Une etude quantitative statique de programmes Pascal

A. Schroeder

INRIA

CENTRE DE ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
BP 105
78153 Le Chesnay Cedex
France
Tél. (3) 954 90 20

Rapports Techniques

N° 29

UNE ÉTUDE QUANTITATIVE STATIQUE DE PROGRAMMES PASCAL

Anne SCHROEDER

Octobre 1983

UNE ETUDE QUANTITATIVE STATIQUE DE PROGRAMMES PASCAL

Anne SCHROEDER

INRIA - B.P. 105
78153 Le Chesnay (France)

Résumé :

L'objet de cet article est de faire le point sur l'information apportée par diverses mesures de complexité statique des programmes proposées dans la littérature. Pour celà, on présente les résultats de traitements statistiques effectués sur un échantillon de telles mesures recueillies sur un millier de blocs Pascal.

Les conclusions concernent la pertinence de quelques mesures classiquement utilisées, et la proposition d'une approche alternative qui semble apporter une information complémentaire.

Abstract:

This paper is an attempt to compare the information conveyed by various metrics of static complexity of programs that are proposed in the literature. The results presented are those of a thorough statistical analysis of a sample of such metrics, gathered on a thousand Pascal blocks.

Conclusions address the relevance of a few classical metrics, and the proposition of an alternative approach which seems to bring complementary information.

Ce travail a été effectué avec l'aide de l'Agence pour l'Informatique (ADI)

+++
+

UNE ETUDE QUANTITATIVE STATIQUE DE PROGRAMMES PASCAL

Anne SCHROEDER

INRIA - B.P. 105
78153 Le Chesnay (France)

-Octobre 1983-

1 – Introduction

Ces dernières années ont vu apparaître une abondante littérature sur la mesurabilité du logiciel [Cook82, Curtis80, Friedman81, Perlis81, parmi beaucoup d'autres]. Les auteurs abordent le sujet sous des angles variés : certains s'intéressent davantage aux mesures de qualité ou de fiabilité, d'autres aux mesures de complexité, d'autres enfin à des mesures purement descriptives et tous aux liaisons possibles entre ces différents types de mesures. Selon les objectifs, les mesures considérées sont statiques (c'est-à-dire ne dépendant que du texte du programme) ou dynamiques (c'est-à-dire relevées à l'exécution et donc dépendantes d'un jeu de données). Pour aborder une étude globale de la mesurabilité du logiciel, nous avons été amenés à concevoir un ensemble d'outils de mesures statiques et dynamiques, s'appliquant actuellement aux programmes Pascal, mais susceptibles d'être prochainement étendus à d'autres langages [Schroeder83a et 83b]. Ces outils ont été développés dans l'environnement de programmation multi-langage *Mentor* [Donzeau75, 80 et 83].

L'objet de cet article est de présenter les résultats de traitements statistiques effectués sur un échantillon de mesures statiques prélevées à l'aide ces outils; cette étude se propose de faire le point sur l'information apportée par diverses mesures proposées dans la littérature. Nous commencerons, au paragraphe 2, par décrire les mesures traitées (variables mesurées et échantillons considérés); puis nous présenterons dans les paragraphes 3 et 4 respectivement la description simple des variables puis l'analyse de l'information mutuelle qu'elles apportent; une analyse particulière de la distribution des opérateurs du langage Pascal fera l'objet du paragraphe 5; finalement, on essayera, au paragraphe 6 d'identifier différents styles de programmation en Pascal.

2 – Description des mesures traitées

2.1 - Variables mesurées

La plupart des mesures couramment utilisées dans la littérature concernant les propriétés quantitatives du logiciel sont présentes dans cette étude.

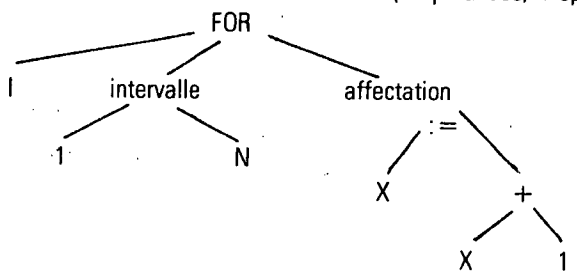
Il s'agit des mesures suivantes :

a - le **nombre d'instructions** exécutables du programme (pour certains langages comme le Fortran, cette mesure est identique au nombre de lignes de code);

b - les distributions et les statistiques d'utilisation des opérateurs du langage et des opérandes : nombres totaux d'opérateurs et d'opérandes utilisés ainsi que nombres d'opérateurs et d'opérandes distincts rencontrés; de ces mesures brutes, on peut déduire simplement une notion de **taille** d'un programme qui est le nombre total d'objets (opérateurs ou opérandes) nécessaires à l'écriture de ce programme, ainsi que le **vocabulaire** employé qui est le nombre de tels objets distincts; à l'exemple d'Halstead [Halstead77, Christensen81], on peut également calculer plusieurs indicateurs qui reposent sur les mêmes mesures brutes et sur un modèle qui reste à justifier (longueur estimée en fonction du seul vocabulaire, volume, effort, difficulté, niveau du langage); il y a plusieurs choix possibles pour la définition des opérandes et des opérateurs d'un langage donné; le nôtre a été guidé naturellement par l'environnement Mentor dans lequel les outils de mesure ont été développés; en effet, les programmes étant, sous Mentor, représentés par un arbre syntaxique abstrait, les opérandes sont tous les nœuds terminaux de cet arbre, à l'exception des noms des procédures appelées, tandis que les opérateurs en sont les nœuds non terminaux auxquels on ajoute les noms des procédures appelées. La figure 1 indique la structure d'arbre associée à quelques constructions courantes, et les statistiques correspondantes.

FOR I:=1 to N do X:=X+1;

(6 opérandes, 4 opérateurs)



IF A>1 THEN B:=0 else WRITE(TRUC,MACHIN);

(6 opérandes, 6 opérateurs)

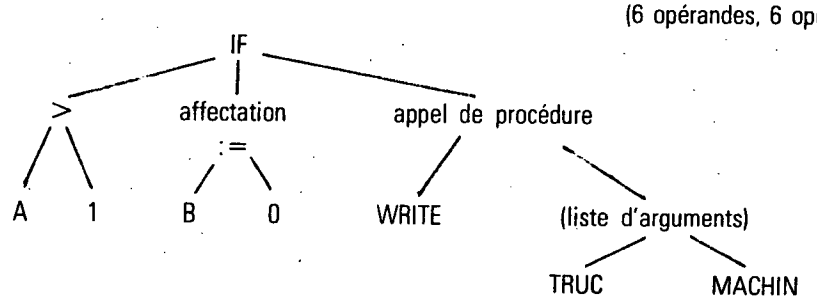


Figure 1

N.B. : Ces mesures relatives à l'arbre syntaxique (ou arbre opérateurs/opérandes) ont été collectées sur le seul code exécutable des programmes, par homogénéité avec d'autres mesures qui n'ont de sens que sur cette partie du code.

c - le **nombre cyclomatique** du graphe de contrôle (ou organigramme) du programme; introduit par MacCabe dans l'analyse des programmes [McCabe76], cet indicateur a ensuite été largement utilisé dans les études de complexité, seul ou joints à d'autres [Hansen78, Zolnowski77]; il mesure le nombre de branchements existant dans un programme et nous en avons considéré deux variantes selon que l'on distingue ou non les différents branchements dûs à une condition composée dans un test IF.

D'autres mesures moins classiques ont également été recueillies dans le but d'une comparaison globale. Il s'agit des mesures suivantes :

d - la **profondeur** de l'arbre syntaxique du programme; contenant à la fois les notions de profondeurs liées à l'imbrication des instructions et des expressions arithmétiques, cette mesure est susceptible de fournir un indicateur de complexité à comparer aux indicateurs classiques;

e - les **niveaux d'imbrication** des instructions de contrôle (instructions générant un branchement : boucles, tests et éventuels Goto), dont on peut déduire deux mesures de complexité : le niveau d'imbrication maximum et le niveau d'imbrication moyen; une information complémentaire également susceptible d'être utilisée est recueillie : la distribution des niveaux d'imbrications sur l'ensemble des instructions de contrôle;

f - des mesures portant sur l' **utilisation des variables** : nombre de références à chaque variable locale, globale ou transmise en argument, dont l'on déduit le nombre total de variables de chaque type utilisées et les taux de référence à ces variables; bien que manifestement différente par nature de l'information de complexité portée par les autres indices, l'information contenue dans la façon dont les différents types de variables sont utilisées nous semble intéressante à croiser avec les dits indices.

Ainsi qu'on le verra au paragraphes 3.2 et 4.1, la variable **taille** joue un rôle prépondérant, étant fortement corrélée à la plupart des autres variables; c'est pourquoi les différentes analyses ont porté successivement sur le fichier de mesures brutes (ou fichier initial), puis sur le fichier dit normalisé dans lequel toutes les variables fortement corrélées à la taille sont divisées par la valeur de cette taille.

2.2 - Echantillons traités

Les mesures ont été prises sur un échantillon de blocs Pascal (programmes, procédures ou fonctions) d'origines diverses : projets Croap (développement de Mentor, Flip et autres outils de programmation) et Verso (conception de bases de données) de l'INRIA, programmes statistiques de l'auteur, et un module extrait d'un méta-compilateur ; au total, 1258 blocs ont été utilisés pour une première étude sur les mesures classiques, et 921 l'ont été pour l'analyse globale sur l'ensemble des mesures citées plus haut.

La répartition des blocs suivant leur origine figure sur la table 1.

	Echantillon 1 (1258)	Echantillon 2 (921)
Projet Verso	119	119
Programmes statistiques (AS)	25	25
Méta–compilateur	211	211
Projet Croap		
(Mentor	527	236
Metal	114	102
Flip	168	149
GK)	94	79

Table 1

Notons que les mesures ont toutes été prises au niveau du bloc Pascal, et non du programme; l'information apportée concerne donc la structure des blocs considérés individuellement et on ne saurait y trouver de renseignements sur la complexité modulaire.

3 – Résultats descriptifs simples portant sur l'ensemble des variables mesurées

Les résultats présentés dans ce paragraphe sont, pour la plupart, obtenus à partir de l'échantillon 2, dans un souci d'homogeneité. Ce choix est légitime, vue la grande stabilité observée dans les chiffres d'un échantillon à l'autre.

3.1 - Description sommaire des variables

Les résultats de base portant sur l'ensemble des variables sont donnés dans la table 2.

Les figures 2 à 9 représentent les distributions des variables les plus caractéristiques.

On notera sur la figure 2, la distribution de la taille (au sens opérateurs et opérandes utilisés) des blocs Pascal considérés dans les différents groupes constituant la population totale. Nous reviendrons sur l'interprétation qui peut en être faite au paragraphe 6, qui traite des comportements des différents groupes de programmes mesurés. Sur les figures suivantes, sont successivement représentées la distribution de la variable brute (intervenant dans le fichier initial), puis celle de la même variable divisée par la taille (intervenant dans le fichier normalisé). La comparaison des histogrammes donne certaines indications sur l'influence de la taille sur les autres variables, nous y reviendrons au paragraphe 3.2.

Variables	Moyenne	Ecart-type	Minimum	Maximum
Nombre d'instructions (Nin)	13.2	19.1	0	190
Nin/NN	0.19	0.05	0.00	0.33
Taille (=NN)	77.7	130.4	2	1337
Vocabulaire (Voc)	26.1	21.4	2	192
Nb. d'opérandes distincts (Ond)	14.4	16.8	1	169
Nb. d'opérateurs distincts (Otd)	11.7	6.7	1	56
Nb. total d'operandes (OnT)	34.4	59.9	1	659
Nb. total d'operateurs (OtT)	43.3	71.7	1	796
VoçNN	0.54	0.21	0.06	1.00
Ond/NN	0.25	0.09	0.04	0.62
Otd/NN	0.28	0.15	0.01	0.80
Nb. cyclomatique médian (cmd)	4.4	6.3	1	60
Nb. cyclomatique maximum (cmx)	4.6	6.6	1	63
cmd/NN	0.08	0.06	0.00	0.50
Profondeur de l'arbre syntaxique (Prf)	7.1	3.4	1	25
Prf/NN	0.20	0.14	0.00	0.62
Niv. d'imbrication maximum (Imx)	1.5	1.5	0	9
Niv. d'imbrication moyen (Imn)	1.1	0.9	0	5.9
Nombre de variables utilisées (Nva)	2.6	4.0	0	35
Nva/NN	0.04	0.05	0.00	0.33
Taux de référence aux variables (Trf)	0.3	0.3	0	1

Table 2

Quelques remarques générales s'imposent :

- moins de 1% des blocs étudiés comportent plus de 50 instructions exécutables (environ 1 page de listing pour un programme Fortran...), ce qui indique une structure très modulaire des programmes analysés;

- 1/3 de blocs ne contient aucune instruction de contrôle, et, dans les 2/3 restant, le niveau d'imbrication maximum ne dépasse pas 8, et n'est qu'exceptionnellement (2%) supérieur à 5 (notons que les histogrammes des figures 7 et 8 ne portent que sur les 622 blocs -soit 67.5%- qui contiennent au moins une instruction de contrôle);

- sur la figure 9, le nombre de variables considéré est le nombre de toutes les variables (locales, globales ou passées en argument) effectivement utilisées (référencées) dans le bloc; on constate qu'environ 30% des blocs ne font référence à aucune variable, que 60% en utilisent de 1 à 5 (dont 23% une seule), et 10% seulement plus de 6.

Distribution de la variable TAILLE dans différents groupes de blocs Pascal

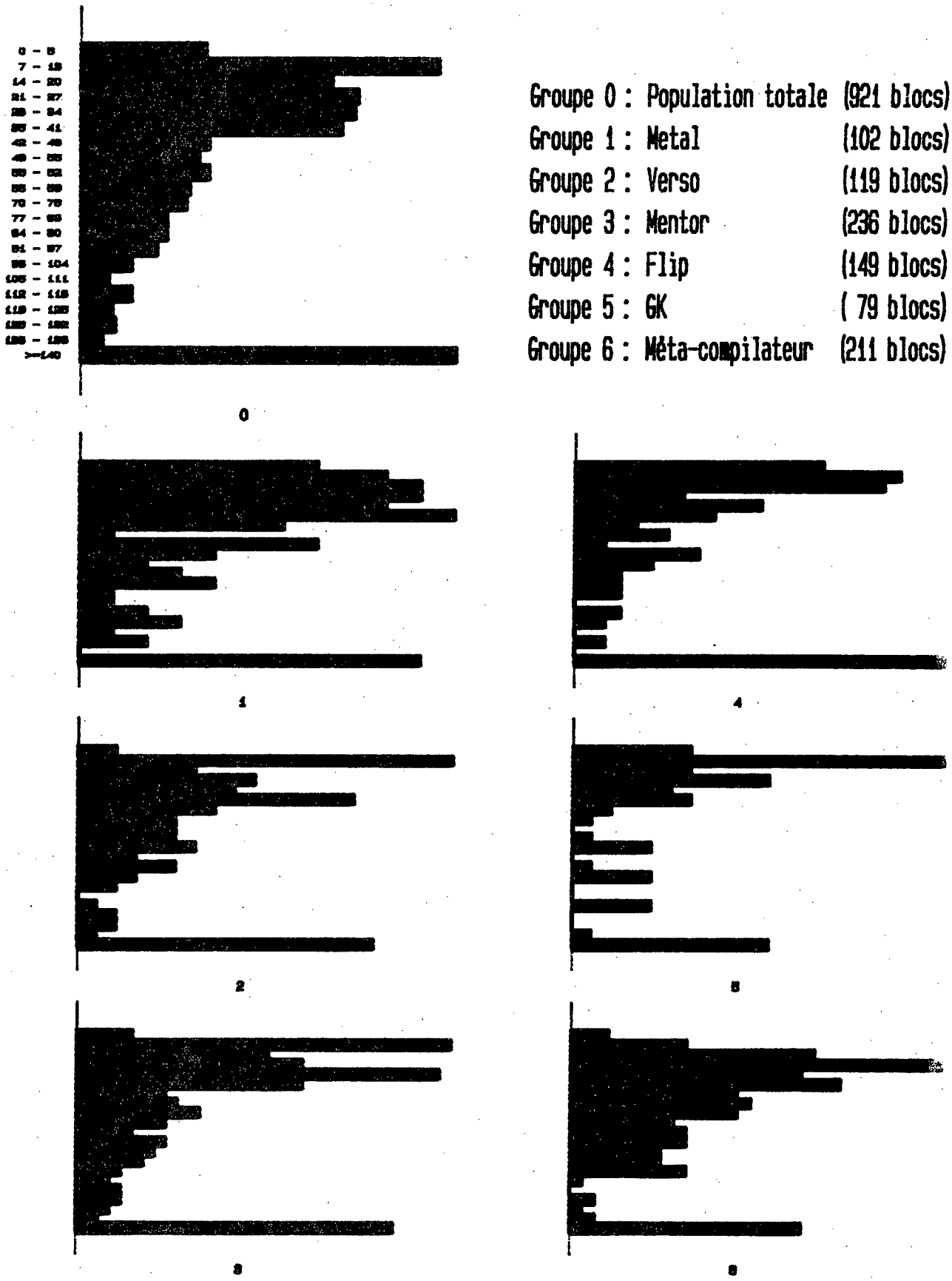


Figure 2

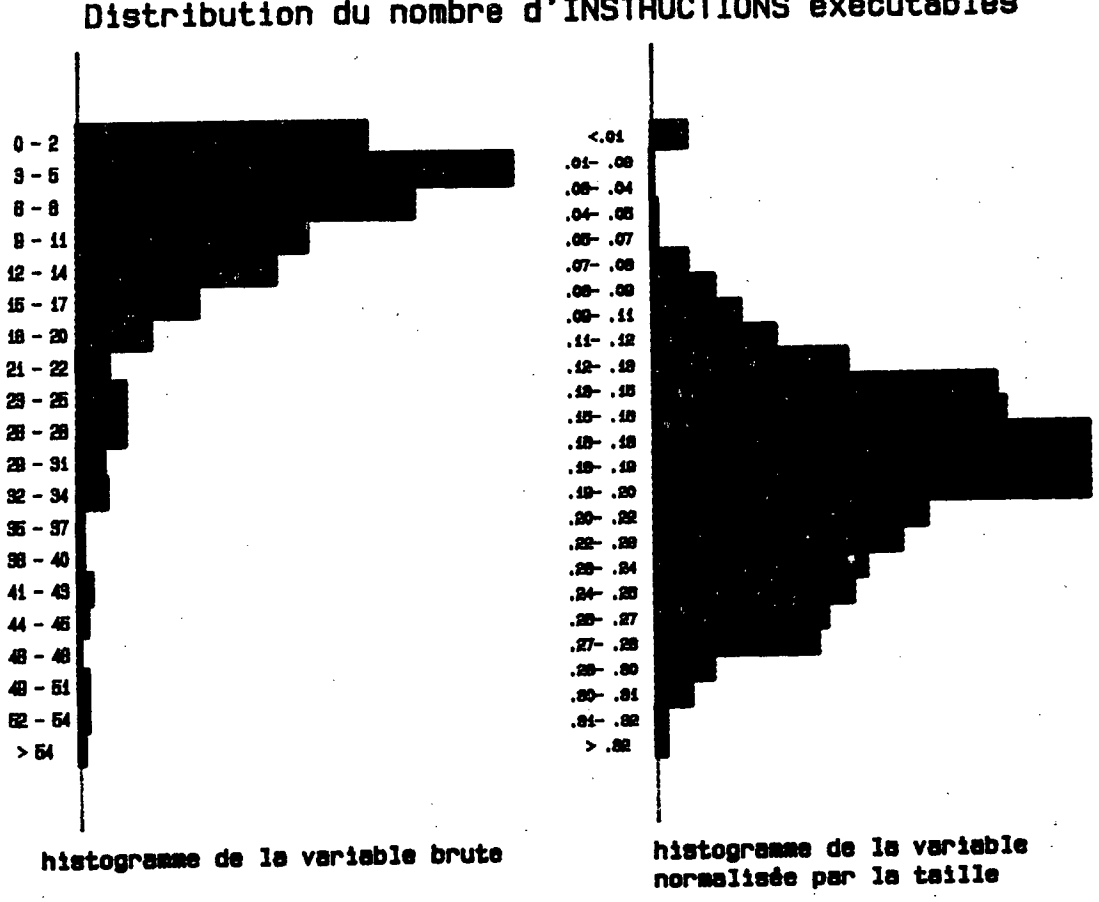


Figure 3

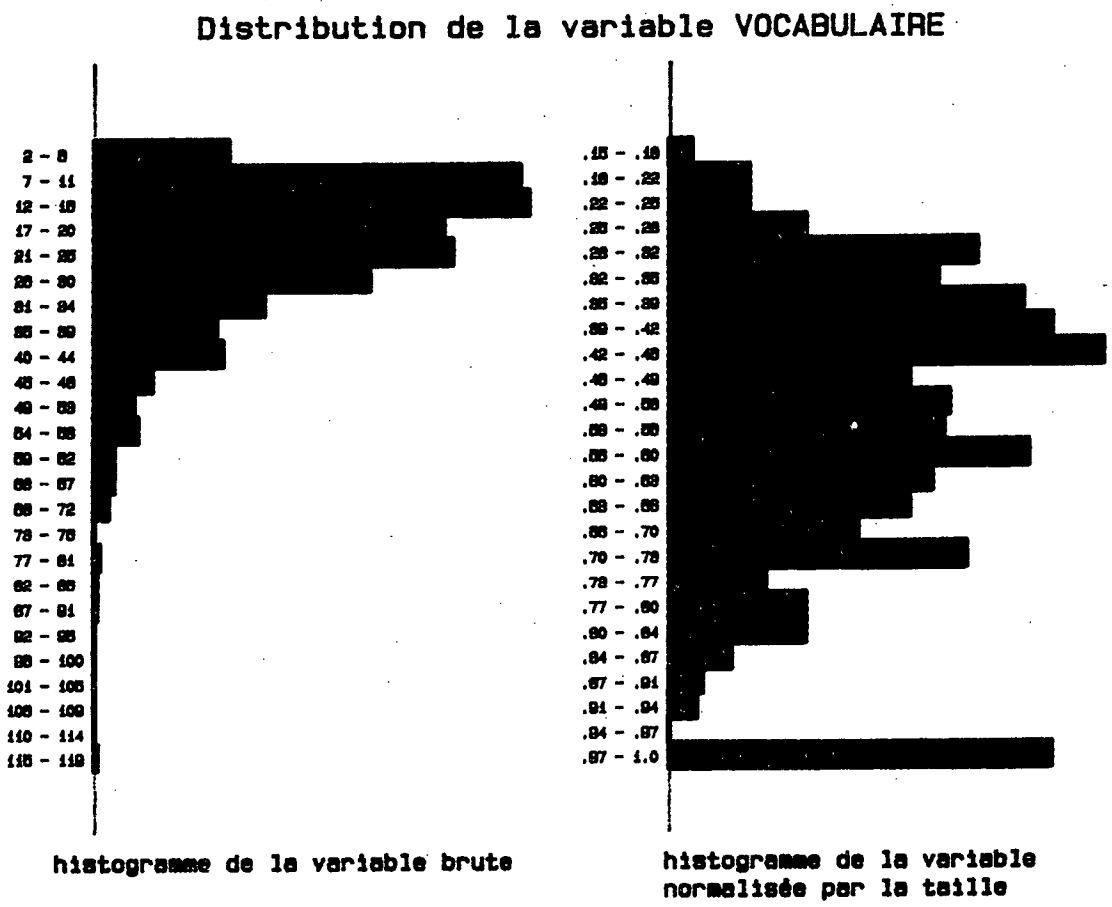


Figure 4

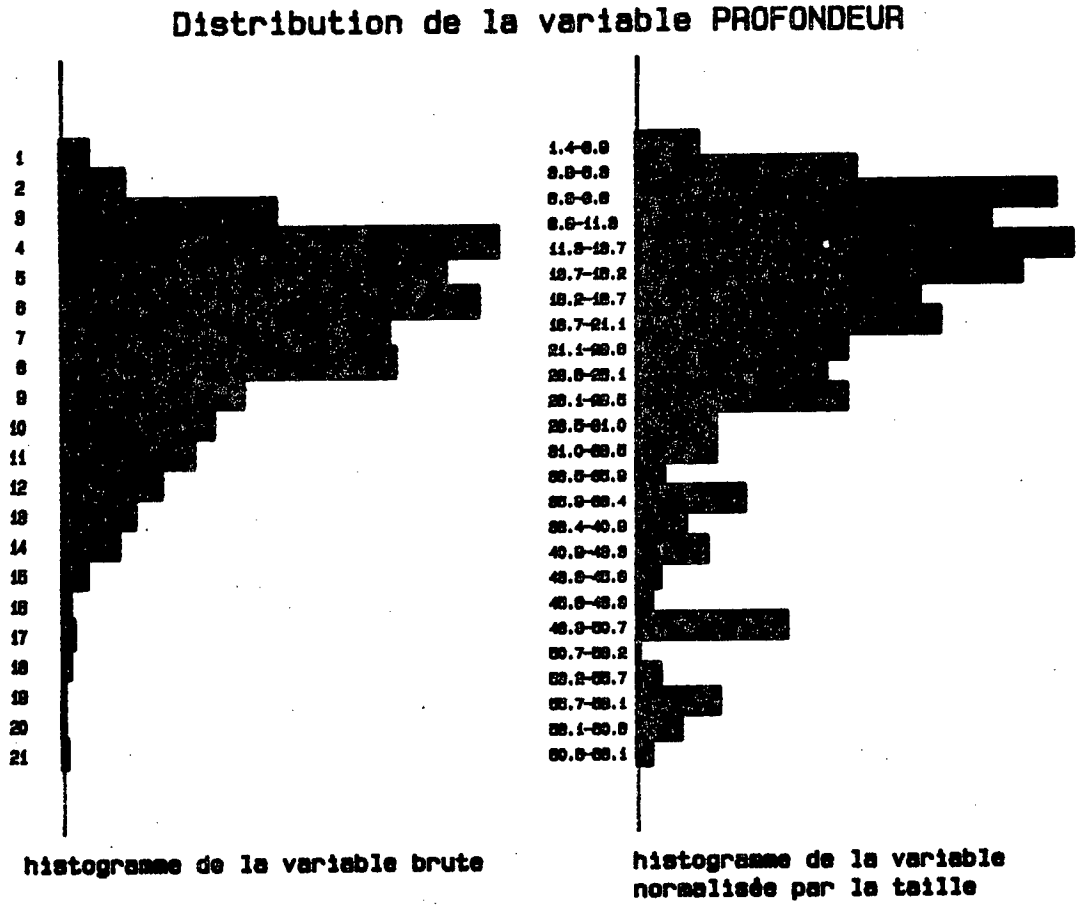
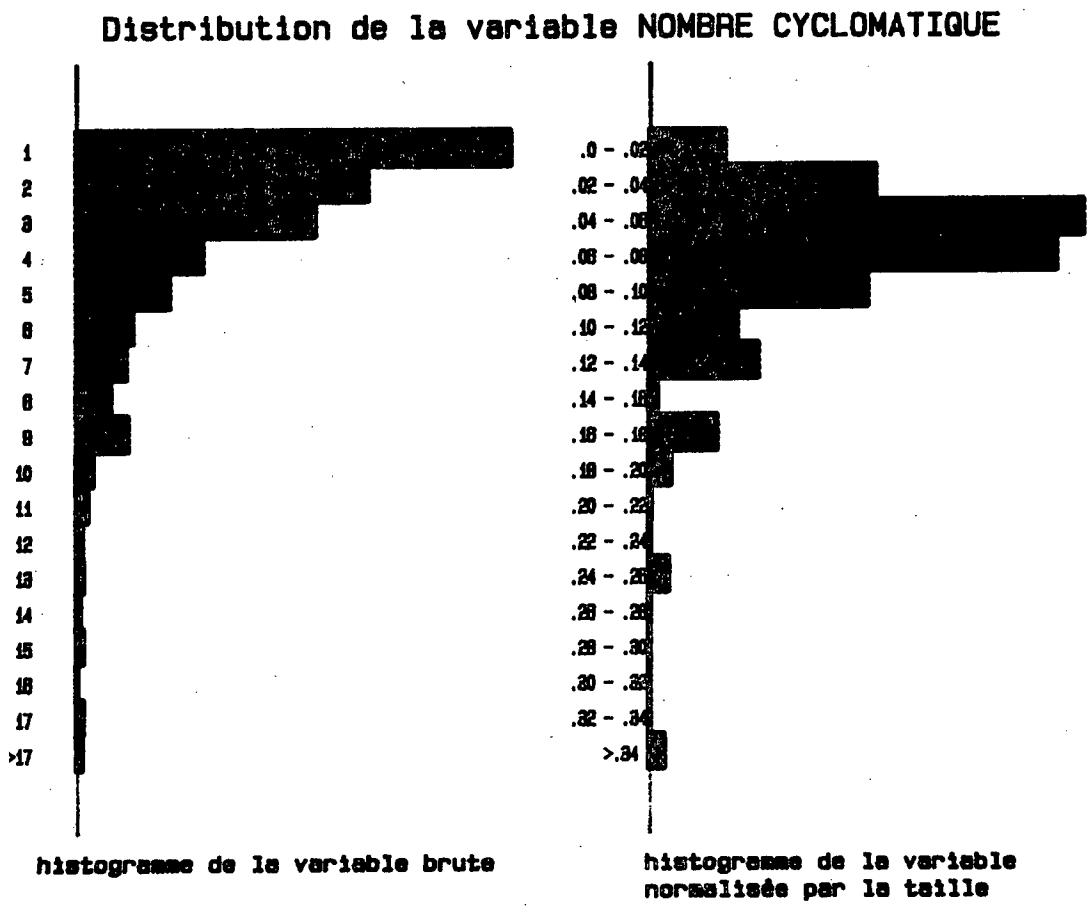


Figure 5

Figure 6



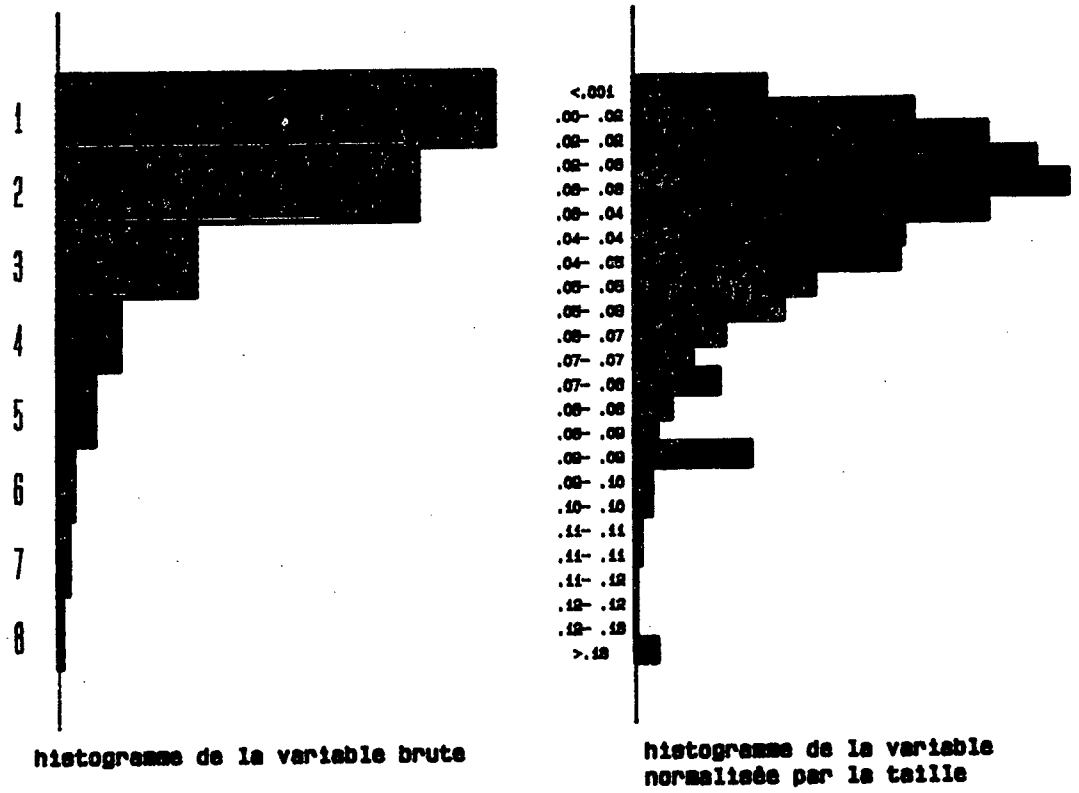
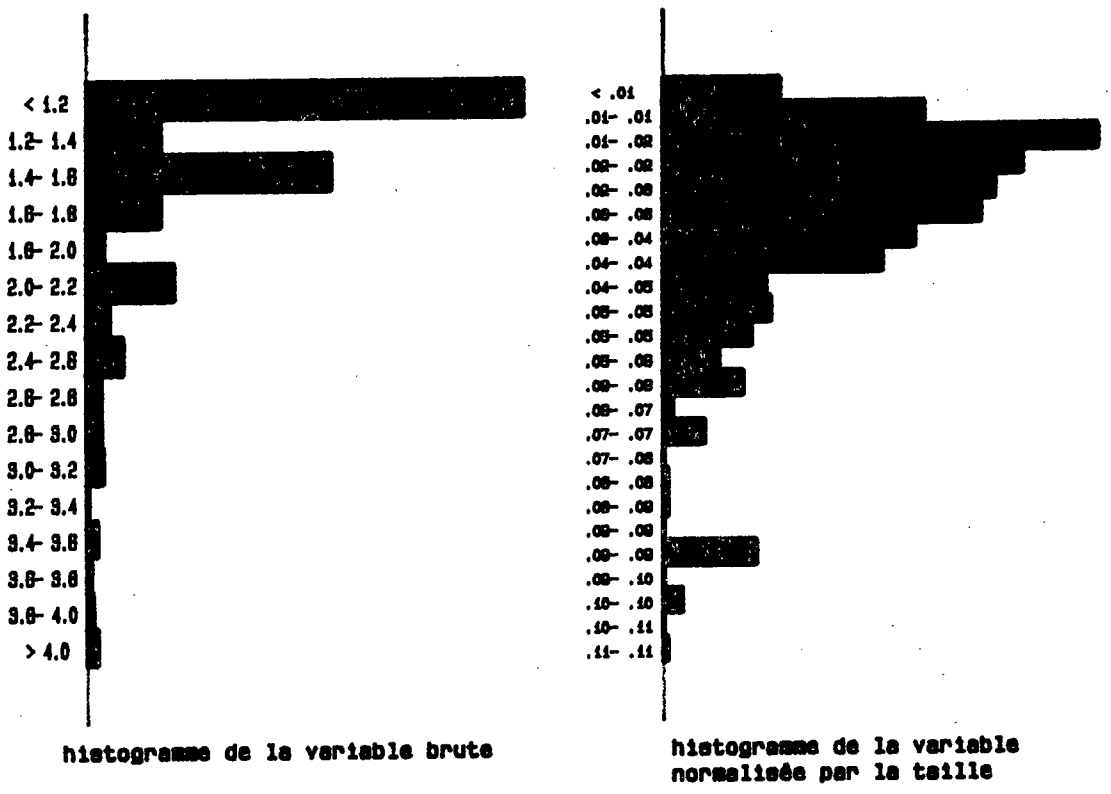


Figure 7

Figure 8



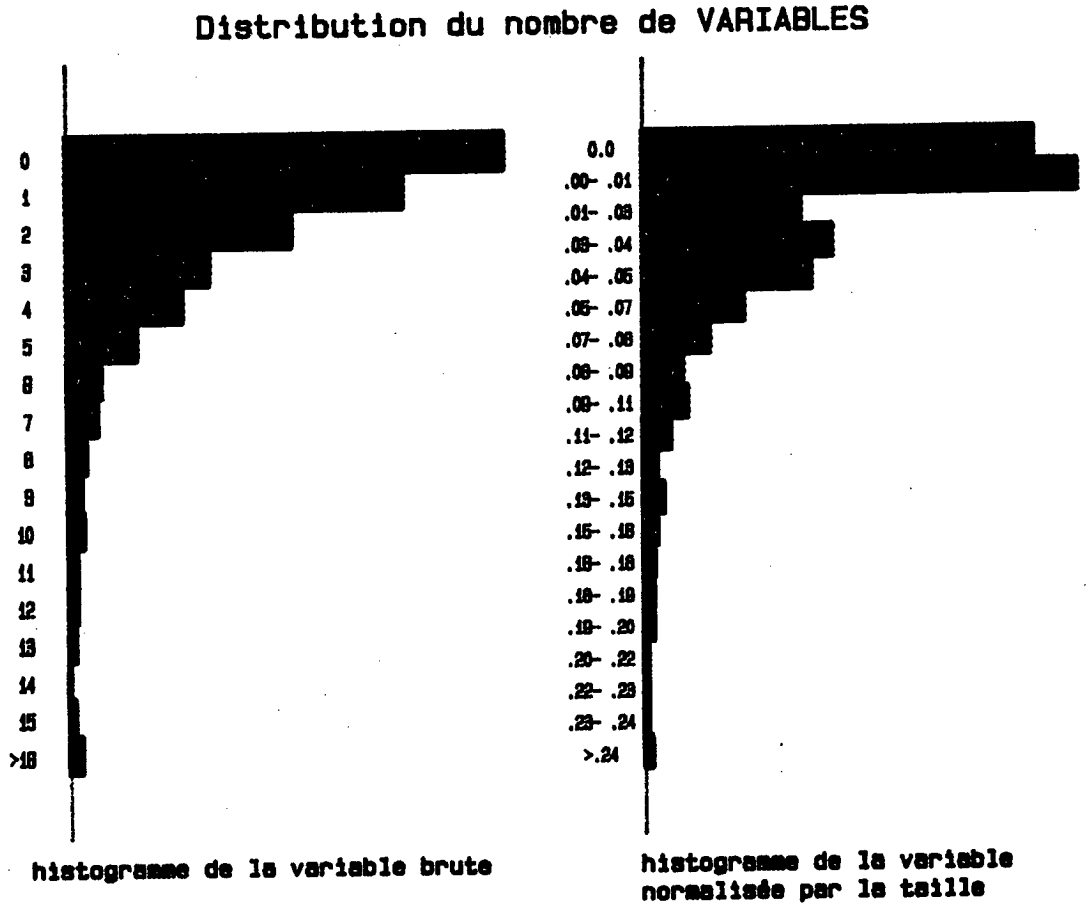


Figure 9

3.2 - Corrélations entre les variables

Les tables 3 et 4 donnent respectivement les coefficients de corrélations relatifs aux variables initiales puis aux variables normalisées.

	Nin	NN	Voc	Ond	Otd	OnT	OtT	cmd	cmx	Prf	Imx	Imn	Nva	Trf
Nin	1.00													
NN	0.95	1.00												
Voc	0.87	0.84	1.00											
Ond	0.84	0.84	0.97	1.00										
Otd	0.68	0.59	0.77	0.58	1.00									
OnT	0.93	0.99	0.82	0.83	0.54	1.00								
OtT	0.96	0.99	0.85	0.84	0.62	0.96	1.00							
cmd	0.77	0.74	0.81	0.75	0.69	0.68	0.77	1.00						
cmx	0.77	0.74	0.80	0.75	0.69	0.68	0.78	1.00	1.00					
Prf	0.56	0.50	0.60	0.45	0.80	0.46	0.53	0.60	0.60	1.00				
Imx	0.55	0.46	0.57	0.42	0.76	0.43	0.48	0.62	0.46	0.83	1.00			
Imn	0.52	0.44	0.56	0.42	0.76	0.41	0.46	0.59	0.40	0.82	0.97	1.00		
Nva	0.60	0.56	0.52	0.42	0.60	0.55	0.55	0.51	0.52	0.49	0.46	0.40	1.00	
Trf	-0.04	-0.05	0.02	-0.00	0.07	-0.05	-0.05	-0.04	-0.04	-0.01	-0.03	-0.02	0.17	1.00

Matrice des corrélations (fichier initial)
Table 3

On remarque les forts coefficients de corrélation de la variable taille avec la plupart des autres variables; ils sont en particulier supérieurs à ceux qui lient le nombre d'instructions (Nin) aux autres variables; la taille d'un programme (définie comme somme des nombres d'opérateurs et d'opérandes utilisés) rend en effet compte, à la fois de la quantité d'instructions à exécuter, et de la complexité due aux expressions et aux listes d'arguments.

On note également le comportement des nombres totaux d'opérateurs et d'opérandes : très corrélés l'un à l'autre et tous deux à la taille, leurs corrélations respectives avec l'ensemble des autres variables sont extrêmement proches, et également proches des corrélations de la taille avec les mêmes variables; en conséquence, nous les retirerons des traitements ultérieurs dans lesquels ils ne pourraient apporter qu'une information redondante.

De la même façon, on supprimera dans la suite la seconde variante du nombre cyclomatique (cmx), qui tient compte des cas où la condition d'un test IF est composée; un rapide examen des données initiales confirme que ces cas sont très rares.

Dans la table 4 (matrice des corrélations du fichier normalisé), les variables divisées par la taille sont identifiées par le nom qui leur a été attribué dans la table 3, suivi de la lettre N; les variables dont le nom ne se termine pas par N sont celles qui, manifestement peu liées à la taille, n'ont pas été normalisées.

	NinN	NN	VocN	OndN	OtdN	cmdN	PrfN	Imx	Imn	NvaN	Trf
NinN	1.00										
NN	-0.17	1.00									
VocN	0.08	-0.56	1.00								
OndN	-0.04	-0.43	0.81	1.00							
OtdN	0.15	-0.53	0.93	0.54	1.00						
cmdN	-0.06	-0.24	0.61	0.51	0.56	1.00					
PrfN	-0.10	-0.46	0.86	0.61	0.86	0.61	1.00				
Imx	0.33	-0.14	-0.03	-0.12	0.03	0.20	-0.04	1.00			
Imn	0.34	-0.19	0.06	-0.05	0.12	0.22	0.04	0.96	1.00		
NvaN	0.04	-0.09	0.02	0.05	-0.00	-0.16	-0.08	-0.06	-0.06	1.00	
Trf	0.08	-0.05	0.00	0.05	-0.02	-0.17	-0.14	-0.01	-0.00	0.68	1.00

Matrice des corrélations (fichier normalisé)
Table 4

On peut également observer, en se reportant aux distributions des variables normalisées ou non (figures 3 à 9), que les liaisons existant entre les variables et la taille sont de natures diverses; on complète cette observation par celle des figures 10 à 14, qui représentent les graphiques de densité des principales variables brutes et normalisées croisées avec la taille.

Les principales remarques que l'on peut faire sont les suivantes :

- la relation évidente qui existe entre taille et nombre d'instructions exécutables est pratiquement linéaire (figure 10);

- les trois variables vocabulaire (figure 11), profondeur (figure 12) et nombre cyclomatique (figure 13) sont aussi fortement corrélées à la taille, mais leurs liaisons respectives avec cette dernière sont de natures différentes : la forme hyperbolique des graphiques relatifs aux valeurs normalisées par la taille de ces variables indique qu'elles sont liées à cette taille par une fonction de poids négligeable par rapport à elle aux voisinages de l'origine et de l'infini; par ailleurs, on peut, moyennant quelques hypothèses sur l'équilibrage d'un arbre, démontrer que sa profondeur varie comme la racine de sa taille; ce résultat s'applique à nos données dans une certaine mesure, car la profondeur est sensiblement plus corrélée avec la racine carrée de la taille (r=0.68), qu'avec la taille elle-même (r=0.51). Pour les deux autres variables (vocabulaire et nombre cyclomatique), nous n'avons fait aucune étude d'ajustement, mais on pourrait, de même, penser à des fonctions logarithmes ou racines. Dans les trois cas, on distingue des plages différentes de valeurs de la taille sur lesquelles une éventuelle relation fonctionnelle est valide (i.e. où des courbes hyperboliques ressortent plus nettement) : ainsi, le nombre cyclomatique est plus directement lié à la taille lorsque celle-ci se situe entre 20 et 100, tandis que pour la profondeur, cette plage se situe entre 30 et 120 environ; dans tous les cas les relations à la taille deviennent très floues dès que celle-ci est grande.

- la figure 14 illustre la relation entre la taille et le *volume* défini par Halstead [Halstead77] de la façon suivante :

$$Volume = Taille \cdot \log_2 (Vocabulaire) ;$$

la forte corrélation existant entre la taille et le vocabulaire entraîne évidemment une liaison des plus étroites entre taille et volume. Le coefficient de corrélation entre taille et volume vaut 1, tandis que les coefficients de corrélation du volume avec autres variables sont pratiquement identiques à ceux de la

taille avec ces mêmes variables. C'est la raison pour laquelle nous n'avons pas pris le volume en compte dans l'ensemble des analyses.

- sur la figure 15, on observe les relations entre les différentes variables concernant le nombre des instructions de contrôle (nombre cyclomatique) et leur répartition (niveaux moyen et maximum d'imbrication). Les niveaux moyen et maximum d'imbrication sont fortement corrélés (r=0.97), ce qui indique un certaine uniformité dans la distribution des imbrications et la redondance probable qui existe entre les deux mesures. Quant au nombre cyclomatique qui est a peu près égal au nombre d'instructions de contrôle, contrairement à ce qu'on aurait pu attendre, il est peu lié à la structure d'imbrication; les programmes comportant de nombreux branchements ne sont donc pas nécessairement trés imbriqués, et réciproquement.

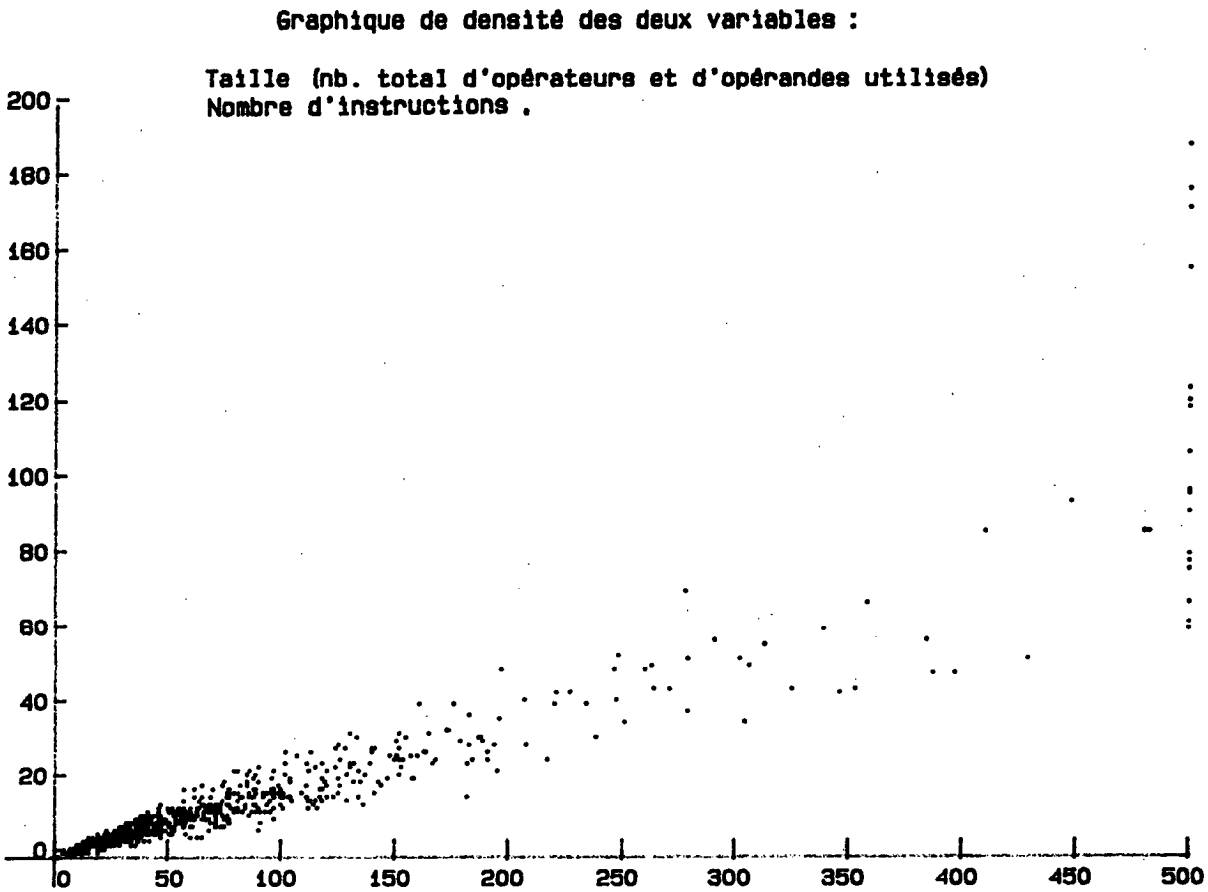


Figure 10

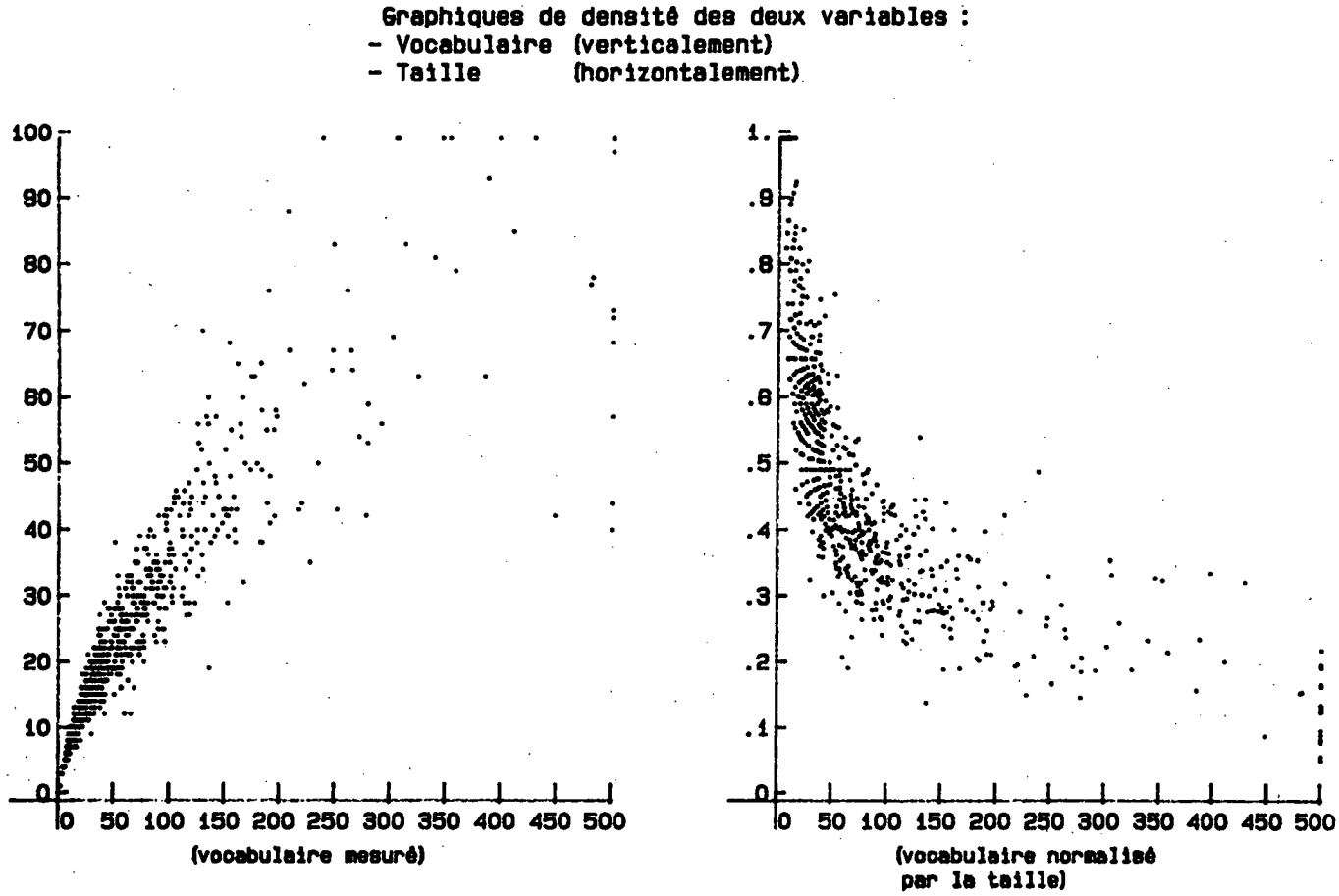
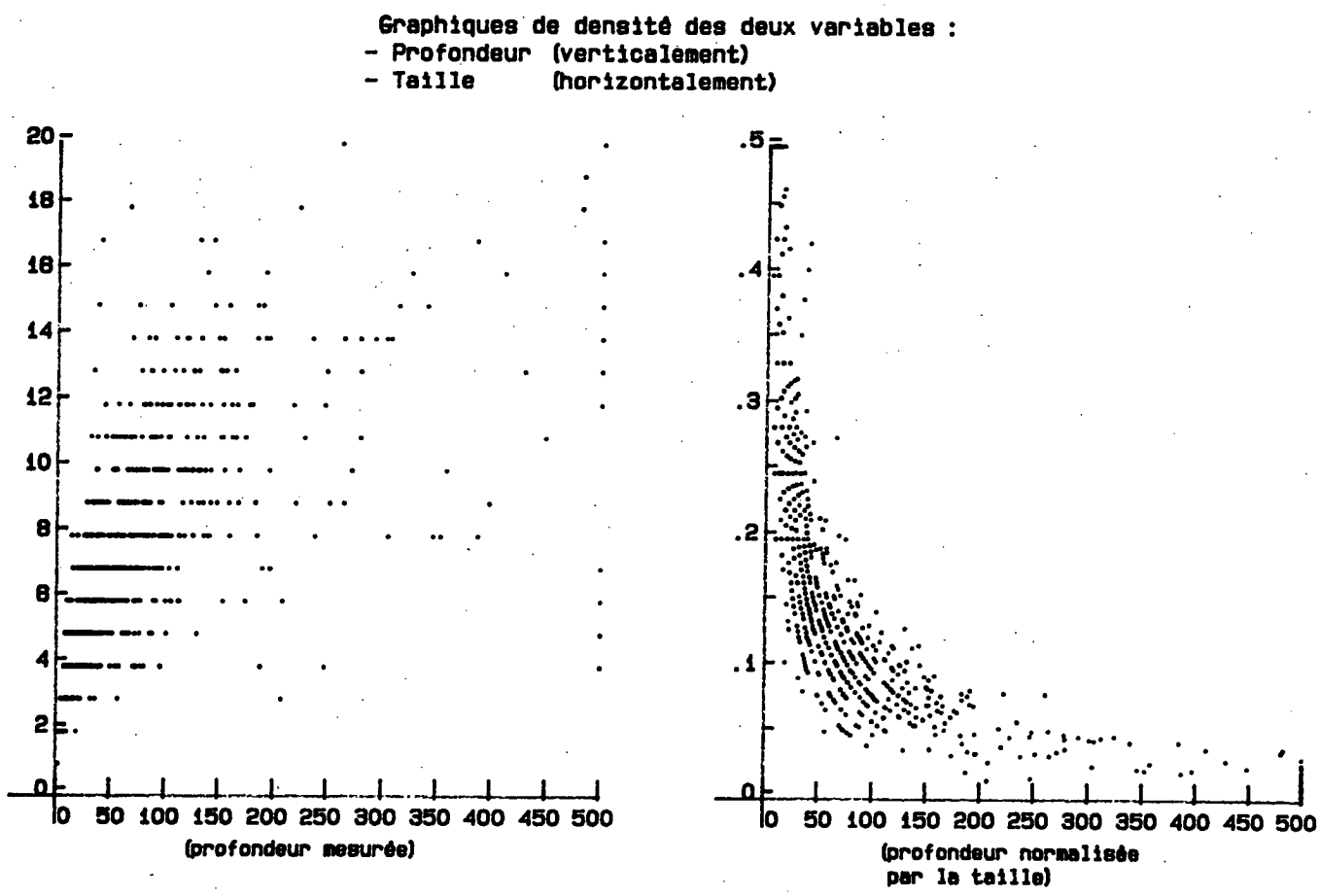


Figure 11

Figure 12



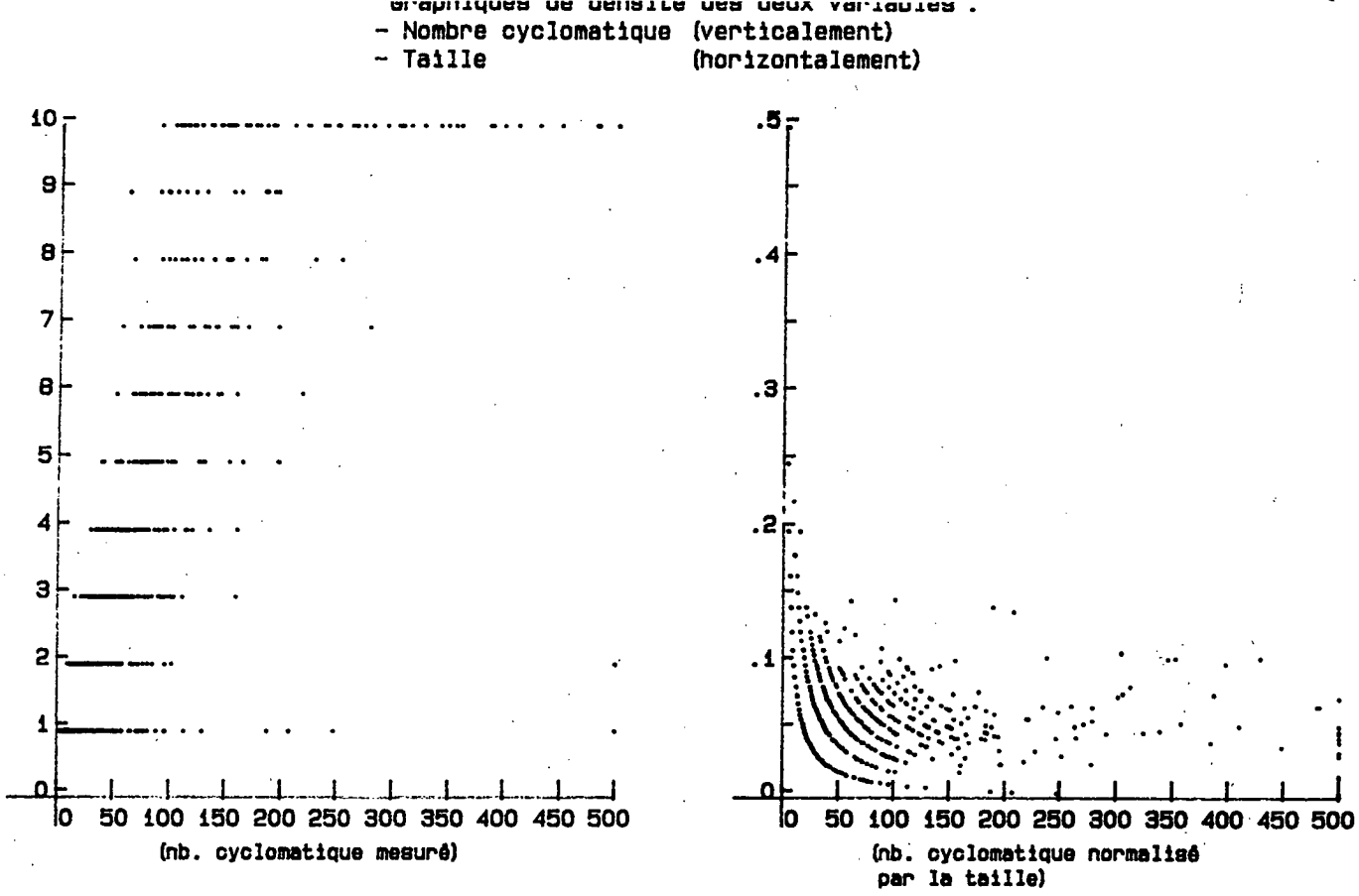
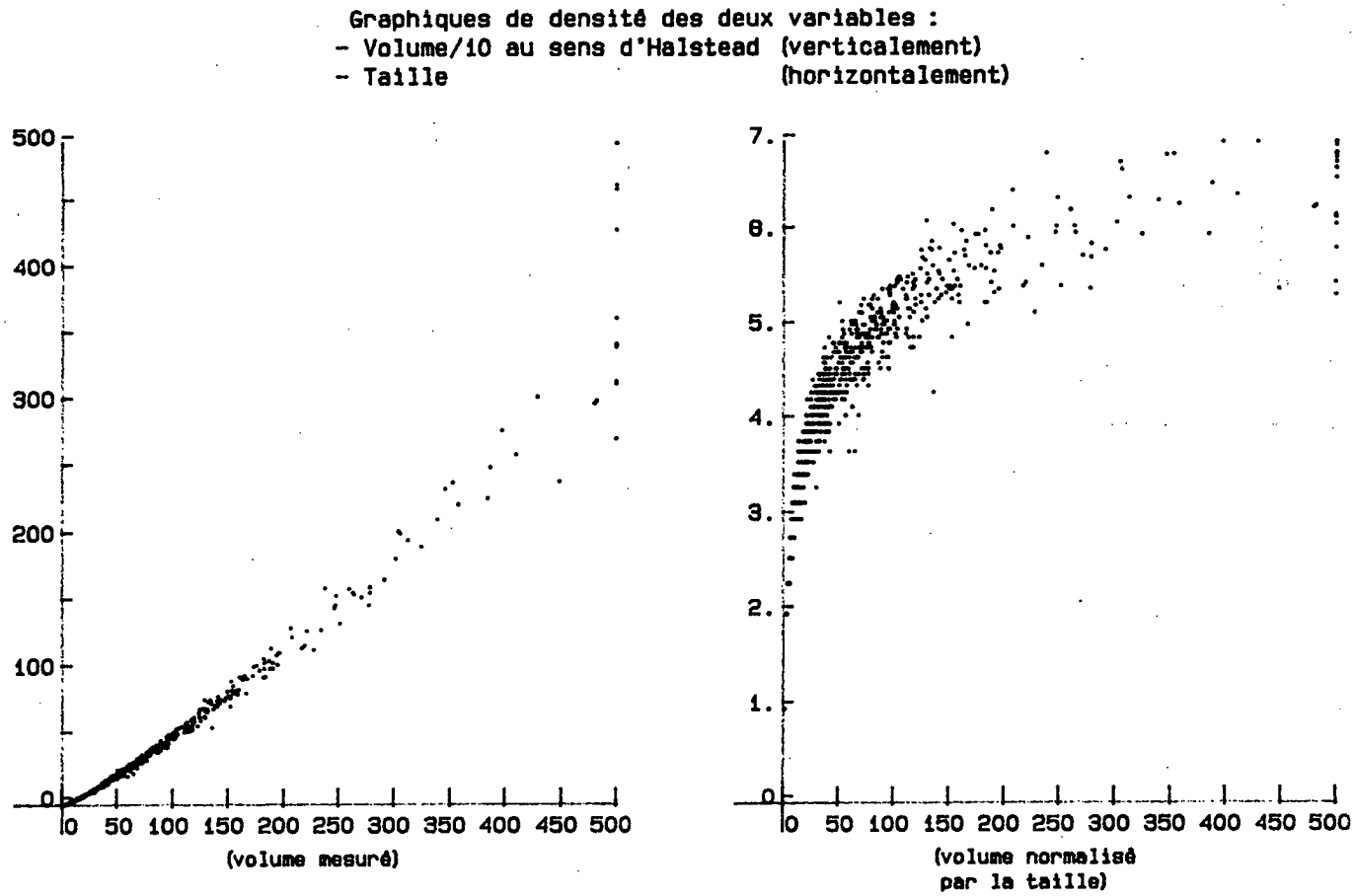


Figure 13

Figure 14



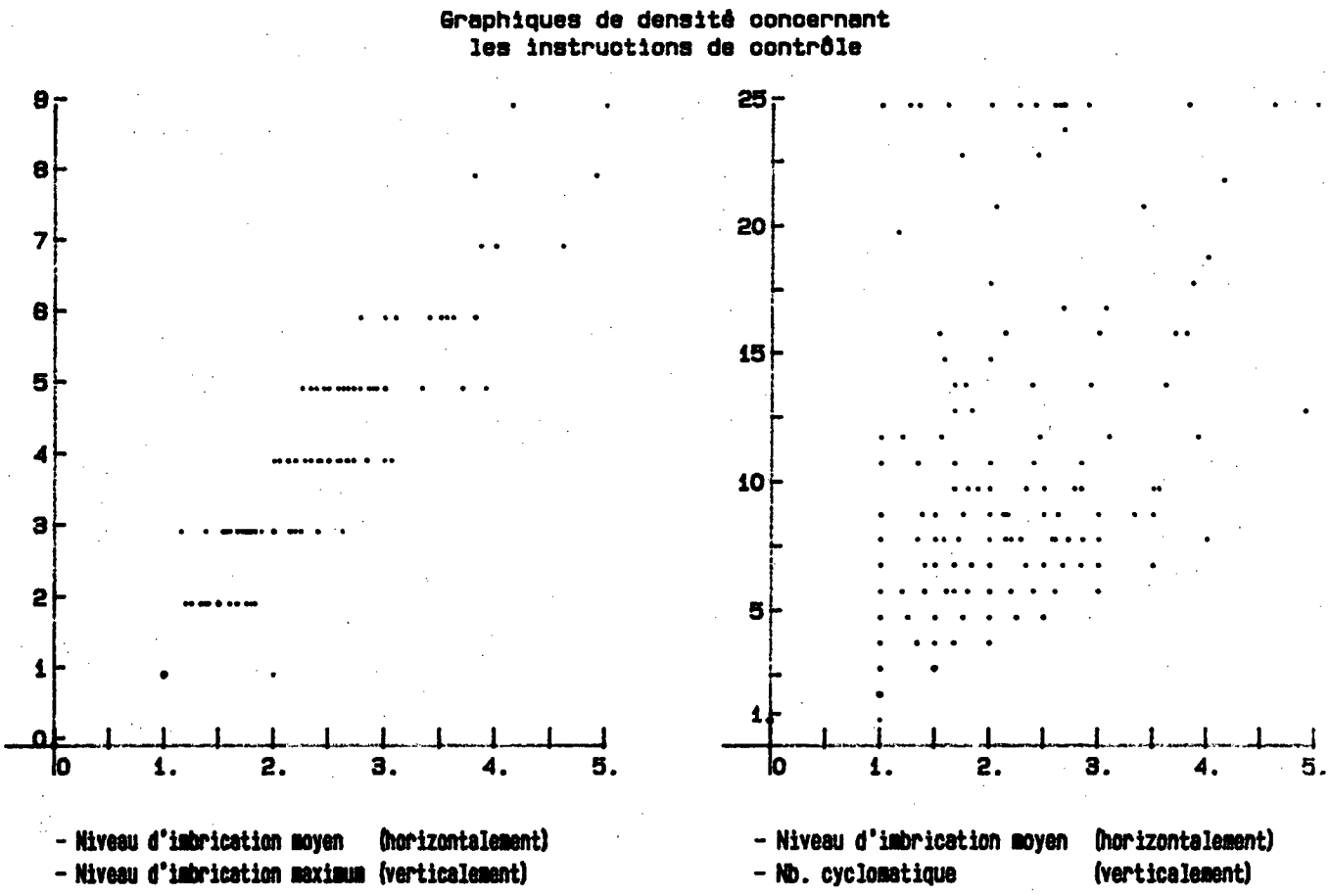

Graphiques de densité concernant
les instructions de contrôle
9
8
7
6
5
4
3
2
1
0
0
1.
2.
3.
4.
5.
- Niveau d'imbrication moyen (horizontalement)
- Niveau d'imbrication maximum (verticalement)
25
20
15
10
5
1
0
1.
2.
3.
4.
5.
- Niveau d'imbrication moyen (horizontalement)
- Nb. cyclomatique (verticalement)

Figure 15

4 – Analyse de l'information mutuelle apportée par les différentes mesures

Pour analyser globalement l'ensemble des mesures, nous avons utilisé des méthodes statistiques multidimensionnelles qui permettent précisément d'étudier un grand nombre de variables simultanément et non prises une à une ou deux à deux comme il a été fait au paragraphe 3. Les deux méthodes utilisées ici sont *l'Analyse des Correspondances* [Benzecri73, Schroeder78] et la classification simultanée de deux ensembles mis en correspondance, ou *Classification Croisée* [Govaert77 et 83]. On trouvera dans les annexes 1 et 2 une brève introduction à chacune de ces méthodes et surtout l'explication des sorties qu'elles produisent et que l'on peut interpréter. Leur description complète ainsi que les développements théoriques qui les supportent se trouvent dans les références indiquées ci-dessus.

Ces techniques d'analyse permettent de décrire un tableau de données rectangulaire dont les lignes représentent traditionnellement des sujets et les colonnes des variables mesurées sur ces sujets; ainsi, si l'on mesure p variables sur n sujets, le tableau de données résultant est le tableau

$$T = (t_{ij})_{i=1,n \text{ et } j=1,p},$$

tel que t_{ij} est la valeur prise par la $j_{ème}$ variable sur le $i_{ème}$ sujet. Dans notre contexte, les sujets sont toujours les blocs Pascal (soit 921 sujets) et les variables sont prises dans l'ensemble des variables mesurées décrites au paragraphe 2.1.

4.1 - Influence globale de la taille

Il a déjà été observé dans les paragraphes précédents que les liaisons entre la taille et les autres variables sont fortes, mais aussi qu'elles peuvent être de natures différentes. Considérée maintenant comme une variable parmi les autres, la taille conserve sa place prépondérante. En effet, dans les analyses multidimensionnelles, elle est representée par un point de très fort poids du fait de sa valeur absolue importante et surtout de son influence indirecte à travers toutes les autres variables. Ainsi, dans les deux premières analyses (annexes 3 et 4) qui prennent en compte toutes les variables (sauf celles relatives aux références aux variables), dont la taille (annexe 3) ou sans elle (annexe 4), et ce sur les fichiers initial (fig. 18 et 20) puis normalisé (fig. 19 et 21), on observe que le 1^{er} axe est toujours presque entièrement expliqué par la taille, soit directement, soit par l'intermédiaire de la variable Nin (nb. d'instructions exécutables). Sur les figures 18 et 19, on observe également l'écrasement de l'ensemble des variables normalisées en présence de la taille. Les remarques precédentes restent valables pour les deux analyses suivantes (annexes 5 et 6) dans lesquelles sont seules actives les variables classiques (i.e. nombre d'instructions et statistiques sur opérateurs et opérandes).

Note sur l'estimation de la taille en fonction du vocabulaire :

Parmi les relations proposées par Halstead dans le cadre de la Science du Logiciel *(Software Science)* [Halstead77], il en est une qui a été souvent utilisée dans la littérature, il s'agit de l'équation de la taille *(length equation),* qui permet d'estimer la taille d'un programme à partir des nombres d'opérateurs et d'opérandes distincts nécessaires son écriture; si n_1 *(resp.* n_2*)* sont les nombres d'opérateurs (resp. opérandes) distincts nécessaires, on a :

$$Taille_{\ estimée} = n_1.\log_2 n_1 + n_2.\log_2 n_2$$

D'autre part, ainsi qu'on l'a signalé au paragraphe 2.1-b, les comptages relatifs aux opérateurs et opérandes peuvent se faire de deux façons différentes selon qu'on ne tient compte que du code exécutable ou aussi des zones de déclarations. La figure 16 illustre la qualité de l'estimateur de la taille selon que l'on tienne compte ou non des zones de déclarations dans les procédures de comptage. L'estimation obtenue en tenant compte des déclarations est légèrement meilleure que l'autre du point de vue du coefficient de correlation. Par ailleurs, un ajustement des moindres carrés avec terme constant, donne les résultats suivants :

$$Taille_{estimée} = 1.02 \; . \; Taille + 52.9 \qquad (1)$$

$$Taille_{estimée} = 0.78 \; . \; Taille + 48.1 \qquad (2)$$

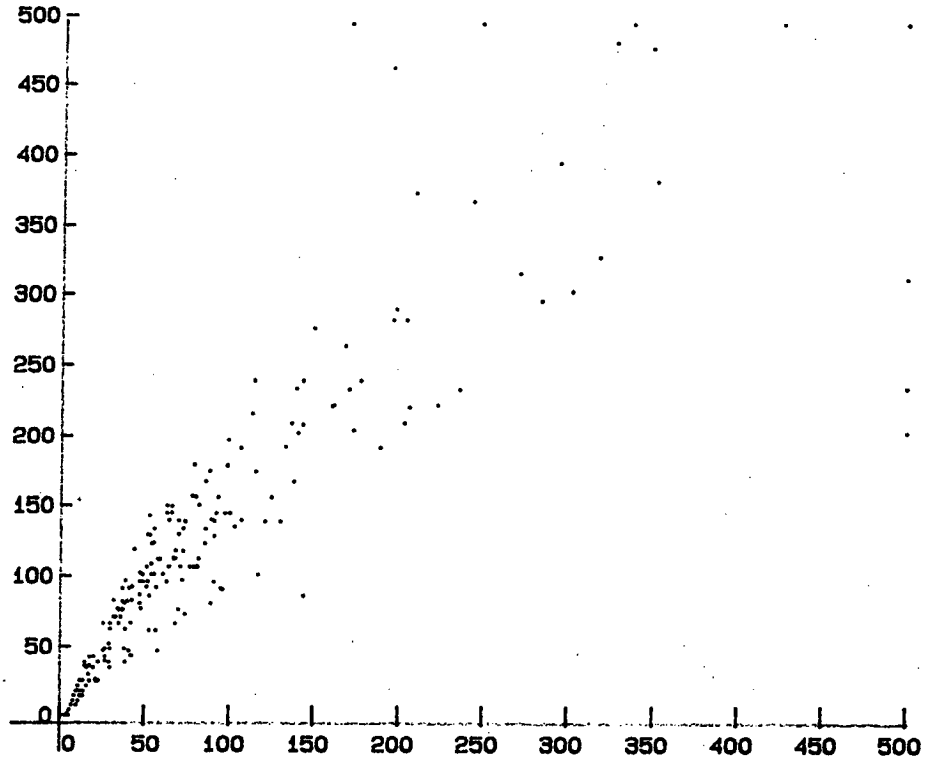
où les équations 1 et 2 sont respectivement relatives aux cas avec et sans les déclarations. Dans les deux cas, on observe un terme constant important qui limite l'intérêt de l'estimateur proposé.

Comparaison de la TAILLE
(nb. total d'opérateurs et d'opérandes)
avec son estimation à partir du vocabulaire

Mesures prises en tenant compte des zones de déclarations

Coefficient de corrélation

0.81



(Horizontalement : taille observée, verticalement : taille estimée)

Mesures prises en ne tenant compte que du code exécutable

Coefficient de corrélation

0.87

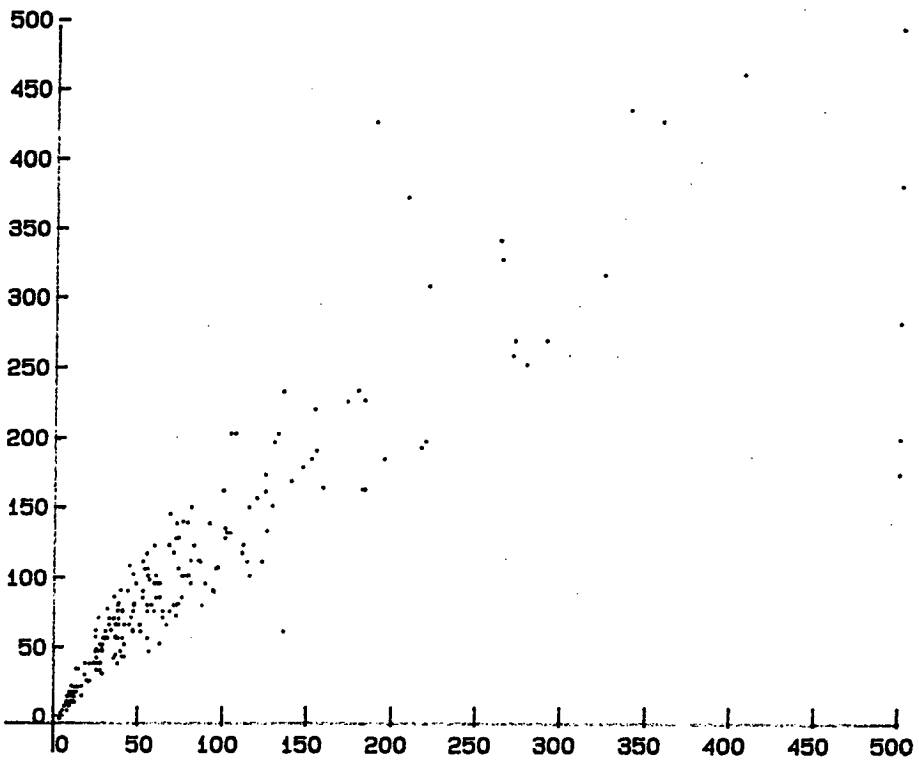


Figure 16

4.2 - Description générale de l'ensemble des variables

L'interprétation des analyses présentées dans les annexes 3 à 7 fournit, outre les remarques propres à la taille déjà indiquées, les résultats suivants :

- l'information relative à l'utilisation des variables, à savoir le nombre (Nva) de variables utilisées, ou référencées, et le taux de référence (Trf = nb. de références faites à une variable / Nva), est pratiquement indépendante de l'ensemble des autres mesures (cf. annexe 7); ceci apparaissait déjà dans les matrices de corrélations du paragraphe 3.2. En conséquence, les deux mesures Nva et Trf ont été supprimées comme variables actives dans toutes les autres analyses (annexes 3 à 6); on les a cependant représentées comme variables illustratives et on constate qu'elles se relient alors à l'ensemble des autres variables par l'effet général de taille : Nva tend à croître avec NN ou Nin, tandis que Trf varie en sens inverse;

- les nombres d'opérateurs et d'opérandes distincts ou non contiennent une information propre. Initialement prélevées de façon à pouvoir calculer les indicateurs d'Halstead, ces mesures ont quand même figuré individuellement dans les traitements statistiques; les nombres totaux d'opérateurs et d'opérandes utilisés n'ont pas été retenus dans les analyses multidimensionnelles compte tenu de leur très forte corrélation avec la taille. Par contre, les nombres d'opérateurs et d'opérandes distincts ont été traités et on note qu'il ont des comportements différents : tandis que le nombre d'opérandes distincts (Ond) est toujours lié à la taille du bloc, le nombre d'opérateurs distincts (Otd), quant à lui, s'associe aux mesures de complexité structurelles; il apporte donc une information propre sur la structure des blocs;

- plusieurs variables sont supposées mesurer la complexité structurelle des blocs; il apparaît qu'en réalité, elles apportent chacune une information qui lui est propre et donnent donc ensemble une vision de la complexité sous plusieurs angles. Parmi l'ensemble de nos variables, les mesures considérées classiquement sont le nombre d'instructions (Nin), les mesures d'Halstead (NN, Voc, Otd, Ond) et le nombre cyclomatique (cmd). Les résultats de l'annexe 5 (analyses où seules sont actives les mesures classiques) indiquent que, lorsque ces seules mesures sont prises en compte, on distingue deux types de complexité : celle mesurée par cmd, et celle mesurée par les valeurs respectives de Ond et Otd; la projection des autres mesures de complexité (Prf, Imn et Imx) comme variables supplémentaires dans l'analyse des Correspondances montre que leur comportement se rapproche de celui de Otd. Les résultats des analyses dans lesquels les deux types de mesures sont simultanément actives (annexe 3) confirment cette interprétation. On peut finalement déduire que la complexité structurelle des blocs est issue de deux facteurs : la quantité d'instructions de contrôle (cmd) et la façon dont elles sont imbriquées (mesurée directement par Imn et Imx, et contenue dans Prf qui tient, en outre, compte de la profondeur des expressions).

5 – Distribution statique des opérateurs du langage

Toutes les variables que nous avons présentées jusqu'ici relèvent de l'évaluation de la complexité des programmes; il est un autre type de mesures statiques fréquemment évoquées dans la littérature [Al-Jarrah79, Brookes82], à savoir la distribution des instructions d'un langage.

La table 5 donne les les fréquences d'utilisation des principaux opérateurs du langage Pascal (i.e. représentant 90% de l'ensemble des utilisations d'opérateurs) dans l'échantillon 2 et dans les quatre sous-échantillons.

La figure 17 montre les histogrammes de l'utilisation des opérateurs particuliers que sont les instructions du langage. Les résultats de la table 5 et de la figure 17 sont de ceux susceptibles d'intéresser les concepteurs de compilateurs ou, plus généralement, de générateurs de tables syntaxiques.

	Total	Croap	Verso+AS	Méta–comp.
NB BLOCS	766	411	144	211
PROGRAMME	16	5	10	1
PROCEDURE	520	259	97	164
FUNCTION	230	147	37	46
Opérateurs :				
LEXP	8034	3406	3390	1238
CALL	5336	2724	1595	1017
ASS	4975	2904	1221	850
INDEX	3100	1054	1713	333
LSTAT	2700	1358	867	475
DOT	2081	907	605	569
IF	1911	1265	425	221
VIDE	1423	598	610	215
EQL	957	591	243	123
COLON	857	597	173	87
LCST	852	592	173	87
PLUS	648	198	339	111
WRITE	545	308	151	86
FOR	513	129	323	61
NEQ	470	321	97	52
UPSTEP	457	110	290	57
WRITELN	305	61	183	61
MINUS	287	160	93	34
SETOF	244	223	6	15
AND	202	126	52	24
LELEM	172	154	5	13
WHILE	159	79	25	55
FORMAT	150	25	108	17
OR	132	92	24	16
LSS	128	77	40	11
NOT	125	80	17	28
GOTO	125	101	24	0
GTR	123	58	46	19
CASE	106	83	7	16
IN	106	86	5	15
WITH	103	66	8	29
LVARBL	102	66	8	28
LCOLON	101	80	7	14
LDEFID	98	0	98	0
MULT	92	12	71	9
LEQ	90	75	8	7
UMINUS	78	70	3	5
LABSTAT	76	65	11	0
REPEAT	72	36	21	15
DEF	69	0	69	0

Table 5

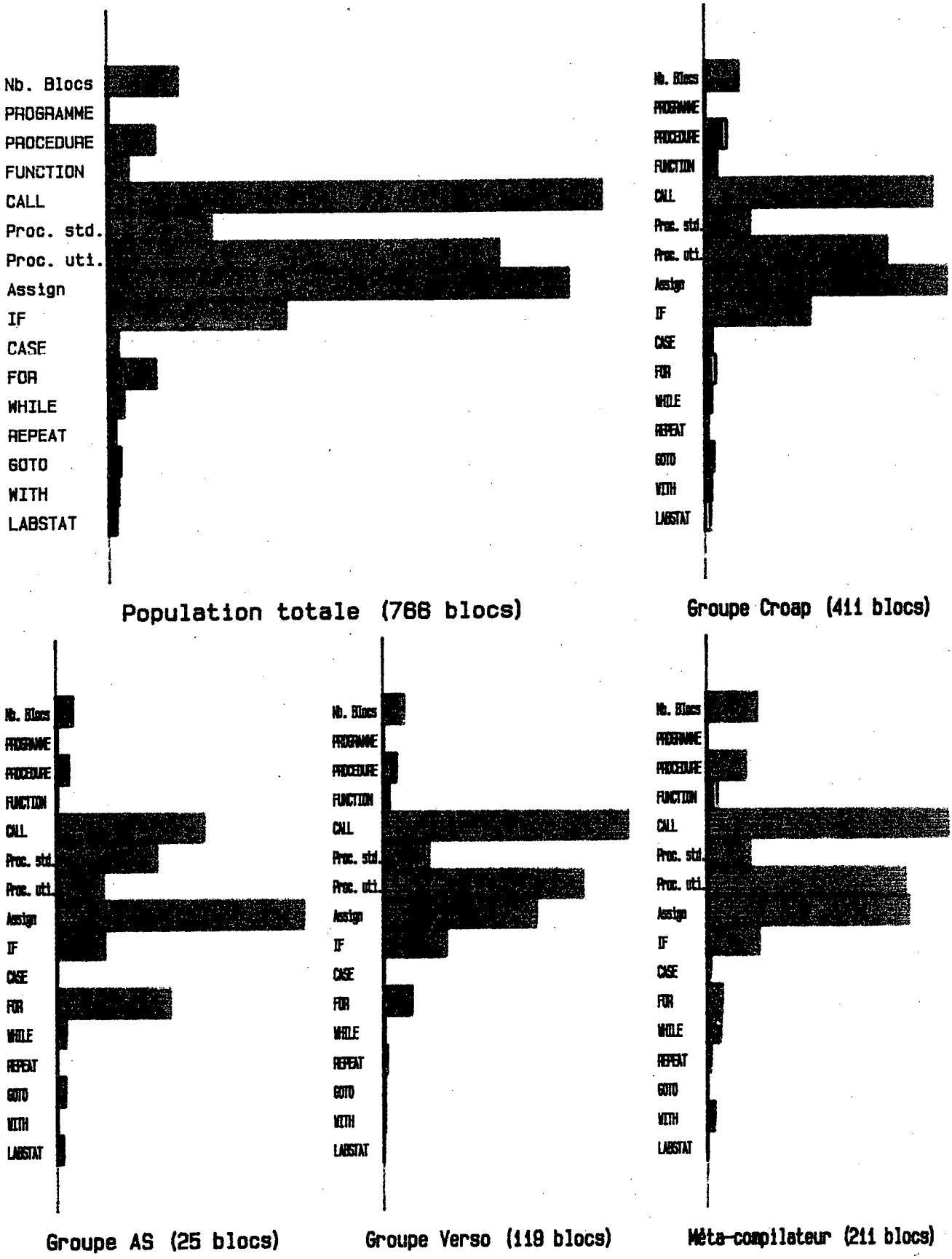

Distribution des instructions dans différents groupes de blocs Pascal
Nb. Blocs
PROGRAMME
PROCEDURE
FUNCTION
CALL
Proc. std.
Proc. uti.
Assign
IF
CASE
FOR
WHILE
REPEAT
GOTO
WITH
LABSTAT
Population totale (786 blocs)
Groupe Croap (411 blocs)
Groupe AS (25 blocs)
Groupe Verso (119 blocs)
Méta-compilateur (211 blocs)

Figure 17

6 – Typologie générale des blocs Pascal étudiés

Les analyses effectuées mettent en évidence quelques différences dans l'utilisation que font de Pascal les programmeurs des groupes ayant fourni les programmes mesurés. Les résultats concernés apparaissent sur les figures 2 et 17 ainsi que dans l'ensemble des annexes.

Notons, pour ces origines, quelques caractéristiques externes :

- groupe 1 : programmes statistiques, un seul programmeur, plus ou moins traduction de Fortran;
- groupe 2 : projet Verso, plusieurs programmeurs, système de gestion de bases de données;
- groupe 3 : source de l'environnement de programmation Mentor (à l'exception de Metal), plusieurs programmeurs;
- groupe 4 : programmes de Mentor relatifs à Metal, un seul programmeur;
- groupe 5 : source du système Flip de génération de transparents, un seul programmeur;
- groupe 6 : programmes utilitaires divers, un seul programmeur;
- groupe 7 : analyseur syntaxique d'un méta-compilateur, un seul programmeur.

Les programmeurs uniques des groupes 1, 4 et 7 sont différents entre eux et différents du programmeur des groupes 5 et 6. L'ensemble des groupes 3, 4, 5 et 6 constitue le projet Croap. Dans les annexes, les origines des blocs figurent avec les numéros de groupes affectés ci-dessus.

Quelques remarques que l'on peut déduire des différentes analyses :

- la distribution de la taille (au sens du nombre total d'opérateurs et d'opérandes utilisés) des blocs dans les différents est assez variable; on peut comparer, par exemple, les histogrammes relatifs au programmeur de Metal et à celui du méta-compilateur : il y a plus de petits blocs dans Metal qui est donc écrit probablement de façon plus modulaire;

- également variable est l'utilisation que font les différents programmeurs des instructions du langage Pascal; les fréquences respectives des boucles FOR et WHILE, par exemple, révèlent des styles différents et une prédilection pour les boucles FOR de la part des anciens programmeurs Fortran;

- au long des analyses de Correspondances présentées dans les annexes 3 à 7, on note que, en tenant compte globalement des toutes les variables, les deux groupes 1 et 6 se distinguent par leur styles particuliers de programmation (ou plutôt d'utilisation de Pascal), puis le groupe 7 dans une moindre mesure; en gros, le groupe 1 comporte de longs programmes assez fortement imbriqués, tandis que le groupe 6 contient plutôt des blocs courts dont la complexité vient du nombre d'opérandes distincts utilisés.

7 – Conclusions

Nous avons tenté, par une analyse statistique approfondie, de faire le point sur l'information qu'apportent diverses mesures de complexité du logiciel. Ayant mesuré un millier de blocs Pascal, et ayant traité ces mesures par des techniques statistiques descriptives classiques et multidimensionnelles, nous sommes arrivés aux conclusions suivantes :

- les blocs se différencient d'abord les uns des autres par leur taille;

- parmi les mesures de complexité proposées par Halstead dans son modèle de *Science du Logiciel,* nous avons trouvé un intérêt propre aux mesures brutes (taille, nombres d'opérandes et d'opérateurs distincts), qui sont des caractéristiques de l'arbre syntaxique d'un bloc; par contre, les mesures dérivées du modèle appliqué à nos données, n'apportent pas d'élément convaincant : l'équation de la longueur n'est pas vérifiée, le volume n'exprime d'autre information que celle exprimée par la taille, et le vocabulaire dissimule les deux types d'information propre apportés par les nombres distincts d'opérateurs et d'opérandes;

- à taille donnée, leur complexité logique n'est que partiellement mesurée par le nombre cyclomatique de leur graphe de contrôle qui ne rend pas compte des imbrications possibles des branchements; nous proposons comme mesure alternative, la profondeur de l'arbre syntaxique qui mesure à la fois l'imbrication des instructions et celle des expressions; cet indicateur présente, de plus, l'avantage de se calculer dans un même parcours d'arbre que les statistiques portant sur les opérateurs et les opérandes, si ces derniers sont effectivement définis comme les nœuds d'un arbre de syntaxe abstraite;

- enfin, on a pu dégager différents modes d'utilisation du langage Pascal, ou styles de programmation typiques, identifiables à l'aide des mesures prélevées.

De toutes façons, la supériorité d'une approche multidimensionnelle pour l'évaluation de la complexité du logiciel apparaît clairement; toutes les mesures évoquées ont leur contenu propre, qui correspond à un des aspects de ce que l'on appelle complexité dans le langage courant.

Bibliographie

[Al-Jarrah79]
M. M. Al–Jarrah and I. S. Torsun
An Empirical Analysis of Cobol Programs
Vol. 9, No. 5, 1979. pp. 341–359

[Benzecri73]
J.–P. Benzecri et Coll.
L'Analyse des Données
Dunod, Paris , 1973.

[Brookes82]
G. R. Brookes, I. R. Wilson and A. M. Addyman
A Static Analysis of Pascal Program Structures
Software–Practice and Experience, Vol. 12, 1982. pp. 959–963

[Christensen81]
K. Christensen, G. P. Fitsos and C. P. Smith
A Perspective on Software Science
IBM Syst. Journal Vol. 20, No. 4, 1981. pp. 372–387

[Cook82]
M. L. Cook
Software Metrics: An Introduction and Annotated Bibliography
Software Engineering Notes, ACM Sigsoft, Vol. 7, No. 2, April 1982. pp. 41–60

[Curtis80]
B. Curtis
Measurement and Experimentation in Software Engineering
Proceedings of the IEEE, Vol. 68, No. 9, September 1980. pp. 1144–1157

[Friedman81]
H. P. Friedman
Statistical Methods in Computer Performance Evaluation
in Experimental Computer Performance and Evaluation D. Ferrari and M. Spadoni (eds.)
North–Holland Publ. Co., 1981. pp. 79–103

[Govaert77]
G. Govaert
Algorithme de classification d'un tableau de contingence
Premières Journées Internationales : Analyse de Données et Informatique, Versailles, Septembre 1977. pp. 487–500

[Govaert83]
G. Govaert
Classification croisée
Thèse d'Etat, Université Paris VI, Institut de Programmation, Juin 83.

[Halstead77]
M. H. Halstead
Elements of Software Science
Elsevier North–Holland Inc., N.Y., 1977.

[Hansen78]
W. J. Hansen
Measurement of Program Complexity by the Pair (Cyclomatic Number, Operator Count)
SIGPLAN Notices, Vol. 13, No. 3, March 1978. pp. 29–33

[McCabe76]
T. J. McCabe
A Complexity Measure
IEEE Transactions on Software Engineering, Vol. SE–2, No. 4, December 1976. pp. 308–320

[Perlis81]
A. J. Perlis, F. G. Sayward and M. Shaw (eds.)
Software Metrics
The MIT Press, 1981.

[Schroeder78]
A. Schroeder
How Multidimensional Data Analysis Techniques can be of Help in the Study of Computer Systems
Proceedings of the CPEUG Meeting, Boston, October 1978.

[Schroeder83a]
A. Schroeder
Outils de mesures de programmes Pascal
Globule 4 – Bulletin du groupe de travail Génie Logiciel de l'AFCET–Informatique , Mai 1983.

[Schroeder83b]
A. Schroeder
Integrated Program Measurement and Documentation Tools
Rapport de Recherche INRIA, No. 227, Juillet 1983.

[Zolnowski77]
J. M. Zolnowski and D. B. Simmons
Measuring Program Complexity
Digest of Papers of Fall COMPCON77, IEEE Cat. No. 77CH1258-3C, 1977. pp. 336-340

Annexe 1

Analyse des Correspondances

Due à J.P. Benzecri [Benzecri73], cette méthode d'analyse statistique de données multidimensionnelles s'apparente à l'analyse des tables de contingence, à l'analyse des corrélations canoniques et à l'analyse en composantes principales.

Très brièvement parlant, son objet est de donner d'un ensemble situé dans un espace à plusieurs dimensions la meilleure représentation approchée possible dans un espace de faible dimension.

Ainsi qu'il a été dit au paragraphe 4, cette méthode permet de décrire un tableau de données rectangulaire

$$T = (t_{ij})_{i=1,n \text{ et } j=1,p},$$

tel que t_{ij} est la valeur prise par la $j_{ème}$ variable sur le $i_{ème}$ sujet. Si les sujets sont les blocs Pascal mesurés (soit 921 sujets) et si les variables considérées sont p variables prises dans l'ensemble de mesures décrit au paragraphe 2.1, on peut considérer l'ensemble des blocs comme un ensemble de 921 points situés dans l'espace R^p à p dimensions, les p coordonnées d'un bloc donné étant les p résultats des mesures effectuées sur ce bloc; de façon analogue, l'ensemble des p mesures peut être considéré comme un ensemble de points dans l'espace R^{921} à 921 dimensions.

Pour chacun des deux ensembles ci-dessus, la représentation approchée de dimension k (en général, k=2,3 ou 4) fournie par l'analyse des Correspondances est sa projection sur son sous-espace E_k de dimension k d'inertie maximum. Ce sous-espace est défini par le choix d'une distance sur R^p (resp. R^{921}); la distance utilisée en analyse des Correspondances est la distance du χ^2, dont les propriétés ont deux conséquences importantes sur l'analyse :

- les deux analyses (dans R^p et R^{921}) se déduisent l'une de l'autre par des relations linéaires simples, ce qui permet d'une part, de n'en effectuer qu'une, et d'autre part de représenter simultanément les deux ensembles mis en correspondance par le tableau de données;
- sur cette représentation, deux variables sont proches l'une de l'autre si leurs distributions sur l'ensemble des blocs sont proches.

La qualité de la représentation obtenue par projection sur un sous-espace donné se mesure par le quotient

Dispersion de l'ensemble projeté / Dispersion totale de l'ensemble de points.

Cette quantité est précisément maximum pour un sous-espace d'inertie maximum. D'autre part, les sous-espaces E_k (k=1,2,...) sont construits de la façon suivante :

- E_1 est l'axe principal d'inertie de l'ensemble de points, ou *1er axe,* sur lequel la projection des points initiaux a une qualité de représentation que l'on calcule en termes de pourcentage de dispersion conservé, ou *pourcentage d'inertie,*

- E_2 est engendré par E_1 et par un axe qui lui est orthogonal, ou *2ème axe,* tel que la qualité de représentation sur E_2 est la somme de celles sur E_1 et sur ce 2ème axe,

- etc...

Finalement, une analyse des Correspondances produit une suite de représentations planes des deux ensembles mis en correspondance, ainsi que les pourcentages d'inertie associés aux axes successifs; le premier plan representé est celui engendré par les deux premiers axes d'inerties et la qualité de la représentation fournie est mesurée par la somme des pourcentages d'inertie relatifs à ces deux axes; le second est celui engendré par les deuxième et troisième axes, et, éventuellement, ainsi de suite. L'interprétation de ces graphiques est, en outre, guidée par l'examen des contributions de chaque variable initiale aux différents axes; on appelle ainsi la part propre à chaque variable dans la dispersion conservée par projection sur chaque axe.

Des variables supplémentaires (ou illustratives) peuvent également être représentées sur les graphiques. Il s'agit de colonnes du tableau de données que l'on n'a pas voulu faire figurer activement dans l'analyse (i.e. contribuer à la dispersion globale et au choix du plan de projection), mais que l'on souhaite pourtant situer par rapport aux variables actives; ce sont donc des points de R^n que l'on projette directement sur les plans principaux. Dans nos graphiques, les noms des variables actives apparaissent en caractères droits, et ceux des variables supplémentaires en italiques.

Annexe 2

Classification Croisée

Etant donnés un ensemble de points dans un espace à plusieurs dimensions et une distance sur cette espace, les méthodes dites de *classification* produisent une partition de l'ensemble optimisant un certain critère; ces méthodes se différencient les unes des autres par le critère utilisé et par la méthode employée pour l'optimiser.

La méthode de Classification Croisée que nous utilisons produit simultanément deux partitions sur les deux ensembles mis en correspondance par un tableau de données (cf. paragraphe 2.1), en optimisant le χ^2 de contingence du tableau T' obtenu en croisant deux partitions de taille (nombre de classes) fixée.

Plus précisément, si $I=(I_1,...,I_N)$ *(resp.* $J=(J_1,...,J_P)$*)* est une partition en N (resp. P) classes des n (resp. p) lignes (resp. colonnes) du tableau T initial, T' est le tableau T' à N lignes et P colonnes, tel que

$$T' = (t'_{IJ})_{I=1,N \text{ et } J=1,P}$$

où t'_{IJ} est la valeur moyenne des t_{ij} pour i dans I et j dans J.

Parmi les sorties possibles du programme, nous examinerons :

- le pourcentage de χ^2, ou de dispersion, conservé après partition du tableau T en T'; lorsque cette valeur est faible, celà signifie qu'aucun couple de partitions des lignes et des colonnes dans les nombres de classes fixés ne permet de restituer la dispersion initiale de façon satisfaisante; on ne retiendra, en général, pas les résultats correspondants
- pour chaque classe de la partition en lignes, son effectif et les groupes d'origine dont la moyenne appartient éventuellement à cette classe;
- les classes de variables obtenues;
- le tableau des $(f_{JK}/f_J.f_K).1000$ qui indique le degré de dépendance existant les classes; plus un terme en JK est éloigné de 1000 (inférieur ou supérieur), plus la $J^{ème}$ classe de blocs est dépendante de la $K^{ème}$ classe de mesures;
- le tableau des pourcentages de dispersion, ou parts de χ^2, conservés par chaque classe, dont la somme est le pourcentage de χ^2 conservé globalement, qui indique celles des classes dont la dispersion est bien conservée après partition;
- le tableau des profils de chaque classe de blocs sur l'ensemble des mesures, soit l'ensemble des moyennes des mesures à l'intérieur des classes de blocs.

Annexe 3

Analyse de l'ensemble des 9 variables actives, taille incluse
Variables illustratives : Nva et Trf

Résultats de la Classification Croisée

1 - sur le tableau initial :

Pourcentage de χ^2 conservé : 69.15

Partition des blocs :

Classe A : 163 blocs, dont les centres de gravité des groupes 2, 3, 4 et 5
Classe B : 52 blocs, dont le centre de gravité du groupe 1
Classe C : 391 blocs
Classe D : 50 blocs, dont le centre de gravité du groupe 6
Classe E : 15 blocs
Classe F : 257 blocs, dont le centre de gravité du groupe 7

Partition des variables :

Classe 1 : Otd, Prf, Imn, Imx
Classe 2 : NN, Nin
Classe 3 : Ond, cmd
Classe 4 : Voc

Tableau des $(f_{JK}/f_J.f_K).1000$:

	1	2	3	4
A	1038	1012	933	972
B	577	1185	923	752
C	1175	703	1065	1350
D	540	946	1574	1148
E	238	1400	652	476
F	1306	876	1014	1169

Tableau des pourcentages de dispersion conservés par chaque classe :

	1	2	3	4	Total
A	2	2	10	8	4
B	80	84	7	84	72
C	66	88	7	89	72
D	74	16	70	61	66
E	96	84	54	96	87
F	45	61	0	71	47
Total	68	77	41	85	69

Tableau des profils :

	A	B	C	D	E	F
Otd	2674	1082	2970	603	302	3189
Prf	1594	559	2107	313	170	1858
Imn	287	103	266	48	29	295
Imx	438	169	312	61	44	401
NN	17584	14225	8009	6670	12005	13737
Nin	3116	2470	1520	976	1568	2647
Ond	3015	1951	2267	1052	1021	3072
cmd	930	736	717	575	287	847
Voc	5689	3032	5237	2655	1323	6262

2 - sur le tableau normalisé par la taille :

Pourcentage de χ^2 conservé : 34.92

Ce pourcentage étant faible, nous n'avons pas retenu les résultats relatifs à cette analyse.

Analyses des Correspondances

1 - sur le tableau initial :

Les graphiques constituent la figure 18. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : NN, puis Voc et Otd;
pour l'axe 2 : Ond;
pour l'axe 3 : cmd, puis Imn, Imx;
pour l'axe 4 : Nin;
pour l'axe 5 : les mêmes que l'axe 3.

2 - sur le tableau normalisé par la taille :

Les graphiques constituent la figure 19. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : NN, puis Nin;
pour l'axe 2 : Imn, Imx;
pour l'axe 3 : Voc, Ond;
pour l'axe 4 : faiblement Ont, Otd et cmd;
pour l'axe 5 : cmd.

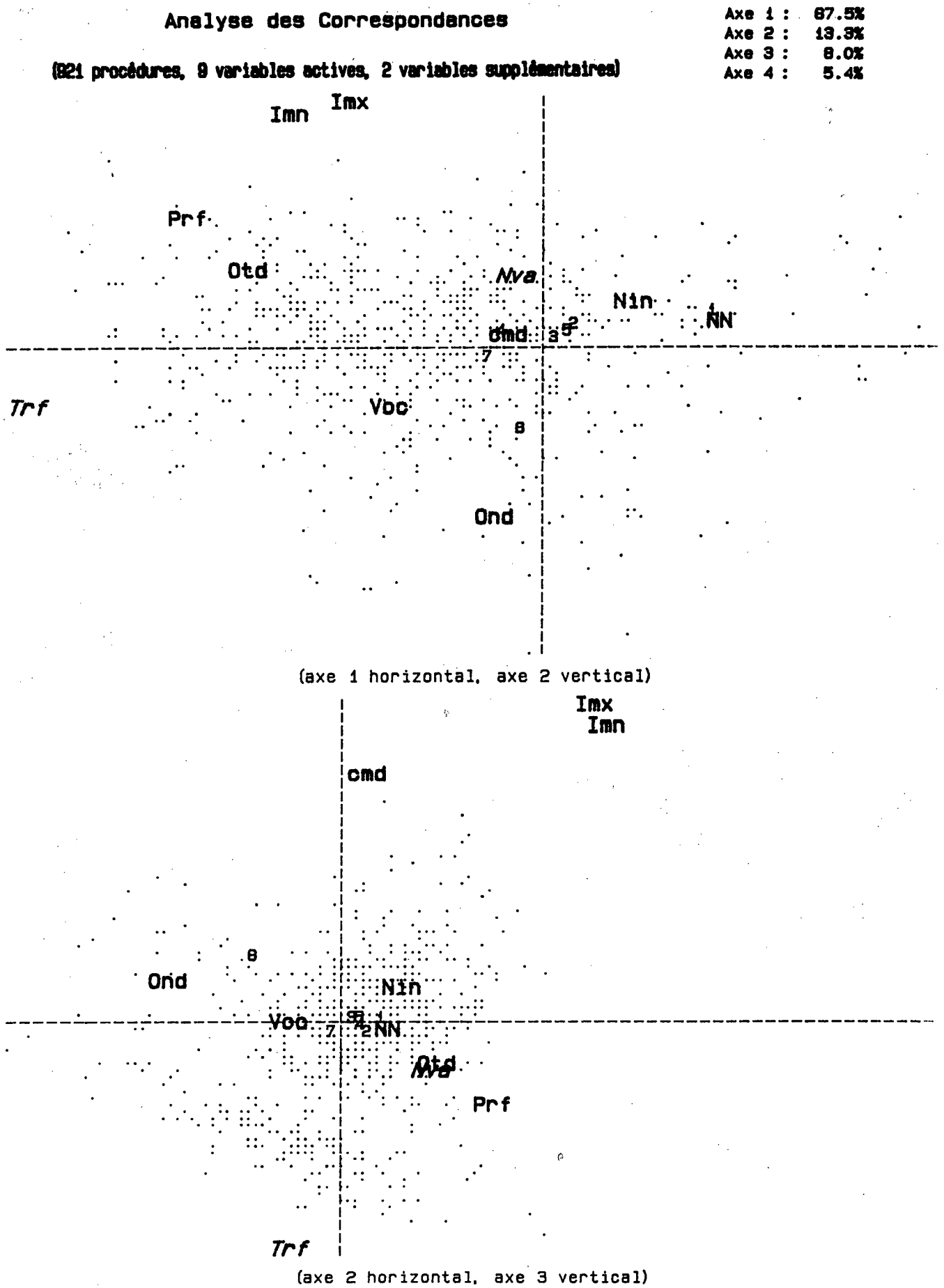
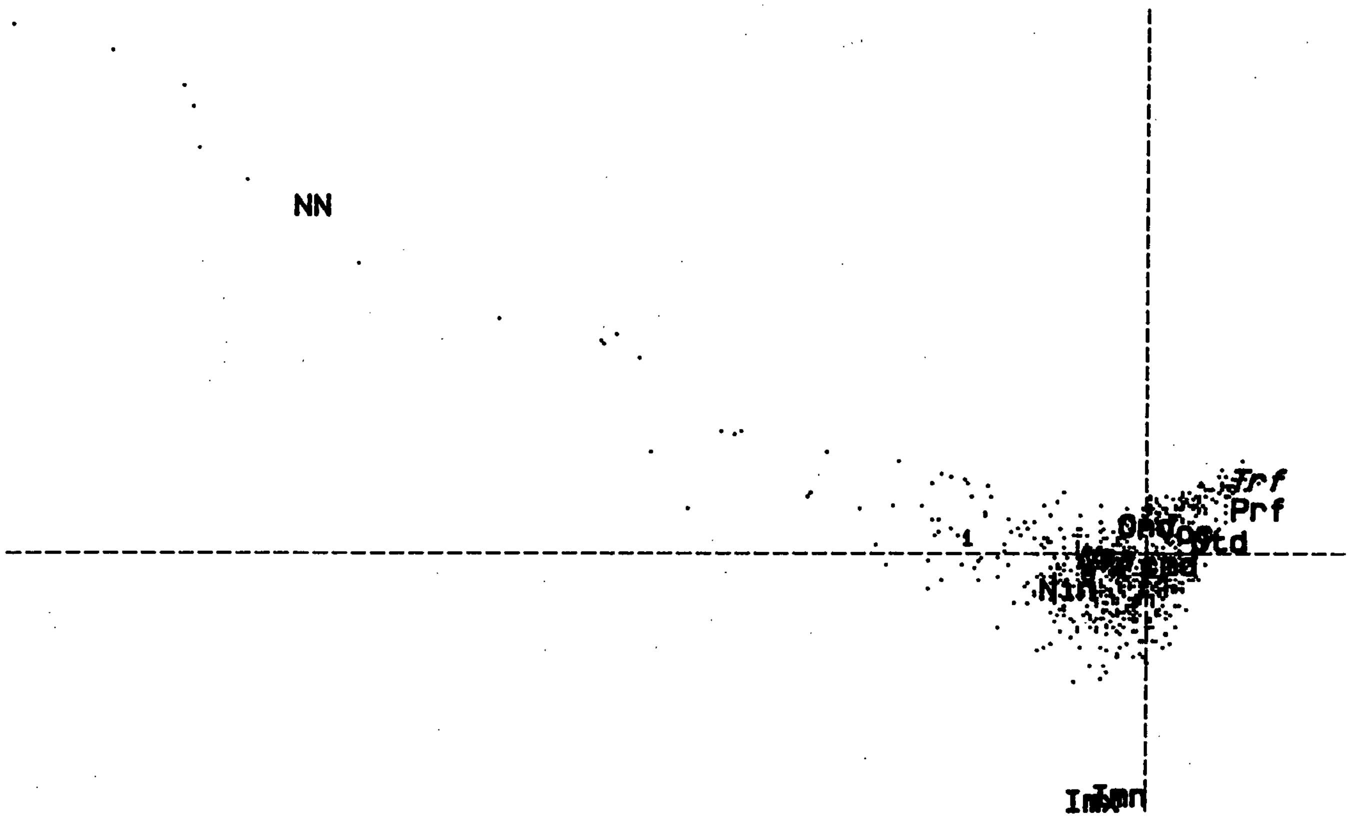

Analyse des Correspondances
(821 procédures, 9 variables actives, 2 variables supplémentaires)
Axe 1 : 67.5%
Axe 2 : 13.3%
Axe 3 : 8.0%
Axe 4 : 5.4%
Imn
Imx
Prf
Otd
Nva
Nin
NN
cmd
Trf
Vbc
Ond
(axe 1 horizontal, axe 2 vertical)
Imx
Imn
cmd
Ond
Nin
Vbc
NN
Otd
Nva
Prf
Trf
(axe 2 horizontal, axe 3 vertical)

Figure 18

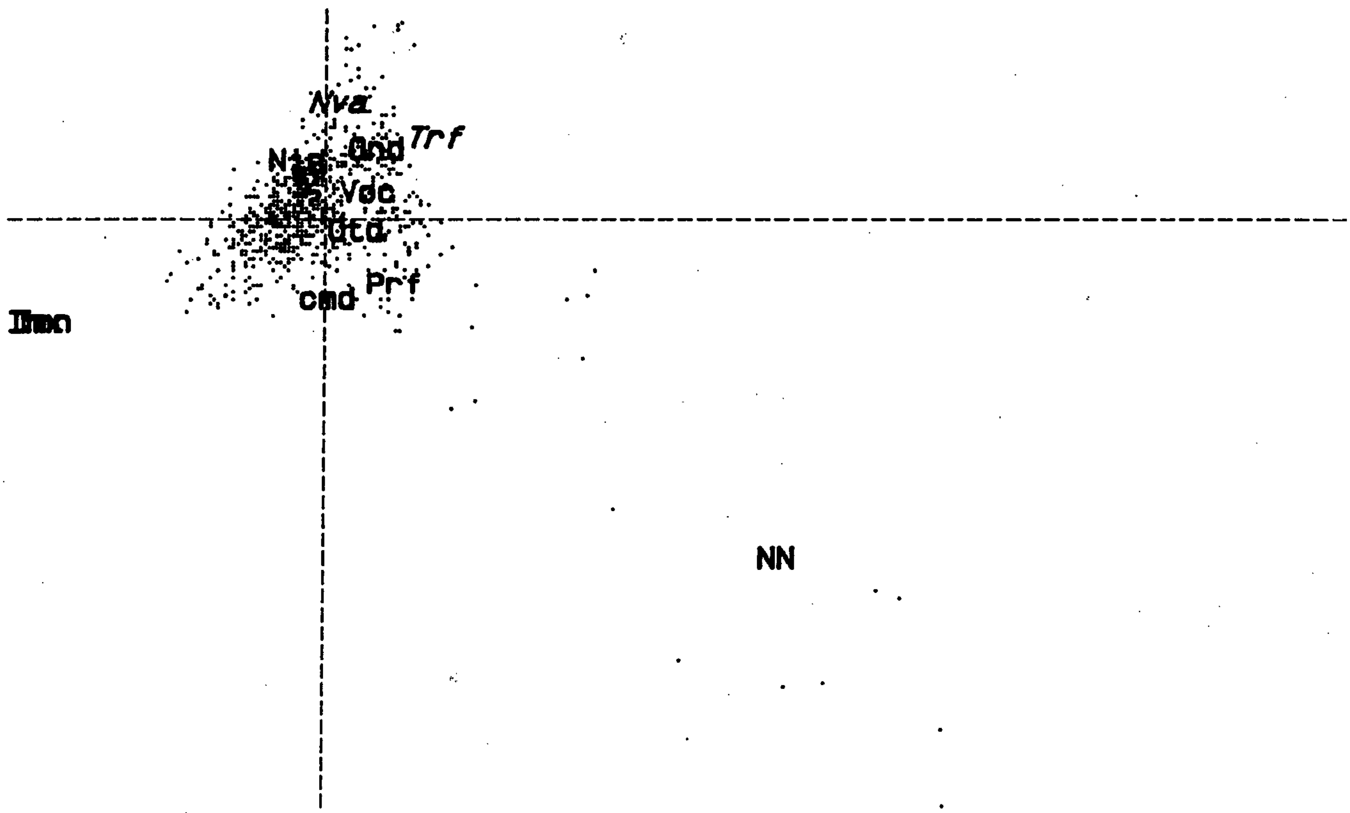
Analyse des Correspondances (fichier normalisé/NN)

(921 procédures, 9 variables actives, 2 variables supplémentaires)

Axe 1 :	44.4%
Axe 2 :	21.1%
Axe 3 :	15.2%
Axe 4 :	9.9%



(axe 1 horizontal, axe 2 vertical)



(axe 2 horizontal, axe 3 vertical)

Figure 19

Annexe 4

Analyse de l'ensemble des 8 variables actives, sans la taille
Variables illustratives : Nva et Trf

Résultats de la Classification Croisée

1 - sur le tableau initial :

Pourcentage de χ^2 conservé : 57.25

Partition des blocs :

Classe A : 142 blocs, dont les centres de gravité des groupes 1, 3 et 5
Classe B : 361 blocs
Classe C : 53 blocs, dont le centre de gravité du groupe 6
Classe D : 326 blocs, dont les centres de gravité des groupes 2, 4 et 7
Classe E : 29 blocs
Classe F : 17 blocs

Partition des variables :

Classe 1 : Nin
Classe 2 : Ond
Classe 3 : Voc, cmd
Classe 4 : Otd, Imx, Imn et Prf

Tableau des $(f_{JK}/f_J.f_K).1000$:

	1	2	3	4
A	1157	978	983	940
B	579	785	999	1405
C	772	1523	1166	552
D	846	911	997	1158
E	1782	894	922	696
F	1669	1625	938	252

Tableau des pourcentages de dispersion conservés par chaque classe :

	1	2	3	4	Total
A	54	2	1	6	14
B	86	71	0	48	57
C	69	90	27	79	70
D	55	32	0	24	26
E	88	34	5	67	63
F	85	91	4	96	83
Total	82	74	8	54	57

Tableau des profils :

	A	B	C	D	E	F
Nin	3357	1394	918	3292	2195	1140
Ond	3089	2054	1971	3857	1199	1208
Voc	5623	4836	2599	7806	1962	1372
cmd	944	691	591	1112	653	102
Otd	2534	2782	628	3949	763	164
Imx	362	316	60	539	137	11
Imn	236	261	48	391	82	10
Prf	1279	2102	325	2396	405	94

2 - sur le tableau normalisé par la taille :

Pourcentage de χ^2 conservé : 46.03

Ce pourcentage étant faible, nous n'avons pas retenu les résultats relatifs à cette analyse.

Analyses des Correspondances

1 - sur le tableau initial :

Les graphiques constituent la figure 20. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : Otd, Prf, puis Nin;
pour l'axe 2 : Voc, puis Ond et Imx;
pour l'axe 3 : cmd;
pour l'axe 4 : Imn, Imx.

2 - sur le tableau normalisé par la taille :

Les graphiques constituent la figure 21. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : Prf, Nin;
pour l'axe 2 : Imn, Imx;
pour l'axe 3 : cmd;
pour l'axe 4 : cmd;
pour l'axe 5 : Voc.

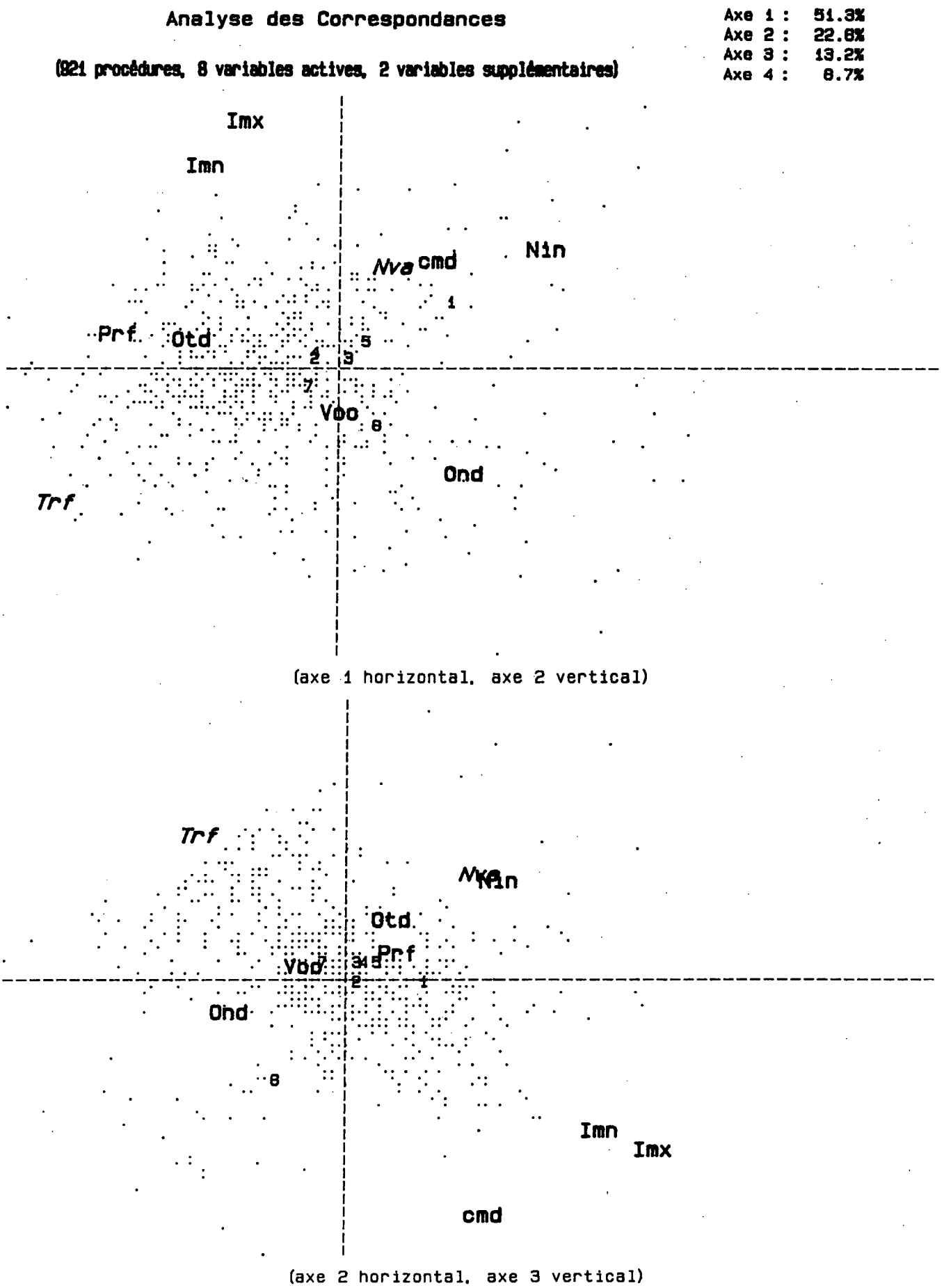

Analyse des Correspondances
(821 procédures, 8 variables actives, 2 variables supplémentaires)
Axe 1 : 51.3%
Axe 2 : 22.6%
Axe 3 : 13.2%
Axe 4 : 8.7%
Imx
Imn
Nva
cmd
Nin
Prf
Otd
Voo
Ond
Trf
1
2
3
4
5
6
7
8
(axe 1 horizontal, axe 2 vertical)
Trf
Nva
Nin
Otd
Prf
Voo
Ond
Imn
Imx
cmd
(axe 2 horizontal, axe 3 vertical)

Figure 20

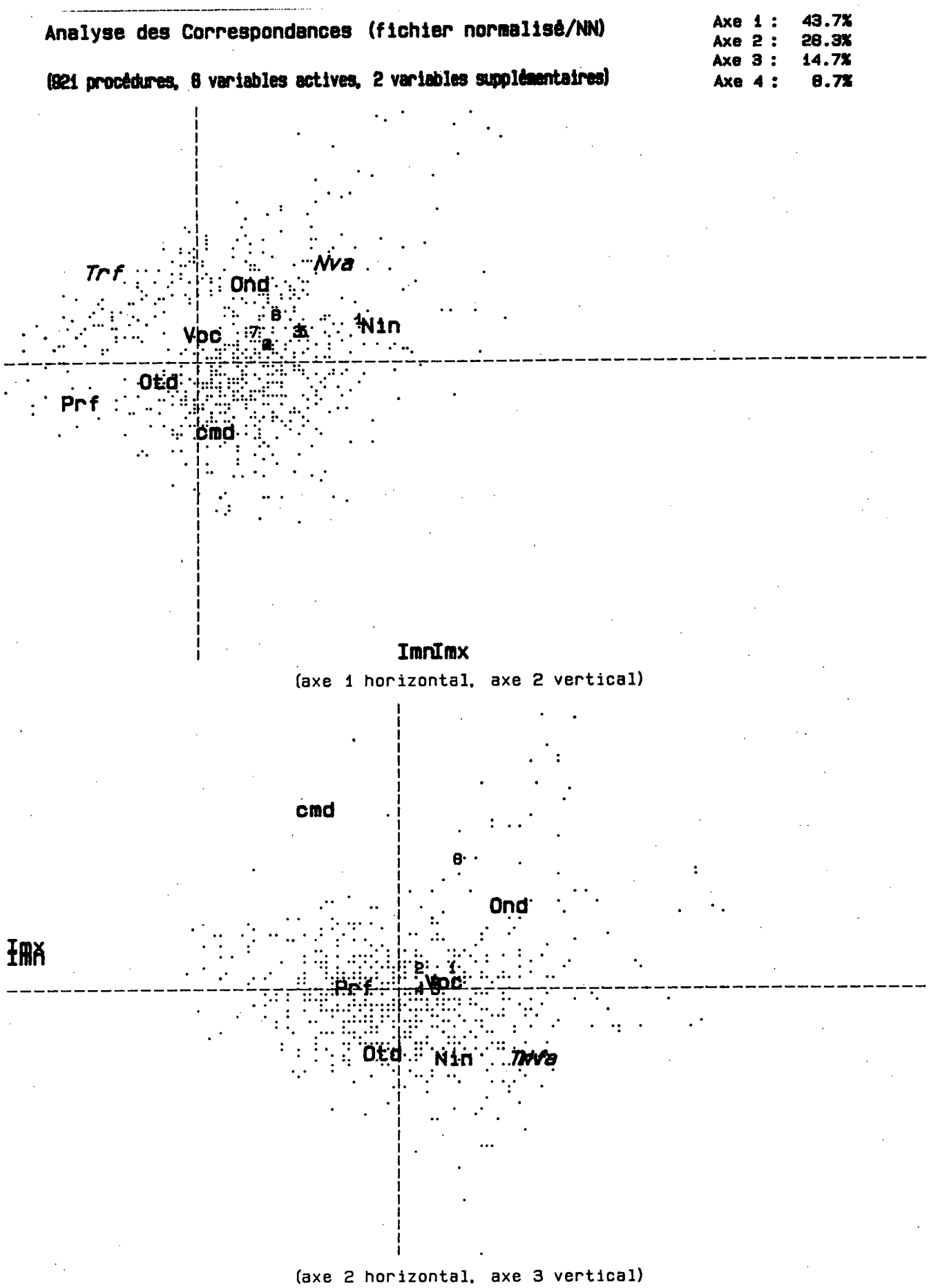

Analyse des Correspondances (fichier normalisé/NN)
(921 procédures, 8 variables actives, 2 variables supplémentaires)
Axe 1 : 43.7%
Axe 2 : 28.3%
Axe 3 : 14.7%
Axe 4 : 8.7%
Trf
Nva
Ond
Vpc
Nin
Otd
Prf
cmd
ImnImx
(axe 1 horizontal, axe 2 vertical)
cmd
Ond
Imx
Imn
Prf
Vpc
Otd
Nin
Nva
(axe 2 horizontal, axe 3 vertical)

Figure 21

Annexe 5

Analyse de l'ensemble des 6 variables classiques actives, taille incluse
Variables illustratives : les 5 variables Prf, Imn, Imx, Nva et Trf

Résultats de la Classification Croisée

1 - sur le tableau initial :

Pourcentage de χ^2 conservé : 70.83

Partition des blocs :

Classe A : 47 blocs, dont le centre de gravité du groupe 1
Classe B : 270 blocs, dont les centres de gravité des groupes 4 et 7
Classe C : 137 blocs, dont les centres de gravité des groupes 2, 3 et 5
Classe D : 47 blocs, dont le centre de gravité du groupe 6
Classe E : 413 blocs
Classe F : 14 blocs

Partition des variables :

Classe 1 : Ond, cmd
Classe 2 : NN, Nin
Classe 3 : Voc, Otd

Tableau des $(f_{JK}/f_J.f_K).1000$:

	1	2	3
A	890	1164	656
B	1009	911	1208
C	934	1032	953
D	1532	913	943
E	1146	743	1546
F	606	1339	376

Tableau des pourcentages de dispersion conservés par chaque classe :

	1	2	3	Total
A	12	83	84	70
B	0	44	68	48
C	9	12	16	12
D	74	43	4	51
E	20	85	83	78
F	64	80	95	85
Total	42	74	80	71

Tableau des profils :

	A	B	C	D	E	F
Ond	1867	3488	930	1525	1167	609
cmd	698	979	944	1557	1080	598
NN	13871	16363	1028	935	728	1387
Nin	2532	3148	1057	782	830	1057
Voc	2828	7101	947	1104	1441	440
Otd	962	3613	967	583	1780	232

2 - sur le tableau normalisé par la taille :

Pourcentage de χ^2 conservé : 58.27

Partition des blocs :

Classe A : 195 blocs
Classe B : 425 blocs, dont les centres de gravité des groupes 6 et 7
Classe C : 32 blocs, dont le centre de gravité du groupe 1
Classe D : 8 blocs
Classe E : 268 blocs, dont les centres de gravité des groupes 2, 3, 4 et 5

Partition des variables :

Classe 1 : NN
Classe 2 : Nin
Classe 3 : Voc, Ond, Otd, cmd

Tableau des $(f_{JK}/f_J.f_K).1000$:

	1	2	3
A	147	556	1077
B	532	1032	997
C	11382	1935	779
D	41265	1936	576
E	1900	1466	918

Tableau des pourcentages de dispersion conservés par chaque classe :

	1	2	3	Total
A	96	85	11	44
B	48	5	0	5
C	82	84	34	77
D	97	93	61	96
E	30	83	14	44
Total	87	79	10	58

Tableau des profils :

	A	B	C	D	E
NN	32	182	135	85	288
Nin	2829	8412	545	95	5289
Voc	15781	23586	598	78	9566
Ond	6923	11030	398	60	5052
Otd	8858	12556	211	18	4514
cmd	2464	3335	166	19	1462

Analyses des Correspondances

1 - sur le tableau initial :

Les graphiques constituent la figure 22. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : NN, Voc puis Otd;
pour l'axe 2 : Ond;
pour l'axe 3 : cmd;
pour l'axe 4 : Nin;

2 - sur le tableau normalisé par la taille :

Les graphiques constituent la figure 23. La représentation sur les plans des axes 1 et 2 a été volontairement omise, l'effet de taille étant tel qu'il la rendait illisible en entassant toutes les variables autres que la taille. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : NN uniquement;
pour l'axe 2 : NN, puis cmd et Nin;
pour l'axe 3 : Ond;
pour l'axe 4 : cmd;

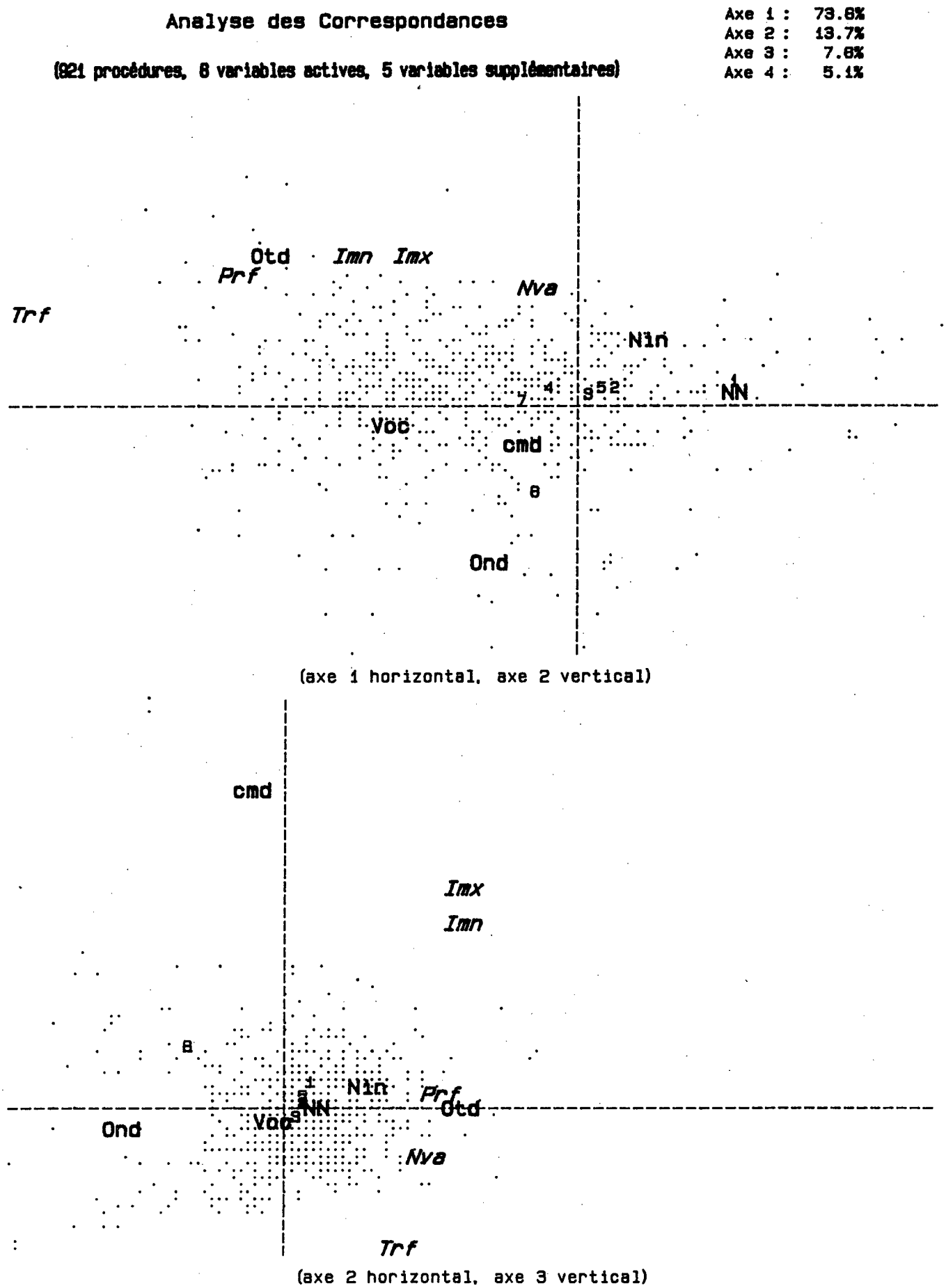

Analyse des Correspondances
(821 procédures, 8 variables actives, 5 variables supplémentaires)
Axe 1 : 73.8%
Axe 2 : 13.7%
Axe 3 : 7.8%
Axe 4 : 5.1%
Otd
Imn
Imx
Prf
Nva
Trf
Nin
NN
Voc
cmd
Ond
(axe 1 horizontal, axe 2 vertical)
cmd
Imx
Imn
Nin
Prf
Otd
NN
Voc
Ond
Nva
Trf
(axe 2 horizontal, axe 3 vertical)

Figure 22

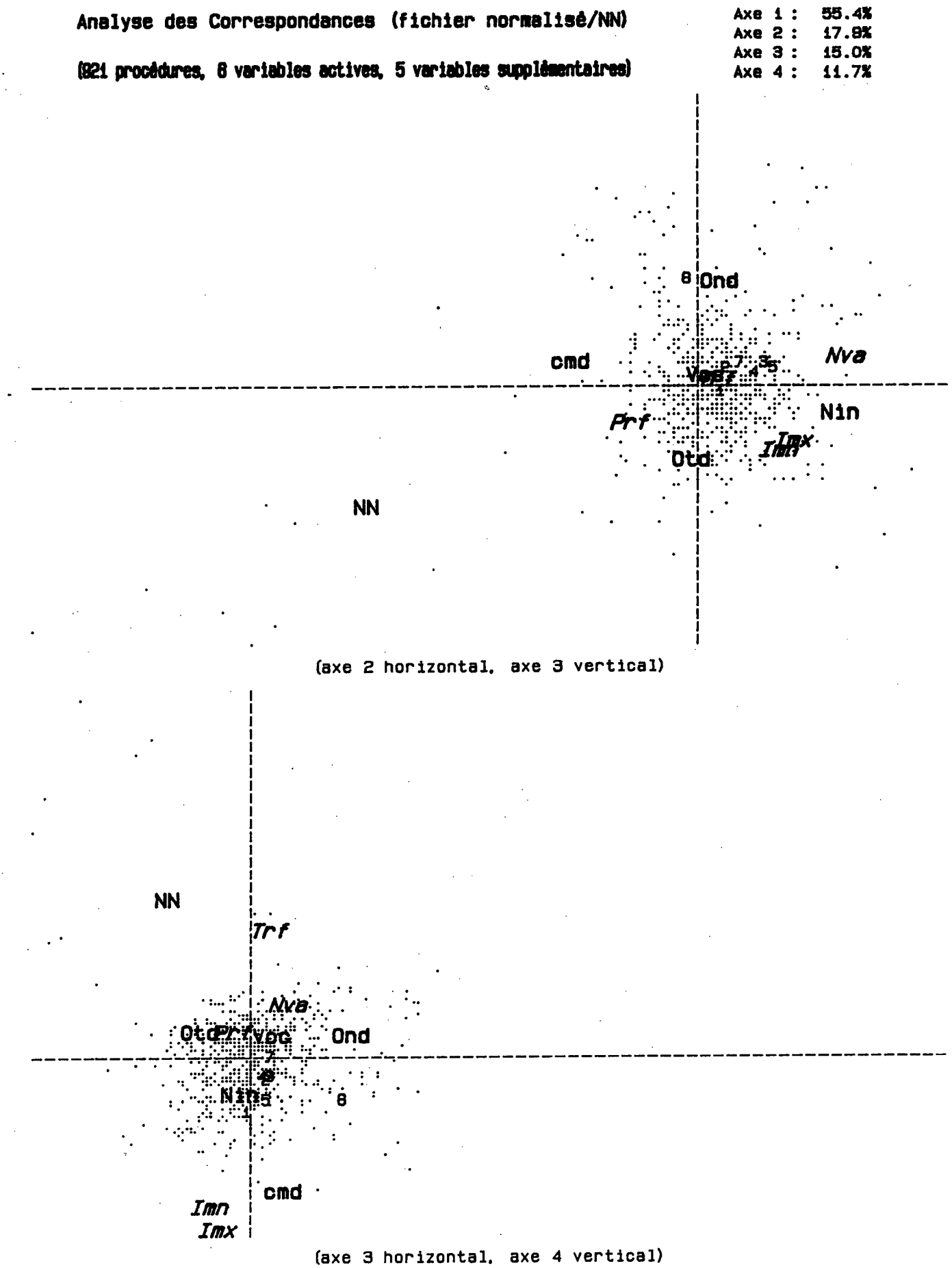

Analyse des Correspondances (fichier normalisé/NN)
(921 procédures, 8 variables actives, 5 variables supplémentaires)
Axe 1 : 55.4%
Axe 2 : 17.8%
Axe 3 : 15.0%
Axe 4 : 11.7%
Ond
cmd
Nva
Nin
Prf
Imx
Imn
Otd
NN
(axe 2 horizontal, axe 3 vertical)
NN
Trf
Nva
Ond
cmd
Imn
Imx
(axe 3 horizontal, axe 4 vertical)

Figure 23

Annexe 6

Analyse de l'ensemble des 5 variables classiques actives, sans la taille
Variables illustratives : les 5 variables Prf, Imn, Imx, Nva et Trf

Résultats de la Classification Croisée

1 - sur le tableau initial :

Pourcentage de χ^2 conservé : 55.69

Ce pourcentage étant faible, nous n'avons pas retenu les résultats relatifs à cette analyse.

2 - sur le tableau normalisé par la taille :

Pourcentage de χ^2 conservé : 48.73

Ce pourcentage étant faible, nous n'avons pas retenu les résultats relatifs à cette analyse.

Analyses des Correspondances

1 - sur le tableau initial :

Pour des raisons diverses, les analyses relatives au tableau de données à 5 et 11 variables actives ont été effectuées postérieurement aux autres analyses; les résultats obtenus sur le fichier normalisé par la taille étant en général plus intéressants, nous n'avons pas effectué pour ces tableaux les analyses portant sur le fichier initial.

2 - sur le tableau normalisé par la taille :

Les graphiques constituent la figure 24. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : Nin, puis Voc;
pour l'axe 2 : Ond, puis Otd;
pour l'axe 3 : cmd, puis Voc;

On note que l'ensemble des points est contenu dans un espace à 3 dimensions (part de dispersion nulle sur l'axe 4); il n'y a donc aucune perte d'information dans les projections de la figure 24.

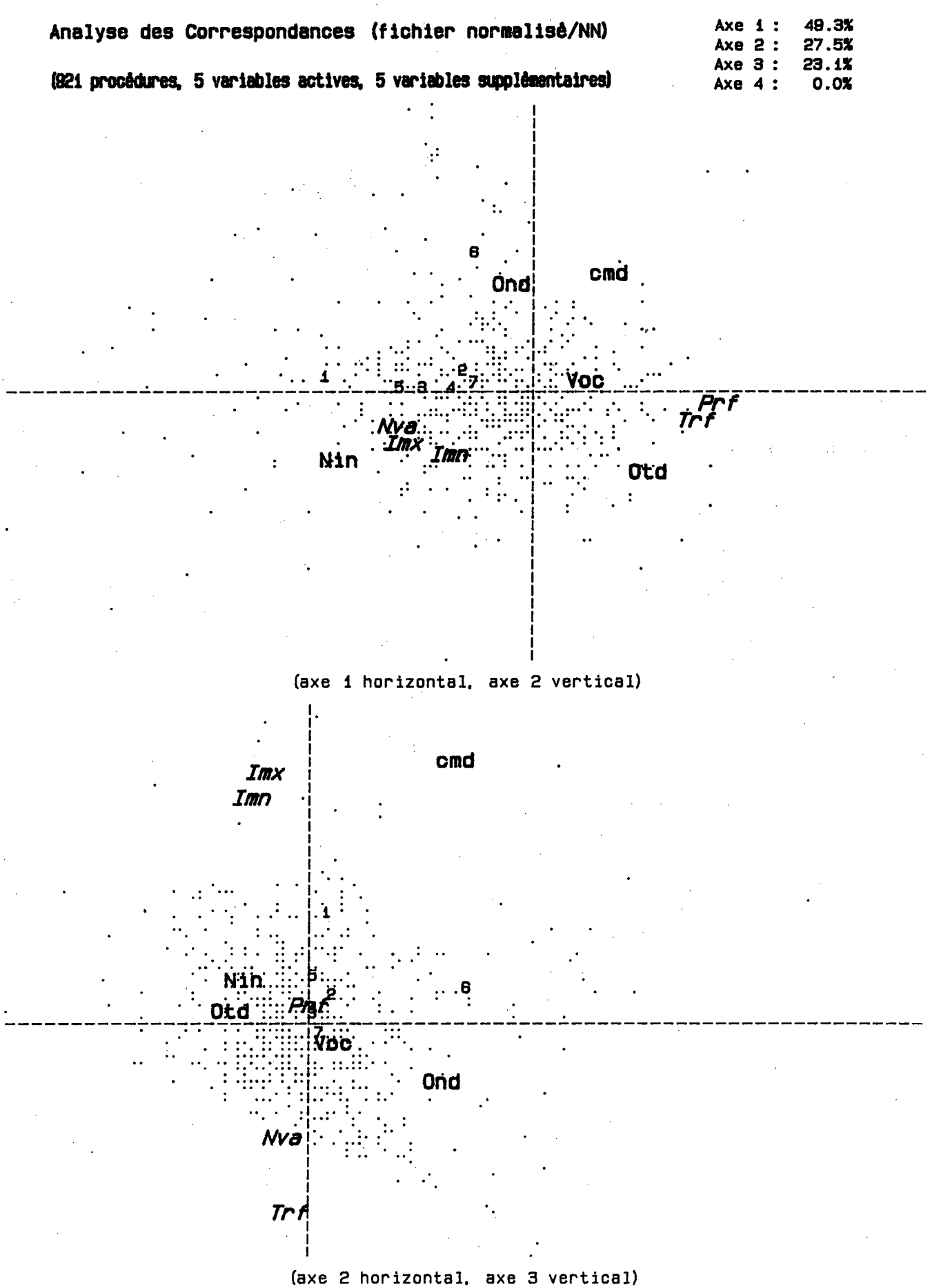

Analyse des Correspondances (fichier normalisé/NN)
(821 procédures, 5 variables actives, 5 variables supplémentaires)
Axe 1 : 49.3%
Axe 2 : 27.5%
Axe 3 : 23.1%
Axe 4 : 0.0%
6
Ond
cmd
1
5
8
4
2
7
Voc
Prf
Trf
Nva
Imx
Imn
Nin
Otd
(axe 1 horizontal, axe 2 vertical)
cmd
Imx
Imn
1
Nin
5
2
6
Otd
Prf
7
Voc
Ond
Nva
Trf
(axe 2 horizontal, axe 3 vertical)

Figure 24

Annexe 7

Analyse de l'ensemble des 11 variables actives, sans la taille

Résultats de la Classification Croisée

1 - sur le tableau initial :

Pour des raisons diverses, les analyses relatives au tableau de données à 5 et 11 variables actives ont été effectuées postérieurement aux autres analyses; les résultats obtenus sur le fichier normalisé par la taille étant en général plus intéressants, nous n'avons pas effectué pour ces tableaux l'analyse du tableau initial.

2 - sur le tableau normalisé par la taille :

Pourcentage de χ^2 conservé : 46.34

Ce pourcentage étant faible, nous n'avons pas retenu les résultats relatifs à cette analyse.

Analyses des Correspondances

1 - sur le tableau initial :

Pour des raisons diverses, les analyses relatives au tableau de données à 5 et 11 variables actives ont été effectuées postérieurement aux autres analyses; les résultats obtenus sur le fichier normalisé par la taille étant en général plus intéressants, nous n'avons pas effectué pour ces tableaux l'analyse du tableau initial.

2 - sur le tableau normalisé par la taille :

Les graphiques constituent la figure 25. Les variables contribuant le plus aux différents axes sont :

pour l'axe 1 : Nva, Prf, Nin;
pour l'axe 2 : Imx;
pour l'axe 3 : Nva, Ond;
pour l'axe 4 : cmd, cmx;
pour l'axe 5 : Ond, Nin;
pour l'axe 6 : Prf.

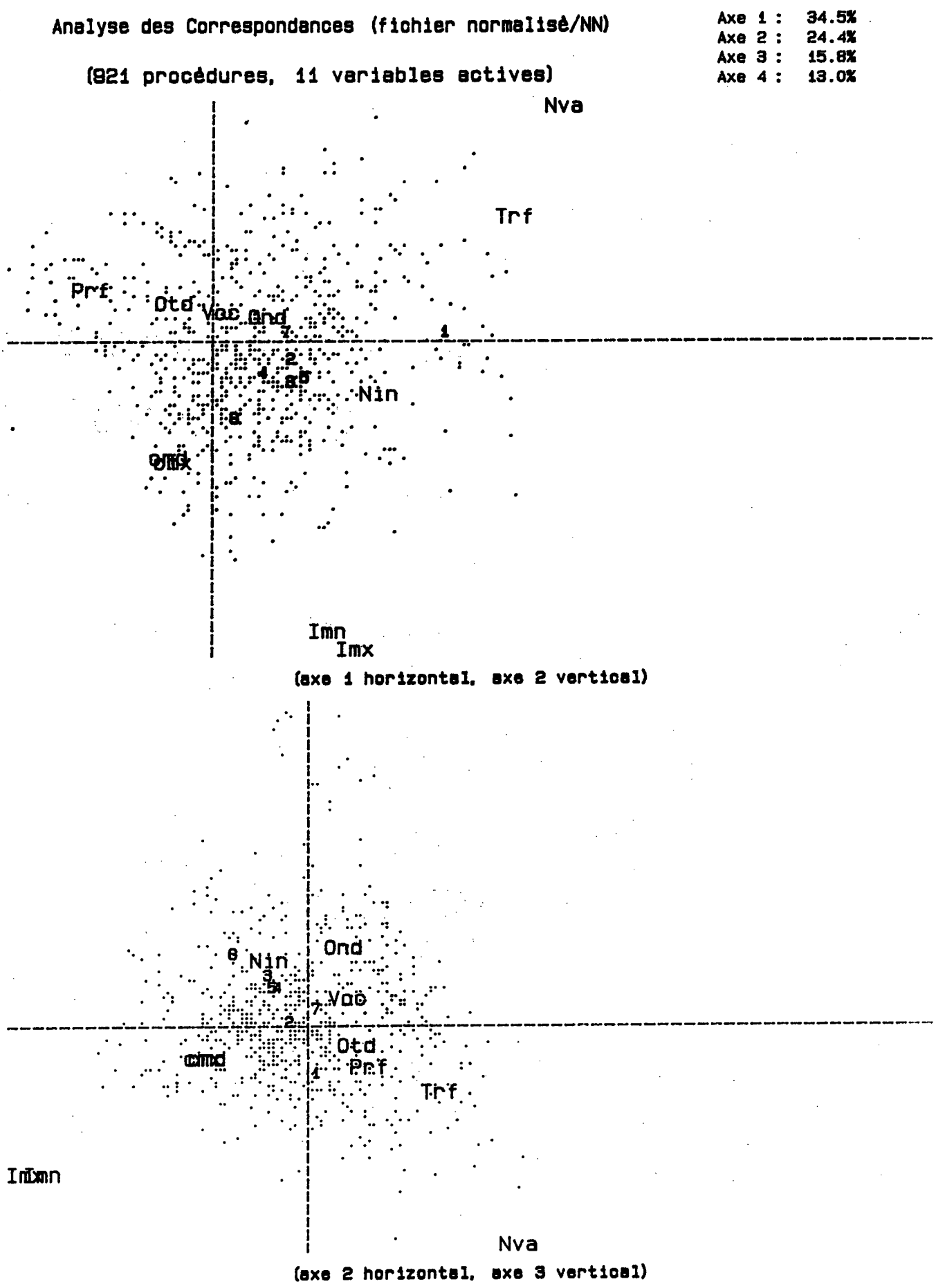

Analyse des Correspondances (fichier normalisé/NN)
(921 procédures, 11 variables actives)
Axe 1 : 34.5%
Axe 2 : 24.4%
Axe 3 : 15.8%
Axe 4 : 13.0%
Nva
Trf
Prf
Otd
Vac
Ond
Nin
Imn
Imx
(axe 1 horizontal, axe 2 vertical)
Ond
Nin
Vac
Otd
Prf
Trf
Nva
(axe 2 horizontal, axe 3 vertical)

Figure 25

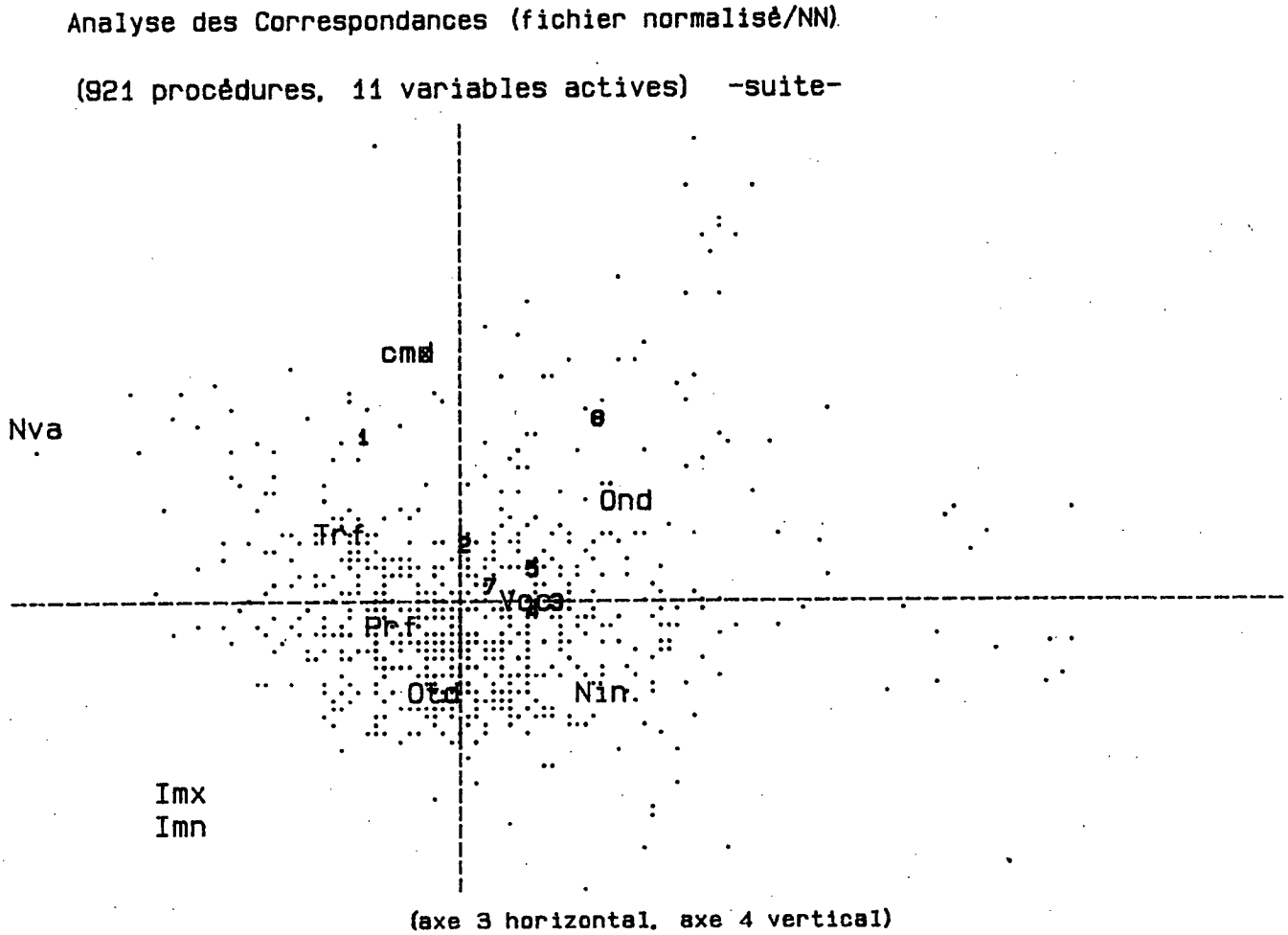


Figure 25 (suite)

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique