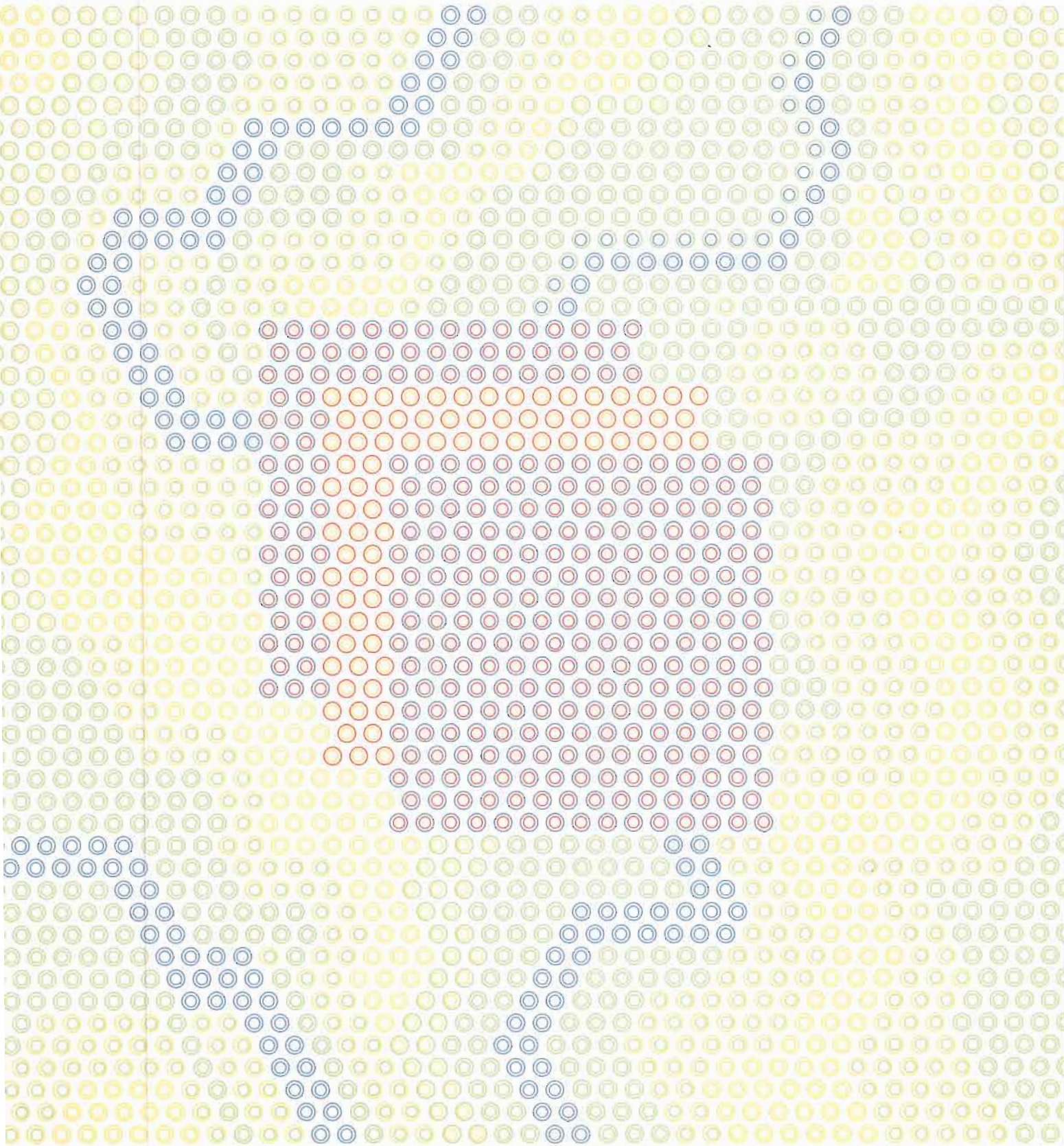
Micheline Cosinschi et Philippe Waniez

RECLUS

Pratique de l'analyse statistique SAS sur PC/PS, mini et gros systèmes



collection reclus modes d'emploi n° 15

Collection Reclus Modes d'Emploi n°15

Pratique de l'analyse statistique SAS sur PC/PS, mini et gros systèmes

Micheline Cosinschi
Institut de Géographie de l'Université de Lausanne

Philippe Waniez
ORSTOM, Maison de la Géographie de Montpellier

GIP
RECLUS

MAISON DE LA GEOGRAPHIE, MONTPELLIER

Pratique de l'analyse statistique, SAS sur PC/PS, mini et gros systèmes /
COSINSCHI Micheline; WANIEZ Philippe
Montpellier: G.I.P. RECLUS, 1989. — 176 p., 43 fig.; 30 cm. —
(Collection Reclus Modes d'Emploi n°15)
ISBN 2-86912-027-2
ISSN 0298-9689

Ce fascicule a été rédigé par Micheline Cosinschi, Agrégée de Faculté à l'Université de Lausanne et Philippe Waniez, Chargé de recherche à l'ORSTOM à la Maison de la Géographie de Montpellier.

Cartographie, traitement de texte et mise en page des auteurs.

Secrétariat de publication: Régine Vanduick

Direction: Roger Brunet

G.I.P. RECLUS, Maison de la Géographie, 17 rue Abbé de l'Epée,
34000 Montpellier
Tél. 67.72.46.10; FAX 67.72.64.04.

Table

Avant-propos

«Pratique de l'Analyse Statistique» est le fruit d'une volonté commune à ses auteurs de proposer un manuel d'initiation au logiciel SAS*, en français, destiné à tous ceux, étudiants, chercheurs, enseignants, etc., qui ont à traiter des données statistiques dans le cadre de leurs travaux de recherche ou d'études.

Cet ouvrage reprend, très partiellement, la publication réalisée en 1986, conjointement par l'ORSTOM et le GIP RECLUS, intitulée «Les Données et le Territoire, Initiation au traitement informatique des données spatialisées». Il en diffère sur de nombreux et importants points.

En premier lieu, on a cherché à mieux définir les conditions scientifiques dans lesquelles s'opère le passage de l'observation des objets aux données statistiques et à l'informatisation. Le premier chapitre montre la place qu'occupent la statistique et l'informatique dans un processus de recherche et rappelle quelques précautions élémentaires dont devrait s'entourer tout analyste de données avant de brancher son ordinateur.

La seconde différence vient du choix des données supportant la présentation du logiciel. Il ne s'agit plus ici de quelques données-prétextes, mais d'un véritable corpus d'une vingtaine de variables démographiques et sociales relatives aux communes du Canton de Vaud. Provenant des travaux de M. Cosinschi sur la Suisse, elles facilitent la présentation des principales étapes du conditionnement et du traitement informatique des matrices d'informations spatiales, type de tableau de données pouvant être efficacement analysé avec SAS.

Enfin la présentation du logiciel à proprement parler a été complètement refondue. Tous les chapitres ont été remaniés en fonction des principales remarques et critiques faites à propos de l'ouvrage paru en 1986. Le plan a été repensé et la présentation du texte et des figures complètement modifiée. Cela permet notamment d'utiliser ce livre dans le cadre de l'enseignement, sous forme de stages durant lesquels l'essentiel des procédures offertes par SAS peuvent être comprises par les élèves, sans les obliger à noter tous les détails de la syntaxe et de la sémantique du langage SAS.

Micheline COSINSCHI-Meunier, diplômée de l'Université d'Ottawa, Canada, est depuis 1973 enseignante à l'Institut de Géographie de l'Université de Lausanne (IGUL). D'abord maître-assistante puis agrégée de Faculté, elle est chargée de la formation des étudiants en méthodes quantitatives et en cartographie appliquées à la géographie humaine. Aux premières loges de la pratique de l'analyse quantitative en géographie dès le début des années soixante-dix aux côtés de Jean-Bernard Racine, elle a largement expérimenté ces techniques alors nouvelles et les a appliquées à des recherches géographiques sur les lieux centraux du Grand Montréal. Aujourd'hui, elle collabore aux travaux de l'équipe de recherche lausannoise sur la Suisse, au sein de laquelle elle assure plus précisément des tâches de construction et de mise à jour des bases de données statistiques, au niveau communal, sur l'ensemble du territoire helvétique. Son expérience en matière d'analyse quantitative des données géographiques lui permet d'apprécier les possibilités respectives des principaux logiciels disponibles sur le marché tant sur micro-ordinateur (Systat, Extatix, Statview, etc.) que sur gros système (SPSS et SAS en particulier).

Philippe WANIEZ est docteur en géographie de l'Université Paris-Sorbonne Paris-IV. De 1979 à 1985, il est chargé du développement du secteur «informatique et recherche en sciences humaines» au centre de calcul de l'Université de Nanterre. Ces fonctions lui permettent d'acquérir une pratique approfondie des méthodes et techniques de traitement informatique des données économiques et sociales. A partir de 1986, en tant que chargé de recherche à l'ORSTOM, il met sur pied un système d'informations économiques et sociales à vocation agricole sur la base des 850 municipios de la région des cerrados au Brésil. Depuis 1988, tout en poursuivant ses recherches sur la différenciation spatiale dans les cerrados du Brésil, il participe aux travaux de la Maison de la Géographie de Montpellier en coordonnant diverses productions cartographiques sur les Départements et Territoires d'Outre-Mer. Il est membre du Groupe Dupont.

Les auteurs tiennent à adresser leurs plus vifs remerciements:

A l'Université de Lausanne, et en particulier à Rémi Jollivet, dont la conviction a permis de réaliser des ateliers d'application de l'informatique graphique et de l'analyse statistique avec SAS en sciences humaines. Que Pierre Kuffer et Anne Perroud, ingénieurs au Centre Informatique, trouvent ici l'expression de notre gratitude pour l'effort qu'ils ont consenti afin de favoriser la promotion de SAS en tant qu'outil universel d'analyse statistique.

Dans le travail quotidien, Jean-Bernard Racine, directeur de l'Institut de Géographie (IGUL) a soutenu et encouragé les efforts de rédaction de cet ouvrage.

Au Département Milieux et Activités Agricoles de l'ORSTOM, en la personne de son chef, Yves Gillon et de la responsable de l'unité de recherche «Analyse régionale et gestion des milieux» (UR 3J), Yveline Poncet, pour avoir favorisé les échanges de savoir faire entre les auteurs.

A la Maison de la Géographie de Montpellier et au GIP RECLUS, et plus précisément à Roger Brunet et à Régine Vanduick, qui ont bien voulu accepter de prendre en charge l'édition de cet ouvrage.

*** SAS est une marque déposée de *SAS Institute Inc.*, Cary, North Carolina, U.S.A.**

Préface

par Jean-Bernard Racine

«Pratique de l'Analyse Statistique. SAS sur PC/PS, minis et gros systèmes» ! Un titre apparamment bien jargonneux pour un ouvrage publié sous les auspices d'une «maison de la géographie», écrit par deux géographes, et d'abord à l'intention de géographes. Et pourtant, sa double référence à la statistique et à l'ordinateur, tels que l'une et l'autre se marient dans l'informatique, n'étonnera plus personne au sein de la discipline ancestrale, tandis que l'absence de toute référence explicite à celle-ci souligne à quel point l'outil étudié est susceptible de rendre service à l'ensemble de la communauté des chercheurs d'aujourd'hui, bien au-delà de toute frontière disciplinaire.

Mais les auteurs sont géographes. Et c'est à un géographe qu'ils ont demandé de témoigner, d'entrée de jeu, de l'importance qu'a pris, dans l'aventure qui est la sienne, de contribuer à la connaissance d'une Terre aujourd'hui comprise comme écriture à déchiffrer, à travers ce nouveau médiateur que lui proposent l'informatique et les techniques d'analyse et de traitement cartographique des données qu'elle autorise aujourd'hui.

Un médiateur qui a ses exigences. Qui sont telles que bien des géographes ont commencé par s'en émouvoir, et ce d'autant plus que les premiers résultats ont peut-être témoigné, au sein d'un mode de connaissance pourtant fort indiscipliné, des risques de réductions et d'illusion surtout («celle d'atteindre la certitude par l'exhaustif» écrit encore P. George en 1989), inhérents à une discipline nouvelle. Une discipline qui allait pourtant se révéler féconde quand, par nécessité de langage, elle obligea à dissocier l'étape du traitement de l'information des autres étapes de la recherche: un effort d'analyse et de définition des objets, des moyens et des buts, dont Jacques Bertin écrivait, dès 1969, qu'il était «probablement le plus grand apport humain de l'informatique».

Vingt ans ont passé. Depuis, le paysage de la géographie francophone a beaucoup changé, et particulièrement sur le plan de l'instrumentation scientifique. Ce qui pouvait apparaître comme de la «Nouvelle Géographie» dans les années 1970, comme par exemple les expériences présentées en 1973 dans l'*Analyse quantitative en géographie*, fait désormais partie des connaissances de base de la majorité des étudiants de second cycle universitaire. Or cela constitue sans doute une avancée dont on ne mesure pas encore complètement l'importance pour la discipline.

Aujourd'hui, la pratique de l'analyse quantitative dans l'ensemble des sciences so-

ciales, et plus particulièrement en géographie, peut se concevoir difficilement sans le recours à l'ordinateur. Et tout en étant consciente des limites de l'instrumentation et de la mesure dans l'élaboration des critères de vérité de la démarche scientifique, partout, et particulièrement à Lausanne où le véritable «stage de formation» auquel correspond l'ouvrage que nous proposent M. Cosinschi et P. Waniez a pu être testé auprès des jeunes chercheurs, l'Université consent actuellement un considérable effort d'équipement informatique pour son application aux disciplines traditionnellement littéraires.

Dans cet effort, l'aide de la Maison de la Géographie à Montpellier fut précieuse au double plan technique et humain. Car il est bien clair qu'en la matière, ce sont les ressources humaines qui sont les plus difficiles à rassembler. C'est bien dans cette perspective qu'il faut apprécier la collaboration des deux auteurs. Leur participation à l'effort collectif d'autoformation des chercheurs est intéressante, non seulement pour l'Université de Lausanne, mais aussi pour les nombreux lecteurs qui pourront franchir, avec eux, et sans encombre, les difficiles premières étapes de l'informatisation d'une partie, souvent en croissance rapide, de leurs pratiques de recherche.

Faut-il ajouter, avec René Berger, autre Lausannois, que «de même que la langue permet d'articuler au plan du dire les rapports qu'une société entretient avec son milieu et ses propres membres, de même l'outil permet d'articuler les figures interrelationnelles qu'une société établit au plan du faire avec son milieu et ses membres ?». Alors le témoignage n'appartient plus au préfacier. C'est l'ouvrage qui témoigne, comme appartenant déjà à la culture, «une culture qu'un humanisme étriqué a trop souvent réduite aux idées, à la littérature, aux beaux-arts, bref à quelques manifestations tenues pour privilégiées et que s'arrogeaient les privilégiés». Mais la culture de l'informatique a vite créé d'autres privilèges, ceux des gardiens du langage codé. Des privilèges que nos deux géographes, pourtant de formation littéraire à l'origine, mais qui ont pris soin de se garantir auprès de statisticiens et informaticiens chevronnés, se dont donné les moyens d'aider à faire tomber. Ce qui est de bonne guerre en cette année du bi-centenaire et qui contribuera justement à la nécessaire «désacralisation» de l'outil.

A cette fin, l'ouvrage présente les principales possibilités de traitement des données avec le système SAS. Le choix d'un tel produit n'est pas fortuit. En effet, les utilisateurs de l'informatique que nous sommes ont très tôt souhaité avoir à disposition un système informatique riche en méthodes de traitement, assez facile d'accès et pouvant traiter un grand volume d'unités statistiques. Les temps héroïques où soumettre un «job» à l'ordinateur constituait une course d'obstacles sont désormais révolus, du moins espérons-le.

Et ce ne sera pas l'un des moindres intérêts de ce volume que de découvrir les raisons de leur choix, en matière de logiciel, qu'ils exposent, de manière très didactique, tout au long de l'ouvrage. Ils auraient pu en choisir un autre, comme SPSS, bien connu des chercheurs d'Outre-Atlantique. L'important cependant réside moins, pour nos disciplines, dans telle ou telle performance exceptionnelle; plus déterminante est la maîtrise d'un instrument qui en garantit sa bonne utilisation. Une maîtrise dont les racines se situent bien en amont des ressources du logiciel, dont l'utilisation doit se soumettre à une philosophie, à un projet, qui littéralement, le «met en sens». Expression surtout d'un souhait, d'une exigence: que la réduction des difficultés techniques et fonctionnelles de l'analyse ne se traduise pas par une réduction du raisonnement, mais qu'au contraire elle le stimule de manière originale en le conduisant à la découverte de messages de moins en moins probables. Car tel est bien notre souhait, qu'exprime Blanché, cité par M. Borillo dans son ouvrage *Informatique pour les sciences de l'homme, limites de la formalisation du raisonnement*: «Un moule à raisonnements n'est pas un raisonnement, pas plus qu'un moule à gâteaux ne peut être mangé comme dessert».

I

Des matrices d'information aux bases de données géographiques

Collecte et organisation des données

Toute recherche expérimentale ou corrélative exige la mise en œuvre d'une instrumentation, c'est -à-dire la sélection d'*instruments de mesure* et de *méthodes* efficaces pour évaluer un problème. Dans ce contexte, un instrument de mesure peut tout autant être un questionnaire, fournissant des données spécifiques dont on maîtrise la pertinence, la validité et la fiabilité des mesures, qu'un système d'observation à vocation plus générale, tel un recensement ou un inventaire, où il sera peut-être plus difficile d'identifier les sources d'erreurs (erreurs de codage, omission de données, arrondis, agrégation, etc.). Ainsi, puisque la méthode statistique transforme des données brutes en information pertinente, la **collecte** et l'**organisation** de ces données de base sont des étapes essentielles dans la recherche et conditionnent en grande partie la qualité du travail d'expérimentation qui s'ensuivra. Il apparaît donc utile de préciser la forme et le contenu des tableaux de données les plus couramment utilisés.

I.1

La matrice d'information

Le conditionnement de l'information dans des matrices d'information est un préalable nécessaire au traitement informatique et statistique. Dans sa forme la plus courante et la plus conventionnelle, une **matrice d'information** est un tableau rectangulaire de données qui met en rapport deux ensembles d'observables: celui des objets de l'étude, les unités d'observation (ou plus simplement dit, les observations), et celui des descripteurs de ces objets, les variables.

Les N_i objets d'étude

Les N_i objets d'étude que compare le chercheur peuvent être des individus, des échantillons, des localités, des parcelles, des observations temporelles ou des prélèvements et forment l'ensemble N des unités d'observations. En géographie, il est courant d'avoir un ensemble d'objets qui sont des *unités spatiales*; c'est la raison pour laquelle on parle de **matrice d'information spatiale**. Ces observations peuvent être notamment des points ou des surfaces. Il s'agira d'un semis disjoint dans l'espace si les observations sont des points (un semis de villes, par exemple). Il s'agira d'un maillage le plus souvent continu, plus ou moins régulier, si les observations sont des surfaces (les maillages administratifs, par exemple). Ces ensembles surfaciques constituent une partition de l'espace, les limites des unités spatiales ne doivent pas se recouper.

Les M_j variables

Les M_j variables, terme équivalent à descripteurs, attributs, caractéristiques ou caractères, sont des qualités, propriétés ou paramètres mesurables, au moyen desquels on peut décrire et comparer les objets d'étude. Ce peut être un ensemble de descripteurs, ou la partition de cet ensemble en catégories pertinentes; les qualités observées, les *variables* de l'analyse, peuvent être de nature qualitative ou quantitative, si bien qu'il conviendra de distinguer différents types de variables. Nous y reviendrons à propos du processus de la mesure.

Toute matrice d'information spatiale G_{nm} est donc à la base un tableau composé de N_i lignes, qui sont les unités d'observation spatiale, et de M_j colonnes, qui sont les variables décrivant ces unités; il renferme ainsi un nombre n * m de cases. On note g_{ij} la case correspondant à la ième observation et à la jème variable (Fig. I.1).

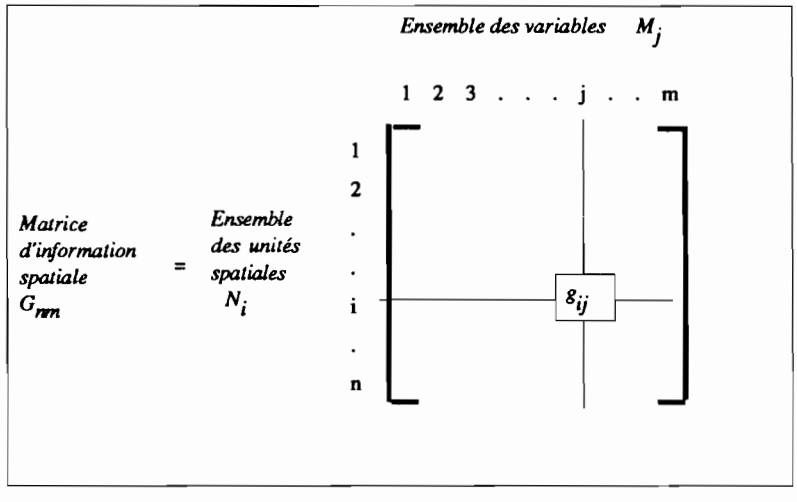


Figure I.1

Les éléments d'une matrice d'information spatiale

Le cube d'information

Un tableau rectangulaire peut se transformer en cube de données si l'on ajoute un troisième ensemble, l'ensemble **T** du temps. Les dimensions de ce cube sont les **n** unités spatiales sur les lignes, les **m** caractéristiques sur les colonnes et, au fur et à mesure que l'on se déplace en profondeur, chaque tranche est une période **k** de temps particulier parmi le nombre total **t** de périodes. Le cube matriciel G_{nmt} compterait un nombre $n*m*t$ de cases où g_{ijk} correspondrait à la ième observation, à la jème variable et à la kème période. En pratique, on enchaînerait les **k** tableaux correspondant aux matrices d'informations G_{nm} pour chaque période. On imagine facilement la complexité de telles matrices puisqu'un grand nombre de hiérarchies sont possibles sur chaque dimension (Fig. I.2).

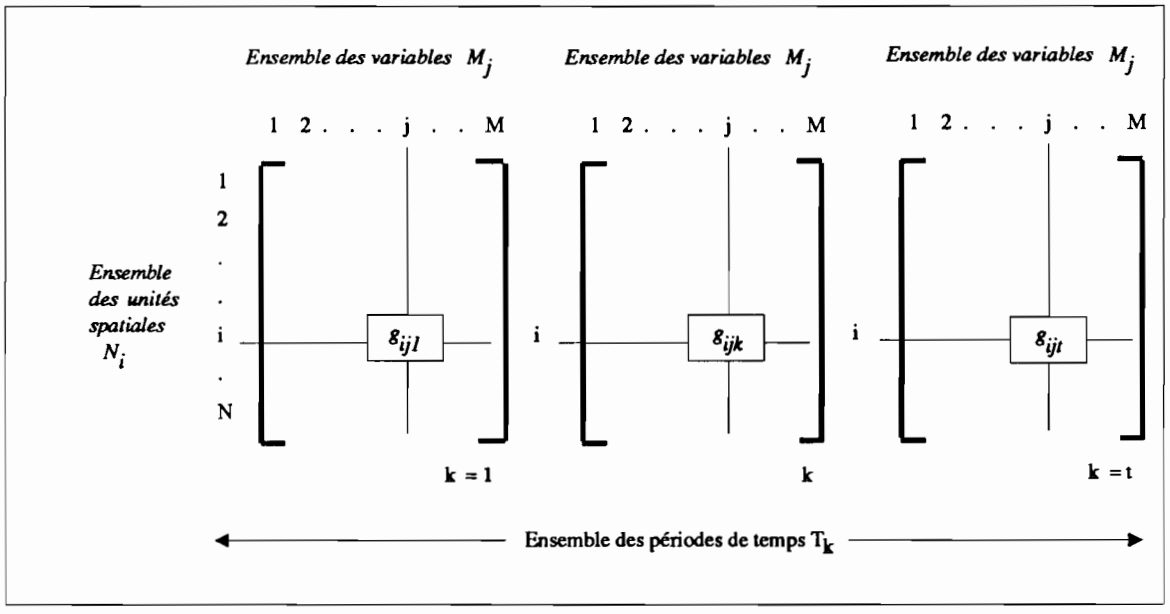


Figure I.2

Le cube matriciel découpé en matrices d'information spatiale

De nombreuses difficultés apparaissent dans la collecte et l'organisation de ces cubes d'information spatiale. Au niveau des N_i unités spatiales, d'abord, se pose le problème de la permanence temporelle des mailles découpant l'espace, par exemple, le découpage des îlots urbains se transformant d'un recensement à l'autre par adjonction de zones périphériques. Au niveau des M_j variables, ensuite, se pose le problème des modifications des nomenclatures qui empêchent toute comparaison directe d'un inventaire à l'autre (les recensements nous fournissent de multiples exemples d'incomparabilité des nomenclatures socio-économiques d'une période à l'autre). Finalement, au niveau des T_k périodes, se pose le problème de la disponibilité de l'information selon une périodicité constante

(annuelle, quinquennale, etc.) rendant difficile l'élaboration de modèles spatio-temporels.

Tableau de flux

Le cube d'information spatiale ne représente que des coupes dans une évolution historique, l'amalgame de caractéristiques particulières, en des lieux particuliers, à des moments particuliers. Mais nous savons qu'à n'importe quelle période k, il existe des échanges entre les unités spatiales: flux de biens, de services, d'information, d'énergie, de personnes, etc. Ainsi, pour un type d'échange donné, il est possible de construire une matrice **Fnn** de flux dans laquelle chaque cellule représente l'interaction entre un lieu d'origine et un lieu de destination, la diagonale du tableau indiquant la rétention des échanges à l'intérieur d'une unité spatiale (Fig. I.3). La troisième dimension peut ajouter l'ensemble des types d'échanges entre les lieux et une quatrième dimension est possible si l'on inclut les flux à des périodes différentes. On le voit bien, l'élaboration de tels cubes multiplie les difficultés rencontrées dans le cas, plus simple, des matrices d'information spatiale.

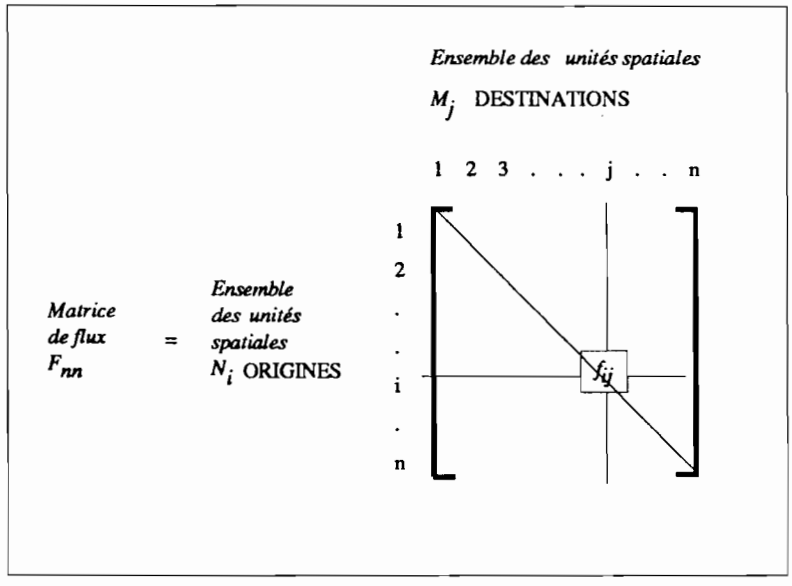


Figure I.3

Les éléments d'une matrice d'information spatiale de flux

Face à ces matrices ou cubes d'information spatiale, notre principal intérêt ne doit cependant pas se situer au niveau de leur structure, aspect qui concerne plutôt l'étape préparatoire d'une recherche. On doit s'interroger, à un autre niveau touchant à la procédure de recherche cette fois, au **contenu** des tableaux. La recherche utilisant la méthodologie de l'analyse statistique requiert ainsi une certaine mise en forme de l'information. C'est à travers le processus de la mesure qu'il sera possible

de construire des *variables* au moyen desquelles sont analysés les objets de l'étude.

I.2

Le processus de la mesure

Le rôle de la mesure est de rendre opératoire les concepts utiles au chercheur. Celui-ci tend à recueillir des données sur un grand nombre de variables ayant trait au problème étudié, à partir desquelles il doit dégager, en utilisant des méthodes d'analyses appropriées, une structure intelligible et signifiante. Le succès de l'analyse et de l'interprétation des résultats est lié d'une part, à l'adéquation entre les données observables et l'abstraction conceptuelle et, d'autre part, à l'adéquation entre les données recueillies et les méthodes possibles de traitement de l'information. Il faut donc être attentif pour l'un au processus de la mesure et, pour l'autre, aux propriétés des nombres recueillis dans la matrice d'information spatiale.

Le processus de mesure

La mesure comme «**processus**» concerne l'élaboration de l'objet par schématisation et pose le problème de la définition des variables dans la recherche. En effet un concept, considéré comme généralisation et abstraction de la réalité empirique, est une idée qui n'est pas directement observable ni mesurable. Il faut donc créer des moyens permettant d'adapter une telle représentation intellectuelle à l'investigation scientifique; ceci

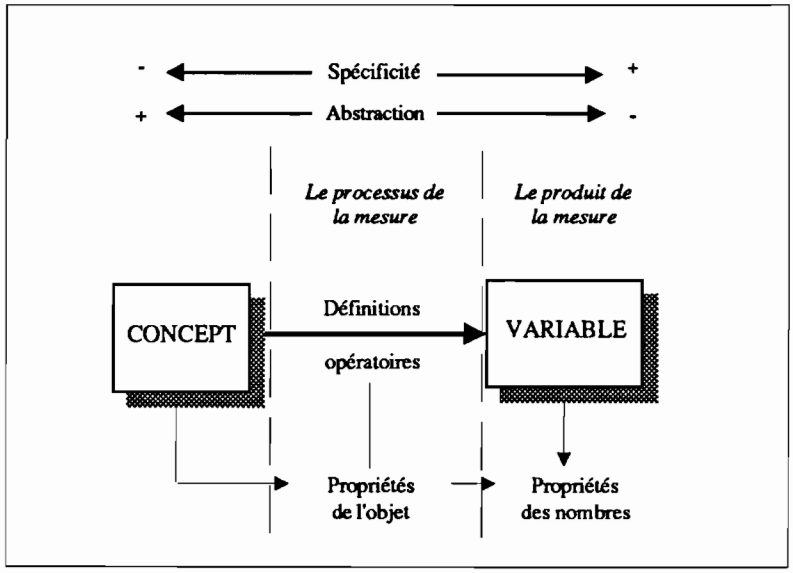


Figure I.4

Du concept aux données

passe par l'adjonction d'une nouvelle dimension, les définitions opératoires, qui doit offrir des possibilités d'observation et de mesure, anticipant la variable, qui permet d'établir les valeurs du concept rendu opératoire (Fig. I.4).

Le produit de la mesure

Au niveau de la variable, on passe à la mesure comme «**produit**». Il convient alors d'être attentifs à la relation entre niveau d'élaboration de l'objet et la théorie des nombres; elle fixera l'adéquation entre la mesure et les types de traitement possibles. Les valeurs prises par les variables peuvent être de deux natures différentes: les données qualitatives et les données quantitatives; plus précisément, elles s'appuient sur quatre niveaux d'échelle de mesure.

Variable qualitative

Une **variable qualitative** indexe les unités d'observation en fonction de modalités ou catégories. Types d'utilisation du sol, homme/femme, espèces végétales, classification régionale, en sont des exemples. Ce type de variable est défini par l'ensemble exhaustif des modalités prises par des unités d'observation. On peut assigner des valeurs numériques, le plus souvent arbitraires, à de telles variables, que ce soit en termes de présence (1)/absence (0) d'une propriété ou en termes de code numérique pour chaque modalité. Les opérations mathématiques seront limitées sur les variables de type qualitatif. Seule la statistique non paramétrique pourra être mise en œuvre sur des variables qualitatives.

Variable quantitative

Une **variable quantitative** permet d'indexer les unités d'observation par des nombres issus de comptage ou de mesure. Nombre de jeunes > 15 ans, nombre d'immeubles construits sont des exemples de données issues de comptages; les valeurs sont alors de nature *discrète*, c'est-à-dire ne prenant que des valeurs entières. Les précipitations sont un bon exemple de données issues de mesures, et peuvent prendre n'importe quelle valeur *continue* en fonction du degré de précision de l'instrument de relevé. Pratiquement toutes les opérations mathématiques seront possibles sur les variables quantitatives. La statistique non paramétrique et surtout la statistique paramétrique seront mises en œuvre sur des variables quantitatives.

Il apparaît parfois intéressant de convertir des données quantitatives en données qualitatives; dans ce cas, les valeurs numériques de la variable quantitative sont groupées dans des modalités non numériques du style «faible», «moyen», «fort», par exemple. Une telle simplification peut faciliter l'interprétation d'une variable à grands effectifs.

Quatre échelles de mesures

La distinction qualitatif/quantitatif peut s'élargir par la classification des variables en termes d'**échelles de mesure**. On reconnaît à la base quatre échelles de mesure, elles définissent l'information contenue dans la variable et les opérations qui peuvent être entreprises sur celle-ci (Fig. I.5).

Echelles de mesure	Propriétés	*Exemples*
QUALITATIVE		
— **Nominale**	1. Equivalence	*Type A,B,C*
Dichotomique		*Présence/absence*
— **Ordinale**	1. Equivalence	*Préférences*
	2. Plus grand que	*Hiérarchies*
QUANTITATIVE		
— **D'intervalle**	1. Equivalence	*Températures*
	2. Plus grand que	*Notes d'examens*
	3. Rapport connu entre intervalles	
— **De rapport**	1. Equivalence	*Distances*
	2. Plus grand que	*Densités*
	3. Rapport connu entre intervalles	*Ages*
		Nombre de ...
	4. Rapport connu entre valeurs	

Figure I.5

Les échelles de mesure

Echelle nominale

La plus rudimentaire est l'**échelle nominale**. Les variables de l'échelle nominale sont qualitatives, les modalités n'ont aucun ordonnancement implicite et, si nous donnons des valeurs numériques à des variables nominales, celles-ci n'ont aucun sens, si ce n'est celui d'identification sur une modalité. Il est nécessaire de construire une échelle exhaustive qui couvre l'ensemble des situations et il faut veiller à ce que les modalités s'excluent mutuellement, c'est-à-dire que chaque unité d'observation ne possède qu'une et une seule modalité. Une échelle binaire est un cas particulier à deux modalités seulement. Sur le plan informatique, il est souvent préférable d'utiliser des chaînes de caractères pour identifier les modalités afin de mieux distinguer les valeurs numériques des autres échelles de mesure.

Echelle ordinale

Si les catégories d'une variable qualitative peuvent se mettre en rang croissant ou décroissant, on parlera d'une **échelle ordinale**. La force d'une opinion exprimée sur une question, la hiérarchie des centres urbains en sont des exemples parmi d'autres. On parlera 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 unité d'observa-

tion et une seule. Même dans ce dernier cas, la différence quantitative entre observations successives est inconnue. La représentation numérique des rangs fait appel à l'ensemble des nombres entiers. La seule contrainte à respecter dans la numérisation est de maintenir l'ordonnancement implicite de la terminologie associée aux catégories ordinales.

Les variables quantitatives, qu'elles soient discrètes ou continues, peuvent de leur côté être scindées en deux échelles de mesure.

Echelle d'intervalle

Sur une **échelle d'intervalle**, chaque unité d'observation a une valeur propre au phénomène mesuré et les différences entre des observations adjacentes sont égales sur l'unité de mesure. Pratiquement toutes les opérations arithmétiques peuvent s'effectuer, à l'exception du calcul des ratios; l'intervalle entre deux valeurs est connu, puisque l'unité de distance entre les nombres est constante à partir du zéro fixé de manière arbitraire. Les repérages de température en Celsius ou Fahrenheit en sont les exemples les plus évidents, chaque échelle définissant un point d'origine à partir duquel les données continues ont des intervalles significatifs.

Echelle de rapport

Au plus haut niveau de possibilités arithmétiques, l'**échelle de rapport** exprime toutes les propriétés de l'échelle d'intervalle avec, en plus, une origine naturelle de zéro et un rapport des différences à une même unité de mesure. Distances, densités, surfaces, exemples parmi d'autres, peuvent subir toutes les opérations arithmétiques ou mathématiques.

Finalement, il est bon de toujours garder en mémoire, lorsqu'on parle de mesure et de variables, les questions suivantes: *avons-nous bien mesuré ce que nous voulions mesurer ?* ou bien: *avec quelle pertinence mon instrument mesure-t-il la réalité à l'étude ?* Une grande partie de la statistique n'étant en fait que l'élaboration, par la présentation schématique des tableaux, des données recueillies, il est indispensable, en amont, de se poser le problème de la mesure des objets que l'on juge pertinents pour décrire un phénomène en son univers. Il s'agit de se poser quatre questions correspondant à quatre qualités de la mesure:

— La **validité**: mon instrument mesure-t-il le phénomène ou la caractéristique que je veux étudier?

— La **fidélité**: mon instrument de mesure étant stable, donne-t-il le même résultat si j'opère deux fois de suite l'opération de mesure?

— La **sensibilité**: quelle est la sensibilité de l'instrument de mesure, quelle différence permet-il de mettre en évidence entre deux objets proches?

— La **généralisabilité**: c'est la qualité d'un instrument de mesure de pouvoir mesurer un caractère particulier ou une classe de caractères plus généraux.

Tout le processus de mesure n'aura de sens que par rapport à l'observable, il implique nécessairement une analyse de l'objet mesuré. L'outil qu'est l'ordinateur ne peut prendre en compte que ce qui est codé sous la forme de mesures, il va traiter celles-ci comme des nombres. Rien n'est dit sur le rapport entre objet et mesure puisqu'on opère sur des chiffres sans se soucier du niveau et de la qualité de ce rapport. En ce sens, l'ordinateur peut être la meilleure comme la pire des choses. Cela doit rendre le chercheur encore plus attentif.

I.3

Le Canton de Vaud en nombres

A titre d'exemple et de support pratique, trois matrices d'information spatiale ont été construites sur un sous-ensemble suisse. Le découpage spatial de la Confédération helvétique met en jeu deux niveaux institutionnels de base: *26 cantons* et *3029 communes*. Nous avons choisi le Canton de Vaud comme territoire de travail, il compte 385 communes (Fig. I.6).

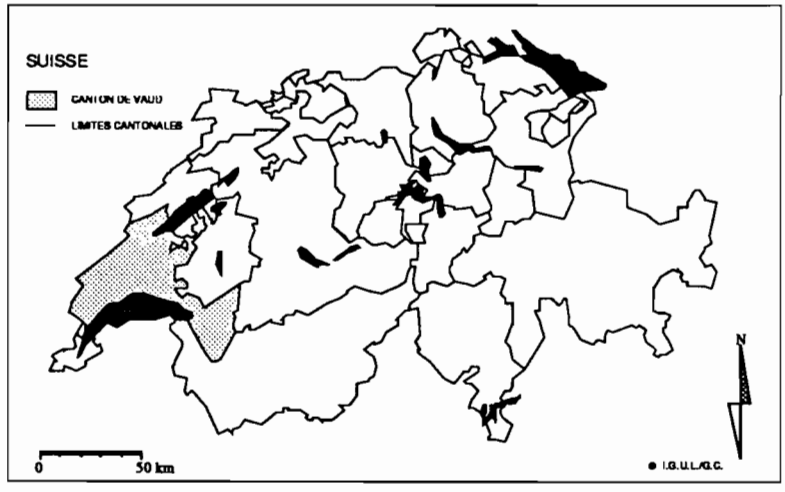


Figure I.6

Le Canton de Vaud en Suisse

Les informations disponibles au niveau spatial relèvent pour la plupart d'organismes publics, de niveau fédéral ou cantonal; ils fournissent régulièrement et systématiquement des données communales à travers les recensements de population (recensements tous les 10 ans, le plus récent datant de

1980) ou les recensements plus spécialisés (recensement des entreprises par exemple, d'une périodicité de 10 ans, le plus récent étant de 1985). Nous avons retenu des informations concernant la *population* et la *construction* tirées du Recensement fédéral de la population et fournies par l'Office fédéral de statistique à Berne en 1960, 1970 et 1980, auxquelles nous ajoutons diverses possibilités d'agrégation des communes, tant au plan administratif que régional, à travers des variables qualitatives identifiant des types d'agrégation. Au total donc, trois matrices d'information spatiale:

Trois matrices à titre d'exemple

1º VDGEO——> *Matrice géographique* des identificateurs spatiaux et types des communes.

2º VDPOP——> *Matrice population* sur des variables démographiques.

3º VDCONST—> *Matrice construction* sur des variables touchant à la construction d'immeubles.

I.3.1

La matrice géographique VDGEO

La matrice «géographique» appelée VDGEO est composée de 385 observations, les communes du Canton de Vaud, et de 9 variables (Fig. I.7).

Les variables NUMERO, DISTRICT, REGION, RUMLEY et AGG80 sont discrètes, relevées sur des échelles nominales; la variable NOM est alphanumérique; les variables X et Y sont continues et mesurées sur une échelle d'intervalle; la variable SURFACE est également continue, numérique relevée sur une échelle de rapport.

L'ensemble des informations de la matrice VDGEO fournissent des renseignements sur l'identification des unités d'observation (NUMERO, NOM), leur localisation géographique (X, Y), leur niveau d'agrégation institutionnel (DISTRICT) ou régional (REGION, RUMLEY, AGG80) et leur surface utile (SURFACE).

A. Identification des communes:

L'Office fédéral de statistique (OFS) gère une liste officielle des communes de la Suisse par nom et numéro de code. Toutes les communes ont un nom et un numéro unique. Les numéros de communes sont enregistrés sur 4 chiffres non consécutifs tenant compte des modifications dans le maillage (communes détachées, disparition ou création de nouvelles communes). La variable NUMERO sert de lien entre les diverses matrices d'information pour établir les correspondances entre les unités d'observation, tant au niveau des ta-

bleaux de données qu'au niveau du maillage cartographique. Les numéros de communes du Canton de Vaud vont de 5401 à 5939. La Figure I.17, à la fin de ce chapitre, représente la carte du maillage communal du Canton de Vaud. On notera la grande hétérogénéité des superficies communales.

1º VDGEO
Matrice «géographique»

Matrice VDGEO
{385.9}

Position	Variable	Description
1	**NUMERO**	Numéro de codage OFS de chaque commune (4 chiffres): 5401 à 5939.
2	**NOM**	Nom alphanumérique de la commune.
3	**X**	Coordonnée en abscisse du centre de la commune (Y de l'Office topographique). Xmin=499 Xmax=583
4	**Y**	Coordonnée en ordonnée du centre de la commune (X de l'Office topographique). Ymin=119 Ymax=200
5	**DISTRICT**	Code hiérarchique des 19 districts (4 chiffres) District=2201 à 2219 (Vaud=22).
6	**REGION**	Code de répartition des communes dans 4 régions physiographiques: 1 = Moyen pays 2 = Jura 3 = Préalpes 4 = Alpes
7	**RUMLEY**	Code de répartition des communes dans 5 régions «écologiques» (P.A. Rumley, 1984): 1 = Grande ville (Lausanne) 2 = Banlieues potentielles des grandes villes 3 = Villes et centres urbains 4 = Zone intermédiaire (rural Moyen pays) 5 = Zone de montagne
8	**AGG80**	Code d'appartenance des communes à une zone urbaine (48 zones urbaines en Suisse au total): 0 = Hors zone urbaine 12 = Agglomération de Genève 15 = Agglomération de Lausanne 27 = Agglomération de Vevey-Montreux 30 = Agglomération d'Yverdon-les-Bains
9	**SURFACE**	Superficie nette habitable en hectares.

Figure I.7

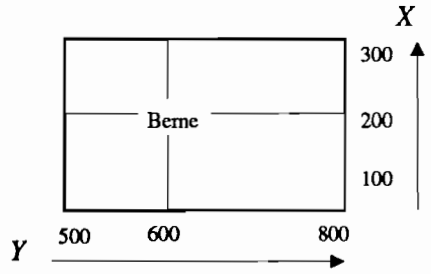
La matrice VDGEO

B. Localisation géographique:

Les variables X et Y

Les variables X et Y nous renseignent sur la position du centre de chaque commune. L'Office fédéral de topographie utilise une grille de référence kilométrique dont l'origine 0,0 se situe près de Bordeaux; à titre de référence, la capitale fédérale Berne se localise aux coordonnées 600,200:

Grille de référence kilométrique de l'Office fédéral de topographie



Pour des raisons techniques, la matrice VDGEO intervertit la grille de référence fédérale, la variable X exprime les coordonnées en abscisse, tandis que la variable Y exprime les coordonnées en ordonnée. Ces variables permettront de mesurer les distances entre les communes.

C. Agrégation et régionalisation:

La variable DISTRICT

Il est souvent intéressant d'agréger les unités d'observation en fonction de critères administratifs ou scientifiques. Sur le plan administratif, la variable DISTRICT permet l'agrégation des 385 communes en 19 districts, découpage politico-administratif de second niveau en Suisse (Fig. I.8).

Sur le plan scientifique, l'agrégation touche au problème de la régionalisation des unités spatiales. Une recherche peut avoir pour objet, en amont, la création de régions ou, en aval, le test de la pertinence de découpages régionaux lorsqu'ils sont confrontés à d'autres phénomènes que ceux qui les ont construits au départ. La matrice VDGEO propose à ce titre divers découpages de l'espace vaudois à travers des typologies régionales. Ils pourront servir de critère d'agrégation tout autant que de variable exogène dans certains modèles statistiques.

La variable REGION

La variable REGION est un simple découpage du canton en 4 régions physiographiques très globales correspondant à des milieux naturels prégnants en pays de Vaud (Fig. I.9).

La variable RUMLEY

La variable RUMLEY est plus complexe et plus discutable. Elle propose une régionalisation empirique fondée sur les milieux «écologiques» (Fig. I.10). Les travaux de P.A. Rumley (1984) ont servi de point de départ aux types définis. L'objectif était de déterminer des groupements de communes pouvant présenter des caractéristiques comparables du point de vue de l'utilisation du sol et des problèmes d'aménagement en mettant l'accent sur l'urbanisation et l'agriculture.

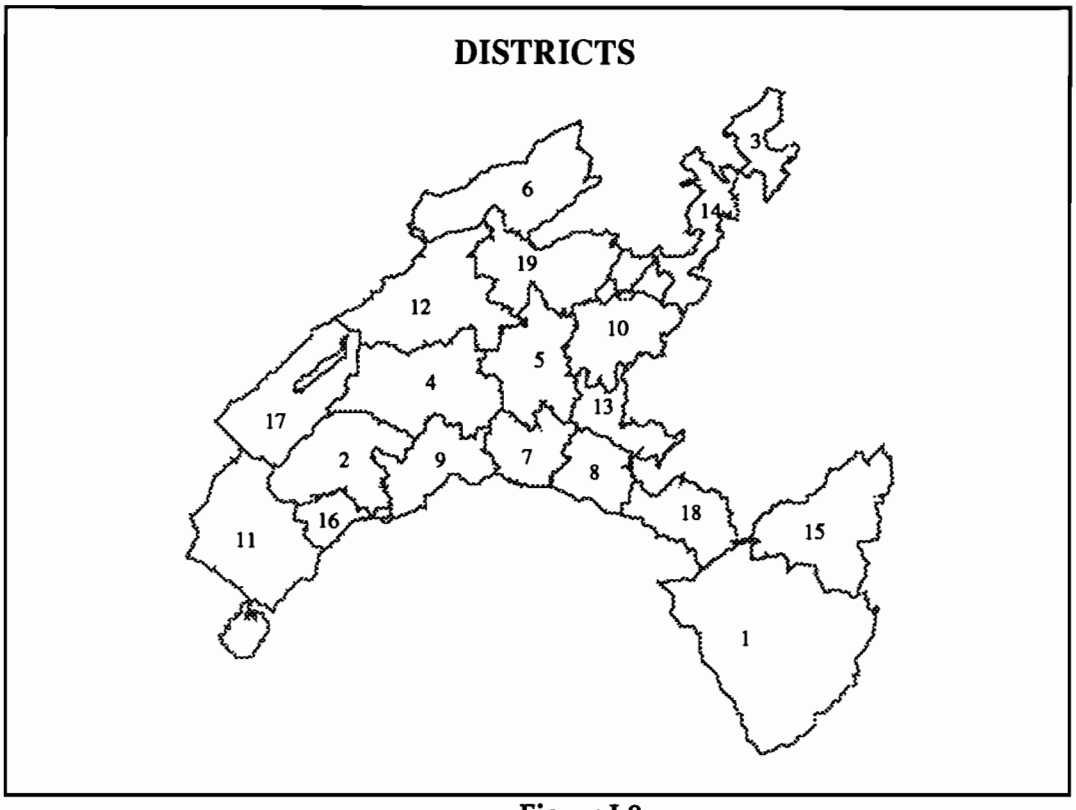


Figure I.8
Le découpage communal en 19 districts

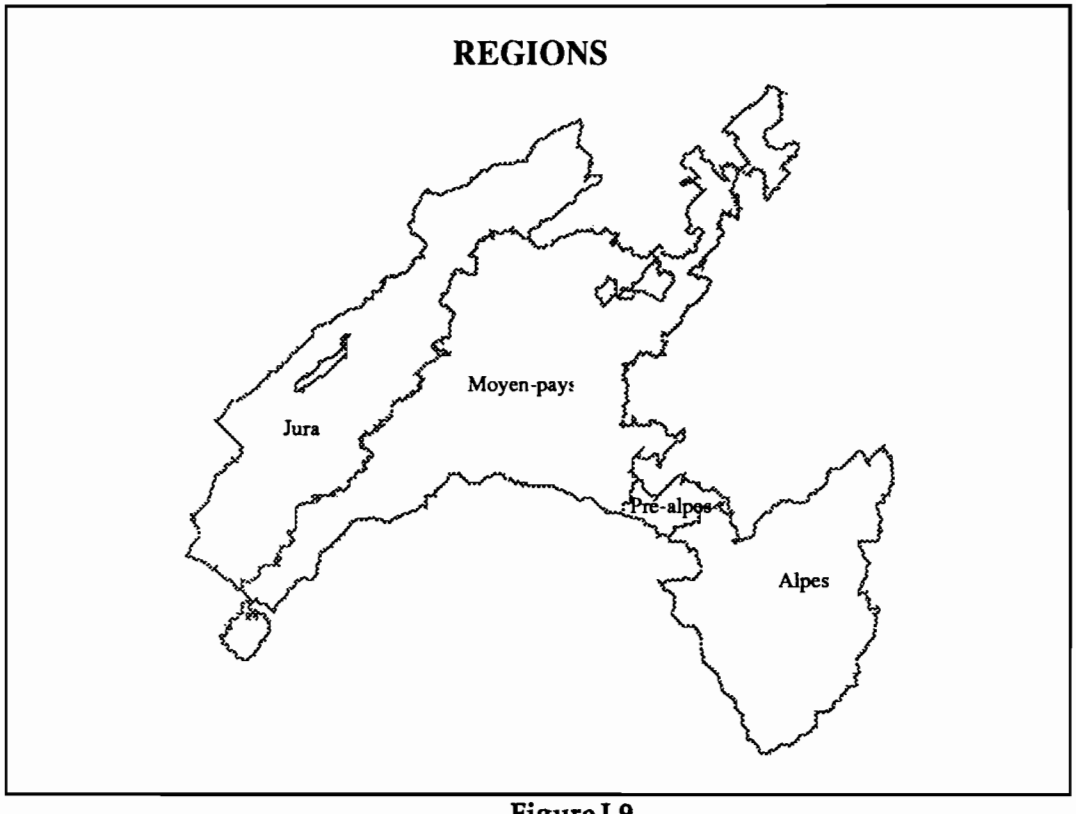


Figure I.9
Le découpage communal en 4 régions physiographiques

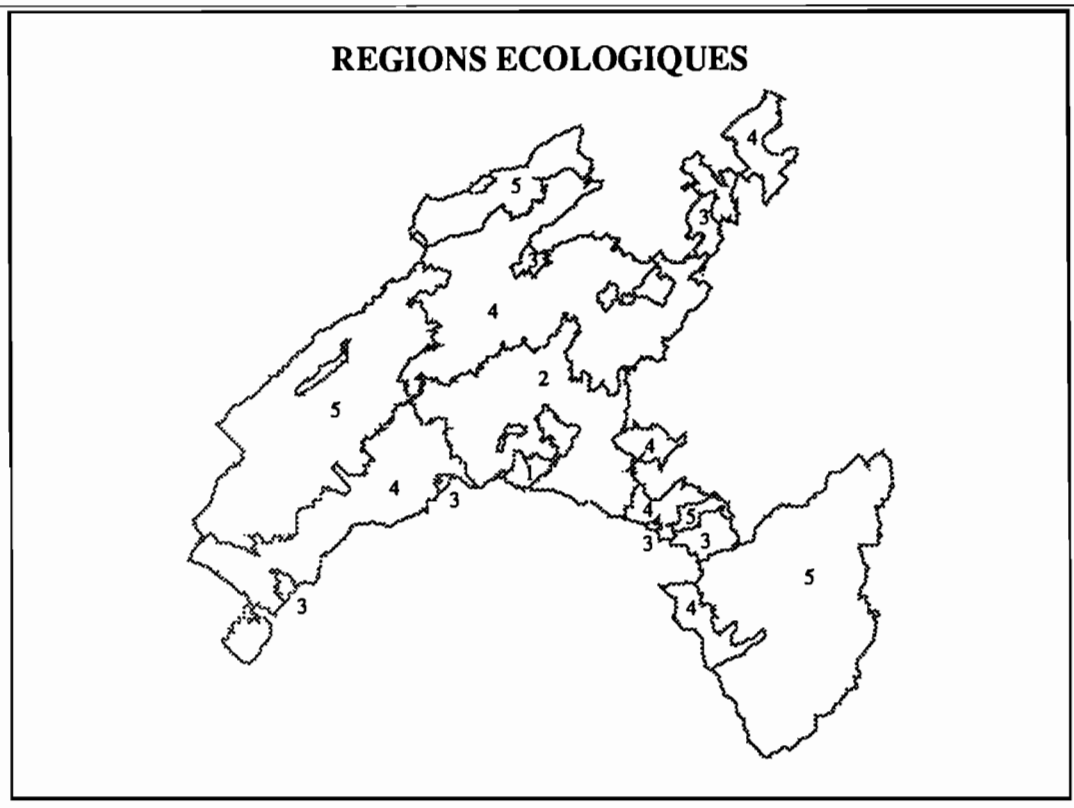


Figure I.10
Le découpage communal en 5 régions «écologiques»

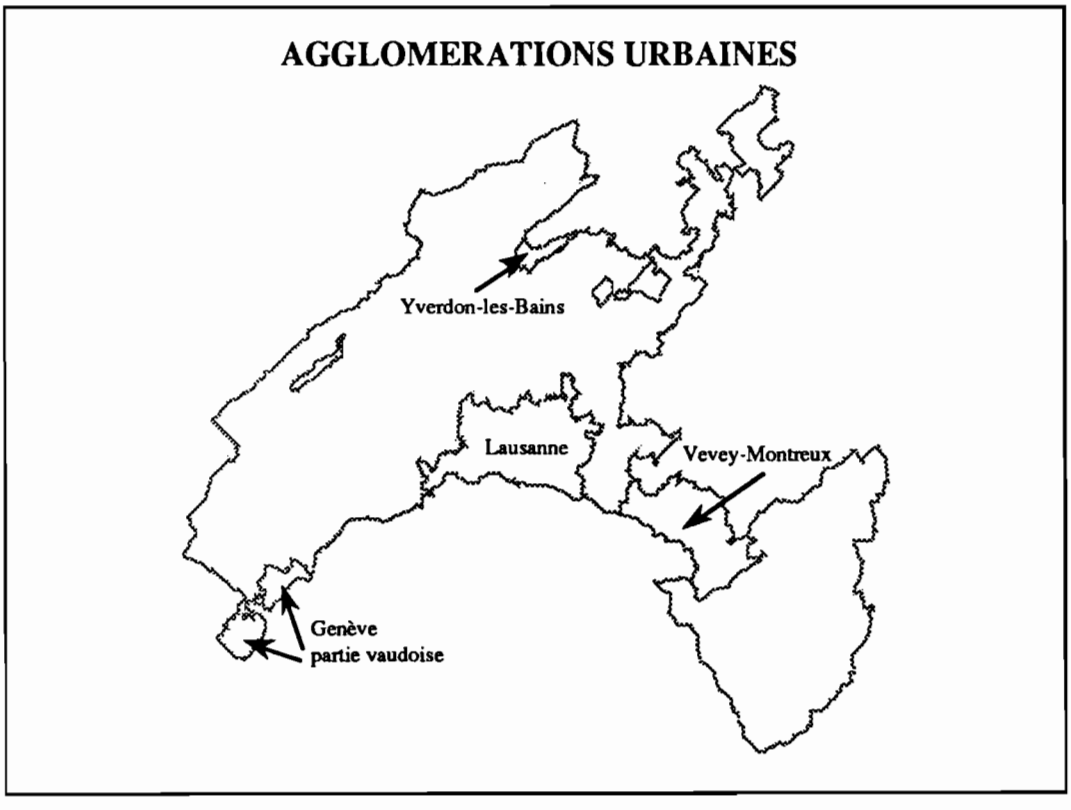


Figure I.11
Le découpage communal en 4 zones urbaines

Les 5 types distingués sont:

1º *Grandes villes*: Seule Lausanne entre dans cette catégorie.

2º *Banlieues potentielles des grandes villes*: Ce sont des communes à proximité d'une grande ville et formant avec cette dernière une agglomération potentielle. Leur définition a été faite sur la base des migrations pendulaires en 1970, de temps de transport ainsi que sur la base d'avis d'experts en aménagement. Il ne s'agit pas des agglomérations officielles de la zone urbaine qui sont sensiblement plus restreintes. Ce deuxième type est critiquable en particulier parce qu'il ne tient pas compte de l'influence de Genève qui s'étend sur la côte lémanique en territoire vaudois.

3º *Villes et centres*: Il s'agit ici de toutes les villes et centres urbains exerçant des fonctions centrales; certaines communes de population supérieure à 10 000 habitants n'en font pas partie.

4º *Zone intermédiaire*: L'intention était ici de prendre en compte les potentialités agricoles (aptitudes des sols, aptitudes climatiques) mais devant les difficultés, ce quatrième type a pris pour base de définition le cadastre de la production animale. La zone intermédiaire identifie les régions agricoles de plaine et de collines.

5º *Zone de montagne*: Les critères utilisés pour la zone intermédiaire ont permi de distinguer ce dernier type de faible urbanisation et d'agriculture de montagne.

La variable AGG80

La variable AGG80 permet l'identification des communes appartenant à une agglomération urbaine. La zone urbaine suisse comprend au total 48 unités subdivisées en 33 agglomérations et 15 villes isolées; elle a été rigoureusement définie (M. Schuler, 1984) sur la base du recensement de 1980 à partir de critères liés à l'importance de la population et de l'emploi, au rapport emplois/résidences, à l'occupation de la population, à la continuité de la zone bâtie, à la densité de la charge humaine (habitants/emplois), au dynamisme démographique et aux mouvements pendulaires. Le Canton de Vaud possède des communes appartenant à 4 agglomérations de la zone urbaine suisse, celle de Genève (communes vaudoises dans la zone d'influence genevoise), de Lausanne, de Vevey-Montreux (agglomération bi-polaire) et d'Yverdon-les-Bains dans le nord vaudois. Toutes les autres communes non affectées à une agglomération ont le code 0 (Fig. I.11).

La variable SURFACE

Dernière variable du fichier géographique, la SURFACE mesure la superficie nette en hectares de chaque commune. Il s'agit de la superficie totale de laquelle on a soustrait les superficies des lacs et cours d'eau, celles des forêts et des terres incultes (zones de montagnes non habitables). Les indicateurs de

densités pourront ainsi être calculés de manière plus réaliste bien que la statistique date de 1972.

I.3.2
La matrice population VDPOP

La matrice VDPOP contient 6 variables démographiques et la variable NUMERO (Fig. I.12).

2e VDPOP
Matrice «population»

Matrice VDPOP
{385.7}

Position	Variable	Description
1	**NUMERO**	Numéro de codage OFS de chaque commune (4 chiffres): 5401 à 5939.
2	**POP60**	Population résidente totale en 1960. (OFS: EINW60)
3	**POP70**	Population résidente totale en 1970. (OFS: EINW70)
4	**POP80**	Population résidente totale en 1980. (OFS: EINW80)
5	**POP85**	Population résidente moyenne en 1985. (ESPOP/OFS: G5GPRT, estimation)
6	**ETR85**	Population résidente moyenne d'origine étrangère en 1985 (permis B et C) (ESPOP/OFS: G5APZT, estimation)
7	**JEUNES**	Population résidente totale âgée de 0 à 14 ans en 1980. (OFS: AL804+AL859+AL81014)

Figure I.12

La matrice VDPOP

Les variables POP, JEUNES et ETR

Les variables de population sont des dénombrements, il s'agit de données continues relevées sur une échelle de rapport.

Les variables POP60, POP70 et POP80, de même que la variable JEUNES, viennent des recensements fédéraux de la

population tandis que les variables POP85 et ETR85 sont des estimations inter-censitaires fournies également par l'Office fédéral de statistique (ESPOP). La population étrangère (ETR85) évalue ici les allochtones possédant un permis de résidence en Suisse, soit temporaire (permis B), soit permanent (permis C) mais ne compte pas les saisonniers (permis A).

Cette matrice d'information permet de s'intéresser à la croissance de la population des communes vaudoises et de lier, le cas échéant, cette dynamique à celle de la construction d'immeubles d'habitation de la troisième matrice et aux typologies régionales des communes de la matrice précédente. Les variables de population 1985 et de population étrangère 1985 permettent quant à elles d'étudier la répartition des allochtones en pays de Vaud. La variable des moins de 15 ans, cette «population en voie de disparition» peut jouer un rôle endogène tant sur le plan de la dynamique démographique que de la demande de logements.

I.3.3

La matrice construction VDCONST

La matrice «construction» VDCONST est composée de 3 variables thématiques et de la variable NUMERO (Fig.I.13)

3e VDCONST
Matrice «construction»

Matrice VDCONST
{385.5}

Position	Variable	Description
1	**NUMERO**	Numéro de codage OFS de chaque commune (4 chiffres): 5401 à 5939.
2	**CONS60**	Immeubles construits avant 1960 (état 1980) (OFS: WGB847+WGB860)
3	**CONS70**	Immeubles construits de 1960 à 1970 (état 1980) (OFS: WGB870)
4	**CONS80**	Immeubles construits de 1970 à 1980 (état 1980) (OFS: WGB875+WGB880)

Figure I.13

La matrice VDCONST

Les variables CONS

Les variables CONS60, CONS70 et CONS80 sont des dénombrements continus sur une échelle de rapport. Elles évaluent en 1980 le nombre d'immeubles construits avant 1960, de 1960 à 1970 et de 1970 à 1980. En d'autres termes, nous mesurons combien d'immeubles en 1980 datent des périodes mentionnées.

La statistique des bâtiments et logements a été recueillie parallèlement au recensement de la population en 1980 et ne s'étend pas à tous les bâtiments. Le recensement ne retient que les immeubles destinés à l'habitation permanente des ménages, la définition exclut les édifices administratifs, sociaux, sportifs, commerciaux ou industriels, hormis ceux qui comptaient au moins un logement habité ou vacant, de même que les habitations improvisées ou mobiles. Les immeubles comprenant des logements constituent la quasi-totalité des bâtiments recensés tandis que les immeubles sans logements, représentant quant à eux des bâtiments abritant des ménages collectifs (hôtels, maisons de retraites, etc.), sont minoritaires; c'est pourquoi nous n'avons pas fait ici de distinction, les variables de la matrice VDCONST exprimant globalement la construction totale des immeubles d'habitation. L'époque de la construction indique, pour les bâtiments agrandis ou transformés, l'époque de construction primitive.

La matrice «construction» permet de suivre globalement le déploiement de la croissance urbaine de manière corrélative à la croissance de la population (matrice VDPOP). Par l'éclatement des villes lié à la pénurie de terrains à bâtir, au redéploiement des activités économiques, aux nouvelles technologies de transport, à la dégradation de la qualité de la vie ou encore aux nouvelles exigences des habitants liées à l'élévation du niveau de vie, la ville centrale éclate d'abord dans ses premières couronnes contiguës (on parlera alors de *suburbanisation*) puis vers des zones beaucoup plus éloignées en pays rural (on parlera de *périurbanisation*). Les mécanismes sont complexes et notre approche ici ne peut qu'être schématique. Les variables sur la construction des immeubles d'habitation ne sont pas idéales pour aborder ces problèmes d'urbanisation, des variables sur l'évolution des *logements* auraient été plus adéquates en ce qu'elles permettent de mieux saisir l'offre résidentielle ventilée par des modalités plus pertinentes, telles la construction de maisons individuelles, d'habitats groupés ou d'immeubles à logements multiples.

I.3.4
L'utilisation de l'information sur le Canton de Vaud

Les trois matrices d'information spatiale construites à titre d'exemple contiennent, on vient de le voir, des informations brutes sur la régionalisation des communes, leur population et leur caractéristiques de construction. Ces variables peuvent subir diverses transformations afin de calculer des indicateurs structurels plus parlants de la dynamique de l'urbanisation, des densités, de la répartition de la population étrangère ou des jeunes, que ce soit à l'échelle du maillage communal ou à un niveau d'agrégation régional particulier. Nous proposons ici, à titre d'exemple, quelques indicateurs généraux qui seront utilisés ultérieurement dans les procédures de traitement SAS.

Calcul d'indicateurs

A. Répartition structurelle de la population: densités

Densité60=POP60/SURFACE
Densité70=POP70/SURFACE
Densité80=POP80/SURFACE
Densité85=POP85/SURFACE

B. Répartition structurelle des allogènes et des jeunes:

% Jeunes80=(JEUNES/POP80)*100
% Etrangers85=(ETR85/POP85)*100

C. Croissance de la population:

% Croissance '60 à '70=((POP70-POP60)/POP60)*100
% Croissance '70 à '80=((POP80-POP70)/POP70)*100

D. Structure de la construction d'immeubles:

Total des immeubles construits jusqu'en '80:CONSTOT
CONSTOT=(CONS60+CONS70+CONS80)

% Construction avant '60=(CONS60/CONSTOT)*100
% Construction de' 60 à '70=(CONS70/CONSTOT)*100
% Construction de '70 à '80=(CONS80/CONSTOT)*100

E. Distance kilométrique du centre d'une commune au centre de Lausanne:

Centre de lausanne: X=538 et Y=152

$$DISTLAUS=\sqrt{(((X-538)*(X-538))+((Y-152)*(Y-152)))}$$

I.4
Le processus d'informatisation des matrices d'information spatiale

Le traitement des matrices d'information spatiale nécessite le recours aux méthodes statistiques et à la cartographie. La mise en œuvre de ces techniques nécessite l'ordinateur. Il faut donc présenter son information sur un support de mémoire en la structurant de manière reconnaissable par l'ordinateur.

Fichiers et supports magnétiques d'enregistrement

On organise généralement ses données en «fichiers» groupant des informations de même nature ou concernant un même sujet. C'est ainsi que nous avons créé trois fichiers sur le Canton de Vaud, correspondant aux trois thèmes définis ci-dessus. Ces fichiers sont enregistrés sur un support magnétique qui, selon le cas, sera une disquette, un disque dur ou une bande magnétique; ces types de support ont en commun la fonction de stockage des données. Les disques ou disquettes permettent l'accès à n'importe quelle partie du fichier par une rotation, puis un déplacement des têtes de lectures perpendiculaires à l'axe central. 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 magnétiques doivent être réservées au stockage et la communication de l'information 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 ou disquettes.

Format d'enregistrement

Une fois constituées, les matrices d'information spatiale doivent être transférées sur le support magnétique. On peut d'une part créer soi-même ses fichiers directement sur disque ou disquette ou, d'autre part, faire appel à une société de service fournissant les informations. Les données seront saisies en «images de cartes» (par référence aux anciennes cartes perforées en papier bristol) selon un «format» précis. Ce format spécifie la répartition des informations: les «champs», qui sont les zones contenant une variable spécifique, définissent les variables de la matrice d'information et sont séparés les uns des autres par un espace blanc en général, il y aura donc autant de champs qu'il y a de variables et chacun d'eux peut définir une variable quantitative (un nombre) ou une variable qualitative (alphabétique ou alphanumérique); le format de chaque champ, soit le nombre de positions occupées par chaque variable dans un champ.

Les fichiers des matrices d'information spatiale doivent être organisés selon le schéma de la Figure I.1: les lignes sont les unités spatiales et les colonnes sont les variables. On parlera d'«enregistrement» pour décrire les lignes, un enregistrement dans notre exemple vaudois sera constitué par l'information con-

cernant une commune; tous les enregistrements d'un même fichier ont donc une structure analogue qui peut être cependant de longueur variable tout en ayant un nombre fini de positions, au maximum 80 (par référence aux images de cartes perforées). En cas de besoin, une commune peut avoir une seconde, une troisième ligne, etc. 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.

Enregistrement «logique» et Enregistrement «physique»

Ainsi, pour bien s'entendre avec les interlocuteurs fournissant ou traitant l'information, il est nécessaire de fournir un dessin d'enregistrement qui permettra, à l'issue de la saisie, de retrouver son information. On doit préciser la position des différentes variables pour chaque observation, c'est-à-dire la délimitation, dans chaque enregistrement physique, des champs contenant les valeurs par le rang de leur caractère de début et celui de leur caractère de fin. La longueur de chaque champ correspondant à chaque variable doit être supérieure ou égale au nombre de caractères de la valeur la plus longue.

Les Figures I.7, I.12 et I.13 décrivent partiellement la structure d'enregistrement de nos matrices d'information spatiale. On y reconnaît la position séquentielle de chaque variable et on sait que chaque fichier contient 385 enregistrements logiques. Il faut, pour compléter le dessin d'enregistrement, décrire les formats d'organisation de chaque champ ou de chaque variable.

A. Le fichier VDGEO:

Cette matrice «géographique» contient 2 enregistrements physiques pour chaque commune (Fig. I.14), le fichier contient donc 2(385) lignes. Le premier enregistrement présente le NUMERO dans les colonnes 1 à 4 et le NOM des communes à partir de la colonne 6 dans un format alphanumérique de longueur variable, aucun espace blanc ne tronquant ce champ. Le deuxième présente les 7 champs des variables qualitatives d'agrégation régionale codées en valeurs numériques et la variable quantitative SURFACE. Ces champs sont de longueur variable, séparés par un ou plusieurs espaces blancs.

```
5401  AIGLE
      564  130  2201   4   4   0    1621
5402  BEX
      567  122  2201   4   5   0    9644
5403  CHESSEL
      558  133  2201   4   4   0     346
5404  CORBEYRIER
      563  133  2201   4   5   0    2195
5405  GRYON
      571  125  2201   4   5   0    1519
5406  LAVEY-MORCLES
      568  119  2201   4   5   0    1389
5407  LEYSIN
      567  133  2201   4   5   0    1850
5408  NOVILLE
      559  137  2201   4   4   0    1000
5409  OLLON
      568  127  2201   4   5   0    5929
5410  ORMONT-DESSOUS
      570  135  2201   4   5   0    6281
5411  ORMONT-DESSUS

...     ...          ...  ...  ...  ....
```

Figure I.14

Dessin d'enregistrement de VDGEO.DAT

B. Le fichier VDPOP:

La matrice «population» a un dessin d'enregistrement plus simple (Fig. I.15). Elle contient 385 enregistrements physiques, chacun présentant le NUMERO en colonne 1 à 4 puis six champs de longueur fixe à 7 caractères, sans partie décimale, débutant à la colonne 5.

```
5401   4381   6532   6233   6407   1408   1184
5402   4667   5069   4843   4863    756    891
5403    209    191    191    210     20     37
5404    253    271    270    291     13     56
5405    706    752    827    856    140    167
5406    855    734    750    796    105    148
5407   2241   2752   2057   1941    611    299
5408    459    460    427    463     39     78
5409   4126   4470   4429   4648   1078    949
5410    996    884    895    903     35    144
5411    921    997   1000    969    110    165
5412    186    232    222    259     40     39
....    ...    ...    ...    ...    ...    ...
```

Figure I.15

Dessin d'enregistrement de VDPOP.DAT

Les six variables démographiques peuvent être décrites par un format fixe utilisant ainsi 42 colonnes: (6F7.0). A l'intérieur d'un champ 7.0, les nombres sont justifiés à droite et peuvent avoir une longueur maximale de 7 caractères numériques.

C. Le fichier VDCONST:

La matrice «construction» est structurée de manière similaire (Fig. I.16). Elle contient 385 enregistrements physiques présentant le NUMERO dans les colonnes 1 à 4 puis les 3 variables quantitatives en format fixe sans partie décimale. On peut décrire ce format par (3F10.0), indiquant que chaque variable est inscrite dans un champ de 10 caractères au maximum, utilisant ainsi les colonnes 5 à 34.

5401	709	109	99
5402	896	162	146
5403	49	7	5
5404	163	36	25
5405	452	192	146
5406	171	26	26
5407	404	114	70
5408	120	17	21
5409	1230	348	250
5410	880	98	55
5411	647	169	114
5412	37	14	13
5413	130	29	16
5414	398	107	51
5415	200	15	18
5421	108	40	64
5422	336	57	36
....	...	...	...

Figure I.16

Dessin d'enregistrement de VDCONST.DAT

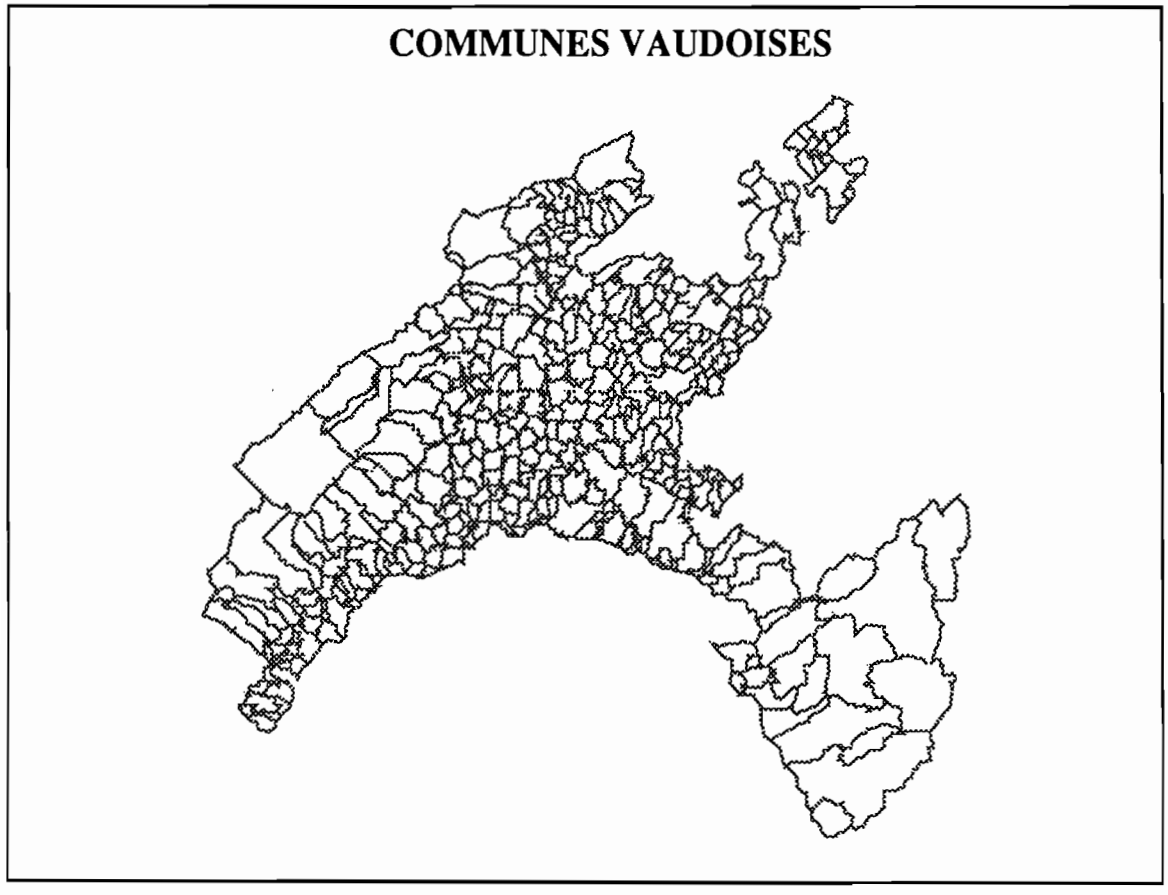


Figure I.17

Le maillage communal du Canton de Vaud

(385 communes)

II

Le système SAS
Présentation générale et dialogue

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 fonctionner sur de gros ordinateurs IBM équipés du système d'exploitation OS/MVS et VM/CMS, on le trouve maintenant sur certains mini-ordinateurs DEC, PRIME et DATA GENERAL. Depuis le dernier trimestre 1985, SAS est également disponible sur micro-ordinateurs IBM PC ou compatibles, et IBM PS.

II.1

Un grand nombre de fonctions

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

BASIC, STATISTIC, IML

A. Les modules d'intérêt général:

BASICS assure la gestion des données ainsi que leur préparation en vue de leur traitement. C'est le cœur du système.

STATISTICS propose toute une panoplie de techniques d'analyse statistique des plus courantes aux plus sophistiquées.

IML ou Interactive Matrix Language est un langage de programmation matriciel qui étend encore plus les possibilités de traitement scientifique des données. Il ressemble au langage APL.

FSP, AF, DMI

B. Les modules qui simplifient l'utilisation interactive de SAS:

FSP, Full Screen Product, améliore de manière notable les travaux de saisie et de personnalisation de documents.

AF, Application Facility, s'adresse tout particulièrement aux concepteurs d'applications "clés en main" pour construire des menus qui souhaitent proposer des fonctions spécifiques à leurs clients.

DMI, Dialog Manager Interface, permet d'adapter SAS au produit IBM connu sous le nom SPF (Support Program Facility) et ainsi de simplifier le dialogue usager/ordinateur.

GRAPH, OR, ETS, QC

C. Les modules adaptés à des besoins plus spécifiques:

GRAPH offre une grande variété de représentations graphiques des données, allant des simples diagrammes à bâtons aux cartes statistiques bi- ou tridimensionnelles.

OR, Operational Research, est indispensable à tous ceux qui désirent mettre en œuvre les techniques de la Recherche Opérationnelle et de la Programmation Linéaire.

ETS, Econometrics and Time Series, traite les séries statistiques temporelles comme celles qu'analysent les économètres.

QC, Quality Control, assure les analyses statistiques nécessaires au contrôle de fabrication.

II.2

Pourquoi choisir SAS ?

Facilité d'apprentissage

SAS est assez facile à apprendre: l'expérience a montré qu'en quelques jours de stage, un élève consciencieux peut obtenir des résultats très élaborés, impossibles à envisager précédemment.

Outil complet

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 qui proposent une grande variété d'options.

Langage de programmation uniforme

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.

Traite un grand volume de données

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.

Traite des tableaux rectangulaires

SAS traite des tableaux observations/variables: ils correspondent parfaitement à un grand nombre de tableaux statistiques commes les enquêtes par questionnaire ou, en géographie, les matrices d'information spatiale.

Convivialité d'utilisation

SAS est «convivial»: à partir de son terminal l'utilisateur 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.

Système ouvert à des procédures extérieures

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.

Système en évolution constante

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

Groupes d'utilisateurs

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.

II.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.

Sur le 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 ses réactions.

Le superviseur

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 tient 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

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 créent 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.

Les bases de données

Enfin, les bases de données, propres aux utilisateurs, ont un répertoire spécifique qui renferme toutes les informations nécessaires à la localisation et à l'identification des tableaux statistiques qui composent la base. On trouve notamment dans ces répertoires, le nom des tableaux, leur type (données, catalogue graphique...) et leur adresse. En second lieu, chaque tableau de la base comprend un enregistrement qui en décrit les caractéristiques et le contenu, c'est-à-dire le nom et l'identification en clair des variables, leur type (numérique ou alphanumérique) ainsi que leur position. A la suite de ce premier enregistrement descriptif, les observations, composées des variables décrites, sont enregistrées séquentiellement.

II.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.

Etapes DATA et PROC

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

Instruction SAS

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.

Noms SAS

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 (_).

Noms de tableaux SAS

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 définition du nom logique diffère selon le système d'exploitation utilisé; l'annexe donne les indications nécessaires à ce propos. La seconde partie est le nom du tableau à proprement parler.

Prenons, par exemple un tableau contenant les données relatives à la population du Canton de Vaud, VDPOP:

WORK.VDPOP est le nom d'un tableau temporaire.

BASE.VDPOP 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: VDPOP sera compris comme WORK.VDPOP.

Opérateurs

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
- pour la soustraction
* pour la multiplication
/ pour la division

II.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 II.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. Entrée des instructions

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. Analyse lexicale et syntaxique

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. 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. Appel à la bibliothèque SAS

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. Exécution et sorties

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. Fin d'exécution et journal de bord LOG

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, il est nécessaire de maîtriser l'utilisation du DISPLAY MANAGER, le gestionnaire d'affichage propre à SAS.

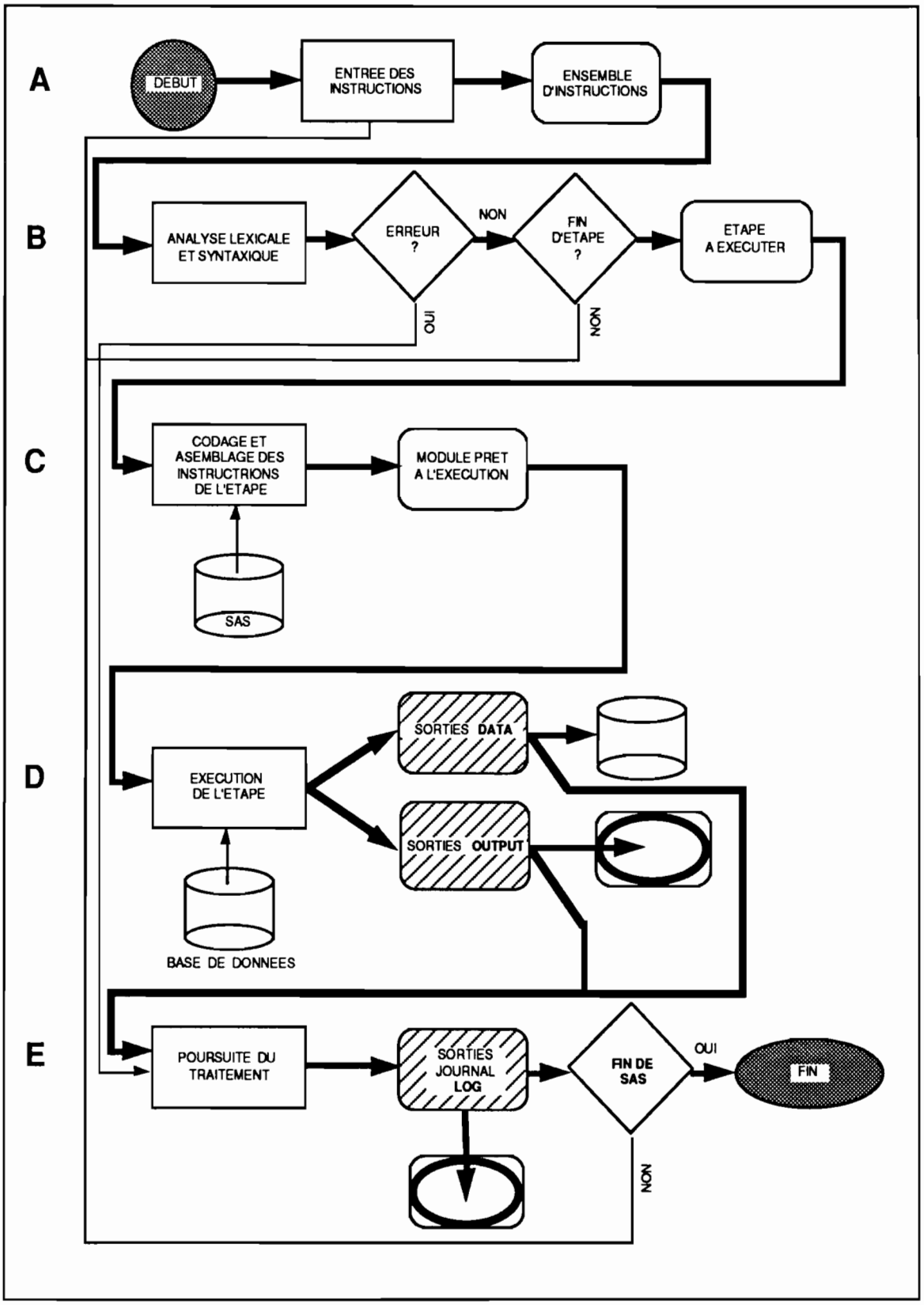


Figure II.1

Le déroulement d'une étape à partir d'un terminal

II.6

Dialoguer avec SAS: le Display Manager

Le gestionnaire d'affichage ou «Display Manager»

L'appel de SAS se solde par l'affichage d'un écran par le Display Manager (sous VAX VMS, il faut entrer tout de suite l'instruction OPTIONS DMS). 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:

L'écran éditeur de programme ou «Program Editor Screen»

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.

L'écran journal de bord ou «Log Screen»

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. II.2).

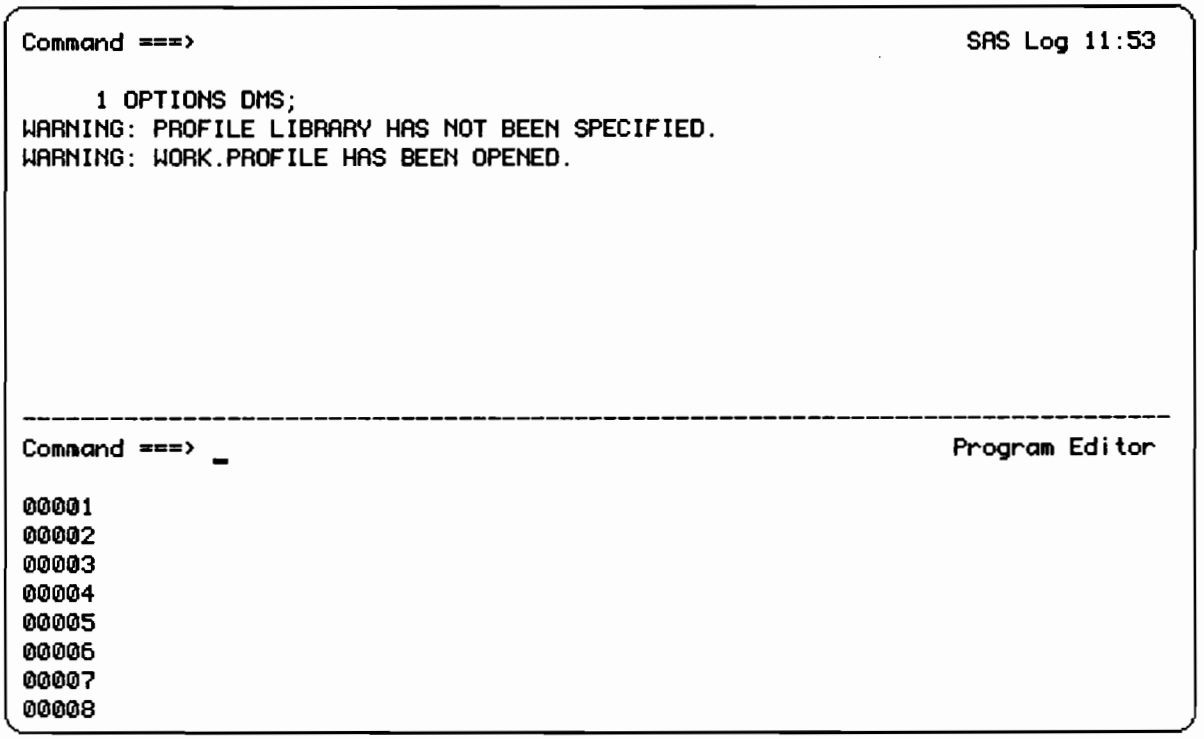


Figure II.2

Le Display Manager après l'appel de SAS

L'écran de sortie des résultats ou «Output Screen»

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.

II.6.1
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 / TOP

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

CHANGE / RCHANGE ou F6

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 F6 correspond à **RCHANGE** pour modifier dans les mêmes termes la prochaine occurrence.

FIND / RFIND ou F5

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 F5 correspondant à **RFIND** assure l'itération de cette recherche jusqu'à la prochaine occurrence.

INCLUDE

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

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

SAVE nom logique du fichier

SUBMIT ou F3

SUBMIT soumet à SAS le programme qui, s'il est correct, est immédiatement exécuté. La touche de fonction F3 correspond à cette commande.

RECALL ou F4

RECALL fait apparaître le dernier programme soumis pour le modifier ou le corriger. La touche de fonction F4 réalise aussi cette opération. Notons qu'en frappant deux fois la touche de

fonction F4, ce sont les deux derniers programmes soumis qui sont rappelés et ainsi de suite.

F7 et F8

Enfin, les touches de fonction F7 et F8 permettent respectivement l'affichage de la partie suivante et précédente du programme.

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

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 et dans ce cas, une seule ligne est insérée.

Dn: supprime

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

Cn: copie

Cn copie (copy) n lignes. Il faut indiquer où doit arriver cette copie dans le programme en entrant soit A (pour «After»),les lignes seront copiées après cette ligne, soit B (pour «Before») les lignes seront copiées avant. Si n est omis, seule la ligne où est entrée la commande 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

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.

II.6.2

Exercice d'entrée d'un programme SAS

L'entrée d'un premier programme doit permettre au lecteur de se familiariser avec le fonctionnement du display manager (Fig. II.3). Voici un exemple de calcul du taux de variation de la population des communes du Canton de Vaud entre 1980 et 1985. Ce petit programme affiche à l'écran l'histogramme de ces taux de variation.

EXEMPLE PGMII-1
Taux de variation de la population entre 1980 et 1985

Ligne de commentaire:
*/*commentaire*/*
à partir de la 2ème position sur la ligne

```
/*Définition du nom logique de base*/
/*dépend du système d'exploitation*/

/*Sélection des variables*/
    DATA POPU; SET BASE.VDPOP;
    KEEP NUMERO POP80 POP85;

/*Afficher le tableau à l'écran*/
    PROC PRINT DATA=POPU(OBS=19);

/*Calculer le taux de variation*/
    DATA POPU; SET POPU;
    TVAR8085=((POP85-POP80)/POP80)*100;

/*Afficher le résultat*/
    PROC PRINT DATA=POPU(OBS=19);
    VAR NUMERO TVAR8085;

/*Afficher un historamme*/
    PROC CHART DATA=POPU;
    HBAR TVAR8085/LEVELS=5;

RUN;
```

II.6.3

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. II.4 et II.5.). 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 afficher directement 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

```
Command ===>                                                    SAS Log 10:26

     1 options dms;
WARNING: PROFILE LIBRARY HAS NOT BEEN SPECIFIED.
WARNING: WORK.PROFILE HAS BEEN OPENED.









--------------------------------------------------------------------------------
Command ===> _                                                  Program Editor
NOTE: 22 lines copied.
00001  /*Definition du nom logique de base*/
00002                  LIBNAME BASE '[]';
00003
00004  /*Selection des variables*/
00005           DATA POPU; SET BASE.VDPOP;
00006   KEEP NUMERO POP80 POP85;
00007
00008  /*Afficherle tableau a l'ecran */
```

Figure II.3

Le programme dans le Program Editor

```
                        SAS      10:23 WEDNESDAY, MARCH 8, 1989  1

        OBS     NUMERO     POP80     POP85

          1       5401      6233      6407
          2       5402      4843      4863
          3       5403       191       210
          4       5404       270       291
          5       5405       827       856
          6       5406       750       796
          7       5407      2057      1941
          8       5408       427       463
          9       5409      4429      4648
         10       5410       895       903
         11       5411      1000       969
         12       5412       222       259
         13       5413       780       881
         14       5414      3573      3582
         15       5415       823       798
         16       5421       833       934
         17       5422      1958      2117
         18       5423       402       396
         19       5424       150       162
```

Figure II.4

La sortie du PROC PRINT

```
                                    SAS      10:24 WEDNESDAY, MARCH 8, 1989  5

                                FREQUENCY BAR CHART
MIDPOINT
TVAR8085                                              FREQ  CUM.   PERCENT    CUM.
                                                            FREQ             PERCENT
          |
        0 |************************************ 273   273    71.09    71.09
          |
       25 |**************                       104   377    27.08    98.18
          |
       50 |*                                      6   383     1.56    99.74
          |
       75 |                                       0   383     0.00    99.74
          |
      100 |                                       1   384     0.26   100.00
          |
          ----+---+---+---+---+---+---+---+---+
             30  60  90  120 150 180 210 240 270

                            FREQUENCY
```

Figure II.5

La sortie du PROC CHART

```
Command ===> _                                                    SAS Log 10:28

NOTE: THE DATA STEP USED 00:00:00.55 CPU SECONDS, 23 PAGEFAULTS.
    15              PROC PRINT DATA=POPU(OBS=20);
    16   VAR NUMERO TVAR8085;
    17  /*Afficher un histogramme*/
    18
NOTE: THE PROCEDURE PRINT USED 00:00:00.41 CPU SECONDS, 7 PAGEFAULTS.
NOTE: THE PROCEDURE PRINTED PAGES 3 THROUGH 4.
    18             PROC CHART DATA=POPU;
    19   HBAR TVAR8085/LEVELS=5;
    20  RUN;
NOTE: THE PROCEDURE CHART USED 00:00:00.47 CPU SECONDS, 8 PAGEFAULTS.
------------------------------------------------------------------------------
Command ===>                                                   Program Editor

00001
00002
00003
00004
00005
00006
00007
00008
```

Figure II.6

Extrait du journal de bord

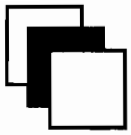
l'image en deux parties composée à sa partie supérieure de l'écran journal de bord et, à sa partie inférieure de l'écran éditeur de programme.

II.6.4

Consultation du journal de bord: l'écran LOG

Comme pour l'écran éditeur de programme, l'écran LOG est composé de deux parties (Fig. II.6): 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 F7 et F8 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 du programme 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.



III

Introduction au traitement des données avec les procédures SAS

Dans un programme SAS, l'étape PROC assure en général un ensemble de traitements sur des tableaux existant déjà. Il aurait donc été logique de faire précéder ce chapitre par un exposé sur l'étape DATA qui réalise le conditionnement des données dans une base SAS. A l'usage, cependant, il est apparu plus efficace d'introduire le lecteur aux procédures élémentaires afin qu'il perçoive toute leur richesse et leur simplicité.

III.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.

Procédures statistiques

A. Statistique:

UNIVARIATE	analyse statistique complète
MEANS	statistique descriptive sommaire
PLOT	tracé de graphiques
CORR	cœfficients de corrélation
RSQUARE	cœfficients de détermination
REG	régression linéaire, etc.
GLM	moindres carrés généralisés

Procédures graphiques

B. Graphique:

GMAP	cartographie thématique

G3D	surfaces en 3 dimensions
GREPLAY	bases de données graphiques

Procédures de calcul matriciel

C. Calcul matriciel:

IML	langage matriciel

Procédures de service

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.

III.2

Structure élémentaire 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

```
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

```
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

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

On trouve cette instruction dans toutes les procédures de régression.

OUTPUT OUT

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;

Chaque procédure possède un jeu d'options qui lui sont particulières. De plus, de nombreuses instructions peuvent s'ajouter à celles présentées ci-dessus 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;

III.3

Déroulement de l'étape PROC

A l'exécution d'une procédure, les résultats sont affichés sur l'écran OUTPUT et peuvent être consultés à l'aide des touches de fonction F7 et F8; 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 on peut aussi envoyer ces résultats sur une imprimante.

III.4

Initiation aux procédures SAS: UNIVARIATE et MEANS

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. La bibliographie indique quelques ouvrages utiles dans diverses branches des sciences sociales.

Afin de bien comprendre le fonctionnement des diverses procédures, un tableau permanent a été construit à partir des 3 tableaux de base VDGEO, VDPOP, VDCONST. Ce nouveau tableau, nommé VDSTAT, comprend 18 variables, pour l'ensemble des communes du canton de Vaud. Voici le texte du programme qui permet de calculer les divers indicateurs statistiques. La signification des instructions des différentes étapes DATA sera exposée dans le prochain chapitre.

EXEMPLE PGMIII-1
Association de tableaux
Calcul d'indicateurs

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Association des 3 tableaux de BASE*/
PROC SORT DATA=BASE.VDGEO OUT=VDGEO;
   BY NUMERO;
PROC SORT DATA=BASE.VDPOP OUT=VDPOP;
   BY NUMERO;
PROC SORT DATA=BASE.VDCONST
   OUT=VDCONST;
   BY NUMERO;

DATA VDSTAT;
   MERGE VDGEO VDPOP VDCONST;
   BY NUMERO;

/*Calcul des indicateurs statistiques*/
DATA VDSTAT; SET VDSTAT;
   DENS80=POP80/SURFACE;
   DENS85=POP85/SURFACE;
   TPOP6070=((POP70-POP60)/POP60)*100;
   TPOP7080=((POP80-POP70)/POP70)*100;
   CONSTOT=SUM(OF CONS60 CONS70 CONS80);
   PCTCON60=(CONS60/CONSTOT)*100;
   PCTCON70=(CONS70/CONSTOT)*100;
   PCTCON80=(CONS80/CONSTOT)*100;
   PCTJEU80=(JEUNES/POP80)*100;
   PCTETR85=(ETR85/POP85)*100;
```

```
  XLAUS=538;
  YLAUS=152;
  DXLAUS=(X-XLAUS)*(X-XLAUS);
  DYLAUS=(Y-YLAUS)*(Y-YLAUS);
  DISTLAUS=SQRT(DXLAUS+DYLAUS);
  KEEP NUMERO NOM X Y DISTRICT REGION
  RUMLEY AGG80 DENS80 DENS85 TPOP6070
  TPOP7080 PCTCON60 PCTCON70 PCTCON80
  PCTJEU80 PCTETR85 DISTLAUS;

DATA BASE.VDSTAT; SET VDSTAT;
RUN;
```

III.4.1

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 minimale est extrêmement simple:

PROC UNIVARIATE

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

Voici le programme pour le cas du tableau permanent VDSTAT:

EXEMPLE PGMIII-2
PROC UNIVARIATE

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

PROC UNIVARIATE DATA=BASE.VDSTAT
  PLOT NORMAL;
  VAR PCTJEU80;
  ID NUMERO;
RUN;
```

La sortie des résultats est organisée en cinq parties. La première contient les moments (Fig. III.1), c'est-à-dire des paramètres de centralité, de dispersion de forme et de symétrie des distributions statistiques. Les paramètres les plus courants sont:

N nombre d'observations non manquantes
SUM somme des valeurs
MEAN moyenne arithmétique

```
                                    SAS     14:14 WEDNESDAY, MARCH 8, 1989 69

                                UNIVARIATE

VARIABLE=PCTJEU80


                                MOMENTS

N                 385  SUM WGTS          385
MEAN          19.5002  SUM            7507.58
STD DEV       3.74776  VARIANCE       14.0457
SKEWNESS   -0.157633  KURTOSIS      0.586014
USS           151793  CSS            5393.55
CV            19.2191  STD MEAN      0.191004
T:MEAN=0      102.093  PROB>|T|        0.0001
SGN RANK      37152.5  PROB>|S|        0.0001
NUM ^= 0          385
D:NORMAL   0.0291933  PROB>D             >.15
```

Figure III.1

Moments d'une distribution avec UNIVARIATE

```
                                    SAS     14:14 WEDNESDAY, MARCH 8, 1989 70

                                UNIVARIATE

VARIABLE=PCTJEU80


                            QUANTILES(DEF=4)

100% MAX      30.8824      99%       27.8868
 75% Q3        21.948      95%       25.7808
 50% MED      19.5219      90%       24.2422
 25% Q1       16.9468      10%       15.1244
  0% MIN      5.55556       5%       13.6812
                             1%       10.2681
 RANGE        25.3268
 Q3-Q1        5.00122
 MODE         16.6667
```

Figure III.2

Quantiles d'une distribution avec UNIVARIATE

```
                                    SAS     14:15 WEDNESDAY, MARCH 8, 1989 71

                                  UNIVARIATE

VARIABLE=PCTJEU80


                                  EXTREMES

                    LOWEST    ID          HIGHEST    ID
                  5.55556(5567    )       27.5(5920    )
                   5.7971(5502    )     27.6923(5862    )
                  6.79612(5674    )     29.0816(5644    )
                  10.8333(5827    )      29.703(5667    )
                  10.9756(5925    )     30.8824(5937    )
```

Figure III.3

Extrêmes d'une distribution avec UNIVARIATE

```
                                    SAS     14:15 WEDNESDAY, MARCH 8, 1989 72

                                  UNIVARIATE

VARIABLE=PCTJEU80

                    HISTOGRAM                    *          BOXPLOT
      31+**                                      1             0
        .*                                       2             0
        .*******                                11             |
        .******************                     29             |
        .*********************************      52             |
        .************************************************* 80  +------+
        .****************************************          69  *--+---*
        .*********************************************     75  +------+
        .**************************                      43     |
        .**********                                      15     |
        .****                                             5     |
        .
        .*                                               1     0
       5+**                                              2     0
          ----+----+----+----+----+----+----+----+----+
          * MAY REPRESENT UP TO 2 COUNTS
```

Figure III.4

Histogramme d'une distribution avec UNIVARIATE

VARIANCE variance
STD DEV «standard deviation», ou écart-type
SKEWNESS coefficient d'assymétrie,
vaut 0 si la distribution est normale;
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 cœfficient d'aplatissement,
vaut 0 si la distribution est normale;
si > 0, elle est aplatie;
si < 0, elle est resserrée.
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 (Fig. III.2) correspondant à des proportions de l'effectif total des observations. Dans les deux colonnes de gauche figurent les quartiles; dans celles de droite, apparaissent les centiles extrêmes. Au-dessous, on obtient l'étendue, l'intervalle interquartile et le mode.

La troisième partie (Fig. III.3) liste les cinq observations ayant les plus petites valeurs et les cinq ayant les plus fortes. En regard de chaque valeur figure l'identifiant de l'observation donné par le paramètre ID.

Avec la quatrième partie commence une série de trois représentations graphiques très utiles correspondant aux options PLOT et NORMAL de la procédure. L'option PLOT (Fig. III.4) affiche un diagramme à bâtons qui donne une idée de la forme de la distribution. Le second graphique est un diagramme en boîte («BoxPlot»), proposé par le statisticien Tuckey; il permet de 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. Le troisième graphique, correspondant à l'option NORMAL (Fig. III.5), montre les distorsions de la variable par rapport à la distribution normale. Les valeurs de la distribution observée sont indiquées par le signe *, celles de la distribution théorique par le signe +. L'échelle des abscisses est mesurée en écarts-types par rapport à la moyenne; celle des ordonnées donne les valeurs extrêmes

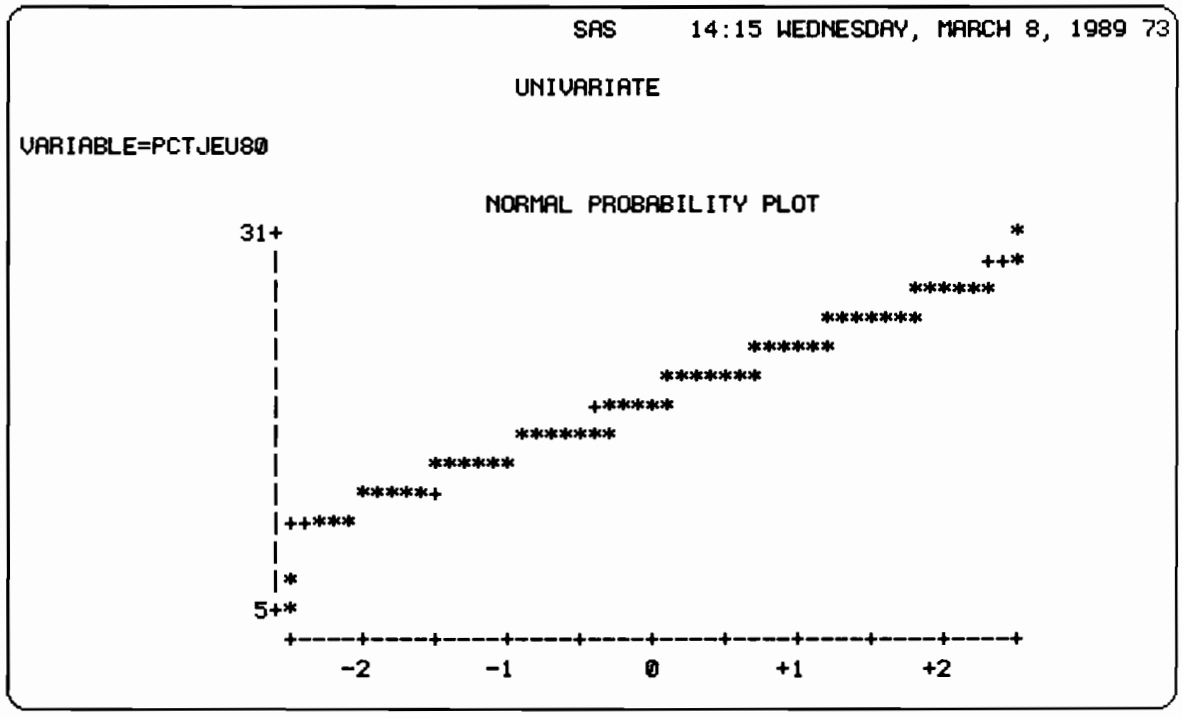


Figure III.5

Ajustement d'une distribution à la loi normale avec UNIVARIATE

SAS 14:54 WEDNESDAY, MARCH 8, 1989 1

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	STD ERROR OF MEAN
X	385	537.21	17.63	499.00	583.00	0.90
Y	385	163.03	16.51	119.00	200.00	0.84
DENS80	385	2.37	6.80	0.07	71.41	0.35
DENS85	385	2.48	6.66	0.08	67.03	0.34
TPOP6070	385	8.45	33.78	-37.10	221.59	1.72
TPOP7080	385	14.72	31.26	-36.59	206.85	1.59
PCTCON60	385	73.28	17.97	21.72	100.00	0.92
PCTCON70	385	11.80	7.91	0.00	37.87	0.40
PCTCON80	385	14.92	12.62	0.00	61.76	0.64
PCTJEU80	385	19.50	3.75	5.56	30.88	0.19
PCTETR85	385	8.62	7.49	0.00	37.77	0.38
DISTLAUS	385	23.97	11.39	0.00	57.43	0.58

Figure III.6

Paramètres calculés avec MEANS

de la variable étudiée.

PROC UNIVARIATE apparaît donc très complète pour décrire une distribution statistique. Elle est cependant très gourmande en ressources. La procédure MEANS peut lui être préférée lorsqu'on ne désire que les principaux paramètres d'une distribution.

III.4.2
La procédure MEANS

La syntaxe de la procédure MEANS est la suivante:

PROC MEANS

```
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 VDSTAT (Fig. III.6):

EXEMPLE PGMIII-3
PROC MEANS

```
/*Définition du nom logique base*/
/*dépend du système d'exploitation*/
PROC MEANS DATA=BASE.VDSTAT MAXDEC=2;
RUN;
```

III.4.3
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:

Agrégation d'observations avec PROC MEANS

```
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;
```

Un exemple d'application de cette technique est donné par le programme ci-dessous. Il calcule la population en 1960, 1970, 1980 et 1985 des régions écologiques de RUMLEY.

EXEMPLE PGMIII-4
Agrégation d'observations

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Association des 2 tableaux de base et tri*/
PROC SORT DATA=BASE.VDGEO OUT=VDGEO;
   BY NUMERO;
PROC SORT DATA=BASE.VDPOP OUT=VDPOP;
   BY NUMERO;
DATA VDRUM;
MERGE VDGEO VDPOP;
   BY NUMERO;

/*Tri du tableau selon RUMLEY*/
PROC SORT DATA=VDRUM OUT=VDRUM2;
   BY RUMLEY;

/*Agrégation selon RUMLEY*/
PROC MEANS DATA=VDRUM2
   MAXDEC=0  NOPRINT;
   VAR  POP60 POP70 POP80 POP85;
   OUTPUT OUT=VDPOPRUM
      SUM=RPOP60 RPOP70 RPOP80 RPOP85;
         BY RUMLEY;

/*Affichage du tableau agrégé*/
PROC PRINT DATA=VDPOPRUM;
RUN;
```

Ce très simple exemple (Fig. III.7) 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.

```
                              SAS      14:58 WEDNESDAY, MARCH 8, 1989  2

 OBS     RUMLEY     RPOP60     RPOP70     RPOP80     RPOP85

   1         1     126328     137383     127349     125004
   2         2      76912     113998     129366     137660
   3         3      72423      89170      89238      88354
   4         4     100942     115896     130870     139740
   5         5      52907      55404      51924      52909
```

Figure III.7

Tableau agrégé de la population par région «écologique»

IV

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

L'une des situations courantes dans laquelle peut se trouver un utilisateur de SAS est la suivante: il a fait saisir ses données sur un support quelconque, disquette ou bande magnétique par exemple. 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.

IV.1

Nommer un tableau en sortie: l'instruction DATA

Créer un tableau SAS à partir d'un fichier de saisie revient à nommer le tableau, à décrire son contenu et à indiquer un mode de lecture correspondant au dessin d'enregistrement.La structure du programme sera donc la suivante:

DATA pour nommer le nouveau tableau
INFILE pour nommer le fichier de saisie
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

DATA nom du tableau à créer;

En respectant les directives relatives au nom d'un tableau, le nouveau tableau sera permanent ou temporaire. Par exemple:

DATA BASE.VDPOP;

définit un tableau permanent qui contiendra les données de la matrice VDPOP. Le nom logique BASE aura dû être défini préalablement à l'aide de l'instruction appropriée, dépendant du système d'exploitation (voir à ce propos l'annexe relative aux systèmes d'exploitation).

IV.2

Nommer un fichier en entrée: 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

INFILE nom logique du fichier en entrée;

Par exemple:

INFILE ENTREE;

définit un fichier en entrée dont le nom logique ENTREE aura dû être défini préalablement à l'aide de l'instruction appropriée, dépendant du système d'exploitation (voir à ce propos l'annexe). Cette instruction (ALLOC sous TSO, FILEDEF sous VM/CMS, ASSIGN sous VMS) permettra de désigner le fichier de saisie VDPOP, par le nom logique ENTREE.

IV.3

Nommer les variables et préciser le mode de lecture : 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

INPUT liste de noms de variables et mode de lecture;

IV.3.1

Nommer les variables

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 VDPOP, on devrait écrire:

```
INPUT NUMERO $ POP60 POP70 POP80 POP85
   ETR85 JEUNES;
```

Notons que les noms des variables alphanumériques doivent nécessairement être suivis du caractère dollar ($) qui précise ce type de donnée. Il est toujours préférable de définir les variables qui ne se prêteront pas à des calculs arithmétiques comme alphanumériques (codes, variables discrètes ou qualitatives, etc.).

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 VDPOP, on obtiendrait:

```
INPUT NUMERO $ POP1-POP4 ETR85 JEUNES;
```

ce qui correspond à

```
INPUT NUMERO $ POP1 POP2 POP3 POP4 ETR85
   JEUNES;
```

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.

IV.3.2

Choisir un mode de lecture

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 (cf. Chap. I.4); il faut, avant la saisie, réfléchir à la manière la plus efficace d'enregistrer son information, sachant qu'elle devra être relue pour être transférée dans une base.

Mode liste

A. Le mode liste.

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 modulable. 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 VDPOP, on aura comme précédemment:

```
INPUT NUMERO $ POP60 POP70 POP80 POP85 ETR85
  JEUNES;
```

Le dessin d'enregistrement est donc changeant, ce qui présente quelques inconvénients, conséquences directes de la simplicité de ce procédé, et qui peuvent avoir des effets déplorables si l'on n'y prend pas garde. Tout d'abord, le séparateur de valeurs étant, par défaut, le caractère espace, les variables alphanumériques ne doivent en aucun cas contenir cette valeur, qui peut, cependant, être redéfinie par l'utilisateur. Notons aussi que, par défaut également, 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 disposer d'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, lors 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.

Mode colonne

B. Le mode colonne.

Dans le cas du 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 VDPOP,

l'instruction INPUT a la forme suivante:

```
INPUT NUMERO $ 1-4
   POP60 5-11 POP70 12-18 POP80 19-25
   POP85 26-32 ETR85 33-39 JEUNES 40-46;
```

Les erreurs possibles avec le mode liste sont donc totalement évitées mais, lorsque le nombre de variables est important, des erreurs sur la position 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 POP85 dans le tableau VDPOP, on écrirait:

```
INPUT  NUMERO  $  1-4
   POP60 5-11  POP70 12-18  POP80 19-25
   ETR85 33-39  JEUNES 40-46;
```

En raison de sa simplicité d'utilisation, le mode colonne devrait être préféré au mode liste lorsque le nombre de variables n'excède pas la vingtaine.

Mode format

C. Le mode format.

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 (caractères, binaire, etc.); SAS propose un grand nombre de formats. Pour les variables numériques enregistrées en caractères, le format le plus courant est du type W.D où W est la longueur d'une zone et D le nombre de décimales. Dans le cas de la matrice VDPOP, l'instruction INPUT pourrait s'écrire de la manière suivante:

```
INPUT  NUMERO  $ (POP60 POP70 POP80 POP85
   ETR85  JEUNES) (7.0);
```

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 NUMERO et POP85, l'instruction INPUT deviendrait:

```
INPUT
  à6 (POP60 POP70 POP80) (7.0)
     +7 (ETR85 JEUNES) (7.0);
```

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.

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 œuvre 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 CODPOP:

```
INPUT NUMERO $ 1-4 POP60 5-11
      POP70
      (POP80 POP85) (7.0)
      ETR85 33-39
      JEUNES 7.0;
```

Les variables NUMERO et POP60 sont lues en mode colonne, POP70 en mode liste; POP80 et POP85 en mode format, ETR85 à nouveau en mode colonne et JEUNES encore 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 très souvent le cas lorsque les volumes d'information sont très grands.

IV.4

Identifier les variables: l'instruction LABEL

Le choix des noms de variables doit être fait de manière à ce qu'ils aient une signification aussi claire que possible pour l'utilisateur des données. Dans le cas du tableau VDPOP, POP80 est une variable contenant la population des communes

du canton de Vaud en 1980; attention, cependant aux unités: ce nombre est-il exprimé en habitants, en milliers d'habitants? Pour la matrice VDGEO, la signification du nom de la variable RUMLEY est évidente pour celui qui a créé le tableau (régions «écologiques» définies par P.A. Rumley), mais assez énigmatique pour un autre usager. C'est pourquoi il est utile de compléter chaque nom de variable par un libellé, dans tous les cas où les données peuvent servir à d'autres personnes, 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. C'est l'instruction LABEL qui affecte un libellé à chaque variable; elle s'écrit:

LABEL

```
LABEL  nom de variable='libellé'
       nom de variable='libellé'    etc... ;
```

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 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 VDPOP, les libellés des variables s'écrivent de la manière suivante:

```
LABEL
   NUMERO='NUMERO DE COMMUNE OFS'
   POP60='POPULATION TOTALE EN 1960'
   POP70='POPULATION TOTALE EN 1970'
   POP80='POPULATION TOTALE EN 1980'
   POP85='POPULATION MOYENNE EN 1985'
   ETR85='POPULATION ETRANGERE EN 1985'
   JEUNES='POPULATION DE 0 A 14 ANS EN 1980';
```

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 personnes utilisant aussi SAS, sans avoir à donner la moindre information complémentaire. C'est un critère important du choix de SAS comme système de gestion des données.

IV.5

Programmes d'entrée des matrices VDGEO, VDPOP et VDCONST

Ces programmes ne renferment pas de difficulté particulière. Le néophyte aura donc tout intérêt à les essayer de manière à se rendre bien compte du rôle de chaque instruction.

A. Lecture de la matrice VDGEO.

EXEMPLE PGMIV-1
Lecture de la matrice-VDGEO

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Définition du nom logique IN*/
/*dépend du système d'exploitation*/

/*Lecture de la matrice*/
   DATA BASE.VDGEO;
   LENGTH NOM $ 30;
      INFILE IN;
   INPUT NUMERO $ 1-4 NOM $
      #2
      X Y
      DISTRICT $ REGION $ RUMLEY $ AGG80 $
      SURFACE;
   LABEL
      NUMERO='NUMERO DE COMMUNE OFS'
      NOM='NOM DE COMMUNE'
      X='ABSCISSE DU CENTRE DE COMMUNE'
      Y='ORDONNEE DU CENTRE DE COMMUNE'
      DISTRICT='DISTRICT'
      REGION='REGION PHYSIOGRAPHIQUE'
      RUMLEY='REGION ECOLOGIQUE RUMLEY'
      AGG80='CODE AGGLOMERATION'
      SURFACE='SUPERFICIE NETTE EN TERRE
         (HA)';
   RUN;
```

Le # permet de continuer l'input sur le second enregistrement de l'observation en cours de lecture.

B. Lecture de la matrice VDPOP.

EXEMPLE PGMIV-2
Lecture de la matrice VDPOP

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Définition du nom logique IN*/
/*dépend du système d'exploitation*/
```

```
/*Lecture de la matrice*/
  DATA BASE.VDPOP;
    INFILE IN;
  INPUT NUMERO $ 1-4
    (POP60 POP70 POP80 POP85) (7.0)
    (ETR85 JEUNES) (7.0);
  LABEL NUMERO='NUMERO DE COMMUNE OFS'
    POP60='POPULATION TOTALE EN 1960'
    POP70='POPULATION TOTALE EN 1970'
    POP80='POPULATION TOTALE EN 1980'
    POP85='POPULATION MOYENNE EN 1985'
    ETR85='POPULATION ETRANGERE EN 1985'
    JEUNES='POPULATION DE 0 A 14 ANS EN
      1980';
  RUN;
```

C. Lecture de la matrice VDCONST

EXEMPLE PGMIV-3
Lecture de la matrice VDCONST

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Définition du nom logique IN*/
/*dépend du système d'exploitation*/

/*Lecture de la matrice*/
  DATA BASE.VDCONST;
    INFILE IN;
  INPUT NUMERO $ 1-4
    (CONS60 CONS70 CONS80) (10.0);
  LABEL NUMERO='NUMERO DE COMMUNE OFS'
    CONS60='IMMEUBLES CONSTRUITS AVANT
      1960'
    CONS70='IMMEUBLES CONSTRUITS DE 1960 A
      1970'
    CONS80='IMMEUBLES CONSTRUITS DE 1970 A
      1980';
  RUN;
```

IV.6

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

Les deux procédures CONTENTS et PRINT permettent de connaître le contenu d'un tableau dans une base SAS, mais

de manière assez différente.

A. Lister le répertoire de la base : la procédure CONTENTS.

PROC CONTENTS

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.VDPOP;
```

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

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

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

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

liste le répertoire du tableau VDPOP 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 le répertoire de la base sans les répertoires des tableaux qui la composent.

B. Afficher les valeurs d'un tableau: la procédure PRINT.

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

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

affiche les valeurs de toutes les variables composant le tableau VDPOP.

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

affiche les valeurs de toutes les variables composant le tableau VDPOP, en arrondissant à trois décimales et en adoptant une présentation uniforme qui facilite la consultation (Fig. IV.1). Notons qu'une colonne supplémentaire OBS donne le rang de chaque observation.

On peut également n'imprimer que les n premières observations d'un tableau (Fig. IV.2) de la manière suivante:

```
PROC PRINT DATA=BASE.VDPOP(OBS=8)
  UNIFORM ROUND;
```

SAS 17:34 WEDNESDAY, MARCH 8, 1989 47

OBS	NOM	DENS85	LDENS85	DISTLAUS
1	RENENS	55.9141	4.02382	3.6056
2	PRILLY	53.2441	3.97489	2.8284
3	MORGES	35.4512	3.56816	10.0000
4	CHAVANNES-PRES-RENENS	30.4524	3.41616	4.1231
5	LAUSANNE	30.1869	3.40741	0.0000
6	PULLY	25.4332	3.23606	2.0000
7	PAUDEX	23.7143	3.16608	3.1623
8	PREVERENGES	17.1905	2.84436	8.0000
9	ECUBLENS	15.4828	2.73973	5.0990
10	EPALINGES	13.3341	2.59032	5.0000
11	ST-SULPICE	13.0432	2.56827	6.0000
12	BUSSIGNY-PRES-LAUSANNE	11.6091	2.45179	7.2111
13	CRISSIER	9.2896	2.22890	5.6569
14	ROMANEL-SUR-LAUSANNE	8.3659	2.12416	5.3852
15	LUTRY	8.0048	2.08004	4.1231
16	CULLY	7.7857	2.05229	8.5440
17	TOLOCHENAZ	6.7394	1.90797	12.0416
18	BELMONT-SUR-LAUSANNE	6.5410	1.87810	4.0000
19	DENGES	5.9188	1.77813	7.0711
20	CHESEAUX-SUR-LAUSANNE	5.6258	1.72737	8.2462
21	ECHANDENS	5.1933	1.64739	7.2801
22	GRANDVAUX	4.9494	1.59929	7.2801
23	CUGY	4.8287	1.57459	7.0711
24	LE-MONT-SUR-LAUSANNE	4.3414	1.46820	5.0990
25	JOUXTENS-MEZERY	3.7989	1.33473	4.4721
26	VILLETTE	3.6985	1.30794	6.3246
27	LONAY	3.2892	1.19066	8.0623
28	ECHICHENS	2.7376	1.00710	10.0499
29	CHIGNY	2.7195	1.00045	12.0416
30	PENTHAZ	2.7146	0.99867	11.4018
31	VILLARS-STE-CROIX	2.3450	0.85230	7.8102
32	MORRENS	2.0516	0.71863	8.0000
33	VUFFLENS-LE-CHATEAU	2.0322	0.70915	12.0416
34	LES CULLAYES	1.9722	0.67916	10.8167
35	FROIDEVILLE	1.6676	0.51139	9.8489
36	SULLENS	1.6666	0.51083	10.2956
37	SAVIGNY	1.5618	0.44586	8.2462
38	VUFFLENS-LA-VILLE	1.5570	0.44277	9.8995
39	MEX	1.3429	0.29488	9.2195
40	BRETIGNY-SUR-MORRENS	1.2245	0.20258	9.0554
41	MONTPREVEYRES	0.6308	-0.46076	11.4018
42	VILLARS-TIERCELIN	0.5349	-0.62562	13.4164

Figure IV.1

Affichage du contenu du tableau VDPOP: agglomération de Lausanne

SAS 16:16 WEDNESDAY, MARCH 8, 1989 2

OBS	NUMERO	POP60	POP70	POP80
1	5401	4381	6532	6233
2	5402	4667	5069	4843
3	5403	209	191	191
4	5404	253	271	270
5	5405	706	752	827
6	5406	855	734	750
7	5407	2241	2752	2057
8	5408	459	460	427

OBS	POP85	ETR85	JEUNES
1	6407	1408	1184
2	4863	756	891
3	210	20	37
4	291	13	56
5	856	140	167
6	796	105	148
7	1941	611	299
8	463	39	78

Figure IV.2

Affichage partiel des valeurs du tableau VDPOP

IV.7

Modification 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. L'appel de la procédure est le suivant:

PROC FSEDIT

PROC FSEDIT DATA=nom du tableau à modifier;

Bien que la procédure FSEDIT permette une initialisation directe des nouveaux tableaux, il est apparu, à l'usage, plus simple de procéder à une initialisation par une étape DATA. Cette opération permet de créer le tableau SAS et de définir correctement les variables par leur nom et leur libellé. Dans le cas de la matrice VDPOP, on procéderait de la manière suivante:

EXEMPLE PGMIV-4
Utilisation de PROC FSEDIT

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau*/
   DATA BASE.VDPOP;
   LENGTH  NUMERO $ 4;
      NUMERO=' ';
      POP60=.;
      POP70=.;
      POP80=.;
      POP85=.;
      ETR85=.;
      JEUNES=.;
   LABEL NUMERO='NUMERO DE COMMUNE'
      POP60='POPULATION TOTALE EN 1960'
      POP70='POPULATION TOTALE EN 1970'
      POP80='POPULATION TOTALE EN 1980'
      POP85='POPULATION MOYENNE EN 1985'
      ETR85='POPULATION ETRANGERE EN 1985'
      JEUNES='POPULATION  0 A 14 ANS EN 1980';

/*Saisie de la matrice*/
   PROC FSEDIT DATA=BASE.VDPOP;

   RUN;
```

L'exécution de ce programme crée un tableau permanent comprenant une observation et où toutes les valeurs sont manquantes. A l'instruction PROC FSEDIT, l'image d'édition s'affiche. Chaque variable à saisir est un champ identifié 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 de la première observation apparaît en haut de l'écran, à droite. Lorsque toutes les variables d'une observation sont entrées, on peut passer à l'observation suivante à l'aide de la touche de fonction F9; 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 F3 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. Les deux commandes suivantes sont très fréquemment utilisées pour repérer une valeur donnée sur une variable du tableau en cours d'édition.

NAME
LOCATE

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

Ceci est très utile pour repérer une valeur fausse: 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 F5 doit être utilisée dès la seconde fois: elle évite d'entrer à nouveau les commandes NAME et LOCATE.

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 F7 provoque l'affichage de l'observation précédente, la touche de fonction F8 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).

V

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, la densité de population par hectare) 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 qui s'effectuent obligatoirement dans une étape DATA.

V.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.

A. 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

SET nom du tableau à copier;

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

DATA VDGEO; SET BASE.VDGEO;

Le tableau temporaire VDGEO aura les mêmes caractéristiques que le tableau permanent VDGEO, c'est-à-dire le même nombre d'observations et de variables. A l'exécution, SAS affiche ces caractéristiques sur le journal de bord.

B. 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

KEEP liste de variables à retenir;

Au second cas:

DROP

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 NOM X Y ;

Ici, on élimine les variables NOM, X et Y. La seconde manière revient à nommer toutes les variables comprises entre une variable de début de liste et une variable de fin de liste et cela d'après leur position dans le tableau:

DROP NOM--REGION;

Cela revient à éliminer les variables comprises entre NOM incluse, et REGION incluse, c'est à dire à DROP NOM, X, Y, CODE et REGION; 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 soient nécessairement dans cet ordre dans

le tableau:

```
DROP POP1-POP4;
```

revient à:

```
DROP POP1 POP2 POP3 POP4;
```

Un seul tiret doit séparer les noms des variables initiale et finale dans ce cas. La création d'un tableau temporaire, copie des variables NUMERO, POP70, POP80 et POP85 du tableau VDPOP pourrait donc prendre les formes suivantes:

EXEMPLE PGMV-1
Création d'un tableau avec sélection de variables.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des variables*/
  DATA VDPOP;
  SET BASE.VDPOP;
    KEEP NUMERO POP70 POP80 POP85;

  DATA VDPOP;
  SET BASE.VDPOP;
    DROP POP60 ETR85 JEUNES;

  DATA VDPOP;
  SET BASE.VDPOP;
    KEEP NUMERO POP70--POP85;

  RUN;
```

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

C. 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

```
IF variable EQ
            GE
            GT
            LE
            LT
            NE valeur 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 ('). Voici le programme qui crée un tableau temporaire nommé LAUSANNE contenant les communes de la seule agglomération de LAUSANNE:

EXEMPLE PGMV-2
Création d'un tableau avec sélection d'observations.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des observations*/
   DATA LAUSANNE;
   SET BASE.VDGEO;
   IF AGG80 NE '15' THEN DELETE;

   RUN;
```

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  EQ
             GE
             GT
             LE
             LT
             NE valeur de la condition;
```

Pour le tableau LAUSANNE, on aurait:

EXEMPLE PGMV-3
Création d'un tableau avec sélection de Lausanne.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des observations*/
   DATA LAUSANNE;
   SET BASE.VDGEO;
   IF AGG80 EQ '15';

   RUN;
```

Notons enfin qu'il est possible de relier entre elles plusieurs conditions par OR (ou) ou bien AND (et) de manière à exprimer une condition composée. Ainsi, pour créer un tableau VDPLAINE, contenant les communes du Moyen Pays (valeur 1 sur la variable REGION du tableau VDGEO) et situées dans la zone intermédiaire plaine (valeur 4 sur la variable RUMLEY), il faut combiner deux conditions de la manière suivante:

EXEMPLE PGMV-4
Création d'un tableau avec sélection des observations 'AND'

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des observations*/
   DATA VDPLAINE; SET BASE.VDGEO;
   IF REGION EQ '1' AND RUMLEY EQ '4';

   RUN;
```

La combinaison des conditions peut donner parfois des résultats 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. On peut aussi, dans le cas de combinaisons complexes enchaîner plusieurs étapes DATA composées chacune d'une condition, par exemple:

EXEMPLE PGMV-5
Création d'un tableau avec combinaison de conditions.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des observations*/
   DATA VDPLAINE; SET BASE.VDGEO;
   IF REGION EQ '1';

   DATA VDPLAINE;
   SET VDPLAINE;
   IF RUMLEY EQ '4';

   RUN;
```

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

Bien entendu, rien ne s'oppose à l'usage conjoint des instructions de sélection des variables et de celles de sélection des observations. Pour le tableau LAUSANNE, on peut ne conserver que les variables NUMERO ainsi que la population de 1960 à 1985:

EXEMPLE PGMV-6
Création d'un tableau avec sélection des observations et des variables.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Création du tableau avec sélection des observations
et des variables*/
   DATA LAUSANNE;
   SET BASE.VDSTAT;
```

```
IF AGG80 EQ '15';
KEEP NUMERO POP60--POP85;

RUN;
```

E. Changer le nom des variables dans le nouveau tableau.

L'instruction RENAME permet de changer le nom des variables dans le tableau créé. Elle s'écrit:

RENAME

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

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

EXEMPLE PGMV-7
Utilisation de RENAME
pour changer les nom des variables

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

  DATA VDPOP; SET BASE.VDPOP;
  RENAME POP60=POP1
    POP70=POP2
    POP80=POP3
    POP85=POP4;

  RUN;
```

Dans le tableau temporaire VDPOP, les variables POP60 à POP85 auront un nouveau nom, respectivement POP1 à POP4. 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:

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

  DATA VDPOP;
    SET BASE.VDPOP;
  RENAME POP60=POP1
    POP70=POP2
    POP80=POP3
    POP85=POP4;
  KEEP POP1 POP2 POP3 POP4;
  RUN;
```

Un message d'erreur indiquant que les variables POP1 à POP4 n'existent pas sera affiché sur le journal de bord. Il faudrait écrire:

EXEMPLE PGMV-8
Utilisation de RENAME et KEEP

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDPOP; SET BASE.VDPOP;
RENAME POP60=POP1
       POP70=POP2
       POP80=POP3
       POP85=POP4;
KEEP POP60 POP70 POP80 POP85;
RUN;
```

Attention: KEEP POP60 et non pas POP1; le RENAME n'a d'effet qu'à l'issue de l'étape. A ce stade POP60 n'a pas encore changé de nom.

L'usage des instructions de sélection IF, KEEP, DROP et RENAME présente donc quelques pièges; il est toujours souhaitable de s'assurer du contenu du nouveau tableau à l'aide des procédures CONTENTS et PRINT.

V.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

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

Par exemple, si à partir des tableaux VDGEO et VDPOP, on désirait constituer un unique tableau renfermant les variables NUMERO, NOM et POP85, on soumettrait le programme suivant:

EXEMPLE PGMV-9
Utilisation de MERGE pour associer plusieurs tableaux.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDPOP;
   MERGE BASE.VDGEO BASE.VDPOP;
      BY NUMERO;
   KEEP NUMERO NOM POP85;
RUN;
```

A l'exécution de ce programme, SAS vérifiera d'abord la présence de la variable NUMERO, de type alphanumérique dans les deux tableaux; l'ordre alphabétique sera donc requis. Si cela n'était pas le cas, un message d'erreur indiquerait la nécessité du tri de la variable d'association et SAS arrêterait l'exécution de l'étape. Le tableau VDPOP ne contiendrait aucune observation.

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

PROC SORT

```
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 doit être différent du nom du tableau trié. Ainsi, pour réaliser correctement la création du tableau VDPOP, il faut exécuter le programme:

EXEMPLE PGMV-10
Utilisation de PROC SORT pour trier des tableaux à associer ensuite avec MERGE.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

PROC SORT DATA=BASE.VDGEO OUT=GEO;
   BY NUMERO;
PROC SORT DATA=BASE.VDPOP OUT=POP;
   BY NUMERO;
DATA VDPOP; MERGE GEO POP;
   BY NUMERO;
   KEEP NUMERO NOM POP85;
RUN;
```

V.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 a 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, ou encore de fonctions.

A. 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:

```
XLAUS=538;
YLAUS=152;
```

Dans le cas où la constante est alphanumérique, la nouvelle variable doit être définie comme alphanumérique à l'aide de l'instruction LENGTH qui a pour syntaxe:

LENGTH

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

Par exemple, si la constante alphanumérique est 'CANTON DE VAUD', c'est-à-dire composée de 14 caractères (les apostrophes, ou «quotes» en anglais, ne servent qu'à délimiter la constante), il faudra programmer:

```
LENGTH CANTON $ 14;
    CANTON='CANTON DE VAUD';
```

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 VDGEO dans un tableau temporaire VDGEO en lui ajoutant une

variable CANTON initialisée avec la chaîne de caractères 'CANTON DE VAUD', il suffirait de soumettre le petit programme suivant:

EXEMPLE PGMV-11
Ajouter une expression constante.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDGEO;
   SET BASE.VDGEO;
   LENGTH CANTON $ 14;
      CANTON='CANTON DE VAUD';
RUN;
```

Pour l'ensemble des observations, CANTON contiendrait la chaîne de caractères 'CANTON DE VAUD'.

B. 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, cela reste le cas le plus simple:

```
ABSCISSE=X;
```

La variable X doit être présente dans le tableau existant déjà ou avoir été définie avant 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 VDGEO dans un tableau temporaire VDGEO en lui ajoutant une variable NOMCOM contenant les mêmes valeurs que la variable NOM (qui a une longueur de 30 caractères comme l'indique la procédure CONTENTS), le programme prendrait la forme suivante:

EXEMPLE PGMV-12
Ajouter une expression constante qui est un nom de variable.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDGEO;
   SET BASE.VDGEO;
   LENGTH NOMCOM $ 30;
      NOMCOM=NOM;
RUN;
```

C. L'expression est une combinaison 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 cinq: 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 VDPOP, on désire calculer les valeurs d'une variable TVAR6070 représentant le taux de variation de la population de 1960 à 1970:

EXEMPLE PGMV-13
Utilisation d'opérateurs arithmétiques.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDPOP;
   SET BASE.VDPOP;
      TVAR6070=((POP70-POP60)/POP60)*100;
RUN;
```

Niveaux de priorité des opérateurs arithmétiques:

Exponentiation **
Multiplication *
Division /
Addition +
Soustraction -

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. C'est la raison de l'utilisation des parenthèses pour la soustraction dans le programme précédent. D'une manière plus générale, il est toujours préférable de mettre entre parenthèses les différents membres d'une expression arithmétique. Attention cependant à fermer autant de parenthèses qu'il y en a d'ouvertes, sinon SAS détecterait une erreur.

Il n'existe qu'un seul opérateur sur les variables et les constantes alphanumériques; c'est l'opérateur de concaténation (qui s'écrit !!) qui met bout à bout les chaînes en une seule. La variable créée doit avoir une longueur égale à la somme des longueurs des chaînes assemblées.

D. 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; ce peut être une variable ou une constante, ou même une expression numérique.

D.1. Fonctions arithmétiques.

ABS

ABS évalue la valeur absolue
ABS(-2) retourne la valeur 2

SQRT

SQRT évalue la racine carrée
SQRT(2) retourne la valeur 1.414

Illustrons le mode d'utilisation de la fonction SQRT avec le calcul de la distance à Lausanne des communes du canton de Vaud. Il s'agit de la distance euclidienne obtenue par application du théorème de Pythagore:

EXEMPLE PGMV-14
Utilisation d'une fonction SAS:
Calcul de la distance à Lausanne.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA DISTLAU;   SET BASE.VDGEO;
   XLAUS=538;
   YLAUS=152;
   DXLAUS=(X-XLAUS)*(X-XLAUS);
   DYLAUS=(Y-YLAUS)*(Y-YLAUS);
   DISTLAU=SQRT(DXLAUS+DYLAUS);
KEEP NUMERO NOM X Y DISTLAU;
RUN;
```

D.2. Fonction de troncature.

INT

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

D.3. Fonctions mathématiques.

EXP

EXP évalue l'exponentielle
EXP(2) retourne la valeur 7.389

LOG

LOG évalue le logarithme naturel
LOG(2) retourne la valeur 0.693

LOG10

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.

Remarquons que, pour calculer les distances au carré, on a préféré la multiplication plutôt que l'exponentiation, impossible sur les valeurs négatives.

D.4. Fonction Somme.

Il existe une fonction particulière, nommée SUM, qui retourne la somme des valeurs prises par une liste de variables; elle s'écrit:

SUM(OF)

SUM(OF liste de noms de variables);

Cette fonction est très utile pour calculer des totaux en lignes. Par exemple, si l'on désire calculer le nombre d'immeubles dans les communes du Canton de Vaud, on peut sommer les variables CONS60 CONS70 CONS80 du tableau VDCONST:

EXEMPLE PGMV-15
Utilisation de SUM:
Total de la construction '60 à '80.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDCONST; SET BASE.VDCONST;
  TOTCONST=SUM(OF CONS60 CONS70 CONS80);
RUN;
```

On aurait pu également calculer la somme de la manière suivante:

```
TOTCONST=CONS60+CONS70+CONS80;
```

Cela aurait donné un résultat semblable, à une expression près: si l'une des variables présente une valeur manquante, elle sera considérée comme nulle par la fonction SUM, comme manquante par l'opérateur "+". Dans ce dernier cas, le résultat sera donc manquant, même si les autres variables ne manquent pas.

V.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 de remplacer les anciennes valeurs par le résultat de l'expression dans le tableau en cours de création. Le programme suivant remplace les contenus des variables POP60, POP70 et POP80 du tableau VDPOP, par les taux de variation de la population de 1960 à 1985:

EXEMPLE PGMV-16
Affectation d'une expression à une variable existante.

On pourrait aussi redéfinir les labels conformément à la transformation opérée.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

DATA VDPOP; SET BASE.VDPOP;
   POP60=((POP70-POP60)/POP60)*100;
   POP70=((POP80-POP70)/POP70)*100;
   POP80=((POP85-POP80)/POP80)*100;
RUN;
```

V.5

Traiter d'un bloc un ensemble de variables: ARRAY

Dans bien des cas, la même expression s'applique à un ensemble de variables. Par exemple, si l'on désire calculer les densités de population en 1960, 1970, 1980 et 1985, il suffit d'appliquer la division de ces variables par la variable surface, c'est-à-dire:

EXEMPLE PGMV-17
Calcul de la densité.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

PROC SORT DATA=BASE.VDGEO  OUT=GEO;
      BY NUMERO;
PROC SORT DATA=BASE.VDPOP  OUT=POP;
      BY NUMERO;
DATA VDPOP;MERGE GEO POP;
      BY NUMERO;
```

```
   DEN60=POP60/SURFACE;
   DEN70=POP70/SURFACE;
   DEN80=POP80/SURFACE;
   DEN85=POP85/SURFACE;
KEEP NUMERO NOM DEN60--DEN85;
RUN;
```

On peut utilement remplacer la série de divisions par l'instruction ARRAY. En effet, la rédaction d'un tel programme, très répétitif, peut être source d'erreur sur les noms des variables, 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

```
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

```
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

```
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:

EXEMPLE PGMV-18
Calcul de la densité avec ARRAY.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

PROC SORT DATA=BASE.VDGEO OUT=GEO;
     BY NUMERO;
PROC SORT DATA=BASE.VDPOP  OUT=POP;
     BY NUMERO;
DATA VDPOP; MERGE GEO POP;
     BY NUMERO;
   ARRAY DENSITE DEN60 DEN70 DEN80 DEN85;
     DO OVER DENSITE;
     DENSITE=DENSITE/SURFACE;
     END;
KEEP NUMERO NOM DEN60--DEN85;
RUN;
```

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)! Notons qu'il existe d'autres formes de boucle ARRAY, adaptées à d'autres cas de figure plus complexes.

VI

Corrélation et régression

Variable endogène «à expliquer» variable exogène «explicative».

Les techniques de corrélation et de régression linéaires sont nécessaires pour l'étude des liens de dépendance entre les valeurs d'une variable centrale pour la recherche (dite endogène, ou à expliquer) et les valeurs d'une ou de plusieurs variables (dites exogènes, ou explicatives). En géographie, réaliser une étude de corrélation/régression, c'est tenter d'éclairer la répartition spatiale d'un phénomène en formulant un modèle qui spécifie une relation linéaire et 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. Cependant, il est rare que toutes les observations se conforment rigoureusement au modèle; il est alors indispensable 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, dont l'effet ne se fait sentir que dans ces sous-espaces.

VI.1

Méthodologie de la corrélation/régression

La plupart des ouvrages de statistique élémentaire exposent les principes et les techniques du calcul de corrélation et de régression. La bibliographie donne à ce propos quelques indications de lecture.

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 choisir une variable *endogène* et 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 * pème 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:

Visualiser la relation

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 respectives 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.

Mesurer l'intensité de la relation

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 corrélaion linéaire de Pearson. Cette tâche sera confiée à la procédure CORR.

Introduire de nouvelles variables endogènes

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.

Estimer les coefficients d'ajustement de la fonction .

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 étude.

VI.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

1º 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

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. Notons que la procédure GPLOT de SAS/GRAPH réalise le même type de représentation, mais avec une bien meilleure qualité et un plus grand nombre de possibilités comme, par exemple, le tracé direct d'une droite de régression.

2º Coefficients de corrélation: procédure CORR.

La syntaxe de la procédure CORR est extrêmement simple:

PROC CORR

PROC CORR DATA=nom du tableau RANK;

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.

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).

3º 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

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

4º Régression linéaire simple et multiple: procédure REG.

REG est l'une des procédures de régression multiple proposée par SAS. Elle calcule les coefficients de régression, réalise une analyse de la variance et peut stocker estimations et résidus dans un tableau. Elle est composée pincipalement des trois instructions suivantes:

PROC REG

```
PROC REG DATA=nom du tableau;
  MODEL nom variable endogène=liste variables
    exogènes;
  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 sortie est composée de deux groupes de résultats; dans la partie supérieure, on trouve une analyse de variance. Deux indicateurs permettent de juger de la validité de la régression. Tout d'abord, la valeur F permet d'évaluer la puissance explicative du modèle qui peut être testée; PROB>F donne le seuil de non-nullité du rapport expliqué/non-expliqué, ceci signifiant qu'il existe bien une différence entre ce qui est expliqué et ce qui est résiduel. Ensuite, R-SQUARE, le coefficient de détermination, permet de juger de l'intensité de la corrélation. La partie inférieure de la sortie 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.

A l'issue de la régression, les estimations et les résidus sont stockés dans un tableau temporaire ou permanent dont le contenu peut être imprimé par la procédure PRINT; notons qu'un tel tableau de sortie comprend également toutes les variables du tableau dont le nom figure dans l'instruction PROC. Signalons enfin que REG accepte la formulation de modèles polynomiaux, mais leurs termes doivent être calculés au préalable.

Autres procédures de régression:

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

AUTOREG

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 n'est utile que pour l'analyse des séries temporelles.

STEPWISE

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.

GLM

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.

VI.2.1
Un exemple de régression simple: Le gradient des densités dans l'agglomération de Lausanne

La décroissance des densités de population du centre d'une agglomération urbaine vers sa périphérie est un phénomène très souvent rencontré en géographie urbaine. Un tel modèle peut s'exprimer sous la forme d'une équation du type:

Densité = a * distance au centre + b

En général, cette décroissance n'est pas régulière. D'abord très rapide à proximité du centre, elle devient bien plus lente vers l'extérieur. De plus, on observe très souvent une remontée des densités dans la première couronne, qui est donc plus densément habitée que le centre lui-même. L'équation du modèle devient donc:

Densité = $e^{a*\text{distance au centre} + b}$

ou bien encore:

lnDensité = a * distance au centre + b

Le programme ci-dessous permet d'étudier une telle relation dans l'agglomération lausannoise en 1985.

EXEMPLE PGMIV-1
Analyse en surfaces de tendance

```
/*Définition du nom logique BASE*/
/*Dépend du système d'exploitation*/

DATA B; SET BASE.VDSTAT; IF AGG80 EQ '15';
   LDENS85=LOG(DENS85);
PROC SORT DATA=B OUT=C;
      BY DESCENDING DENS85;
PROC PRINT; VAR NOM DENS85 LDENS85
      DISTLAUS;
PROC PLOT; PLOT LDENS85*DISTLAUS /
      VAXIS=1 TO 4.5 BY .4;
PROC REG; MODEL LDENS85=DISTLAUS;
RUN;
```

On spécifie DESCENDING afin d'obtenir le tri à partir de la plus forte valeur jusqu'à la plus faible. C'est l'ordre inverse qui est appliqué par défaut.

La Figure VI.1 est le résultat du PROC PRINT. Les communes étant classées de la plus forte densité à la plus faible, Lausanne n'apparaît qu'en cinquième rang. En effet, cette commune centrale est moins densément peuplée que sa proche périphérie, en raison d'une importante surface non bâtie. Ce premier aspect du modèle est donc vérifié. On observe

SAS 17:34 WEDNESDAY, MARCH 8, 1989 47

OBS	NOM	DENS85	LDENS85	DISTLAUS
1	RENENS	55.9141	4.02382	3.6056
2	PRILLY	53.2441	3.97489	2.8284
3	MORGES	35.4512	3.56816	10.0000
4	CHAVANNES-PRES-RENENS	30.4524	3.41616	4.1231
5	LAUSANNE	30.1869	3.40741	0.0000
6	PULLY	25.4332	3.23606	2.0000
7	PAUDEX	23.7143	3.16608	3.1623
8	PREVERENGES	17.1905	2.84436	8.0000
9	ECUBLENS	15.4828	2.73973	5.0990
10	EPALINGES	13.3341	2.59032	5.0000
11	ST-SULPICE	13.0432	2.56827	6.0000
12	BUSSIGNY-PRES-LAUSANNE	11.6091	2.45179	7.2111
13	CRISSIER	9.2896	2.22890	5.6569
14	ROMANEL-SUR-LAUSANNE	8.3659	2.12416	5.3852
15	LUTRY	8.0048	2.08004	4.1231
16	CULLY	7.7857	2.05229	8.5440
17	TOLOCHENAZ	6.7394	1.90797	12.0416
18	BELMONT-SUR-LAUSANNE	6.5410	1.87810	4.0000
19	DENGES	5.9188	1.77813	7.0711
20	CHESEAUX-SUR-LAUSANNE	5.6258	1.72737	8.2462
21	ECHANDENS	5.1933	1.64739	7.2801
22	GRANDVAUX	4.9494	1.59929	7.2801
23	CUGY	4.8287	1.57459	7.0711
24	LE-MONT-SUR-LAUSANNE	4.3414	1.46820	5.0990
25	JOUXTENS-MEZERY	3.7989	1.33473	4.4721
26	VILLETTE	3.6985	1.30794	6.3246
27	LONAY	3.2892	1.19066	8.0623
28	ECHICHENS	2.7376	1.00710	10.0499
29	CHIGNY	2.7195	1.00045	12.0416
30	PENTHAZ	2.7146	0.99867	11.4018
31	VILLARS-STE-CROIX	2.3450	0.85230	7.8102
32	MORRENS	2.0516	0.71863	8.0000
33	VUFFLENS-LE-CHATEAU	2.0322	0.70915	12.0416
34	LES CULLAYES	1.9722	0.67916	10.8167
35	FROIDEVILLE	1.6676	0.51139	9.8489
36	SULLENS	1.6666	0.51083	10.2956
37	SAVIGNY	1.5618	0.44586	8.2462
38	VUFFLENS-LA-VILLE	1.5570	0.44277	9.8995
39	MEX	1.3429	0.29488	9.2195
40	BRETIGNY-SUR-MORRENS	1.2245	0.20258	9.0554
41	MONTPREVEYRES	0.6308	-0.46076	11.4018
42	VILLARS-TIERCELIN	0.5349	-0.62562	13.4164

Figure VI.1

Classement des communes en fonction de leur densité de population

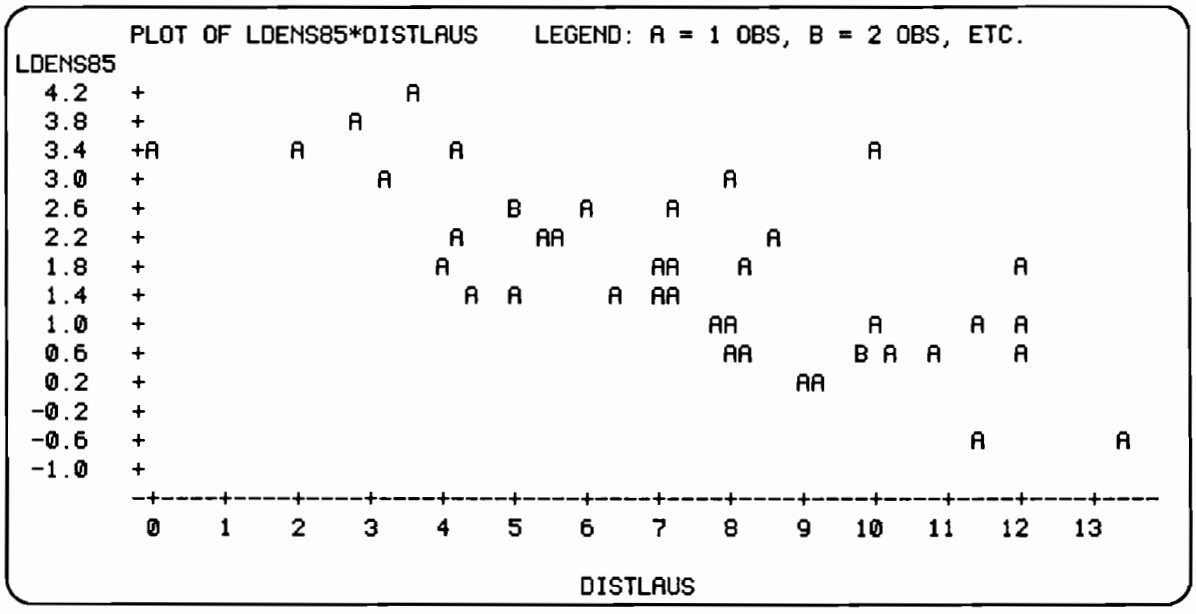


Figure VI.2

Graphique de relation des 42 communes de l'agglomération lausannoise sur la densité de population 1985 et la distance à Lausanne.

DEP VARIABLE: LDENS85

ANALYSIS OF VARIANCE

SOURCE	DF	SUM OF SQUARES	MEAN SQUARE	F VALUE	PROB>F
MODEL	1	29.54398359	29.54398359	44.982	0.0001
ERROR	40	26.27198264	0.65679957		
C TOTAL	41	55.81596623			

ROOT MSE	0.8104317	R-SQUARE	0.5293
DEP MEAN	1.694623	ADJ R-SQ	0.5175
C.V.	47.82373		

PARAMETER ESTIMATES

VARIABLE	DF	PARAMETER ESTIMATE	STANDARD ERROR	T FOR H0: PARAMETER=0	PROB > |T|
INTERCEP	1	3.72204148	0.32713605	11.378	0.0001
DISTLAUS	1	-0.273595	0.04079348	-6.707	0.0001

Figure VI.3

Modèle de régression liant la densité de population 1985 à la distance à Lausanne

également une diminution générale des densités quand s'accroît la distance. La dernière couronne suburbaine comprend les communes les moins denses. Cette première impression est confirmée par le graphique réalisé avec PROC PLOT (Fig. VI.2). La tendance est nette même si le nuage de points est assez dispersé. Les écarts les plus forts correspondent aux deux centres secondaires de l'agglomération, Renens à 3,6 km et Morges à 10 km. On est donc fondé à mener une régression: la variable endogène est le logarithme de la densité de population, la variable exogène représente la distance au centre de Lausanne. Le résultat de cette régression s'exprime par l'équation (Fig. VI.3):

lnDensité 85 = -0,27 Distance à Lausanne + 3.72

Cette relation reste ténue puisque le coefficient de détermination atteint seulement 0,5; le modèle explique une partie importante de la variance de manière significative (au seuil de 0,0001). En conclusion, il apparaît clairement que la présence du lac Léman, influençant l'orientation des voies de communication, perturbe le modèle théorique. La cartographie des valeurs résiduelles devrait permettre de confirmer cette assertion.

VI.3

Analyse en surfaces de tendance avec REG

A. Quand la localisation devient une variable explicative.

L'analyse en surfaces de tendance (AST) est une technique de décomposition d'un phénomène spatial en composantes d'échelles. Son utilisation repose sur l'hypothèse de l'existence de processus spatiaux se déroulant à des échelles différentes. Les mesures statistiques en expriment le cumul, que diverses techniques sont chargées de décomposer. En simplifiant, on distingue plusieurs niveaux différents sur lesquels se développent d'une part des processus généraux affectant non seulement l'aire d'étude, mais également les espaces voisins, et, d'autre part, des processus régionaux couvrant l'espace étudié proprement dit. Enfin, des mouvement locaux, ne concernant que des portions du territoire en question, rendent les répartitions spatiales encore plus complexes.

A chacun des différents niveaux correspond une composante, qui peut être représentée en deux dimensions sous l'aspect d'une carte en isolignes joignant les points de même valeur. Ces cartes de tendances spatiales ont été souvent confondues, à tort, avec les cartes lissées, qui se présentent sous le même aspect; elles ne se rapportent pas à des composantes d'échelles, mais directement à leur cumul simplifié et généralisé. L'informatique offre également la possibilité de tracer des blocs-diagrammes tridimensionnels avec des angles de vue variés, souvent spectaculaires mais aussi bien plus difficiles à étudier. Cette approche peut être généralisée à tout phénomène physique (météorologie, climatologie) ou économique dont les variables représentatives sont mesurées dans un espace bi-dimensionnel, sur des points dont on connaît les coordonnées.

B. La formulation du modèle d'AST.

Les matrices d'information tendancielle sont des tableaux rectangulaires où les lignes figurent les unités spatiales et où les 3 colonnes représentent les coordonnées des centres de ces unités dans un repère orthonormé (x et y), ainsi qu'une mesure statistique du phénomène que l'on prétend décomposer en tendances (z). Le modèle de base de l'analyse en surfaces de tendance postule une relation fonctionnelle telle qu'en chaque point i de la surface, ayant pour coordonnées le couple (x_i,y_i), on puisse estimer une valeur z_i, c'est-à-dire : $z_i=f(x_i,y_i)+e_i$. Ce dernier terme, e_i, est un résidu considéré comme aléatoire et inexplicable par la fonction.

Lorsque les valeurs z_i sont relevées sur une échelle d'intervalles ou de rapports, le modèle de régression polynomial est habituellement adopté. Au premier degré, on obtient une surface plane mais inclinée, qui est ensuite rendue plus complexe par les surfaces quadratiques et cubiques (ordres 2 et 3 du polynôme). Bien entendu, il est possible de poursuivre les ajustements à des degrés encore supérieurs, mais se posent alors de nombreux problèmes liés à la précision des calculs et à la difficulté de compréhension des résultats.

Les différentes surfaces calculées doivent pouvoir être considérées comme des composantes d'échelle et non pas seulement comme de simples ajustements selon des critères géométriques n'ayant rien à voir avec la géographie. C'est bien là que réside la principale difficulté de la méthode. Deux questions importantes doivent trouver une réponse : d'une part, quel est le nombre d'ajustements à interpréter, d'autre part, ces ajustements correspondent-ils à autant de

composantes d'échelles (générale, régionale ou locale). Des indicateurs statistiques, comme l'histogramme des taux de variance expliquée, permettent de déterminer le nombre de composantes à retenir, ainsi que d'en déterminer le rang. L'examen des configurations spatiales présentées par la carte de chaque tendance complète une analyse à la fois difficile et très féconde.

Des variantes de ce modèle général sont bien adaptées lorsque l'échelle de mesure de z_i est nominale (surfaces de probabilités), ou bien lorsque les tendances, au lieu d'être monotones, sont supposées périodiques (analyse spectrale).

C. Un programme d'AST recourant à la procédure REG.

Ce programme SAS, un peu complexe, réalise les calculs nécessaires à une analyse en surfaces de tendance polynomiales jusqu'au degré 5. Il procède en 3 étapes. Tout d'abord, les puissances et les produits croisés des coordonnées des centres des unités spatiales sont calculés dans une étape DATA. Ensuite, six modèles sont formulés dans la procédure de régression REG. Enfin, les estimations à chaque degré du polynôme sont rassemblées dans un seul tableau VDSURF. Ces variables peuvent ensuite être cartographiées.

EXEMPLE PGMVI-2
Analyse en surfaces de tendance

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Calcul des puissances et produits croisés*/
DATA VDSTAT; SET BASE.VDSTAT;
   X2=X*X;
      XY=X*Y;
      Y2=Y*Y;
   X3=X2*X;
      X2Y=X2*Y;
      XY2=X*Y2;
      Y3=Y2*Y;
   X4=X3*X;
      X3Y=X3*Y;
      X2Y2=X2*Y2;
      XY3=X*Y3;
      Y4=Y3*Y;
   X5=X4*X;
      X4Y=X4*Y;
      X3Y2=X3*Y2;
      X2Y3=X2*Y3;
      XY4=X*Y4;
```

```
      Y5=Y4*Y;
    X6=X5*X;
      X5Y=X5*Y;
      X4Y2=X4*Y2;
      X3Y3=X3*Y3;
      X2Y4=X2*Y4;
      XY5=X*Y5;
      Y6=Y5*Y;

/*Régressions polynomiales jusqu'à l'ordre 6*/
PROC REG DATA=DONNEES OUTEST=EQUA;
      M1:MODEL  PCTCON80=X Y ;
   OUTPUT OUT=SURF1  P=ESTIM1 R=RESID1 ;
      M2:MODEL  PCTCON80=X Y
         X2 XY Y2;
   OUTPUT OUT=SURF2  P=ESTIM2 R=RESID2 ;
      M3:MODEL  PCTCON80=X Y
         X2 XY Y2
         X3 X2Y XY2 Y3;
   OUTPUT OUT=SURF3  P=ESTIM3 R=RESID3 ;
      M4:MODEL  PCTCON80=X Y
         X2 XY Y2
         X3 X2Y XY2 Y3
         X4 X3Y X2Y2 XY3 Y4;
   OUTPUT OUT=SURF4  P=ESTIM4 R=RESID4 ;
      M5:MODEL  PCTCON80=X Y
         X2 XY Y2
         X3 X2Y XY2 Y3
         X4 X3Y X2Y2 XY3 Y4
         X5 X4Y X3Y2 X2Y3 XY4 Y5;
   OUTPUT OUT=SURF5  P=ESTIM5 R=RESID5 ;
      M6:MODEL  PCTCON80=X Y
         X2 XY Y2
         X3 X2Y XY2 Y3
         X4 X3Y X2Y2 XY3 Y4
         X5 X4Y X3Y2 X2Y3 XY4 Y5
         X6 X5Y X4Y2 X3Y3 X2Y4 XY5 Y6;
   OUTPUT OUT=SURF6 P=ESTIM6 R=RESID6 ;

/*Assembler les estimations à chaque ordre*/
DATA VDSURF;
MERGE SURF1 SURF2 SURF3 SURF4 SURF5 SURF6;
      BY NUMERO;
   KEEP NUMERO
      ESTIM1 ESTIM2 ESTIM3 ESTIM4 ESTIM5
         ESTIM6 X Y;
RUN;
```

Voir exemple au chapitre X.5.

VII

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 antagonistes 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 dispose de toute une panoplie dont il peut user à volonté, pour peu que soient respectées quelques règles élémentaires d'utilisation de ces techniques.

Depuis plusieurs années, les chercheurs et enseignants regroupés au sein de l'Association pour le Développement et la Diffusion de l'Analyse des Données (ADDAD) proposent une immense bibliothèque de programmes d'un très grand intérêt, tant pour la préparation des tableaux que pour leur traitement par des techniques factorielles ou des classifications automatiques. Elle complète les choix offerts par les procédures SAS, en particulier dans le domaine de l'analyse des données «à la française», et, plus précisément, donne accès à l'analyse des correspondances et aux aides à l'interprétation des résultats d'analyses multivariées. Une procédure ADDAD, disponible sous le système d'exploitation OS/MVS, a pour fonction d'adresser les commandes nécessaires aux programmes de l'ADDAD. L'utilisateur de SAS peut ainsi bénéficier de toutes les ressources de la bibliothèque ADDAD.

Ce chapitre abordera principalement trois procédures d'analyse des données. Les deux premières, FACTOR et CLUSTER, font partie intégrante de SAS. La troisième, PROC ADDAD, donne accès à la bibliothèque du même nom.

VII.1

Analyse factorielle

Deux procédures d'analyse factorielle: PRINCOMP et FACTOR.

SAS propose deux procédures d'analyse factorielle. PRINCOMP réalise des analyses en composantes principales (ACP) très élémentaires et dépouillées, trop peut-être, puisque certains paramètres statistiques comme les corrélations entre les variables et les composantes principales n'y apparaissent même pas. On lui préférera donc, en général, la procédure FACTOR, plus complexe à utiliser, mais bien plus complète, avec ses batteries de rotation, ses nombreux graphiques, et surtout sa grande diversité de méthodes (Image, Harris, etc...). Notons cependant que, malgré cette grande richesse, SAS n'a pas encore intégré l'analyse factorielle des correspondances. Pour pratiquer l'analyse des données «à la française», il faudra donc avoir recours soit à un programme écrit en «Interactive Matrix Language» (IML), ce qui suppose une bonne pratique de l'algèbre matricielle, soit, sous le système d'exploitation OS/MVS IBM, à la procédure ADDAD (voir ci-dessous). Le présent exposé sera donc limité à la mise en œuvre de la procédure FACTOR.

VII.1.1

Analyse en composantes principales avec FACTOR

Pour réaliser une ACP sur l'ensemble des variables d'un tableau, il suffit d'appeler la procédure FACTOR de la manière suivante:

PROC FACTOR

```
PROC FACTOR DATA=nom du tableau;
```

Dans ce cas, on obtient tout d'abord une sortie standard composée des valeurs propres suivies des taux d'inertie et des corrélations entre les variables et les composantes principales («factor pattern»). Le nombre de facteurs retenus dépend du critère nommé MINEIGEN, qui ne prend en compte que les composantes présentant une valeur propre supérieure ou

égale à 1. Enfin, la sortie se termine par le tableau des communautés finales («final communality estimates»), permettant d'apprécier la contribution de chaque variable à l'inertie absorbée par l'ensemble des facteurs retenus.

Ces sorties limitées à l'essentiel peuvent être complétées par les options suivantes:

Tableau des scores factoriels standardisés.

— Un tableau SAS qui contient la totalité des variables analysées plus les coordonnées des observations sur les composantes principales retenues («factor scores»); ces nouvelles variables sont nommées FACTOR1, FACTOR", etc. Attention! Ces scores sont toujours centrés et réduits (standardisés). Pour les utiliser en entrée d'une autre procédure, il faudra, dans la majorité des cas, les multiplier par la racine carrée de la valeur propre correspondante.

Sept algorithmes de rotation des axes factoriels.

— Des rotations, au nombre de sept, permettent de simplifier la composition des facteurs retenus en leur faisant subir une optimisation selon un critère choisi. Ces rotations peuvent être orthogonales ou obliques. Leur utilisation requiert une bonne connaissance des principes mathématiques sur lesquels elles s'appuient. VARIMAX est sans doute l'une des mieux connues.

Graphiques des plans factoriels.

— Des graphiques représentant les corrélations entre les variables et les composantes principales. Ils permettent d'apprécier la position d'une variable vis-à-vis des autres et, ainsi, d'envisager les groupes d'association et d'opposition dans chaque plan factoriel. Pour représenter les observations dans chaque plan, il faut utiliser PROC PLOT, avec en entrée le tableau réalisé par FACTOR.

VII.1.2

ACP de l'urbanisation du Canton de Vaud

Sur le tableau VDSTAT, une analyse en composantes principales doit conduire à apprécier les principales composantes de l'urbanisation des communes du Canton de Vaud. Le programme PGMVII-1 ci-dessous est composé des instructions nécessaires à la réalisation de ce traitement.

EXEMPLE PGMVII-1
Analyse en composantes principales.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/
```

```
/*Transformation des densités en logarithmes*/
DATA VD; SET BASE.VDSTAT;
   LDEN80=LOG(DENS80);

/*ACP*/
PROC FACTOR DATA=VD OUT=VDFSC
   N=2   SCREE
   ROTATE=VARIMAX PLOT;
   VAR LDEN80 TPOP6070 TPOP7080
      PCTCON60 PCTCON70 PCTCON80
      PCTJEU80 PCTETR85;

/*Graphiques des communes dans le plan F1 et F2*/
DATA VDFSC; SET VDFSC;
      FACTOR1=FACTOR1*SQRT(2.98);
      FACTOR2=FACTOR2*SQRT(2.78);
   PROC PLOT DATA=VDFSC;
      PLOT FACTOR1*FACTOR2 /
      HAXIS=-4 TO 8 BY .5
      VAXIS=-3 TO 7 BY .5;
RUN;
```

Ce programme produit, en sortie, un grand nombre de pages composées de tableaux dont il faut connaître la signification. On obtient tout d'abord, pour chaque facteur, les valeurs propres (eigenvalues) dont le cumul représente l'inertie totale. On calcule donc un taux d'inertie qui représente en quelque sorte le niveau de généralité de chaque facteur. Ici (fig. VII.1), on observe qu'il existe un facteur très important, absorbant plus de 53% de l'inertie totale suivi de loin par le second facteur avec 18.2% et plusieurs autres facteurs moins importants. Le critère NFACTOR indique que seuls les deux premiers facteurs méritent, en première analyse, d'être retenus, ce que confirme le graphique des valeurs propres (SCREE PLOT) car il présente un net palier à partir du second facteur. Ainsi on n'améliore pas véritablement la compréhension du tableau de données en multipliant les facteurs à interpréter.

On ne ne recherche donc les corrélations entre les variables d'origine et les composantes principales que pour les deux premières d'entre elles (FACTOR PATTERN). On a choisi de pratiquer une rotation Varimax afin d'obtenir une structure factorielle plus aisément interprétable (Fig. VII.2). Ce sont les corrélations entre les variables et les composantes principales après rotation qui sont représentées sur la Figure VII.3.

On observe plusieurs regroupements de variables. Sur le pôle positif de la première composante, on note qu'un fort pourcentage de jeunes dans la population totale s'accompagne en général d'un accroissement de la population ainsi que de la construction (variables C, F, G). En opposition, on trouve la

part des immeubles construits avant 1960 (variable D). Lorsqu'on liste les coordonnées des communes sur le facteur 1 (Fig. VII.4), les coordonnées fortement positives qualifient les nouvelles zones périurbaines telles Chavannes-des-Bois, Le Vaud, souvent aux confins de l'agglomération genevoise et en milieu rural, alors que les valeurs négatives correspondent aux centres anciens.

Le deuxième facteur apporte à ce syndrome oppositionnel une importante nuance. Les communes qu'il qualifie, celles qui ont une forte valeur positive, correspondent aux banlieues densément peuplées et présentant une forte proportion de population étrangère (variables A et H). Sur le pôle négatif apparaissent les communes demeurées rurales.

Dans le cadran positif du plan factoriel, les variables E et B semblent faire la liaison entre le périurbain et l'urbain à proprement parler. Ces variables expriment l'évolution de la population et de la construction entre 1960 et 1970. On peut donc lire sur ce plan la succession des cycles de densification des périphéries des principaux centres urbains qui exercent leur polarisation sur l'espace vaudois.

On comprend que l'interprétation de tels résultats ne peut se satisfaire de la sommaire analyse présentée ici. Plus précisément, il est indispensable d'examiner les relations qu'entretiennent ces facteurs et donc de classer chaque commune dans un groupe plus ou moins homogène résultant de la partition du plan où chaque commune est localisée sur chacune des deux composantes (Fig. VII.4). A terme, la cartographie de ces groupes peut conduire à une interprétation géographique de ces phénomènes d'urbanisation. Ces coordonnées, en sortie de la procédure FACTOR, constituent le tableau d'entrée de la procédure de classification ascendante hiérarchique CLUSTER.

```
                                SAS       15:59 FRIDAY, MARCH 10, 1989 13

INITIAL FACTOR METHOD: PRINCIPAL COMPONENTS

                PRIOR COMMUNALITY ESTIMATES: ONE

                  EIGENVALUES OF THE CORRELATION MATRIX
                 TOTAL =         8   AVERAGE =          1

                              1         2         3         4
          EIGENVALUE    4.308715  1.453979  0.781227  0.630144
          DIFFERENCE    2.854736  0.672752  0.151084  0.237443
          PROPORTION      0.5386    0.1817    0.0977    0.0788
          CUMULATIVE      0.5386    0.7203    0.8180    0.8968

                              5         6         7         8
          EIGENVALUE    0.392701  0.244675  0.188560  0.000000
          DIFFERENCE    0.148026  0.056115  0.188560
          PROPORTION      0.0491    0.0306    0.0236    0.0000
          CUMULATIVE      0.9458    0.9764    1.0000    1.0000

             2 FACTORS WILL BE RETAINED BY THE NFACTOR CRITERION

       INITIAL FACTOR METHOD: PRINCIPAL COMPONENTS

                                SCREE PLOT OF EIGENVALUES


            E        5 +
            I          |     1
            G          |
            E          |
            N          |           2    3    4
            V        0 +                          5    6    7    8
            A          +---------+---------+---------+---------+
            L          0         2         4         6         8
            U
            E          NUMBER

       INITIAL FACTOR METHOD: PRINCIPAL COMPONENTS

                                FACTOR PATTERN

                                   FACTOR1   FACTOR2

                          LDEN80     0.62764   0.60260
                          TPOP6070   0.76686   0.30760
                          TPOP7080   0.75272  -0.47169
                          PCTCON60  -0.93422   0.19080
                          PCTCON70   0.75074   0.06650
                          PCTCON80   0.85929  -0.31330
                          PCTJEU80   0.43412  -0.53618
                          PCTETR85   0.63000   0.58930

                        VARIANCE EXPLAINED BY EACH FACTOR

                              FACTOR1   FACTOR2
                              4.308715  1.453979
```

Figure VII.1

```
INITIAL FACTOR METHOD: PRINCIPAL COMPONENTS

                FINAL COMMUNALITY ESTIMATES: TOTAL =   5.762694

      LDEN80   TPOP6070   TPOP7080   PCTCON60
    0.757060   0.682691   0.789073   0.909163

    PCTCON70   PCTCON80   PCTJEU80   PCTETR85
    0.568036   0.836540   0.475955   0.744176


ROTATION METHOD: VARIMAX

                        ORTHOGONAL TRANSFORMATION MATRIX

                                1          2

                    1       0.73131    0.68205
                    2      -0.68205    0.73131

                          ROTATED FACTOR PATTERN

                                FACTOR1    FACTOR2

                LDEN80          0.04799    0.86877
                TPOP6070        0.35101    0.74798
                TPOP7080        0.87218    0.16844
                PCTCON60       -0.81333   -0.49765
                PCTCON70        0.50367    0.56067
                PCTCON80        0.84209    0.35696
                PCTJEU80        0.68318   -0.09602
                PCTETR85        0.05880    0.86065


ROTATION METHOD: VARIMAX

                        VARIANCE EXPLAINED BY EACH FACTOR

                              FACTOR1    FACTOR2
                              2.980716   2.781978

                FINAL COMMUNALITY ESTIMATES: TOTAL =   5.762694

      LDEN80   TPOP6070   TPOP7080   PCTCON60
    0.757060   0.682691   0.789073   0.909163

    PCTCON70   PCTCON80   PCTJEU80   PCTETR85
    0.568036   0.836540   0.475955   0.744176

                SCORING COEFFICIENTS ESTIMATED BY REGRESSION

        SQUARED MULTIPLE CORRELATIONS OF THE VARIABLES WITH EACH FACTOR

                              FACTOR1    FACTOR2
                              1.000000   1.000000
```

Figure VII.2

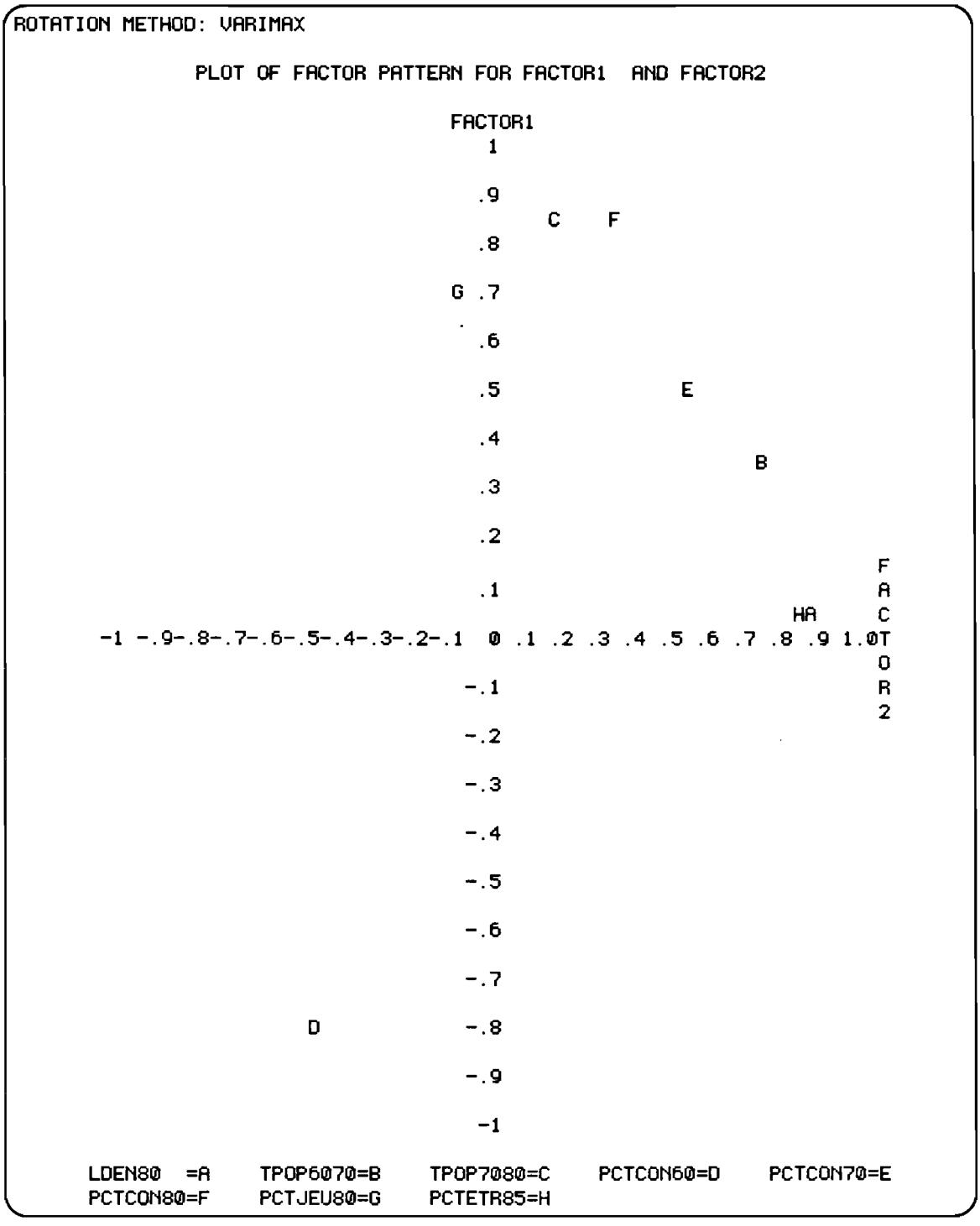
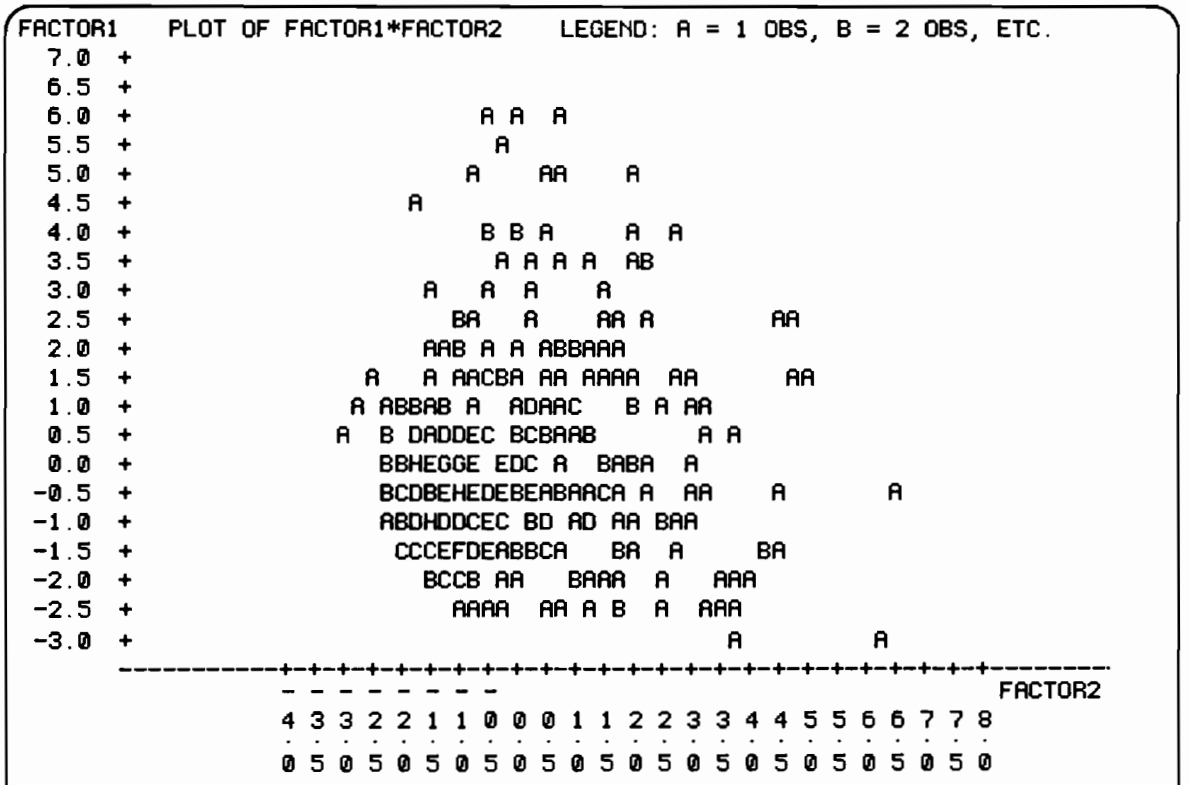

ROTATION METHOD: VARIMAX
PLOT OF FACTOR PATTERN FOR FACTOR1 AND FACTOR2
FACTOR1
FACTOR2
LDEN80 =A TPOP6070=B TPOP7080=C PCTCON60=D PCTCON70=E
PCTCON80=F PCTJEU80=G PCTETR85=H

Figure VII.3



OBS	NUMERO	NOM	FACTOR1
1	5890	VEVEY	-3.8477
2	5674	LOVATENS	-3.6234
3	5586	LAUSANNE	-3.3541
4	5589	PRILLY	-3.1798
5	5567	ROMAIRON	-3.1759
6	5707	CHAVANNES-DE-BOGIS	5.5792
7	5718	GENOLIER	5.9105
8	5909	CHESEAUX-NOREAZ	6.0438
9	5731	LE-VAUD	6.1433
10	5708	CHAVANNES-DES-BOIS	7.4190

OBS	NUMERO	NOM	FACTOR2
1	5750	LES CULLAYES	-2.9423
2	5667	CORREVON	-2.6718
3	5937	VUGELLES-LA-MOTHE	-2.4515
4	5760	RANCES	-2.3516
5	5562	MAUBORGET	-2.3495
6	5643	PREVERENGES	4.8868
7	5890	VEVEY	4.9293
8	5589	PRILLY	5.0375
9	5591	RENENS	6.1477
10	5635	ECUBLENS	6.6102

Figure VII.4

VII.2

Classification automatique

Deux procédures de classification automatique: FASTCLUS et CLUSTER.

Deux procédures de classification automatique confèrent à SAS une grande performance dans ce domaine. FASTCLUS réalise des classifications non hiérarchiques sur de grands ensembles d'observations (en général, plus de 1000). Cette procédure est directement inspirée des travaux de MacQueens sur l'algorithme des k-moyennes. Dans le domaine des classifications ascendantes hiérarchiques, CLUSTER constitue une imposante procédure avec ses 11 critères différents d'agrégation. Examinons ici un cas de figure classique, celui dans lequel les classes sont constituées de manière à être les plus homogènes possibles, avec une variance intra-classe minimale et une variance inter-classe maximale: il s'agit ici du critère de Ward.

VII.2.1

Classification ascendante hiérarchique selon le critère de Ward

La réalisation d'une telle classification se fait très simplement, avec une seule instruction:

PROC CLUSTER

```
PROC CLUSTER DATA=nom du tableau
  METHOD=WARD;
```

Dans ce cas, on obtient une sortie qui décrit le déroulement de la formation des classes, de la première, celle qui ne rassemble que les deux observations les plus proches, jusqu'à la dernière, la plus hétérogène, qui comprend la totalité du tableau traité. A chacun de ces niveaux est associé un indice nommé R-SQUARED qui n'est en fait qu'un taux d'inertie absorbé par les classes. En classification, on cherche le plus petit nombre de classes possible pour le plus fort taux d'inertie. L'analogie avec l'analyse factorielle est donc évidente: en acceptant de perdre une petite partie de l'inertie totale, on peut réduire le nombre d'observations à un petit nombre de classes.

Il faut bien comprendre que la procédure CLUSTER ne fait qu'élaborer une hiérarchie de classes emboîtées; pour définir une partition, c'est-à-dire couper l'arbre de la hiérarchie

à un niveau jugé significatif, il faut enchaîner les procédures CLUSTER et TREE de la manière suivante:

PROC TREE

```
PROC CLUSTER DATA=nom du tableau (en entrée)
  OUTTREE=nom du tableau de la hiérarchie
     (en sortie)
  METHOD=WARD;
PROC TREE DATA=nom du tableau de la hiérarchie
     (en entrée)
  OUT=nom du tableau de la partition (en sortie)
  HEIGHT=RSQ  LEVEL=taux d'inertie choisi;
```

La procédure TREE assure principalement deux fonctions. D'une part, elle trace l'arbre de la hiérarchie, ce qui permet d'apprécier les principales discontinuités du processus de hiérarchisation et, ainsi, d'apprécier les coupures significatives et donc le nombre de classes à retenir. Lorsque le nombre d'observations dépasse la centaine, cette figure devient très difficile à lire et il est préférable de ne pas l'imprimer, en ajoutant l'option NOPRINT. La seconde fonction assurée par TREE revient à couper l'arbre au niveau choisi avec le paramètre LEVEL; dans le tableau de sortie, chaque observation sera assortie d'un numéro de classe dans une nouvelle variable nommée CLUSTER.

VII.2.2

CAH d'après les composantes de l'urbanisation du Canton de Vaud

Sur les deux composantes principales issues de la procédure CLUSTER, on peut classer les communes du Canton de Vaud. Le programme PGMVII-2 ci-dessous est composé des instructions nécessaires à la réalisation de ce traitement.

EXEMPLE PGMVII-2
Classification ascendante hiérarchique.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Transformation des densités en logarithmes*/
DATA VD; SET BASE.VDSTAT;
  LDEN80=LOG(DENS80);

/*ACP*/
PROC FACTOR DATA=VD OUT=VDFSC
  SCREE N=2
  ROTATE=VARIMAX PLOT;
  VAR LDEN80 TPOP6070 TPOP7080
```

```
      PCTCON60 PCTCON70 PCTCON80
      PCTJEU80 PCTETR85;

/*Classification ascendante hiérarchique*/
DATA VDFSC; SET VDFSC;
      FACTOR1=FACTOR1*SQRT(2.98);
      FACTOR2=FACTOR2*SQRT(2.78);
   PROC CLUSTER DATA=VDFSC
      OUTTREE=TREEFSC METHOD=WARD;
      VAR FACTOR1 FACTOR2;
      ID NUMERO;

/*Recherche d'une partition*/
   PROC TREE DATA=TREEFSC
      OUT=CLASFSC HEIGHT=RSQ  LEVEL=0.7
         NOPRINT;
      COPY FACTOR1 FACTOR2;
      ID NUMERO;

/*Tableau du nombre de communes par classes*/
   PROC FREQ DATA=CLASFSC;
      TABLES CLUSTER;

/*Calcul des profils moyens des classes sur les variables
d'origine*/
   PROC SORT  DATA=VD  OUT=VD2;
      BY NUMERO;
   PROC SORT  DATA=CLASFSC  OUT=CLASFSC2;
      BY NUMERO;
   DATA CLASFSC MERGE VD2 CLASFSC2;
      BY NUMERO;

   PROC SORT  DATA=CLASFSC  OUT=CLASF;
      BY CLUSTER;

   PROC MEANS  DATA=CLASF
      MAXDEC=2 MIN MAX MEAN STD N;
      VAR DENS80 TPOP6070 TPOP7080
         PCTCON60 PCTCON70 PCTCON80
         PCTJEU80 PCTETR85 FACTOR1 FACTOR2;

   PROC MEANS  DATA=CLASF
      MAXDEC=2 MIN MAX MEAN STD N;
      VAR DENS80 TPOP6070 TPOP7080
         PCTCON60 PCTCON70 PCTCON80
         PCTJEU80 PCTETR85 FACTOR1 FACTOR2;
      BY CLUSTER;
RUN;
```

Les figures VII.5 et VII.6 illustrent une partie des résultats globaux, en particulier pour 5 groupes expliquant 71,1%.

SAS 17:23 FRIDAY, MARCH 10, 1989 35

WARD'S MINIMUM VARIANCE CLUSTER ANALYSIS

NUMBER OF CLUSTERS	CLUSTERS JOINED		FREQUENCY OF NEW CLUSTER	SEMIPARTIAL R-SQUARED	R-SQUARED
10	CL15	CL33	103	0.011917	0.867424
9	CL12	CL29	66	0.013942	0.853482
8	CL26	CL17	24	0.015264	0.838219
7	CL11	CL32	71	0.020578	0.817640
6	CL10	CL14	189	0.045397	0.772243
5	CL9	CL16	82	0.060787	0.711456
4	CL8	CL13	43	0.061883	0.649573
3	CL7	CL4	114	0.086543	0.563030
2	CL5	CL3	196	0.257140	0.305890
1	CL2	CL6	385	0.305890	0.000000

CLUSTER	FREQUENCY	PERCENT	CUMULATIVE FREQUENCY	CUMULATIVE PERCENT
1	189	49.1	189	49.1
2	82	21.3	271	70.4
3	71	18.4	342	88.8
4	19	4.9	361	93.8
5	24	6.2	385	100.0

VARIABLE	MINIMUM VALUE	MAXIMUM VALUE	MEAN	STANDARD DEVIATION	N
DENS80	0.07	71.41	2.37	6.80	385
TPOP6070	-37.10	221.59	8.45	33.78	385
TPOP7080	-36.59	206.85	14.72	31.26	385
PCTCON60	21.72	100.00	73.28	17.97	385
PCTCON70	0.00	37.87	11.80	7.91	385
PCTCON80	0.00	61.76	14.92	12.62	385
PCTJEU80	5.56	30.88	19.50	3.75	385
PCTETR85	0.00	37.77	8.62	7.49	385
FACTOR1	-3.85	7.42	-0.00	1.73	385
FACTOR2	-2.94	6.61	0.00	1.67	385
---------- CLUSTER=1 ----------					
DENS80	0.07	1.20	0.45	0.19	189
TPOP6070	-37.10	21.88	-9.77	9.61	189
TPOP7080	-36.59	42.25	2.31	12.79	189
PCTCON60	56.60	100.00	85.73	8.44	189
PCTCON70	0.00	32.08	7.01	5.60	189
PCTCON80	0.00	28.57	7.25	5.53	189
PCTJEU80	5.56	30.88	19.11	4.02	189
PCTETR85	0.00	13.94	3.43	2.79	189
FACTOR1	-3.62	1.41	-0.59	0.90	189
FACTOR2	-2.94	-0.03	-1.21	0.55	189

Figure VII.5

Classification ascendante hiérarchique et profils des variables

VARIABLE	MINIMUM VALUE	MAXIMUM VALUE	MEAN	STANDARD DEVIATION	N
		CLUSTER=2			
DENS80	0.43	71.41	7.43	13.28	82
TPOP6070	-21.77	184.78	20.44	27.19	82
TPOP7080	-33.33	65.75	1.45	14.66	82
PCTCON60	52.55	94.19	74.47	8.98	82
PCTCON70	0.00	28.40	13.54	5.29	82
PCTCON80	1.89	30.61	11.99	5.63	82
PCTJEU80	5.80	21.73	17.48	2.43	82
PCTETR85	2.43	37.77	16.59	7.41	82
FACTOR1	-3.85	0.14	-1.30	0.88	82
FACTOR2	0.06	6.61	1.89	1.50	82
		CLUSTER=3			
DENS80	0.16	3.70	1.06	0.78	71
TPOP6070	-32.29	65.63	10.69	19.38	71
TPOP7080	-19.03	65.94	23.52	18.54	71
PCTCON60	38.58	78.46	59.96	9.38	71
PCTCON70	1.45	35.28	16.36	7.29	71
PCTCON80	5.88	44.88	23.68	8.72	71
PCTJEU80	12.50	29.08	20.50	2.96	71
PCTETR85	0.93	26.87	9.51	5.08	71
FACTOR1	-0.53	2.84	1.05	0.86	71
FACTOR2	-1.56	1.94	0.14	0.82	71
		CLUSTER=4			
DENS80	0.17	4.62	1.40	0.95	19
TPOP6070	-15.94	112.00	30.69	30.01	19
TPOP7080	72.60	206.85	110.89	37.54	19
PCTCON60	21.72	44.51	34.01	7.23	19
PCTCON70	11.18	33.61	20.51	5.90	19
PCTCON80	31.87	61.76	45.48	8.22	19
PCTJEU80	19.95	26.69	24.14	1.74	19
PCTETR85	1.75	21.57	10.82	5.19	19
FACTOR1	3.24	7.42	4.72	1.12	19
FACTOR2	-1.73	1.97	0.02	0.76	19
		CLUSTER=5			
DENS80	1.44	16.46	4.87	3.65	24
TPOP6070	-2.34	221.59	86.76	61.45	24
TPOP7080	7.25	104.08	55.65	23.59	24
PCTCON60	24.06	55.30	41.66	8.52	24
PCTCON70	14.44	37.87	23.13	6.49	24
PCTCON80	23.91	54.92	35.21	7.97	24
PCTJEU80	18.09	26.61	22.86	2.07	24
PCTETR85	7.28	32.59	17.84	6.11	24
FACTOR1	0.66	4.03	2.28	1.00	24
FACTOR2	1.26	4.89	2.66	1.15	24

Figure VII.6

Profils moyens des classes sur les variables d'origine

La première partie du listing comprend l'historique de la hiérarchisation des communes en classes. On s'est contenté ici de présenter les étapes finales de ce processus. On apprécie de cette manière le nombre de classes qu'il est nécessaire de retenir pour rendre compte de la plus grande partie possible de variance sans augmenter indéfiniment le nombre de classes. Par exemple, si l'on ne retenait qu'une seule classe, on aurait 0% de variance expliqué, c'est-à-dire que toutes les communes seraient considérées comme semblables, ce qui est absurde. Avec deux classes, on n'exprime que 30.6% de la variance totale, ce qui reste encore insuffisant pour apprécier non seulement les ressemblances, mais aussi les différences. Dans le cas présent, le taux de variance expliquée s'accroît très vite puisqu'avec dix classes on atteint la valeur de 86,8%. Le nombre de classes à retenir est donc compris entre deux et dix classes. Une observation plus fine de la croissance des taux de variance, en fonction du nombre de classes, montre qu'à partir de cinq classes le gain devient de plus en plus modeste. C'est la raison pour laquelle on a préféré retenir cinq classes.

Bien entendu, le nombre de communes par classes n'est pas égal. La procédure FREQ appliquée à la variable CLUSTER, c'est-à-dire au numéro de classe issu de la procédure CLUSTER, donne un tableau de fréquence par classe. La classe N°1 apparaît la plus nombreuse avec près de la moitié des communes du canton de Vaud (189). Les classes N°2 et N°3 représentent chacune 20% de l'effectif total, alors que les classes N°4 et N°5 sont peu représentées, respectivement 4,9% et 6,2% de l'ensemble des communes.

L'étape qui suit le choix du nombre de classes conduit à l'interprétation des classes par appréciation de leurs profils sur les variables de l'analyse. On cherche à évaluer les différences qui existent entre une classe donnée et l'ensemble complet des communes ainsi qu'entre une classe donnée et toutes les autres classes.

La procédure MEANS permet non seulement de calculer le profil moyen de l'ensemble des communes du canton, mais aussi celui de chaque classe en utilisant l'option BY CLUSTER. Le tableau obtenu se compose, mis à part le nom de chaque variable, de ses valeurs minimale et maximale, de sa moyenne, de son écart-type et de son effectif. Par exemple, les classes N°4 et N°5 sont celles qui regroupent les communes les plus anciennement urbanisées (classe N°5) et celles qui ont connu, durant la période 1970-1980, la plus forte croissance urbaine. Entre ces deux classes, on observe une croissance plus forte, en 1960-1970 pour la classe N°5, qui ralentit par la suite; corrélativement, la classe N°4 connaît une croissance extrêmement forte (+110% en moyenne) durant la période 1970-1980, mais il s'agit en général de communes plus éloignées des centres-villes (la densité y est plus faible).

VII.3

Un complément utile: la bibliothèque ADDAD

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

```
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 ou *AFC*
—ANCOMP analyse en composantes principales ou *ACP*
—CAH2CO classification ascendante hiérarchique ou *CAH*
—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 deux exemples d'utilisation de ces instructions pour une analyse factorielle des correspondances et pour une classification ascendante hiérarchique.

VII.3.1

Un exemple d'analyse factorielle des correspondances (bibliothèque ADDAD)

EXEMPLE PGMVII-3
AFC avec l'ADDAD

```
PROC ADDAD
  MEMBER1=ANCORR DATA=BASE.VDCONST
  OUT1=FAC NOMISS;
VAR CONS60 CONS70 CONS80;
IDEN NUMERO;
  PARAM IMPFI IMPFJ NF=2;
    GRAPHE X=_F1 Y=_F2 IP;
    GRAPHE X=_F1 Y=_F2 JP;
    GRAPHE X=_F1 Y=_F2 IP JP;
  TITRE AFC IMMEUBLES;
RUN;
```

Ici, on cherche à présenter la répartition des immeubles selon leur période de construction. Les variables CONS60 à CONS80 font l'objet d'une analyse factorielle des correspondances. Les coordonnées des communes sur les facteurs sont rangées dans le tableau temporaire FAC qui aura NF facteurs nommés _F1, et _F2, et une variable identifiant, copie de la variable NUMERO, nommée _NOM. Les instructions GRAPHE commandent les sorties successives du premier plan factoriel avec, d'une part, les observations principales (IP, il peut y avoir des observations

supplémentaires projetées sur ce plan), d'autre part, des variables principales (JP, il peut y avoir des variables supplémentaires), et, enfin, à la fois des observations et des variables.

VII.3.2

Un exemple de classification ascendante hiérarchique (bibliothèque ADDAD)

EXEMPLE PGMVII-4
CAH avec l'ADDAD

```
PROC ADDAD
   MEMBER1=CAH2CO DATA=BASE.VDCONST
   OUT=CAH NOMISS;
VAR CONS60 CONS70 CONS80;
IDEN NUMERO;
PARAM IOPT=2 HISTO DESCRI ARBRE;
TITRE CAH IMMEUBLES;
RUN;
```

On cherche ici à classer les communes du canton de Vaud selon la ou les périodes de construction de leurs immeubles. Les variables CONS60 à CONS80 font l'objet d'une classification ascendante hiérarchique. La hiérarchie est stockée dans le tableau temporaire CAH. Les paramètres de l'analyse précisent la métrique retenue pour mesurer les distances entre observations (ici la métrique du khi-2) et le choix des sorties (histogrammes des indices de niveaux, description de la hiérarchie, arbre de classification).

VII.3.3

Recherche de partitions

La procédure CAH2CO ne produit pas directement une partition, elle n'affecte pas un numéro de classe et un seul à chaque unité spatiale. CAH2CO se contente d'échafauder une hiérarchie. La recherche de partitions se fait après coup, avec la procédure CLACAH. A partir d'une hiérarchie, on peut définr plusieurs partitions en fonction du niveau auquel on coupe l'arbre de la hiérarchie. Les niveaux pertinents peuvent être repérés sur l'histogramme des indices de niveau.

EXEMPLE PGMVII-5
Recherche de partition d'une CAH avec l'ADDAD.

```
PROC ADDAD
   MEMBER1=CLACAH DATA=BASE.INDIC
   DATACAH=CAH OUT=CAHPART NOMISS;
IDEN NUMERO;
PARAM NP=3;
PARMCARDS;
   2 4 5;
TITRE CAH IMMEUBLES;
RUN;
```

Le nombre de partitions est affecté au paramètre NP (ici, 3 partitions) et le nombre de classes pour chaque partition figure après PARMCARDS; Dans le tableau CAHPART, mis à part l'identifiant _NOM, on récupérera 3 nouvelles variables, nommées _P1, _P2 et _P3. Notons enfin qu'il est nécessaire de faire précéder CLACAH de la procédure CAH2CO, car il faut donner à CLACAH les noms des tableaux contenant les données d'origine (VDCONST) et la description de la hiérarchie (CAH).

VII.3.4

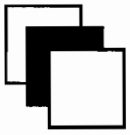
L'ADDAD

Pour en savoir plus sur la bibliothèque de l'ADDAD, adressez-vous au siège de l'association, à l'adresse suivante:

Références de l'ADDAD.

Association pour le développement et la diffusion de l'analyse des données
(ADDAD).
22, rue Charcot
75013 Paris.

☎ 45.85.40.23



VIII

Le langage MACRO

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'achevant par un point virgule. Il existe un autre niveau de langage appelé «macro language» ou langage macro en français.

VIII.1

Structure d'une macro

Comme son nom l'indique, le langage macro se situe au-dessus du langage SAS de base. Il a pour fonction principale l'assemblage d'étapes SAS programmées en vue d'une application particulière et reproductible. Ces programmes, nommés «macros SAS», sont paramétrables et peuvent constituer la boîte à outils propre à chaque utilisateur.

Chaque macro est composée d'une ou plusieurs étapes DATA ou PROC paramétrées et commençant avec l'instruction:

Au début: %MACRO

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

et s'achevant par l'instruction:

A la fin: %MEND

%MEND nom de la macro;

Les paramètres sont des noms SAS; ils doivent être séparés par des virgules. On les appelle aussi «macro-va-

riables». 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'on y fera référence dans les instructions SAS, ce nom sera précédé du caractère «&» («et» commercial), &TAB, par exemple:

La référence &

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

L'appel de cette macro se fait par son nom, et lui fait immédiatement suite. Il faut lui adresser le paramètre qui lui est nécessaire. Ceci peut se faire avec 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(VDSTAT);
%TABLEAU(VDGEO);
```

pour obtenir l'affichage du contenu et les valeurs des variables des tableaux permanents VDSTAT et VDGEO.

VIII.2

Les instructions du langage macro

Il existe un grand nombre de macro 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:

Créer une macro variable

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.

Modifier inconditionnellement le déroulement d'une macro

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;

Modifier conditionnellement le déroulement d'une macro.

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

%IF condition %THEN action;

Par exemple:

%IF &PARAM eq . %THEN %GOTO %FIN;

cette partie de la macro est ignorée si
le paramètre PARAM n'a pas de valeur

%FIN;

cette partie de la macro est exécutée

%MEND nom de la macro;

Le programme ci-dessous 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 NOMVAR de la macro) d'un tableau (son nom est donné par le paramètre NOMTAB), en fonction de son type (défini par le paramètre TYPE) qui peut être CONT s'il s'agit d'une variable continue ou DISC pour une variable discrète. Si la variable est discrète (comme c'est le cas de la variable RUMLEY du tableau VDGEO), la procédure FREQ comptera les modalités; dans le cas d'un variable continue (comme c'est le cas de la variable DENS80 du tableau INDIC), la procédure UNIVARIATE en donnera une étude statistique complète. Dans tous les cas, la procédure PRINT affichera le tableau choisi.

EXEMPLE PGMVIII-1
Définition et appel d'une macro.

La définition de la MACRO ne donne pas de résultats.

L'appel de la MACRO provoque l'exécution des instructions figurant dans la définition et la sortie des résultats des procédures.

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Définition de la macro*/
%MACRO ETUDSTAT(NOMTAB,NOMVAR,TYPE);
  %LET TITRE='VARIABLE: ';
       TITLE &TITRE&NOMVAR;
    PROC PRINT DATA=BASE.&NOMTAB;
      VAR NUMERO &NOMVAR;

  %IF &TYPE EQ DISC %THEN %GOTO DISC;
  %IF 1TYPE NE CONT %THEN %GOTO FIN;

    PROC UNIVARIATE DATA=BASE.&NOMTAB;
      VAR &NOMVAR;
  %GOTO FIN;
  %DISC: ;

    PROC FREQ DATA=BASE.&NOMTAB;
      TABLES &NOMVAR;
  %FIN: ;
%MEND ETUDSTAT;

/*Appel de la macro*/
%ETUDSTAT(VDGEO,RUMLEY,DISC);
%ETDUSTAT(VDSTAT,DENS80,CONT);

RUN;
```

Note à propos du 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. Par ailleurs, SAS Institute propose désormais l'«Interactive Matrix Language» qui remplace la procédure MATRIX, avec de nombreuses possibilités supplémentaires et une interface utilisateur particulière. Il s'agit d'un langage de programmation très complet permettant l'écriture directe d'expressions algébriques matricielles. Il est ainsi possible de programmer assez facilement des applications particulières, pour peu qu'on dispose 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.

IX

SAS/ GRAPH et l'environnement graphique

Le module 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.

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.

IX.1

Les unités graphiques

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.

Les numériseurs ou digitaliseurs

A. Les numériseurs (ou digitaliseurs) assurent 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 controle 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.

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.

Les écrans graphiques

B. 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. 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.

La couleur coûte relativement cher, 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 (soustractive, sur écran) du rouge, du vert et du bleu, ou bien (additive, sur papier) du jaune, du cyan et du magenta. Dans les écrans bas 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.

Les traceurs à plumes

C. 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.

Les traceurs électrostatiques

D. Les traceurs électrostatiques 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 où 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 des traceurs électrostatiques s'améliore constamment; la dernière innovation dans ce domaine a été l'introduction sur la marché d'un traceur électrostatique couleur par la société VERSATEC.

Les traceurs à jet d'encre

E. Les traceurs à jet d'encre 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.

IX.2

L'utilisation des unités graphiques avec SAS/GRAPH

Pour utiliser SAS/GRAPH avec une unité graphique donnée, il est nécessaire de disposer de son pilote (DRIVER). Celui-ci permet de transformer les images, indépendantes de l'unité d'affichage ou de tracé, produites par SAS, en images affichables ou imprimables sur cette unité. La liste des pilotes est donnée dans le manuel de référence SAS/GRAPH. Notons qu'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; cela requiert de bonnes connaissances en informatique graphique, et doit donc être réalisé par un véritable spécialiste.

«Universal Driver» de SAS

IX.2.1

Les options graphiques

Préalablement à l'exécution des procédures de SAS/GRAPH, il faut définir l'environnement graphique de la session SAS en cours, si cela n'a pas déjà été fait. Ceci est réalisé très simplement à l'aide de l'instruction GOPTIONS. Elle a pour syntaxe:

GOPTIONS

```
GOPTIONS
   DEVICE=nom de l'unité graphique BORDER;
```

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;
```

IX.2.2

Utilisation élémentaire des procédures GPLOT et GCHART

Ces deux procédures assurent des traitements semblables à PLOT et CHART du SAS de base. Elle offrent cependant un bien plus grand nombre d'options et la qualité graphique du résultat est, bien entendu, nettement supérieure. Voici un petit programme SAS qui permettra de familiariser le lecteur avec l'utilisation des procédures graphiques. Il réalise l'étude statistique de deux variables du tableau VDSTAT, la densité de population et la distance à Lausanne, avec le tracé des histogrammes et trace la droite de régression dans le nuage de points des communes sur ces deux variables.

EXEMPLE PGMIX-1
Graphique bivarié

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Mettre en place l'environnement graphique*/
GOPTIONS DEVICE=IBM3279 BORDER;

/*Etude de la variable DENS85*/
TITLE 'DENSITE DE POPULATION 1980';
PROC UNIVARIATE DATA=BASE.VDSTAT PLOT;
   VAR DENS80;

TITLE C=RED H=10 PCT F=TITALIC
      'DENSITE DE POPULATION 1980';
PROC GCHART DATA=BASE.VDSTAT;
   HBAR DENS80 / LEVELS=10;

/*Etude de la variable DISTLAUS*/
TITLE 'DISTANCE A LAUSANNE';
PROC UNIVARIATE DATA=BASE.VDSTAT PLOT;
   VAR DISTLAUS;

TITLE C=RED H=10 PCT F=TITALIC
      'DISTANCE A LAUSANNE';
PROC GCHART DATA=BASE.VDSTAT;
   HBAR DISTLAUS / LEVELS=10;

/*Etude de la relation densité-distance à Lausanne*/
TITLE 'DENSITE/DISTANCE A LAUSANNE';
PROC REG DATA=BASE.VDSTAT;
   MODEL DENS80=DISTLAUS;
```

```
   OUTPUT OUT=RESIDUS R=RESID;

TITLE C=RED H=10 PCT F=TITALIC
      'DENSITE/DISTANCE A LAUSANNE';
PROC GPLOT DATA=BASE.VDSTAT;
   PLOT DENS80*DISTLAUS;
      SYMBOL='.' I=RL;

TITLE C=RED H=10 PCT F=TITALIC
      'DENSITE/RESIDUS';
PROC GPLOT DATA=RESIDUS;
   PLOT DENS80*RESID;

RUN;
```

IX.2.3

Les textes des graphiques

Dans l'exemple précédent, nous avons habillé les graphiques par des textes identifiant les traitements. Deux instructions facilitent ce travail: TITLE pour les textes figurant au-dessus du tracé, FOOTNOTE pour ceux figurant au-dessous. La syntaxe est du type:

TITLE

```
TITLE C=couleur F=police de caractères
      H=hauteur PCT 'titre';
```

ou bien:

FOOTNOTE

```
FOOTNOTE C=couleur F=police de caractères
      H=hauteur PCT 'note';
```

S'il doit y avoir plusieurs titres ou plusieurs notes (1 à 10 au maximum), TITLE 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, GREEN, 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 apostrophes. Par exemple:

```
TITLE2 C=RED H=10 PCT F=TITALIC
    'DENSITE DE POPULATION 1980';
```

affichera sur la seconde ligne le titre 'DENSITE DE POPULATION 1980', 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).

X

Cartographie automatique avec SAS/ GRAPH

L'un des apports importants de SAS/GRAPH, par comparaison à d'autres logiciels d'analyse statistique, consiste en l'environnement cartographique complet qu'il propose. Ainsi, la réalisation de cartes statistiques ne se heurte plus à des difficultés techniques, parfois difficiles à surmonter voici quelques années seulement.

X.1

Les procédures de cartographie

Quatre procédures de SAS/GRAPH ont une orientation cartographique très marquée.

Projections cartographiques: GPROJECT

GPROJECT réalise plusieurs calculs de projection sur un tableau contenant des coordonnées angulaires sur la sphère; ces coordonnées doivent être exprimées en radians.

Généralisation des contours: GREDUCE

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.

Agrégation de polygones: GREMOVE

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 du fond de carte, d'une variable d'agrégation.

Cartes choroplèthes: GMAP

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, les résultats sont aussi cartographiques.

Interpolation en isolignes: GCONTOUR

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.

Surfaces tridimensionelles: G3D

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 utiles à la cartographie. Notons également la procédure GREPLAY qui permet d'enregistrer les dessins pour les réafficher ultérieurement, à la demande, et de les monter en planches avec d'autres.

Un examen approfondi de l'ensemble de ces procédures nécessiterait un très long exposé; seules GREDUCE, GREMOVE, GMAP et G3D seront étudiées ici.

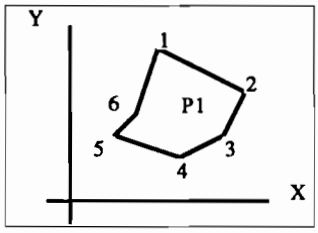
X.2

Les fonds de carte SAS

SAS GRAPH ne contient pas de procédure d'enregistrement, de numérisation des fonds de carte. La solution la plus simple pour réaliser ce type de travail est d'utiliser un micro-ordinateur connecté à un numériseur. On trouve dans le commerce de bons logiciels de numérisation. On peut aussi utiliser le logiciel extrêmement simple et élémentaire publié par la Maison de la Géographie de Montpellier. Si vous disposez d'un Macintosh, la numérisation avec le logiciel CARTOGRAPHIE-2D est très simple. Des programmes de conversion sont disponibles; ils permettent de créer le fichier de numérisation dans un format compatible

avec les procédures de cartographie de SAS/GRAPH.

Enregistrement ponctuel des polygones:



Identifiant	Coord. X	Coord. Y
P1	180	200
P1	240	160
P1	230	135
P1	190	118
P1	135	140
P1	170	155

Le dénominateur commun entre les diverses organisations de fichiers de fonds de cartes est l'enregistrement des unités spatiales sous forme de polygones. Chaque polygone est décrit par une séquence d'enregistrements égal à celui de ses sommets. Chaque enregistrement commence par l'identifiant du polygone, suivi des coordonnées, d'abord sur l'axe des abscisses (X), puis sur l'axe des ordonnées (Y). Cette organisation extrêmement élémentaire provient habituellement de la saisie avec un numériseur (Cabos, Waniez, 1987). Mais un tel fichier peut, dans le cas d'un fond composé de quelques zones seulement (les 22 régions françaises, par exemple), être également saisi à l'aide d'un éditeur de texte comme, par exemple, WORD. Dans ce cas, il faut procéder de la manière suivante : dessiner le fond de carte sur du papier millimétré et coter chaque angle par ses coordonnées mesurées sur le papier millimétré; pour chaque unité spatiale, saisir les enregistrements qui la composent (autant d'enregistrements qu'il y a de côtés): identifiant, un espace, abscisse (X), un espace, ordonnée (Y), retour chariot. Notons qu'il est inutile de reproduire à la fin le premier enregistrement pour fermer le polygone.

Traitement des enclaves

Lorsque qu'une unité spatiale comprend plusieurs polygones, comme dans le cas d'enclaves ou bien encore d'archipels appartenant à un même pays, une variable supplémentaire se nommant obligatoirement SEGMENT doit figurer dans chaque enregistrement du fond de carte. Elle contient le numéro du polygone auquel appartient le point enregistré. Ce numéro doit être compris entre 1 et n, n étant le nombre total de polygones.

X.3

Les cartes choroplèthes

Les cartes choroplèthes 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 de teinte, il faut

choisir des trames de densités assez différentes et en nombre limité. Cela conduit à transformer l'échelle de rapport en échelle d'intervalle. Chaque intervalle est ainsi représenté par la trame qui lui correspond. C'est la procédure GMAP qui réalise le tracé des cartes choroplèthes.

X.3.1

La discrétisation des variables

Il existe trois techniques usuelles pour rendre «discrètes» des variables continues; à chacune d'elle correspond une procédure SAS. Dans tous les cas, il est nécessaire de fixer au départ le nombre de modalités à représenter.

1. Les seuils fixés a 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 a une signification particulière et ne peut donc être remplacée par aucune autre.

La procédure FORMAT crée 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 ambiguïté possible avec un nom de variable). La syntaxe de la procédure est la suivante (pour q modalités):

PROC FORMAT

PROC FORMAT;
  VALUE nom du premier format
    première borne - seconde borne = première modalité
    .....
    q-1ème borne - q^{ème} borne = q^{ème} modalité;

Notons que si la première borne est la valeur minimale, elle peut prendre la valeur LOW; de même, si la qème borne est la valeur maximale, elle peut prendre la valeur HIGH. Par exemple, pour obtenir deux masques de recodage relatifs à la variable PCTETR85 du tableau VDSTAT en trois classes, il faut écrire le programme:

EXEMPLE PGMX-1
Discrétisation par seuils fixés a priori: PROC FORMAT

```
PROC FORMAT;
  VALUE FMTETRA
    LOW - 70 = '< 70 %'
    70 - 85 = '70 - 85 %'
    85 - HIGH = '> 85 %';
  VALUE FMTETRB
    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 référence 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.

2. Le centrage et la réduction: la procédure STANDARD.

Le centrage et la réduction consistent à exprimer les valeurs sur une échelle standardisée ayant une moyenne nulle et un écart-type égal à l'unité. Cette opération revient à soustraire de 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 créer un masque de recodage après cette standardisation pour pouvoir utiliser ensuite la procédure GMAP.

Voici la syntaxe de la procédure STANDARD:

PROC STANDARD

```
PROC STANDARD
   DATA=nom du tableau à transformer
      OUT=nom du tableau transformé
      MEAN=0 STD=1;
```

Pour discrétiser la variable PCTJEU85 du tableau VDSTAT à l'aide de la procédure STANDARD, il faut procéder à deux étapes PROC:

EXEMPLE PGMX-2
Discrétisation centrée/réduite: PROC STANDARD

```
PROC STANDARD
   DATA=BASE.VDSTAT
      OUT=JEUNES MEAN=0 STD=1;
   VAR PCTJEU85;
PROC FORMAT;VALUE FMTJEU
   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 JEUNES et le nom du format FMTJEU.

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 faut choisir de préférence un nombre impair de classes (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. Il faudra ensuite créer un masque de recodage exprimant le nombre de modalités choisi. La syntaxe de la procédure RANK est la suivante:

PROC RANK

GROUPS=4 pour les quartiles
GROUPS=10 pour les déciles

```
PROC RANK DATA=nom du tableau à transformer
   OUT=nom du tableau transformé
   GROUPS=nombre de modalités;
```

La discrétisation de la variable CONPCT60 du tableau VDSTAT sera donc réalisée en deux étapes PROC:

EXEMPLE PGMX-3
Discrétisation par quantiles: PROC RANK et PROC FORMAT

```
PROC RANK DATA=BASE.VDSTAT
   OUT=VDRANK GROUPS=4;
   VAR CONPCT80;
PROC FORMAT; VALUE FMTCON
   1='1er quartile'
```

2='2ème quartile'
3='3ème quartile'
4='4ème quartile';

A l'appel de la procédure GMAP, le nom du tableau contenant les données sera VDRANK et le nom du format sera FMTCON.

X.3.2

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'instruction PATTERN (ici pour la première trame) permet de choisir la couleur et le type de la trame:

PATTERN

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= nom de la couleur, 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:

M1L45 trame claire, hachures orientées à gauche à 45º.

M5X90 trame foncée, hachures en croisillons.

M3R60 trame médiane, hachures orientées à droite à 60º.
Différentes combinaisons possibles sont présentées

dans le manuel de référence SAS/GRAPH (figure 5.1, page 58).

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 entre valeurs statistiques et trames.

X.3.3

Le tracé de la carte: la procédure GMAP.

Le tracé d'une carte choroplèthe (Fig. X.1) 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

```
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';

En remplaçant l'instruction CHORO par les instructions PRISM ou SURFACE, on obtient des cartes d'aspects très différents qui donnent une impression de volume. Les premiers utilisateurs de SAS/GRAPH ont beaucoup abusé de ces procédés spectaculaires mais dont le résultat est difficile à lire.

La carte de la part des étrangers dans la population totale des communes du Canton de Vaud en 1985 (Fig. X.1) donne une bonne idée de ce qu'il est possible d'obtenir en sortie de GMAP. Cette carte en noir et blanc, retravaillée sur Macintosh II et imprimée sur LaserWriter Apple NTX-II, nécessite de faire appel à quelques «trucs» pour adapter la gamme de couleurs à une gamme de gris ordonnés. Un meilleur contrôle des densités de gris peut être obtenu avec la procédure UNISAS présentée en X.5.

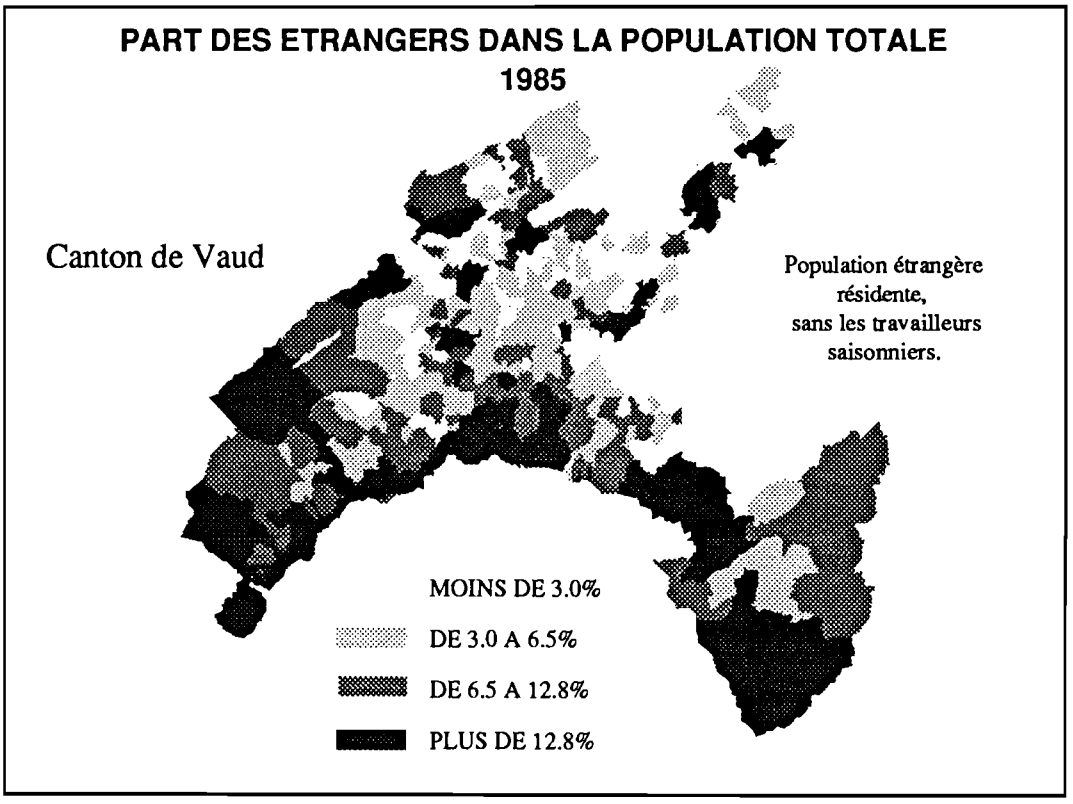


Figure X.1

Exemple d'une carte choroplèthe au moyen de la procédure GMAP

X.3.4

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:

EXEMPLE PGMX-4
Tracer un fond de carte vide

```
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/LEVELS=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.

X.3.5

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. Notons que la variable contenant la partition, et dont le nom doit figurer à la fois dans l'instruction CHORO et l'instruction BY, doit obligatoirement faire l'objet d'un tri préalable par la procédure SORT.

X.4

Le traitement des fonds de carte: GREDUCE et GREMOVE.

SAS/GRAPH propose trois procédures de traitement des fonds de carte. GPROJECT assure la projection d'un fond numérisé sous forme de coordonnées angulaires exprimées en radians. Elle permet de choisir entre un nombre limité de types de projection, Albers, Lambert et GNOMIC. GREDUCE réalise la généralisation des contours, c'est-à-dire l'élimination de points inutiles selon certains critères. Enfin, GREMOVE assure l'agrégation d'unités spatiales appartenant à un même niveau géographique (des communes aux agglomérations, par exemple), efface les contours intérieurs, et ne laisse donc subsister que les contours extérieurs de chaque nouvelle unité spatiale, au niveau d'agrégation retenu.

X.4.1

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 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'œuvre 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 VDFOND, numérisé de manière extrêmement précise à l'Université de Zurich (Herzog, A. et al. (1983), contient près de 10 000 coordonnées pour un peu plus de 300 communes. On compte donc, en moyenne, près de 30 points par commune, ce qui semble beaucoup pour la réalisation de cartes statistiques. La procédure GREDUCE doit permettre d'alléger ce fond, et donc d'en simplifier l'utilisation. Il s'agit de l'algorithme le plus fréquemment utilisé dans ce cadre connu sous le nom de Douglas-Peucker (1973). L'utilisation de GREDUCE requiert la syntaxe suivante:

PROC GREDUCE

```
PROC GREDUCE
  DATA=nom du tableau fond de carte initial
    OUT=nom du tableau fond de carte généralisé;
    ID nom de la variable identifiant les observations;
```

La généralisation est donc réalisée en fonction de plusieurs distances minimales séparant deux points consécutifs. Le tableau de fond de carte à généraliser possède une variable supplémentaire nommée DENSITY, qui prend 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 des niveaux de généralisation intermédiaire avec un plus petit nombre de points.

A la suite de GREDUCE, la procédure FREQ permet d'obtenir le nombre de points à chaque niveau de généralisation.

PROC FREQ

```
PROC FREQ
  DATA=nom du tableau fond de carte généralisé;
    TABLES DENSITY;
```

Enfin, on peut constituer le fond de carte final dans une étape DATA, en sélectionnant le niveau supérieur retenu en fonction du nombre de coordonnées choisi. Par exemple, si l'on ne désire retenir que les points nécessaires à un fond de carte généralisé aux niveaux 0, 1, 2 et 3, il faut soumettre l'étape DATA suivante:

```
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.

X.4.2

Agréger les unités spatiales

La technique nécessaire à l'agrégation des données à un niveau géographique supérieur à celui auquel elles ont été produites a été présenté au Chapitre III.4.3. Il s'agissait de calculer la population des régions écologiques de RUMLEY, en 1960, 1970 1980 et 1985. Nous allons examiner ici comment élaborer le fond de carte de ces régions écologiques alors que le fond de carte d'origine est disponible au niveau de la

commune.

La procédure GREMOVE assure le changement de niveau géographique. Sa syntaxe est la suivante:

PROC GREMOVE

```
PROC GREMOVE
  MAP=nom du tableau fond de carte initial
     OUT=nom du tableau fond de carte
     au niveau retenu;
     ID identifiant des unités spatiales
     dans le fond d'origine;
  BY variable d'agrégation.
```

Voici le programme nécessaire à la génération du fond de carte des régions RUMLEY:

EXEMPLE PGMX-5
Produire un fond agrégé

```
/*Définition du nom logique BASE*/
/*dépend du système d'exploitation*/

/*Ajouter la variable RUMLEY au fond VDFOND*/
PROC SORT DATA=BASE.VDGEO OUT=VDGEO;
      BY NUMERO;
DATA VDGEO; SET VDGEO;
      DROP X Y;
PROC SORT DATA=BASE.VDFOND OUT=VDFOND;
      BY NUMERO;
DATA VDFOND; MERGE VDFOND VDGEO;
      BY NUMERO;
   KEEP NUMERO SEGMENT X Y RUMLEY;

/*Générer le fond de carte RUMLEY*/
PROC SORT DATA=VDFOND OUT=VDFOND2;
      BY RUMLEY;
PROC GREMOVE
      DATA=VDFOND2 OUT=RUMFOND;
      ID NUMERO;
      BY RUMLEY;

/*Tracer le fond de carte vide*/
PROC GMAP DATA=RUMFOND
      MAP=RUMFOND ALL;
   ID RUMLEY;
   CHORO X/LEVELS=1;
   PATTERN1 V=E  C=CYAN;
RUN;
```

X.5

Représentation tridimensionelle de surfaces: la procédure G3D

La procédure G3D assure la représentation de surfaces dans un repère tridimensionnel. Elle permet donc de tracer des blocs-diagrammes. Ceci est extrêmement utile pour visualiser des surfaces de tendance dont les formes ont été calculées par une procédure de régression comme REG. Reprenons, par exemple, les surfaces calculées dans le Chapitre VI (VI.3.C). A l'issue de ces traitements, on dispose d'un tableau nommé VDSURF contenant les coordonnées X et Y des centres des unités spatiales, auxquelles s'ajoutent 6 variables nommées ESTIM1 à ESTIM6, donnant la valeur calculée en chaque point, pour chaque surface de degré 1 à 6. La procédure G3D réalise la représentation d'une surface de la manière suivante:

EXEMPLE PGMX-6
Générer une surface 3D

```
/*Suite du programme du Chapitre VI.3.C*/

/*Placer une grille carrée sur la surface*/
PROC G3GRID DATA=VDSURF OUT=MAILLE;
   GRID X*Y=ESTIM1 /
      NEAR=10 NAXIS1=10 NAXIS2=10;

/*Réaliser le tracé*/
PROC G3D DATA=MAILLE;
   PLOT Y*X=ESTIM1 /
      CTEXT=GREEN CAXIS=YELLOW
      CTOP=RED CBOTTOM=CYAN
      TILT=50 ROTATE=20;
RUN;
```

La première étape revient à placer une maille rectangulaire (ou carrée si les côtés sont égaux) sur la surface à représenter. La procédure G3GRID assure l'estimation des valeurs à chaque nœud de la maille. Plusieurs options sont disponibles; ici, NEAR produit une estimation qui dépend des 10 points les plus proches du point d'estimation. La surface est d'autant plus lissée que le nombre choisi pour ce paramètre est grand. NAXIS1 et NAXIS2 donnent les dimensions de la maille rectangulaire.

La seconde étape concerne le tracé de la surface proprement dit (Fig. X.2). Noter l'inversion de X et de Y dans l'expression PLOT, par rapport à GRID de G3GRID. Il est possible d'affecter une couleur, pour le dessous de la surface,

différente de celle du dessus. Signalons aussi la possibilité de faire tourner la figure autour de l'axe Z avec le paramètre ROTATE (ici ESTIM1) et des axes X et Y avec TILT.

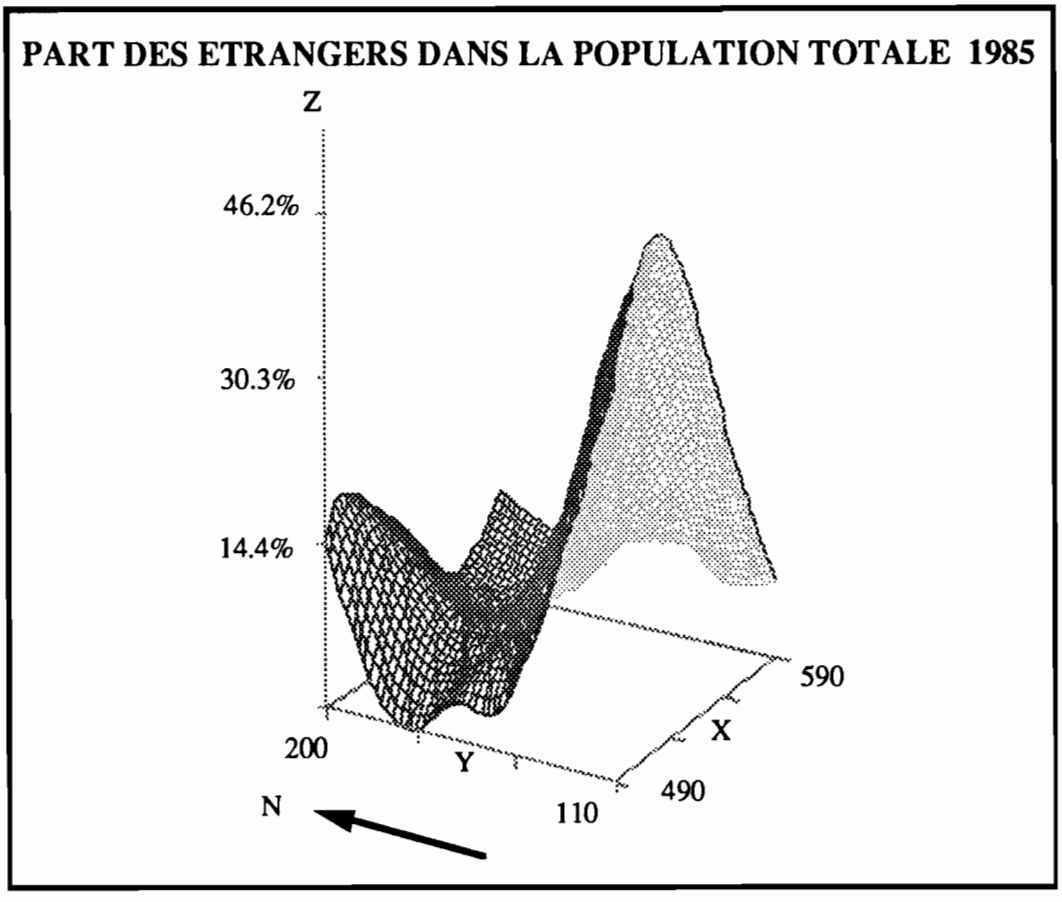


Figure X.2

Exemple d'une représentation d'une surface de tendance avec la procédure G3D

Cette surface est calculée à partir des mêmes données que celles utilisées pour réaliser la carte X.1. Il s'agit d'une tendance obtenue par ajustement d'un polynôme du troisième degré qui explique 44 % de la variance totale du pourcentage des étrangers résidents en 1985. L'angle de vue a été choisi de manière à bien faire apparaître les fortes valeurs sur le pourtour du lac Léman ainsi qu'à la frontière jurassienne. L'observateur est situé dans les environ du Pays de Gex. G3D permet de faire varier l'angle et le point de vue ainsi que les couleurs du dessus et du dessous de la surface afin de donner au lecteur l'illusion de la perspective. Pour spectaculaires quelles soient, il ne faut pas abuser de ces représentations qui n'ont véritablement d'intérêt que pour visualiser des surfaces de tendance.

X.6

Cartographie avec UNISAS

Utiliser la bilbiothèque graphique UNIRAS avec SAS: UNISAS

La Maison de la Géographie de Montpellier met à la disposition des chercheurs du réseau du GIP RECLUS un ensemble de programmes de cartographie statistique interfacés avec SAS qui porte le nom de la principale procédure: UNISAS. Réalisé par P. Brossier, cet ensemble complète SAS/GRAPH de manière très utile, mais uniquement sous le système d'exploitation OS/MVS. Il permet à l'utilisateur courant de SAS, sans quitter son système favori, de profiter d'une partie ressources offertes par la bibliothèque graphique UNIRAS, et plus précisément du module cartographique GEOPAK. C'est dire combien l'ouverture de SAS à des programmes extérieurs, déjà très utile pour l'analyse des données avec ADDAD, se revèle être, une fois encore, un argument de choix décisif.

UNISAS se compose des procédures suivantes:

UNISAS	Tracé de cartes statistiques en tous genres.
UPLAY	Récupération et habillage de graphiques UNIRAS.
SASUNI	Tracé de graphiques SAS sur des unités graphiques utilisables avec UNIRAS et non par SAS.
CENTRE	Calcul des coordonnées des centres des polygones figurant les unités spatiales dans les fonds de carte SAS.
CPOST	Génération d'un fichier PostScript à partir d'un fichier de segments UNIRAS.
PALET	Impression d'une palette de couleurs nécessaire au choix des couleurs dans la procédure UNISAS.

Le programme PGMX-7 donne un exemple d'utilisation de la procédure UNISAS pour le tracé d'une carte en isolignes de la densité de population dans le canton de Vaud (Fig. X.3). La première partie du programme permet de définir les bornes des classes avec PROC FORMAT, c'est-à-dire de définir les différents niveaux de densité. La seconde partie contient l'appel de la procédure UNISAS proprement dite. La signification de ces différentes instructions est donnée dans le manuel UNISAS rédigé par P. Brossier et disponible à la Maison de la Géographie. Notons que le type de carte est donné par l'instruction COURNIV.

EXEMPLE PGMX-7
UNISAS: carte en isolignes

```
/*Densité de population 1980*/

PROC FORMAT;
   VALUE FMTA
      LOW - 0.3    ='MOINS DE 0.3 HAB/HA.'
      0.3 - 0.4    =' 0.3 A 0.4'
      0.4 - 0.6    =' 0.4 A 0.6'
      0.6 - 1.6    =' 0.6 A 1.6'
      1.6 - 10.0   =' 1.6 A 10.0'
      10.0 - HIGH =' PLUS DE 10.0';

PROC UNISAS  DATA=BASE.VDSTAT
   MAP=BASE.VDFOND
      SOUTLINE=0 UNIT=TEK4692 BORDER;
   ID NUMERO;
   COURNIV DENS80
      POINT=BASE.VDGEO/INTER=1 NBCEL=8000;
   FORMAT DENS80 FMTA.;
      COULEUR N=10 C1=10 MODE=3;
      COULEUR N=11 C1=20 MODE=3;
      COULEUR N=12 C1=30 MODE=3;
      COULEUR N=13 C1=50 MODE=3;
      COULEUR N=14 C1=80 MODE=3;
      COULEUR N=15 C1=100 MODE=3;
   TITRE T='DENSITE DE POPULATION 1980' J=1;
   NOTE T='CANTON DE VAUD - MICHELINE
      COSINSCHI ET PHILIPPE WANIEZ' J=1;
%TEK4692;

/*Fin du programme*/
```

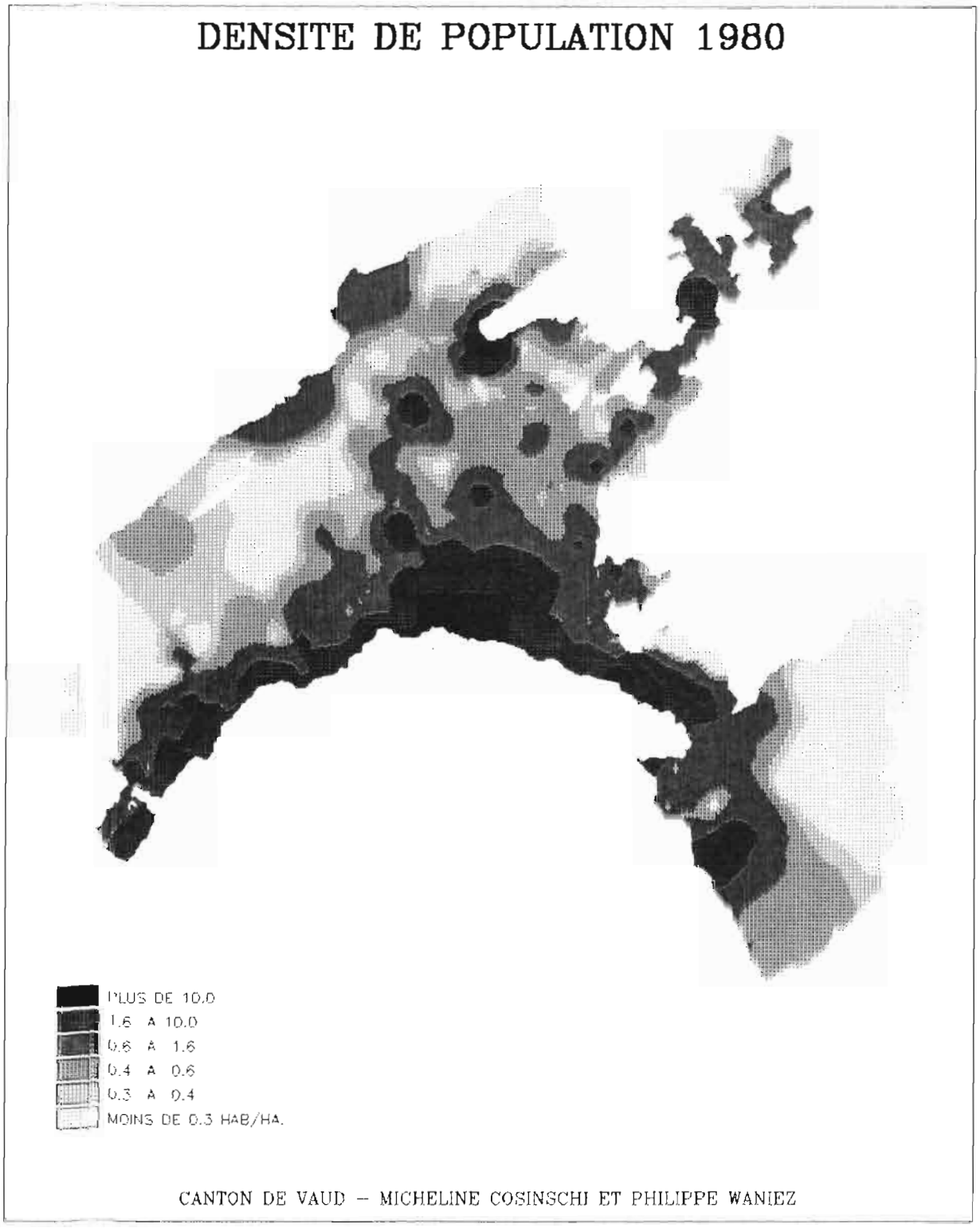


Figure X.3

Exemple d'une carte en isolignes avec UNISAS

Si le tracé de cartes en courbes de niveaux constitue sans doute l'une des sorties les plus spectaculaires d'UNISAS (en couleur, c'est encore plus saisissant!), UNISAS assure également le tracé de cartes plus classiques comme les cartes choroplèthes (commandées par l'instruction CHORO), ou bien encore les cartes en symboles proportionnels (commandées par l'instruction CERCLE), plus difficiles à réaliser avec GMAP. Les programmes PGMX-8 et PGMX-9 présentent des exemples de ces cartes choroplèthes et ponctuelles, respectivement sur l'évolution de la population des communes du Canton de Vaud entre 1970 et 1980 (Fig. X.4) et sur la population en 1980 (Fig. X.5).

EXEMPLE PGMX-8
UNISAS: carte choroplèthe

```
/*Evolution de la population entre 1970 et 1980*/

PROC FORMAT;
VALUE
   FMTA LOW - -18.2='MOINS DE -18.2%'
   -18.2 - -11.4          =' -18.2 A -11.4'
   -11.4 -  0.0           =' -11.4 A  0.0'
   0.0 -  18.1            ='  0.0 A +18.1'
   18.0 <  65.8           =' +18.0 A +65.8'
   65.8 < HIGH            ='PLUS DE +65.8%';
PROC UNISAS DATA=BASE.VDSTAT
   MAP=BASE.VDFOND
   SOUTLINE=1 UNIT=TEK4692 BORDER;
ID NUMERO;
CHORO TPOP7080;
FORMAT  TPOP7080  FMTA.;
   COULEUR N=10 C1=10  MODE=3;
   COULEUR N=11 C1=15  MODE=3;
   COULEUR N=12 C1=30  MODE=3;
   COULEUR N=13 C1=50  MODE=3;
   COULEUR N=14 C1=80  MODE=3;
   COULEUR N=15 C1=100 MODE=3;
TITRE T='EVOLUTION DE LA POPULATION
      1970-1980' J=1;
NOTE  T='CANTON DE VAUD - MICHELINE
      COSINSCHIET PHILIPPE WANIEZ' J=1;
%TEK4692;

/*Fin du programme*/
```

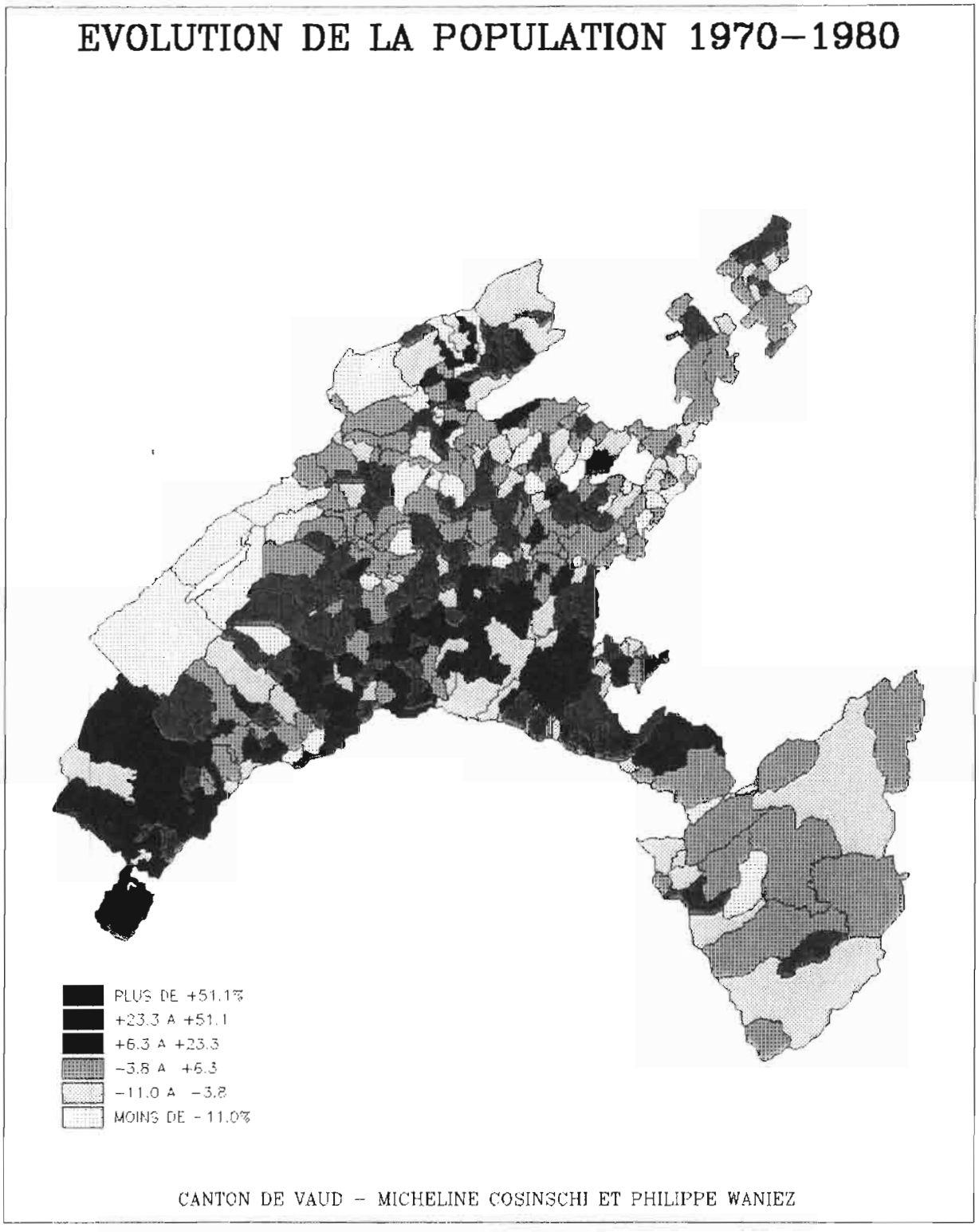


Figure X.4

Exemple d'une carte choroplèthe avec UNISAS

EXEMPLE PGMX-9
UNISAS: carte ponctuelle

```
/*Population en 1980*/

DATA VDPOP; SET BASE.VDPOP; INDIC=1;
PROC UNISAS DATA=VDPOP
   MAP=BASE.VDFOND
      SOUTLINE=1UNIT=TEK4692 LEGH=0 BORDER;
   ID NUMERO;
   CERCLE POP80 INDIC
      POINT=BASE.VDGEO/ RMIN=0.06 RMAX=5.0
      COUTLINE=1 LEGH=2 CIRCON;
   COULEUR N=12 C1=100 MODE=3;
TITRE T='POPULATION TOTALE 1980'J=1;
NOTE  T='CANTON DE VAUD - MICHELINE
      COSINSCHI ET PHILIPPE WANIEZ' J=1;
%TEK4692;

/*Fin du programme*/
```

Pour de plus amples informations sur l'environnement UNISAS, adressez-vous à la Maison de la Géographie, à l'adresse suivante:

Groupement d'Intérêt Public RECLUS
Maison de la Géographie
17, rue Abbé de l'Epée
34000 Montpellier

☎ 67 72 46 10
(Fax) 67 72 64 04

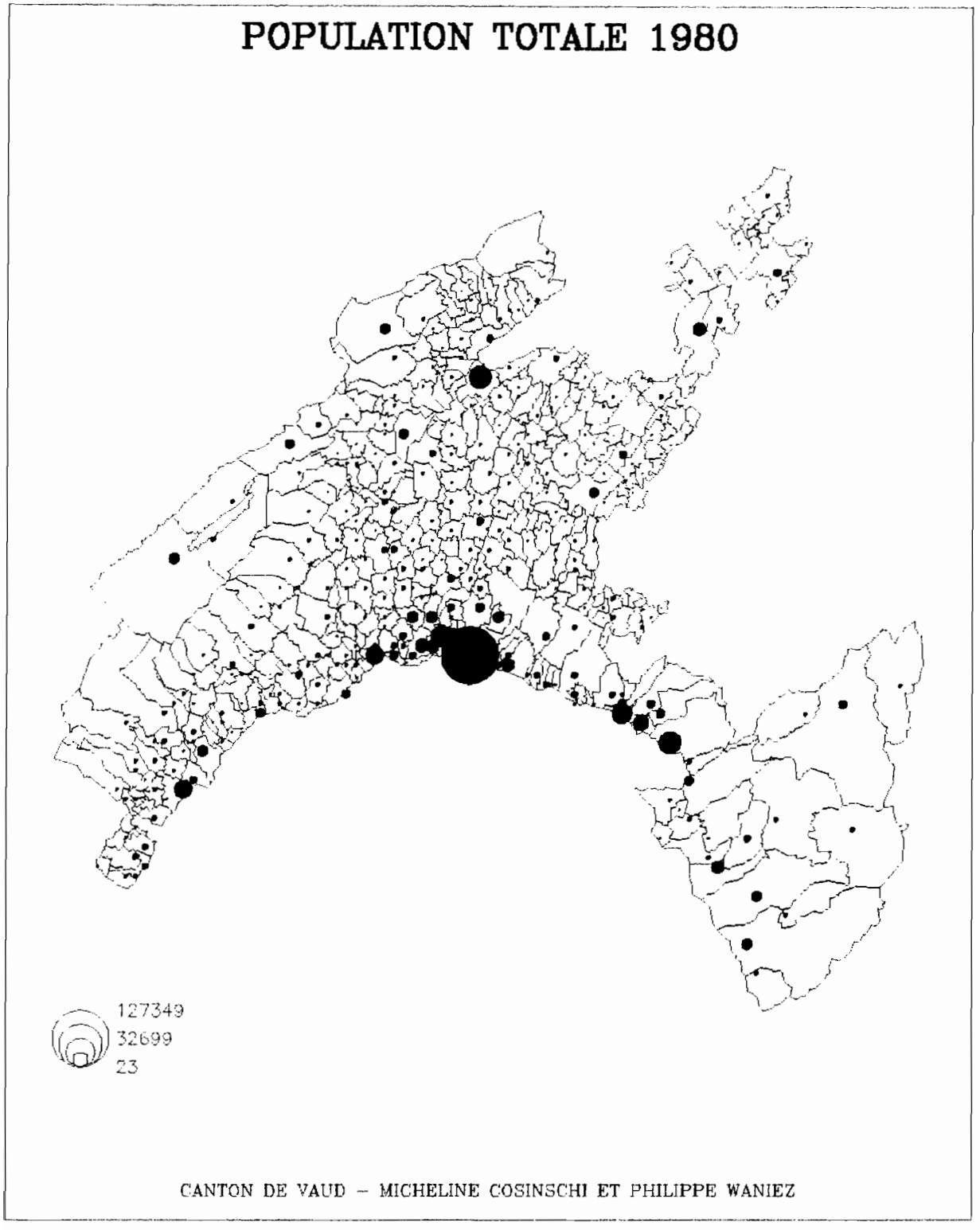


Figure X.5

Exemple d'une carte en cercles proportionnels avec UNISAS

Bibliographie

Références citées

CABOS, V. et P. WANIEZ (1987) *Les données et le territoire. Initiation à la numérisation pour la cartographie statistique.* Reclus, Montpellier, GIP Reclus.

DOUGLAS, D.H., et T.K. PEUCKER (1973) «Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature». *The Canadian Cartographer*, Vol. 10, Nº 2, pp. 112-122.

HERZOG, A., et al. (1983) *Ein System zur Generierung von Grenzliniendateien Administrativer Einheiten der Schweiz*, Geo-Processing Reihe, Geoprocessing Series, Vol. 3, Geographisches Institut der Universität, Zürich.

OFS *Recensement fédéral de la population 1980.* Berne, Statistique de la Suisse.

RUMLEY, P.-A. (1984) *Aménagement du territoire et utilisation du sol. Evolution passée et schémas prospectifs de l'utilisation du sol en Suisse.* Berichte zur Orts-, Regional- und Landesplanung Nr. 50, Zürich, Institut für Orts-, Regional- und Landesplanung, ETH Zürich.

SCHULER, M. (1984) *Délimitation des agglomérations en Suisse 1980.* Contributions à la statistique suisse/105^{e} fascicule, Berne et Lausanne, Institut de recherche sur l'environnement construit (IREC- Lausanne) et Office fédéral de la statistique (OFS, Berne).

Orientations de lecture

La présente bibliographie ne prétend pas à l'exhaustivité. Le lecteur y trouvera un choix d'ouvrages en français classés par thèmes et niveaux de difficulté. Ces derniers sont estimés sur le plan technique et mathématique et correspondent à peu près à l'échelle suivante :

* facile (bac littéraire)
** assez facile (bac scientifique, ou premier cycle de géographie)
*** difficile (premier cycle supérieur scientifique)
**** très difficile (second et troisième cycles scientifiques).

Ouvrages de base en statistique ou mathématique.

BARBUT, M. (1970) **
Mathématiques des sciences humaines.
Vol. 1, *Combinatoire et algèbre*, 254 p.
Vol. 2, *Nombres et mesures*, 293 p.
Paris, PUF.

GILBERT, N. (trad. J.G. Savard) (1978) *
Statistiques.
Montréal, Eds. HRW Ltée, 384 p.

HODGES, J.R., D. KRECH, R.S. CRUTCHFIELD (1979) *
Statlab. Paris, Economica, 373 p.

LIORZOU, A. (1973) *
Initiation pratique à la statistique.
Paris, Eyrolles, 314 p.

MONJALLON, A. (1969) *
Introduction à la méthode statistique.
Paris, Librairie Vuibert, 278 p.

Analyse des données.

BENZECRI, J.P. et F. (1980) ***
Pratique de l'analyse des données.
Vol. 1, *Analyse des correspondances, exposé élémentaire.*
Paris, Dunod, 424 p.

CAILLEZ, F. et J.P. PAGES (1976) ****
Introduction à l'analyse des données.
Paris, Asu-Buro-Smash, 616 p.

CIBOIS, P. (1983) **
L'analyse factorielle.
Paris, PUF, Col. Que sais-je?, n°2095, 127 p.

DIDAY, E., J. LEMAIRE, J. POUGET, F. TESTU (1982) ***
Eléments d'analyse des données.
Paris, Dunod, 462 p.

ESCOFIER, B. et J. PAGES (1988) ***
Analyses factorielles simples et multiples.
Paris, Dunod, 241 p.

FENELON, J.P. (1981) **
Qu'est-ce que l'analyse des données.
Paris, Lefonen, 311 p.

JAMBU, M. et M.O. LEBEAUX (1978) ***
Classification automatique pour l'analyse des données.
Vol. 1, *Méthodes et algorithmes*, 310 p.
Vol. 2, *Logiciels*, 399 p.
Paris, Dunod.

LEBART, L., A. MORINEAU, J.P. FENELON (1979) ***
Traitement des données statistiques, méthodes et programmes.
Paris, Dunod, 511 p.

LERMAN, I.C. (1981) ****
Classification et analyse ordinale des données.
Paris, Dunod, 740 p.

ROMEDER, J.M. (1973) ****
Méthodes et programmes d'analyse discriminante.
Paris, Dunod, 274 p.

TOMASSONE, R., E. LESQUOY, C. MILLIER (1983) **
La régression. Nouveaux regards sur une ancienne méthode statistique.
Actualités scientifiques et agronomiques de l'INRA, 13, Paris, Masson, 180 p.

VOLLE, M. (1978) ***
Analyse des données.
Paris, Economica, 267 p.

Applications à l'analyse géographique.

BEGUIN, M. (1979) **
Méthodes d'analyse géographique quantitative.
Paris, Litec, 252 p.

CAUVIN, C., H. REYMOND, A. SERRADJ (1987) **
Discrétisation et représentation cartographique.
Montpellier, RECLUS, Col. Reclus modes d'emploi, 116 p.

CAUVIN, C. et H. REYMOND (1986) *
Nouvelles méthodes en cartographie.
Montpellier, RECLUS, Col. Reclus modes d'emploi, 56 p.

CHARRE, J. et P. DUMOLARD (1989) *
Initiation aux pratiques informatiques en géographie.
Paris, Masson, 199 p.

CICERI, M.F., B. MARCHAND, S. RIMBERT (1977)**
Introduction à l'analyse de l'espace.
Paris, Masson, 173 p.

CLAVAL, P. (1977) *
La nouvelle géographie.
Paris, PUF, Col. Que sais-je?, n°1693, 126 p.

DANDOY, G. et col. (1986) *
Traitement des données localisées.
Paris, ORSTOM, Col. Colloques et séminaires, 304 p.

Groupe CHADULE (1987) *
Initiation aux pratiques statistiques en géographie.
Paris, Masson, 189 p.

RACINE, J.B. et H. REYMOND (1973) **
L'analyse quantitative en géographie.
Paris, PUF, Col. SUP-le géographe, n°12, 316 p.

SANDERS, L. et F. DURAND-DASTES (1985) **
L'effet régional.
Montpellier, RECLUS, Col. Reclus modes d'emploi, 48 p.

WANIEZ, P. (1988) **
Les données et le territoire : initiation à l'analyse en surface de tendance.
Paris, ORSTOM, Col. IDT, 45 p., disquette.

*

* *

Le logiciel SAS comporte plusieurs modules assurant les fonctions de gestion, de préparation et de traitement des données. La diversité des procédures et des options disponibles rend impossible leur description exhaustive et il est fortement recommandé d'avoir recours à la documentation fournie par SAS Institute Inc., disponible, jusqu'à ce jour, en anglais uniquement.

Documentation SAS.

SAS User's Guide: Basics, Version 5 Edition.

SAS User's Guide: Statistics, Version 5 Edition.

SAS/GRAPH User's Guide, Version 5 Edition.

SAS/FSP User's Guide, Version 5 Edition.

Documentation SAS pour micro-ordinateurs.

SAS Introductory Guide for Personal Computers.

SAS Language Guide for Personal Computers.

SAS Procedures Guide for Personal Computers.

SAS/STAT Guide, for Personal Computers.

Introduction à SAS.

CODY, R.P. and J.K. SMITH (1987)
Applied Statistics and the SAS Programming Language. 2nd ed. New York, North-Holland, 280 p.

Pour prendre contact avec SAS Institute:

SAS Institute Inc.
Box 8000
Cary, North Carolina 27511-8000
U.S.A.

SAS Institute S.A.
50, avenue Daumesnil
75013 Paris

ANNEXE

Notes sur les systèmes d'exploitation

SAS fonctionne sur les ordinateurs aussi différents que les gros systèmes IBM, sous OS/MVS, les micro-ordinateurs IBM PC et PS avec MS-DOS, ou bien encore, hors de la gamme IBM, les mini-ordinateurs VAX de Digital Equipment, avec le système d'exploitation VMS. Sans entrer dans le détail de ces différents sytèmes d'exploitation, il apparaît souhaitable de signaler quelques points utiles à ceux qui souhaitent utiliser SAS rapidement, sur le système auquel ils ont accès. Pour plus d'information, on se reportera aux manuels fournis par les constructeurs, et, éventuellement aux notices disponibles dans les centres informatiques.

A.1

SAS sous MS-DOS (IBM PC et PS)

L'arrivée sur le marché de SAS PC permet aux utilisateurs de SAS d'utiliser leur logiciel préféré sur des machines bon marché. A la différence de la majorié des logiciels fonctionnant sous ce système d'exploitation, SAS PC n'est pas interfacé avec des menus. Comme avec les autres machines sur lesquelles fonctionne SAS, l'utilisateur doit obligatoirement apprendre le langage. On ne peut donc raisonnablement recourir à SAS sur PC de manière occasionnelle sauf si l'on est déjà un utilisateur de SAS sur mini ou gros système.

La configuration minimale requise est un ordinateur IBM PC, PS ou compatible disposant d'au moins de 512k octets de mémoire centrale et d'un disque dur de 20 Mo. L'utilisation de SAS/GRAPH pour faire de la cartographie nécessite une

extension de la mémoire centrale. Enfin, un coprocesseur arithmétique 8087 ou 80287 est vivement recommandé pour le traitement des grands tableaux.

Le logiciel est divisé en plusieurs modules. Tous les utilisateurs doivent avoir le module nommé BASE/SAS qui contient le cœur du système. Les autres modules sont SAS/STAT pour la statistique, SAS/GRAPH pour les graphiques (y compris la cartographie), SAS/IML pour le calcul matriciel et SAS/AF pour le développement d'applications clés en mains. D'autres modules sont en préparation.

A. Le Display Manager et le dialogue avec SAS

L'installation du système se fait assez simplement en suivant les instructions du manuel. Ensuite, pour appeler SAS, il suffit d'entrer la commande SAS en étant dans le répertoire qui contient le système (donc après avoir fait CD nom du répertoire).

SAS affiche alors l'image du Display Manager. Elle apparaît quelque peu différente de celle des autres systèmes d'exploitation: l'écran OUTPUT figure dans la partie supérieure (en bleu clair), l'écran LOG au centre (en gris) et l'écran PROGRAM EDITOR (en bleu foncé) dans la partie inférieure.

A partir de ce moment, il n'y a pas de différence d'utilisation, si ce n'est la définition des touches de fonction qui est, par défaut, la suivante:

F1 Help
F2 Définition des touches de fonction
F3 Affichage de l'écran LOG
F4 Affichage de l'écran OUTPUT
F6 Affichage de l'écran PROGRAM EDITOR
F7 Touche de Zoom
F9 Rappel du programme RECALL
F10 Soumission du programme SUBMIT

B. Accéder à une base SAS et aux fichiers extérieurs.

Sous MS DOS, une base SAS est un répertoire. Chaque tableau SAS est un fichier de ce répertoire dont le nom s'achève par le suffixe SSD. Le contenu de la base peut être listé par la procédure CONTENTS. Avant toute utilisation, il apparaît donc nécessaire d'indiquer le répertoire qui contient la base

SAS. On accède à ce répertoire avec la commande« libname». Elle permet de définir le nom logique de la base, nommé BASE dans tout cet ouvrage. Elle s'écrit:

LIBNAME BASE 'nom du répertoire';

Le nom du répertoire (pathname) doit comprendre non seulement le nom proprement dit, mais aussi le chemin à parcourir pour l'atteindre. Par exemple, le nom suivant 'C:\GEO' indique que les tableaux SAS seront enregistrés dans un répertoire nommé GEO du disque C. Bien entendu, ce répertoire doit exister et donc avoir été créé avec la commande MD nom du répertoire, propre à MS DOS.

La lecture d'un fichier externe ne requiert pas une instruction supplémentaire pour établir la relation entre le nom logique, connu par SAS, et le nom physique sur le disque. Il suffit d'indiquer dans l'instruction INFILE le nom physique:

INFILE 'nom du fichier';

Notons que le nom du fichier doit s'achever par le suffixe .DAT et qu'il peut contenir l'ensemble du chemin (pathname).

Les fichiers programmes peuvent être créés soit par un traitement de texte (dans ce cas, il faut sauvegarder en ASCII, en TEXTE ou en NON-DOCUMENT, selon la terminologie du logiciel retenu), soit directement par le Display Manager. Un fichier est créé par l'instruction:

SAVE 'nom du fichier'

entrée sur la ligne COMMAND du Program Editor; par exemple,

SAVE MONPROG

Si un fichier de ce nom existe déjà, le Display Manager demande de confirmer son remplacement par la nouvelle version. Enfin, l'inclusion du contenu d'un fichier programme dans le Program Editor du Display Manager se fait par la commande:

INCLUDE 'nom du fichier'

entrée sur la ligne COMMAND du Program Editor; par exemple,

INCLUDE MONPROG

Notons à nouveau qu'il ne faut pas donner le suffixe .SAS, bien que le fichier le possède obligatoirement.

A.2

SAS sous VAX VMS

A. Modes conversationnel et non conversationnel.

L'appel de SAS avec VAX/VMS (Digital Equipment Corporation) se fait de plusieurs manières. En mode conversationnel, il suffit d'entrer la commande SAS:

SAS/DMS

SAS demande alors le type d'écran et affiche ensuite le DISPLAY MANAGER.

Il est aussi possible d'utiliser SAS en mode non interactif en faisant suivre la commande SAS d'un nom de fichier contenant un programme SAS. Ce fichier doit posséder le suffixe .DAT et contenir un programme correct; par exemple, si le programme figure dans un fichier nommé MONPROG.SAS, on peut l'exécuter en mode non interactif en entrant la commande:

SAS MONPROG

Les résultats des traitements seront écrits dans un fichier nommé MONPROG.LIS et le journal de bord dans MONPROG.LOG.

B. Accéder à une base SAS et aux fichiers extérieurs.

Avec VAX VMS (Digital Equipment Corporation), une base SAS est un répertoire. Chaque tableau SAS est un fichier de ce répertoire dont le nom s'achève par le suffixe SSD. Le contenu de la base peut être listé par la procédure CONTENTS. Avant toute utilisation, il apparaît donc nécessaire d'indiquer le répertoire qui contient la base SAS. On accède à ce répertoire avec la commande LIBNAME. Elle permet de définir le nom logique de la base, nommé BASE dans tout cet ouvrage. Elle s'écrit:

LIBNAME BASE '[nom du répertoire]';

S'il s'agit du répertoire par défaut, celui dans lequel on a appelé SAS, cette commande peut être abrégée de la manière suivante:

LIBNAME BASE '[]';

L'accès à un fichier extérieur à la base se fait avec la commande ASSIGN qui établit la connexion entre le nom physique et le nom logique. Elle a pour syntaxe:

ASSIGN [nom du répertoire] nom du fichier nom logique

S'il s'agit du répertoire par défaut, celui dans lequel on a appelé SAS, cette commande peut être abrégée de la manière suivante:

ASSIGN nom du fichier nom logique

Notons aussi que les commandes VMS comme ASSIGN, par exemple, peuvent être entrées dans une ligne de programme du type:

X 'ASSIGN nom du fichier nom logique';

C. Utilisation de fichiers programmes avec le Display Manager.

Comme nous l'avons déjà indiqué, les noms des fichiers programmes doivent s'achever par le suffixe .SAS. De tels fichiers peuvent être créés soit par l'éditeur de VMS, soit directement par le Display Manager. Un fichier est créé par l'instruction:

SAVE nom du fichier

entrée sur la ligne COMMAND du Program Editor; par exemple,

SAVE MONPROG

crée un fichier qui s'appelera MONPROG.SAS. Dans ce cas, il ne faut donc pas donner le suffixe .SAS. Ce dernier le fera automatiquement. Si un fichier de ce nom existe déjà, le Display Manager demande de confirmer son remplacement par la nouvelle version. Il n'y a aucun risque de faire une grave erreur puisque VMS complète les noms de fichiers par un numéro de version. Un fichier «écrasé» par erreur reste donc toujours récupérable en demandant la version précédente (si

entre temps elle n'a pas été effacée, bien entendu...).

Enfin, l'inclusion du contenu d'un fichier programme dans le Program Editor du Display Manager se fait par la commande:

INCLUDE nom du fichier

entrée sur la ligne COMMAND du Program Editor; par exemple,

INCLUDE MONPROG

Notons à nouveau qu'il ne faut pas donner le suffixe .SAS, bien que le fichier le possède obligatoirement.

A.3

SAS sous OS/MVS

Sous ce système d'exploitation, SAS fonctionne aussi bien en traitement par lot (batch processing) qu'en conversationnel sous TSO. Avec OS/MVS, une base SAS est un unique fichier, d'organisation DA (direct access). Ce fichier peut contenir un ou plusieurs tableaux SAS et son contenu listé par la procédure CONTENTS. En batch, comme sous TSO, il faudra donc définir ce fichier avant toute utilisation. Si le fichier contenant la base n'existe pas, il faudra le créer. S'il existe, on pourra y avoir accès en lecture seulement (ressource partagée) ou bien en lecture et en écriture (ressource non partagée).

A. Traitement par lot.

Il est nécessaire de faire précéder le programme SAS d'un flot de JCL qui précise la nature et la quantité de ressources nécessaires à la bonne exécution de ce programme.

```
// carte JOB
//    EXEC SAS, REGION=2000k
//BASE DD DSN=XXX1111.UTIL.FICHIER,
//   DISP=OLD
DATA VDSTAT; SET BASE.VDSTAT;
//
```

Dans ce cas de figure, la carte DD, de ddname BASE

donne accès au fichier contenant la base SAS. Cette instruction prend différentes formes selon l'utilisation de la base. En premier lieu, la base n'existe pas et il faut donc la créer, c'est-à-dire la définir complètement.

```
//BASE DD DSN=XXX1111.UTIL.FICHIER,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=SYSDA,VOL=SER=DISK1,
//    SPACE=(TRK,10)
```

En second lieu, la base existe déjà, et on souhaite en modifier le contenu; la carte DD aura donc le libellé suivant:

```
//BASE DD DSN=XXX1111.UTIL.FICHIER,
//    DISP=OLD
```

Enfin, si on souhaite accéder à une base en lecture seulement, ou en permettre l'accès en lecture seulement à plusieurs utilisateurs, le paramètre DISP=OLD doit être remplacé par DISP=SHR (pour «sharable», partagé).

Notons enfin que l'accès à un fichier extérieur à la base se fait également par une carte DD. Par exemple, pour la lecture du fichier VDPOP, on soumettra le job suivant:

```
// carte JOB
//    EXEC SAS, REGION=2000k
//BASE DD DSN=XXX1111.UTIL.FICHIER,
//    DISP=OLD
//IN DD DSN=XXX1111.UTIL.VDPOP,DISP=SHR
DATA BASE.VDPOP;
INFILE IN;
INPUT  NUMERO $ 1-4 POP60 6-11 POP70
  (POP80 POP85) (7.0) ETR85 34-39 JEUNES 7.0;
LABEL
NUMERO='NUMERO DE COMMUNE OFS'
POP60='POPULATION TOTALE EN 1960'
POP70='POPULATION TOTALE EN 1970'
POP80='POPULATION TOTALE EN 1980'
POP85='POPULATION MOYENNE EN 1985'
ETR85='POPULATION ETRANGERE EN 1985'
JEUNES='POPULATION DE 0 A 14 ANS EN 1980';
//
```

B. Conversationnel sous TSO.

L'appel de SAS se fait en entrant la commande SAS soit sous TSO, après le message READY, soit sous SPF 1.6. Comme dans toute utilisation de TSO, la commande ALLOC remplace la carte DD. En voici la syntaxe pour une base en création, pour

une base en lecture seulement, ou une base en lecture et écriture:

```
ALLOC DA('XXX1111.UTIL.FICHIER') FI(BASE)
   NEW SP(10) TRACKS
ALLOC DA('XXX1111.UTIL.FICHIER') FI(BASE) SHR
ALLOC DA('XXX1111.UTIL.FICHIER') FI(BASE) OLD
```

Cette commande ALLOC peut être entrée soit avant l'appel de SAS, soit sous SAS avec l'instruction:

```
X ALLOC DA('XXX1111.UTIL.FICHIER') FI(BASE) OLD;
```

Enfin, on accède aussi à un fichier extérieur avec la commande ALLOC:

```
X ALLOC DA('XXX1111.UTIL.VDPOP) FI(IN) SHR;
```

C. Programmes SAS dans un fichier.

Il est courant de conserver les programmes SAS dans des fichiers. La saisie des programmes peut être réalisée sous n'importe quel éditeur, couramment avec 1.2 de SPF(EDIT). Dans ce cas, le programme est rangé dans une bibliothèque SPF, qui est en fait un fichier partitionné. Pour exécuter ce programme, il faut l'inclure dans le programme source SAS à l'aide de l'instruction INCLUDE. Supposons que le programme de lecture de VDPOP ait été saisi dans le fichier nommé XXX1111.UTIL.PGM(LECPOP), voici ce qu'il faudrait faire en batch et sous TSO:

```
// carte JOB
//    EXEC SAS, REGION=2000k
//BASE DD DSN=XXX1111.UTIL.FICHIER,
//    DISP=OLD
//IN DD DSN=XXX1111.UTIL.VDPOP,DISP=SHR
//SASIN DD DSN=XXX1111.UTIL.PGM(LECPOP),
//    DISP=SHR
%INCLUDE SASIN;
//

SAS
X ALLOC DA('XXX1111.UTIL.FICHIER')
   FI(BASE) OLD;
X ALLOC DA('XXX1111.UTIL.VDPOP) FI(IN) SHR;
X ALLOC DA('XXX1111.UTIL.PGM(LECPOP)
   FI(SASIN) SHR;
%INCLUDE SASIN;
RUN;
```

Index

Imprimeur : OFFSET 2000
Adresse : Espace Commercial - Lot 49 ,
Montpellier - Fréjorgues
34130 MAUGUIO (France)
Tél : 67 64 63 64

COLLECTION RECLUS MODES D'EMPLOI

1. Observatoire de la dynamique des localisations. Création de l'information. Manuel pour l'emploi du bordereau d'enregistrement des données (avril 1985).

2. Pour la Géographie Universelle, charte de la rédaction (juillet 1985).

3. Thérèse SAINT-JULIEN, La diffusion spatiale des innovations (septembre 1985).

4. Lena SANDERS, François DURAND-DASTES, L'effet régional: les composantes explicatives dans l'analyse spatiale (novembre 1985).

5. Robert FERRAS, L'Espagne, écritures de géographie régionale (décembre 1985).

6. Colette CAUVIN, Henri REYMOND, Nouvelles méthodes en cartographie (janvier 1986).

7. Jean-Paul VOLLE, Bulgarie: les Systèmes de peuplement (février 1986).

8. André DAUPHINÉ, Jean-Yves OTTAVI, Atlas structurel des climats de la France (mai 1986).

9. Colette CAUVIN, Henri REYMOND, Abdelaziz SERRADJ, Discrétisation des données et représentation cartographique (mars 1987).

10. Anne-Marie LAKOTA, Christian MILELLI (coord.), Emplois, entreprises et équipements en Ile-de-France (avril 1987).

11. Fernand JOLY, Carte géomorphologique de la France au 1:1 000 000, quart Nord-Ouest (juin 1987).

12. André DAUPHINÉ, Christine VOIRON-CANICIO, Variogrammes et structures spatiales (février 1988).

13. Fernand JOLY, Carte géomorphologique de la France au 1:1 000 000, quart Nord-Est (novembre 1988).

14. Robert FERRAS, Les Géographies Universelles et le monde de leur temps (mai 1989).

15. Micheline COSINSCHI, Philippe WANIEZ, Pratique de l'analyse statistique SAS sur PC/PS, mini et gros systèmes (juillet 1989).

Prix: 95 F