espace. On nommera I l'ensemble des observations, i la ième observation ; i prendra des valeurs entières de 1 à n, n étant le nombre total d'observations.

Les attributs géographiques sont des propriétés mesurables des observations propres à donner une représentation exhaustive du phénomène étudié. Il peut s'agir soit d'un ensemble dans son entier, soit de la partition de cet ensemble en catégories pertinentes ; dans les deux cas, on qualifiera de « variables » ces attributs géographiques. On nommera J l'ensemble des variables, j la jème observation ; j prendra des valeurs entières de 1 à p, p étant le nombre total de variables.

Toute matrice d'information spatiale est donc un tableau X composé de n lignes et de p colonnes et renfermant ainsi un nombre n*p de cases. On note xij la case correspondant à la ième observation et à la jème variable (fig. 1.2).

On pourrait imaginer d'établir une matrice d'information spatiale en ajoutant un troisième ensemble, l'ensemble K du temps. On noterait k le kième temps ; k prendrait des valeurs entières de 1 à t, t étant le nombre total de sections dans le temps. Le résultat serait un bloc matriciel spatial X possédant un nombre n*p*t de cases. xijk correspondrait à la ième observation, à la jème variable et au kième temps. En pratique, on juxtaposerait les matrices d'information spatiale correspondant aux différents temps (fig. 1.3). L'élaboration de tels blocs multiplie les difficultés rencontrées dans le cas, plus simple, des matrices d'information spatiale. Sur l'ensemble I des observations se pose le problème de la permanence temporelle des mailles découpant l'espace lorsqu'il s'agit de surfaces, ou bien de la permanence des semis de points. Par exemple, les îlots urbains sont souvent modifiés d'un recensement à l'autre ; c'est aussi le cas pour les unités urbaines qui croissent par adjonction des zones périphériques. En ce qui concerne l'ensemble J des variables, il faut veiller à la permanence du contenu des catégories ; par exemple, il est impossible de comparer directement les catégories socio-professionnelles du recensement de la population française de 1975 avec les PCS de celui de 1982. Enfin, sur l'ensemble K des temps, il est souvent impossible de disposer de l'information selon une périodicité constante (annuelle...) ; cela rend très difficile l'élaboration des modèles spatiaux et temporels.

1.3. L'endettement des Pays les Moins Avancés (PMA) : trois exemples de matrices d'information spatiale

Afin de préciser la définition des matrices l'information spatiale, il paraît utile d'en donner quelques exemples. Les trois présentés ci-après serviront de support de travaux pratiques ; ils ont trait à 23 pays parmi les moins avancés du monde (carte 1.1).

a. Matrice CODPOP (tab. 1.1)

Composée de 23 observations (n = 23) et de 4 variables (p = 4), elle permet d'identifier les pays les moins avancés.

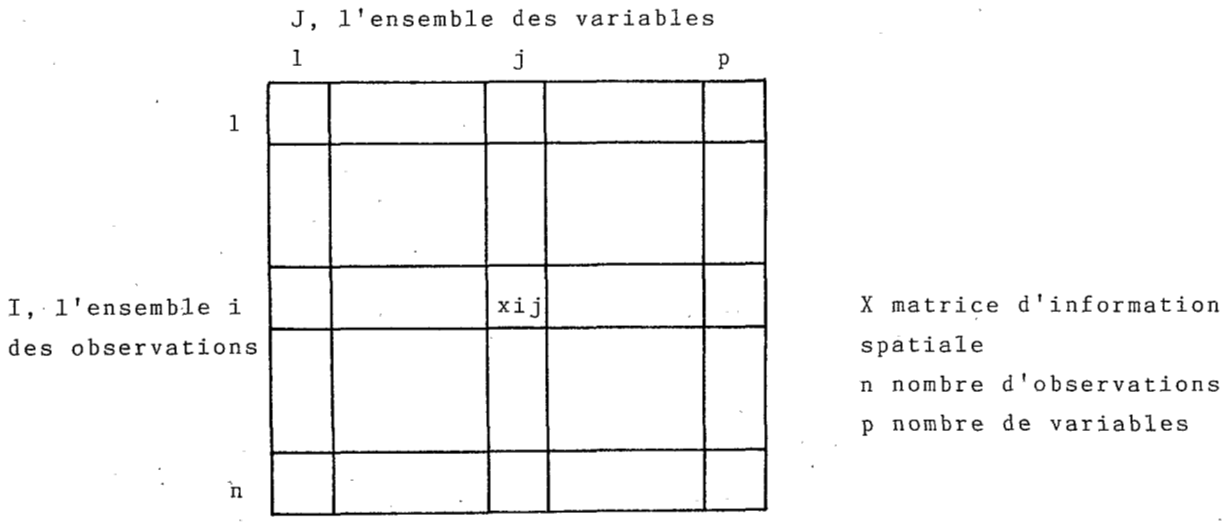
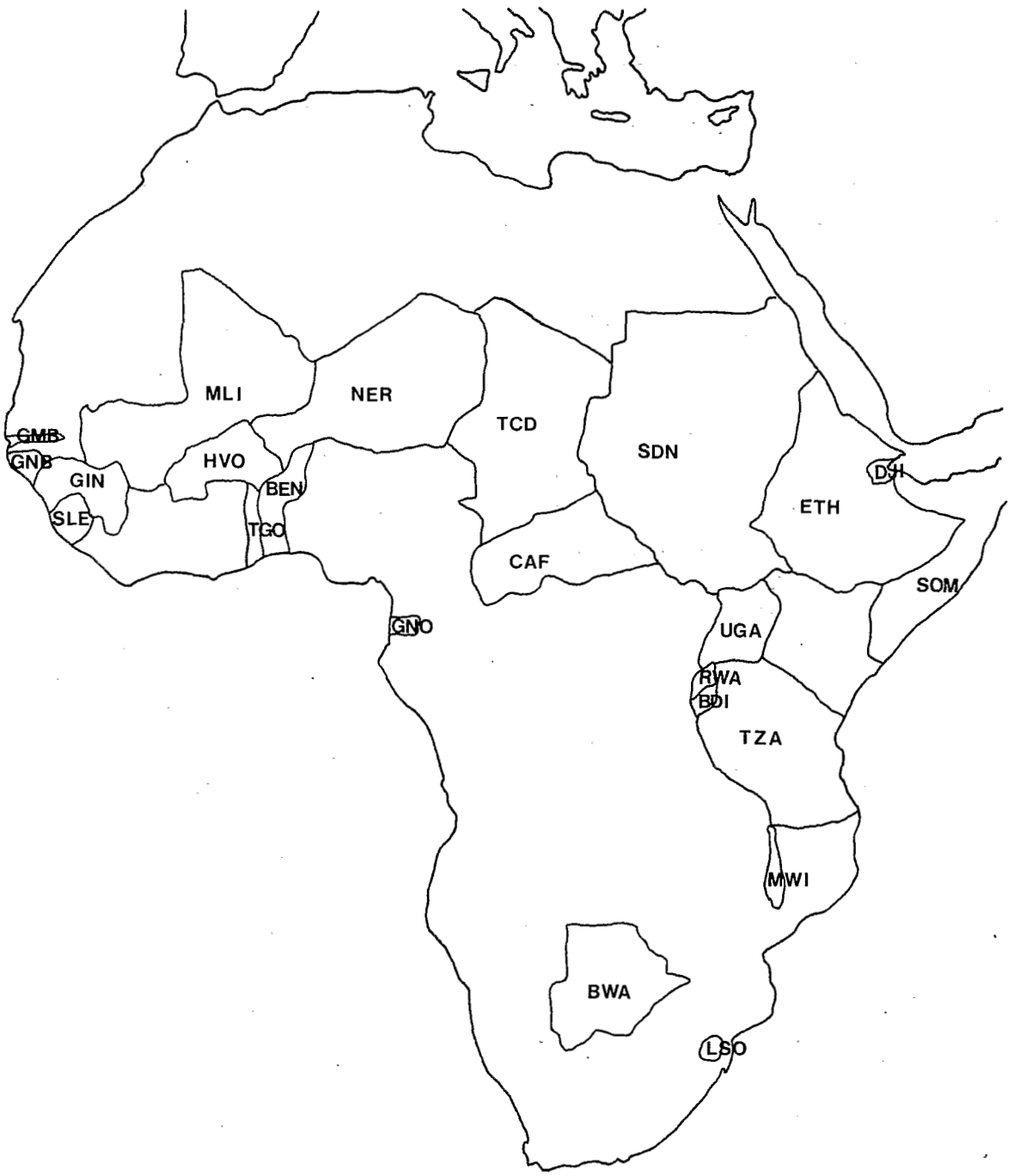


Fig. 1.2. – *Les éléments d'une matrice d'information spatiale.*



Carte 1.1. – *23 pays parmi les moins avancés d'Afrique.*

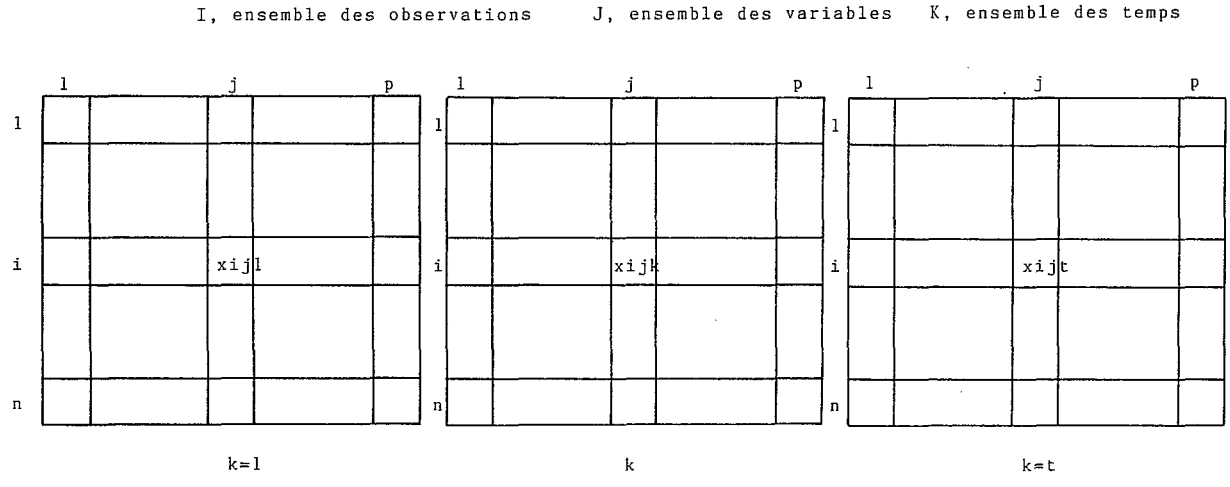


Fig. 1.3. – *Le bloc matriciel spatial découpé en matrices d'information spatiale.*

variable
n° 1 : identificateurs des Etats, sur 3 caractères, conformes à la norme ISO de 1981.
n° 2 : population (en millions d'habitants) en 1982.
n° 3 : taux d'accroissement annuel (%) 1970-1982.
n° 4 : noms des Etats en clair.

Les variables n° 1 et 4 sont discrètes, alphanumériques, relevées sur des échelles nominales ; la variable n° 2 qui est un dénombrement, est continue, numérique, relevée sur une échelle d'intervalles ; la variable n° 3 est continue, numérique, relevée sur une échelle de rapports.

b. Matrice DETTES (tab. 1.2)

Les données figurant dans cette matrice sont issues de la revue Marchés Tropicaux du 6/09/1985. Constituée de 23 observations et 7 variables, elle

BEN	95.0	76.0	4.4	110.1	566.4	51.8
BWA	101.6	92.9	8.7	16.1	387.8	30.7
HVO	213.0	201.0	12.0	43.5	343.8	20.4
BDI	130.5	120.6	6.0	30.4	227.7	5.5
DJI	58.3	54.8	3.5	0.3	43.1	3.3
ETH	216.7	199.8	0.1	19.4	897.6	61.5
GMB	50.0	43.0	4.6	-2.6	151.4	7.2
GIN	66.6	60.2	5.2	20.3	1283.7	79.3
GNB	69.2	59.7	8.5	0.8	126.8	3.5
GNQ	14.0	13.0	1.0	-4.9	0.0	0.0
LSO	89.7	86.8	2.9	3.9	152.2	11.2
MWI	121.5	121.4	0.1	7.0	779.1	7.3
MLI	206.9	159.3	35.2	9.2	828.7	12.0
NER	254.0	167.6	84.8	44.4	691.6	149.9
UGA	133.2	127.4	5.7	34.0	546.2	124.0
CAF.	89.9	88.7	1.2	11.5	215.1	4.0
TZA	710.1	656.6	39.0	56.8	2001.4	118.9
RWA	150.8	148.9	1.9	3.1	194.8	5.7
SLE	79.8	80.8	1.4	2.3	317.9	41.2
SOM	468.3	279.0	183.3	158.2	978.3	32.6
SDN	743.1	554.4	185.7	68.6	5473.1	91.3
TCD	64.7	60.9	3.8	-1.7	178.3	1.2
TGO	77.2	73.4	3.8	21.4	812.0	35.3

Tab. 1.2. – *La matrice DETTES.*

BEN	3.62	2.5	BENIN
BWA	0.97	3.8	BOTSWANA
HVO	6.36	1.4	BURKINA-FASO
BDI	4.26	1.8	BURUNDI
DJI	0.33	6.4	DJIBOUTI
ETH	32.78	2.4	ETHIOPIE
GMB	0.63	2.9	GAMBIE
GIN	5.06	2.1	GUINEE
GNB	0.85	4.1	GUINEE-BISSAU
GNQ	0.37	2.0	GUINEE EQUATORIALE
LSO	1.41	2.4	LESOTHO
MWI	6.27	2.6	MALAWI
MLI	7.34	2.1	MALI
NER	5.61	2.6	NIGER
UGA	14.12	3.1	OUGANDA
CAF	2.39	2.1	REP. CENTRAFRICAINE
TZA	20.23	3.4	TANZANIE
RWA	5.51	3.3	RWANDA
SLE	3.41	1.6	SIERRA LEONE
SOM	5.08	5.1	SOMALIE
SDN	19.79	3.0	SOUDAN
TCD	4.68	2.1	TCHAD
TGO	2.68	2.4	TOGO

Tab. 1.1. – *La matrice CODPOP.*

BEN	MI	23	0
BWA	PP	29	-6
HVO	MI	41	-15
BDI	MI	23	-8
DJI	NC	3	.
ETH	MI	485	-26
GMB	PP	.	-6
GIN	PU	44	-16
GNB	MI	.	2
GNQ	MI	5	.
LSO	DE	.	7
MWI	DE	22	-4
MLI	MI	46	-15
NER	MI	16	-6
UGA	PR	852	-20
CAF	PP	12	-4
TZA	PU	285	-13
RWA	MI	18	-5
SLE	PR	19	-8
SOM	MI	150	-8
SDN	PU	470	1
TCD	MI	62	-24
TGO	MI	22	-8

Tab. 1.3. – *La matrice REGIME.*

fournit des renseignements relatifs à l'endettement des 23 pays les moins avancés, c'est-à-dire :
variable
n° 1 : identificateurs des états.
n° 2 : aide extérieure en 1982, publique au développement, en millions de dollars.
n° 3 : aide extérieure publique au développement, en 1982, en millions de dollars, en provenance du Comité d'Aide au développement des pays de l'OCDE (CAD).
n° 4 : aide extérieure publique au développement, en 1982, en millions de dollars, en provenance des pays de l'OPEP.
n° 5 : aide extérieure aux conditions du marché, en millions de dollars.
n° 6 : encours de la dette extérieure en 1982, en millions de dollars.
n° 7 : règlements pour le service de la dette en 1982, en millions de dollars.

La variable n° 1 est discrète, alphanumérique, relevée sur une échelle nominale. Les variables n° 2 à 7 sont continues, numériques et relevées sur une échelle d'intervalles.

c. Matrice REGIME (tab 1.3)

Comprenant variables et 23 observations, la matrice REGIME donne les informations complémentaires suivantes : variable
n° 1 : identificateurs des pays.
n° 2 : régimes politiques en 1982, avec ces modalités :
NC = non connu
MI = militaire
PP = parlementaire pluripartite
PU = parti unique
DE = despotique
PR = parlementaire restreint.
Cette information provient de : Kidron (M.), Smith (D), 1983, Atlas du Monde Armé. Paris, Calmann-Levy, 40 planches.
n° 3 : budget militaire en 1981, en millions de dollars.
n° 4 : déficience (-) ou surplus (+) calorique, en % de la moyenne mondiale de 1980, d'après : Ayrton (P.), 1984, World Views. London, Pluto Press, 218 p.

Notons que certaines données sont restées introuvables ; elles ont été indiquées par un point (.) à la place de la valeur manquante. Les variables n° 1 et 2 sont discrètes, alphanumériques, relevées sur des échelles nominales. La variable n° 3 est continue, numérique, relevée sur une échelle d'intervalles ; la variable n° 4 est continue, numérique, relevée sur une échelle de rapport.

1.4 Des matrices d'information spatiale aux bases de données géographiques

Traiter des matrices d'information spatiale de manière efficace, c'est recourir aux méthodes statistiques d'analyse multivariée, ainsi qu'à la cartographie automatique. La mise en oeuvre de ces techniques passe nécessairement par l'ordinateur. Il faut alors pouvoir enregistrer son information sur un support magnétique reconnaissable par l'ordinateur. Selon le cas, on parle de fichier sur disque (ou disquette) ou de fichier sur bande. En effet, ces deux types de support d'information ont en commun la fonction de stockage des données. Les disques permettent l'accès à n'importe quelle partie du fichier par une rotation,puis un déplacement des têtes de lecture perpendiculaire à l'axe. Les bandes magnétiques ne permettent pas l'accès direct ; pour utiliser une partie d'un fichier, il faut faire défiler toute la partie de la bande qui la précède. Les bandes doivent être réservées au stockage et la communication de l'information, cela dans la grande majorité des cas. A l'exception des étapes de saisie et d'archivage de l'information, on préfèrera toujours les fichiers sur disque.

Une fois constituées, les matrices d'information spatiale doivent être tranférées sur un support magnétique. En général, cette opération est assurée par une société de services informatiques. Les données seront saisies en format « images de cartes » (par référence aux anciennes cartes perforées en papier bristol) ; chaque enregistrement aura un nombre fini de positions, au maximum 80. En cas de besoin, il faudra ajouter un second, un troisième... enregistrement pour saisir toutes les variables. On dit qu'une observation est un enregistrement logique (correspondant à la logique de traitement), et qu'une image de carte est un enregistrement physique (sur un support d'information) ; il peut donc y avoir plusieurs enregistrements physiques pour un seul enregistrement logique. Pour bien s'entendre avec une société de services informatiques, il est nécessaire de lui fournir un dessin d'enregistrement qui permettra, à l'issue de la saisie, de retrouver son information ; on doit préciser à cette société de service la position des différentes variables pour chaque observation, c'est-à-dire la délimitation, dans chaque enregistrement physique, des zones contenant les valeurs par le rang de leur caractère de début et celui de leur caractère de fin. La longueur de chaque zone correspondant à chaque variable doit être supérieure ou égale au nombre de caractères de la valeur la plus longue. Par exemple, les deux matrices CODPOP et DETTES sont stockées sur un disque magnétique. Voici leurs dessins d'enregistrement (fig. 1.4) :

Matrice CODPOP

Nom	Position	Type
variable n° 1	1-3	alphanumérique
n° 2	5-9	numérique
n° 3	12-14	numérique
n° 5	16-34	numérique

Matrice DETTES

variable n° 1	1-3	alphanumérique
n° 2	7-11	numérique
n° 3	15-19	numérique
n° 4	23-27	numérique
n° 5	31-35	numérique
n° 6	38-43	numérique
n° 7	46-50	numérique

Dans le cas, peu courant, où l'usager peut assurer lui-même la saisie, il n'est pas obligatoire d'avoir ces dessins d'enregistrement ; seul l'ordre des variables est nécessaire (mais ces dessins sont toujours utiles).

A l'issue de cette première étape, on dispose d'une bande magnétique ou d'une disquette contenant les données. Pour utiliser cette information, il faut la transférer sur un disque magnétique. Cette opération est simplifiée si on respecte les standards IBM, c'est-à-dire :

a. pour les bandes : le format IBM standard label, une densité d'enregistrement de 1600 bpi (bit per inch), 9 pistes, le codage EBCDIC.

b. pour les disquettes : le format IBM PC, double face, double densité. La conversion de support, c'est-à-dire de bande ou disquette vers disque se fait à l'aide d'un programme utilitaire. Par exemple, le programme IEBGENER sur des configurations IBM réalise très simplement l'opération à partir d'une bande vers un disque. Pour réaliser ce travail, il est très vivement conseillé de s'assurer de la collaboration d'un informaticien.

L'étape de transfert se solde par la création d'un ou plusieurs fichiers sur disque. Ceux-ci peuvent être directement exploités par un programme ad hoc ; mais il est préférable de les regrouper dans une base de données, c'est-à-dire dans un fichier ayant une organisation particulière qui, associé à un langage de gestion des données, en facilitera l'exploitation ultérieure. Il doit être possible, par exemple, de sélectionner des variables dans plusieurs matrices pour les analyser simultanément, d'agréger les observations selon un critère géographique (passer des communes aux départements...), d'exclure les observations pour lesquelles manquent des valeurs, etc... Il y a un grand nombre de systèmes de gestion de bases de données (SGBD) ; citons ADABAS, sgbd hiérarchique, SQL, sgbd relationnel. Le Statistical Analysis System, SAS, quoique très différent des précédents est très bien adapté aux bases de données géographiques.

```
positions d'enregistrement
      1 ------------------------------------
      2 identifiants des états
      3 ------------------------------------
      4
      5 ------------------------------------
      6
      7 population en millions d'habitants
      8
      9 ------------------------------------
     10
     11
     12 ------------------------------------
     13 taux de croissance moyen annuel de
        la population 1970-1982 (%)
     14 ------------------------------------
     15
     16 ------------------------------------
     17
     18
     19
     20
     21
     22
     23 noms des états en clair
     24
     25
     26
     27
     28
     29
     30
     31
     32
     33
     34 ------------------------------------
     35
```

Fig. 1.4. – *Le dessin d'enregistrement de la matrice CODPOP.*

2. Vue d'ensemble du système SAS

Le « Statistical Analysis System » (SAS) ou système d'analyse statistique est apparu sur le marché du progiciel à la fin des années 1970. Initialement conçu pour s'exécuter sur de gros ordinateurs IBM équipés du système d'exploitation OS/MVS et VM/CMS, on le trouve maintenant sur des mini-ordinateurs à 32 bits DEC, PRIME et DATA GENERAL ; enfin, depuis le dernier trimestre 1985, SAS est disponible sur micro-ordinateurs IBM PC ou compatibles.

2.1. Un grand nombre de fonctions

Parmi l'ensemble des modules commercialisés à la fin 1985, on peut distinguer :

a. les modules d'intérêt général :

BASICS assurant la gestion des données ainsi que leur préparation en vue de leur traitement.
STATISTICS proposant toute une panoplie de techniques d'analyse statistique parmi les plus courantes.
IML ou Interactive Matrix Language, un langage de programmation matriciel multipliant les possibilités de traitement scientifique des données.

b. les modules simplifiant l'usage de SAS sur un terminal :

FSP, Full Screen Product, simplifie notamment les travaux de saisie et de personnalisation de documents.
AF s'adresse tout particulièrement aux concepteurs d'applications « clés en main » pour construire des menus proposant des fonctions spécifiques à leurs clients (Application Facility).
DMI permet d'adapter SAS au produit IBM connu sous le nom SPF (Support Program Facility) et ainsi de simplifier le dialogue usager-ordinateur, d'où son nom de Dialog Manager Interface.

c. les modules adaptés à des besoins plus spécifiques :

GRAPH offre une grande variété de représentations graphiques des données allant de simples diagrammes à bâtons aux cartes thématiques.
OR est indispensable à tous ceux qui désirent mettre en œuvre les techniques de la Recherche Opérationnelle et de la Programmation Linéaire.
ETS traite les séries statistiques temporelles utilisées par les économètres (Econometrics and Time Series).

2.2. Pourquoi avoir choisi SAS

SAS est très facile à apprendre : l'expérience a montré qu'en quelques jours de stage, un élève consciencieux peut arriver à réaliser des opérations élaborées qu'il ne lui était pas possible d'envisager précédemment.
SAS est un outil de recherche complet : il assure à la fois les fonctions de gestion, de préparation et de traitement des données. Les étapes DATA bénéficient de la flexibilité d'un langage de programmation propre à manipuler aisément les données. Les étapes PROC provoquent l'exécution de procédures de traitement simples, rapides et proposant une grande variété d'options.
Le langage SAS est uniforme : il s'agit d'un véritable langage de programmation disposant d'un vocabulaire étendu et précis et d'une syntaxe très facile d'emploi. Le programmeur SAS peut ainsi concentrer ses efforts sur les fonctions à réaliser, en s'affranchissant des contraintes habituelles de la rédaction d'un programme.
Le volume des données à traiter peut être très grand : SAS ne connaît comme limites que celles de la configuration du centre informatique de l'usager, c'est-à-dire la région mémoire maximale, le temps maximal d'exécution, la capacité disponible sur support magnétique.

SAS traite des tableaux observations/variables : ils correspondent parfaitement au conditionnement informatique des matrices d'information spatiale.

SAS est conversationnel : à partir de son terminal le chercheur peut entrer son programme, déclencher son exécution et vérifier son bon déroulement, visualiser immédiatement ses résultats. Bien sûr, lorsque le temps de calcul ou la sortie des résultats sont trop grands, SAS peut aussi s'exécuter en traitement par lot : on perd toutefois la faculté de dialoguer avec le système.

SAS est ouvert : les programmeurs professionnels ont la faculté d'écrire leurs propres procédures dans un langage évolué comme FORTRAN ou PL/1. La Maison de la Géographie de Montpellier dispose ainsi de procédures de cartographie interfaçant SAS à la bibliothèque graphique UNIRAS.

SAS est très évolutif : chaque version apporte sont lot de nouveautés intéressantes comblant les lacunes ou complétant les procédures déjà existantes. Connaitre SAS, c'est se donner les moyens de progresser en même temps que l'évolution des techniques.

Les usagers sont écoutés par SAS INSTITUTE : il existe, de par le Monde, de nombreux clubs d'usagers se réunissant annuellement en congrès, SUGI aux Etats-Unis, SEUGI en Europe. C'est là l'occasion d'échanger des savoir-faire, d'entrer en contact avec les dernières nouveautés et, enfin, de proposer des modifications ou des extensions aux chefs de projets de SAS Institute.

2.3. Ce qu'il faut savoir de l'architecture du système

Lorsque l'utilisateur demande l'exécution de SAS, il se met en situation d'entrer son programme, d'en demander son exécution sur des données qu'il aura choisies et, enfin, d'en obtenir des résultats sous forme imprimée ou dans un nouveau tableau de données pouvant, ultérieurement, faire l'objet d'autres traitements. Ces différentes phases sont assurées par des échanges d'information entre le cœur du système, le superviseur, la bibliothèque de programmes et les bases de données.

Au plan technique, SAS est très sophistiqué ; c'est ce qui en fait un outil efficace et facile d'accès. Il est hors de propos de procéder à un examen minutieux de son fonctionnement interne. Quelques informations sur l'architecture du système permettront cependant de mieux comprendre son fonctionnement.

Le superviseur assure de nombreuses tâches parmi lesquelles il faut citer le contrôle de la bonne utilisation du langage de programmation. Les instructions sont analysées tant au plan du lexique qu'à celui de la syntaxe ; seules les instructions correctes pourront être interprétées et exécutées ensuite par l'ordinateur. Le superviseur génère aussi un journal de bord nommé LOG qui contient un grand nombre d'informations sur la réalisation des diverses opérations demandées par l'usager ; on y trouve notamment les temps d'exécution des étapes en unité centrale, le nom et le nombre d'observations des tableaux, les messages d'erreur de programmation ou d'exécution. Enfin, le superviseur admet des options définissant l'environnement du système, par exemple, le type de terminal employé, le nombre maximal d'erreurs admises ou bien encore la localisation du programme SAS à exécuter.

La bibliothèque (Library) renferme deux types de modules assez différents. Les « parsing modules » assurent l'interprétation des instructions composant le programme. Dans le cas de l'étape DATA, ils génèrent un module exécutable réalisant les fonctions demandées. Pour l'étape PROC, ils assurent la communication d'informations aux programmes de traitement. En effet, la seconde partie de la bibliothèque est composée de modules exécutables réalisant un type précis de traitement (par exemple, la régression linéaire) auxquels il faut adresser des ordres leur précisant les données à analyser, les options de traitement choisies, l'éventuel stockage des résultats.

Enfin, les bases de données, propres aux utilisateurs, sont organisées toujours de la même manière. Elles comprennent en premier lieu un répertoire spécifique, initialisé à la création de la base ; il renferme toutes les informations nécessaires à la localisation et l'identification des tableaux la composant, c'est-à-dire leur nom, leur nombre d'observations, leur type (données, catalogue graphique...) et leur adresse. En second lieu, chaque tableau de la base comprend un enregistrement décrivant ses caractéristiques et son contenu, plus précisément le nom et l'identification en clair des variables, leur type (numérique ou caractère) ainsi que leur position. A la suite de ce premier enregistrement descriptif, les observations, composées des variables susdécrites, sont enregistrées séquentiellement.

2.4. Qu'est-ce qu'un programme SAS ?

Contrairement à des progiciels plus anciens, SAS est gouverné par un langage de programmation dont l'acquisition par l'usager peut se faire de manière progressive ; il aura donc nécessairement à sa charge la rédaction de programmes écrits en langage SAS. Un programme est un ensemble d'instructions interprétées en séquence par le système avant toute exécution. Il est composé d'étapes dont le nom détermine la fonction. Une étape DATA définit une séquence de préparation de données, c'est-à-dire d'entrée de l'information dans un tableau d'une base, de gestion et de transformation des données et de sortie sur un support extérieur à la base. Une étape PROC indique au système qu'il doit exécuter un traitement particulier en recourant à la bibliothèque SAS. En simplifiant à l'extrême, on peut dire qu'une étape DATA permet de gérer des données qui seront ensuite analysées dans une étape PROC. Notons que DATA et PROC peuvent apparaître dans n'importe quel ordre dans un programme, que le nombre d'étapes n'est pas limité et que chacune d'elles se termine par l'instruction

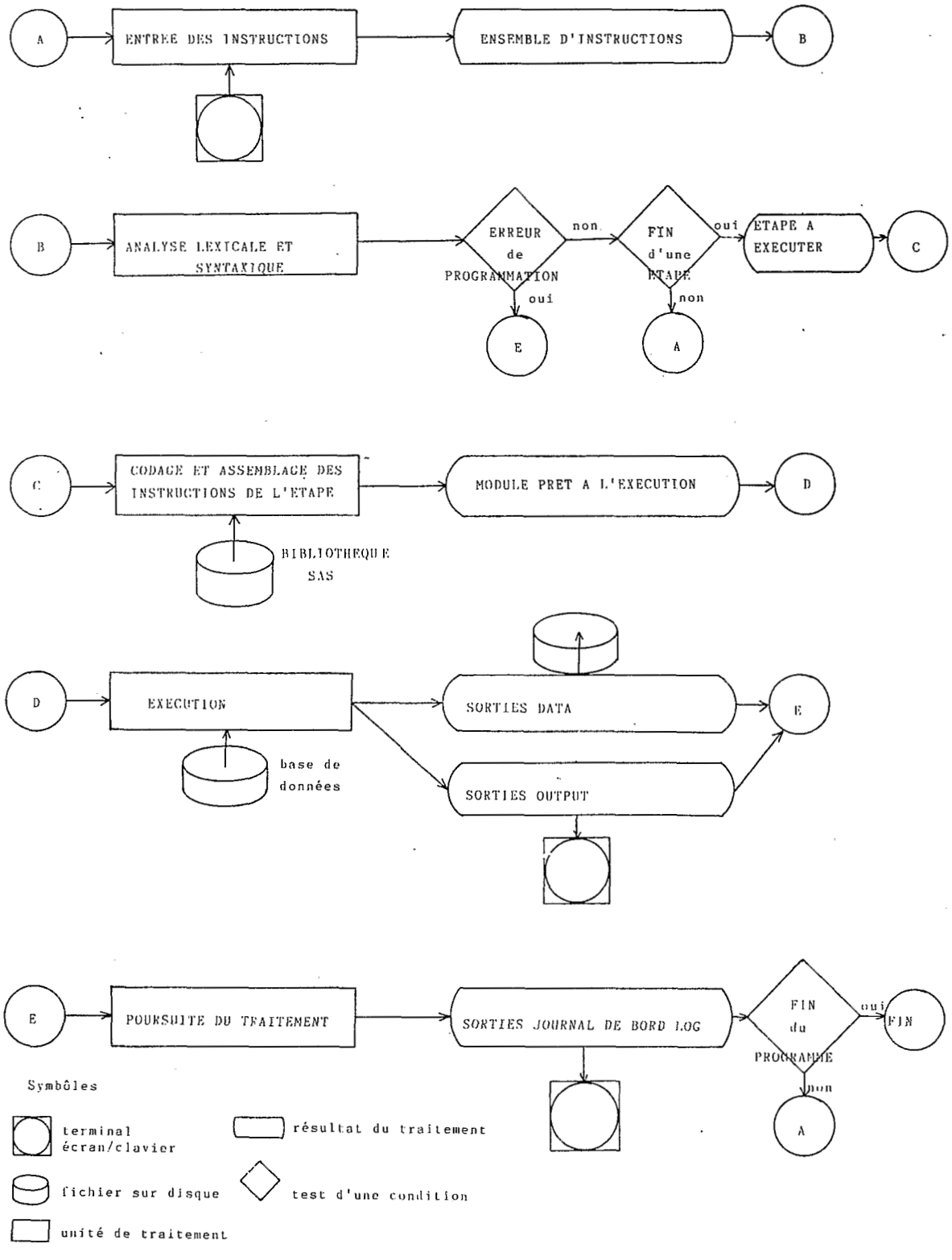


Fig. 2.1. – *Le déroulement d'une étape à partir d'un terminal. écran/clavier.*

RUN. Le langage SAS est composé d'instructions propres à l'étape DATA ou à l'étape PROC et d'instructions communes aux deux.

Une instruction SAS est une phrase qui peut être composée de mots clés, de noms et d'opérateurs ; elle s'achève toujours par un point-virgule (;). Chaque instruction commence par un mot clé qui en détermine sa fonction. Par exemple, SET permet d'aller chercher le contenu d'un tableau dans une étape DATA ; MODEL offre la possibilité de spécifier un modèle dans une procédure de régression. Les noms SAS désignent des tableaux, des variables, des procédures ou des options. Ils comprennent un à huit caractères alphanumériques (lettre ou chiffre) et ne peuvent commencer que par une lettre (A...Z) ou le caractère souligné (_). Les noms de tableaux doivent respecter des contraintes particulières indiquant au système s'il s'agit d'un tableau temporaire (effacé à la fin du travail) ou permanent (conservé pour un usage ultérieur). Ces noms sont composés de deux parties reliées entre elles par un point. La première partie est le nom logique de la base ; ce sera WORK pour un tableau temporaire (de travail), et tout autre nom SAS valide pour un tableau permanent. La seconde partie est le nom du tableau à proprement parler. Prenons, par exemple un tableau contenant les dettes des PMA, le tableau DETTES :

WORK.DETTES est le nom d'un tableau temporaire,

BASE.DETTES est le nom d'un tableau permanent de nom logique BASE. Il est possible de désigner les tableaux temporaires par un nom à une seule partie : DETTES sera compris comme WORK.DETTES. Les opérateurs sont des caractères spéciaux (autres que lettres ou chiffres) reliant des noms ou des mots clés en leur indiquant l'opération à réaliser. On trouve le plus souvent :

= pour l'affectation
+ − pour l'addition et la soustraction
* / pour la multiplication et la division.

2.5. *Comment se déroule une étape ?*

Il existe d'importantes différences entre les étapes DATA et PROC ; pour mieux comprendre le fonctionnement de SAS, il est cependant utile de bien connaître le schéma de la figure 2.1 ; il représente l'exécution d'un programme à partir d'un terminal écran/clavier. On peut découper le déroulement d'une étape en cinq unités de traitement :

a. Le programmeur entre un programme à partir de son clavier ; le texte, sous forme d'un ensemble d'instructions s'affiche alors sur l'écran.

b. SAS procède alors à l'analyse lexicale et syntaxique du programme. Les éventuelles erreurs de programmation provoquent un débranchement vers l'unité de traitement pour impression des messages d'erreur sur le fichier LOG (ou journal de bord). Dans le programme est recherchée la première étape à exécuter, c'est-à-dire l'instruction RUN ; ou bien une nouvelle définition d'étape (DATA ou PROC). Si aucune étape n'est terminée, un autre débranchement vers l'unité de traitement « a » conduit le programmeur à l'achèvement de l'entrée de l'étape en cours ; sinon, une étape prête à l'exécution est soumise à l'unité de traitement « c ».

c. Le codage et l'assemblage de l'étape précédemment programmée font appel à la bibliothèque SAS, c'est-à-dire aux parsing modules assurant la traduction des instructions et aux load modules spécifiques aux procédures. Un module prêt à l'exécution est alors produit.

d. Pour que cette étape s'exécute, il faut le plus souvent qu'elle fasse appel aux données que l'utilisateur a préalablement stockées dans sa base. Le résultat de l'exécution peut être soit un nouveau tableau dans une base, soit une sortie de procédure à l'écran du terminal, soit les deux

e. L'exécution s'achève par l'impression du journal de bord (LOG) contenant toutes les informations sur le temps en unité centrale, sur la taille des tableaux créés, sur les erreurs qui ont pu être détectées. Il est alors possible de retourner à l'unité a pour enchaîner avec une nouvelle étape ou bien achever l'exécution de SAS.

Pour rendre aussi conversationnelle que possible la saisie et l'exécution d'un programme ainsi que la visualisation des résultats au terminal, la version 5 propose une nouveauté très attrayante : le DISPLAY MANAGER ou gestionnaire d'affichage qu'il est nécessaire de connaître avant d'apprendre à programmer en SAS.

3. Comment utiliser SAS ?

Aujourd'hui, plusieurs centres informatiques de recherche français proposent SAS à leurs usagers ; citons par exemple le Centre Inter Régional de Calcul Electronique (C.I.R.C.E.) du C.N.R.S. à Orsay, le Centre National Universitaire Sud de Calcul du Ministère de l'Education Nationale (C.N.U.S.C.) de Montpellier ou bien encore le centre de calcul de l'Ecole Normale Supérieure de Paris (Ulm). De plus, de nombreuses sociétés privées utilisent SAS à des fins de recherche appliquée. Bien que cet exposé s'appuie sur la configuration du C.N.U.S.C., il est aussi valable, dans ses grandes lignes pour le C.I.R.C.E..

3.1. Communiquer avec un ordinateur

Le C.N.U.S.C. de Montpellier est doté de deux systèmes informatiques IBM mettant en oeuvre deux systèmes d'exploitation assez différents : OS/MVS d'une part, VM/CMS d'autre part. Ils assurent l'ensemble des tâches de gestion des ressoures nécessaires au bon fonctionnement des ordinateurs. Bien que SAS puisse fonctionner sous ces deux systèmes d'exploitation, OS/MVS aussi implanté au C.I.R.C.E. a été préféré pour la rédaction du présent document. Le lecteur pourra ainsi passer assez aisément d'un centre informatique à l'autre.

Très schématiquement, il y a deux manières de travailler avec un ordinateur. La première, la plus ancienne est le traitement par lot (en anglais Batch Processing) : l'usager doit entièrement définir un travail (Job), c'est-à-dire préciser les ressources qui lui sont nécessaires (taille mémoire, temps de calcul...), puis déterminer l'ensemble des traitements à assurer sur des données convenablement enregistrées. Un tel travail est ensuite soumis d'un bloc pour son exécution, sans qu'il soit possible d'y modifier quoi que ce soit. S'il est correct, il s'exécutera jusqu'au bout et produira les résultats des traitements demandés ; dans le cas contraire, s'il contient des erreurs, il s'arrêtera et des messages d'erreur seront imprimés. Dans ce dernier cas, il faudra recommencer la soumission après correction des erreurs.

La seconde manière de travailler est de dialoguer avec le système en lui adressant des commandes à partir d'un terminal écran/clavier relié directement au site central par une ligne de communication. On parle alors de temps partagé puisque plusieurs terminaux peuvent travailler en même temps, se partageant ainsi les ressources et le temps réel d'exécution. Sous OS/MVS, le sous-système de temps partagé s'appelle TSO (Time Sharing Option). Cette interactivité offre une très grande souplesse d'usage car il n'est pas nécessaire de définir totalement un travail ; il est ainsi aisé d'adapter le choix des traitements à réaliser en fonction des besoins du moment, notamment en rapport avec les derniers résultats obtenus. SAS admet aussi bien le traitement par lot que le temps partagé ; ce dernier semble bien être la technique appelée à se développer sur tous les ordinateurs. De plus, le mode de traitement interactif est généralisé sur tous les micro-ordinateurs. C'est pourquoi nous n'envisagerons ici que le mode de traitement interactif, sous TSO.

3.2. Ouvrir une session TSO

Pour entrer dans une session TSO, il faut avoir accès à un terminal écran/clavier relié au site central. La figure 3.1 représente un terminal de la gamme IBM3270, le terminal IBM3279 avec sa silhouette bien connue dans de nombreux centres informatiques. L'écran peut afficher, en sept couleurs, 24 lignes de 80 caractères de longueur ; il admet le « mode pleine page » et un modèle a de plus des possibilités graphiques. Le mode pleine page présente des zones d'affichage ou d'entrée des données simplifiant les dialogues usager/ordinateur. Le clavier (fig. 3.2) est très complet. Il est divisé en quatre parties principales : la plus grande constitue le clavier alphanumérique

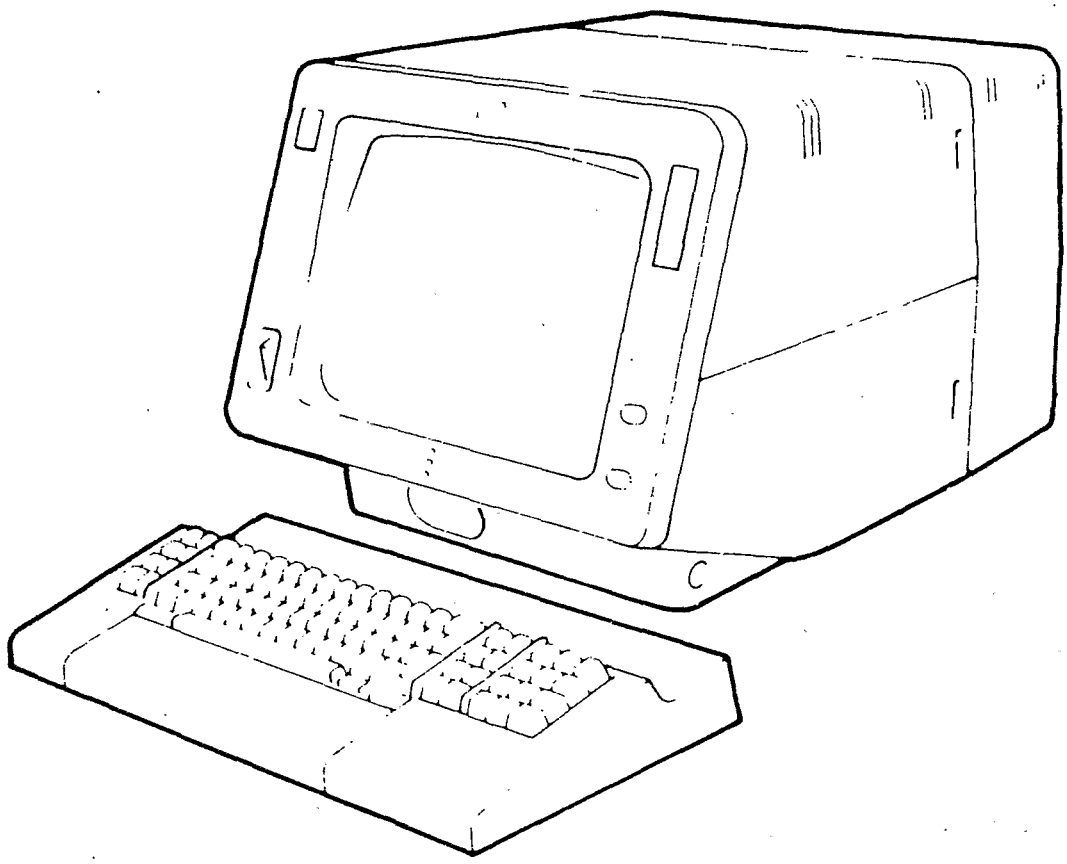


Fig. 3.1. – *Le terminal IBM 3279.*

(lettres et chiffres) habituel sur les machines à écrire ; seules quatre touches spécifiques permettent d'une part de faire déplacer le curseur (la position courante du chariot sur une machine à écrire) d'une zone à l'autre, à droite et à gauche ou sur la ligne suivante et, d'autre part de recommencer en cas d'erreur de manipulation et de blocage du curseur (touche Reset) ; enfin, la touche « Entrée » communique au système le contenu des zones saisies. A droite du clavier alphanumérique figure le clavier d'édition facilitant le déplacement du curseur sur l'écran, à droite, à gauche, au dessus ou au dessous ; il est aussi possible d'insérer un ou plusieurs caractères à partir de la position courante du curseur ou d'en supprimer un. A l'extrémité droite du clavier se trouve le pavé des 12 touches de fonction évitant d'écrire les commandes in-extenso ; elles représentent des commandes particulières selon le type de traitement en cours. La signification de ces touches sera indiquée par la suite lorsque cela sera nécessaire. Enfin, à l'extrémité gauche figurent les touches de contrôle de session ; seules nous intéressent ici la touche Attn (Attention) interrompant sur le champ un traitement en cours et la touche de recopie d'écran sur imprimante afin de conserver sur papier le contenu d'un écran.

Si cela n'est déjà fait, il faut connecter le terminal en mettant l'interrupteur rouge à gauche de l'écran en position I (In, en ligne). Après quelques secondes, le dialogue de début de session peut commencer.

Questions (en rouge)	Réponses (en vert)
Nom de l'application choisie	TSO
Identificateur de session	STAGE
Code comptable	ORS1234
Mot de passe associé	GEO
Autre paramètre à mot clé	REGION = 2000k

Chaque réponse doit être validée par la touche Entrée. En cas d'erreur, il suffit de répondre à nouveau à la question ayant eu une réponse erronée. Cette première phase d'ouverture de session s'achève par l'affichage de messages d'information (en rouge) se terminant par READY. Il est désormais possible d'adresser une commande à TSO. Il suffit de taper la commande SAS5 pour entrer dans le Statistical Analysis System, version 5 (la dernière en date).

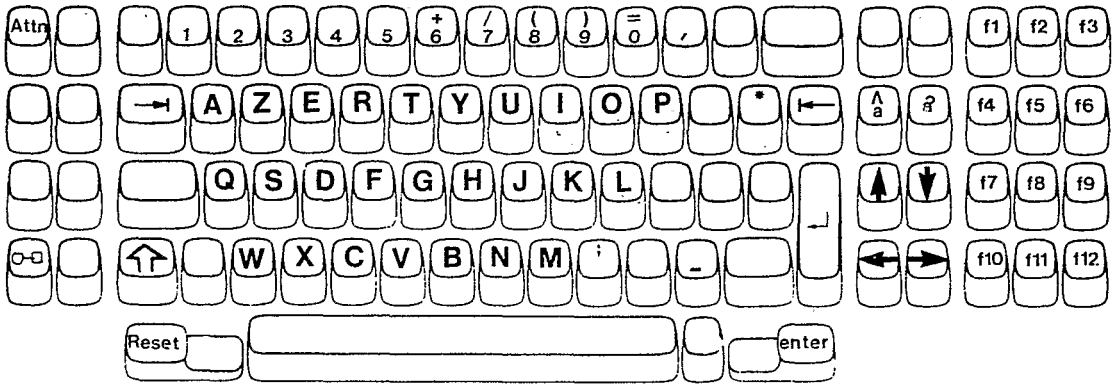


Fig. 3.2. – *Le clavier (simplifié) du terminal IBM 3279.*

3.3. Dialoguer avec SAS : le Display Manager

L'appel de SAS5 se solde par l'affichage d'un écran par le Display Manager. Il s'agit d'une aide au dialogue visant à simplifier l'entrée d'un programme SAS, la visualisation des résultats et l'examen du journal de bord. Le Display Manager se compose de trois écrans :

a. L'écran éditeur de programme (program editor screen) offre toutes les possibilités d'un éditeur de texte, c'est-à-dire l'entrée des instructions, leur visualisation et leur correction, l'insertion de nouvelles instructions, la sauvegarde du programme dans un fichier.

b. L'écran journal de bord (log screen) renferme tout ce qui concerne l'interprétation et l'exécution du programme enregistré précédemment, notamment le nom des tableaux créés et les temps d'exécution. La première image transmise par SAS est un écran en deux parties : au dessus, le journal de bord et, au dessous, l'éditeur de programme (fig. 3.3).

c. L'écran de sortie des résultats (output screen) facilite la consultation des résultats des traitements. Il est possible de lire directement la partie souhaitée à l'aide de touches de fonction.

3.3.1. Les commandes de l'éditeur de programme

Cet écran est composé de deux parties : une première ligne de commande d'une part, l'ensemble des lignes à enregistrer d'autre part. La ligne de commande permet d'entrer des commandes agissant sur l'ensemble du programme. Voici les plus importantes :

BOTTOM ou BOT affiche la dernière ligne du programme alors que TOP affiche la première.

CHANGE ou C modifie la première occurrence, à partir de la première ligne affichée sur l'écran, d'une chaîne de caractères par une autre chaîne. Elle s'écrit :

C « chaîne à modifier » « chaîne modifiée »

La touche de fonction n° 6 correspond à RCHANGE pour modifier dans les mêmes termes la prochaine occurrence.

FIND ou F localise et affiche la première occurrence, à partir de la première ligne figurant sur l'écran, d'une chaîne de caractères :

F « chaîne à rechercher »

La touche de fonction n° 5 correspondant à RFIND assure l'itération de cette recherche jusqu'à la prochaine occurrence.

INCLUDE introduit un programme contenu dans un fichier dans l'écran éditeur de programme. Il est alors aisé de modifier ou d'exécuter à nouveau ce programme. Pour trouver ce fichier, il est indispensable d'en donner son nom logique :

INCLUDE nom logique du fichier

SAVE réalise l'opération inverse, c'est-à-dire la sauvegarde du programme édité dans un fichier :

SAVE nom logique du fichier

SUBMIT soumet à SAS le programme qui, s'il est correct, est immédiatement exécuté. La touche de fonction n° 3 correspond à cette commande.

RECALL fait apparaître le dernier programme soumis pour le modifier ou le corriger. La touche de fonction n° 4 réalise aussi cette opération. Notons qu'en frappant deux fois la touche de fonction n° 4, ce sont les deux derniers programmes soumis qui sont rappelés et ainsi de suite.

X suivi d'une commande TSO transmet cette dernière directement au sous-système TSO.

Enfin, les touches de fonction n° 7 et n° 8 permettent respectivement l'affichage de la partie suivante et précédente du programme.

```
COMMAND ===>                                                  SAS(R) LOG  18:04

NOTE: COPYRIGHT (C) 1984 SAS INSTITUTE INC., CARY, N.C.  27511, U.S.A.
NOTE: SAS RELEASE 5.08 AT SAS INSTITUTE S.A. - TRIAL INSTALLATION (02816001).

 NEWS: - DRIVER VARIAN DISPONIBLE SOUS TSO
       - GRAPHIQUES SUR IMPRIMANTE COULEUR IBM3287 POSSIBLES SOUS TSO
       POUR EN SAVOIR PLUS FAIRE: HELP NEWS;

--------------------------------------------------------------------------------
COMMAND ===>                                                  PROGRAM EDITOR

00001
00002
00003
00004
00005
00006
00007
00008
```

Fig. 3.3. – *Le display manager après l'appel de SAS.*

```
COMMAND ===>                                                  SAS(R) LOG  18:12

NOTE: COPYRIGHT (C) 1984 SAS INSTITUTE INC., CARY, N.C.  27511, U.S.A.
NOTE: SAS RELEASE 5.08 AT SAS INSTITUTE S.A. - TRIAL INSTALLATION (02816001).

 NEWS: - DRIVER VARIAN DISPONIBLE SOUS TSO
       - GRAPHIQUES SUR IMPRIMANTE COULEUR IBM3287 POSSIBLES SOUS TSO
       POUR EN SAVOIR PLUS FAIRE: HELP NEWS;

--------------------------------------------------------------------------------
COMMAND ===>                                                  PROGRAM EDITOR

00001  /*-------------------------------------------------*/
00002  /* EXEMPLE D'UTILISATION DU DISPLAY MANAGER DE SAS */
00003  /*-------------------------------------------------*/
00004
00005  /*-----> ALLOCATION DE LA BASE */
00006
00007 X ALLOC DA(.STAGBASE) FI(BASE) SHR;
00008
```

Fig. 3.4. – *Le display manager après correction du programmme contenu dans le fichier .STAGPROG(EXPL1).*

Les lignes de programme à enregistrer sont précédées par des numéros sur lesquels peuvent être entrées des commandes de ligne s'adressant non pas à l'ensemble du programme, mais seulement aux lignes concernées. Ne figurent ci-après que les commandes de ligne les plus importantes.

In insère (input) n lignes à partir de celle où est entrée la commande ; ces lignes sont vierges et peuvent recevoir de nouvelles instructions de programme. n peut être omis ; dans ce cas, une seule ligne est insérée.

Dn supprime (delete) n lignes ; si n est omis, seule la ligne où est entrée cette commande est supprimée.

Cn copie (copy) n lignes. Il faut indiquer où doit arriver cette copie dans le programme en entrant soit A (after) – les lignes seront copiées après cette ligne –, soit B (before) – les lignes seront copiées avant. Si n est omis, seule la ligne où est entrée C est copiée. Il est aussi très aisé de copier un groupe de lignes en le délimitant par CC à son début, CC à sa fin et A ou B comme précédemment.

Mn déplace (move) n lignes. Il faut également indiquer où doivent être transférées ces lignes en entrant soit A – les lignes seront alors transférées après cette ligne –, soit B – les lignes seront copiées avant. Si n est omis, seule la ligne où est entrée la commande M est déplacée. Un groupe de lignes à déplacer peut être délimité en entrant MM à son début, MM à sa fin et A ou B comme précédemment.

3.3.2. Exercice d'utilisation de l'écran éditeur de programme

Afin de familiariser le lecteur avec l'éditeur de programme, l'exercice suivant lui est proposé. Un programme stocké dans un fichier où les instructions figurent avec des erreurs, dans un ordre erroné a été préalablement préparé. Le fichier se nomme .STAGPROG(EXPL1). Il faut :

a. récupérer ce programme dans l'éditeur,
b. corriger le texte,
c. remettre les instructions dans le bon ordre de séquence,
d. exécuter le programme.

a. récupérer ce programme dans l'éditeur, c'est d'abord identifier le fichier le contenant par son nom logique à l'aide de la commande ALLOCATE de TSO (les fonctions précises de cette commande seront étudiées plus loin) ; il faut donc entrer cette commande sur la ligne de commande qui est, rappelons-le, la première de l'éditeur :

```
X ALLOC DA(.STAGPROG(EXPL1)) FI(PROG)
SHR
```

Après avoir validé cette commande par la touche Entrée, le curseur se positionne à nouveau en début de ligne ; la commande INCLUDE permet de récupérer le programme :

```
INCLUDE PROG
```

Le programme erroné s'affiche maintenant dans l'éditeur (fig 3.6). Il va falloir le corriger. Les lignes entre caractères /* et */ sont des commentaires.

b. le texte correct de l'exercice est présenté en figure 3.5. La fonction de chaque étape est indiquée en clair dans le programme ; le détail des instructions fera l'objet des chapitres suivants. Pour l'heure, il s'agit de reconstituer le programme correct à partir du programme erroné (fig. 3.6) qui vient d'être inclus dans l'éditeur de programme. Ce travail doit être réalisé en deux étapes ; tout d'abord, remplacer les points d'interrogation par les bonnes chaînes de caractères.

c. une fois la précédente opération réalisée, il faut réorganiser les diverses étapes afin d'obtenir la bonne séquence d'étapes.

d. il est maintenant possible de soumettre ce programme à l'aide de la touche de fonction n° 3 ; en cas d'erreur il faut consulter le journal de bord puis rappeler le programme dans l'éditeur par la touche de fonction n° 4 et procéder aux corrections nécessaires avant de soumettre à nouveau.

3.3.3. Visualisation des résultats : l'écran OUTPUT

Lorsqu'il est nécessaire d'afficher des résultats, SAS efface l'écran programme/journal de bord et le remplace par l'écran OUTPUT (fig. 3.7). Quelques touches de fonctions facilitent la consultation :

F7 défilement arrière pour voir ce qui précède,
F8 défilement avant pour regarder la suite,
F11 partie gauche,
F12 partie droite. Ces deux dernières touches sont nécessaires lorsque la sortie est au format listing, sur 132 caractères ; l'écran est alors une fenêtre pouvant être déplacée sur la page,
F3 affichage direct de la fin de la sortie. Une seconde pression sur F3 fait quitter définitivement l'écran OUTPUT. A la suite de deux F3, SAS affiche à nouveau l'image en deux parties composée à sa partie supérieure de l'écran journal de bord et, à sa partie inférieure de l'écran éditeur de programme (fig. 3.8).

3.3.4. Consultation du journal de bord : l'écran LOG

De même que pour l'écran éditeur de programme, l'écran LOG est composé de deux parties : la première ligne de commande facilite grandement la consultation ; les lignes suivantes contiennent le texte du journal. Les commandes les plus couramment utilisées sont TOP, BOTTOM et FIND ; elles ont le même rôle que pour l'éditeur de programme. Les touches de fonction n° 7 et n° 8 permettent respectivement de visualiser ce qui précède ou ce qui suit la page en cours d'affichage.

Les écrans éditeur de programme et journal de bord figurant sur la même image, il est nécessaire pour leur adresser des commandes de positionner le curseur dans leur propre ligne de commande à l'aide d'une des

touches de tabulation. En guise d'exercice, il est intéressant de consulter le journal de bord (s'affichant en vert) du programme EXPL1 précédemment soumis ; remarquons que les tableaux (DATASETS) créés sont indiqués avec leur nombre de variables et d'observations. Le temps d'exécution (CPU TIME) est très modeste.

```
/*-----------------------------------------------*/
/* EXEMPLE D'UTILISATION DU DISPLAY MANAGER DE SAS */
/*-----------------------------------------------*/

/*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) SHR;

/*-----> TRIS DES TABLEAUX ET SELECTION DES VARIABLES */

PROC SORT DATA=BASE.CODPOP OUT=CODPOP;BY CODE;
    PROC SORT DATA=BASE.DETTES OUT=DETTES;BY CODE;
        PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;

/*-----> ASSOCIATION DES 3 TABLEAUX */

DATA TOTAL;MERGE CODPOP DETTES REGIME;
    KEEP CODE NOM POP82 ENCOUR82 REGPOL82;

/*-----> CALCUL DETTE ET BUDGET MILITAIRE/HABITANT */

DATA TOTAL;SET TOTAL;
     DETTE=ENCOUR82/POP82;
          LABEL DETTE=DETTE EN $ PAR HABITANT 1982;

/*-----> AFFICHAGE DU TABLEAU */

 PROC PRINT DATA=TOTAL UNIFORM;ID NOM;
      TITLE 'ENDETTEMENT ET REGIMES POLITIQUES';

/*----> TRACES DE DIAGRAMMES A BATONS */

PROC CHART DATA=TOTAL;
     HBAR DETTE/LEVELS=5;
     VBAR REGPOL82/DISCRETE;
RUN;
```

Fig. 3.5. – *Le texte correct du programme EXPL1.*

```
/*-----------------------------------------------*/
/* EXEMPLE D'UTILISATION DU DISPLAY MANAGER DE SAS */
/*-----------------------------------------------*/

/*-----> ASSOCIATION DES 3 TABLEAUX */

DATA TOTAL;MERGE ?????????? DETTES REGIME;
    KEEP CODE NOM POP82 ENCOUR82 REGPOL82;

/*-----> TRIS DES TABLEAUX ET SELECTION DES VARIABLES */

PROC SORT DATA=BASE.CODPOP OUT=CODPOP;BY ??????????;
    PROC SORT DATA=BASE.DETTES OUT=DETTES;BY CODE;
        PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;

/*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(??????????) SHR;

/*-----> AFFICHAGE DU TABLEAU */

 PROC PRINT DATA=TOTAL UNIFORM;ID NOM;
      TITLE '?????????????????????????????????';

/*-----> CALCUL DETTE ET BUDGET MILITAIRE/HABITANT */

DATA TOTAL;SET TOTAL;
     DETTE=ENCOUR82/POP82;
          LABEL DETTE=DETTE EN $ PAR HABITANT 1982;

/*----> TRACES DE DIAGRAMMES A BATONS */

PROC CHART DATA=TOTAL;
     ?????????? DETTE/LEVELS=5;
     ?????????? REGPOL82/DISCRETE;
RUN;
```

Fig. 3.6. – *Le texte erroné du programme EXPL1.*

```
                    ENDETTEMENT ET REGIMES POLITIQUES

                          FREQUENCY BAR CHART
MIDPOINT
 DETTE   DETTE EN $ PAR HABITANT 1922        FREQ  CUM.  PERCENT   CUM.
                                                   FREQ           PERCENT
         |
    0    |*************************           5     5    21.74    21.74
         |
   80    |***********************************  7    12    30.43    52.17
         |
  160    |******************************      6    18    26.09    78.26
         |
  240    |***************                     3    21    13.04    91.30
         |
  320    |**********                          2    23     8.70   100.00
         |
         ----+----+----+----+----+----+----+
             1    2    3    4    5    6    7

                        FREQUENCY
```

Fig. 3.7. – *Une partie de l'output du programme EXPL1.*

```
COMMAND ===>                                            SAS(R) LOG  13:23

  5  /*-----> ALLOCATION DE LA BASE */
  6
  7 X ALLOC DA(.STAGBASE) FI(BASE) SHR;
  8
  9  /*-----> TRIS DES TABLEAUX ET SELECTION DES VARIABLES */
 10
 11 PROC SORT DATA=BASE.CODPOP OUT=CODPOP;BY CODE;
 12
NOTE: 1 CYLINDER DYNAMICALLY ALLOCATED ON SYSDA FOR EACH OF 3 SORT WORK DATA
SETS
NOTE: DATA SET WORK.CODPOP HAS 23 OBSERVATIONS AND 4 VARIABLES. 1116 OBS/TRK.

 12       PROC SORT DATA=BASE.DETTES OUT=DETTES;BY CODE;
 13
NOTE: DATA SET WORK.DETTES HAS 23 OBSERVATIONS AND 7 VARIABLES. 852 OBS/TRK.

 13             PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;
 14
 15  /*-----> ASSOCIATION DES 3 TABLEAUX */
--------------------------------------------------------------------------
COMMAND ===>                                              PROGRAM EDITOR

00001
00002
00003
00004
00005
00006
00007
00008
```

Fig. 3.8. – *Une partie de la SASLOG du programme EXPL1.*

3.4. Clore une session SAS et une session TSO

Après avoir réalisé ses traitements, l'usager doit quitter son terminal en deux opérations. La première consiste à clore la session SAS. Ceci se fait très simplement en entrant la commande ENDSAS sur la ligne de commande de l'éditeur de programme ; un message s'affiche alors sur l'écran journal de bord donnant l'adresse de SAS Institute en Allemagne Fédérale. L'écran s'efface ensuite pour laisser place au message READY. Il faut maintenant fermer la session TSO par la commande LOGOFF. A la suite apparaît la facture détaillant le coût des diverses ressources mises en oeuvre depuis l'ouverture de la session TSO ainsi que le prix total.

4. Créer un tableau SAS à partir de données extérieures à la base

Créer un tableau SAS revient soit à créer une base SAS en même temps que son premier tableau, soit à ajouter un nouveau tableau à une base existant déjà. Dans ces deux cas, le programme SAS sera le même ; seule la commande de tso ALLOCATE sera différente.

4.1. Allouer une base et nommer les tableaux SAS

Une base SAS est un fichier sur support magnétique, le plus souvent sur disque. Son organisation physique est totalement transparente pour l'utilisateur qui n'a qu'à se préoccuper d'identifier correctement son information (noms de tableaux, noms et types des variables, etc...). Seules deux opérations d'ordre plus informatique restent à la charge de l'usager.

4.1.1. Définir les caractéristiques physiques de la base

Par caractéristiques physiques, il faut entendre le nom de la base, sa taille, sa disposition.

a. Le nom de la base suit les règles en vigueur au CNUSC, c'est-à-dire un nom en trois parties reliées entre elles par des points (.), composé du code comptable (par exemple ORS1234), du nom d'utilisateur (ici STAGE) et du nom de la base à proprement parler (1 à 8 caractères alphanumériques au plus, commençant par une lettre). Par exemple, 'ORS1234.STAGE.STAGBASE' est un nom correct pour TSO ; notons que, pour être compris par le système, il doit être encadré d'apostrophes (on dira par la suite qu'il est « entre quotes »). Lors du dialogue d'ouverture de la session TSO, les deux premières parties ont déjà été indiquées ; il n'est donc pas indispensable, si la base appartient en propre à l'usager de TSO de donner ce nom in extenso. Seule la dernière des trois parties, le nom de la base à proprement parler, reste indispensable. Il doit être précédé par un point (.). Par exemple, .STAGBASE est un nom correct pour TSO, si l'identificateur de session est STAGE et le code comptable ORS1234. Par contre, si la base appartient à un autre identificateur de session, elle devra être nommée par son nom composé.

b. La taille de la base dépend du volume de données statistiques qu'elle doit contenir. Il est assez difficile de calculer précisément la taille des tableaux composant la base. Le manuel de référence BASICS donne les modalités de ce calcul. En pratique, à l'exception de très grosses bases, on demande une taille suffisante pour les tableaux initiaux ainsi que pour ceux des tableaux résultant de calculs, lorsqu'il est nécessaire de les conserver. La taille est exprimée en pistes de disque (TRACKS). Les matrices d'information spatiale présentées dans le premier chapitre (CODPOP, DETTES, REGIME) n'occupent qu'une seule piste chacune.

c. La disposition de la base indique au système s'il faut créer une nouvelle base ; dans ce cas, la disposition sera NEW. Par contre, si la base existe déjà, deux dispositions sont possibles. En premier lieu, si la base appartient à la session TSO, la disposition pourra être OLD ; il sera donc possible de modifier les tableaux, d'en ajouter ou d'en supprimer. En second lieu, si la base est la propriété d'un autre identificateur de session, elle sera nécessairement SHR (sharable, partageable), ne pouvant être modifiée en quoi que ce soit, mais accessible à plusieurs utilisateurs simultanés. Une base appartenant à la session TSO peut aussi être SHR, son propriétaire s'interdisant ainsi de la modifier mais laissant à d'autres la possibilité de s'en servir.

4.1.2. Etablir la relation nom physique/nom logique

Le nom physique de la base est contenu dans le catalogue des fichiers sur disque du système. Il n'est

pas possible à SAS de reconnaître directement une base par son nom physique ; il ne connaît que des noms symboliques, en quelque sorte des surnoms, plus généraux. Allouer une base, c'est aussi établir la relation nom physique du fichier/nom logique. Ce dernier servira à désigner les tableaux composant la base. Comme tous les autres noms par la suite, un nom logique comprend 1 à 8 caractères alphanumériques au plus et commence obligatoirement par une lettre.

4.1.3. Syntaxe de la commande ALLOCATE

Cette commande est composée de trois parties : le nom de la commande, la relation entre noms de fichiers, les caractéristiques physiques.

ALLOCATE ou ALLOC est le nom de la commande.

DATASET(nom physique) ou DA(nom physique) donne le nom du fichier sur disque selon les modalités décrites en 4.1.1.a .

FILE(nom logique) ou FI(nom logique) permet de choisir le nom logique par lequel sera désignée la base.

NEW, OLD ou SHR caractérise la disposition de la base.

SPACE(taille) ou SP(taille) donne la taille exprimée en pistes d'un disque magnétique si la disposition est NEW.

TRACKS indique que la taille est exprimée en pistes si la disposition est NEW. Chaque paramètre de la commande est séparé des autres par au moins un espace.

ALLOC DA(.STAGBASE) FI(BASE) NEW SP(10) TRACKS

définit une nouvelle base de 10 pistes, de nom logique BASE et appartenant à la session TSO sous le nom .STAGBASE .

ALLOC DA('ZZZ1357.PIERRE.BASVIL') FI(BAS2) SHR

définit une base ancienne appartenant à un autre utilisateur que celui de la session TSO, sous le nom logique BAS2.

4.1.4. Compléments sur les fichiers

La commande ALLOCATE peut être entrée à tout moment sur la ligne de commande de l'éditeur de programme précédée de X (voir 3.3) ; elle peut aussi figurer dans le programme de la même manière, mais doit alors s'achever par un point-virgule (;). Plusieurs erreurs peuvent se produire.

a. Si un fichier existe déjà alors que la disposition est NEW, la commande est ignorée et un message d'erreur (en anglais) en donne la raison.

b. Si un fichier n'existe pas alors que la disposition est OLD ou SHR, le résultat est le même que précédemment.

c. L'effet de la commande ALLOCATE étant permanent au cours d'une session TSO, deux fichiers ne peuvent partager le même nom logique ; si deux commandes ALLOCATE successives requièrent le même nom logique, le système demande s'il faut ignorer la première (dans ce cas, c'est le second fichier qui sera désigné par le nom logique choisi) ou bien la seconde.

d. Enfin, la taille d'un fichier ne peut dépasser la capacité totale maximale disponible sur l'un des disques au moment de l'entrée de la commande. Dans le cas contraire, il faut reporter la création de la base à un autre moment ou bien, lorsque cela est possible, recourir aux bandes magnétiques. Nous n'envisagerons pas ici ce cas de figure.

La destruction d'une base SAS est extrêmement simple. Il suffit d'entrer la commande DELETE ou DEL suivie du nom du fichier et précédée de X sur la ligne de commande de l'éditeur de programme.

DEL .STAGBASE

détruit la base de nom STAGBASE appartenant à la session TSO en cours.

DEL 'ZZZ1357.PIERRE.BASVIL'

ne sera acceptée que si l'identificateur de session est PIERRE et le code comptable ZZZ1357 ; dans le cas contraire, la commande contraire sera ignorée et un mesage d'erreur apparaîtra à l'écran. Notons que la commande DELETE est irréversible et qu'il faut donc veiller à l'utiliser avec la plus grande prudence.

4.1.5. Nommer les tableaux SAS

Il existe deux types de tableaux : les tableaux temporaires de travail, qui seront supprimés à la fin d'une session SAS, et les tableaux permanents rangés dans la base. Les noms de tableaux sont composés de deux parties reliées entre elles par un point ; la première est le nom logique de la base le contenant : cela peut être WORK s'il s'agit d'un tableau temporaire, ou bien le nom logique donné à l'aide du paramètre FILE de la commande ALLOCATE. Pour désigner un tableau temporaire, WORK. peut être supprimé ; dans ce cas, le nom du tableau sera composé d'une seule partie, le nom du tableau à proprement parler.

BASE.DETTES désigne un tableau permanent.
WORK.DETTES désigne un tableau temporaire.
DETTES désigne le même tableau temporaire.

4.2. De la matrice d'information spatiale aux tableaux SAS : les instructions d'entrée des données

L'une des situations courantes dans laquelle peut se trouver un chercheur est la suivante : il a fait saisir ses données sur un support quelconque sous la forme d'images de cartes. Avec l'aide d'un informaticien, une conversion de support a été réalisée et la matrice

d'information spatiale constitue un fichier sur disque magnétique. Pour profiter de toutes les facultés de gestion et de traitement de SAS, il est nécessaire de transférer le contenu de ce fichier, la matrice d'information spatiale dans un tableau SAS. Un autre cas courant, où la saisie est réalisée directement par le chercheur dans une base SAS, sera traité plus loin.

Créer un tableau SAS à partir d'un fichier en format image de carte revient à nommer le tableau, décrire son contenu, indiquer un mode de lecture s'appuyant sur le dessin d'enregistrement.La structure du programme sera donc la suivante :

DATA pour nommer le nouveau tableau.
INFILE pour nommer le fichier en format image de carte.
INPUT pour nommer les variables et préciser le mode de lecture.
LABEL pour libeller, donc décrire le contenu des variables.

La création d'un tableau débute le plus souvent par l'instruction DATA ; elle s'écrit :

DATA nom du tableau à créer ;

En respectant les directives du paragraphe 4.1.5., le tableau sera permanent ou temporaire. Par exemple,

DATA BASE.CODPOP ;
définit un tableau permanent qui contiendra les données de la matrice CODPOP.

4.2.1. Nommer un fichier en entrée : l'instruction INFILE

Dans le cas de la création d'un tableau à partir d'un fichier en format image de carte, la seconde instruction de l'étape DATA est l'instruction INFILE. Sa fonction est d'indiquer à SAS dans quel fichier il faut aller chercher les données. Elle s'écrit :

INFILE nom logique du fichier en entrée ;

Puisqu'il faut donner un nom logique, il est nécessaire d'avoir préalablement entré la commande ALLOCATE sur la ligne de commande de l'éditeur de programme afin d'établir la connexion nom physique/nom logique du fichier en entrée ; ici pour la matrice CODPOP :

X ALLOC DA(.CODPOP) FI(ENTREE) SHR

On écrira alors l'instruction :

INFILE ENTREE ;

4.2.2. Nommer les variables et préciser le mode de lecture : l'instruction INPUT

L'instruction INPUT sert à préciser les noms des variables composant le tableau ainsi que le mode de lecture des données, en relation avec le dessin d'enregistrement ; l'instruction INPUT s'écrit de la manière suivante :

INPUT liste de noms de variables et mode de lecture ;

a. la liste des noms de variables peut être donnée de deux manières. En premier lieu, on peut énumérer les noms derrière le mot INPUT ; pour la matrice CODPOP (voir fig. 1.4), on devrait écrire :

INPUT CODE $ POP82 VAR7082 NOM $;

Notons que les noms des variables alphanumériques doivent nécessairement être suivis du caractère dollar ($) qui précise ce type de donnée. Le second procédé pour nommer les variables revient à les numéroter en leur affectant un numéro de début et un numéro de fin. Pour la matrice DETTES, on obtiendrait :

INPUT CODE $ DET1-DET6 ;

Notons que l'instruction suivante est incorrecte :

INPUT CODE $ DET1-DET10 ;

Il faudrait écrire :

INPUT CODE $ DET01-DET10 ;

Cela correspondrait à :

INPUT CODE $ DET01 DET02 DET03 DET04 DET05 DET06 DET07 DET08 DET09 DET10 ;

Notons qu'il est toujours souhaitable de réserver dans les noms deux ou trois caractères afin d'identifier l'année. Bien que très pratique, ce procédé doit être limité aux travaux à très court terme car il est préférable de pouvoir donner une signification aux noms des variables.

b. SAS propose trois modes de lecture des données : le mode liste, le mode colonne et le mode format. Le choix d'un de ces modes dépend pour une large part du dessin d'enregistrement des données ; il est donc utile, avant la saisie, de réfléchir à la manière la plus efficace d'enregistrer son information, sachant qu'elle devra être relue pour être transférée dans une base SAS.

Le mode liste est très utile pour faire vite ; au moment de la saisie, chaque valeur correspondant à chaque variable pour une observation est séparée de celle qui la suit par au moins un caractère espace. Le dessin d'enregistrement devient ainsi variable. Le mode liste correspond donc à l'absence de précision sur les différentes zones d'enregistrement dans l'instruction INPUT qui aura dans ce cas sa plus simple expression. Par exemple, pour la matrice CODPOP, on aura comme précédemment :

INPUT CODE $ POP82 VAR7082 NOM $;

La variabilité du dessin d'enregistrement présente néanmoins quelques inconvénients, conséquences directes de sa simplicité, pouvant avoir des effets déplorables si l'on n'y prête pas attention. Tout d'abord, le séparateur de valeurs étant le caractère espace, les variables alphanumériques ne doivent en aucun cas prendre cette valeur. Dans le cas de la

dixième observation de la matrice CODPOP, les valeurs des variables seront les suivantes :

CODE GNO
POP82 0.37
VAR7082 2.0
NOM GUINEE

La suite de la valeur de la variable NOM (EQUATORIALE) sera ignorée sans aucun message d'erreur. Notons aussi que, par défaut, les variables alphanumériques ne peuvent avoir une longueur supérieure à huit caractères ; dans le cas où il faut plus de huit caractères, il est nécessaire de définir la longueur nécessaire par l'instruction LENGTH. La seconde faiblesse du mode liste réside dans l'obligation de posséder une valeur par variable ; si une valeur manque, elle doit nécessairement être figurée par le caractère point (.). Dans le cas contraire, SAS affecte à la variable manquante la valeur de la variable suivante, etc. On imagine les dégâts provoqués par ce genre d'erreur de lecture, dégâts apparaissant le plus souvent bien plus tard, au moment de l'analyse des données. En définitive, mieux vaut éviter le mode liste chaque fois que le fichier de saisie n'a pas fait l'objet d'un contrôle observation par observation.

Pour le mode colonne, des positions d'enregistrement précisent le début et la fin d'une zone ; il faut donc délimiter une zone par variable. Les délimitations suivent chaque nom. Le dessin d'enregistrement des données facilite la rédaction de cette instruction ; il n'est plus nécessaire de séparer les valeurs par un espace. Dans le cas de la matrice CODPOP (voir fig. 1.4), l'instruction INPUT a la forme suivante :

INPUT CODE $ 1-3 POP82 5-9 VAR7082 12-14 NOM $ 16-34 ;

Les erreurs possibles avec le mode liste sont donc totalement évitées, mais lorsque le nombre de variables est important, des erreurs de positions d'enregistrement peuvent survenir, provoquant là encore des dégâts difficiles à détecter dans l'immédiat, surtout lorsque les valeurs ne sont pas séparées par des espaces. Notons également qu'il est possible de sauter pour toutes les observations la valeur d'une variable figurant dans le fichier en entrée, mais ne devant pas être dans le tableau de la base. Pour ne pas faire entrer la variable VAR7082 dans le tableau CODPOP, on écrirait :

INPUT CODE $ 1-3 POP82 5-9 NOM $ 16-34 ;

En raison de sa simplicité d'utilisation, le mode colonne devrait être préféré au mode liste lorsque le nombre de variables est petit.

Le mode format est plus adapté que le précédent lorsque plusieurs variables occupent des zones de longueurs identiques ; il est d'un usage plus délicat mais raccourcit considérablement la rédaction de l'instruction INPUT. Le principe de fonctionnement du mode format est le suivant :

INPUT (liste de noms de variables) (format de lecture) ;

Le format de lecture précise le type de codage des données (image de carte, binaire, etc...) ; SAS propose un grand nombre de formats. Pour les variables numériques enregistrées sur des images de cartes, le format est du type W.D où W est la longueur d'une zone et D le nombre de décimales. Un bon exemple de l'usage de ce format peut être donné par la matrice DETTES (tab. 1.2) : outre le code alphanumérique de chaque pays, elle contient six variables numériques, enregistrées sur huit caractères, avec une seule décimale à chaque fois. L'instruction INPUT s'écrirait alors de la manière suivante :

INPUT CODE $ (AIDPUB82 AIDCAD82 AIDOPE82 AIDMAR82 ENCOUR82 REGSER82) (8.1) ;

Des pointeurs autorisent le saut, pour toutes les observations, de la valeur d'une variable figurant dans l'enregistrement en entrée, mais ne devant pas rester dans le tableau de la base. Ces pointeurs sont de deux types : le type « se placer sur la position d'enregistrement n° q » qui s'écrit àq et le type « sauter q positions d'enregistrement » s'écrivant +q . Dans l'exemple précédent, en ignorant les variables CODE, AIDMCAD82 et AIDMAR82, l'instruction INPUT deviendrait :

INPUT à4* AIDPUB82 8.1 +8 AIDOPE82 8.1 +8 (ENCOUR82 REGSER82) (8.1) ;

Remarquons que lorsqu'un format s'applique à une liste de variables ne contenant qu'un seul nom de variable, les parenthèses tombent car ce genre de mise en facteurs n'est plus nécessaire. On aurait également pu programmer en factorisant 8.1 +8, c'est-à-dire :

INPUT à4 (AIDPUB82 AIDOPE82) (8.1 +8) (ENCOUR82 REGSER82) (8.1) ;

Bien entendu, rien ne s'oppose au panachage des trois modes de lecture dans la même instruction INPUT. Bien au contraire, c'est l'une des souplesses de mise en oeuvre de SAS que de permettre à la programmation de s'adapter aux données, dans la forme où elles se trouvent et non pas l'inverse. Voici un exemple (un peu artificiel, il est vrai) de panachage des modes de lecture de la matrice DETTES :

INPUT CODE $ 1-3 AIDPUB82 4-11 AIDCAD82 AIDOPE82 +8 (ENCOUR82 REGSER82) (8.1) ;

Les variables CODE et AIDPUB82 sont lues en mode colonne, AIDCAD82 et AIDOPE82 en mode liste ; AIDMAR82 est ignorée puis ENCOUR82 et REGSER82 sont lues en mode format.

Pour achever cette présentation de l'instruction INPUT, notons qu'il existe bien d'autres paramètres facilitant, optimisant et complétant les possibilités de lecture des données, par exemple lorsqu'il y a plusieurs enregistrements physiques pour une observation, ou bien lorsque les fichiers sont hiérarchisés comme c'est

* Dans l'environnement IBM, lors de l'utilisation de claviers AZERTY, « à » est interprété par SAS comme « @ » (arobas) ; c'est le caractère que l'on trouve dans les manuels de langue anglaise. De même : « £ » est interprété comme « # » et « ^ » comme « ¬ ».

```
                    CONTENTS OF SAS MEMBER BASE.DETTES

CREATED BY TSO USERID WANIEZ   ON CPUID 63-3081-001652   AT 11:24 THURSDAY, DECEMBER 12, 1985   BY SAS RELEASE 5.08
INFILE(DSN=[redacted]   VOL=SER=MVS104)   DSNAME=[redacted].STAGBASE   OBSERVATIONS PER TRACK =852
BLKSIZE=23434   LRECL=55   GENERATED BY DATA
NUMBER OF OBSERVATIONS: 23   NUMBER OF VARIABLES: 7
MEMTYPE:   DATA

                          ----ALPHABETIC LIST OF VARIABLES AND ATTRIBUTES-----
 #  VARIABLE  TYPE   LENGTH  POSITION  FORMAT      INFORMAT    LABEL
 3  AIDCAD82  NUM       8        15                           aide ext. pays cad millions $ 1982
 5  AIDMAR82  NUM       8        31                           aide ext. marche millions $ 1982
 4  AIDOPE82  NUM       8        23                           aide ext. pays opep millions $ 1982
 2  AIDPUB82  NUM       8         7                           aide ext. publique millions $ 1982
 1  CODE      CHAR      3         4                           codes des etats iso
 6  ENCOUR82  NUM       8        39                           encours dette millions $ 1982
 7  REGSER82  NUM       8        47                           reglements service dette millions $ 1982

                          ----LIST OF VARIABLES AND ATTRIBUTES BY POSITION----
 #  VARIABLE  TYPE   LENGTH  POSITION  FORMAT      INFORMAT    LABEL
 1  CODE      CHAR      3         4                           codes des etats iso
 2  AIDPUB82  NUM       8         7                           aide ext. publique millions $ 1982
 3  AIDCAD82  NUM       8        15                           aide ext. pays cad millions $ 1982
 4  AIDOPE82  NUM       8        23                           aide ext. pays opep millions $ 1982
 5  AIDMAR82  NUM       8        31                           aide ext. marche millions $ 1982
 6  ENCOUR82  NUM       8        39                           encours dette millions $ 1982
 7  REGSER82  NUM       8        47                           reglements service dette millions $ 1982
```

Fig. 4.1. – *Le contenu du tableau permanent DETTES.*

très souvent le cas lorsque les volumes d'information sont très grands.

4.2.3. Identifier les variables : l'instruction LABEL

Le choix des noms de variables doit être fait après une grande réflexion, de manière à ce qu'ils aient une signification aussi claire que possible pour l'usager des données. Dans le cas du tableau CODPOP, POP82 est une variable contenant la population des pays les moins avancés en 1982 ; on perd dans ce nom l'unité de mesure, ici le million d'habitants. Pour la matrice DETTES, le nom de la variable REGSER82 est évident pour celui qui a créé le tableau (règlements pour le service de la dette en 1982, en millions de dollars), mais assez énigmatique pour un autre usager. C'est pourquoi il est utile d'affecter un libellé à chaque nom de variable, dans tous les cas où les données peuvent servir à d'autres chercheurs, un jour ou l'autre, ce qui est le cas de l'essentiel des travaux en équipe. A l'aide d'une procédure (PROC CONTENTS), il sera possible de lister les noms des variables et les libellés correspondants d'un tableau particulier ou de tous les tableaux d'une base, et de retrouver ainsi la signification précise des noms des variables. Ces libellés, d'une longueur maximale de 40 caractères doivent contenir, si possible, la signification de la variable, l'unité de mesure et l'année. L'instruction LABEL affectant un libellé à chaque variable s'écrit :

```
LABEL nom de variable =
      libellé nom de variable=libellé ;
```

Le nom de variable identifie la variable qu'il faut libeller ; cette variable doit être réellement présente dans le tableau en cours de création. Dans le cas contraire, un message d'erreur est affiché sur le journal de bord. Notons que s'il manque un ou plusieurs noms de variables dans l'instructions LABEL, ces variables existeront quand-même dans le tableau en cours de création, mais sans libellé bien sûr. Pour le tableau DETTES (tab. 1.2), les libellés des variables s'écrivent de la manière suivante :

```
LABEL CODE=CODE DES ETATS ISO
      AIDPUB82 = AIDE EXT. PUBLIQUE
                 MILLIONS $ 1982
      AIDCAD82 = AIDE EXT. PAYS CAD
                 MILLIONS $ 1982
      AIDOPE82 = AIDE EXT. PAYS OPEP
                 MILLIONS $ 1982
      AIDMAR82 = AIDE EXT. MARCHE
                 MILLIONS $ 1982
      ENCOUR82 = ENCOURS DETTE
                 MILLIONS $ 1982
      REGSER82 = REGLEMENTS SERVICE
                 DETTE MILLIONS $ 1982 ;
```

Le résultat de la procédure CONTENTS est donné par la figure 4.1 : à chaque nom de variable correspond un libellé figurant dans la colonne LABEL. Grâce à l'instruction LABEL, la gestion d'une base de données SAS présente un intérêt supplémentaire puisqu'il devient possible d'échanger ses données avec d'autres chercheurs utilisant aussi SAS, sans avoir à donner la moindre information complémentaire. C'est un aspect important du choix de SAS comme système de gestion des données.

4.2.4. Exemples de programmes d'entrée de la matrice DETTES

a. Lecture en mode colonne : programme EXPL2. Il faut d'abord entrer sur la ligne de commande de l'éditeur de programme les deux commandes suivantes :

```
X ALLOC DA(.STAGPROG(EXPL2)) FI(PROG)
      SHR INCLUDE PROG
```

```
/*-------------------------------------------------*/
/* CREATION DU TABLEAU EN MODE COLONNE             */
/*-------------------------------------------------*/

/*-----> ALLOCATION DU FICHIER EN ENTREE */

X ALLOC DA(.DETTES) FI(ENTREE) SHR;

/*-----> CREATION DU TABLEAU TEMPORAIRE DETTES */

DATA DETTES; INFILE ENTREE;
INPUT CODE $ 1-3 AIDPUB82 4-11 AIDCAD82 12-19 AIDOPE82 20-27
                 AIDMAR82 28-35 ENCOUR82 36-43 REGSER82 44-51;
LABEL CODE=CODE DES ETATS ISO
      AIDPUB82=AIDE EXT. PUBLIQUE MILLIONS $ 1982
      AIDCAD82=AIDE EXT. PAYS CAD MILLIONS $ 1982
      AIDOPE82=AIDE EXT. PAYS OPEP MILLIONS $ 1982
      AIDMAR82=AIDE EXT. MARCHE MILLIONS $ 1982
      ENCOUR82=ENCOURS DETTE MILLIONS $ 1982
      REGSER82=REGLEMENTS SERVICE DETTE MILLIONS $ 1982;
RUN;
```

Fig. 4.2. – *Le programme EXPL2 : création d'un tableau avec lecture des données en mode colonne.*

```
/*-------------------------------------------------*/
/* CREATION DU TABLEAU DETTES EN MODE FORMAT        */
/*-------------------------------------------------*/

/*-----> ALLOCATION DU FICHIER EN ENTREE */

X ALLOC DA(.DETTES) FI(ENTREE) SHR;

/*-----> CREATION DU TABLEAU TEMPORAIRE DETTES */

DATA DETTES; INFILE ENTREE;
INPUT CODE $ 1-3 a4 (AIDPUB82 AIDCAD82) (8.1)
                    AIDOPE82 20-27
                    (AIDMAR82 ENCOUR82 REGSER82) (8.1);
LABEL CODE=CODE DES ETATS ISO
      AIDPUB82=AIDE EXT. PUBLIQUE MILLIONS $ 1982
      AIDCAD82=AIDE EXT. PAYS CAD MILLIONS $ 1982
      AIDOPE82=AIDE EXT. PAYS OPEP MILLIONS $ 1982
      AIDMAR82=AIDE EXT. MARCHE MILLIONS $ 1982
      ENCOUR82=ENCOURS DETTE MILLIONS $ 1982
      REGSER82=REGLEMENTS SERVICE DETTE MILLIONS $ 1982;
RUN;
```

Fig. 4.3. – *Le programme EXPL3 : création d'un tableau avec lecture des données en mode format.*

```
                    CONTENTS PROCEDURE
          PHYSICAL CHARACTERISTICS OF THE OS DATA SET

DSNAME= [redacted].[redacted].STAGBASE   UNIT=SYSDA   VOL=SER=MVS102.   DISP=OLD   DEVICE=3380 DISK
CREATED ON DECEMBER 12, 1985

                    SAS DATA LIBRARY DIRECTORY

NAME        MEMTYPE     #OBS        TRACKS      EXTENTS

CODPOP      DATA        23              1           1
DETTES      DATA        23              1           1
REGIME      DATA        23              1           1

                TOTAL TRACKS USED =     4
                HIGH  TRACKS USED =     4
```

Fig. 4.4. – *Le répertoire de la base .STAGBASE.*

On obtient alors le texte de la figure 4.2 ; il est possible de soumettre ce programme par la touche de fonction n° 3. Il s'agit de la création d'un tableau temporaire nommé DETTES (en réalité WORK.DETTES). Il n'y a donc pas de base à allouer puisque la base temporaire WORK est allouée à l'entrée de la commande tso SAS5. Si l'on avait voulu créer un tableau permanent, il aurait fallu inclure dans le programme, après la première comande ALLOCATE, une seconde, de la manière suivante :

X ALLOC DA(.STAGBASE) FI(BASE) OLD ;

et écrire ensuite :

DATA BASE.DETTES ; au lieu de DATA DETTES ;

b. lecture panachée, mode colonne et mode format : programme EXPL3. Il est nécessaire d'opérer comme précédemment pour inclure le texte du programme dans l'écran éditeur de programme. Le texte de la figure 4.3 vient alors s'afficher dans la moitié inférieure de l'écran. On peut soumettre ce programme avec la touche de fonction n° 3. Le résultat est identique à celui du programme EXPL2. Notons aussi que le déclenchement de l'étape DATA intervient après l'instruction RUN ;

4.3. Création et correction interactive d'un tableau : la procédure FSEDIT

Lorsque le volume des données ne dépasse pas quelques centaines de nombres, la saisie directe par le chercheur peut être envisagée. De plus, après avoir constitué un tableau, des erreurs de saisie peuvent être détectées. La procédure FSEDIT réalise la double fonction de saisie et de correction des données. Il s'agit

d'une procédure de SAS/FSP, c'est-à-dire de l'usage de tableau SAS en mode pleine page à partir d'un terminal de type IBM 3279. L'appel de la procédure est le suivant :

PROC FSEDIT NEW = nom du tableau à créer ;
PROC FSEDIT DATA = nom du tableau à modifier ;

Dans les deux cas, la base doit obligatoirement être allouée en OLD afin de pouvoir être modifiée, soit par ajout d'un tableau, soit par modification d'un tableau existant déjà.

Lorsque le tableau est en création, une image écran pour sa définition (fig. 4.5) apparait en premier. Définir un tableau, c'est donner le nom des variables, leur type, leur longueur et leur libellé en clair ; cette opération est assurée par le remplissage de l'image de définition : on donne d'abord le nom de la variable, puis, à l'aide de la touche de tabulation, le curseur doit être placé sur la colonne TYPE, qui peut être C si la variable est alphanumérique ou N si elle est numérique. La longueur doit être fournie sur le zone LENGTH : elle sera la vraie longueur maximale des chaînes de caractères dans le cas d'une variable alphanumérique ; la longueur vaudra le plus souvent 8 pour les variables numériques. Enfin, le libellé peut être entré sur 40 caractères au maximum. Pour définir la variable suivante, il faut amener le curseur sur la ligne suivante à l'aide d'une touche de tabulation. Lorsque les variables d'un tableau à créer sont toutes définies, la saisie proprement dite peut commencer en avec la touche de fonction n° 3.

L'image d'édition s'affiche alors (fig. 4.6). Chaque variable à saisir est un champ identifiée par le nom de la variable. Il faut amener le curseur sur le premier champ à l'aide d'une touche de tabulation et entrer la valeur de la première variable pour la première observation. Puis le curseur est amené sur le second champ correspondant à la seconde variable qui est saisie et ainsi de suite. Notons que le numéro

FSEDIT Data Set Definition for BASE.REGIME

Command ===>

Name	Type	Length	Label	Format
code	c	3	code des etats iso	
regpol82	c	2	regimes politiques 1982	
budmil82	n	8	budget militaire millions $ 1982	
defcal80	n	8	deficit calorique % moy. mondiale 1980	

Fig. 4.5. – *L'écran de définition d'un nouveau tableau avec FSEDIT.*

```
CODE:      BEN

REGPOL82: MI

BUDMIL82: 23 _____

DEFCAL80: 0_____
```

Fig. 4.6. – *L'écran de saisie avec FSEDIT.*

d'observation apparaît en haut de l'écran, à droite (NEW 1). Lorsque toutes les variables d'une observation sont entrées, on passe à l'observation suivante à l'aide de la touche de fonction n° 9 ; le numéro d'observation s'accroît d'une unité, et une image d'édition vide proposant des champs identifiés par les noms des variables apparaît. Il faut saisir l'observation comme précédemment et ainsi de suite.

Une pression sur la touche de fonction n° 3 achève la saisie. On se retrouve devant l'écran éditeur de programme/journal de bord.

La procédure FSEDIT admet quelques commandes d'édition facilitant la saisie qui doivent être entrées sur la ligne de commande visible sur la figure 4.6. Les deux commandes suivantes sont très utilisées :

NAME nom d'une variable (puis ENTREE)
L valeur (puis ENTREE)

Ceci permet de repérer une valeur d'une variable du tableau en cours d'édition. Ceci est très utile pour repérer une valeur erronée : le curseur se positionne dessus et la bonne valeur peut être saisie à sa place. Si plusieurs observations doivent être localisées, ayant la même valeur sur la même variable, la touche de fonction n° 5 doit être utilisée dès la seconde fois : elle évite d'entrer à nouveau les commandes NAME et L.

Pour repérer une observation par son rang dans le tableau, il suffit de donner son numéro sur la ligne de commande et cette observation sera affichée. La touche de fonction n° 7 provoque l'affichage de l'observation précédente, la touche de fonction n° 8 l'observation suivante, si elles existent déjà dans le tableau.

Notons enfin qu'il n'est pas obligatoire de saisir une valeur dans chaque champ pour chaque observation ; en l'absence d'une valeur, celle-ci est manquante dans le tableau (. pour les variables numériques et espace pour les variables alphanumériques).

4.4. Connaître le contenu d'une base : procédures CONTENTS et PRINT

Deux procédures permettent de connaître le contenu d'un tableau dans une base SAS. La procédure CONTENTS liste la composition d'une base en général ou d'un tableau en particulier, c'est-à-dire le répertoire de la base avec les noms des tableaux et les répertoires des tableaux avec les noms des variables. Ces derniers peuvent être listés en abrégé (c'est-à-dire sans leurs libellés), dans l'ordre alphabétique des variables, ou d'après leur position dans le tableau (c'est-à-dire dans l'ordre de la liste des noms de variables dans l'instruction INPUT). La procédure CONTENTS admet plusieurs options correspondant à ces possibilités :

```
PROC CONTENTS DATA=BASE. - ALL -;
```

liste le répertoire de la base et celui de tous les tableaux, les noms des variables étant dans l'ordre alphabétique.

```
PROC CONTENTS DATA=BASE.DETTES;
```

liste le répertoire du tableau DETTES, les noms des variables étant dans l'ordre alphabétique.

```
PROC CONTENTS DATA=
        BASE.DETTES POSITION;
```

liste le répertoire du tableau DETTES, les noms de variables étant dans l'ordre alphabétique et dans l'ordre des positions dans le tableau.

```
PROC CONTENTS DATA=
    BASE.DETTES POSITION SHORT;
```

liste le répertoire du tableau DETTES en abrégé, les noms des variables étant dans l'ordre alphabétique et dans l'ordre de position dans le tableau.

```
PROC CONTENTS DATA=
        BASE. - ALL - NODS;
```

liste uniquement le répertoire de la base, c'est-à-dire les noms des tableaux. La figure 4.4 correspond à la sortie sur l'écran OUTPUT de ce dernier cas de figure. Elle est composée de deux parties : les caractéristiques physiques de la base sont dans la partie supérieure ; il s'agit du nom physique (DSNAME), du support (UNIT), ici un disque magnétique (SYSDA), du nom du support (VOL=SER), de la disposition du fichier (DISP) et du type de support (DEVICE), ici un disque de la série IBM 3380. Dans la partie inférieure de la sortie figurent le répertoire de la base contenant de nombreuses informations, notamment le nom des tableaux (NAME), le nombre d'observations (OBS), la taille de chaque tableau exprimée en pistes de disque IBM 3380 (TRACKS). Le nombre total de pistes (TOTAL TRACKS USED) est calculé à partir de la taille de chaque tableau et de la taille du répertoire de la base. La figure 4.1 vue en 4.2.3. correspond au programme :

```
PROC CONTENTS DATA=
        BASE.DETTES POSITION;
```

On y trouve des informations relatives au seul tableau DETTES. Cette sortie est composée de deux parties : les caractéristiques physiques du tableau figurent dans la partie supérieure ; on peut y lire notamment la date de création, la version de SAS (ici 5.08), le nom du fichier en entrée (INFILE), le nom physique de la base (DSNAME), le nombre d'observations (23) et le nombre de variables (7). La partie inférieure de la sortie donne la liste des variables par

ordre alphabétique et par ordre de position dans le tableau. Ici figurent le numéro d'ordre de chaque variable, son nom, son type (CHAR pour alphanumérique, NUM pour numérique), ainsi que son libellé (LABEL). La liste des variables par ordre de position dans le tableau simplifie énormément l'usage qui pourra en être fait ultérieurement.

Connaître le contenu d'un tableau, c'est aussi pouvoir lister les valeurs des variables le composant. La procédure PRINT est chargée de cette opération.

```
PROC PRINT DATA=BASE.DETTES ;
```

affiche les valeurs de toutes les variables composant le tableau DETTES (tab. 4.1).

```
PROC PRINT DATA=BASE.DETTES
                UNIFORM ROUND ;
```

affiche les valeurs de toutes les variables composant le tableau DETTES, en arrondissant à trois décimales et en adoptant une présentation uniforme facilitant la consultation. Notons qu'une colonne supplémentaire OBS donne le rang de chaque observation.

OBS	CODE	AIDPUB82	AIDCAD82	AIDOPE82	AIDMAR82	ENCOUR82	REGSER82
1	BEN	95.000	76.000	4.400	110.100	566.40	51.800
2	BWA	101.600	92.900	8.700	16.100	387.80	30.700
3	HVO	213.000	201.000	12.000	43.500	343.80	20.400
4	BDI	130.500	120.600	6.000	30.400	227.70	5.500
5	DJI	58.300	54.800	3.500	0.300	43.10	3.300
6	ETH	216.700	199.800	0.100	19.400	897.60	61.500
7	GMB	50.000	43.000	4.600	-2.600	151.40	7.200
8	GIN	66.600	60.200	5.200	20.300	1283.70	79.300
9	GNB	69.200	59.700	8.500	0.800	126.80	3.500
10	GNO	14.000	13.000	1.000	-4.900	0.00	0.000
11	LSO	89.700	86.800	2.900	3.900	152.20	11.200
12	MWI	121.500	121.400	0.100	7.000	779.10	7.300
13	MLI	206.900	159.300	35.200	9.200	828.70	12.000
14	NER	254.000	167.600	84.800	44.400	691.60	149.900
15	UGA	133.200	127.400	5.700	34.000	546.20	124.000
16	CAF	89.900	88.700	1.200	11.500	215.10	4.000
17	TZA	710.100	656.600	39.000	56.800	2001.40	118.900
18	RWA	150.800	148.900	1.900	3.100	194.80	5.700
19	SLE	79.800	80.800	1.400	2.300	317.90	41.200
20	SOM	468.300	279.000	183.300	158.200	978.30	32.600
21	SDN	743.100	554.400	185.700	68.600	5473.10	91.300
22	TCD	64.700	60.900	3.800	-1.700	178.30	1.200
23	TGO	77.200	73.400	3.800	21.400	812.00	35.300

Tab. 4.1. – *L'affichage du tableau permanent DETTES.*

5. Créer un tableau SAS à partir d'un ou plusieurs tableaux existant déjà dans la base

Une fois créé, un tableau SAS peut faire directement l'objet d'une analyse statistique ; dans la grande majorité des cas, les données brutes figurant dans la base doivent être transformées en indicateurs pertinents (par exemple, le montant de la dette par habitant) avant d'être analysées. Cette opération nécessite la création de nouveaux tableaux permanents ou temporaires. De plus, il est bien souvent utile de combiner les variables figurant dans plusieurs tableaux différents pour procéder à des corrélations. Ce chapitre présente les instructions indispensables à la réalisation de tous ces traitements se déroulant obligatoirement dans une étape DATA.

5.1. Créer un nouveau tableau à partir d'un seul tableau existant déjà

Quatre cas de figure peuvent se présenter : en premier lieu, on peut vouloir créer un tableau représentant la copie intégrale d'un tableau existant déjà ; en second lieu, on peut ne désirer conserver qu'une partie des variables ; dans le troisième cas, seules quelques observations doivent être recopiées ; enfin, les deux derniers cas peuvent se combiner pour sélectionner quelques variables sur quelques observations uniquement.

5.1.1. Copie intégrale

Il faut, dans un nouveau tableau, copier l'intégralité d'un tableau existant déjà. L'instruction SET assure cette opération. Elle a pour syntaxe :

SET nom du tableau à copier ;

Par exemple, la création d'un tableau temporaire, copie intégrale du tableau permanent CODPOP se programme de la manière suivante :

DATA CODPOP ; SET BASE.CODPOP ;

Le tableau temporaire CODPOP aura, comme le tableau permanent CODPOP 23 observations et 4 variables (tab. 5.1). A l'exécution, cette dernière information est affichée sur le journal de bord.

5.1.2. Copie d'une partie des variables sur l'ensemble des observations

Le plus souvent, il n'est pas nécessaire de copier l'intégralité d'un tableau dans un autre. Dans ce cas, il est préférable de ne sélectionner que les variables désirées ; cette sélection peut se faire par choix explicite des variables retenues, ou bien par élimination explicite des variables non retenues. Au premier cas de figure correspond l'instruction :

KEEP liste de variables à retenir ;

Au second cas :

DROP liste de variables à éliminer ;

La liste de variables peut prendre trois formes, comme dans tous les cas où elle sera nécessaire par la suite. La première, la plus simple et la plus longue, revient à nommer les variables en séparant les noms par des espaces :

DROP VAR7082 NOM ;

Ici, on élimine les variables VAR7082 et NOM. La seconde manière revient à nommer toutes les variables comprises entre une variable initiale et une variable finale d'après leur position dans le tableau :

DROP POP82--NOM ;

Cela revient à éliminer les variables POP82 VAR7082 et NOM, c'est-à-dire DROP POP82 VAR7082 NOM ; deux tirets sont nécessaires pour séparer les variables initiale et finale dans la liste. Enfin, lorsque les variables sont numérotées, la liste peut exprimer toutes les variables comprises entre une variable initiale et une variable finale, sans qu'elles

OBS	CODE	NOM	POP82	VAR7082
1	BEN	BENIN	3.6200	2.50000
2	BWA	BOTSWANA	0.9700	3.80000
3	HVO	BURKINA-FASO	6.3600	1.40000
4	BDI	BURUNDI	4.2600	1.80000
5	DJI	DJIBOUTI	0.3300	6.40000
6	ETH	ETHIOPIE	32.7800	2.40000
7	GMB	GAMBIE	0.6300	2.90000
8	GIN	GUINEE	5.0600	2.10000
9	GNB	GUINEE-BISSAU	0.8500	4.10000
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000
11	LSO	LESOTHO	1.4100	2.40000
12	MWI	MALAWI	6.2700	2.60000
13	MLI	MALI	7.3400	2.10000
14	NER	NIGER	5.6100	2.60000
15	UGA	OUGANDA	14.1200	3.10000
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000
17	TZA	TANZANIE	20.2300	3.40000
18	RWA	RWANDA	5.5100	3.30000
19	SLE	SIERRA LEONE	3.4100	1.60000
20	SOM	SOMALIE	5.0800	5.10000
21	SDN	SOUDAN	19.7900	3.00000
22	TCD	TCHAD	4.6800	2.10000
23	TGO	TOGO	2.6800	2.40000

Tab. 5.1. – *La copie intégrale du tableau permanent CODPOP.*

soient nécessairement dans cet ordre dans le tableau :

```
DROP DET01-DET05 ;
```

revient à :

```
DROP DET01 DET02 DET03 DET04 DET05 ;
```

Un seul tiret doit séparer les noms des variables initiale et finale dans ce cas. La création d'un tableau temporaire, copie des variables CODE et NOM du tableau CODPOP pourra prendre les formes suivantes :

```
DATA CODNOM ; SET BASE.CODPOP ;
                  KEEP CODE NOM ;
```

ou bien :

```
DATA CODNOM ; SET BASE.CODPOP ;
                  DROP POP82 VAR7082 ;
```

Le choix entre KEEP et DROP est opéré de manière à donner une liste de variables aussi courte que possible (tab. 5.2).

OBS	CODE	NOM
1	BEN	BENIN
2	BWA	BOTSWANA
3	HVO	BURKINA-FASO
4	BDI	BURUNDI
5	DJI	DJIBOUTI
6	ETH	ETHIOPIE
7	GMB	GAMBIE
8	GIN	GUINEE
9	GNB	GUINEE-BISSAU
10	GNO	GUINEE EQUATORIALE
11	LSO	LESOTHO
12	MWI	MALAWI
13	MLI	MALI
14	NER	NIGER
15	UGA	OUGANDA
16	CAF	REP. CENTRAFICAINE
17	TZA	TANZANIE
18	RWA	RWANDA
19	SLE	SIERRA LEONE
20	SOM	SOMALIE
21	SDN	SOUDAN
22	TCD	TCHAD
23	TGO	TOGO

Tab. 5.2. – *Le tableau CODNOM.*

OBS	CODE	NOM	POP82	VAR7082
1	BEN	BENIN	3.62000	2.50000

Tab. 5.3. – *Le tableau BENIN.*

5.1.3. Copie d'une partie des observations sur l'ensemble des variables.

Copier une partie des observations revient à exprimer une condition pour les retenir ou non. Cette condition est précisée par l'instruction IF dont la syntaxe prend deux formes ; voici la première :

IF variable condition EQ
GE
GT
LE
LT
NE valeur de la condition
THEN DELETE ;

Cela revient à supprimer les observations pour lesquelles la variable condition a une valeur égale (EQ), plus grande ou égale (GE), strictement plus grande (GT), plus petite ou égale (LE), strictement plus petite (LT) ou différente (NE) de la valeur de la condition. Pour les variables alphanumériques, ces valeurs doivent figurer entre quotes ('). Créer un tableau BENIN contenant pour le seul BENIN les quatre variables du tableau CODPOP revient à soumettre le programme suivant (tab ; 5.3) :

```
DATA BENIN ; SET BASE.CODPOP ;
IF CODE NE 'BEN' THEN DELETE ;
```

De même, pour créer un tableau comprenant les pays ayant une variation moyenne annuelle de leur population supérieure ou égale à 3%, il faudrait exécuter le programme suivant :

```
DATA VARSUP3 ; SET BASE.CODPOP ;
IF VAR7082 LT 3 THEN DELETE ;
```

Le tableau temporaire VARSUP3 (tab. 5.4) contient 8 observations (Botswana, Djibouti, Guinée-Bissau, Ouganda, Tanzanie, Rwanda, Somalie et Soudan) et les quatre variables du tableau CODPOP.

La seconde forme de l'instruction IF revient à conserver les observations si la variable condition a une valeur égale (etc. comme précédemment) à la valeur de la condition, c'est-à-dire :

IF variable condition EQ
GE
GT
LE
LT
NE valeur de la condition ;

Pour le tableau BENIN, on aurait :

```
DATA BENIN ; SET BASE.CODPOP ;
                IF CODE EQ 'BEN' ;
```

et pour le tableau VARSUP3 :

```
DATA VARSUP3 ; SET BASE.CODPOP ;
                IF VAR7082 GE 3 ;
```

Notons enfin qu'il est possible de relier entre elles plusieurs conditions par OR (ou) ou bien AND (et) de manière à expimer une condition composée. Ainsi, pour créer un tableau composé des pays ayant une croissance démographique supérieure ou égale à 3% par an entre 1970 et 1982 et renfermant au moins 10 millions d'habitants, il faut soumettre le programme :

```
DATA VARPOP ; SET BASE.CODPOP ;
IF VAR7082 GE 3 AND POP82 GE 10 ;
```

Le tableau VARPOP (tab. 5.5) contient 3 observations (Ouganda, Tanzanie, Soudan) correspondant à cette condition composée et les quatre variables du tableau CODPOP. On aurait aussi pu procéder par élimination explicite des observations non conformes :

OBS	CODE	NOM	POP82	VAR7082
1	BWA	BOTSWANA	0.9700	3.80000
2	DJI	DJIBOUTI	0.3300	6.40000
3	GNB	GUINEE-BISSAU	0.8500	4.10000
4	UGA	OUGANDA	14.1200	3.10000
5	TZA	TANZANIE	20.2300	3.40000
6	RWA	RWANDA	5.5100	3.30000
7	SOM	SOMALIE	5.0800	5.10000
8	SDN	SOUDAN	19.7900	3.00000

Tab. 5.4. – *Le tableau VARSUP3.*

OBS	CODE	NOM	POP82	VAR7082
1	UGA	OUGANDA	14.1200	3.10000
2	TZA	TANZANIE	20.2300	3.40000
3	SDN	SOUDAN	19.7900	3.00000

Tab. 5.5. – *Le tableau VARPOP.*

```
DATA VARPOP ; SET BASE.CODPOP ;
IF VAR7082 LT 3 OR POP82 LT 10 THEN
DELETE ;
```

Ceci donnerait strictement le même résultat. Il faut faire très attention à remplacer AND par OR dans ce cas ; la composition des conditions peut donner parfois des résultats qui, tout en étant logiques peuvent apparaître inattendus ; seule une bonne pratique permet de formuler ces conditions de manière sûre. Dans tous les cas, il est souhaitable de vérifier les résultats à l'aide de la procédure PRINT après la création du nouveau tableau.

5.1.4. Copie d'une partie des variables et des observations

Bien entendu, rien ne s'oppose à l'usage conjoint des instructions de sélection des variables et de celles de sélection des observations :

```
DATA BENIN ; SET BASE.CODPOP ;
KEEP CODE NOM ; IF CODE EQ 'BEN' ;
```

Ce programme crée un tableau temporaire contenant l'observation dont le code est BEN et cela uniquement pour les deux variables CODE et NOM (tab. 5.6). De plus, il est possible de réaliser une sélection d'observations sur une variable présente dans le tableau à copier mais ne figurant pas dans le tableau résultat :

```
DATA VARSUP3 ; SET BASE.CODPOP ;
IF VAR7082 LT 3 THEN DELETE ;
DROP VAR7082 ;
```

Comme précédemment, le tableau temporaire VARSUP3 (tab. 5.7) renferme les huit observations ayant des valeurs supérieures ou égales à 3 sur la variable VAR7082 et uniquement les variables CODE et NOM. C'est donc bien au moment de la copie du contenu du tableau existant dans le tableau à créer que disparaissent de ce dernier les variables et les observations non retenues.

5.1.5. Changer le nom des variables dans le nouveau tableau

L'instruction RENAME permet de changer le nom

OBS	CODE	NOM
1	BEN	BENIN

Tab. 5.6. – *La tableau BENIN sans POP82 ni VAR7082.*

OBS	CODE	NOM
1	BWA	BOTSWANA
2	DJI	DJIBOUTI
3	GNB	GUINEE-BISSAU
4	UGA	OUGANDA
5	TZA	TANZANIE
6	RWA	RWANDA
7	SOM	SOMALIE
8	SDN	SOUDAN

Tab. 5.7. – *Le tableau VARSUP3 sans POP82 ni VAR7082.*

OBS	PAYS	NOM	POP1982	VAR7082
1	BEN	BENIN	3.6200	2.50000
2	BWA	BOTSWANA	0.9700	3.80000
3	HVO	BURKINA-FASO	6.3600	1.40000
4	BDI	BURUNDI	4.2600	1.80000
5	DJI	DJIBOUTI	0.3300	6.40000
6	ETH	ETHIOPIE	32.7800	2.40000
7	GMB	GAMBIE	0.6300	2.90000
8	GIN	GUINEE	5.0600	2.10000
9	GNB	GUINEE-BISSAU	0.8500	4.10000
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000
11	LSO	LESOTHO	1.4100	2.40000
12	MWI	MALAWI	6.2700	2.60000
13	MLI	MALI	7.3400	2.10000
14	NER	NIGER	5.6100	2.60000
15	UGA	OUGANDA	14.1200	3.10000
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000
17	TZA	TANZANIE	20.2300	3.40000
18	RWA	RWANDA	5.5100	3.30000
19	SLE	SIERRA LEONE	3.4100	1.60000
20	SOM	SOMALIE	5.0800	5.10000
21	SDN	SOUDAN	19.7900	3.00000
22	TCD	TCHAD	4.6800	2.10000
23	TGO	TOGO	2.6800	2.40000

Tab. 5.8. – *Le tableau CODPOP après un RENAME de CODE et de POP1982.*

des variables dans le tableau créé. Elle s'écrit :

```
RENAME ancien nom=nouveau nom ...
       ancien nom=nouveau nom ;
```

Une seule instruction RENAME est nécessaire pour opérer tous les changements de noms désirés :

```
DATA CODPOP ; SET BASE.CODPOP ;
RENAME CODE=PAYS POP82=POP1982 ;
```

Dans ce cas, le tableau CODPOP est composé de 23 observations et des quatre variables suivantes : PAYS, POP1982, VAR7082 et NOM. Attention, si RENAME et KEEP ou DROP sont utilisés simultanément, KEEP ou DROP est exécuté en premier ; cela signifie que la liste de variables figurant derrière KEEP ou DROP doit être composée des anciens noms. Le programme ci-après est incorrect :

```
DATA CODPOP ; SET BASE.CODPOP ;
RENAME CODE=PAYS POP82=POP1982 ;
KEEP PAYS POP1982 ;
```

Un message d'erreur indiquant que les variables PAYS et POP1982 n'existent pas sera affiché sur le journal de bord. Il faudrait écrire par contre :

```
DATA CODPOP ; SET BASE.CODPOP ;
RENAME CODE=PAYS POP82=POP1982 ;
KEEP CODE POP82 ;
```

Le tableau temporaire CODPOP (tab. 5.8) aurait 23 observations et les quatre variables suivantes : PAYS, POP1982, VAR7082 et NOM. L'usage des instructions de sélection IF, KEEP, DROP et RENAME présente quelques pièges ; il est toujours souhaitable de s'assurer du contenu du nouveau tableau à l'aide de la procédure CONTENTS.

5.2. Créer un nouveau tableau à partir de plusieurs autres tableaux existant déjà

Dans bien des cas, il est utile de sélectionner quelques variables dans plusieurs tableaux afin de rechercher des corrélations. L'association des différents tableaux ne devra se faire que si la même variable (même nom et même type) identifie les observations dans chacun d'eux ; les observations doivent, de plus, figurer dans l'ordre croissant des valeurs de cette variable ou dans l'ordre alphabétique si elle est alphanumérique. L'instruction permettant d'associer plusieurs tableaux a pour syntaxe :

```
MERGE liste de noms de tableaux ;
      BY variable d'association ;
```

Par exemple, si à partir des tableaux CODPOP et REGIME on désirait constituer un unique tableau renfermant les variables CODE POP82 NOM et BUDMIL82 (cela afin de calculer le montant des dépenses militaires par habitant), on soumettrait le programme suivant :

```
DATA CODREG ; MERGE BASE.CODPOP
                    BASE.REGIME ; BY CODE ;
KEEP CODE POP82 NOM BUDMIL82 ;
```

A l'exécution de ce programme, SAS vérifiera d'abord la présence de la variable CODE, de type alphanumérique dans les deux tableaux ; l'ordre alphabétique sera donc requis. Ici, cela n'est pas le cas (HVO est encadré par BWA et BDI) : un message d'erreur indiquera que la variable d'association doit nécessairement être triée. SAS arrêtera l'exécution de

OBS	CODE	NOM	POP82	BUDMIL82
1	BDI	BURUNDI	4.2600	23
2	BEN	BENIN	3.6200	23
3	BWA	BOTSWANA	0.9700	29
4	CAF	REP. CENTRAFICAINE	2.3900	12
5	DJI	DJIBOUTI	0.3300	3
6	ETH	ETHIOPIE	32.7800	485
7	GIN	GUINEE	5.0600	44
8	GMB	GAMBIE	0.6300	.
9	GNB	GUINEE-BISSAU	0.8500	.
10	GNQ	GUINEE EQUATORIALE	0.3700	5
11	HVO	BURKINA-FASO	6.3600	41
12	LSO	LESOTHO	1.4100	.
13	MLI	MALI	7.3400	46
14	MWI	MALAWI	6.2700	22
15	NER	NIGER	5.6100	16
16	RWA	RWANDA	5.5100	18
17	SDN	SOUDAN	19.7900	470
18	SLE	SIERRA LEONE	3.4100	19
19	SOM	SOMALIE	5.0800	150
20	TCD	TCHAD	4.6800	62
21	TGO	TOGO	2.6800	22
22	TZA	TANZANIE	20.2300	285
23	UGA	OUGANDA	14.1200	852

Tab. 5.9. – *Le tableau CODREG.*

l'étape et le tableau CODREG ne contiendra aucune observation.

Pour trier les observations des tableaux permanents CODPOP et REGIME, il est nécessaire de soumettre un programme constitué de deux étapes PROC SORT. La syntaxe de la procédure SORT est la suivante :

```
PROC SORT DATA=nom du tableau à trier
          OUT=nom du tableau trié ;
          BY liste de variables de tri ;
```

Notons que le nom du tableau à trier est différent du nom du tableau trié. Ceci n'est pas une obligation : on peut trier un tableau sur lui-même, mais en cas de fin anormale de l'instruction, le tableau peut être perdu. Ainsi, pour réaliser correctement la création du tableau CODREG, il faut exécuter le programme :

```
PROC SORT DATA=BASE.CODPOP OUT=
                    CODPOP ;BY CODE ;
PROC SORT DATA=BASE.REGIME OUT=
                    REGIME ;BY CODE ;
DATA CODREG ;MERGE CODPOP REGIME ;
                    BY CODE ;
KEEP CODE POP82 NOM BUDMIL82 ;
```

Le tableau CODREG (tab. 5.9) est composé de 23 observations et des quatre variables retenues dans l'instruction KEEP ; les variables se trouvent dans l'ordre où elles étaient dans leurs tableaux initiaux (et non pas nécessairement dans l'ordre de la liste du KEEP), puis dans l'ordre des tableaux figurant dans l'instruction MERGE. Les observations sont rangées dans l'ordre alphabétique des modalités de la variable CODE. L'instruction MERGE étant très puissante, on vérifiera dans le journal de bord si le nombre d'observations dans le tableau créé est bien celui que l'on désire, et que ces observations ont bien été jointes à partir des deux tableaux (dans le cas contraire, les observations non jointes contiendraient des valeurs manquantes pour les variables provenant du tableau non utilisé).

5.3. *Ajouter de nouvelles variables en créant un nouveau tableau à partir d'un tableau existant déjà*

Pour ajouter une nouvelle variable à un tableau en cours de création, il suffit de la nommer et de lui affecter une valeur, de la manière suivante :

nom de la nouvelle variable=expression de la valeur ;

En l'absence de KEEP ou de DROP, cette nouvelle variable est ajoutée à la liste des variables du ou des tableaux existant déjà (selon que l'on ait SET ou MERGE).

L'expression figurant à droite peut être composée de constantes, d'un nom de variable, de noms de variables ou de constantes reliés par des opérateurs, de fonctions.

5.3.1. L'expression est une constante

Deux cas de figure peuvent se présenter. Si la variable est numérique, la constante est numérique ; c'est le cas le plus simple :

```
NOUVAR=10.5 ;
```

Dans le cas où la constante est alphanumérique, il faut que la nouvelle variable ait été définie comme alphanumérique ; il faut faire précéder l'affectation de la constante par la définition de la variable en nombre de caractères par l'instruction LENGTH qui a pour syntaxe :

```
LENGTH nom de la variable à définir $
       nombre de caractères ;
```

Par exemple, si la constante alphanumérique est 'PAYS', c'est-à-dire composée de quatre caractères (les quotes ne servent qu'à délimiter la constante comme précédemment), il faudra programmer :

```
LENGTH NOUVAR $ 4 ; NOUVAR='PAYS' ;
```

Affecter une constante à une variable en cours de création revient à l'initialiser : toutes les observations auront la même valeur. S'il s'agissait de copier le tableau permanent CODPOP dans un tableau temporaire CODPOP en lui ajoutant une variable NOUVAR initialisée avec la chaîne de caractères 'PAYS', il suffirait de soumettre le petit programme suivant :

```
DATA CODPOP ; SET BASE.CODPOP ;
LENGTH NOUVAR $ 4 ; NOUVAR='PAYS' ;
```

Le tableau temporaire CODPOP (tab. 5.10) aurait 23 observations et 5 variables (CODE, POP82, VAR7082, NOM, NOUVAR). Pour toutes les observations, NOUVAR contiendrait la chaîne de caractères 'PAYS'.

5.3.2. L'expression est un nom de variable

Faire figurer dans l'expression un seul nom de variable revient à faire une copie de cette variable sous un autre nom. A nouveau, deux cas peuvent se présenter. Si la variable est numérique, c'est toujours le cas le plus simple :

```
NOUVAR=POP82 ;
```

La variable POP82 doit être présente dans le tableau existant déjà ou avoir été définie plus haut dans la même étape DATA. Si la variable est alphanumérique, il faut faire précéder l'affectation par la définition de la nouvelle variable en nombre de caractères à l'aide de l'instruction LENGTH. Par exemple, s'il s'agissait de copier le tableau permanent CODPOP dans un tableau temporaire CODPOP en lui ajoutant une variable ETAT contenant les mêmes valeurs que la variable CODE (qui a une longueur de trois caractères comme l'indiquait la procédure CONTENTS), le programme prendrait la forme suivante :

```
DATA CODPOP ;SET BASE.CODPOP ;
LENGTH ETAT $ 3 ;ETAT=CODE ;
```

Le tableau temporaire CODPOP (tab. 5.11) contiendrait 23 observations et 5 variables (CODE, POP82, VAR7082, NOM, ETAT). A chaque observation, la variable ETAT aurait le même contenu que la variable CODE.

5.3.3. L'expression est une liste de noms de variables reliés par des opérateurs

Dans le cas où les variables et les constantes sont numériques, les opérateurs sont arithmétiques. On en compte 5 : l'addition (+), la soustraction (-), la multiplication (*), la division (/), l'exponentiation (**). Un simple exemple montrera l'usage pouvant être fait de ces opérateurs. Dans un tableau temporaire CODPOP, on désire avoir une variable POP1982 contenant la population exprimée en milliers d'habitants et non pas en millions d'habitants comme c'est le cas de la variable POP82 du tableau CODPOP. Le

OBS	CODE	NOM	POP82	VAR7082	NOUVAR
1	BEN	BENIN	3.6200	2.50000	PAYS
2	BWA	BOTSWANA	0.9700	3.80000	PAYS
3	HVO	BURKINA-FASO	6.3600	1.40000	PAYS
4	BDI	BURUNDI	4.2600	1.80000	PAYS
5	DJI	DJIBOUTI	0.3300	6.40000	PAYS
6	ETH	ETHIOPIE	32.7800	2.40000	PAYS
7	GMB	GAMBIE	0.6300	2.90000	PAYS
8	GIN	GUINEE	5.0600	2.10000	PAYS
9	GNB	GUINEE-BISSAU	0.8500	4.10000	PAYS
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000	PAYS
11	LSO	LESOTHO	1.4100	2.40000	PAYS
12	MWI	MALAWI	6.2700	2.60000	PAYS
13	MLI	MALI	7.3400	2.10000	PAYS
14	NER	NIGER	5.6100	2.60000	PAYS
15	UGA	OUGANDA	14.1200	3.10000	PAYS
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000	PAYS
17	TZA	TANZANIE	20.2300	3.40000	PAYS
18	RWA	RWANDA	5.5100	3.30000	PAYS
19	SLE	SIERRA LEONE	3.4100	1.60000	PAYS
20	SOM	SOMALIE	5.0800	5.10000	PAYS
21	SDN	SOUDAN	19.7900	3.00000	PAYS
22	TCD	TCHAD	4.6800	2.10000	PAYS
23	TGO	TOGO	2.6800	2.40000	PAYS

Tab. 5.10. – *Le tableau CODPOP avec variable NOUVAR.*

OBS	CODE	NOM	POP82	VAR7082	ETAT
1	BEN	BENIN	3.6200	2.50000	BEN
2	BWA	BOTSWANA	0.9700	3.80000	BWA
3	HVO	BURKINA-FASO	6.3600	1.40000	HVO
4	BDI	BURUNDI	4.2600	1.80000	BDI
5	DJI	DJIBOUTI	0.3300	6.40000	DJI
6	ETH	ETHIOPIE	32.7800	2.40000	ETH
7	GMB	GAMBIE	0.6300	2.90000	GMB
8	GIN	GUINEE	5.0600	2.10000	GIN
9	GNB	GUINEE-BISSAU	0.8500	4.10000	GNB
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000	GNO
11	LSO	LESOTHO	1.4100	2.40000	LSO
12	MWI	MALAWI	6.2700	2.60000	MWI
13	MLI	MALI	7.3400	2.10000	MLI
14	NER	NIGER	5.6100	2.60000	NER
15	UGA	OUGANDA	14.1200	3.10000	UGA
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000	CAF
17	TZA	TANZANIE	20.2300	3.40000	TZA
18	RWA	RWANDA	5.5100	3.30000	RWA
19	SLE	SIERRA LEONE	3.4100	1.60000	SLE
20	SOM	SOMALIE	5.0800	5.10000	SOM
21	SDN	SOUDAN	19.7900	3.00000	SDN
22	TCD	TCHAD	4.6800	2.10000	TCD
23	TGO	TOGO	2.6800	2.40000	TGO

Tab. 5.11. – *Le tableau CODPOP avec copie de CODE dans ETAT.*

programme suivant réalise simplement cette opération :

```
DATA CODPOP ; SET BASE.CODPOP ;
POP1982=POP82*1000 ;
```

Le tableau temporaire CODPOP (tab. 5.12) a 23 observations et 5 variables (CODE, POP82, VAR7082, NOM, POP1982) ; à chaque observation, la variable POP1982 prend la valeur de la variable POP82 multipliée par 1000.

Dans le tableau CODREG dont le mode de création est donné en 5.2., on dispose de tous les éléments pour calculer la dépense militaire par habitant (ici, en dollars par habitant). La variable DEPMIL82 contiendra ce résultat :

```
DATA CODREG ;SET CODREG ;
DEPMIL82=BUDMIL82/POP82 ;
```

Puisque BUDMIL82 est exprimé en millions de dollars et POP82 en millions d'habitants, le résultat sera bien exprimé en dollars par habitant (tab. 5.13).

Les opérateurs arithmétiques ont des niveaux de priorité en commençant par l'exponentiation, la multiplication et la division, l'addition et la soustraction. Lorsqu'une opération doit être réalisée avec les résultats d'une opération de priorité inférieure, il faut mettre cette dernière entre parenthèses. Si on désire calculer 12/11+1=1, il ne faut pas écrire 12/11+1=2.09 mais 12/(11+1). D'une manière plus générale, il est toujours préférable de mettre entre parenthèses les différents membres d'une expression arithmétique.

Il n'existe qu'un seul opérateur sur les variables et les constantes alphanumériques ; c'est l'opérateur de concaténation (!!) assemblant les chaînes en une seule. La variable créée doit avoir une longueur égale à la somme des longueurs des chaînes assemblées. Si on désire créer une variable dans un tableau temporaire CODPOP contenant le mot 'AFRIQUE' suivi du code de chaque Etat, on écrira :

```
DATA CODPOP ; SET BASE.CODPOP ;
LENGTH CONETAT $ 10 ;
CONETAT='AFRIQUE' !!CODE ;
```

La variable chaîne de caractères CONETAT a une longueur de 10 caractères car CODE en a trois et 'AFRIQUE' en a sept, soit au total 10. Le tableau créé (tab. 5.14) possède à nouveau 5 variables (CODE, POP82, VAR7082, NOM, CONETAT). A chaque observation, la variable CONETAT contient AFRIQUE suivi du code de l'Etat (par exemple AFRIQUEBEN pour BENIN). Cette opération pourrait servir à créer de nouveaux codes par continents.

5.3.4. L'expression est une fonction SAS

Une fonction SAS évalue un résultat à partir des valeurs qui lui sont données en argument, entre parenthèses derrière le nom de la fonction. Elles s'écrivent donc :

nom de fonction(arguments)

Il existe un grand nombre de fonctions se répartissant en catégories correspondant à des besoins spécifiques. Seules seront citées ici celles présentant un intérêt direct pour le traitement des variables numériques des matrices d'information spatiale ; dans tous ces cas, un seul argument est requis pouvant être une variable ou une constante, ou même une expression numérique.

a. Fonctions arithmétiques.

ABS évalue la valeur absolue
ABS(-2) retourne la valeur 2
SQRT évalue la racine carrée
SQRT(2) retourne la valeur 1.414

b. Fonction de troncature.

INT évalue la partie entière
INT(2.543) retourne la valeur 2

c. Fonctions mathématiques.

EXP évalue l'exponentielle
EXP(2) retourne la valeur 7.389
LOG évalue le logarithme naturel
LOG(2) retourne la valeur 0.693
LOG10 évalue le logarithme décimal
LOG10(2) retourne la valeur 0.301

Si l'argument est une constante, la variable ajoutée au tableau en cours de création sera initialisée par le résultat de la fonction ; si c'est une variable, chaque valeur de la variable ajoutée dépendra de la valeur de l'argument. Notons que lorsque l'évaluation du résultat n'est pas possible (cas du calcul d'une racine carrée sur une valeur négative), le résultat est la valeur manquante (.) et un message d'erreur est affiché sur le journal de bord.

Un exemple d'usage de la fonction SQRT peut être tiré de la cartographie thématique où on cherche à exprimer les quantités par des cercles de surface proportionnelle aux quantités. Le rayon de chaque cercle peut être trouvé par la formule S=3.14*(R**2) ou S est la quantité à représenter, la surface du cercle et R son rayon. Donc R=SQRT(S/3.14). Dans le cas du tableau permanent CODPOP, on peut désirer représenter la population de chaque pays par un cercle de surface proportionnelle au nombre d'habitants. Le petit programme suivant réalisera aisément cette opération dans le tableau temporaire CODPOP (tab. 5.15) :

```
DATA CODPOP ; SET BASE.CODPOP ;
RAYON=SQRT(POP82*1000000)/3.14) ;
```

La valeur de chaque rayon figurera dans la variable RAYON après multiplication de POP82 par un million, division par pi et calcul de la racine carrée. A partir de ces valeurs, on choisira une unité de mesure sur la carte pour tracer chaque cercle. Notons que dans ce cas, l'argument de la fonction SQRT est lui même une expression.

Il existe une fonction particulière, nommée SUM, qui retourne la somme des valeurs prises par une liste

OBS	CODE	NOM	POP82	VAR7082	POP1982
1	BEN	BENIN	3.6200	2.50000	3620.0
2	BWA	BOTSWANA	0.9700	3.80000	970.0
3	HVO	BURKINA-FASO	6.3600	1.40000	6360.0
4	BDI	BURUNDI	4.2600	1.80000	4260.0
5	DJI	DJIBOUTI	0.3300	6.40000	330.0
6	ETH	ETHIOPIE	32.7800	2.40000	32780.0
7	GMB	GAMBIE	0.6300	2.90000	630.0
8	GIN	GUINEE	5.0600	2.10000	5060.0
9	GNB	GUINEE-BISSAU	0.8500	4.10000	850.0
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000	370.0
11	LSO	LESOTHO	1.4100	2.40000	1410.0
12	MWI	MALAWI	6.2700	2.60000	6270.0
13	MLI	MALI	7.3400	2.10000	7340.0
14	NER	NIGER	5.6100	2.60000	5610.0
15	UGA	OUGANDA	14.1200	3.10000	14120.0
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000	2390.0
17	TZA	TANZANIE	20.2300	3.40000	20230.0
18	RWA	RWANDA	5.5100	3.30000	5510.0
19	SLE	SIERRA LEONE	3.4100	1.60000	3410.0
20	SOM	SOMALIE	5.0800	5.10000	5080.0
21	SDN	SOUDAN	19.7900	3.00000	19790.0
22	TCD	TCHAD	4.6800	2.10000	4680.0
23	TGO	TOGO	2.6800	2.40000	2680.0

Tab. 5.12. – *Le tableau CODPOP avec population en milliers dans POP1982.*

OBS	CODE	NOM	POP82	BUDMIL82	DEPMIL82
1	BDI	BURUNDI	4.2600	23	5.3991
2	BEN	BENIN	3.6200	23	6.3536
3	BWA	BOTSWANA	0.9700	29	29.8969
4	CAF	REP. CENTRAFICAINE	2.3900	12	5.0209
5	DJI	DJIBOUTI	0.3300	3	9.0909
6	ETH	ETHIOPIE	32.7800	485	14.7956
7	GIN	GUINEE	5.0600	44	8.6957
8	GMB	GAMBIE	0.6300	.	.
9	GNB	GUINEE-BISSAU	0.8500	.	.
10	GNO	GUINEE EQUATORIALE	0.3700	5	13.5135
11	HVO	BURKINA-FASO	6.3600	41	6.4465
12	LSO	LESOTHO	1.4100	.	.
13	MLI	MALI	7.3400	46	6.2670
14	MWI	MALAWI	6.2700	22	3.5088
15	NER	NIGER	5.6100	16	2.8520
16	RWA	RWANDA	5.5100	18	3.2668
17	SDN	SOUDAN	19.7900	470	23.7494
18	SLE	SIERRA LEONE	3.4100	19	5.5718
19	SOM	SOMALIE	5.0800	150	29.5276
20	TCD	TCHAD	4.6800	62	13.2479
21	TGO	TOGO	2.6800	22	8.2090
22	TZA	TANZANIE	20.2300	285	14.0880
23	UGA	OUGANDA	14.1200	852	60.3399

Tab. 5.13. – *Le tableau CODREG avec dépenses militaires par habitant.*

OBS	CODE	NOM	POP82	VAR7082	CONETAT
1	BEN	BENIN	3.6200	2.50000	AFRIQUEBEN
2	BWA	BOTSWANA	0.9700	3.80000	AFRIQUEBWA
3	HVO	BURKINA-FASO	6.3600	1.40000	AFRIQUEHVO
4	BDI	BURUNDI	4.2600	1.80000	AFRIQUEBDI
5	DJI	DJIBOUTI	0.3300	6.40000	AFRIQUEDJI
6	ETH	ETHIOPIE	32.7800	2.40000	AFRIQUEETH
7	GMB	GAMBIE	0.6300	2.90000	AFRIQUEGMB
8	GIN	GUINEE	5.0600	2.10000	AFRIQUEGIN
9	GNB	GUINEE-BISSAU	0.8500	4.10000	AFRIQUEGNB
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000	AFRIQUEGNO
11	LSO	LESOTHO	1.4100	2.40000	AFRIQUELSO
12	MWI	MALAWI	6.2700	2.60000	AFRIQUEMWI
13	MLI	MALI	7.3400	2.10000	AFRIQUEMLI
14	NER	NIGER	5.6100	2.60000	AFRIQUENER
15	UGA	OUGANDA	14.1200	3.10000	AFRIQUEUGA
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000	AFRIQUECAF
17	TZA	TANZANIE	20.2300	3.40000	AFRIQUETZA
18	RWA	RWANDA	5.5100	3.30000	AFRIQUERWA
19	SLE	SIERRA LEONE	3.4100	1.60000	AFRIQUESLE
20	SOM	SOMALIE	5.0800	5.10000	AFRIQUESOM
21	SDN	SOUDAN	19.7900	3.00000	AFRIQUESDN
22	TCD	TCHAD	4.6800	2.10000	AFRIQUETCD
23	TGO	TOGO	2.6800	2.40000	AFRIQUETGO

Tab. 5.14. – *Le tableau CODPOP avec nouvelle variable CONETAT.*

OBS	CODE	NOM	POP82	VAR7082	RAYON
1	BEN	BENIN	3.6200	2.50000	1073.72
2	BWA	BOTSWANA	0.9700	3.80000	555.80
3	HVO	BURKINA-FASO	6.3600	1.40000	1423.19
4	BDI	BURUNDI	4.2600	1.80000	1164.77
5	DJI	DJIBOUTI	0.3300	6.40000	324.18
6	ETH	ETHIOPIE	32.7800	2.40000	3231.02
7	GMB	GAMBIE	0.6300	2.90000	447.93
8	GIN	GUINEE	5.0600	2.10000	1269.43
9	GNB	GUINEE-BISSAU	0.8500	4.10000	520.29
10	GNO	GUINEE EQUATORIALE	0.3700	2.00000	343.27
11	LSO	LESOTHO	1.4100	2.40000	670.11
12	MWI	MALAWI	6.2700	2.60000	1413.09
13	MLI	MALI	7.3400	2.10000	1528.91
14	NER	NIGER	5.6100	2.60000	1336.65
15	UGA	OUGANDA	14.1200	3.10000	2120.57
16	CAF	REP. CENTRAFICAINE	2.3900	2.10000	872.44
17	TZA	TANZANIE	20.2300	3.40000	2538.24
18	RWA	RWANDA	5.5100	3.30000	1324.68
19	SLE	SIERRA LEONE	3.4100	1.60000	1042.11
20	SOM	SOMALIE	5.0800	5.10000	1271.94
21	SDN	SOUDAN	19.7900	3.00000	2510.49
22	TCD	TCHAD	4.6800	2.10000	1220.84
23	TGO	TOGO	2.6800	2.40000	923.85

Tab. 5.15. – *Le tableau CODPOP après calcul du rayon.*

de variables ; elle s'écrit :

SUM(OF liste de noms de variables)

Cette fonction est très utile au calcul de totaux en ligne. Le tableau DETTES peut donner un bon exemple d'usage de cette fonction ; les variables AIDCAD82, AIDOPE82 et AIDMAR82 constituent une ventilation des aides (à une différence près, d'origine géogaphique inconnue) selon les pays composant, d'une part, le Comité d'Aide au Développement de l'OCDE, et, d'autre part, les pays de l'OPEP et les emprunts aux conditions du marché. Il serait intéressant de connaître la proportion de chaque origine de l'aide dans le total des aides d'origines connues. Ce petit programme permet de le savoir :

```
DATA AIDES ; SET BASE.DETTES ;
AIDORG82=SUM(OF AIDCAD82 AIDOPE82
        AIDMAR82) ;
CADPCT82=AIDCAD82/AIDORG82*100 ;
OPEPCT82=AIDOPE82/AIDORG82*100 ;
MARPCT82=AIDMAR82/AIDORG82*100 ;
KEEP CODE NOM CADPCT82--MARPCT82 ;
```

Ici, AIDORG82 contient l'aide totale d'origine connue et CADPCT82, OPEPCT82, MARPCT82 la proportion de chaque aide selon l'origine connue. On notera des pourcentages négatifs de pays ayant restitué des fonds aux conditions du marché ! Le tableau temporaire AIDES contient, en définitive, 23 observations et 4 variables (CODE, CADPCT82, OPEPCT82 et MARPCT82 (tab. 5.16).

5.4. Modifier dans le tableau créé les valeurs des variables présentes dans le tableau existant déjà

Rien ne s'oppose à l'affectation d'une expression à une variable existant déjà. Cela a pour effet d'écraser les anciennes valeurs par le résultat de l'expression dans le tableau en cours de création. Dans le précédent exemple, il était totalement inutile de créer trois nouvelles variables. Le programme aurait pu avoir la forme suivante :

```
DATA AIDES ; SET BASE.DETTES ;
AIDORG82=SUM(OF AIDCAD82 AIDOPE82
        AIDMAR82) ;
AIDCAD82=AIDCAD82/AIDORG82*100 ;
AIDOPE82=AIDOPE82/AIDORG82*100 ;
AIDMAR82=AIDMAR82/AIDORG82*100 ;
KEEP CODE AIDCAD82 AIDOPE82
AIDMAR82 ;
```

Aux noms de variables près, le tableau AIDES est identique à celui défini précédemment. Attention : Les précédents libellés de variables sont conservés !

Dans bien des cas, la même expression s'applique à un ensemble de variables. Prenons le cas du tableau DETTES. Tous les montants sont exprimés en millions de dollars. On souhaiterait les voir exprimés en milliers de dollars. Il faut donc multiplier chaque variable par 1000. Le programme suivant réaliserait cette opération :

```
DATA DETTES ; SET BASE.DETES ;
AIDPUB82=AIDPUB82*1000 ;AIDMAR82=
        AIDMAR82*1000 ;
AIDOPE82=AIDOPE82*1000 ;AIDCAD82=
        AIDCAD82*1000 ;
ENCOUR82=ENCOUR82*1000 ;REGSER82=
        REGSER82*1000 ;
```

Ici, le tableau temporaire DETTES sera une copie du tableau permanent DETTES ; tous les montants seront multipliés par 1000. Outre le fait qu'elle est fastidieuse, la rédaction d'un tel programme peut être

OBS	CODE	CADPCT82	OPEPCT82	MARPCT82
1	BEN	39.895	2.3097	57.795
2	BWA	78.929	7.3917	13.679
3	HVO	78.363	4.6784	16.959
4	BDI	76.815	3.8217	19.363
5	DJI	93.515	5.9727	0.512
6	ETH	91.108	0.0456	8.846
7	GMB	95.556	10.2222	-5.778
8	GIN	70.245	6.0677	23.687
9	GNB	86.522	12.3188	1.159
10	GNQ	142.857	10.9890	-53.846
11	LSO	92.735	3.0983	4.167
12	MWI	94.475	0.0778	5.447
13	MLI	78.203	17.2803	4.516
14	NER	56.469	28.5714	14.960
15	UGA	76.242	3.4111	20.347
16	CAF	87.475	1.1834	11.341
17	TZA	87.267	5.1834	7.549
18	RWA	96.751	1.2346	2.014
19	SLE	95.621	1.6568	2.722
20	SOM	44.964	29.5407	25.496
21	SDN	68.554	22.9628	8.483
22	TCD	96.667	6.0317	-2.698
23	TGO	74.442	3.8540	21.704

Tab. 5.16. – *Le tableau AIDES.*

```
/*-------------------------------------------------------------*/
/*     PROGRAMMES DE CREATION DES TABLEAUX DU CHAPITRE 5       */
/*-------------------------------------------------------------*/

DATA CODPOP;SET BASE.CODPOP;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.1 COPIE INTEGRALE DU TABLEAU PERMANENT CODPOP;

DATA CODNOM;SET BASE.CODPOP;KEEP CODE NOM;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.2 TABLEAU CODNOM;

DATA BENIN;SET BASE.CODPOP;
IF CODE NE 'BEN' THEN DELETE;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.3 TABLEAU BENIN;

DATA VARSUP3;SET BASE.CODPOP;
IF VAR7082 LT 3 THEN DELETE;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.4 TABLEAU VARSUP3;

DATA VARPOP;SET BASE.CODPOP;
IF VAR7082 GE 3 AND POP82 GE 10;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.5 TABLEAU VARPOP;

DATA BENIN;SET BASE.CODPOP;KEEP CODE NOM;
IF CODE EQ 'BEN';
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.6 TABLEAU BENIN SANS POP82 NI VAR7082;

DATA VARSUP3;SET BASE.CODPOP;
IF VAR7082 LT 3 THEN DELETE;DROP POP82 VAR7082;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.7 TABLEAU VARSUP3 SANS POP82 NI VAR7082;

DATA CODPOP;SET BASE.CODPOP;RENAME CODE=PAYS POP82=POP1982;
PROC PRINT UNIFORM ROUND;
   TITLE TAB. 5.8 TABLEAU CODPOP APRES UN RENAME DE CODE ET DE POP1982;

PROC SORT DATA=BASE.CODPOP OUT=CODPOP;BY CODE;
PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;
DATA CODREG;MERGE CODPOP REGIME;BY CODE;
KEEP CODE POP82 NOM BUDMIL82;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.9 TABLEAU CODREG;

DATA CODPOP;SET BASE.CODPOP;
LENGTH NOUVAR $ 4;NOUVAR='PAYS';
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.10 TABLEAU CODPOP AVEC VARIABLE NOUVAR;

DATA CODPOP;SET BASE.CODPOP;
LENGTH ETAT $ 3;ETAT=CODE;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.11 TABLEAU CODPOP AVEC COPIE DE CODE DANS ETAT;

DATA CODPOP;SET BASE.CODPOP;POP1982=POP82*1000;
PROC PRINT UNIFORM ROUND;
TITLE TAB. 5.12 TABLEAU CODPOP AVEC POPULATION EN MILLIERS DANS POP1982

DATA CODREG;SET CODREG;DEPMIL82=BUDMIL82/POP82;
PROC PRINT UNIFORM ROUND;
TITLE TAB. 5.13 TABLEAU CODREG AVEC DEPENSES MILITAIRES PAR HABITANTS;

DATA CODPOP;SET BASE.CODPOP;LENGTH CONETAT $ 10;
CONETAT='AFRIQUE'||CODE;
PROC PRINT UNIFORM ROUND;
TITLE TAB. 5.14 TABLEAU CODPOP AVEC NOUVELLE VARIABLE CONETAT;

DATA CODPOP;SET BASE.CODPOP;RAYON=SQRT((POP82*1000000)/3.14);
PROC PRINT UNIFORM ROUND;
TITLE TAB. 5.15 TABLEAU CODPOP APRES CALCUL DU RAYON;

DATA AIDES;SET BASE.DETTES;AIDORG82=SUM(OF AIDCAD82 AIDOPE82 AIDMAR82);
CADPCT82=AIDCAD82/AIDORG82*100;
OPEPCT82=AIDOPE82/AIDORG82*100;
MARPCT82=AIDMAR82/AIDORG82*100;
KEEP CODE CADPCT82--MARPCT82;
PROC PRINT UNIFORM ROUND;
     TITLE TAB. 5.16 TABLEAU AIDES;
```

Fig. 5.1. – *Le programme EXPL4 : création de tableaux temporaires.*

```
 /*-------------------------------------------------------------*/
 /*      PROGRAMME DE CREATION DU TABLEAU INDIC                 */
 /*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> TRI DES 3 TABLEAUX A ASSOCIER SELON CODE */

PROC SORT DATA=BASE.CODPOP OUT=CODPOP;BY CODE;
PROC SORT DATA=BASE.DETTES OUT=DETTES;BY CODE;
PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;

 /*-----> ASSOCIER LES 3 TABLEAUX TRIES */

DATA INDIC;MERGE CODPOP DETTES REGIME;BY CODE;

 /*-----> CALCULER DES INDICATEURS */

DATA INDIC;SET INDIC;

DETHAB82=ENCOUR82/POP82;
         REGHAB82=REGSER82/POP82;
                  PUBHAB82=AIDPUB82/POP82;
                           MARHAB82=AIDMAR82/POP82;
                                    MILHAB82=BUDMIL82/POP82;

CADPUB82=AIDCAD82/AIDPUB82*100;
         OPEPUB82=AIDOPE82/AIDPUB82*100;

LABEL DETHAB82=ENCOURS DETTE $/HABITANT 82
      REGHAB82=REGLEMENTS SERVICE DETTE $/HABITANT 82
      PUBHAB82=AIDE EXT. PUBLIQUE $/HABITANT 82
      MARHAB82=AIDE EXT. MARCHE $/HABITANT 82
      MILHAB82=DEPENSES MILITAIRES $/HABITANT 82
      CADPUB82=% PAYS CAD DANS AIDE PUBLIQUE 82
      OPEPUB82=% PAYS OPEP DANS AIDE PUBLIQUE 82;

KEEP CODE NOM VAR7082 DEFCAL80 DETHAB82--OPEPUB82;

 /*-----> RANGEMENT DU TABLEAU INDIC DANS LA BASE */

DATA BASE.INDIC;SET INDIC;

 /*-----> IMPRESSION DU CONTENU DU TABLEAU INDIC */

PROC CONTENTS DATA=BASE.INDIC POSITION;

 /*-----> IMPRESSION DU TABLEAU INDIC */

PROC PRINT DATA=INDIC UNIFORM ROUND;
```

Fig. 5.2. – *Le programme EXPL5 : création du tableau INDIC.*

source d'erreurs tant sur les noms des variables que sur le multiplicateur, et cela même sur des expressions très simples. Dans ce cas, il est préférable de traiter toutes les variables à modifier d'un seul bloc ; pour ce faire, il ne faut utiliser que les trois nouvelles instructions suivantes :

ARRAY nom de bloc liste de noms de variables composant le bloc ;

pour nommer le bloc de variables à traiter et indiquer son contenu.

DO OVER nom de bloc ;

afin de préciser que le traitement doit être réalisé sur l'ensemble des variables du bloc défini dans ARRAY ;

END ;

pour limiter dans le programme l'expression ou l'ensemble d'expressions à affecter à l'ensemble des variables du bloc. Ce nouveau programme réalise exactement les mêmes calculs que le précédent :

DATA DETTES ; SET BASE.DETTES ;
ARRAY MONTANTS AIDPUB82--REGSER82 ;
DO OVER MONTANTS ; MONTANTS=
MONTANTS*1000 ; END ;

```
                                    SAS                                 1
                                                  11:05 SUNDAY, JANUARY 5, 1986

                    CONTENTS OF SAS DATA SET BASE.INDIC

TRACKS USED=1  SUBEXTENTS=1  OBSERVATIONS=23  CREATED BY OS JOB SHWANIEZ
ON CPUID 00-3033-012033  AT 11:05 SUNDAY, JANUARY 5, 1986
BY SAS RELEASE 82.4  DSNAME=GTQ4797.COURS.STAGBASE  BLKSIZE=6178
LRECL=98  OBSERVATIONS PER TRACK=441  GENERATED BY DATA

                      LIST OF VARIABLES BY POSITION

     #  VARIABLE  TYPE LENGTH POSITION  FORMAT          INFORMAT
                  LABEL

     1  CODE      CHAR      3        4
                  codes des etats iso
     2  NOM       CHAR     19        7
                  noms des etats
     3  VAR7082   NUM       8       26
                  variation population 1970-82
     4  DEFCAL80  NUM       8       34
                  deficit calorique % moy. mondiale 1980
     5  DETHAB82  NUM       8       42
                  ENCOURS DETTE $/HABITANT 82
     6  REGHAB82  NUM       8       50
                  REGLEMENTS SERVICE DETTE $/HABITANT 82
     7  PUBHAB82  NUM       8       58
                  AIDE EXT. PUBLIQUE $/HABITANT 82
     8  MARHAB82  NUM       8       66
                  AIDE EXT. MARCHE $/HABITANT 82
     9  MILHAB82  NUM       8       74
                  DEPENSES MILITAIRES $/HABITANT 82
    10  CADPUB82  NUM       8       82
                  % PAYS CAD DANS AIDE PUBLIQUE 82
    11  OPEPUB82  NUM       8       90
                  % PAYS OPEP DANS AIDE PUBLIQUE 82

                    ALPHABETIC LIST OF VARIABLES

     #  VARIABLE  TYPE LENGTH POSITION  FORMAT          INFORMAT
                  LABEL

    10  CADPUB82  NUM       8       82
                  % PAYS CAD DANS AIDE PUBLIQUE 82
     1  CODE      CHAR      3        4
                  codes des etats iso
     4  DEFCAL80  NUM       8       34
                  deficit calorique % moy. mondiale 1980
     5  DETHAB82  NUM       8       42
                  ENCOURS DETTE $/HABITANT 82
     8  MARHAB82  NUM       8       66
                  AIDE EXT. MARCHE $/HABITANT 82
     9  MILHAB82  NUM       8       74
                  DEPENSES MILITAIRES $/HABITANT 82
     2  NOM       CHAR     19        7
                  noms des etats
    11  OPEPUB82  NUM       8       90
                  % PAYS OPEP DANS AIDE PUBLIQUE 82
     7  PUBHAB82  NUM       8       58
                  AIDE EXT. PUBLIQUE $/HABITANT 82
     6  REGHAB82  NUM       8       50
                  REGLEMENTS SERVICE DETTE $/HABITANT 82
     3  VAR7082   NUM       8       26
                  variation population 1970-82
```

Fig. 5.3. – *Le résultat du PROC CONTENTS sur le tableau INDIC.*

OBS	CODE	NOM	VAR7082	DEFCAL80
1	BDI	BURUNDI	1.80000	-8
2	BEN	BENIN	2.50000	0
3	BWA	BOTSWANA	3.80000	-6
4	CAF	REP. CENTRAFICAINE	2.10000	-4
5	DJI	DJIBOUTI	6.40000	.
6	ETH	ETHIOPIE	2.40000	-26
7	GIN	GUINEE	2.10000	-16
8	GMB	GAMBIE	2.90000	-6
9	GNB	GUINEE-BISSAU	4.10000	2
10	GNQ	GUINEE EQUATORIALE	2.00000	.
11	HVO	BURKINA-FASO	1.40000	-15
12	LSO	LESOTHO	2.40000	7
13	MLI	MALI	2.10000	-15
14	MWI	MALAWI	2.60000	-4
15	NER	NIGER	2.60000	-6
16	RWA	RWANDA	3.30000	-5

OBS	DETHAB82	REGHAB82	PUBHAB82	MARHAB82
1	53.451	1.2911	30.634	7.136
2	156.464	14.3094	26.243	30.414
3	399.794	31.6495	104.742	16.598
4	90.000	1.6736	37.615	4.812
5	130.606	10.0000	176.667	0.909
6	27.383	1.8761	6.611	0.592
7	253.696	15.6719	13.162	4.012
8	240.317	11.4286	79.365	-4.127
9	149.176	4.1176	81.412	0.941
10	0.000	0.0000	37.838	-13.243
11	54.057	3.2075	33.491	6.840
12	107.943	7.9433	63.617	2.766
13	112.902	1.6349	28.188	1.253
14	124.258	1.1643	19.378	1.116
15	123.280	26.7201	45.276	7.914
16	35.354	1.0345	27.368	0.563

OBS	MILHAB82	CADPUB82	OPEPUB82
1	5.3991	92.414	4.5977
2	6.3536	80.000	4.6316
3	29.8969	91.437	8.5630
4	5.0209	98.665	1.3348
5	9.0909	93.997	6.0034
6	14.7956	92.201	0.0461
7	8.6957	90.390	7.8078
8	.	86.000	9.2000
9	.	86.272	12.2832
10	13.5135	92.857	7.1429
11	6.4465	94.366	5.6338
12	.	96.767	3.2330
13	6.2670	76.994	17.0130
14	3.5088	99.918	0.0823
15	2.8520	65.984	33.3858
16	3.2668	98.740	1.2599

OBS	CODE	NOM	VAR7082	DEFCAL80
17	SDN	SOUDAN	3.00000	1
18	SLE	SIERRA LEONE	1.60000	-8
19	SOM	SOMALIE	5.10000	-8
20	TCD	TCHAD	2.10000	-24
21	TGO	TOGO	2.40000	-8
22	TZA	TANZANIE	3.40000	-13
23	UGA	OUGANDA	3.10000	-20

OBS	DETHAB82	REGHAB82	PUBHAB82	MARHAB82
17	276.559	4.6134	37.5493	3.4664
18	93.226	12.0821	23.4018	0.6745
19	192.579	6.4173	92.1850	31.1417
20	38.098	0.2564	13.8248	-0.3632
21	302.985	13.1716	28.8060	7.9851
22	98.932	5.8774	35.1013	2.8077
23	38.683	8.7819	9.4334	2.4079

OBS	MILHAB82	CADPUB82	OPEPUB82
17	23.7494	74.606	24.9899
18	5.5718	101.253	1.7544
19	29.5276	59.577	39.1416
20	13.2479	94.127	5.8733
21	8.2090	95.078	4.9223
22	14.0880	92.466	5.4922
23	60.3399	95.646	4.2793

Tab. 5.17. – *Le tableau INDIC.*

Il est bien plus simple et satisfaisant que le précédent. Attention : le nom du bloc doit être différent des noms des variables de l'étape DATA en cours (et non pas seulement différent des noms de variables le composant) !

5.5. La création du tableau INDIC à partir des tableaux CODPOP, DETTES et REGIME

Le fichier nommé .STAGPROG(EXPL4) de la figure 5.1 est un programme SAS contenant la majeure partie des exemples du chapitre n° 5. Il est intéressant de le soumettre afin de bien se rendre compte du fonctionnement de l'étape DATA. Il suffit d'entrer les trois commandes suivantes sur la ligne de commande de l'éditeur de programme :

```
X ALLOC DA(.STAGBASE) FI(BASE) SHR
X ALLOC DA(.STAGPROG(EXPL4)) FI(PROG)
                SHR
INCLUDE PROG
```

et de le soumettre avec la touche de fonction n° 3. Ces petites illustrations des facilités de gestion des données par SAS peuvent être complétées par un exemple complet d'élaboration d'un tableau en vue de son analyse statistique. C'est l'objet du programme EXPL5 qui, à partir des tableaux permanents CODPOP, DETES et REGIME élabore un tableau permanent d'indicateurs nommé INDIC (fig. 5.2). Ce programme est composé de plusieurs étapes de traitement, elles mêmes décomposées en plusieurs étapes DATA ou PROC.

Après l'allocation de la base avec la disposition OLD, car le tableau final sera rangé dans la base, les trois tableaux permanents sont triés selon la variable d'association nommée CODE ; chaque tableau est rangé dans un tableau temporaire de même nom. Puis, dans un tableau temporaire INDIC, les trois tableaux triés sont associés selon la variable CODE. L'étape DATA suivante assure les calculs sur le tableau temporaire INDIC. Deux types d'indicateurs sont produits : d'une part des montants (dette, règlements, aide publique, aides aux conditions du marché et dépenses miliaires) en dollars par habitant et, d'autre part les proportions des pays CAD et de l'OPEP dans l'aide publique, en pourcentages. En plus de ces indicateurs, l'instruction KEEP sélectionne les variables CODE, VAR7082 et DEFCAL80. Enfin, avec une ultime étape DATA, le tableau temporaire INDIC est recopié dans la base sous le même nom et devient ainsi permanent. La procédure CONTENTS et la procédure PRINT permettent respectivement de lister le contenu (fig. 5.3) du tableau INDIC, puis de l'afficher (tab. 5.17). C'est à partir de ces indicateurs que vont pouvoir être présentées les procédures d'analyse statistique.

6. Le traitement des données avec les procédures SAS

L'étape DATA présentée dans les chapitres 5 et 6 a pour principale fonction l'organisation, le conditionnement des matrices d'information spatiale dans des tableaux composant une base SAS. Le principal intérêt de cette étape est de préparer les données de manière à faciliter leur traitement ; celui-ci est plus particulièrement assuré par l'étape PROC.

6.1. Des procédures adaptées à de nombreux besoins

Une étape PROC correspond à l'appel d'une procédure en vue de la réalisation d'un traitement particulier sur les variables d'un tableau. Il existe un très grand nombre de procédures.

a. statistique :

UNIVARIATE	analyse statistique approfondie
MEANS	statistique descriptive élémentaire
PLOT	tracé de graphiques
CORR	coefficients de corrélation
RSQUARE	coefficients de détermination
REG	régression linéaire
ADDAD	analyse des données

b. graphique :

GMAP	cartographie thématique
GREPLAY	bases de données graphiques

c. calcul matriciel :

MATRIX ou IML langage matriciel

d. procédures de service :

FSEDIT	saisie et correction d'un tableau
CONTENTS	contenu des bases et des tableaux
PRINT	impression des tableaux
SORT	tri
COPY	copie d'une base dans une autre
DELETE	suppression des tableaux

Ces procédures sont les plus couramment employées pour le traitement des matrices d'information spatiale ; il en existe beaucoup d'autres avec lesquelles le lecteur pourra faire connaissance en consultant les manuels de référence.

6.2. Structure de l'étape PROC

Une étape PROC est composée d'une ou plusieurs instructions ; la première est toujours l'instruction PROC. La syntaxe type d'une étape PROC est la suivante :

PROC nom de la procédure DATA=nom du tableau à traiter options diverses ;
VAR liste des noms des variables à traiter ;
TITLE titre du travail ;

Ces trois instructions peuvent être présentes dans la quasi-totalité des étapes PROC. On y trouve aussi très couramment trois autres instructions :

BY liste de noms de variables ;

Cette instruction permet d'itérer le traitement selon les modalités d'une ou plusieurs variables nominales ; les observations doivent avoir été préalablement triées selon leurs modalités à l'aide de la procédure SORT.

MODEL nom de la variable endogène=liste des noms des variables exogènes ;

On trouve cette instruction dans toutes les procédures de régression.

OUTPUT OUT=nom du tableau contenant les résultats résultat désiré=noms des variables les contenant ;

Cette instruction est souvent utilisée pour conserver, dans un nouveau tableau, les résultats d'une procédure.

Un mot clé indique le résultat à retenir (par exemple MIN pour le minimum) et une liste donne les noms des variables contenant les résultats. Notons qu'il existe une autre forme de récupération constituant une option de l'instruction PROC, de la manière suivante :

PROC nom de la procédure DATA=nom du tableau à traiter
OUT=nom du tableau contenant les résultats ;

6.3. Déroulement de l'étape PROC

A l'exécution d'une procédure (après l'instruction RUN, PROC ou DATA), les résultats sont affichés sur l'écran OUTPUT et peuvent être consultés à l'aide des touches de fonction n° 7 et n° 8 ; le cas échéant, les résultats désirés sont rangés dans un tableau, puis le temps d'exécution et la taille des tableaux créés sont affichés sur l'écran LOG. Les résultats apparaissent à l'écran et peuvent être transférés sur imprimante à l'aide de la touche de recopie d'écran (hardcopy) sur la gauche du clavier. Mais il peut être très utile de d'envoyer ces résultats dans un fichier temporaire et d'imprimer ce fichier par la suite.

6.3.1. Envoyer les résultats vers une imprimante

Pour pouvoir imprimer les sorties de procédures, il faut tout d'abord allouer un fichier temporaire à imprimer, puis indiquer, à l'aide de la procédure PRINTTO, que les sorties doivent être rangées dans ce fichier et, enfin, l'imprimer à l'aide de la commande tso PRINTOFF. Voici un exemple complet :

```
X ALLOC DA(PRT) FI(FT25F001) NEW
  SPACE(50) TRACKS ;
PROC PRINTTO NEX UNIT=25 ;
PROC PRINT DATA=BASE.DETTES ;
X PRINTOFF PRT DEST(LUGEOG07) ;
PROC PRINTTO ;
PROC PRINT DATA=BASE.DETTES ;
```

La commande ALLOCATE alloue le fichier temporaire PRT (et non .PRT qui serait permanent) qui a pour nom logique FT25F001 (conseillé). La procédure PRINTTO initialise ce fichier (le met à blanc) et alloue la sortie des procédures vers ce fichier. La procédure PRINT liste le tableau permanent DETTES dans le fichier PRT ; la commande PRINTOFF l'envoie vers l'imprimante connectée sur la septième prise du contrôleur LUGEOG (Maison de la Géographie de Montpellier). Puis, un autre appel de la procédure PRINTTO, sans option, rétablit la sortie vers l'écran OUTPUT et la procédure PRINT affiche le tableau permanent DETTES à l'écran.

6.3.2. Options des procédures et instructions optionnelles

Chaque procédure possède un jeu d'options qui lui est particulière. De plus, de nombreuses instructions peuvent s'ajouter à celles pésentées en 6.2. de manière à préciser le traitement à réaliser. Les chapitres suivants présenteront les options les plus courantes ; elles correspondent à l'obtention de résultats standardisés pouvant ne pas convenir à la totalité des cas de figure. Seule une bonne connaissance théorique et pratique des méthodes d'analyse et une lecture approfondie des manuels de référence permettront de pallier ces inconvénients. Enfin, notons que le déclenchement de l'exécution d'une procédure est provoqué par l'instruction RUN ;.

7. UNIVARIATE et MEANS, deux procédures de statistique descriptive

Il est facile de produire des statistiques descriptives à l'aide de SAS, mais avant d'utiliser les deux principales procédures de ce domaine, il est indispensable de bien connaître la signification des indicateurs qu'elles produisent. Il existe un excellent ouvrage d'auto-éducation en statistique publié aux Editions Economica en 1979, intitulé « Statlab » et écrit par J.L. Hodges, D. Krech et R.S. Crutchfield.

7.1. La procédure UNIVARIATE

Elle réalise une étude statistique très complète sur les variables d'un tableau : moments, quantiles, diagrammes à bâtons visualisant les distributions, graphiques d'adéquation à la loi normale. Sa syntaxe est extrêmement simple :

```
PROC UNIVARIATE DATA=nom du tableau
  PLOT NORMAL ;
ID variable identifiant les observations ;
```

Voici le programme pour le cas du tableau permanent INDIC :

```
PROC UNIVARIATE DATA=BASE.INDIC
  PLOT NORMAL ;
ID CODE ;
```

L'affichage des résultats pour la variable DEFCAL80 est donné en figure 7.1.

La sortie des résultats est organisée en quatre parties. La première contient les moments, c'est-à-dire des indicateurs de centralité, de dispersion et de forme de la distribution des valeurs de la variable. Les indicateurs les plus courants sont :

N	nombre d'observations n'ayant pas de valeur manquante
SUM	somme des valeurs
MEAN	moyenne arithmétique
VARIANCE	variance
STD DEV	standard deviation, ou écart-type
SKEWNESS	coefficient d'assymétrie, vaut 0 si la distribution est symétrique ; si > 0, elle est allongée vers les fortes valeurs, vers la droite ; si < 0, elle est allongée vers les faibles valeurs, vers la gauche.
KURTOSIS	coefficient d'aplatissement, vaut 0 si la distribution est normale ; si > 0, elle est aplatie ; si < 0, elle est mince (flat).
CV	coefficient de variation (écart-type/moyenne)
USS	uncorrected sums of squares, somme des carrés non corrigés
CSS	corrected sums of squares, somme des carrés des écarts à la moyenne arithmétique.

La seconde partie contient les quantiles correspondant à des proportions de l'effectif total des observations. A chaque pourcentage correspond une valeur. Sur la première colonne, on obtient les quartiles, le minimum, le maximum et la médiane ; sur la seconde colonne figurent les percentiles extrêmes.

La troisième partie liste les cinq observations ayant les plus petites valeurs et les cinq ayant les plus grandes. En regard de chaque valeur figure l'identifiant de l'observation donné par le paramètre ID.

Enfin, la quatrième partie présente trois représentations graphiques très utiles correspondant aux options PLOT et NORMAL de la procédure. L'option PLOT affiche un diagramme à bâtons (STEM LEAF) donnant une idée de la forme de la distribution. Noter que dans ce cas de figure, les valeurs de la colonne STEM doivent être multipliées par 10. On obtient une partition en huit classes d'étendues égales ; on connaît les valeurs de la variable dans chaque classe de la manière suivante : -2 signifie -20 (-2*10), 6 signifie qu'il faut ajouter 6, donc la première classe contient la valeur -26. Le second graphique est un diagramme en boîte, proposé par le statisticien Tuckey ; il permet de

UNIVARIATE

VARIABLE=DEFCAL80 deficit calorique % moy. mondiale 1980

MOMENTS

N	21	SUM WGTS	21
MEAN	-8.66667	SUM	-182
STD DEV	8.43998	VARIANCE	71.2333
SKEWNESS	-0.397871	KURTOSIS	-0.0831415
USS	3002	CSS	1424.67
CV	-97.3844	STD MEAN	1.84176
T:MEAN=0	-4.70566	PROB>|T|	.000135631
SGN RANK	-93	PROB>|S|	0.00054401
NUM ¬= 0	20		
W:NORMAL	0.963452	PROB<W	0.569

QUANTILES(DEF=4)

100% MAX	7	99%	7
75% Q3	-4	95%	6.49997
50% MED	-8	90%	1.79999
25% Q1	-15	10%	-23.2
0% MIN	-26	5%	-25.8
		1%	-26
RANGE	33		
Q3-Q1	11		
MODE	-8		

EXTREMES

LOWEST	ID	HIGHEST	ID
-26	(ETH)	-4	(MWI)
-24	(TCD)	0	(BEN)
-20	(UGA)	1	(SDN)
-16	(GIN)	2	(GNB)
-15	(MLI)	7	(LSO)

MISSING VALUE .
COUNT 2
% COUNT/NOBS 8.70

```
STEM LEAF                            #             BOXPLOT
   0 7                               1                |
   0 12                              2                |
  -0 440                             3             +-----+
  -0 88886665                        8             *--+--*
  -1 3                               1             +-----+
  -1 655                             3                |
  -2 40                              2                |
  -2 6                               1                |
     ----+----+----+----+
     MULTIPLY STEM.LEAF BY 10**+01
```

NORMAL PROBABILITY PLOT

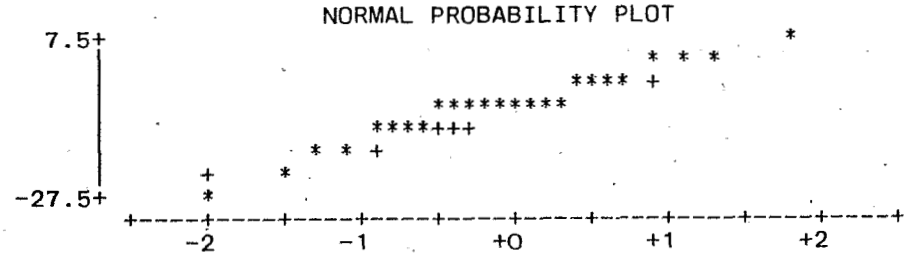


Fig. 7.1. – *Les résultats de PROC UNIVARIATE sur la variable DEFCAL80 du tableau INDIC.*

visualiser très efficacement une distribution à l'aide de ses paramètres. Les deux lignes limitées par les signes + correspondent aux premier et troisième quartiles ; la ligne limitée par les signes * est la médiane. Le signe + sur la ligne centrale indique la moyenne. Les valeurs inférieures ou supérieures aux premier ou troisième quartiles et inférieures ou supérieures à trois écarts-types sont désignées par le signe 0 sur la ligne centrale ; en deçà ou au delà, elles sont indiquées sur la ligne centrale par le signe *. La figure 7.2 présente le diagramme en boîte pour une distribution décalée vers le haut (CADPUB82), vers le bas (MILHAB82) et concentrée autour de la moyenne (MARHAB82). Le troisième graphique, correspondant à l'option NORMAL montre les distorsions de la variable par rapport à la distribution normale. Les valeurs de la première sont indiquées par le signe *, celles de la seconde par le signe +. En abscisse figure le nombre d'écarts-types en plus ou en moins de la moyenne et, en ordonnée, les valeurs extrêmes de la variable.

```
VARIABLE=CADPUB82        % PAYS CAD DANS AIDE PUBLIQUE 82

          STEM LEAF                          #          BOXPLOT
            10 01                            2             |
             9 01222344456799               14          *------*
             8 066                           3          +--+--+
             7 57                            2             |
             6 06                            2             0
             5                                             0
               ----+----+----+----+
               MULTIPLY STEM.LEAF BY 10**+01
```

```
VARIABLE=MILHAB82        DEPENSES MILITAIRES $/HABITANT 82

          STEM LEAF                          #          BOXPLOT
             6 0                             1             *
             5
             4
             3 00                            2
             2 4                             1             0
             1 3445                          4          +--+--+
             0 334556666899                 12          *-----*
               ----+----+----+----+
               MULTIPLY STEM.LEAF BY 10**+01
```

```
VARIABLE=MARHAB82        AIDE EXT. MARCHE $/HABITANT 82

          STEM LEAF                          #          BOXPLOT
             3 01                            2             *
             2
             1 7                             1             |
             0 11111112333457788            17          *--+--*
            -0 40                            2             |
            -1 3                             1             0
               ----+----+----+----+
               MULTIPLY STEM.LEAF BY 10**+01
```

Fig. 7.2. – *Des exemples de stem leaf et de boxplot sur des distributions statistiques différentes.*

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE
VAR7082	23	2.83	1.15	1.40	6.40
DEFCAL80	21	-8.67	8.44	-26.00	7.00
DETHAB82	23	134.77	101.29	0.00	399.79
REGHAB82	23	8.04	8.28	0.00	31.65
PUBHAB82	23	45.74	39.10	6.61	176.67
MARHAB82	23	5.07	9.71	-13.24	31.14
MILHAB82	20	13.49	13.71	2.85	60.34
CADPUB82	23	89.12	10.83	59.58	101.25
OPEPUB82	23	9.07	10.28	0.05	39.14

Tab. 7.1. – *Les résultats de la procédure MEANS sur le tableau INDIC.*

```
/*-------------------------------------------------------------*/
/*      PROGRAMMES D'ANALYSE STATISTIQUE DU TABLEAU INDIC      */
/*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE);

 /*-----> ANALYSE PAR PROC UNIVARIATE */

PROC UNIVARIATE DATA=BASE.INDIC PLOT NORMAL;ID CODE;
TITLE STATISTIQUES DESCRIPTIVES DU TABLEAU INDIC PAR PROC UNIVARIATE;

 /*-----> ANALYSE PAR PROC MEANS */

PROC MEANS DATA=BASE.INDIC MAXDEC=2;
TITLE STATISTIQUES DESCRIPTIVES DU TABLEAU INDIC PAR PROC MEANS;
```

Fig. 7.3. – *Le programme EXPL5 :statistiques descriptives sur le tableau INDIC.*

```
/*-------------------------------------------------------------*/
/*        AGREGATION D'OBSERVATIONS AVEC PROC MEANS             */
/*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> TRI DES TABLEAUX DETTES ET REGIME SELON CODE */

PROC SORT DATA=BASE.DETTES OUT=DETTES;BY CODE;
     PROC SORT DATA=BASE.REGIME OUT=REGIME;BY CODE;

 /*-----> ASSOCIATION DES TABLEAUX REGIME ET DETTES */

DATA AIDES;MERGE REGIME DETTES;BY CODE;
     KEEP REGPOL82 AIDPUB82--REGSER82;
PROC PRINT DATA=AIDES UNIFORM ROUND;

 /*-----> TRI DU TABLEAU AIDES SELON REGPOL82 */

PROC SORT DATA=AIDES;BY REGPOL82;

 /*-----> ELABORATION DU TABLEAU AGREGE AIDREG */

PROC MEANS DATA=AIDES NOPRINT;BY REGPOL82;
                OUTPUT OUT=AIDREG SUM=AIDPUB82 AIDCAD82 AIDOPE82
                                      AIDMAR82 ENCOUR82 REGSER82;

PROC PRINT DATA=AIDREG UNIFORM ROUND;
```

Fig. 7.4. – *Le programme EXPL6 : agrégation d'observations à l'aide de PROC MEANS.*

Proc UNIVARIATE est donc d'une grande puissance pour la description d'une variable. Elle est cependant très gourmande en ressources. La procédure MEANS doit lui être préférée lorsqu'on ne désire que les principaux paramètres d'une distribution.

7.2. La procédure MEANS

La syntaxe de la procédure MEANS est la suivante :

PROC MEANS DATA=nom du tableau à analyser
MAXDEC=nombre maximum de décimales ;

Ainsi, le programme suivant calculera la moyenne, l'écart-type, le minimum et le maximum de toutes les variables numériques du tableau permanent INDIC (tab. 7.1) :

PROC MEANS DATA=BASE.INDIC
MAXDEC=2 ;

Le programme EXPL5 contient un exemple d'utilisation de la procédure UNIVARIATE et de la procédure MEANS sur le tableau INDIC (fig. 7.3).

OBS	REGPOL82	AIDPUB82	AIDCAD82	AIDOPE82
1	MI	130.500	120.600	6.000
2	MI	95.000	76.000	4.400
3	PP	101.600	92.900	8.700
4	PP	89.900	88.700	1.200
5	NC	58.300	54.800	3.500
6	MI	216.700	199.800	0.100
7	PU	66.600	60.200	5.200
8	PP	50.000	43.000	4.600
9	MI	69.200	59.700	8.500
10	MI	14.000	13.000	1.000
11	MI	213.000	201.000	12.000
12	DE	89.700	86.800	2.900
13	MI	206.900	159.300	35.200
14	DE	121.500	121.400	0.100
15	MI	254.000	167.600	84.800
16	MI	150.800	148.900	1.900
17	PU	743.100	554.400	185.700
18	PR	79.800	80.800	1.400
19	MI	468.300	279.000	183.300
20	MI	64.700	60.900	3.800
21	MI	77.200	73.400	3.800
22	PU	710.100	656.600	39.000
23	PR	133.200	127.400	5.700

OBS	AIDMAR82	ENCOUR82	REGSER82
1	30.400	227.70	5.500
2	110.100	566.40	51.800
3	16.100	387.80	30.700
4	11.500	215.10	4.000
5	0.300	43.10	3.300
6	19.400	897.60	61.500
7	20.300	1283.70	79.300
8	-2.600	151.40	7.200
9	0.800	126.80	3.500
10	-4.900	0.00	0.000
11	43.500	343.80	20.400
12	3.900	152.20	11.200
13	9.200	828.70	12.000
14	7.000	779.10	7.300
15	44.400	691.60	149.900
16	3.100	194.80	5.700
17	68.600	5473.10	91.300
18	2.300	317.90	41.200
19	158.200	978.30	32.600
20	-1.700	178.30	1.200
21	21.400	812.00	35.300
22	56.800	2001.40	118.900
23	34.000	546.20	124.000

Tab. 7.2.a. – *Le tableau AIDES.*

OBS	REGPOL82	AIDPUB82	AIDCAD82	AIDOPE82
1	DE	211.20	208.20	3.000
2	MI	1960.30	1559.20	344.800
3	NC	58.30	54.80	3.500
4	PP	241.50	224.60	14.500
5	PR	213.00	208.20	7.100
6	PU	1519.80	1271.20	229.900

OBS	AIDMAR82	ENCOUR82	REGSER82
1	10.900	931.30	18.500
2	433.900	5846.00	379.400
3	0.300	43.10	3.300
4	25.000	754.30	41.900
5	36.300	864.10	165.200
6	145.700	8758.20	289.500

Tab. 7.2.b. – *Le tableau AIDES agrégé selon les modalités de REGPOL82.*

7.3. Agréger des observations à l'aide de la procédure MEANS

En géographie, il est courant d'avoir besoin d'agréger les observations pour produire des statistiques à un échelon supérieur à celui représenté dans la base. La procédure MEANS réalise cette opération de la manière suivante : son exécution est itérée par l'usage de l'instruction BY ; le résultat est rangé dans un tableau grâce à l'instruction OUTPUT. Le programme a la structure suivante :

PROC MEANS DATA=nom du tableau à agréger ;
BY nom de la variable d'agrégation ;
OUTPUT OUT=nom du tableau agrégé
SUM=liste des noms des variables créées ;
ID variable identifiant les observations agrégées ;

Un exemple de mise en oeuvre de cette possibilité peut être donné par le tableau permanent DETTES qu'on aimerait obtenir non par pays, mais par type de régime politique. La variable donnant cette information s'appelle REGPOL82 dans le tableau REGIME. Le programme EXPL6 (fig. 7.4) réalise cette opération ; il se déroule en plusieurs étapes : tout d'abord, les tableaux permanents DETTES et REGIME sont triés selon CODE pour être associés dans le tableau temporaire AIDES. Ensuite, ce dernier est trié selon la variable du régime politique REGPOL82 ; puis les observations agrégées sont rangées dans le tableau AIDREG qui est imprimé par la suite (tab. 7.2). Ce très simple exemple ne donne que le principe de l'agrégation des observations selon les modalités d'une variable discrète ; il existe d'autres possibilités basées sur d'autres procédures.

8. Corrélation et régression

Les techniques de corrélation et de régression linéaires sont intéressantes lors de l'étude des liens de dépendance entre les valeurs d'une variable centrale pour la recherche (dite dépendante ou endogène) et les valeurs d'une ou de plusieurs variables (dites indépendantes ou exogènes). En géographie, réaliser une étude de corrélation/régression, c'est tenter d'éclairer la répartition spatiale d'un phénomène en élaborant un modèle spécifiant une relation linéaire dont les paramètres seront estimés à partir des données. Dans les cas les plus favorables, la régression conduit à exprimer rigoureusement la forme d'une relation et son intensité. Il est rare que toutes les observations se conforment rigoureusement au modèle ; il est très utile de procéder à une étude géographique des résidus, entendus comme déviations par rapport aux valeurs estimées par le modèle ; une telle méthode de recherche met parfois en évidence des sous-espaces particuliers pour lesquels le modèle devra être complété par des variables supplémentaires qui leur sont particulières.

La plupart des ouvrages de statistique élémentaire exposent les principes et les techniques de calcul de corrélation et de régression. Pour aller plus loin, il est conseillé de consulter la troisième partie du remarquable livre de L. Lebart, A. Morineau et J.P. Fénelon : « Traitement des données statistiques », aux Editions Dunod. Un bon résumé est proposé par M. Le Guen dans « Méthodologie de la régression linéaire » publié par l'Institut Orléanais de Finance (Rue de Blois, Domaine Universitaire, 45046 Orléans CEDEX, Tél. : 38.63.22.69).

8.1. Méthodologie de la corrélation/régression

Avant de procéder à ce type d'étude, il est indispensable d'avoir une idée de la relation à explorer, c'est-à-dire de disposer d'une variable endogène et de une ou plusieurs variables exogènes, de manière à pouvoir formuler un modèle du type :

Variable endogène = a0 +
a1 * première variable exogène +
a2 * seconde variable exogène +
. . . . . .
ap * pième variable exogène

Ce qui signifie que la variation des valeurs de la variable endogène est une fonction linéaire des valeurs de p variables exogènes chacune multipliée par un coefficient (nommé coefficient de régression) et d'une constante, qui devront être estimés. Si la relation peut être considérée comme la réalisation d'un processus aléatoire (c'est-à-dire si les valeurs constituent un échantillon de l'ensemble des valeurs possibles), il sera nécessaire de procéder à des tests statistiques de validité des coefficients estimés. La méthodologie d'élaboration du modèle peut être la suivante :

a. En premier lieu, on cherchera à visualiser la relation existant entre la variable endogène et chacune des variables exogènes. Pour chacune de ces dernières, on tracera un graphique où chaque axe sera gradué en fonction des valeurs respectivement de la variable endogène et de la variable exogène. Le nuage de points représentant les observations devra avoir une forme allongée pour que la variable exogène puisse être retenue. La procédure PLOT réalisera cette opération.

b. Puis on tentera de mesurer, également pour chaque variable exogène, l'intensité de la relation qui la lie à la variable endogène, à l'aide du coefficient de corrélation linéaire de Pearson. Cette tâche sera confiée à la procédure CORR.

c. En troisième lieu, on essaiera de mesurer l'influence de l'introduction d'une variable, de deux, etc. dans le modèle. Cette influence sera mesurée par le coefficient de détermination multiple, carré du coefficient de corrélation multiple, mesurant la part de la variance de la variable endogène imputable à la

variation des variables exogènes. La procédure RSQUARE réalisera cette opération pas à pas.

d. Enfin, on procédera à l'ajustement d'une droite, d'un plan ou d'un hyperplan au nuage de points, c'est-à-dire à l'estimation des coefficients de régression. Puis, après l'estimation des valeurs données par le modèle, on calculera les résidus qui pourront être stockés dans un tableau pour faire ultérieurement l'objet d'une carte.

Le programme EXPL7 (fig. 8.1) réalise toutes ces opérations pour étudier la relation :

PUBHAB82=VAR7082 DEFCAL80

c'est-à-dire que l'intensité de l'aide publique par habitant en 1982 serait une fonction de la variation de la population de 1970 à 1982 et du déficit calorique.

8.2. Procédures d'étude de corrélation/régression

Quatre procédures sont utiles à ce type d'études : PLOT, CORR, RSQUARE et REG. Pour chacune d'elles est indiquée ci-après la syntaxe la plus courante.

8.2.1. Tracé de graphiques : procédure PLOT

La procédure PLOT trace des graphiques croisant deux variables d'un tableau dont le nom doit être donné en paramètre :

PROC PLOT DATA=nom du tableau ;

Plusieurs graphiques peuvent être tracés à l'aide de plusieurs instructions PLOT :

PLOT nom variable ordonnée * nom variable abscisse ;

Le graphique obtenu comporte deux axes gradués, orthogonaux, figurant les valeurs des deux variables. Chaque observation est représentée par la lettre A en fonction de ces valeurs sur les axes. Si deux observations sont superposées, elles sont figurées par la lettre B, etc. La figure 8.2 donne un exemple de la sortie de la procédure PLOT correspondant au programme EXPL7.

8.2.2. Coefficients de corrélation : procédure CORR

La syntaxe de la procédure CORR est extrêmement simple :

PROC CORR DATA=nom du tableau RANK ;

```
/*-------------------------------------------------------------*/
/* PROGRAMME DE CORRELATION/REGRESSION SUR DETTES ET CODPOP    */
/*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> TRACES DES GRAPHIQUES AVEC PLOT */

PROC PLOT DATA=BASE.INDIC;
          PLOT VAR7082*PUBHAB82;
               PLOT DEFCAL80*VAR7082;

 /*-----> CORRELATIONS AVEC CORR */

PROC CORR DATA=BASE.INDIC RANK;
     VAR VAR7082 DEFCAL80;WITH PUBHAB82;

 /*-----> CORRELATIONS MULTIPLES AVEC RSQUARE */

PROC RSQUARE DATA=BASE.INDIC;
     MODEL PUBHAB82=VAR7082 DEFCAL80;

 /*-----> REGRESSION AVEC REG */

PROC REG DATA=BASE.INDIC;
     MODEL PUBHAB82=VAR7082 DEFCAL80 / PARTIAL;
           OUTPUT OUT=ESTRES P=ESTIM R=RESID;

 /*-----> IMPRESSION DU TABLEAU ESTRES */

PROC PRINT DATA=ESTRES UNIFORM ROUND;
     VAR CODE ESTIM RESID PUBHAB82;
```

Fig. 8.1. – *Le programme EXPL7 : corrélations et régressions sur les tableaux DETTES et CODPOP.*

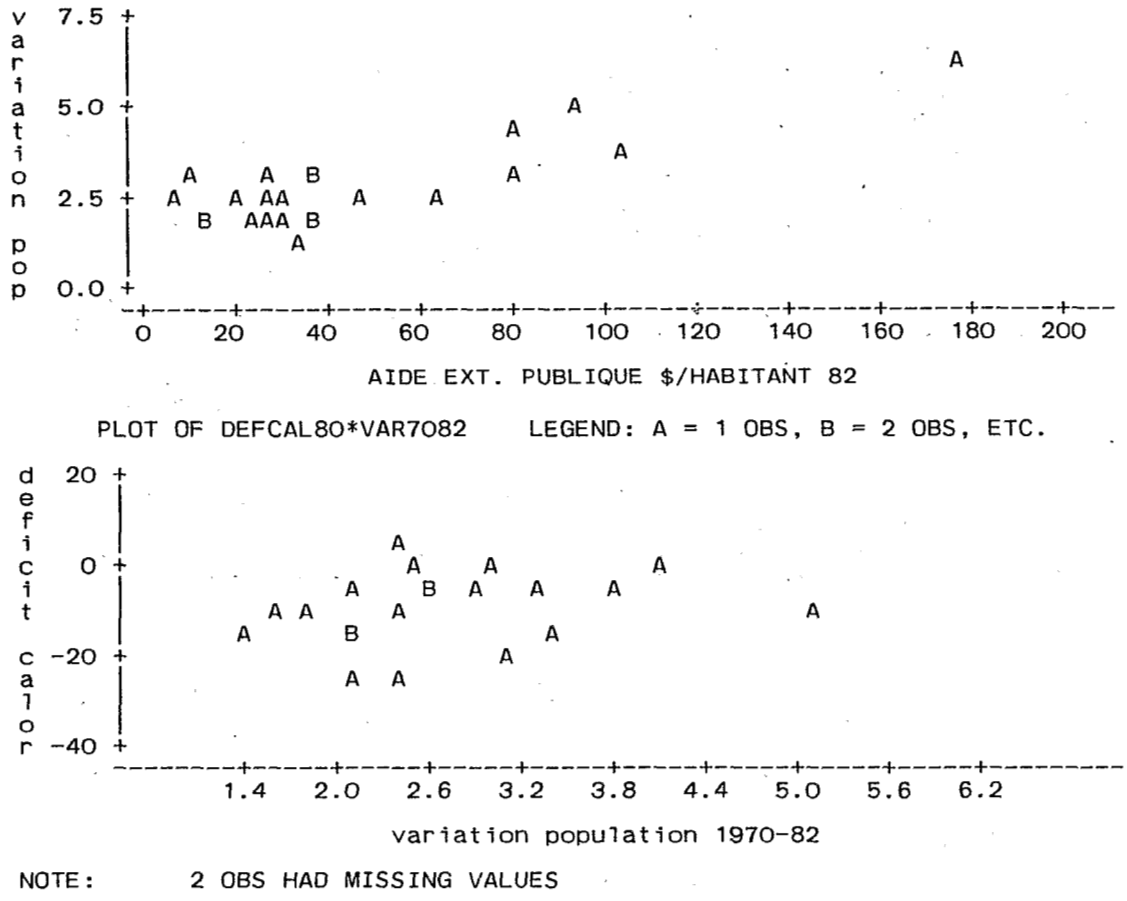


Fig. 8.2. – *Des graphiques réalisés avec PROC PLOT.*

VARIABLE	N	MEAN	STD DEV	SUM	MINIMUM	MAXIMUM
VAR7082	23	2.834783	1.154284	65.2000	1.40000	6.40000
DEFCAL80	21	-8.666667	8.439984	-182.0000	-26.00000	7.00000
PUBHAB82	23	45.735142	39.095279	1051.9083	6.61074	176.66667

CORRELATION COEFFICIENTS / PROB > |R| UNDER HO:RHO=0
/ NUMBER OF OBSERVATIONS

PUBHAB82 AIDE EXT. PUBLIQUE $/HABITANT 82

VAR7082	DEFCAL80
0.81970	0.52385
0.0001	0.0148
23	21

Fig. 8.3. – *Les statistiques descriptives et les coefficients de corrélation linéaire calculés par PROC CORR.*

L'option RANK affiche les coefficients de corrélation dans l'ordre décroissant des valeurs. La sortie de la procédure comprend d'abord des indicateurs statistiques élémentaires (nombre d'observations, moyenne, écart-type, somme, minimum, maximum (fig. 8.3).

Ensuite, les coefficients de corrélation de chaque variable avec toutes les autres (ou celles figurant dans l'instruction VAR si elle est présente) sont affichés dans l'ordre décroissant des valeurs. Pour chaque variable, on obtient le coefficient à proprement parler, le seuil de non nullité du coefficient, le nombre d'observations entrant dans le calcul (à l'exclusion de celles ayant des valeurs manquantes).

8.2.3. Recherche d'un modèle : procédure RSQUARE

La procédure RSQUARE calcule le coefficient de détermination (carré du coefficient de corrélation multiple) entre une variable endogène et une ou plusieurs variables exogènes. Elle nécessite la présence de deux instructions :

> PROC RSQUARE DATA=nom du tableau ;
> MODEL nom variable endogène=liste des noms variables exogènes ;

La figure 8.4 donne un exemple de sortie avec deux variables exogènes. Au premier pas, on calcule le coefficient de détermination entre la variable PUBHAB82 et chacune des deux variables exogènes (DEFCAL80 et VAR7082) ; au second pas, le coefficient de détermination avec les deux variables exogènes est plus élevé que ceux du premier pas. Il est donc intéressant d'intégrer ces deux variables au modèle de régression.

8.2.4. Analyse de régression : procédure REG

REG est l'une des procédures de régression multiple proposée par SAS. Elle calcule les coefficients de

```
WARNING:        2 OF THE      23 OBSERVATIONS ARE NOT INCLUDED IN
           THIS ANALYSIS DUE TO MISSING VALUES.

N=      21      REGRESSION MODELS FOR DEPENDENT VARIABLE PUBHAB82

NUMBER IN       R-SQUARE      VARIABLES IN MODEL
  MODEL

     1        0.27441367      DEFCAL80
     1        0.43109422      VAR7082
-----------------------------------
     2        0.57009776      VAR7082 DEFCAL80
-----------------------------------------------
```

Fig. 8.4. – *Les coefficients de détermination calculés par PROC RSQUARE.*

```
DEP VARIABLE: PUBHAB82 AIDE EXT. PUBLIQUE $/HABITANT 82

                        SUM OF          MEAN
SOURCE        DF       SQUARES        SQUARE       F VALUE      PROB>F

MODEL          2       8950.214      4475.107        11.935      0.0005
ERROR         18       6749.223       374.957
C TOTAL       20      15699.437

     ROOT MSE        19.363803      R-SQUARE         0.5701
     DEP MEAN        39.876369      ADJ R-SQ         0.5223
     C.V.            48.55959


                     PARAMETER      STANDARD    T FOR HO:
VARIABLE      DF      ESTIMATE         ERROR   PARAMETER=0    PROB > |T|

INTERCEP       1      2.711228     16.063092         0.169        0.8678
VAR7082        1     17.833902      5.068518         3.519        0.0025
DEFCAL80       1      1.277460      0.529521         2.412        0.0267
```

Fig. 8.5. – *Les coefficients de régression calculés par PROC REG.*

régression, réalise une analyse de la variance et peut stocker estimations et résidus dans un tableau. Elle est composée principalement des trois instructions suivantes :

PROC REG DATA=nom du tableau ;
MODEL nom variable endogène=liste variables exogènes / partial ;
OUTPUT OUT=nom tableau de stockage des résultats
P=nom de la variable des estimations (predicted values)
R=nom de la variable des résidus (residuals) ;

La figure 8.5 comprend deux parties ; au-dessus figure l'analyse de la variance. Deux indicateurs permettent de juger de la validité de la régression. Tout d'abord, la valeur F, correspondant au rapport de la variance expliquée par la régression et de la variance résiduelle, peut être testée ; PROB »F donne le seuil de non-nullité de ce rapport, ceci signifiant qu'il existe bien une différence entre ce qui est expliqué et ce qui ne l'est pas. Ensuite, R-SQUARE, le coefficient de détermination, permet de juger de l'intensité de la corrélation. La partie inférieure de la figure 8.5 comprend l'estimation des coefficients de régression, c'est-à-dire du terme constant (l'ordonnée à l'origine) et des coefficients de chaque variable exogène. Si l'option PARTIAL est spécifiée dans l'instruction MODEL, des graphiques de corrélation partielle sont obtenus permettant de juger de l'intérêt d'introduire les variables dans le modèle (fig. 8.6).

A l'issue de la régression, les estimations et les résidus sont stockés dans le tableau temporaire ESTRES qui est imprimé par la procédure PRINT (fig. 8.7) ; notons que le tableau ESTRES comprend aussi les autres variables du tableau donné dans

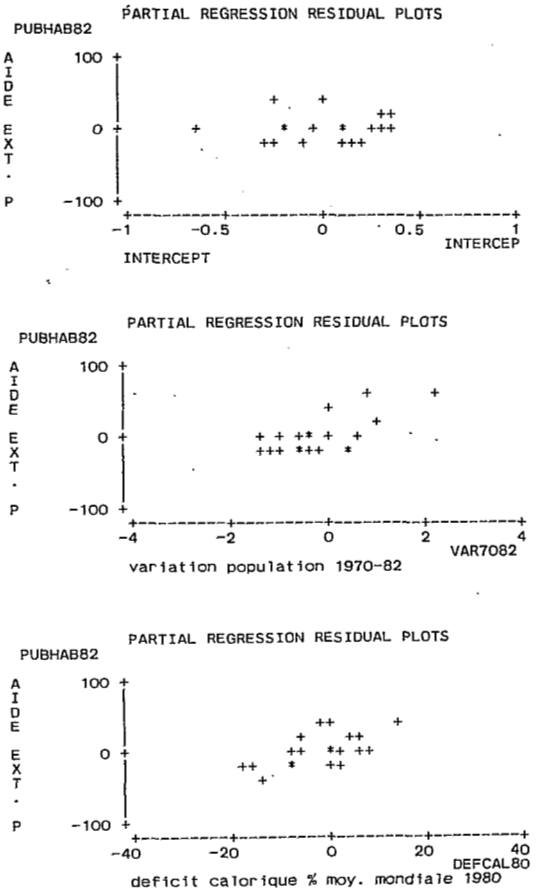


Fig. 8.6. – *Des graphiques de corrélation partielle réalisés par . PROC REG.*

OBS	CODE	ESTIM	RESID	PUBHAB82
1	BDI	24.5926	6.041	30.634
2	BEN	47.2960	-21.053	26.243
3	BWA	62.8153	41.927	104.742
4	CAF	35.0526	2.562	37.615
5	DJI	.	.	176.667
6	ETH	12.2986	-5.688	6.611
7	GIN	19.7231	-6.561	13.162
8	GMB	46.7648	32.600	79.365
9	GNB	78.3851	3.027	81.412
10	GNQ	.	.	37.838
11	HVO	8.5168	24.974	33.491
12	LSO	54.4548	9.162	63.617
13	MLI	21.0005	7.187	28.188
14	MWI	43.9695	-24.592	19.378
15	NER	41.4146	3.862	45.276
16	RWA	55.1758	-27.807	27.368
17	SDN	57.4904	-19.941	37.549
18	SLE	21.0258	2.376	23.402
19	SOM	83.4445	8.741	92.185
20	TCD	9.5034	4.321	13.8248
21	TGO	35.2929	-6.487	28.8060
22	TZA	46.7395	-11.638	35.1013
23	UGA	32.4471	-23.014	9.4334

Fig. 8.7. – *Le stockage des estimations et des résidus calculés par PROC REG.*

l'instruction PROC, notamment CODE et la variable endogène.

8.2.5. Autres procédures de régression

Mise à part la procédure REG, SAS propose trois autres procédures de régression à utiliser dans des situations particulières :

GLM (general linear model) traite la totalité des modèles linéaires à l'aide du critère des moindres carrés, c'est-à-dire la régression à proprement parler (où toutes les variables sont mesurées sur une échelle d'intervalles ou de ratios), l'analyse de la variance (où les variables exogènes sont nominales) et enfin, l'analyse de covariance. GLM accepte la formulation de modèles polynomiaux dont les termes n'ont pas à être calculés au préalable.

AUTOREG doit être utilisé lorsque les résidus sont autocorrélés, c'est-à-dire lorsque le critère Durbin-Watson est très différent de 2. AUTOREG donne des estimateurs non-biaisés des coefficients de régression et ainsi, une amélioration des estimations. Cette procédure est surtout utile pour l'analyse des séries temporelles.

Une alternative à RSQUARE est proposée par la procédure STEPWISE pour sélectionner un modèle lorsqu'on dispose d'une variable endogène et d'un jeu de variables exogènes. La sélection peut être réalisée selon cinq méthodes dont la plus courante est l'ajout de variables exogènes successives tant que croît de manière significative le coefficient de détermination.

Toutes ces procédures de régression sont bien expliquées dans le document réalisé par M. Le Guen auquel il est conseillé de se reporter.

9. Analyse des données

L'analyse des données comporte diverses techniques de statistique descriptive multidimensionnelle comprenant deux familles assez différentes. D'une part, les techniques factorielles, qui s'apparentent aux analyses factorielles des psychologues du début du siècle (comme Spearman par exemple), mettent en évidence les principales structures d'un tableau de données à l'aide de calculs d'ajustements linéaires. D'autre part, les techniques de classification automatique regroupent les variables ou les observations en classes, en fonction d'un critère de ressemblance. Ces deux familles ne sont nullement antagoniques et peuvent être utilisées conjointement : après avoir visualisé variables et observations à l'aide d'une technique factorielle, on peut chercher à regrouper ces éléments en classes homogènes. Le chercheur a dispose de toute une panoplie dont il peut user à volonté pour peu que soient respectées quelques règles élémentaires d'interprétation des résultats.

Le lecteur peu familier de l'analyse des données devrait lire in-extenso l'excellent ouvrage de J.P. Fénelon « Qu'est-ce que l'analyse des données » disponible aux Editions LEFONEN, 26, rue des Cordelières, 75013 Paris (Tel. : 43.31.17.59). Il s'agit d'une présentation « intuitive, pratique, non-mathématisée » convenant très bien à la plupart des chercheurs en sciences sociales. Pour ceux qui ont un niveau en mathématiques comparable à celui de la licence en sciences économiques, le livre de L. Lebart, A. Morineau et J.P. Fénelon déjà cité, recèle une mine d'informations qui en fait un usuel qu'il faudrait toujours avoir à portée de la main. Enfin, la lecture des Cahiers de l'Analyse des Données, publiés également par les Editions DUNOD présente toujours un grand intérêt, notamment en raison des exemples de traitements pris dans de nombreuses disciplines.

Depuis plusieurs années, les analystes de données utilisant les ordinateurs du CIRCE ou du CNUSC ont accès à la bibliothèque de l'Association pour le Développement et la Diffusion de l'Analyse des Données (ADDAD). Cette bibliothèque de programmes est d'une immense richesse tant pour la préparation des tableaux que pour leur traitement par des techniques factorielles ou des classifications automatiques. Elle correspond mieux que les procédures SAS à l'analyse des données « à la française », surtout dans le domaine de l'analyse des correspondances et dans celui des aides à l'interprétation des résultats d'analyses multivariées (notons que SAS Institute a en projet un module d'analyse des données comprenant l'AFC). La bibliothèque de l'ADDAD fonctionne avec des cartes de commande assez fastidieuses à remplir et nécessitant le recours au traitement par lot. Depuis plus d'un an, le Département de Mathématiques Appliquées du Centre d'Etudes Sociologiques (CES/DMA) du CNRS a réalisé une procédure ADDAD qui a pour fonction d'adresser les commandes nécessaires aux programmes de l'ADDAD, tout en ne quittant pas la session TSO. L'utilisateur de SAS peut ainsi bénéficier de toutes les ressources de la bibliothèque ADDAD.

9.1. Procédure ADDAD

Après l'ouverture de la session TSO, le début de la session SAS est légèrement différent car il faut entrer une commande dépendant de chaque centre informatique (après le message READY), au lieu de la commande SAS5. Ensuite, on se trouve dans une session SAS classique, mais permettant l'exécution de la procédure ADDAD. Ici ne seront présentées que les instructions communes aux principaux programmes ; il faudra y ajouter les instructions spécifiques à chaque méthode. L'appel de la procédure ADDAD se fait de la manière suivante :

```
PROC ADDAD MEMBER1=nom du programme ADDAD
DATA=nom du tableau à traiter
OUT (ou OUT1)=nom du tableau contenant les résultats
NOMISS ;
```

Le nom du programme ADDAD dépend de la méthode d'analyse choisie ; cela peut être :

ANCORR analyse factorielle des correspondances
ANCOMP analyse en composantes principales
CAH2CO classification ascendante hiérarchique
CLACAH partition après le programme CAH2CO

ou bien d'autres programmes figurant dans le manuel ADDAD.

Le nom du tableau à traiter contient les variables qui seront données en liste, plus une variable identifiant les observations. Le nom du tableau contenant les résultats peut être celui d'un tableau permanent ou temporaire. Dans le cas des programmes cités ci-dessus, il peut s'agir de coordonnées factorielles (ANCORR ou ANCOMP), de la description d'une hiérarchie (CAH2CO) ou d'une partition (CLACAH) ; à ces variables est ajoutée une variable identifiant les observations, qui est une copie sur quatre caractères de la variable identifiant les observations du tableau contenant les données à traiter. Notons que d'autres paramètres peuvent parfois figurer à l'appel de la procédure. Enfin, NOMISS signifie qu'il faut supprimer de l'analyse les observations ayant au moins une valeur manquante sur les variables entrant dans l'analyse.

A la suite de cet appel de la procédure ADDAD, on trouve toujours deux instructions complémentaires. L'instruction VAR indique quelles sont les variables entrant dans l'analyse :

VAR liste de noms de variables ;

L'instruction IDEN donne le nom de la variable identifiant les observations :

IDEN nom de la variable identifiant les observations ;

Après ces instructions figurent d'autres instructions spécifiques à chaque programme nommé dans le paramètre MEMBER1, donc dépendant de chaque technique d'analyse. Il existe de plus une instruction TITRE permettant d'afficher le titre de l'analyse :

TITRE titre de l'analyse ;

Voici quelques exemples de ces trois instructions :

```
PROC ADDAD MEMBER1=ANCOMP
DATA=BASE.INDIC OUT1=FACOMP
NOMISS ;
VAR VAR7082--OPEPUB82 ;
IDEN CODE ;
TITRE ACP DU TABLEAU INDIC ;
```

Ici, l'ensemble des variables numériques du tableau permanent INDIC fait l'objet d'une analyse en composantes principales. Les coordonnées des observations sur ces composantes sont rangées dans le tableau temporaire FACOMP qui aura NF compsantes principales (voir plus loin la présentation du paramètre NF) nommées __ F1, __ F2 ... __ FNF, et une variable identifiant, copie de la variable CODE, nommée __ NOM.

```
PROC ADDAD MEMBER1=CAH2CO
DATA=BASE.INDIC OUT=DESCAH
NOMISS ;
VAR VAR7082--OPEPUB82 ;
IDEN CODE ;
TITRE CAH SUR LE TABLEAU INDIC ;
```

Dans ce cas, l'ensemble des valeurs numériques du tableau INDIC fait l'objet d'une classification ascendante hiérarchique portant sur toutes les variables, avec comme critère d'agrégation des observations le moment centré d'ordre deux d'une partition. Le tableau DESCAH en sortie contient la description de la hiérarchie pouvant être utilisée par la suite pour calculer des partitions.

9.2. *Analyses factorielles*

Les techniques d'analyse factorielle peuvent être envisagées de plusieurs points de vue. Pour une majorité de statisticiens, il ne s'agit que de représenter les positions relatives des observations ou des variables dans un espace mathématique à une ou deux dimensions, alors que l'espace initial est multidimensionnel. Pour L. Lebart, les méthodes factorielles « utilisent des calculs d'ajustement qui font essentiellement appel à l'algèbre linéaire et produisent des représentations graphiques où les objets à décrire deviennent des points sur un axe ou dans un plan ». J.M. Bouroche exprime un point de vue semblable : « les méthodes d'analyse des données permettent une étude globale des variables (...). Pour cela, on plonge individus et variables dans des espaces géométriques, tout en faisant la plus grande économie d'hypothèses, et on transforme les données pour les visualiser dans un plan (...) ». Ici, l'analyse factorielle est avant tout un processus géométrique de compression d'un espace mathématique.

Un point de vue un peu différent est parfois exprimé par les utilisateurs finaux. Pour Migneron, « l'analyse factorielle est une technique d'analyse des données chiffrées fournissant une description condensée de variables associées ». L'image du tamis statistique exprimée par J.B. Racine est très heureuse : « l'analyse factorielle n'est rien d'autre qu'un procédé d'induction quantitative, un tamis ou un filtre d'une réalité trop complexe pour être appréhendée de façon directe par l'observateur ».

Un troisième point de vue, plus riche mais aussi plus difficile à justifier, est celui des théoriciens qui voient en l'analyse factorielle un moyen de vérifier leurs hypothèses : il s'agit alors d'assimiler les facteurs produits par l'analyse à des facteurs fondamentaux : « l'analyse factorielle permet, à partir d'un ensemble d'éléments atomisés, de dégager des régularités théoriques d'un ensemble parfois important de prédicteurs. Ceci repose de toute évidence sur le postulat que de telles régularités ne sont pas qu'une pure abstraction, mais qu'elles sont le reflet véritable des déterminants du comportement de la réalité étudiée ». P. Claval exprime ce point de vue d'une autre manière :

« l'analyse factorielle dégage, derrière le foisonnement des données, les quelques facteurs qu'elles représentent et qui expliquent, au sens statistique du terme, l'essentiel de la variation analysée ».

En fait, ces trois points de vue n'apparaissent en aucun cas incompatibles ; ils correspondent simplement à des niveaux d'interprétation différents car, en matière d'analyse factorielle, l'intuition dans l'interprétation est guidée par des indicateurs quantitatifs rigoureux. C'est cette complémentarité du qualitatif au quantitatif qui fait toute la richesse de ces techniques. Dans le cadre de cet ouvrage, seuls les programmes d'analyse en composantes principales (ACP) et d'analyse factorielle des correspondances (AFC) peuvent être présentés. Notons qu'il existe deux procédures SAS d'ACP : la procédure FACTOR proposant une grande variété d'options de rotation est la plus complète ; elle doit être suivie par la procédure SCORE pour le calcul des coordonnées des observations sur les composantes principales. La procédure PRINCOMP est moins complète, mais plus facile à utiliser.

Les lecteurs ayant un bon niveau en mathématiques auraient avantage à lire un très bon ouvrage de J.P. et F. Benzécri, publié en 1980 aux Editions DUNOD et intitulé « Pratique de l'analyse des données », tome 1 : Analyse des correspondances, exposé élémentaire« . On y trouve un exposé mathématique, un exemple numérique détaillé et surtout deux parties fort intéressantes, l'une consacrée à la lecture des listages informatiques, l'autre à des exemples variés d'applications. Ce livre présente également plusieurs solutions de codages préalable de l'information qu'il n'est pas possible d'aborder ici.

Qu'il s'agisse du programme ANCORR ou du programme ANCOMP, il est nécessaire de compléter les instructions communes à tous les programmes par des instructions particulières à ces deux techniques ; l'instruction PARAM spécifie les paramètres de chaque analyse, c'est-à-dire le nombre de facteurs à calculer (7 au maximum), s'il faut afficher le tableau des facteurs sur l'ensemble des observations (ensemble I) et sur l'ensemble des variables (ensemble J) :

PARAM NF=3 IMPFI IMPFJ ;

signifie qu'il faut afficher le tableau des trois premiers facteurs sur les variables et les observations.

On fait appel à l'instruction GRAPHE, de manière optionnelle, pour formuler la demande d'un graphique. Il peut y avoir au plus 10 instructions GRAPHE composée des paramètres suivants :

GRAPHE X=rang du facteur en abscisse
Y=rang du facteur en ordonnée
IP (ou JP, ou IP JP) ;

IP signifie qu'il s'agit du graphique des observations, JP du graphique des variables. En analyse des correspondances, IP et JP peuvent être présents en même temps ; on obtient alors une représentation simultanée des deux ensembles :

GRAPHE X=1 Y=2 IP ;

spécifie une demande de graphique du premier plan factoriel avec une représentation des observations uniquement par leurs coordonnées sur les deux premiers facteurs.

En regroupant les observations communes à tous les programmes avec celles particulières aux méthodes factorielles, on peut aisément réaliser une analyse en composantes principales. Le programme EXPL8 (fig. 9.1) donne un bon exemple. Il s'agit de l'analyse en composantes principales du tableau permanent INDIC. Les coordonnées des observations sur les NF facteurs demandés sont stockées dans le tableau temporaire FACINDIC nommé par le paramètre OUT1 (ce serait OUT2 pour les coordonnées des variables). Ce programme demande de plus le graphique des observations dans le premier plan factoriel, ainsi que celui des variables. La procédure PRINT imprime ensuite le tableau FACINDIC. Notons également que l'usage du programme ANCORR est tout à fait semblable, et que l'instruction :

VARSUP liste de noms de variables ;

permet de tester la position des variables supplémentaires dans l'espace factoriel.

La sortie des résultats est organisée de la manière suivante :

a. Une première partie (fig. 9.2) présente la traduction des instructions SAS en cartes de commanes ADDAD ; on y trouve en particulier le titre de l'analyse et les noms des variables tronqués à quatre caractères.

b. Après diagonalisation de la matrice des coefficients de corrélation de Pearson, on obtient l'histogramme des valeurs propres (fig. 9.3). La colonne NUM contient le rang des facteurs. Chaque composante (une par ligne) est représentée par sa valeur propre (ou sa variance), par le rapport valeur propre/variance totale, ainsi que par les cumuls des ces rapports sur les composantes successives. L'histogramme autorise la formulation d'un premier jugement sur le nombre de composantes à retenir.

c. La figure 9.4 présente les coordonnées des observations (ou factor scores) sur les trois premières composantes principales. Les cinq premières colonnes de nombres contiennent, dans l'ordre, le numéro d'observation (I1), la qualité des projections sur les trois composantes (QLT), le poids de chaque observation (POID) et sa contribution à la variance totale du nuage. Ensuite, trois colonnes par composante donnent les coordonnées (1*F), la qualité de projection (COR) et la contribution à la variance de chaque composante. Le même tableau est affiché pour l'ensemble J1 des variables (fig. 9.5).

d. L'instruction GRAPHE commande un graphique figurant les coordonnées des variables ou des observations (en fonction de la présence de JP ou de IP) dans l'instruction GRAPHE. Les figures 9.6.a et 9.6.b proposent ces graphiques pour le premier plan factoriel, celui des composantes n° 1 et n° 2.

```
/*-------------------------------------------------------------*/
/*          PROGRAMME DE D'ACP SUR LE TABLEAU INDIC            */
/*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> ACP PAR LE PROGRAMME ANCOMP */

PROC ADDAD MEMBER1=ANCOMP DATA=BASE.INDIC
     OUT1=FACINDIC NOMISS;
          VAR VAR7082--OPEPUB82;
              IDEN CODE;
                   PARAM IMPFI IMPFJ NF=3;
                         GRAPHE X=1 Y=2 IP;
                         GRAPHE X=1 Y=2 JP;
                               GRAPHE X=1 Y=3 IP;
                               GRAPHE X=1 Y=3 JP;
TITRE ACP DU TABLEAU INDIC;

 /*-----> IMPRESSION DU TABLEAU FACINDIC */

PROC PRINT DATA=FACINDIC UNIFORM ROUND;
```

Fig. 9.1. – *Le programme EXPL8 : ACP du tableau INDIC.*

```
B I B L I O T H E Q U E   A D D A D  - VERS. 83 -

ANALYSE EN COMPOSANTES PRINCIPALES (ANCOMP - 202)
D'APRES : YAGOLNITZER ET TABET

CARTE  1 - TITRE DE L'ANALYSE
           ACP DU TABLEAU INDIC  S

CARTE  2 - PARAMATRES GENERAUX : NI,NJ,NF,NI2,NJ2,IPOND,LECIJ,STFI,STFJ
              18     9     3     0     0     0     1     1     0

CARTE  3 - OPTIONS : IOUT,IVP,IF,JF,IGR
                0     0     1     1     1

CARTE  5 - DEMANDE DE GRAPHIQUES :
           12100431001201043100131004310013010431000000000000000000000000000000000000000000000000

CARTE  6 - NOMS DES VARIABLES :
           VAR7 DEFC DETH REGH PUBH MARH MILH CADP OPEP

CARTE  7 - FORMAT DES DONNEES
           (20A4)
LES POIDS DES  COLONNES SONT MULTIPLIES PAR  10 **  0
```

NOMJ(J)	VAR7	DEFC	DETH	REGH	PUBH	MARH	MILH	CADP	OPEP
PJ(J)	47	-185	2472	151	613	129	247	1594	171	18

Fig. 9.2. – *Les paramètres de l'ACP.*

LES VALEURS PROPRES VAL(1)= 4.17870

NUM	ITER	VAL PROPRE	POURCENT	CUMUL	*	HISTOGRAMME DES VALEURS PROPRES DE LA MATRICE
1	0	4.17870	46.430	46.430	*	********** | ********** | ********** **********
2	1	1.46147	16.239	62.669	*	********** | ******
3	2	1.21464	13.496	76.165	*	********** | **
4	2	0.77097	8.566	84.731	*	**********
5	2	0.49677	5.520	90.251	*	*******
6	3	0.37283	4.143	94.393	*	*****
7	4	0.28724	3.192	97.585	*	****
8	2	0.19414	2.157	99.742	*	***
9	1	0.02321	0.258	100.000	*	

Fig. 9.3. – *L'histogramme des valeurs propres.*

e. Le tableau 9.1 montre le contenu du tableau temporaire FACINDIC, nommé par le paramètre OUT1. Notons que les coordonnées des observations se nomment __ F1, __ F2 et __ F3.

9.3. Classification automatique

Les techniques de classification forment l'autre branche de l'analyse des données. Elles permettent de rassembler les observations en classes homogènes. P. Dumolard a recensé un grand nombre d'algorithmes et montre qu'en ce domaine l'optimalité est irréelle et impossible : « Pour partitionner un nuage de points, par exemple, il n'existe pas une seule façon de procéder quelle que soit la configuration. Parmi les façons de partager, il n'en existe pas de meilleure a priori, indépendante de toute contingence, encore moins d'optimale, quel que soit le cas de figure ». Les principaux procédés se distinguent par la progression des algorithmes (ascendants ou descendants), par la construction de hiérarchies ou de partitions, et enfin par le nombre de classes connu ou inconnu à priori. Le chercheur se retrouve devant un difficile problème de choix d'une technique. En effet, il ne suffit pas de procéder à une seule classification, il faut aussi valider statistiquement les classes obtenues.

Aujourd'hui, les divers algorithmes connus sous le nom de classification ascendante hiérarchique (CAH) présentent une grande fiabilité : ils ont fait l'objet de nombreuses applications qui ont démontré leur capacité à résoudre une grande variété de problèmes. C'est pour cette raison que l'usage de la classification ascendante hiérarchique sera particulièrement détaillé ci-après, à partir d'un seul des programmes de la bibliothèque ADDAD ; notons que cette dernière propose également un programme de nuées dynamiques traitant très efficacement plusieurs milliers d'observations. SAS possède également une procédure nommée CLUSTER remaniée, pouvant fort bien faire l'affaire dans de nombreux cas. Le lecteur désireux de compléter sa connaissance des techniques de classification doit nécessairement disposer de la remarquable somme de M. Jambu et M.O. Lebeaux intitulée « Classification automatique pour l'analyse des données » publiée en deux tomes au Editions DUNOD, en 1978.

9.3.1. Le programme CAH2CO

De même que les programmes d'analyse factorielle, celui de classification ascendante hiérarchique, CAH2CO nécessite la présence de quelques instructions complémentaires. L'instruction PARAM spécifie

	I1	QLT	POID	INR	1#F	COR	CTR	2#F	COR	CTR	3#F	COR	CTR
1	BDI	764	56	18	1218	496	20	-488	79	9	752	189	26
2	BEN	425	56	61	-1364	190	25	-1391	197	74	613	38	17
3	BWA	962	56	159	-3846	573	197	-1055	43	42	-2993	347	410
4	CAF	627	56	22	1125	350	17	-975	262	36	231	15	2
5	ETH	671	56	53	2306	614	71	1488	256	84	-114	1	1
6	GIN	251	56	24	442	50	3	-584	88	13	-662	113	20
7	HVO	633	56	25	1522	570	31	-181	8	1	474	55	10
8	MLI	641	56	22	477	63	3	510	72	10	1349	505	83
9	MWI	627	56	27	1375	435	25	-912	191	32	67	1	0
10	NER	562	56	85	-2293	384	70	-477	17	9	1490	162	102
11	RWA	317	56	32	1213	284	20	-276	15	3	310	19	4
12	SDN	432	56	52	-1804	385	43	-19	0	0	624	46	18
13	SLE	872	56	28	1577	546	33	-1170	301	52	-335	25	5
14	SOM	954	56	217	-5252	784	367	2094	125	167	1259	45	72
15	TCD	893	56	44	2190	671	64	1251	219	60	158	4	1
16	TGO	721	56	23	-184	9	0	-1335	486	68	-909	225	38
17	TZA	356	56	10	382	91	2	580	209	13	-300	56	4
18	UGA	862	56	97	915	53	11	2940	550	329	-2017	259	186
				1000			1000			1000			1000

Fig. 9.4. – *Les coordonnées des observations sur les trois premières composantes.*

	J1	QLT	POID	INR	1#F	COR	CTR	2#F	COR	CTR	3#F	COR	CTR
1	VAR7	735	56	111	-745	555	133	399	159	109	-144	21	17
2	DEFC	683	56	111	-481	232	55	-646	417	285	184	34	28
3	DETH	775	56	111	-679	462	110	-393	155	106	-398	159	130
4	REGH	653	56	111	-595	355	85	-336	113	77	-431	186	153
5	PUBH	754	56	111	-854	730	175	-79	6	4	-133	18	15
6	MARH	603	56	111	-768	589	141	-65	4	3	97	9	8
7	MILH	873	56	111	-309	95	23	693	480	328	-545	298	245
8	CADP	911	56	111	756	572	137	-243	59	40	-529	280	231
9	OPEP	868	56	111	-768	590	141	261	68	47	459	211	173
				1000			1000			1000			1000

Fig. 9.5. – *Les coefficients de corrélation entre les variables et les composantes principales.*

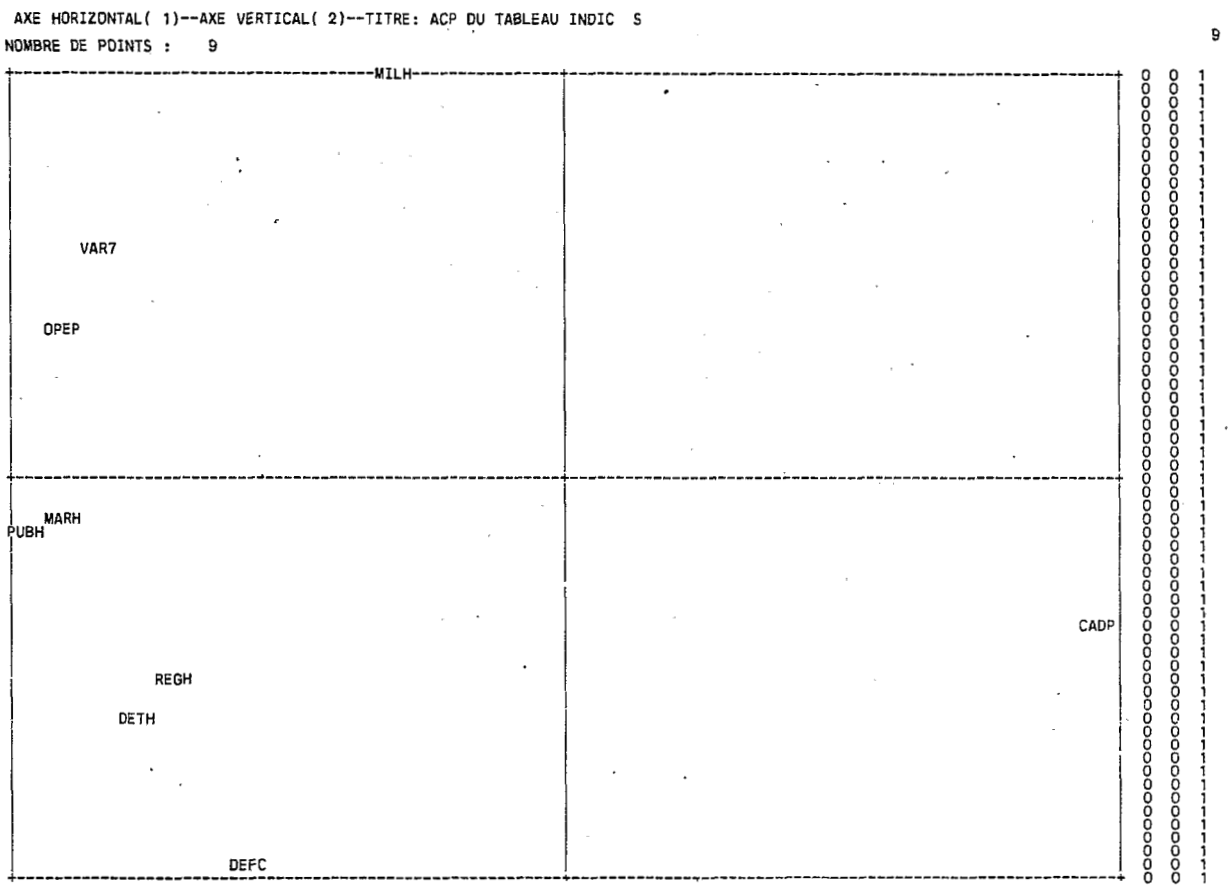


Fig. 9.6.a. – *Les variables dans le plan des deux premières composantes principales.*

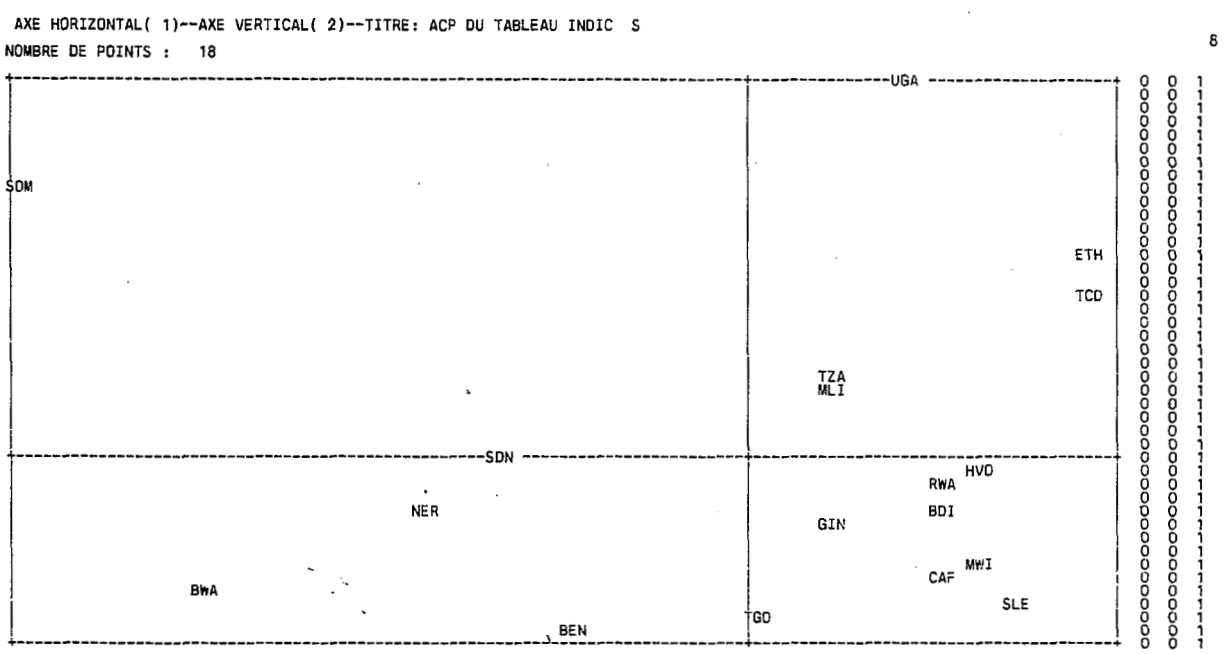


Fig. 9.6.b. – *Les observations dans le plan des deux premières composantes principales.*

les paramètres de chaque analyse, c'est-à-dire s'il faut afficher l'histogramme des indices de niveau (HISTO), la description de la hiérarchie (DESCRI) et l'arbre représentant le dendrogramme (ARBRE) ; il faut y ajouter une option précisant si le tableau des données est un tableau de facteurs (IOPT=1, la métrique euclidienne usuelle est choisie et les variables conservent leur variance), ou s'il s'agit d'un tableau de correspondance (IOPT=2, les distances sont calculées avec la métrique du chi-deux), ou enfin d'un tableau de mesures (IOPT=3, la métrique euclidienne usuelle est précédée du centrage et de la réduction des variables) :

PARAM IOPT=3 HISTO DESCRI ARBRE ;

signifie que l'analyse sera réalisée sur un tableau de mesures avec affichage de l'histogramme des indices de niveaux, de la description de la hiérarchie et de l'arbre.

En regroupant à nouveau les instructions communes à tous les programmes avec l'instruction PARAM, on obtient le programme EXPL9 (fig. 9.7). Il s'agit d'une classification ascendante hiérarchique d'après le critère du moment centré d'ordre deux. La description de la hiérarchie est stockée dans le tableau temporaire CAHINDIC. La sortie des résultats est organisée de la manière suivante :

a. La première partie de la sortie présente les commandes adressées à ADDAD (fig. 9.8) comme pour l'ACP. Outre le titre et les noms des variables, on y trouve les paramètres de l'analyse et les options de sortie.

b. L'histogramme des indices de niveaux de la hiérarchie (fig. 9.9) permet de juger du nombre de classes à retenir. Les colonnes représentant dans l'ordre : le numéro du noeud (J), l'inertie du noeud (I(J)), l'aîné et le benjamin (A(J) et B(J)), qu'il s'agisse d'un noeud ou d'une observation, le taux d'inertie du noeud par rapport à la variance totale du nuage (T(J)) ainsi que les taux cumulés (T(Q)).

c. A l'histogramme des indices de niveau doit être associée la description des classes successives, à l'aide des noms des observations (fig. 9.10).

d. L'arbre de la hiérarchie (fig. 9.11) visualise cette hiérarchie et donne ainsi un élément de plus de choix du nombre de classes.

```
 /*-----> ALLOCATION DE LA BASE */
X ALLOC DA(.STAGBASE) FI(BASE) OLD;
 /*-----> CAH PAR LE PROGRAMME CAH2CO */
PROC ADDAD MEMBER1=CAH2CO DATA=BASE.INDIC
     OUT1=CAHINDIC NOMISS;
          VAR VAR7082--OPEPUB82;
              IDEN CODE;
                   PARAM IOPT=3 HISTO DESCRI ARBRE;
TITRE CAH DU TABLEAU INDIC;
```

Fig. 9.7. – *Le programme EXPL9 : CAH sur le tableau INDIC.*

```
B I B L I O T H E Q U E   A D D A D  - VERS. 83 -

CLASSIFICATION ASCENDANTE HIERARCHIQUE (CAH2CO - 203)
AUTEUR : M.JAMBU

CARTE  1 - TITRE DE L'ANALYSE
           CAH DU TABLEAU INDIC  S
CARTE  2 - PARAMETRES GENERAUX : NI,NJ,IOPT,NPLACE,LECIJ,STCAH
                18     9     3    36     1     1
CARTE  3 - OPTIONS : HISTO,DESCRI,ARBRE
                1     1     1
CARTE  4 - NOMS DES VARIABLES :
           VAR7 DEFC DETH REGH PUBH MARH MILH CADP OPEP
CARTE  5 - FORMAT DES DONNEES :
           (20A4)
```

Fig. 9.8. – *Les paramètres de la CAH.*

SOMME DES INDICES DE NIVEAU 0.90000E+01

J	I(J)	A(J)	B(J)	T(J)	T(Q)	HISTOGRAMME DES INDICES DE NIVEAU DE LA HIERARCHIE
35	2870	32	34	319	319	**********
34	1433	31	33	159	478	**********
33	890	3	14	99	577	**********
32	857	30	18	95	672	**********
31	621	28	29	69	741	**********
30	547	27	26	61	802	**********
29	455	2	22	51	853	**********
28	365	10	12	41	893	**********
27	268	24	25	30	923	**********
26	249	19	8	28	951	**********
25	121	17	21	13	964	**********
24	94	23	13	10	975	**********
23	74	20	7	8	983	**********
22	66	6	16	7	990	**********
21	43	9	11	5	995	**********
20	29	1	4	3	998	**********
19	18	5	15	2	1000	**********

Fig. 9.9. – *L'histogramme des indices de niveau.*

J	I(J)	A(J)	B(J)	P(J)	DESCRIPTION DES CLASSES DE LA HIERARCHIE
35	2870	32	34	18	
34	1433	31	33	7	NER SDN BEN GIN TGO BWA SOM
33	890	3	14	2	BWA SOM
32	857	30	18	11	BDI CAF HVO SLE TZA MWI RWA ETH TCD MLI UGA
31	621	28	29	5	NER SDN BEN GIN TGO
30	547	27	26	10	BDI CAF HVO SLE TZA MWI RWA ETH TCD MLI
29	455	2	22	3	BEN GIN TGO
28	365	10	12	2	NER SDN
27	268	24	25	7	BDI CAF HVO SLE TZA MWI RWA
26	249	19	8	3	ETH TCD MLI
25	121	17	21	3	TZA MWI RWA
24	94	23	13	4	BDI CAF HVO SLE
23	74	20	7	3	BDI CAF HVO
22	66	6	16	2	GIN TGO
21	43	9	11	2	MWI RWA
20	29	1	4	2	BDI CAF
19	18	5	15	2	ETH TCD

Fig. 9.10. – *La description de la hiérarchie.*

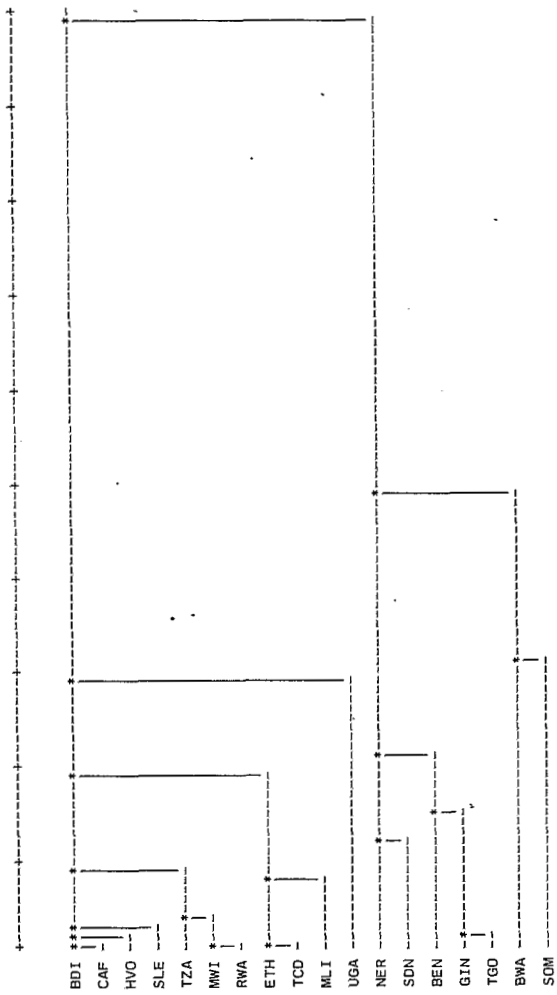


Fig. 9.11. – *L'arbre représentant la hiérarchie.*

```
 /*-----> ALLOCATION DE LA BASE */
X ALLOC DA(.STAGBASE) FI(BASE) OLD;
 /*-----> SUPPRESSION DONNEES MANQUANTES */
DATA INDIC;SET BASE.INDIC;
     ARRAY MANQ VAR7O82--OPEPUB82;
           DO OVER MANQ;IF MANQ EQ . THEN DELETE;END;
 /*-----> CAH PAR LE PROGRAMME CAH2CO */
PROC ADDAD MEMBER1=CAH2CO DATA=INDIC
     OUT1=CAHINDIC NOMISS;
          VAR VAR7O82--OPEPUB82;
              IDEN CODE;
                   PARAM IOPT=3 HISTO DESCRI ARBRE;
TITRE CAH DU TABLEAU INDIC;
 /*-----> PARTITION AVEC LE PROGRAMME CAH2CO */
PROC ADDAD MEMBER1=CLACAH DATA=INDIC
     DATACAH=CAHINDIC
             OUT=CAHPART
                 NOMISS;
                    IDEN CODE;
                         PARAM NP=3;
PARMCARDS;
2 3 4
;
TITRE CAH DU TABLEAU INDIC;
 /*-----> RANGEMENT DU TABLEAU CAHPART DANS LA BASE */
DATA CAHPART;SET CAHPART;
     LENGTH CODE $ 3;CODE=LEFT(_NOM);
            DROP _NOM;
PROC SORT DATA=CAHPART OUT=BASE.CAHPART;BY CODE;
PROC PRINT DATA=BASE.CAHPART UNIFORM ROUND;
 /*-----> CALCUL DES STATISTIQUES PAR CLASSE */
DATA STATCLAS;MERGE BASE.INDIC BASE.CAHPART;BY CODE;
     ARRAY MANQ _P1-_P3;
           DO OVER MANQ;IF MANQ EQ . THEN DELETE;END;
PROC SORT DATA=STATCLAS;BY _P1;
     PROC MEANS MAXDEC=2;BY _P1;
          VAR VAR7O82--OPEPUB82;
PROC SORT DATA=STATCLAS;BY _P2;
     PROC MEANS MAXDEC=2;BY _P2;
          VAR VAR7O82--OPEPUB82;
PROC SORT DATA=STATCLAS;BY _P3;
     PROC MEANS MAXDEC=2;BY _P3;
          VAR VAR7O82--OPEPUB82;
```

Fig. 9.12. – *Le programme EXPL10 : CAH et statistiques par classes sur le tableau INDIC.*

9.3.2. Le programme CLACAH

Il s'agit d'un complément normal du précédent : à partir de la description de la hiérarchie, il produit dans un tableau une ou plusieurs partitions au nombre de classes fixé a priori. Il réalise donc une ou plusieurs coupures du dendrogramme.

L'instruction PARAM fixe le nombre de partitions à élaborer :

PARAM NP=nombre de partitions ;

Puis il est nécessaire de fixer le nombre de classes de chaque partition de la manière suivante :

PARMCARDS ;
liste de nombres de classes
;

Les nombres de classes par partition doivent être séparés par des espaces ; il doit y en avoir autant que NP l'indique.

Le programme EXPL10 (fig. 9.12) reprend la classification du précédent programme en y ajoutant une étape de trois partitions de l'arbre en 2, 3 et 4 classes (fig. 9.13). En fin de traitement, le tableau temporaire PARINDIC est rangé dans la base sous le même nom (tab. 9.1) ; la variable identifiant les observations __ NOM qui a une longueur de 4 caractères est recopiée sous le nom CODE, avec 3 caractères seulement. Ces trois partitions pourront donc faire ultérieurement l'objet de cartes. La suite du traitement produit les statistiques par classe sur les variables de l'analyse (fig. 9.14). Notons l'usage qui est fait de l'instruction BY pour itérer le calcul des statistiques descriptives selon les classes de chaque partition.

OBS	_NOM	_INER	_POID
1	BDI	2.9937	1
2	BEN	9.8100	1
3	BWA	25.8354	1
4	CAF	3.6204	1
5	ETH	8.6607	1
6	GIN	3.8800	1
7	HVO	4.0655	1
8	MLI	3.5999	1
9	MWI	4.3479	1
10	NER	13.7050	1
11	RWA	5.1888	1
12	SDN	8.4400	1
13	SLE	4.5549	1
14	SOM	35.1567	1
15	TCD	7.1477	1
16	TGO	3.6685	1
17	TZA	1.6074	1
18	UGA	15.7146	1

OBS	_F1	_F2	_F3
1	1.2180	-0.4877	0.7520
2	-1.3644	-1.3911	0.6132
3	-3.8461	-1.0554	-2.9927
4	1.1254	-0.9747	0.2315
5	2.3062	1.4884	-0.1139
6	0.4423	-0.5841	-0.6619
7	1.5220	-0.1806	0.4742
8	0.4769	0.5101	1.3488
9	1.3747	-0.9124	0.0675
10	-2.2931	-0.4771	1.4904
11	1.2129	-0.2759	0.3102
12	-1.8037	-0.0194	0.6244
13	1.5774	-1.1702	-0.3350
14	-5.2518	2.0944	1.2588
15	2.1900	1.2511	0.1584
16	-0.1837	-1.3352	-0.9092
17	0.3822	0.5797	-0.3002
18	0.9148	2.9400	-2.0167

Tab. 9.1. – *Le stockage des coordonnées des observations sur les trois premières composantes principales.*

OBS	_P1	_P2	_P3	CODE
1	1	1	1	BDI
2	2	2	2	BEN
3	2	3	3	BWA
4	1	1	1	CAF
5	1	1	1	ETH
6	2	2	2	GIN
7	1	1	1	HVO
8	1	1	1	MLI
9	1	1	1	MWI
10	2	2	2	NER
11	1	1	1	RWA
12	2	2	2	SDN
13	1	1	1	SLE
14	2	3	4	SOM
15	1	1	1	TCD
16	2	2	2	TGO
17	1	1	1	TZA
18	1	1	1	UGA

Tab. 9.2. – *Le tableau CAHPART contenant les trois partitions.*

```
B I B L I O T H E Q U E   A D D A D  - VERS. 83 -

STOCKAGE DE PARTITIONS A PARTIR DES PARAMETRES D'UNE CAH (CLACAH - 305)
AUTEUR : M.-O. LEBEAUX

CARTE  1 - TITRE DE L'ANALYSE
           CAH DU TABLEAU INDIC  S

CARTE  2 - PARAMETRES GENERAUX : NI,NP,LECAH,LECNI,STAF
              18     3     1     1     1

CARTE  3 - NOMBRE DE CLASSES PAR PARTITION :
              2     3     4

CARTE  4 - FORMAT DE LECTURE DES IDENTICATEURS :
           (20A4)


NUMEROS DES CLASSES DES PARTITIONS

PARTITION  1  :   1 (  32)-  2 (  34)-
PARTITION  2  :   1 (  32)-  2 (  31)-  3 (  33)-
PARTITION  3  :   1 (  32)-  2 (  31)-  3 (   3)-  4 (  14)-

ON A ECRIT, SUR LE FICHIER  20.    18 OBSERVATIONS, ET POUR CHAQUE OBSERVATION,    3 PARTITIONS
```

Fig. 9.13. – *Les paramètres de la partition.*

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE
				=1	
VAR7082	11	2.35	0.66	1.40	3.40
DEFCAL80	11	-12.91	7.89	-26.00	-4.00
DETHAB82	11	69.67	34.79	27.38	124.26
REGHAB82	11	3.53	3.79	0.26	12.08
PUBHAB82	11	24.10	10.54	6.61	37.62
MARHAB82	11	2.56	2.61	-0.36	7.14
MILHAB82	11	12.54	16.41	3.27	60.34
CADPUB82	11	94.25	6.57	76.99	101.25
OPEPUB82	11	4.31	4.77	0.05	17.01
				=2	
VAR7082	7	3.07	1.05	2.10	5.10
DEFCAL80	7	-6.14	5.67	-16.00	1.00
DETHAB82	7	243.62	94.68	123.28	399.79
REGHAB82	7	16.08	9.94	4.61	31.65
PUBHAB82	7	49.71	34.94	13.16	104.74
MARHAB82	7	14.50	11.92	3.47	31.14
MILHAB82	7	15.61	11.66	2.85	29.90
CADPUB82	7	79.58	13.59	59.58	95.08
OPEPUB82	7	17.63	14.57	4.63	39.14

Fig. 9.14.a. – *Des statistiques descriptives sur deux classes.*

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE
				=1	
VAR7082	11	2.35	0.68	1.40	3.40
DEFCAL80	11	-12.91	7.89	-26.00	-4.00
DETHAB82	11	69.67	34.79	27.38	124.26
REGHAB82	11	3.53	3.79	0.26	12.08
PUBHAB82	11	24.10	10.54	6.61	37.62
MARHAB82	11	2.53	2.61	-0.36	7.14
MILHAB82	11	12.54	16.41	3.27	60.34
CADPUB82	11	94.25	6.57	76.99	101.25
OPEPUB82	11	4.31	4.77	0.05	17.01
				=2	
VAR7082	5	2.52	0.33	2.10	3.00
DEFCAL80	5	-5.80	6.87	-16.00	1.00
DETHAB82	5	222.60	78.39	123.28	302.99
REGHAB82	5	14.90	7.90	4.61	26.72
PUBHAB82	5	30.21	12.14	13.16	45.28
MARHAB82	5	10.76	11.19	3.47	30.41
MILHAB82	5	9.97	8.04	2.85	23.75
CADPUB82	5	81.21	11.76	65.98	95.08
OPEPUB82	5	15.15	13.21	4.63	33.39
				=3	
VAR7082	2	4.45	0.92	3.80	5.10
DEFCAL80	2	-7.00	1.41	-8.00	-6.00
DETHAB82	2	296.19	146.52	192.58	399.79
REGHAB82	2	19.03	17.84	6.42	31.65
PUBHAB82	2	98.46	8.88	92.19	104.74
MARHAB82	2	23.87	10.28	16.60	31.14
MILHAB82	2	29.71	0.26	29.53	29.90
CADPUB82	2	75.51	22.53	59.58	91.44
OPEPUB82	2	23.85	21.62	8.56	39.14

Fig. 9.14.b. – *Sur trois classes.*

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE
				=1	
VAR7082	11	2.35	0.68	1.40	3.40
DEFCAL80	11	-12.91	7.89	-26.00	-4.00
DETHAB82	11	69.67	34.79	27.38	124.26
REGHAB82	11	3.53	3.79	0.26	12.08
PUBHAB82	11	24.10	10.54	6.61	37.62
MARHAB82	11	2.53	2.61	-0.36	7.14
MILHAB82	11	12.54	16.41	3.27	60.34
CADPUB82	11	94.25	6.57	76.99	101.25
OPEPUB82	11	4.31	4.77	0.05	17.01
				=2	
VAR7082	5	2.52	0.33	2.10	3.00
DEFCAL80	5	-5.80	6.87	-16.00	1.00
DETHAB82	5	222.60	78.39	123.28	302.99
REGHAB82	5	14.90	7.90	4.61	26.72
PUBHAB82	5	30.21	12.14	13.16	45.28
MARHAB82	5	10.76	11.19	3.47	30.41
MILHAB82	5	9.97	8.04	2.85	23.75
CADPUB82	5	81.21	11.76	65.98	95.08
OPEPUB82	5	15.15	13.21	4.63	33.39
				=3	
VAR7082	1	3.80	.	3.80	3.80
DEFCAL80	1	-6.00	.	-6.00	-6.00
DETHAB82	1	399.79	.	399.79	399.79
REGHAB82	1	31.65	.	31.65	31.65
PUBHAB82	1	104.74	.	104.74	104.74
MARHAB82	1	16.60	.	16.60	16.60
MILHAB82	1	29.90	.	29.90	29.90
CADPUB82	1	91.44	.	91.44	91.44
OPEPUB82	1	8.56	.	8.56	8.56
				=4	
VAR7082	1	5.10	.	5.10	5.10
DEFCAL80	1	-8.00	.	-8.00	-8.00
DETHAB82	1	192.58	.	192.58	192.58
REGHAB82	1	6.42	.	6.42	6.42
PUBHAB82	1	92.19	.	92.19	92.19
MARHAB82	1	31.14	.	31.14	31.14
MILHAB82	1	29.53	.	29.53	29.53
CADPUB82	1	59.58	.	59.58	59.58
OPEPUB82	1	39.14	.	39.14	39.14

Fig. 9.14.c. – *Sur quatre classes.*

10. Les différents niveaux du langage SAS

Les étapes DATA et PROC sont régies par un langage de même niveau appelé jusqu'ici langage SAS ; les phrases constituées à l'aide de ce langage sont des instructions commençant par un mot clé donnant leur nom aux instructions et s'achèvent par un point virgule. Il existe deux autres niveaux de langage, l'un supérieur au précédent, appelé « macro language », l'autre inférieur nommé « Interactive Matrix Language ».

10.1. Macro langage

Il a pour fonction principale l'assemblage d'étapes SAS programmées en vue d'une application particulière et systématique. Ces programmes, nommés « macros SAS » sont paramétrables et peuvent constituer la boite à outil propre à chaque utilisateur.

10.1.1. Macros SAS.

Il s'agit d'un programme composé d'une ou plusieurs étapes DATA ou PROC paramétrés et commençant avec l'instruction :

%MACRO nom de la macro(liste de paramètres) ;

et s'achevant par l'instruction :

%MEND nom de la macro ;

Les paramètres sont des noms SAS ; ils doivent être séparés par des virgules. On les appelles aussi « macro-variables ». Par exemple, on désire afficher systématiquement le contenu d'un tableau à l'aide de PROC CONTENTS et afficher ce tableau à l'aide de PROC PRINT. La macro TABLEAU réalisant cette fonction a donc un seul paramètre, le nom du tableau ; ce paramètre a pour nom symbolique TAB dans la liste des paramètres de la macro figurant entre parenthèses. Chaque fois qu'il y sera fait référence, ce nom sera précédé du « et » commercial (&), &TAB.

```
%MACRO TABLEAU(TAB) ;
                PROC CONTENTS
                DATA=BASE.&TAB
                POSITION ;
                PROC PRINT DATA=
                BASE.&TAB UNIFORM
                ROUND ;
                RUN ;
%MEND TABLEAU ;
```

A la suite de la programmation, il faut appeler cette macro par son nom, en lui adressant le paramètre qui lui est nécessaire. Ceci peut être fait à l'aide de l'instruction :

%nom de la macro(valeurs des paramètres) ;

Les valeurs des paramètres doivent être séparées par des virgules. Dans le cas précédent, il n'y en a qu'un seul. Il faut donc écrire :

```
%TABLEAU(CODPOP) ;
%TABLEAU(DETTES) ;
%TABLEAU(REGIME) ;
```

pour obtenir l'affichage du contenu et les valeurs des variables des tableaux permanents CODPOP, DETTES et REGIME.

10.1.2. Macros instructions

Il existe un grand nombre de macros instructions faisant du macro langage un véritable langage de programmation gouvernant le langage SAS des étapes DATA et PROC. Les envisager toutes reviendrait à écrire à nouveau le manuel de référence qui n'est, par ailleurs, pas toujours très clair à ce sujet. Il est préférable de ne retenir ici que les instructions les plus courantes :

a. Créer une macro variable et lui affecter une valeur est extrêmement simple (cette macro variable ne figure donc pas dans la liste des paramètres) :

%LET nom de la macro variable=valeur ;
%LET TITRE=VARIABLE ;

affectera la valeur 'VARIABLE' à la macro variable TITRE qui pourra ainsi être utilisée sous le nom &TITRE.

b. Modifier inconditionnellement le déroulement séquentiel de la macro est réalisable à l'aide des instructions :

%GOTO étiquette ;
%étiquette : ;

Par exemple :

%GOTO FIN ;
...
... cette partie est ignorée à l'exécution
...
%FIN : ;
...
... cette partie est exécutée
...
%MEND nom de la macro ;

c. Modifier conditionnellement le déroulement séquentiel de la macro s'effectue par l'instruction :

%IF condition %THEN action ;

Par exemple :

%IF &NOM eq FINI %THEN %GOTO %FIN ;
. . . . . .
. . . . . cette partie de la macro est ignorée si
..... le paramètre NOM a pour valeur FINI
. . . . . .
%FIN ;
. . . . . cette partie de la macro est exécutée
. . . . . .
%MEND nom de la macro ;

```
/*-------------------------------------------------------------*/
/*    MACRO ETUDSTAT POUR L'ETUDE STATISTIQUE DE VARIABLES     */
/*          CARACTERES (NOMINALES) OU NUMERIQUES               */
/*-------------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> DEFINITION DE LA MACRO ETUDSTAT */

%MACRO ETUDSTAT(TAB,NOM,TYPE);

PROC PRINT DATA=BASE.&TAB;VAR CODE &NOM;

%LET TITRE='VARIABLE : ';
%IF &TYPE EQ CAR %THEN %GOTO CAR;
    %IF &TYPE NE NUM %THEN %GOTO FIN;

     /*-----> SI VARIABLE NUMERIQUE */

PROC MEANS DATA=BASE.&TAB MAXDEC=2;
     VAR &NOM;
         TITLE &TITRE&NOM

%GOTO FIN;

     /*-----> SI VARIABLE CARACTERE */

%CAR:;

PROC FREQ DATA=BASE.&TAB;
     TABLES &NOM;
            TITLE &TITRE&NOM;

     /*-----> SI VARIABLE NON NUMERQIUE OU NON CARACTERE
              OU SI FIN DE L'ANALYSE */

%FIN:;

%MEND ETUDSTAT;

%ETUDSTAT(REGIME,REGPOL82,CAR);
         %ETUDSTAT(INDIC,PUBHAB82,NUM);
```

Fig. 10.1. – *Le programme EXPL11 : un exemple d'utilisation du macro-langage.*

Le programme EXPL11 (fig. 10.1) donne un exemple très simple de ce qu'il est possible de faire à l'aide du macro langage. Il s'agit de l'étude statistique d'une variable (son nom est donné par le paramètre NOM de la macro) d'un tableau (son nom est donné par le paramètre TAB), en fonction de son type (défini par le paramètre TYPE) qui peut être CAR s'il s'agit d'une variable alphanumérique ou NUM pour numérique. Si la variable est alphanumérique (comme c'est le cas de la variable REGPOL82 du tableau REGIME), la procédure FREQ comptera les modalités ; dans le cas d'une variable numérique (comme c'est le cas de la variable PUBHAB82 du tableau INDIC), la procédure MEANS en donnera les principales caractéristiques statistiques. La figure 10.2 représente les résultats possibles. Dans tous les cas, la procédure PRINT affichera le tableau choisi.

10.2. Le langage matriciel

Le langage matriciel était accessible dans la version 1982 de SAS sous le nom de PROC MATRIX. Cette procédure continue à exister dans la VERSION 5, mais n'est plus documentée ; il faut se référer au manuel 1982 STATISTICS pour en connaître le détail. La VERSION 5 proposera, courant 1986, l'« Interactive Matrix Language » qui remplacera la procédure MATRIX, avec, en général, les même instructions et quelques possibilités supplémentaires.

Matrix est un langage de programmation très complet permettant l'écriture directe des expressions algébriques matricielles. Si une application ne figure pas dans les procédures, il est possible de la programmer assez facilement pour peu qu'on dispose

OBS	CODE	REGPOL82
1	BEN	MI
2	BWA	PP
3	HVO	MI
4	BDI	MI
5	DJI	NC
6	ETH	MI
7	GMB	PP
8	GIN	PU
9	GNB	MI
10	GNO	MI
11	LSO	DE
12	MWI	DE
13	MLI	MI
14	NER	MI
15	UGA	PR
16	CAF	PP
17	TZA	PU
18	RWA	MI
19	SLE	PR
20	SOM	MI
21	SDN	PU
22	TCD	MI
23	TGO	MI

VARIABLE : REGPOL82

regimes politiques 1982

REGPOL82	FREQUENCY	CUM FREQ	PERCENT	CUM PERCENT
DE	2	2	8.696	8.696
MI	12	14	52.174	60.870
NC	1	15	4.348	65.217
PP	3	18	13.043	78.261
PR	2	20	8.696	86.957
PU	3	23	13.043	100.000

Fig. 10.2.a. – *Le résultat de la macro ETUDSTAT si type=CAR.*

OBS	CODE	PUBHAB82
1	BDI	30.634
2	BEN	26.243
3	BWA	104.742
4	CAF	37.615
5	DJI	176.667
6	ETH	6.611
7	GIN	13.162
8	GMB	79.365
9	GNB	81.412
10	GNQ	37.838
11	HVO	33.491
12	LSO	63.617
13	MLI	28.188
14	MWI	19.378
15	NER	45.276
16	RWA	27.368
17	SDN	37.549
18	SLE	23.402
19	SOM	92.185
20	TCD	13.825
21	TGO	28.806
22	TZA	35.101
23	UGA	9.433

VARIABLE : PUBHAB82

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE
PUBHAB82	23	45.74	39.10	6.61	176.67

Fig. 10.2.b. – *Le résultat de la macro ETUDSTAT si type=NUM.*

des formules matricielles. Cela permet de remplacer un grand nombre de programmes écrits en FORTRAN, par exemple, tout en bénéficiant de la puissance du système SAS en matière de gestion des données.

Une présentation approfondie de MATRIX nécessiterait plusieurs dizaines de pages, ce qui n'est pas possible dans le cadre fixé ici. La description d'une application donne une bonne idée de ce qu'il est possible d'attendre de ce langage. Il s'agit simplement de la programmation d'une régression linéaire multiple dont l'expression matricielle est donnée par la figure 10.3. Pour estimer les coefficients, la formule matricielle est très simple :

$$B=(X'X)^{-1}X'Y$$

où X est une matrice renfermant les variables exogènes et une constante, Y est un vecteur contenant la variable endogène et B est le vecteur des coefficients de régression. A partir de ceux-ci, on peut calculer les estimations :

$$E=X\ B$$

ainsi que les résidus :

$$R=Y-E$$

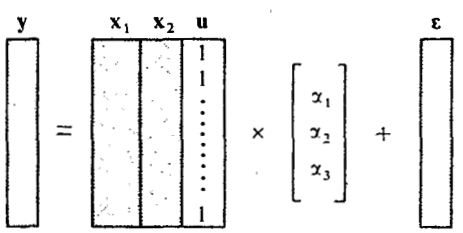


Fig. 10.3. – *Le système d'équations de régression sous forme matricielle.*

La figure 10.4 montre, en quatre étapes comment réaliser ce traitement avec le programme EXPL12.

a. La première étape constitue l'allocation de la base qui doit être en disposition OLD pour pouvoir y ranger les estimations et les résidus en fin de traitement.

b. la sélection des données est assurée par la seconde étape. On cherche là à exprimer le modèle suivant : PUBHAB82=VAR7082 DEFCAL80, ainsi que cela

```
/*-------------------------------------------------------------*/
/*      PROGRAMME DE PRESENTATION DU LANGAGE MATRICIEL         */
/*                      MATRIX OU IML                          */
/*      ETUDE DE CAS : LA REGRESSION LINEAIRE MULTIPLE         */
/*         SUR LE MODELE PUBHAB82=VAR7082 DEFCAL80             */
/*-------------------------------------------------------------*/

/*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

/*-----> SELECTION DES DONNEES */

DATA SELECT;SET BASE.INDIC;KEEP CODE PUBHAB82 VAR7082 DEFCAL80;
     ARRAY MANQ PUBHAB82 VAR7082 DEFCAL80;
           DO OVER MANQ;IF MANQ EQ . THEN DELETE;END;

DATA Y;SET SELECT;KEEP CODE PUBHAB82;
       DATA X;SET SELECT;CTE=1;KEEP CODE VAR7082 DEFCAL80 CTE;

/*-----> PROGRAMMATION DE LA REGRESSION */

PROC MATRIX;

       /*-----> IDENTIFIER          COLONNES */

   VAREXOG='VAR7082' 'DEFCAL80' 'CTE';
   VARENDO='PUBHAB82';

        /*-----> AFFECTER LES VALEURS AUX MATRICES X ET Y */

   FETCH Y DATA=Y ROWNAME=CODE ;
   PRINT Y ROWNAME=CODE COLNAME=VARENDO;

   FETCH X DATA=X ROWNAME=CODE ;
   PRINT X ROWNAME=CODE COLNAME=VAREXOG;

        /*-----> ESTIMER LES COEFFICIENTS */

   XTX=X'*X;
   PRINT XTX ROWNAME=VAREXOG COLNAME=VAREXOG;

       INVXTX=INV(XTX);
       PRINT INVXTX ROWNAME=VAREXOG COLNAME=VAREXOG;

              XTY=X'*Y;
              PRINT XTY ROWNAME=VAREXOG COLNAME=VAREXOG;

                  COEF=INVXTX*XTY;
                  PRINT COEF ROWNAME=VAREXOG COLNAME=VAREXOG;

       /*-----> CALCULER ESTIMATIONS ET RESIDUS */

    ESTIM=X*COEF;
    VARESTIM='ESTIMAT';
    PRINT ESTIM ROWNAME=CODE COLNAME=VARESTIM;

    RESID=Y-ESTIM;
    VARRESID='RESIDUS';
    PRINT RESID ROWNAME=CODE COLNAME=VARRESID;

       /*-----> RANGER LES RESIDUS DANS LE TABLEAU RESIDUS */

    OUTPUT RESID OUT=RESIDUS(RENAME=(ROW=CODE COL1=RESIDUS))
                 ROWNAME=CODE;

/*-----> AFFICHER LE TABLEAU RESIDUS */

PROC PRINT DATA=RESIDUS;RUN;
```

Fig. 10.4. – *Le programme EXPL12 : un exemple d'utilisation du langage matriciel.*

Y	PUBHAB82
BDI	30.6338
BEN	26.2431
BWA	104.742
CAF	37.6151
ETH	6.61074
GIN	13.1621
GMB	79.3651
GNB	81.4118
HVO	33.4906
LSO	63.617
MLI	28.188
MWI	19.378
NER	45.2763
RWA	27.3684
SDN	37.5493
SLE	23.4018
SOM	92.185
TCD	13.8248
TGO	28.806
TZA	35.1013
UGA	9.43343

X	VAR7082	DEFCAL80	CTE
BDI	1.8	-8	1
BEN	2.5	0	1
BWA	3.8	-6	1
CAF	2.1	-4	1
ETH	2.4	-26	1
GIN	2.1	-16	1
GMB	2.9	-6	1
GNB	4.1	2	1
HVO	1.4	-15	1
LSO	2.4	7	1
MLI	2.1	-15	1
MWI	2.6	-4	1
NER	2.6	-6	1
RWA	3.3	-5	1
SDN	3	1	1
SLE	1.6	-8	1
SOM	5.1	-8	1
TCD	2.1	-24	1
TGO	2.4	-8	1
TZA	3.4	-13	1
UGA	3.1	-20	1

Fig. 10.5. – *La matrice Y (variable endogène) et la matrice X (variables exogènes et terme constant).*

a déjà été fait dans le chapitre nº 8. La première étape DATA SELECT met dans ce tableau temporaire les trois variables du modèle, ainsi que la variable CODE, tout en éliminant les observations présentant au moins une observation manquante (à l'aide de l'instruction ARRAY et DO OVER). La seconde étape DATA Y retient dans ce tableau temporaire Y la variable CODE ainsi que la variable endogène PUBHAB82. La troisième étape DATA conserve dans ce tableau temporaire X la variable CODE, les variables exogènes VAR7082 et DEFCAL80, tout en y ajoutant une

XTX	VAR7082	DEFCAL80	CTE
VAR7082	169.18	-455.4	56.8
DEFCAL80	-455.4	3002	-182
CTE	56.8	-182	21

INVXTX	VAR7082	DEFCAL80	CTE
VAR7082	0.0685142	-0.00177297	-0.20068
DEFCAL80	-0.00177297	0.000747798	0.0112764
CTE	-0.20068	0.0112764	0.68814

XTY	VAR7082
VAR7082	2589.38
DEFCAL80	-4780.07
CTE	837.404

COEF	VAR7082
VAR7082	17.8339
DEFCAL80	1.27746
CTE	2.71123

Fig. 10.6. – *L'estimation des coefficients de régression.*

ESTIM	ESTIMAT
BDI	24.5926
BEN	47.296
BWA	62.8153
CAF	35.0526
ETH	12.2986
GIN	19.7231
GMB	46.7648
GNB	78.3851
HVO	8.51679
LSO	54.4548
MLI	21.0005
MWI	43.9695
NER	41.4146
RWA	55.1758
SDN	57.4904
SLE	21.0258
SOM	83.4445
TCD	9.50339
TGO	35.2929
TZA	46.7395
UGA	32.4471

RESID	RESIDUS
BDI	6.04123
BEN	-21.0529
BWA	41.927
CAF	2.56248
ETH	-5.6879
GIN	-6.56101
GMB	32.6003
GNB	3.02662
HVO	24.9738
LSO	9.16221
MLI	7.18748
MWI	-24.5915
NER	3.86168
RWA	-27.8074
SDN	-19.9411
SLE	2.37597
SOM	8.74059
TCD	4.3214
TGO	-6.48694
TZA	-11.6382
UGA	-23.0137

Fig. 10.7. – *Les matrices ESTIM (estimations) et RESID (résidus).*

nouvelle variable CTE dont toutes les valeurs vaudront 1 (il s'agit bien sûr de la constante associée à la matrice des variables exogènes). A l'issue de ces étapes DATA, on dispose donc d'un tableau Y contenant la variable endogène et d'un tableau X renfermant les variables exogènes ainsi qu'une constante. Ces deux tableaux ont également la variable commune CODE.

c. L'appel de la procédure MATRIX se fait sans autre paramètre ou option :

PROC MATRIX ;

Le traitement est alors réalisé en cinq étapes .

c1. Deux instructions servent à définir des vecteurs d'identification des colonnes. Le vecteur VAREXOG possède trois éléments alphanumériques (VAR7082, DEFCAL80 et CTE) ; le vecteur alphanumérique VARENDO n'a, lui, qu'un seul élément (PUBHAB82). Ces deux vecteurs seront utilisés lors de l'affichage des matrices.

c2. La seconde étape va affecter à des matrices le contenu des tableaux Y et X (ces matrices auront également ces noms), puis afficher leur contenu. Deux instructions sont nécessaires :

FETCH nom de la matrice DATA=nom du tableau SAS
ROWNAME=nom de la variable identifiant les observations ;

Pour le matrice Y, on doit écrire :

FETCH Y DATA=Y ROWNAME=CODE ;

Puis, pour imprimer cette matrice Y, il faut utiliser l'instruction PRINT :

PRINT nom de la matrice
ROWNAME=nom de la variable identifiant les observations
COLNAME=nom du vecteur identifiant les variables ;

Pour la matrice Y (fig. 10.5), il faut écrire :

PRINT Y ROWNAME=CODE COLNAME=VARENDO ;

c3. Puis, suit le calcul de l'estimation des coefficients. Chaque phase est ici décomposée et le résultat imprimé. Le produit X' par X est rangé dans la matrice XTX ; notons que l'opérateur de multiplication matricielle est tout simplement l'astérisque (*). L'inverse de la matrice X'X, calculé à l'aide de la fonction INV est mis dans la matrice INVXTX. Le produit X' par Y figure dans la matrice XTY. Enfin, la matrice COEF contient l'estimation des coefficients de régression calculés par multiplication de la matrice INVXTX par XTY. La figure 10.6 donne le contenu de ces matrices successives.

OBS	CODE	RESIDUS
1	BDI	6.041
2	BEN	-21.053
3	BWA	41.927
4	CAF	2.562
5	ETH	-5.688
6	GIN	-6.561
7	GMB	32.600
8	GNB	3.027
9	HVO	24.974
10	LSO	9.162
11	MLI	7.187
12	MWI	-24.592
13	NER	3.862
14	RWA	-27.807
15	SDN	-19.941
16	SLE	2.376
17	SOM	8.741
18	TCD	4.321
19	TGO	-6.487
20	TZA	-11.638
21	UGA	-23.014

Tab. 10.1. – *Le tableau RESIDUS.*

c4. La phase suivante est le calcul des estimations et des résidus. Les estimations de la matrice ESTIM sont calculées en multipliant X par COEF, les résidus, en soustrayant ESTIM de Y (fig. 10.7).

c5. Pour conserver le contenu d'une matrice dans un tableau SAS, il est nécessaire de faire appel à l'instruction OUTPUT qui a pour syntaxe :

OUTPUT nom de la matrice OUT=nom du tableau(options)
ROWNAME=nom de la variable identifiant les observations ;

Les options permettent de modifier les noms des variables ROW (pour les lignes) et COL (pour les colonnes) dans le tableau ainsi créé.

d. La quatrième étape montre qu'il est possible de traiter le tableau précédemment rangé dans la base SAS comme un tableau ordinaire, ici à l'aide de l'instruction PRINT (tab. 10.1).

11. Les unités graphiques

La réalisation de graphiques avec un ordinateur nécessite de disposer d'une part d'un progiciel graphique et, d'autre part, d'unités spécialisées ayant pour fonction principale l'affichage des images.

Le progiciel graphique SAS/GRAPH propose un grand choix d'options et de procédures facilitant la réalisation de diagrammes, de courbes, de cartes thématiques, tout en permettant de bénéficier des nombreuses autres possibilités de gestion et d'analyse des données du reste du système. Les chapitres suivants aborderont la méthodologie nécessaire à la réalisation des cartes thématiques.

Les unités graphiques se répartissent en trois grandes familles : les numériseurs, les écrans graphiques, les traceurs et imprimantes graphiques. Il est assez difficile d'obtenir une information claire sur le sujet ; il apparaît donc très utile de préciser ici le rôle des principaux matériels composant ces familles.

11.1. Les numériseurs

Les numériseurs (ou digitaliseurs) sont des unités informatiques assurant la conversion d'une localisation sur un plan en une série de couples de coordonnées adressées directement à un ordinateur. Trois principaux éléments les composent : le plus visible est la table constituant la surface à proprement parler ; elle comprend au verso un circuit électronique formant la surface sensible. Un stylet pouvant être déplacé sur cette surface jusqu'à la position désirée constitue le moyen d'interaction entre l'utilisateur et l'ordinateur auquel est connecté le numériseur ; ce stylet a parfois des touches de contrôle additionnelles. Le contrôleur sert à capter les signaux électroniques transmis par la table et le stylet, et à interpréter la localisation en la convertissant sous une forme numérique assimilable par l'ordinateur.

Résolution, précision, linéarité et fiabilité sont quelques-unes des qualités les plus importantes des numériseurs. La résolution peut être définie comme le nombre de points discernables sur une longueur donnée ; le plus souvent, elle est comprise entre 100 et 1000 points par pouce dans chaque direction. La table étant un capteur de coordonnées absolues, la précision peut être définie comme le rapport entre les coordonnées transmises et les coordonnées réelles ; elle est souvent de l'ordre de 0.005 pouce. La variation de la précision sur de grandes surface constitue une estimation de la linéarité : si une ligne droite est tracée sur la table, les coordonnées relevées figurent-elles réellement une ligne droite ? La fiabilité peut être vue comme la variabilité des mesures d'une même position sur la table ; certains numériseurs sont dotés de dispositifs logiciels éliminant les mesures aberrantes.

Trois principaux modes de numérisation sont proposés par les fabricants de tables : le relevé des coordonnées point par point, le relevé en continu avec un nombre de coordonnées fixe par unité de temps et, enfin, le relevé en continu avec un nombre de points dépendant du programme de numérisation. Le second mode peut permettre le relevé de 200 coordonnées par seconde, à condition que la connexion à l'ordinateur soit en mode série, et que son disque soit d'une capacité suffisante.

Deux technologies sont couramment mises en oeuvre pour la construction des numériseurs. Les tables résistantes sont composées de feuilles de matériaux, l'un conducteur, l'autre résistant ; le relevé des coordonnées se fait par différence de potentiel provoquée par la pression du stylet à la surface de la table, différence proportionnelle à la distance aux bords de la table. Les tables résistantes sont actuellement les moins chères du marché. Elles présentent les inconvénients de ne donner aucune trace lorsque le stylet n'est pas pressé, mais aussi une faible vitesse de numérisation. Les tables électromagnétiques sont les plus répandues : un circuit imprimé est gravé sous la table selon des lignes régulièrement espacées, dans l'axe des abscisses sur l'une des faces du circuit, selon l'axe des ordonnées sur l'autre face. Le stylet, équipé d'une bobine reçoit un signal électrique dont l'intensité permet de déterminer les coordonnées. Cette technolo-

gie offre une bonne résolution et une vitesse de relevé en continu assez élevée. Avec ce matériel, il est possible de visualiser en permanence sur un écran la position courante du stylet, même lorsqu'il n'est pas activé pour l'acquisition des coordonnées. Un inconvénient cependant : le stylet doit être relié à la table par un cordon électrique.

Les stylets sont habituellement équipés d'un interrupteur plume baissée/plume levée. Ils peuvent être encrés ou non ; l'encre laisse une trace sur le dessin, ce qui peut être très pratique pour certaines applications. Parfois, ils sont équipés d'un bouton poussoir auxiliaire servant à choisir le mode de numérisation le mieux adapté, ou bien à indiquer à l'ordinateur le début et la fin d'une séquence.

Le CNUSC possède un numériseur BENSON 6201 ayant une table au format A0 (1200 X 870 mm). Sa résolution est de plus ou moins 0.02 mm, avec une linéarité au stylet de plus ou moins 0.5 mm. Le relevé se fait point par point, ou bien en continu, à une cadence comprise entre 1 et 60 Hz. Un petit écran affiche les coordonnées X et Y. Le stylet peut être remplacé par une loupe pour obtenir une meilleure précision du relevé.

11.2. Les écrans graphiques

Les écrans graphiques sont majoritairement des tubes à rayons cathodiques (TRC) ; on trouve aussi sur le marché des tubes à plasma. Il existe deux techniques pour réaliser une image sur l'écran d'un tube cathodique. La technique vectorielle (ou stroker) consiste à tracer des segments, les uns après les autres, comme on pourrait le faire à la main. Les ressources centrales nécessaires sont modestes : un tracé composé de 20000 vecteurs sur un écran à haute résolution de 4096 X 4096 points demande moins de 644 Koctets de mémoire. L'autre technique est celle de la télévision, où l'image est construite par un faisceau d'électrons se déplaçant de gauche à droite ; on parle aussi de tracé de lignes point par point (ou raster scan). Le nombre de lignes va de 512 à 4096 et le nombre de points par ligne de 256 à 4096. Une image en couleur composée de 1024 X 1024 points demande environ 1000 Koctets de mémoire.

La couleur coûte relativement cher tant en matériel qu'en exploitation, mais elle élargit énormément les possibilités d'expression graphique. La couleur est le résultat d'une juxtaposition de teintes, une sensation découlant de l'association de points émis avec des intensités variables des trois couleurs fondamentales ; il peut s'agir de la combinaison du rouge, du vert et du bleu, ou bien du jaune, du cyan et du magenta. Dans les écrans base de gamme, l'utilisateur a le choix entre 4, 8 et parfois 16 couleurs définies dès l'origine. Dans le haut de gamme, la composition conique des couleurs chez Tektronix, par exemple, permet de choisir 8 couleurs parmi 120 teintes de 64 couleurs.

La Maison de la Géographie de Montpellier dispose de plusieurs terminaux graphiques. Le terminal Tektronix CX4107 est composé d'un écran ayant 640 X 480 points permettant d'afficher 16 couleurs parmi 64 ; plusieurs touchesde fonction servent à définir l'état du terminal et à mettre en oeuvre les fonctions du processeur graphique (zoom, etc...). Une imprimante à jet d'encre connectée au terminal assure la recopie couleur de l'image affichée sur l'écran. Le terminal Radiance 320 de la société GIXI a une définition de 640 X 488 points en 256 couleurs possibles ; il peut réaliser à la fois des fonctions d'affichage d'images informatiques, de visualisation d'une prise de vue par caméra vidéo, de conception assistée par ordinateur ou de suivi de processus.

11.3. Les traceurs à plumes

Les traceurs à plumes existent depuis près de 30 ans. A l'origine, ils fonctionnaient uniquement en mode vectoriel, c'est-à-dire en traçant des lignes à partir d'informations transmises par l'ordinateur central pour dessiner des éléments graphiques de base : arcs de cercle, caractères alphanumériques, remplissage de zones (trames). Aujourd'hui, de même que la plupart des traceurs, ils sont connectés à un contrôleur qui libère l'ordinateur central tout en permettant le mode point par point (raster).

Les tracés sont assurés par des stylos feutres, des stylos à bille ou bien des plumes tubulaires à encre de chine liquide. Un grand nombre de traceurs ont jusqu'à 8 plumes de couleur différente et quelques uns vont jusqu'à 14.

Les traceurs à plumes sont les unités graphiques de haute résolution les plus lentes parce que leur vitesse dépend de la capacité des plumes : ceci est encore plus vrai pour les plumes à encre liquide, plus lentes que les stylos à bille. En fait, la vitesse est fonction de trois paramètres : la résistance de la plume aux accélérations, le temps de levée/descente et la rapidité d'exécution des instructions par le contrôleur.

Jusqu'à présent, les traceurs étaient particulièrement adaptés aux tracés finaux ; les travaux intermédiaires étaient assurés par des traceurs électrostatiques. Ces derniers sont aujourd'hui d'une grande précision, possèdent la couleur et donnent des résultats aussi bons, sinon meilleurs que les traceurs à plume, avec des incidents de tracé beaucoup moins fréquents.

Le CIRCE possède deux traceurs à plume BENSON non connectés sur les ordinateurs centraux : les tracés doivent être copiés sur une bande magnétique puis être relus par un lecteur associé à chaque traceur réalisant le dessin ; on dit que leur fonctionnement est OFF LINE. Les traceurs BENSON 112 et BENSON 1232 possèdent trois plumes, ont une résolution de 0.1 millimètre et une longueur maximale de tracé de 50 mètres. Ils diffèrent par la vitesse et la largeur maximale du tracé, respectivement 1500 mm/s et 320 mm pour le 112 et 2500 mm/s et 730 mm pour le 1232. Notons que ces matériels ne peuvent être directement utilisés par SAS/GRAPH : il faut recourir à la passerelle SAS/GPGS (GPGS est le logiciel graphique de base en mode vecteur du CIRCE).

11.4. Les traceurs électrostatiques

Ils sont apparus sur le marché dans les années 1960. A l'inverse des traceurs à plume, leur technologie s'appuie sur le point ou pixel. Dans la tête d'impression, on peut compter jusqu'à 22000 aiguilles fixes disposées selon un pas régulier et sélectables afin d'adresser des charges électrostatiques aux endroits ou un point doit être tracé. Le papier défile ligne par ligne sous la tête, puis passe dans un bain fixant des particules de carbone sur les points chargés électriquement. Les dessins sont donc composés d'une série de points noircis sur le papier et devenant coalescents à la lecture.

La résolution de ces traceurs s'accroît beaucoup, et la dernière innovation dans ce domaine a été l'introduction sur la marché d'un traceur électrostatique couleur par la société VERSATEC. Deux types de problèmes se posent parfois : en premier lieu, les tracés peuvent être masqués lorsque la résolution demandée est supérieure à celle du traceur ; en second lieu, des barres grises apparaissent lorsque le traceur s'arrête en cours de dessin. La qualité du résultat final est en grande partie fonction des positions relatives des aiguilles sur la tête d'impression. BENSON a résolu les problèmes posés par une seule rangée d'aiguilles, en les disposant en quinconce (tête quadrascan).

Les traceurs électrostatiques sont le plus souvent connectés au site central (on dit qu'ils sont ON LINE), un contrôleur ayant à sa charge l'opération de « rastorization », c'est-à-dire de conversion des tracés vectoriels en tracés point par point.

Le CIRCE a un traceur-imprimante électrostatique VERSATEC 1200A connecté à un contrôleur. Les dimensions maximales sont de 268 mm et 130 m, la résolution étant de 0.1 mm ; en mode graphique, la vitesse de tracé est de 254 mm/s. Notons qu'il faut aussi recourir à la passerelle SAS/GPGS pour faire appel à ce matériel.

Le CNUSC est équipé d'un traceur-imprimante électrostatique BENSON VARIAN 9424 fonctionnant aussi ON LINE. Les dimensions maximales du tracé sont 594 X 420 mm, la résolution étant de 100 points par cm ; en mode graphique, la vitesse est de 25 mm/s. Cette unité peut être directement utilisée avec SAS/GRAPH.

11.5. Les traceurs à jet d'encre

Ce sont des unités de haute précision apparues au cours des 20 dernières années. Il s'agit du développement naturel de la technologie des imprimantes ligne à ligne pour lesquelles l'encre est transférée sur le papier par l'intermédiare d'une tête d'écriture. Ici, en particulier, l'encre est tranférée sur le papier par un diaphragme et une buse pouvant envoyer une goutte d'encre à un point précis du papier, à haute vitesse. Les traceurs à jet d'encre sont particulièrement indiqués lorsque les tracés sont composés de surfaces totalement imprimées (des à plats). Notons que les qualités d'impression sont très variables d'un matériel à l'autre.

Le CIRCE est équipé d'un traceur à jet d'encre APPLICON OFF LINE, auquel est associé un lecteur de bande magnétique. Les tracés doivent donc être copiés sur bande avant d'être relus par le lecteur de l'APPLICON. Le papier est placé sur un cylindre tournant à haute vitesse constante ; le système d'impression est composé de trois jets d'encre (rouge, jaune et bleu) se déplaçant parallèlement à l'axe du cylindre et réalisant le tracé en un seul passage. Plus de 15000 nuances peuvent être obtenues à partir des 7 couleurs de base. La surface utile sur une feuille est de 540 X 840 mm ; la résolution est de 50 points par millimètre, chaque point pouvant recevoir une ou deux couleurs. Le traceur APPLICON ne peut, lui aussi, être utilisé directement par SAS/GRAPH ; il faut également recourir à la passerelle SAS/GPGS/UNIRAS (UNIRAS est le logiciel en mode point par point du CIRCE).

Le CNUSC s'est doté d'un traceur à jet d'encre Tektronix 4691 directement connecté au site central. Le système d'impression est composé de trois jets d'encre parallèles au défilement d'une feuille de papier au format A3 (297 X 420 mm). La résolution est de 6 points par mm. Le traceur ne peut être utilisé par SAS/GRAPH. Seuls quelques programmes UNIRAS, réalisés par la Maison de la Géographie de Montpellier et pouvant être exécutés sous la forme d'une procédure SAS (PROC UNISAS) recourent à cette unité graphique.

12. SAS/GRAPH et la cartographie automatique

Quatre procédures de SAS/GRAPH ont une orientation cartographique très marquée :

GPROJECT réalise plusieurs calculs de projection sur un tableau contenant des coordonnées angulaires (latitudes/longitudes) sur la sphère ; ces coordonnées doivent être exprimées en radians.

GREDUCE traite des fonds de carte de manière à optimiser la précision du tracé en fonction de l'échelle. A grande échelle, le nombre de points nécessaires pour tracer les contours d'une unité spatiale sera plus grand que pour une plus petite échelle. Cette procédure assure donc la généralisation des contours des unités spatiales.

GREMOVE transforme les fonds de carte en agrégeant les polygones appartenant à une même unité spatiale de niveau supérieur (pour passer par exemple d'une carte au niveau départemental à une autre, au niveau régional). Les côtés des anciens polygones figurant à l'intérieur des nouveaux sont effacés. Cette procédure nécessite la présence dans le tableau fond de carte d'une variable d'agrégation.

GMAP trace des cartes choroplèthes ou des cartes en prismes. C'est la procédure de cartographie thématique par excellence. En adaptant les fonds de carte, cette procédure peut aussi tracer des cartes ponctuelles.

Deux autres procédures représentent des données en deux ou trois dimensions ; si les deux premières sont constituées par des coordonnées géographiques, on peut aussi dire qu'elles tracent des cartes géographiques :

GCONTOUR représente des surfaces en deux dimensions ; elles sont figurées par des courbes de niveaux résultant d'une interpolation réalisée à partir des points de relevé.

G3D trace des surfaces, en trois dimensions, en perspective cavalière. Les données doivent également faire l'objet d'une interpolation à partir des points de relevé.

SAS/GRAPH propose beaucoup d'autres procédures, moins intéressantes pour la représentation des données spatialisées. Notons également la procédure GREPLAY qui permet de disposer d'une véritable base de données cartographiques, c'est à dire de cartes pouvant être réaffichées à la demande et montées en planches avec d'autres.

Un examen approfondi de l'ensemble de ces procédures nécessiterait un très long exposé ; seules GPROJECT, GREDUCE et GMAP seront présentées ici.

12.1. Les options graphiques

Préalablement à l'exécution des procédures de SAS/GRAPH, il est indispensable de définir l'environnement graphique de la session SAS en cours. Cela se fait très simplement à l'aide de l'unique instruction GOPTIONS. Elle a pour syntaxe :

```
GOPTIONS DEVICE=nom de l'unité graphique
                BORDER NOTEXT82 ;
```

Si on désire afficher le tracé sur l'écran IBM3279 avec un cadre, l'instruction GOPTIONS doit avoir la forme suivante :

```
GOPTIONS DEVICE=IBM3279 BORDER
NOTEXT82 ;
```

Pour utiliser une unité graphique avec SAS/GRAPH, il faut donc disposer du pilote (DRIVER) qui permet de transformer les images indépendantes de l'unité produites par SAS en images affichables ou imprimables sur cette unité. La liste des pilotes est donnée dans le manuel de référence SAS/GRAPH. Il est possible de construire un pilote adapté à une unité ne figurant pas dans la liste de ceux fournis par SAS à partir du squelette nommé UNIVERSAL DRIVER ; notons que cela requiert un bon niveau en informatique et doit donc être réalisé par de véritables spécialistes.

12.2. Les textes des graphiques

Il est indispensable d'habiller les graphiques par des textes identifiant les données, l'auteur de la carte, l'année de référence, etc... Deux instructions facilitent ce travail : TITLE pour les textes figurant au dessus du tracé, FOOTNOTE pour ceux figurant au desous. La syntaxe est du type :

```
TITLE C=couleur F=police de caractères
                H=hauteur PCT 'titre' ;
```

ou bien :

```
FOOTNOTE C=couleur F=police de caractères
                H=hauteur PCT 'note' ;
```

S'il doit y avoir plusieurs titres ou plusieurs footnote (1 à 10 au maximum), TTITLE et FOOTNOTE doivent être suivis du rang de la ligne sur laquelle ils se trouveront :

```
TITLE5 C=BLUE F=SIMPLEX H=3 PCT 'titre' ;
```

signifie que le titre sera sur la cinquième ligne à partir du bord supérieur. Pour les titres, la numérotation se fait de haut en bas, pour les notes, de bas en haut. Les couleurs peuvent être, sur terminal IBM3279 BLACK, WHITE, RED, GREN, BLUE, YELLOW, PINK et CYAN. Les polices de caractères sont présentées dans le manuel de référence SAS/GRAPH ; on uttilise très souvent SIMPLEX et TRIPLEX, XSWISS et TITALIC. La hauteur est exprimée en pourcentage de la hauteur de l'écran si PCT suit H=hauteur ; cette option simplifie grandement la composition des textes. Enfin, le texte du titre ou de la note doivent figurer entre quotes. Par exemple :

```
TITLE2 C=RED H=10 PCT F=TITALIC
                'REGIME POLITIQUE' ;
```

affichera sur la seconde ligne le titre 'REGIME POLITIQUE', en caractères italiques rouges d'une hauteur équivalent à 10% de celle de l'écran. Notons également qu'il est possible de justifier le texte en ajoutant le paramètre J= suivi de L (pour Left, jutification à gauche), C (pour Center, texte centré) ou R (pour Right, texte justifié à droite).

13. Les fonds de carte SAS

Pour réaliser des cartes, il est nécessaire de disposer d'un fond de carte numérisé et mis sous une forme compatible avec les procédures SAS.

13.1. Numérisation et généralisation d'un fond de carte

La carte 1.1 représente 23 pays d'Afrique parmi les moins avancés du globe. Pour numériser et générer ce fond de carte, les opérations suivantes sont nécessaires :

a. les contours des unités spatiales sont simplifiés de manière à ce que chacun d'eux constitue un ploygone (carte 13.1). Chacun des angles de ces polygones est numéroté. Les sommets des angles sont numérisés : à chaque numéro d'angle correspond un couple de coordonnées X, Y. Cette opération peut être réalisée à l'aide d'un numériseur ; dans ce cas, chaque enregistrement contiendra un couple de coordonnées et les enregistrements seront dans l'ordre croissant des numéros d'angles. Pour créer un tableau SAS contenant cette numérisation, il faudra exécuter le programme suivant :

```
X ALLOC DA(.nom du fichier de numérisation) FI(IN) ;
X ALLOC DA(.STAGBASE) FI(BASE) OLD ;
DATA BASE.POINTS ; INFILE IN ; INPUT X Y ; NUM=__N__ ;
```

La variable NUM contient le rang des points dans l'ordre où ils figurent dans le fichier de numérisation. Dans le cas où le nombre de points n'est pas très important, le relevé peut être fait avec l'aide du papier millimétré, puis saisi directement à l'aide de la procédure FSEDIT. Il faudrait donc soumettre le programme :

```
X ALLOC DA(.STAGBASE) FI(BASE) OLD ;
DATA BASE.POINTS ; X=. ; Y=. ; NUM=. ;
PROC FSEDIT DATA=BASE.POINTS ;
```

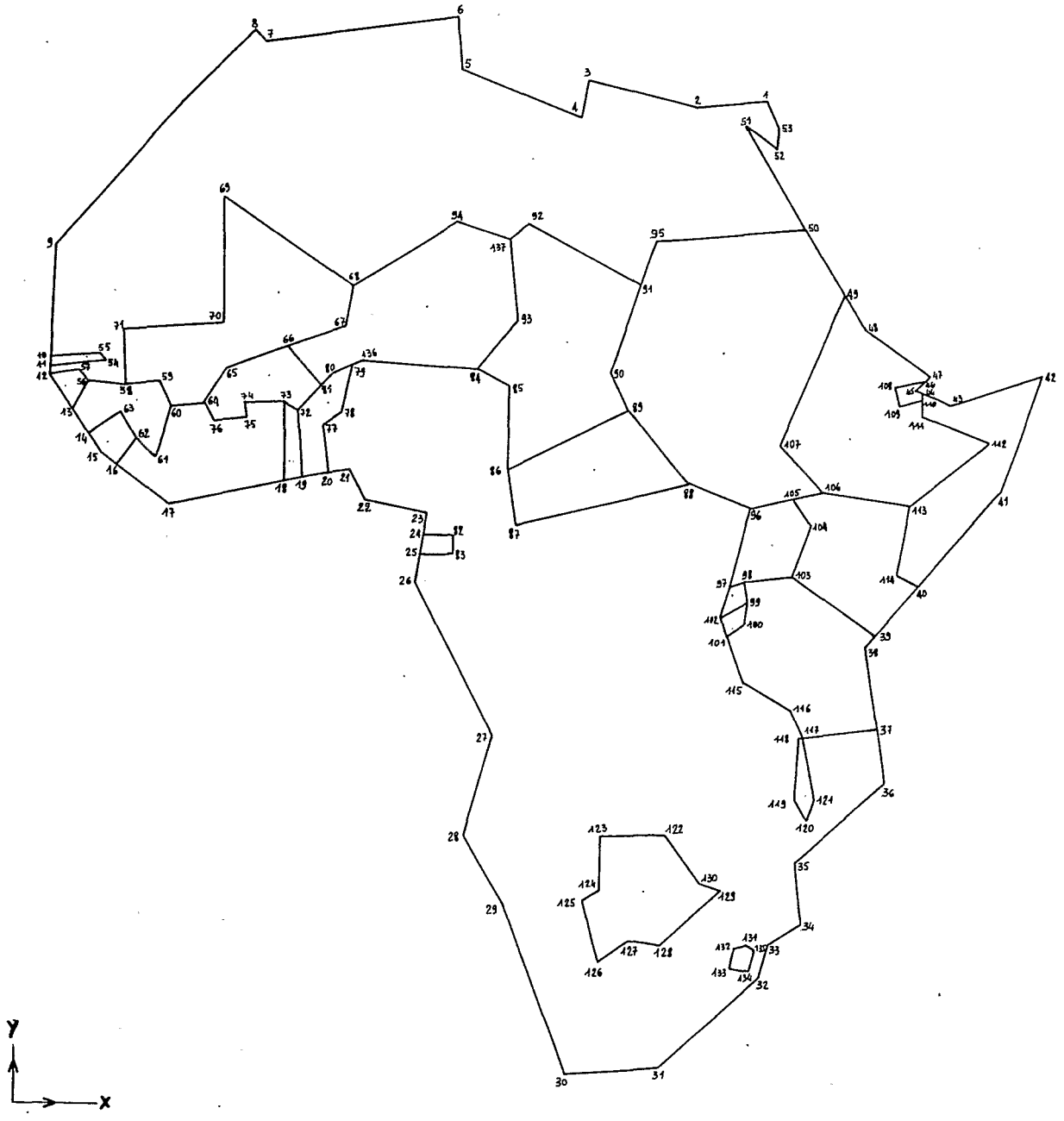
Le tableau 13.1 présente le tableau POINTS.

b. un second tableau doit décrire les contours des polygones par les points qui les composent. Ce tableau contient nécessairement une variable CODE et un nombre de variables numérotées égal au nombre maximum de points pour un polygone (dans le cas de l'Afrique des pays PMA). Le tableau peut être saisi directement par la procédure FSEDIT avec le petit programme suivant :

```
X ALLOC DA(.STAGBASE) FI(BASE) OLD ;
DATA BASE.POINTUS ; LENGTH CODE $ 3 ;
CODE= ;
ARRAY TAB P01-P53 ;
DO OVER TAB ; TAB=. ; END ;
PROC FSEDIT DATA=BASE.POINTUS ;
```

Les variables P01 à P53 contiennent les numéros des points décrivant les contours de chaque polygone. Le tableau 13.3 présente ce tableau POINTUS.

c. le fond de carte SAS bien constitué résulte de l'association des tableaux POINTS et POINTUS. Le tableau 13.4 est la liste du fond de carte final. Ils est composé pour chaque polygone d'un nombre d'observations égal au nombre de points décrivant son contour ; chaque observation de ce tableau comprend trois variables : CODE identifiant les unités spatiales, X et Y les coordonnées du point. Notons qu'il n'est pas nécessaire de fermer le polygone en répétant les coordonnées du premier point après celles du dernier.



Carte 13.1. – *Le fond de carte simplifié des 23 pays les moins avancés d'Afrique.*

OBS	X	Y	NUM
1	174	240	1
2	158	238	2
3	132	244	3
4	131	236	4
5	102	246	5
6	101	259	6
7	56	252	7
8	54	255	8
9	7	204	9
10	6	177	10
11	6	175	11
12	6	173	12
13	12	165	13
14	16	159	14
15	19	155	15
16	22	152	16
17	35	143	17
18	62	149	18
19	66	150	19
20	72	151	20
21	77	151	21
22	81	144	22
23	96	142	23
24	95	136	24
25	94	131	25
26	93	125	26
27	112	88	27
28	105	65	28
29	115	49	29
30	130	8	30
31	152	10	31
32	175	32	32
33	178	40	33
34	185	45	34
35	184	59	35
36	204	78	36
37	203	91	37
38	200	111	38
39	202	114	39
40	212	126	40
41	231	148	41
42	240	175	42
43	219	168	43
44	212	171	44
45	211	172	45
46	213	174	46
47	214	175	47
48	198	186	48
49	193	194	49
50	184	210	50
51	169	234	51
52	177	229	52
53	177	234	53
54	20	176	54
55	18	178	55
56	15	171	56
57	13	174	57
58	24	170	58
59	32	172	59
60	35	166	60
61	31	154	61
62	27	158	62
63	23	164	63
64	43	166	64
65	48	175	65
66	62	180	66
67	76	185	67
68	78	195	68
69	47	215	69
70	47	181	70
71	24	184	71
72	65	165	72
73	61	167	73
74	52	167	74
75	53	163	75
76	45	162	76
77	71	162	77
78	75	165	78
79	77	176	79
80	73	174	80
81	70	171	81
82	102	136	82
83	102	132	83
84	107	175	84
85	115	171	85
86	115	152	86
87	117	139	87
88	157	144	88
89	143	166	89
90	139	175	90
91	145	196	91
92	119	210	92
93	116	187	93
94	102	210	94
95	149	207	95
96	172	143	96
97	167	125	97
98	181	126	98
99	182	121	99
100	181	116	100
101	167	113	101
102	165	118	102
103	182	127	103
104	186	140	104
105	182	146	105
106	189	147	106
107	179	158	107
108	206	172	108
109	207	168	109
110	212	169	110
111	212	165	111
112	228	159	112
113	209	145	113
114	207	128	114
115	171	102	115
116	182	95	116
117	185	89	117
118	184	89	118
119	183	74	119
120	186	69	120
121	188	74	121
122	153	75	122
123	138	75	123
124	138	62	124
125	134	60	125
126	137	45	126
127	145	50	127
128	152	49	128
129	166	62	129
130	161	64	130
131	172	60	131
132	169	59	132
133	168	54	133
134	173	53	134
135	174	58	135
136	80	187	136
137	114	217	137

Tab. 13.1. – *La numérisation des angles des polygones.*

Pour obtenir ce tableau fond de carte, il suffit d'exécuter le simple programme EXPL13 (fig. 13.1). Le tableau fond de carte définitif se nomme FONDAFRI ; il est permanent.

d. en plus du tableau fond de carte, il est très utile de disposer d'un tableau contenant les coordonnées estimées des centres géométriques des polygones. Le tableau aura également trois variables CODE, X et Y. Le tableau 13.2 présente le contenu du tableau permanent CENTRES.

13.2. La généralisation des contours

Généraliser un fond de carte, c'est éliminer des points figurant des sommets d'angles de manière à simplifier les contours des polygones représentant les unités spatiales. Cette simplification est réalisée pour deux raisons. En premier lieu, elle limite le volume des ressources informatiques nécessaires au tracé de la carte : passer de plusieurs milliers de points à quelques centaines, c'est rendre réellement possible

```
/*-------------------------------------------------------*/
/* PROGRAMME DE CONSTITUTION D'UN FOND DE CARTE          */
/* A PARTIR DES TABLEAUX POINTS (COORDONNEES DES         */
/* SOMMETS DES POLYGONES) ET POINTUS (DESCRIPTION        */
/* DES POLYGONES PAR LEURS POINTS).                      */
/*-------------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) OLD;

 /*-----> REORGANISATION DU TABLEAU POINTUS */

DATA POINTUS;SET BASE.POINTUS;
             ARRAY TAB P01-P53;
             DO OVER TAB;
                     IF TAB NE . THEN DO;
                     NUM=TAB;OUTPUT;
                     END;
             END;
             DROP P01-P53;

DATA POINTUS;SET POINTUS;RANG=_N_;

 /*-----> ASSOCIER LES TABLEAUX POINTUS ET BASE.POINTS */

PROC SORT DATA=POINTUS;BY NUM;

DATA FOND;MERGE BASE.POINTS POINTUS;BY NUM;

PROC SORT DATA=FOND;BY RANG;

 /*-----> CREER LE TABLEAU FOND DE CARTE */

DATA BASE.FONDAFRI;SET FOND;
                   KEEP CODE X Y;
```

Fig. 13.1. – *Le programme EXPL13 : création d'un fond de carte après numérisation des angles des polygones.*

OBS	CODE	X	Y
1	GMB	15	175
2	GNB	10	170
3	GIN	27	167
4	SLE	20	157
5	MLI	60	193
6	HVO	58	173
7	BEN	72	167
8	TGO	64	153
9	NER	95	190
10	TCD	128	182
11	CAF	135	152
12	GNQ	98	134
13	SDN	165	180
14	DJI	209	171
15	ETH	195	160
16	SOM	220	145
17	UGA	177	135
18	RWA	168	123
19	BDI	168	115
20	TZA	185	110
21	MWI	186	75
22	BWA	148	52
23	LSO	171	25

Tab. 13.2. – *La numérisation des centres des polygones.*

l'utilisation interactive de l'ordinateur. Le second motif est plus fondamental : il est inutile d'introduire des perturbations dans la « perception rétinienne » à l'oeuvre dans la lecture des cartes thématiques par un luxe de détail des contours ; en effet, ceux-ci n'ont qu'une modeste part dans la totalité de l'information apportée par ce type de carte.

Le tableau fond de carte FONDAFRI a été numérisé puis constitué à partir d'un fond de carte simplifié pour les besoins de l'exposé. Il est donc quelque peu artificiel de vouloir, dans un second temps, le généraliser ; mais puisque le procédé serait identique dans le cas d'un fond de carte plus complexe, rien ne s'oppose, au plan technique, à la généralisation du fond de carte FONDAFRI à l'aide de la procédure GREDUCE. Sa syntaxe est la suivante :

PROC GREDUCE DATA=nom du tableau fond de carte initial
OUT=nom du tableau fond de carte généralisé
E1=distance n° 1 E2=distance n° 2
E3=distance n° 3 E4=distance n° 4
E5=distance n° 5 ;

La généralisation est donc réalisée en fonction de plusieurs distances minimales séparant deux points consécutifs. Le tableau fond de carte à généraliser aura une variable supplémentaire nommée DENSITY qui pourra prendre des valeurs comprises entre 0 et 6. La valeur 0 correspond au fond de carte le plus simplifié où chaque point est commun à plus de deux côtés de polygone. La valeur 6 définit le fond de carte d'origine. De 1 à 5 sont définis les points respectifs des distances E1 à E5.

A la suite de la procédure GREDUCE, il faut constituer le fond de carte dans une étape DATA, en sélectionnant les points devant y figurer, en fonction de leur valeur sur la variable DENSITY. Par exemple, si on ne désire retenir que les points nécessaires à un fond de carte généralisé au niveau 3 (correspondant à une valeur E3 choisie lors du GREDUCE), il faut, préalablement à tout tracé, effectuer une étape DATA :

DATA FOND ; SET FONDRED ;
IF DENSIY LE 3 ;

FOND est le nom du tableau fond de carte généralisé ; FONDRED est le nom du tableau fond de carte à généraliser issu de la procédure GREDUCE et contenant de ce fait la variable DENSITY.

Le programme EXPL14 (fig. 13.2) donne un exemple de généralisation du fond de carte permanent FONDAFRI. Ce programme est composé de quatre étapes : l'allocation de la base, le généralisation du fond avec GREDUCE, produisant le tableau AFRIGENE, la sélection de sept fonds de carte correspondant aux densités 6 à 0 et, enfin, le tracé de ces fonds de carte à l'aide de la procédure GMAP (le fonctionnement de cette procédure sera expliqué au chapitre 14). Les cartes 13.2 présentent le résultat de ces tracés de la densité 6 à la densité 1 ; outre des détails de contours, il arrive que des pays disparaissent complètement.

13.3. Les projections

Lorsque les coordonnées géographiques sont connues sous la forme de mesures angulaires en latitude et longitude, il est absurde de demander à la

```
/*-----------------------------------------------------*/
/* PROCEDURE GREDUCE SUR LE FOND DE CARTE AFRIQUE     */
/*-----------------------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE);

 /*-----> GENERALISATION DU FOND */

PROC GREDUCE DATA=BASE.FONDAFRI OUT=AFRIGENE
     E1=5 E2=4 E3=3 E4=2 E5=1;ID CODE;

PROC PRINT DATA=AFRIGENE UNIFORM ROUND;

 /*-----> SELECTION DES FONDS DE CARTE */

DATA AFR6;SET AFRIGENE;IF DENSITY LE 6;D=6;
     DATA AFR5;SET AFRIGENE;IF DENSITY LE 5;D=5;
          DATA AFR4;SET AFRIGENE;IF DENSITY LE 4;D=4;
               DATA AFR3;SET AFRIGENE;IF DENSITY LE 3;D=3;
                    DATA AFR2;SET AFRIGENE;IF DENSITY LE 2;D=2;
                         DATA AFR1;SET AFRIGENE;IF DENSITY LE 1;D=1;
                              DATA AFR0;SET AFRIGENE;IF DENSITY LE 0;D=0;

 /*-----> TRACE DES CARTES */

GOPTIONS DEVICE=IBM3279;

PROC GMAP DATA=AFR6 MAP=AFR6 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR5 MAP=AFR5 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR4 MAP=AFR4 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR3 MAP=AFR3 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR2 MAP=AFR2 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR1 MAP=AFR1 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;
PROC GMAP DATA=AFR0 MAP=AFR0 ALL;ID CODE;CHORO D/DISCRETE;PATTERN1 V=E;

RUN;
```

Fig. 13.2. – *Le programme EXPL14 : généralisation des contours et tracé des fonds de carte successifs.*

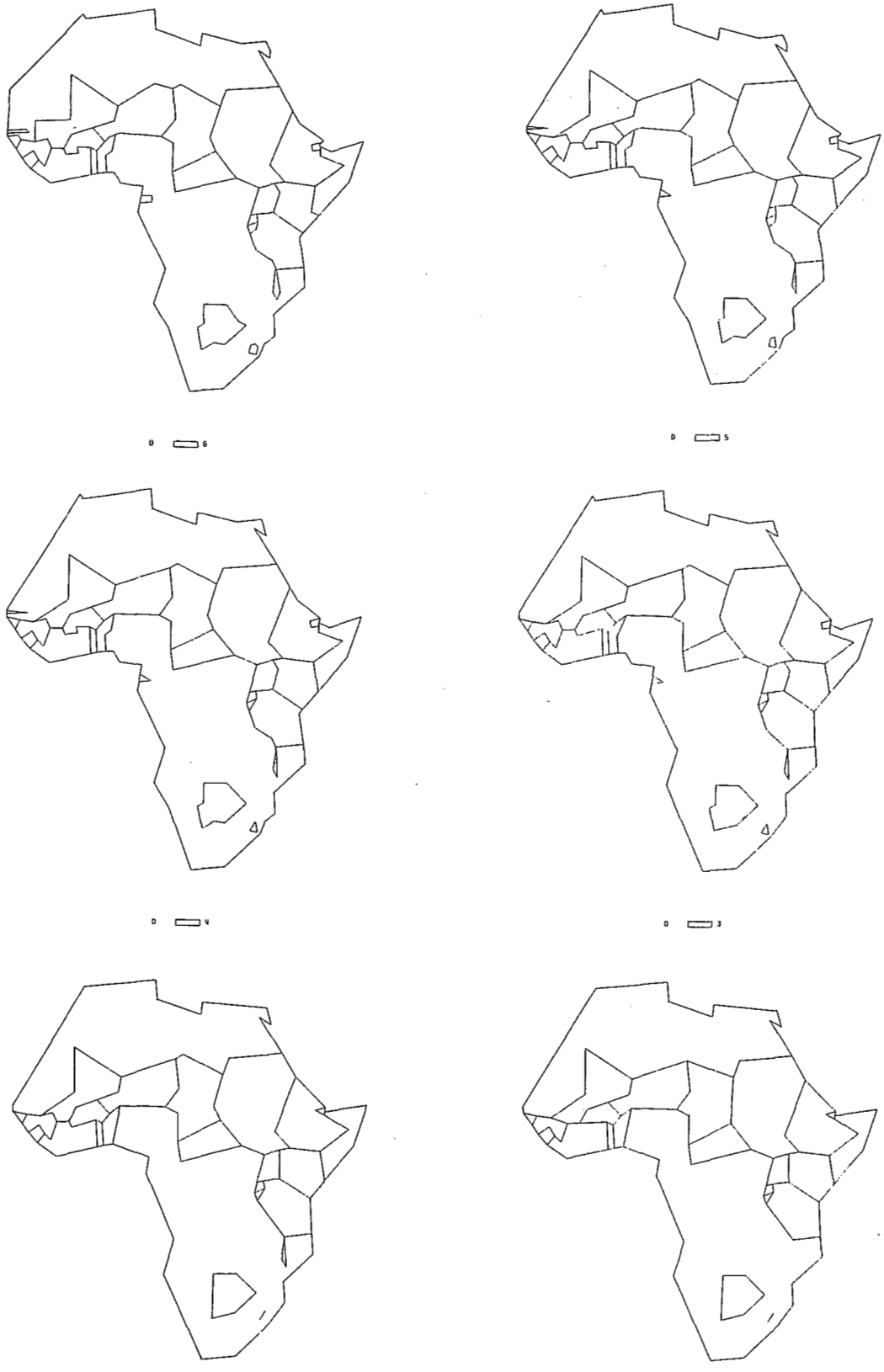
procédure GMAP de tracer la carte ; il faut préalablement adopter un mode de projection approprié à la nature des phénomènes géographiques à représener. SAS fournit un tableau fond de carte du Monde où les contours sont décrits par des coordonnées géographiques exprimées en radians. Ce fond de carte comprenant plus de 150000 points a été généralisé et le résultat figure dans le tableau WORLD de la base de nom physique .STAGBASE.

Il y a deux manières de réaliser ces projections : si les formules sont connues, une simple étape DATA permet de transformer les mesures d'angles en coordonnées sur une carte ; dans le cas contraire, la procédure GPROJECT propose un choix limité à quelques types de projection. Afin de préciser la manière d'opérer, le programme EXPL15 (fig. 13.3) est composé de deux parties principales. Dans la première, quatre étapes DATA appliquent respectivement la projection de Mercator, la projection sinusoïdale, la projection zénitale équidistante et, enfin, la projection de Bonne.

Dans le cas de la projection cylindrique de Mercator (carte 13.3), les méridiens et les parallèles se recoupent à angle droit, et les méridiens sont également parallèles. De ce fait, les surfaces sont énormément grossies aux latitudes élevées. En conséquence, cette technique est à bannir pour représenter des aires d'extension ; elle est très utile, par contre, figurer des directions.

La projection polaire zénitale équidistante (carte 13.5) ne s'applique qu'à un seul hémisphère à la fois. Elle ne déforme pas les distances mesurées à partir du pôle, ni les directions, mais les surfaces sont rès exagérées à proximité de l'équateur. Ce mode de projection doit donc être réservé à la représentation de phénomènes ayant une direction Nord-Sud.

La projection conique de Bonne (carte 13.6) s'applique également à un seul hémisphère à la fois. Elle nécessite le choix préalable d'un parallèle standard auquel le cône sera tangent, et passant de préférence par le centre de la zone à représenter. Les surfaces



Carte 13.2. – *Les tracés des fonds de carte de la densité 6 à la densité 1.*

OBS	CODE	P01	P02	P03	P04	P05	P06
1	GMB	10	11	54	55	.	.
2	GNB	12	13	56	57	.	.
3	GIN	13	14	63	62	61	60
4	SLE	14	15	16	62	63	.
5	MLI	58	59	60	64	65	66
6	HVO	64	76	75	74	73	72
7	TGO	18	19	72	73	.	.
8	BEN	19	20	77	78	79	80
9	NER	68	67	66	81	80	79
10	TCD	84	85	86	89	90	91
11	CAF	86	87	88	89	.	.
12	SDN	49	50	95	91	90	89
13	ETH	46	47	48	49	107	106
14	DJI	44	45	46	108	109	110
15	SOM	40	41	42	43	44	110
16	UGA	96	97	98	103	104	105
17	RWA	97	102	99	98	.	.
18	BDI	102	101	100	99	.	.
19	TZA	37	38	39	103	98	99
20	MWI	117	118	119	120	121	.
21	BWA	122	123	124	125	126	127
22	LSO	131	132	133	134	135	.
23	GNQ	24	25	83	82	.	.
24	AFR	1	2	3	4	5	6

P07	P08	P09	P10	P11	P12	P13	P14
.	.	.	.	.	.	.	.
59	58	56	.	.	.	.	.
.	.	.	.	.	.	.	.
67	68	69	70	71	.	.	.
81	66	65	.	.	.	.	.
.	.	.	.	.	.	.	.
81	72	.	.	.	.	.	.
136	84	93	137	94	.	.	.
92	137	93	.	.	.	.	.
.	.	.	.	.	.	.	.
88	96	105	106	107	.	.	.
113	112	111	110	109	108	.	.
.	.	.	.	.	.	.	.
111	112	113	114	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
100	101	115	116	117	.	.	.
128	129	130	.	.	.	.	.
.	.	.	.	.	.	.	.
7	8	9	10	11	12	13	14

Tab. 13.3. – *Des fragments du tableau de description des contours des polygones.*

sont bien figurées sur une bande Nord-Sud n'excédant pas 80° de longitude et centrée sur un méridien.

Une extension de la projection de Bonne est proposée par la projection sinusoïdale (carte 13.4) également connue sous le nom Sanson-Flamsted. Le parallèle standard étant ici l'équateur, les surfaces sont bien respectées si elles sont comprises à l'intérieur d'une bande allant du pôle Sud au pôle Nord et n'excédant pas 160° de longitude centrée sur un méridien.

Les formules de projection, ainsi que beaucoup d'autres (notamment les projections hémisphèriques équatoriales) sont données dans un excellent article de Dr. Gren paru dans le second numéro de 1985 de la revue Cartographica et intitulé « SAS/GRAPH for cartography : map projections and labelled choropleth maps ». L'auteur y développe également quelques calculs permettant de représenter sur les cartes les parallèles et les méridiens ; plusieurs programmes écrits en langage SAS assortis de quelques « trucs » de programmation complètent cette très intéressante présentation.

La seconde partie du programme EXPL15 fait appel à la procédure GPROJECT. Elle propose trois projections : la projection cônique à deux parallèles standards de Albers et de Lambert ainsi que la projection polaire Gnomonique. On trouvera la présentation de ces techniques assez délicates d'emploi (notamment des deux premières en raison du choix de deux parallèles standards) dans le gros ouvrage de P. Richardus publié en 1972 aux éditions North-Holland, intitulé « Map projections » ; ce livre donne, en outre, les formules pour les calculs des coordonnées.

Les cartes 13.7 et 13.8 présentent respectivement la projection de Albers ett celle de Lambert nécessitant deux appels à la procédure GPROJECT pour le calcul du fond de carte à partir des coordonnées angulaires ; la syntaxe est assez simple :

PROC GPROJECT DATA=nom du tableau fond de carte à projecter

OBS	X	Y	CODE
1	6	177	GMB
2	6	175	GMB
3	20	176	GMB
4	18	178	GMB
5	6	173	GNB
6	12	165	GNB
7	15	171	GNB
8	13	174	GNB
9	12	165	GIN
10	16	159	GIN
11	23	164	GIN
12	27	158	GIN
13	31	154	GIN
14	35	166	GIN
15	32	172	GIN
16	24	170	GIN
17	15	171	GIN
18	16	159	SLE
19	19	155	SLE
20	22	152	SLE
21	27	158	SLE
22	23	164	SLE
23	24	170	MLI
24	32	172	MLI
25	35	166	MLI
26	43	166	MLI
27	48	175	MLI
28	62	180	MLI
29	76	185	MLI
30	78	195	MLI
31	47	215	MLI
32	47	181	MLI
33	24	184	MLI
34	43	166	HVO
35	45	162	HVO
36	53	163	HVO
37	52	167	HVO
38	61	167	HVO
39	65	165	HVO
40	70	171	HVO
41	62	180	HVO
42	48	175	HVO
43	62	149	TGO
44	66	150	TGO
45	65	165	TGO
46	61	167	TGO
47	66	150	BEN
48	72	151	BEN
49	71	162	BEN
50	75	165	BEN
51	77	176	BEN
52	73	174	BEN
53	70	171	BEN
54	65	165	BEN
55	78	195	NER
56	76	185	NER
57	62	180	NER
58	70	171	NER
59	73	174	NER
60	77	176	NER
61	80	187	NER
62	107	175	NER
63	116	187	NER
64	114	217	NER
65	102	210	NER
66	107	175	TCD
67	115	171	TCD
68	115	152	TCD
69	143	166	TCD
70	139	175	TCD
71	145	196	TCD
72	119	210	TCD
73	114	217	TCD
74	116	187	TCD
75	115	152	CAF
76	117	139	CAF
77	157	144	CAF
78	143	166	CAF
79	193	194	SDN
80	184	210	SDN
81	149	207	SDN
82	145	196	SDN
83	139	175	SDN
84	143	166	SDN
85	157	144	SDN
86	172	143	SDN
87	182	146	SDN
88	189	147	SDN
89	179	158	SDN
90	213	174	ETH
91	214	175	ETH
92	198	186	ETH
93	193	194	ETH
94	179	158	ETH
95	189	147	ETH
96	209	145	ETH
97	228	159	ETH
98	212	165	ETH
99	212	169	ETH
100	207	168	ETH
101	206	172	ETH
102	212	171	DJI
103	211	172	DJI
104	213	174	DJI
105	206	172	DJI
106	207	168	DJI
107	212	169	DJI
108	212	126	SOM
109	231	148	SOM
110	240	175	SOM
111	219	168	SOM
112	212	171	SOM
113	212	169	SOM
114	212	165	SOM
115	228	159	SOM
116	209	145	SOM
117	207	128	SOM
118	172	143	UGA
119	167	125	UGA
120	181	126	UGA
121	182	127	UGA
122	186	140	UGA
123	182	146	UGA
124	167	125	RWA
125	165	118	RWA
126	182	121	RWA
127	181	126	RWA
128	165	118	BDI
129	167	113	BDI
130	181	116	BDI
131	182	121	BDI
132	203	91	TZA
133	200	111	TZA
134	202	114	TZA
135	182	127	TZA
136	181	126	TZA
137	182	121	TZA
138	181	116	TZA
139	167	113	TZA
140	171	102	TZA
141	182	95	TZA
142	185	89	TZA
143	185	89	MWI
144	184	89	MWI
145	183	74	MWI
146	186	69	MWI
147	188	74	MWI
148	153	75	BWA
149	138	75	BWA
150	138	62	BWA
151	134	60	BWA
152	137	45	BWA
153	145	50	BWA
154	152	49	BWA
155	166	62	BWA
156	161	64	BWA
157	172	60	LSO
158	169	59	LSO
159	168	54	LSO
160	173	53	LSO
161	174	58	LSO
162	95	136	GNQ
163	94	131	GNQ
164	102	132	GNQ
165	102	136	GNQ
166	174	240	AFR
167	158	238	AFR
168	132	244	AFR
169	131	236	AFR
170	102	246	AFR
171	101	259	AFR
172	56	252	AFR
173	54	255	AFR
174	7	204	AFR
175	6	177	AFR
176	6	175	AFR
177	6	173	AFR
178	12	165	AFR
179	16	159	AFR
180	19	155	AFR
181	22	152	AFR
182	35	143	AFR
183	62	149	AFR
184	66	150	AFR
185	72	151	AFR
186	77	151	AFR
187	81	144	AFR
188	96	142	AFR
189	95	136	AFR
190	94	131	AFR
191	93	125	AFR
192	112	88	AFR
193	105	65	AFR
194	115	49	AFR
195	130	8	AFR
196	152	10	AFR
197	175	32	AFR
198	178	40	AFR
199	185	45	AFR
200	184	59	AFR
201	204	78	AFR
202	203	91	AFR
203	200	111	AFR
204	202	114	AFR
205	212	126	AFR
206	231	148	AFR
207	240	175	AFR
208	219	168	AFR
209	212	171	AFR
210	211	172	AFR
211	213	174	AFR
212	214	175	AFR
213	198	186	AFR
214	193	194	AFR
215	184	210	AFR
216	169	234	AFR
217	177	229	AFR
218	177	234	AFR

Tab. 13.4. – *Le fond de carte bien constitué.*

OUT=nom du tableau fond de carte projeté
PROJECT=nom de la projection
EASTLONG
PARALEL1=parallèle standard le plus petit
PARALEL2=parallèle standard le plus grand ;

ID variable identifiant les polygones ; Les noms de projections possibles sont ALBERS, LAMBERT ou GNOMIC ; EASTLONG signifie que les longitudes sont mesurées de l'Est vers l'Ouest.

OBS	X	Y	CODE	DENSITY
1	6	177	GMB	0
2	6	175	GMB	0
3	20	176	GMB	4
4	18	178	GMB	6
5	6	178	GNB	0
6	12	165	GNB	0
7	15	171	GNB	0
8	13	174	GNB	6
9	12	165	GIN	0
10	16	159	GIN	0
11	23	164	GIN	0
12	27	158	GIN	0
13	31	154	GIN	1
14	35	166	GIN	0
15	32	172	GIN	2
16	24	170	GIN	0
17	15	171	GIN	0
18	16	159	SLE	0
19	19	155	SLE	6
20	22	152	SLE	0
21	27	158	SLE	0
22	23	164	SLE	0
23	24	170	MLI	0
24	32	172	MLI	2
25	35	166	MLI	0
26	43	166	MLI	0
27	48	175	MLI	0
28	62	180	MLI	0
29	76	185	MLI	1
30	78	195	MLI	0
31	47	215	MLI	1
32	47	185	MLI	1
33	24	184	MLI	6
34	43	166	HVO	0
35	45	162	HVO	2
36	53	163	HVO	4
37	52	167	HVO	4
38	61	167	HVO	0
39	65	165	HVO	0
40	70	171	HVO	0
41	62	180	HVO	0
42	48	175	HVO	0
43	62	149	TGO	0
44	66	150	TGO	0
45	65	165	TGO	0
46	61	167	TGO	0
47	66	150	BEN	0
48	72	151	BEN	0
49	71	162	BEN	3
50	75	165	BEN	4
51	77	176	BEN	0
52	73	174	BEN	6
53	70	171	BEN	0
54	65	165	BEN	0
55	78	195	NER	0
56	76	185	NER	1

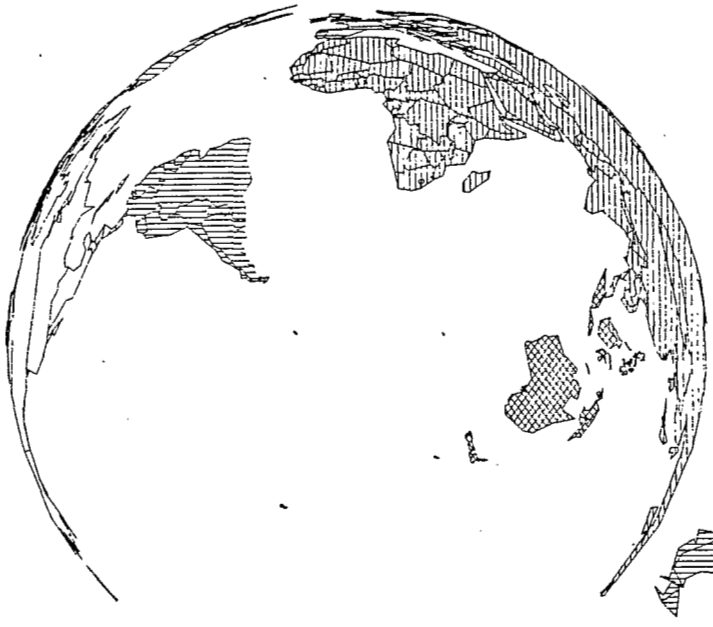
Tab. 13.5. – *Un fragment de fond de carte après le PROC GREDUCE.*

Carte 13.3. – *La projection de Mercator.*

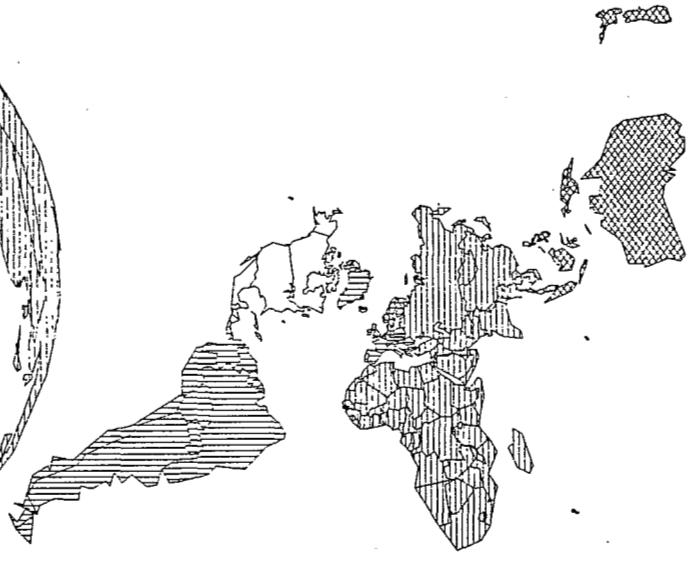
Carte 13.4. – *La projection sinusoïdale.*

Carte 13.5. – *La projection zénitale équidistante pour l'hémisphère Nord.*

Carte 13.6. – *La projection conique de Bonne pour l'hémisphère Nord.*



Carte 13.7. – *La projection conique de Albers.*



Carte 13.8. – *La projection conique de Lambert.*

```
 /*-----------------------------------------------*/
 /* PROJECTIONS DES PAYS DU MONDE DANS ETAPE DATA */
 /*-----------------------------------------------*/

 /*-----> ALLOUER BASE  ET OPTIONS GRAPHIQUES */

X ALLOC DA(.STAGBASE) FI(BASE) SHR;

GOPTIONS DEVICE=IBM3279 BORDER NOTEXT82;

 /*-----> MACRO DE CARTOGRAPHIE */

%MACRO CARTE(TIT);

PROC GMAP DATA=FOND MAP=FOND ALL;ID ID;CHORO CONT/DISCRETE NOLEGEND;
          PATTERN1 C=BLUE V=E;
          TITLE C=BLUE F=TRIPLEX H=5 PCT &TIT;
          FOOTNOTE C=BLUE F=TRIPLEX H=3 PCT 'PHILIPPE WANIEZ - ORSTOM';
RUN;

%MEND CARTE;

 /*-----> COPIER LE TABLEAU */

DATA MONDE;SET BASE.WORLD;
           LONG=X; LAT=Y;CONT=1;

 /*-----> PROJETER DANS ETAPE DATA ET TRACER LA CARTE */


         /*-----> PROJECTION DE MERCATOR */

DATA FOND;SET MONDE;
          SECLAT=1/COS(LAT);
          X=LONG;
          Y=LOG(SECLAT+TAN(LAT));
%CARTE('PROJECTION DE MERCATOR');


         /*-----> PROJECTION SINUSOIDALE */

DATA FOND;SET MONDE;
          X=LONG*COS(LAT);
          Y=LAT;
%CARTE('PROJECTION SINUSOIDALE');


         /*-----> PROJECTION ZENITALE EQUIDISTANTE */

                /*-----> HEMISPHERE NORD */

DATA MOND1;SET MONDE;IF LAT GE 0;
           PI=3.1415927;
           LAT=LAT;

DATA FOND;SET MOND1;
          R=PI/2-LAT;
          X=R*COS(LONG);
          Y=R*SIN(LONG);
%CARTE('PROJECTION ZENITALE EQUIDISTANTE HEMISPHERE NORD');


         /*-----> PROJECTION CONIQUE DE BONNE   */

                /*-----> HEMISPHERE NORD */

DATA MOND1;SET MONDE;IF LAT GE 0;
           PI4=3.1415927/4;
           LAT=LAT;

DATA FOND;SET MOND1;
          SP=PI4;
          COTSP=COS(SP)/SIN(SP);
          R=COTSP+SP-LAT;
          A=LONG*COS(LAT)/R;
          X=R*COS(A);
          Y=R*SIN(A);
%CARTE('PROJECTION CONIQUE DE BONNE HEMISPHERE NORD');


 /*-----------------------------------------------*/
 /* PROJECTIONS PAYS DU MONDE AVEC PROC GPROJECT */
 /*-----------------------------------------------*/

PROC GPROJECT DATA=BASE.WORLD OUT=FOND PROJECT=ALBERS
              EASTLONG PARALEL1=-40 PARALEL2=-60;ID ID;
%CARTE('PROJECTION CONIQUE DE ALBERS');

PROC GPROJECT DATA=BASE.WORLD OUT=FOND PROJECT=LAMBERT
              EASTLONG PARALEL1=45 PARALEL2=55;ID ID;
%CARTE('PROJECTION CONIQUE DE LAMBERT');
```

Fig. 13.3. – *Le programme EXPL15 : exemples de calculs pour les projections.*

14. Les cartes choroplèthes

Les cartes choroplèthes servent à représenter des valeurs relatives. Elles sont constituées par une composante géographique zonale (le fond de carte) et une composante quantitative qui est implantée sur chaque zone. Les variations des valeurs de la variable à représenter sont matérialisées par des trames de densités variant du très clair au très foncé et en une seule couleur, le vide étant habituellement réservé aux valeurs manquantes. L'œil humain n'étant pas capable de différencier les petites variations, il faut choisir des trames de densités assez différentes et en nombre limité. Cela conduit à transformer l'échelle de rapport en échelle d'intervalles. Chaque intervalle est ainsi représenté par la trame qui lui correspond. C'est la procédure GMAP qui réalise le tracé des cartes choroplèthes.

14.1. La discrétisation des variables

Il existe principalement trois techniques pour rendre discrètes des variables continues ; à chacune d'elle correspond un procédure SAS. Dans tous les cas, il est nécessaire de fixer au départ le nombre de modalités à représenter.

14.1.1. Les seuils fixés à priori : la procédure FORMAT.

La technique des seuils fixés a priori est à réserver à la représentation de variables dont les distributions ont des formes très complexes ne pouvant être simplement résumées par les paramètres courants des statistiques descriptives (moyenne et écart-type, médiane et quantiles). Cette technique peut parfois servir aussi lorsque les seuils ont une signification précise comme c'est le cas, par exemple pour le traitement des statistiques électorales où la valeur 50 % des suffrages exprimés ne peut dans la plupart des cas être remplacée par aucune autre.

La procédure FORMAT génère un ou plusieurs masques de recodage ; ceux-ci ne transforment pas les données à proprement parler, mais ils indiquent à la procédure GMAP comment il faut voir la variable – comment il faut la découper – au moment précis de sa représentation. Chaque masque de recodage doit avoir un nom (il est conseillé de commencer par FMT pour lever toute ambiguité possible avec un nom de variable). La syntaxe de la procédure est la suivante (pour q modalités) :

```
PROC FORMAT ;
      VALUE nom du premier format
            première borne – seconde borne =
            première modalité

            q-1ième borne – qième borne =
            qième modalité ;
      VALUE nom du second format
            première borne – seconde borne =
            première modalité
            . . . . . .
            q-1ième borne – qième borne =
            qième modalité ;
```

Notons que si la première borne est la valeur minimale, elle peut prendre la valeur LOW ; de même, si la qième borne est la valeur maximale, elle peut prendre la valeur HIGH. Par exemple, pour générer deux masques de recodage relatifs à la variable CADPUB82 du tableau permanent INDIC en trois classes, il faut écrire le programme :

```
PROC FORMAT ;
      VALUE FMTCADA
            LOW – 70 ='< 70 %'
            70 – 85 ='70 – 85 %'
            85 – HIGH ='> 85 %' ;
      VALUE FMTCADB
            LOW – 85 ='< 85 %'
            85 – 95 ='85 – 95 %'
            95 – HIGH ='> 95 %' ;
```

Lors de l'exécution, deux messages indiqueront sur le journal de bord que ces formats ont bien été définis et qu'il est désormais possible d'y faire référencee dans la procédure GMAP. Cela se fera en ajoutant aux instructions de cartographie l'instruction :

FORMAT nom de la variable à cartographier nom du format. ;

Notons que le nom du format doit toujours être immédiatement suivi d'un point (.) qui indique à l'analyseur d'instructions qu'il s'agit d'un nom de format et non pas d'un nom de variable.

14.1.2. Le centrage et la réduction : la procédure STANDARD.

Le centrage et la réduction consistent à exprimer les valeurs sur un échelle standardisée ayant une moyenne nulle et un écart-type égal à l'unité. Cette opération revient à soustraire à chaque valeur la moyenne et à diviser le résultat par l'écart-type. Une fois opérée cette transformation, on choisira un nombre impair de modalités (3, 5 ou 7 en général) de manière à centrer respectivement la seconde modalité, la troisième ou la quatrième sur la moyenne. Il faudra donc nécessairement générer un masque de recodage après cette standardisation pour pouvoir utiliser ensuite la procédure GMAP. Voici la syntaxe de la procédure STANDARD :

```
PROC STANDARD DATA=nom du tableau à transformer
              OUT=nom du tableau transformé
              MEAN=0 STD=1 ;
```

Pour discrétiser la variable CADPUB82 du tableau permanentINDIC à l'aide de la procédure STANDARD, il faut procéder à deux étapes PROC :

```
PROC STANDARD DATA=BASE.INDIC
OUT=INDICSTD MEAN=0 STD=1 ;
PROC FORMAT ;VALUE FMTSTD
              LOW --0.5 ='moins de 0.5 écart-type'
              -0.5 -0.5 ='-0.5 à 0.5 écart-type'
              0.5 -HIGH='plus de 0.5 écart-type' ;
```

Lors de l'utilisation de la procédure GMAP, le nom du tableau contenant les données sera INDICSTD et le nom du format FMTSTD. .

14.1.3. Les quantiles : la procédure RANK

Lorsque les distributions des variables sont très dissymétriques, le centrage et la réduction ne sont plus légitimes pour les discrétiser. Il est préférable d'utiliser les quantiles ; à nouveau, il est faut choisir de préférence un nombre impair de classe (3, 5 ou 7 en général) de manière à centrer respectivement la seconde, la troisième ou la quatrième sur la médiane. Les modalités ainsi définies auront le même nombre d'observations, aux ex-aequo près. Il faudra ensuite générer un masque de recodage exprimant le nombre de modalités choisi. la syntaxe de la procédure RANK est la suivante :

```
PROC RANK DATA=nom du tableau à transformer
          OUT=nom du tableau transformé
          GROUPS=nombre de modalités ;
```

La discrétisation de la variable CADPUB82 du tableau permanent INDIC sera donc réalisée en deux étapes PROC :

```
PROC RANK DATA=BASE.INDIC
          OUT=INDICRAK
          GROUPS=3 ;
PROC FORMAT ; VALUE FMTRK
              1='1er tiers'
              2='2e tiers'
              3='3e tiers' ;
```

A l'appel de la procédure GMAP, le nom du tableau contenant les données sera INDICRAK et le nom du format sera FMTRK. .

14.2. Le choix de trames

A chaque modalité de la variable discrétisée doit correspondre une trame. La densité des trames doit croître avec les valeurs des modalités pour figurer les variations des valeurs de la variable représentée. La définition d'une trame se fait à l'aide de l'instruction PATTERN suivie du numéro d'ordre de la modalité qu'elle représente. La syntaxe de l'insruction PATTEERN (ici pour la première trame) permet de choisir la couleur et le type de la trame :

PATTERN1 C=couleur V=type de trame L= type de ligne ;

Il faut donc définir autant de PATTERN qu'il y a de modalités composant le masque de recodage de la variable.

Le type de trame est fixé par le paramètre V. Sa valeur peut être :

E : Empty, vide
S : Solid, plein, la surface du polygone sera remplie par la couleur choisie par le paramètre C= .

ou bien une chaîne de caractères composée de quatre valeurs fixant l'orientation des hachures et leur espacement de la manière suivante : La lettre M indique qu'il s'agit d'une trame composée ; elle est suivie par un chiffre de 1 à 5 donnant sa densité (1=le plus clair, 5=le plus foncé juste avant Solid). Ensuite, une lettre indique le sens des hachures : R pour Right, orientation à droite, L pour Left, orientation à gauche et X pour Crosshatched, croisillons. Voici quelques exemples de types de trame :

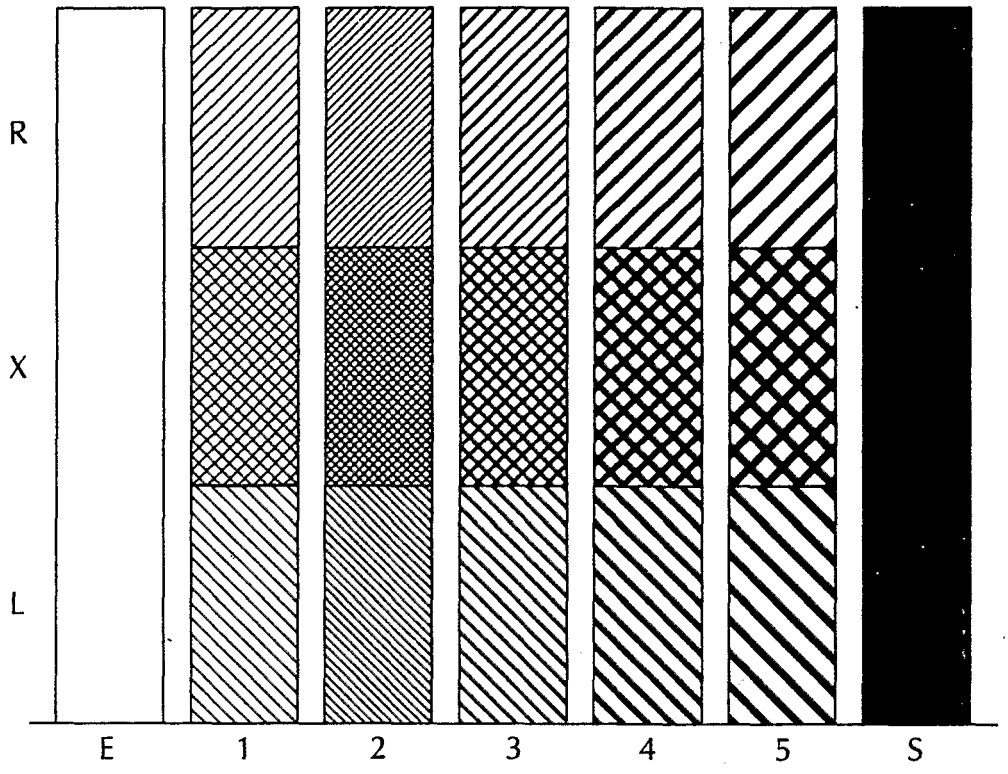


Fig. 14.1. – *Des exemples de trames.*

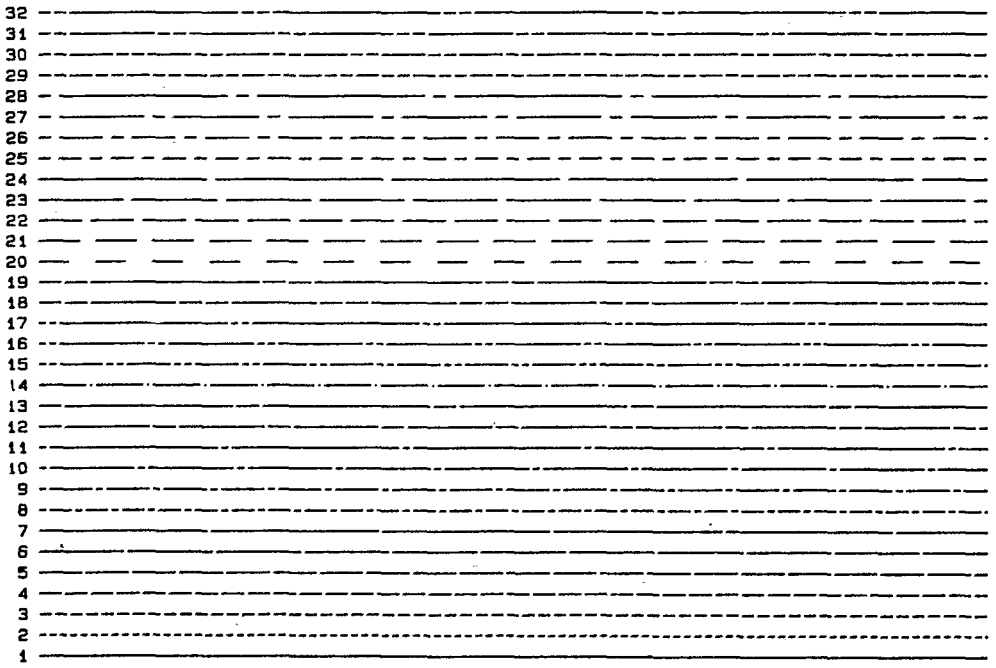
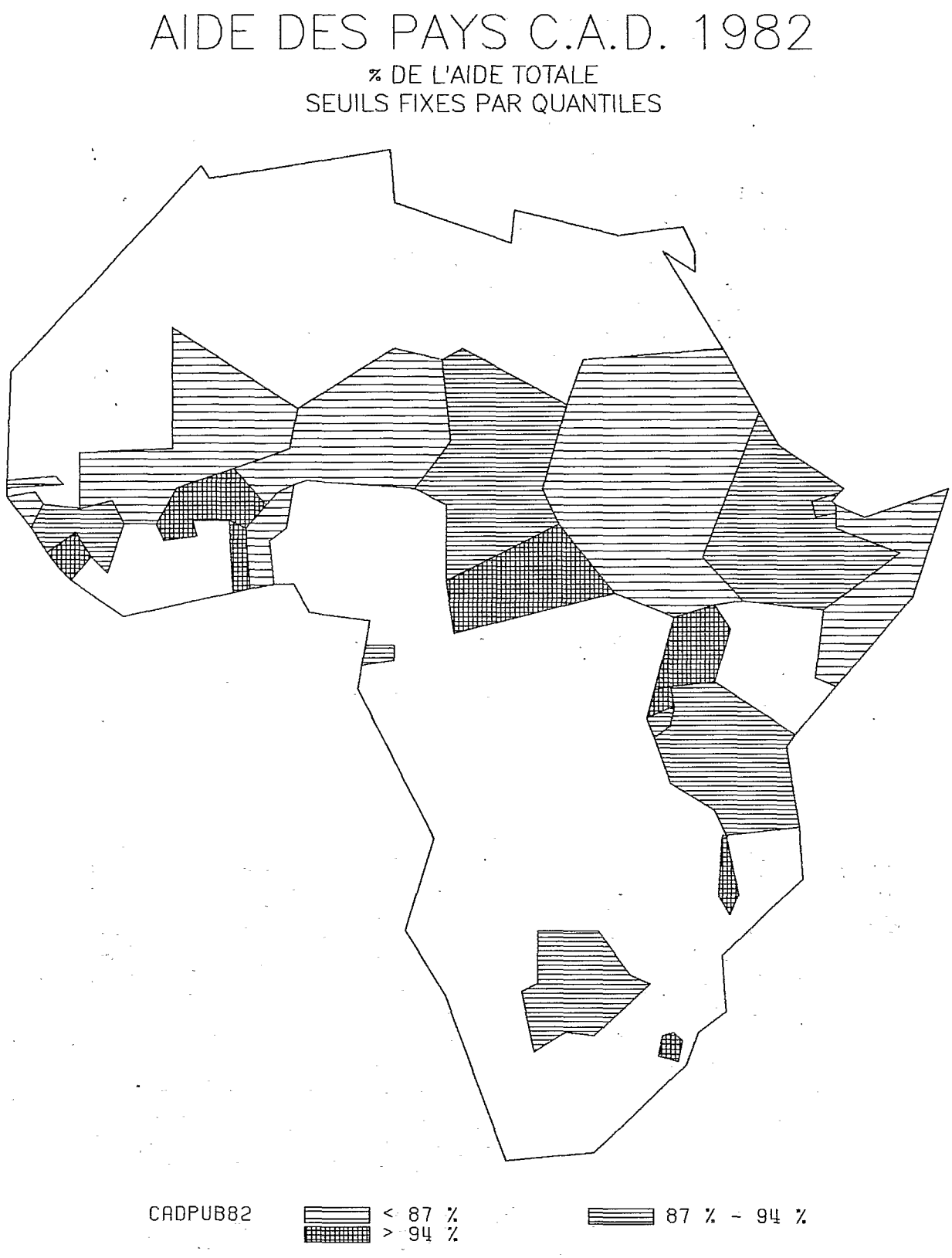


Fig. 14.2. – *Les différents types de lignes.*

```
/*-----------------------------------------*/
/* TRACE DE CARTES CHOROPLETHES AVEC GMAP */
/*-----------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) SHR;

 /*-----> CHOIX OPTIONS GRAPHIQUES */

GOPTIONS DEVICE=IBM3279 BORDER NOTEXT82;

 /*-----> SEUILS FIXES A-PRIORI */

PROC FORMAT;
     VALUE FMTCADA LOW-85='< 85 %'
                    85-95='85 % - 95 %'
                 95-HIGH='> 95 %';

PROC GMAP DATA=BASE.INDIC MAP=BASE.FONDAFRI ALL;
     ID CODE;
     CHORO CADPUB82/DISCRETE;
     FORMAT CADPUB82 FMTCADA. ;
     TITLE1 C=BLUE F=SIMPLEX H=4 PCT 'AIDE DES PAYS C.A.D. 1982';
     TITLE2 C=BLUE F=SIMPLEX H=2 PCT '% DE L''AIDE TOTALE';
     TITLE3 C=BLUE F=SIMPLEX H=2 PCT 'SEUILS FIXES A-PRIORI';
     FOOTNOTE C=BLUE F=SIMPLEX H=2 PCT 'P. WANIEZ - ORSTOM';
     PATTERN1 C=BLUE V=M2R0;
     PATTERN2 C=BLUE V=M4R0;
     PATTERN3 C=BLUE V=M4X0;

 /*-----> SEUILS FIXES PAR CENTRAGE ET REDUCTION */

PROC STANDARD DATA=BASE.INDIC OUT=INDIC MEAN=0 STD=1;
PROC FORMAT;
     VALUE FMTSTD  LOW - -0.5  ='< 83.3 %'
                  -0.5 - 0.5   ='83.3 % - 94.5 %'
                   0.5 - HIGH  ='> 94.5 %';

PROC GMAP DATA=INDIC MAP=BASE.FONDAFRI ALL;
     ID CODE;
     CHORO CADPUB82/DISCRETE;
     FORMAT CADPUB82 FMTSTD. ;
     TITLE1 C=BLUE F=SIMPLEX H=4 PCT 'AIDE DES PAYS C.A.D. 1982';
     TITLE2 C=BLUE F=SIMPLEX H=2 PCT '% DE L''AIDE TOTALE';
     TITLE3 C=BLUE F=SIMPLEX H=2 PCT 'SEUILS FIXES PAR STANDARDISATION';
     FOOTNOTE C=BLUE F=SIMPLEX H=2 PCT 'P. WANIEZ - ORSTOM';
     PATTERN1 C=BLUE V=M2R0;
     PATTERN2 C=BLUE V=M4R0;
     PATTERN3 C=BLUE V=M4X0;

 /*-----> SEUILS FIXES PAR QUANTILES */

PROC RANK DATA=BASE.INDIC OUT=INDIC GROUPS=3;
PROC FORMAT;
     VALUE FMTRANK 0='< 87 %'
                   1='87 % - 94 %'
                   2='> 94 %';
PROC GMAP DATA=INDIC MAP=BASE.FONDAFRI ALL;
     ID CODE;
     CHORO CADPUB82/DISCRETE;
     FORMAT CADPUB82 FMTRANK. ;
     TITLE1 C=BLUE F=SIMPLEX H=4 PCT 'AIDE DES PAYS C.A.D. 1982';
     TITLE2 C=BLUE F=SIMPLEX H=2 PCT '% DE L''AIDE TOTALE';
     TITLE3 C=BLUE F=SIMPLEX H=2 PCT 'SEUILS FIXES PAR QUANTILES';
     FOOTNOTE C=BLUE F=SIMPLEX H=2 PCT 'P. WANIEZ - ORSTOM';
     PATTERN1 C=BLUE V=M2R0;
     PATTERN2 C=BLUE V=M4R0;
     PATTERN3 C=BLUE V=M4X0;
```

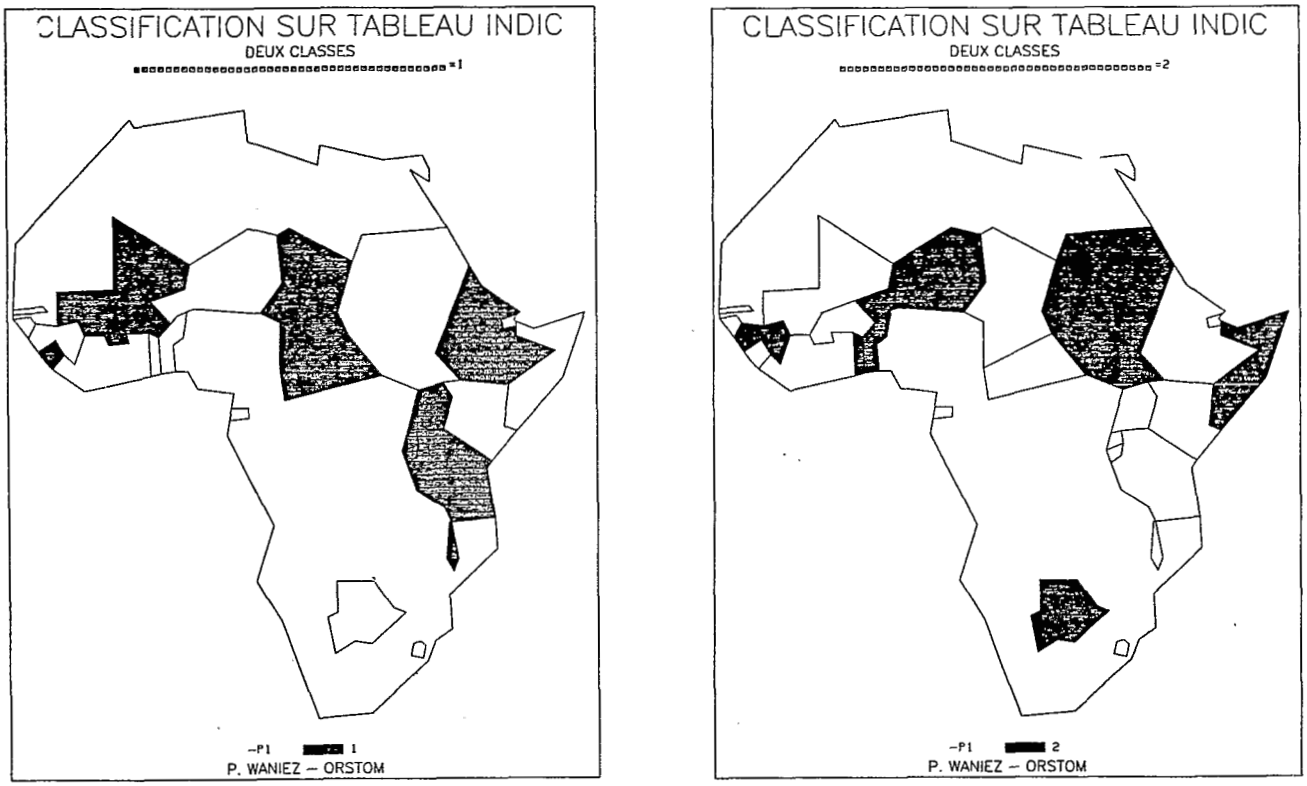
Fig. 14.3. – *Le programme EXPL16 : tracé de cartes choroplèthes avec GMAP.*



Carte 14.1. – ***Une carte choroplèthe de la variable CADPUB82.***

```
/*-----------------------------------------*/
/* TRACE D'UNE PARTITION APRES C.A.H. ADDAD*/
/*-----------------------------------------*/

 /*-----> ALLOCATION DE LA BASE */

X ALLOC DA(.STAGBASE) FI(BASE) SHR;

 /*-----> CHOIX OPTIONS GRAPHIQUES */

GOPTIONS DEVICE=IBM3279 BORDER NOTEXT82;

 /*-----> UNE CARTE POUR TOUTE LA CAH */

PROC GMAP DATA=BASE.CAHPART MAP=BASE.FONDAFRI ALL;
     ID CODE;
     CHORO _P1/DISCRETE;
     TITLE C=BLUE F=SIMPLEX H=4 PCT 'CLASSIFICATION SUR TABLEAU INDIC';
     TITLE2 C=BLUE F=SIMPLEX H=2 PCT 'DEUX CLASSES';
     FOOTNOTE C=BLUE F=SIMPLEX H=2 PCT 'P. WANIEZ - ORSTOM';
     PATTERN1 C=PINK V=S;PATTERN2 C=CYAN V=S;

 /*-----> UNE CARTE PAR CLASSE DE CAH */

PROC SORT DATA=BASE.CAHPART OUT=CAHPART;BY _P1;

PROC GMAP DATA=CAHPART MAP=BASE.FONDAFRI ALL;
     ID CODE;
     CHORO _P1/DISCRETE;
     TITLE C=BLUE F=SIMPLEX H=4 PCT 'CLASSIFICATION SUR TABLEAU INDIC';
     TITLE2 C=BLUE F=SIMPLEX H=2 PCT 'DEUX CLASSES';
     FOOTNOTE C=BLUE F=SIMPLEX H=2 PCT 'P. WANIEZ - ORSTOM';
     PATTERN1 C=GREEN V=S;
     BY _P1;
```

Fig. 14.4. – *Le programme EXPL17 : tracé de classes de CAH.*



Carte 14.2. – *Les cartes des classes de la CAH du chapitre 9.*

M1L45 trame la plus claire avec hachures orientées à gauche, à 45°.
M5X90 trame la plus foncée avec hachures en croisillons verticaux et horizontaux.
M3R60 trame médiane avec hachures orientées à droite, à 60°.

La figure 14.1 extraite du manuel de référence SAS/GRAPH montre les différentes combinaisons possibles.

Enfin, le type de ligne est un numéro qui doit être choisi dans la table des lignes (fig. 14.2) ; il doit prendre une valeur comprise entre 1 et 32 ; si ce paramètre est omis, le type de ligne est 1.

En résumé, voici un exemple de trames pour trois classes :

```
PATTERN1 C=BLUE V=M2R0 ;
PATTERN2 C=BLUE V=M4R0 ;
PATTERN3 C=BLUE V=M4X0 ;
```

Notons enfin que l'effet visuel produit par ces trames n'est pas absolument indépendant de l'unité graphique choisie ; il est dans tous les cas indispensable de faire un essai pour vérifier la bonne adéquation valeurs statistiques/trames.

14.3. Le tracé de la carte : la procédure GMAP

Le tracé d'une carte choroplèthe est assuré par la procédure GMAP ; il faut une instruction PROC fixant le nom du tableau des données et celui du tableau fond de carte, une instruction ID indiquant le nom de la variable identifiant les observations, une instruction CHORO donnant le nom de la variable à cartographier, une instruction FORMAT pour choisir le masque de recodage de la variable à cartographier, une ou plusieurs instructions PATTERN afin de choisir les trames et, enfin, une ou plusieurs instructions TITLE ou FOOTNOTE pour finir l'habillage. Voici donc la séquence d'instructions type pour tracer une carte choroplèthe :

```
PROC GMAP DATA=nom du tableau des données
MAP=nom du tableau fond de carte
ALL ;
ID nom de la variable identifiant les observations ;
CHORO nom de la variable à cartographier/
DISCRETE COUTLINE=couleur des contours ;
FORMAT nom de la variable à cartographier nom du format. ;
PATTERN1 C=couleur V=trame ;
PATTERN2 C=couleur V=trame ;
PATTERN3 c=couleur V=trame ;
TITLE1 C=couleur F=police de caractères H=hauteur PCT 'titre n° 1' ;
FOOTNOTE C=couleur F=police de caractères H=hauteur PCT 'note n° 1' ;
```

Le programme EXPL16 (fig. 14.3) présente le tracé de trois cartes à l'aide de la procédure GMAP de la variable CADPUB82 discrétisée à l'aide des trois techniques rappelées ci-dessus. La carte 14.1 représente la variable CADPUB82 cartographiée à l'aide de la technique des quantiles.

14.4. Tracer un fond de carte vide

Il est rare que la numérisation du fond de carte se fasse sans erreur. Pour vérifier la validité des contours, il est indispensable de pouvoir afficher le fond de carte vide. La procédure GMAP peut aussi réaliser ce tracé en appliquant une petite astuce qui consiste à remplacer l'option DISCRETE de l'instruction CHORO par l'option LEVELS=1 et à définir une trame vide. Voici le principe de ce programme :

```
PROC GMAP DATA=nom du tableau fond de carte
          MAP=nom du tableau fond de carte ALL ;
ID nom de la variable identifiant les observations ;
CHORO X/LEEVELS=1 ;
PATTERN1 V=E C=YELLOW ;
```

Dans ce cas, les contours seront tracés en jaune. Notons que le tableau DATA est identique au tableau MAP qui contient le fond de carte.

14.5 La représentation des classes d'une hiérarchie

Une extension intéressante consiste en l'application de la procédure GMAP à la représentation des classes d'une CAH. Il y a deux manières de s'y prendre : soit affecter une couleur à chaque classe (une instruction PATTERN par classe est alors nécessaire), soit tracer une carte par classe à l'aide de l'instruction BY. Le programme EXPL17 (fig. 14.4) présente les deux cas sur une partition en deux classes issue de la CAH réalisée à partir du tableau permanent INDIC au chapitre n° 9 (programme EXPL9, fig.9.7). Deux cartes (carte 14.2) représentent chacune une classe. Notons que la variable contenant la partition et dont le nom figure à la fois dans l'instruction CHORO et l'instruction BY doit obligatoirement faire l'objet d'un tri préalable par la procédure SORT.

Conclusion

La mise en œuvre de l'outil informatique ne s'improvise pas : formation, équipement, coûts sont à prendre en compte dans les budgets des programmes de recherche. Mais l'attention doit être attirée sur le fait qu'il est très difficile de maîtriser les dépenses et cela pour plusieurs raisons. En premier lieu, l'acquisition de données statistiques informatisées (ou bien la mise sur support informatique de celles qui ne le sont pas) coûte cher, sans qu'il soit toujours possible de fixer a priori le montant de la dépense. Ensuite, il est rare d'obtenir les bons résultats dès la première analyse ; il faut souvent itérer les traitements en fonction des résultats précédents. Enfin, de nouvelles possibilités, tant pour les matériels que pour les logiciels apparaissent chaque mois ; il serait regrettable de ne pas en profiter.

On reprochera peut-être à ce manuel de trop mettre l'accent sur SAS. Il fallait bien choisir un progiciel si l'on voulait éviter de parler dans le vide, de présenter ce qu'il faut faire sans dire comment le faire ; dans cette perspective, SAS était l'un des meilleurs choix possibles. D'autre part, tous ceux qui, durant quelques années, auront programmé en langage SAS ne devraient pas avoir de difficulté à s'adapter à d'autres systèmes. L'important est que le lecteur ait réellement compris et pratiqué les différentes opérations, qu'il ne procède pas par imitation.

Enfin, si comme on peut le penser, SAS devient un vrai standard dans le domaine du traitement des données statistiques, il serait souhaitable que les futures publications méthodologiques s'accompagnent des programmes SAS permettant la mise à l'épreuve de ces méthodes. Le présent manuel ne serait alors qu'une simple introduction à ces travaux.

Annexes

Annexe n° 1 : principales abréviations

ADDAD	:	Association pour le Développement et la diffusion de l'Analyse des Données
ACP	:	Analyse en Composantes Principales
AFC	:	Analyse Factorielle des Correspondances
CAH	:	Classification Ascendante Hiérarchique
TSO	:	Time Sharing Option ; sous-système de temps partagé du système d'exploitation Operating System/Multiple Virtual Storage (OS/MVS) d'IBM
SAS	:	Statistical analysis System ; système d'analyse statistique développé par SAS Institute Inc.

Annexe n° 2 : index des principales instructions SAS

Instruction du langage	Chapitre du livre	Manuel de référence	Chapitre et page du manuel	
Instruction TSO				
ALLOCATE	4.1.3.			
DELETE	4.1.4			
Instructions pouvant figurer n'importe où dans le programme				
FOOTNOTE	12.2.	BASICS	12	410
GOPTIONS	12.1	GRAPH	6	91
TITLE	6.2.	BASICS	12	441
Instructions de l'étape DATA				
ABS etc.	5.3.4.	BASICS	6	229
DATA	4.2.	BASICS	3	25
DROP	5.1.1.	BASICS	4	76
IF	5.1.3.	BASICS	4	96
INFILE	4.2.1.	BASICS	4	99
INPUT	4.2.2.	BASICS	4	130
KEEP	5.1.1.	BASICS	4	149
LABEL	4.2.3.	BASICS	4	150
LENGTH	5.3.1.	BASICS	4	152
MERGE	5.2.	BASICS	4	167
RENAME	5.1.5.	BASICS	4	194
SET	5.1.1.	BASICS	4	205
- + * /	5.3.3.	BASICS	5	219
Instructions de l'étape PROC				
BY	6.2.	BASICS	9	337
MODEL	6.2./8.2.4.	BASICS	9	345
OUTPUT	6.2./8.2.4.	BASICS	9	346
PATTERN	14.2.	GRAPH	5	57
PROC	6.2.	BASICS	8	331
RUN	6.3.2.	BASICS	12	437
VAR	6.2.	BASICS	9	349

Instruction du langage	Chapitre du livre	Manuel de référence	Chapitre et page du manuel	
Procédures				
CORR	8.2.2.	BASICS	32	861
FORMAT	14.1.1.	BASICS	35	913
GMAP	14.3.	GRAPH	13	243
GPROJECT	13.3.	GRAPH	17	311
GREDUCE	13.2.	GRAPH	18	319
MEANS	7.2.	BASICS	38	959
PLOT	8.2.1.	BASICS	42	985
PRINTTO	6.3.	BASICS	44	1019
UNIVARIATE	7.1.	BASICS	54	1181
RANK	14.1.3.	STATISTICS	30	647
REG	8.2.4.	STATISTICS	31	655
RSQUARE	8.2.3.	STATISTICS	32	711
STANDARD	14.1.2.	STATISTICS	30	647
Bibliothèque ADDAD				
ADDAD	9.1.	ADDAD		
ANCOMP	9.2.	ADDAD		
ANCORR	9.2.	ADDAD		
CAH2CO	9.3.1.	ADDAD		
CLACAH	9.3.2.	ADDAD		
GRAPHE	9.2.	ADDAD		
PARAM	9.2.	ADDAD		
Macro langage				
%GOTO	10.1.2.	BASICS	19	665
%LET	10.1.2.	BASICS	19	670
%MACRO	10.1.1.	BASICS	19	645
%MEND	10.1.1.	BASICS	19	645
Langage Matriciel				
FETCH	10.2.	IML	Annexe n° 3	
MATRIX	10.2.	IML	Annexe n° 3	
OUTPUT	10.2.	IML	Annexe n° 3	
PRINT	10.2.	IML	4	146

ORSTOM Éditeur
Dépôt légal : 3e trim. 1986

Cet ouvrage est une initiation pratique au traitement informatique des tableaux statistiques de la forme observations/variables, et en particulier des matrices d'information spatiale où les observations sont des unités spatiales (départements, pays, etc.)

Les 14 chapitres sont ventilés en trois parties. La première montre comment organiser l'information en base de données, ce qui nécessite le choix d'un progiciel. La seconde partie conduit, en partant de l'analyse statistique descriptive univariée, à la modélisation par régression et à l'analyse de données. Enfin, l'ensemble de la troisième partie introduit le lecteur à l'utilisation de l'informatique graphique, et en particulier à la cartographie thématique automatique.

La totalité des exposés s'appuie sur le « Statistical Analysis System » (SAS) largement diffusé en France dans les gros centres informatiques, mais aussi sur les micro-ordinateurs IBM PC et compatibles. 19 programmes écrits en langage SAS et un thème d'étude, l'endettement des pays les moins avancés d'Afrique facilitent l'apprentissage.

Ce livre s'adresse à tous ceux, chercheurs ou ingénieurs, qui ont à traiter des données statistiques du domaine des sciences sociales. Les étudiants des second et troisième cycles universitaires en géographie, sociologie et sciences économiques, ainsi qu'en mathématiques appliquées aux sciences sociales devraient y trouver les clés nécessaires au traitement des données de leurs mémoires et thèses. Enfin, les sociétés disposant de SAS y verront le moyen de simplifier l'accès de leur personnel à leur infocentre.

Géographe et sociologue, Philippe Waniez a été chargé du développement du secteur informatique recherche en sciences sociales à l'Université Paris-X Nanterre. Aujourd'hui attaché de recherche à l'ORSTOM, il participe à la réalisation d'un système d'information géographique à vocation agro-pastorale au Brésil (programme SISGEO).

ISSN : 0071-9021
ISBN : 2-7099-0815-8
Éditions de l'ORSTOM
70, route d'Aulnay F-93140 BONDY