NOTICE D'INSTALLATION ET D'UTILISATION
DE PROGRAMMES BASES SUR L'ALGORITHME DE KOHONEN
ET DEDIES A L'ANALYSE DE DONNEES

3/12/2005 (V8.2-V9.1.3)

Patrick Letrémy
MATISSE-SAMOS UMR CNRS 8595
Université Paris 1
pley@univ-paris1.fr

Algorithme d'apprentissage de Kohonen (SOM)
Algorithme KACP (Analyse d'un tableau de données quantitatives)
Algorithme KACPX (Analyse d'un tableau de données quantitatives avec valeurs manquantes)
Algorithme KFAST (SCL : Kohonen à zéro voisin)
Algorithme KBATCH (Kohonen dans sa version déterministe)
Algorithme KORRESP (Analyse d'un tableau de contingence)
Algorithmes KACM(j=1,2) et KDISJ (Analyse d'une table de Burt ou d'un tableau Disjonctif complet)

I - CONTENU DE LA DISQUETTE (OU DU REPERTOIRE), INSTALLATION

La distribution contient les trois fichiers suivants :
AVERTI.TXT : le fichier d'annonce
DABOR.TXT : le fichier d'introduction
KSET_V8.ZIP : le fichier à décompresser
L'installation par DEFAUT comporte 2 étapes:
1) - Décompresser le fichier KSET_V8.ZIP de la disquette (ou du répertoire) sur c:
2) - Charger et soumettre (Submit ou F3) dans SAS l'exécutable KINSTAL.sas situé dans le répertoire C:\K8.

Pour plus de détails concernant l'installation par défaut, lire le paragraphe II "DESCRIPTION DE LA PROCEDURE D'INSTALLATION" de la présente notice.
L'installation terminée, il est conseillé de lire le paragraphe III "CONTENU DES PROCEDURES ET REFERENCES" de la présente notice.
Pour une information complète sur l'utilisation des programmes, il faut consulter les paragraphes IV,V,VI,VII et VIII du présent fichier C:\K8\LADOK.DOC.

II - DESCRIPTION DE LA PROCEDURE D'INSTALLATION

Vous avez suivi les indications de la section I, qui consistent en deux étapes:
- Décompression du fichier KSET_V8.ZIP de la disquette (ou du répertoire) sur c:
- Chargement et lancement dans SAS de l'exécutable KINSTAL.sas situé dans le répertoire c:\k8.
Vous trouverez ci-dessous la liste des fichiers créés au cours de l'installation.
L'étape 1) génère sur le disque dur C: l'arborescence suivante

C:\K8
C:\K8\ABRI
C:\K8\ABOUL
C:\K8\AKOR
C:\K8\AMAC

Contenu du répertoire K8 :

1. *LADOK.DOC le présent fichier.*
2. *Les 7 batch de lancement des algorithmes :*
   BKOHN.sas, DKACP.sas, DKACPX.sas, DKFAST.sas, DKBATCH.sas, DKORR.sas et DKACM.sas
3. *KOPTIONS.sas est une macro qui définit les chemins d'accès des catalogues et des jeux de données.*
4. *KOLCLUS.sas donne un aperçu des 22 couleurs disponibles pour les Clusters (Super Classes).*
5. *KOLCAM.sas donne un aperçu des 13 couleurs disponibles pour les secteurs de Camemberts*
6. *KOMAC.sas permet la création du catalogue de macros SASMACR.sas7bcat dans C:\K8\AKOR (cf. le paragraphe VIII " TRAITEMENTS COMPLEMENTAIRES ").*
7. *KINSTAL.sas déjà cité.*
8. *DSTAT.sas est un utilitaire qui permet de calculer, pour une classification obtenue à partir de variables quantitatives, les moyennes conditionnelles, l'analyse de la variance et certaines statistiques de tests multidimensionnels.*
9. *DTAIL.sas est un utilitaire qui permet, à partir d'un tableau de réponses, de créer le tableau disjonctif complet et la table de Burt sous forme de deux tables sas; dans le cas où les réponses sont codées sous forme numérique, l'utilitaire fournit une troisième table qui correspond à un recodage des réponses sous forme de libellés.*
10. *DMISS.sas est un utilitaire qui permet de contrôler la présence (en nombre, en pourcentage et en position) de données manquantes dans une table sas.*
11. *DEVIATION.sas est un utilitaire qui permet, à partir de classifications d'individus et de modalités, de calculer les effectifs observés et théoriques, les déviations, les contributions au Khi deux, les statistiques du khi deux et de contrôler le bon positionnement des modalités.*
12. *DEVAL.SAS est un utilitaire qui permet d'étudier le croisement d'une classification avec des variables qualitatives (questions d'enquête) et qui fournit des graphiques (camemberts et histogrammes), ainsi que des tests du khi deux, et des valeurs tests.*
13. *DCAM.sas est un utilitaire qui permet de placer, sur une structure de ficelle ou de grille, des camemberts construits à partir d'une variable qualitative.*
14. *DUPLIK.sas est un utilitaire qui permet de dupliquer les lignes d'une table sas selon une pondération donnée.*
15. *DISKRETIZ.sas est un utilitaire qui permet de discrétiser des variables continues.*

Contenu du répertoire K8\ABRI : 279 fichiers de programme qui seront compilés dans des catalogues IML permanents
Contenu du répertoire K8\ABOUL : jeux de données de 11 tables sas = BLAYO.sas7bdat, BLAYO2. sas7bdat, BLAYO3.sas7bdat, BOPAY3.sas7bdat, BXPAY3.sas7bdat, SUPPL96.sas7bdat, SUPPLBX3.sas7bdat, MONUMENT.sas7bdat, CHIENS.sas7bdat, PIRON.sas7bdat et TBPAY3.sas7bdat.

Contenu du répertoire K8\AKOR : au niveau de l'étape 1) il est vide, à l'étape 2) le batch KINSTAL.sas permet la création et le stockage dans le répertoire C:\K8\AKOR de 10 catalogues IML permanents :

1. *KOUPEL.sas7bcat (9 entrées) utilisé par KOHONEN*
2. *KOMPA.sas7bcat (39 entrées) qui est commun à tous les algorithmes*
3. *KOKAR.sas7bcat (40 entrées) qui est commun à l'utilisation de KACP, KFAST et KBATCH*
4. *KALO.sas7bcat (29 entrées) qui est propre à l'utilisation de KACPX*
5. *KOUL.sas7bcat (20 entrées) qui est propre à l'utilisation de KORRESP et KDISJ*
6. *KALM.sas7bcat (24 entrées) qui est commun à l'utilisation de KACM, KACM1 et KACM2*
7. *KOUTO.sas7bcat de 28 entrées (pour les traitements complémentaires)*
8. *KABA.sas7bcat de 29 entrées (pour les traitements complémentaires)*
9. *KANIF.sas7bcat de 25 entrées (pour les traitements complémentaires)*
10. *KAISSE.sas7bcat de 36 entrées (pour les traitements complémentaires)*

A la fin de l'exécution de KINSTAL, pour vérification, apparaissent dans la fenêtre OUTPUT de SAS : le nom et le contenu de ces 10 catalogues IML.

*Contenu du répertoire K8\AMAC : 20 fichiers de macro-commandes = MCONT.sas, MKLOBS.sas, MKOC.sas, MDIMA.sas, MDIST.sas, MFIND.sas, MFMOD.sas, MGVAR.sas, MKAMF1.sas, MKAMF2.sas, MKAMG.sas, MKLUF1.sas, MKLUF2.sas, MKLUG.sas, MSCAL.sas, MSCOL.sas, MSTAT.sas, MSUPP.sas, MSUPPX.sas et MVC3D.sas qui seront, **à chaque session**, compilés et stockés dans le catalogue C:\K8\AKOR\SASMACR.sas7bcat après la soumission de KOMAC.sas.*

III - CONTENU DES PROCEDURES ET REFERENCES

Thème : Programmes conversationnels pour :

- *l'algorithme d'apprentissage non supervisé de KOHONEN (SOM)*
- *l'algorithme KACP (version Kohonen de L'ACP)*
- *l'algorithme KACPX (version Kohonen de L'ACP avec données manquantes)*
- *l'algorithme KFAST (SCL : version Kohonen à zéro voisin)*
- *l'algorithme KBATCH (version déterministe de Kohonen)*
- *l'algorithme KORRESP (version Kohonen de L'AFC)*
- *les algorithmes KACM(j=1,2) et KDISJ (versions Kohonen de L'ACM)*

- *traitements complémentaires à ces algorithmes*
- *Le paragraphe IV décrit l'exécution de l'algorithme de KOHONEN pour des entrées qui sont des points de R^2 choisis au hasard dans un carré.*
- *Le paragraphe V décrit le déroulement de KACP(X), KFAST et KBATCH sur un jeu de données.*
- *Le paragraphe VI décrit le déroulement de KORRESP sur un jeu de données.*
- *Le paragraphe VII décrit le déroulement de KACM(j=1,2) et de KDISJ sur un jeu de données.*
- *Le paragraphe VIII décrit les traitements complémentaires.*

Références :

- F.Blayo et M.Verleysen, *Les réseaux de neurones artificiels*, Collection Que sais-je ?, vol 3042, PUF, 1996.
- M.Cottrell et P.Letrémy, Classification et analyse des correspondances au moyen de l'algorithme de Kohonen : application à l'étude de données socio-économiques, *Actes de Neuro Nîmes, 1994*.
- M.Cottrell et S.Ibbou, Multiple Correspondence Analysis of a crosstabulation matrix using the Kohonen algorithm, *Proc. of ESANN'95*, Editions D Facto, Bruxelles, 1995.
- M.Cottrell et E. de Bodt, A Kohonen map representation to avoid misleading interpretation, *Proc. of ESANN'96*, Editions D Facto, Bruxelles, 1996.
- Demartines, Analyse de données par réseaux de neurones auto-organisés, Thèse de Doctorat, Laboratoire TIRF, Institut National Polytechnique de Grenoble.
- Classification, Analyse des Correspondances et Méthodes Neuronales, Thèse de S. Ibbou soutenue le 20/1/98.
- Applications des Algorithmes d'Auto-Organisation à la Classification et à la Prévision, Thèse de P. Rousset soutenue le 3/12/99.
- M.Cottrell et P.Letrémy, Analysing surveys using the Kohonen algorithm, *ESANN'2003*, Bruges,2003, M. Verleysen Ed., Editions D Facto, Bruxelles.
- M.Cottrell, S.Ibbou, P.Letrémy SOM-based algorithms for qualitative variables, Neural Networks, 17, p. 1149-1167, 2004.

IV – DEMONSTRATION

Conseils pour la saisie dans les fenêtres IML qui sont assez rudimentaires ! !
Se déplacer de champ en champ avec la touche " TAB ". Pour entrer une nouvelle valeur : se placer en début de champ et supprimer (" Suppr ") l'ancienne valeur avant de saisir la nouvelle valeur.

Algorithme d'apprentissage de KOHONEN pour des points dans un carré de R^2

Charger et soumettre dans SAS le fichier BKOHN.sas qui se trouve dans c:\k8. Ce programme de démonstration ne nécessite aucune donnée. TOUTE LA SUITE EST CONVERSATIONNELLE
Une fenêtre "CHOIX" apparaît, elle correspond à une boucle sans fin dont on ne peut sortir que par choix : A (Arrêt définitif). Sinon l'utilisateur peut choisir entre un réseau de dimension 1, (ficelle choix : F) ou un réseau de dimension 2 (grille choix : G). Pour toutes les fenêtres, le choix retenu est validé par F3.

- *Si le choix est : F, une fenêtre "FICELLE" s'ouvre pour demander la taille de la ficelle (n), le nombre maximum d'itérations (tmax) et la périodicité de l'affichage (periode).*

Par exemple : n = 20, tmax = 200 et periode = 25 suivi de F3 pour valider.
Alors, 200/25+1=9 écrans graphiques seront produits et stockés provisoirement dans le catalogue WORK.GSEG.
A la fin des 200 itérations, l'utilisateur peut revenir sur l'un des 9 graphiques (à l'aide de "CATALOG WORK.GSEG") pour une éventuelle impression.

Dans le catalogue WORK.GSEG, les 9 graphiques sont nommés IMLG à IMLG8.
On peut toujours revenir à la fenêtre "CHOIX" avec la commande Window.

- *Si le choix est : G, une fenêtre "GRILLE" s'ouvre pour demander la taille de la grille (m = nombre de lignes et n = nombre de colonnes), le nombre maximum d'itérations (tmax) et la périodicité de l'affichage (periode).*

Par exemple : n = 7, m = 7, tmax = 500 et periode = 50 suivi de F3 pour valider.
Alors, 500/50+1=11 graphiques nommés IMLG à IMLG10 seront produits et stockés dans le catalogue WORK.GSEG.
Si on veut éviter un mélange de numérotation entre les graphiques "ficelles" et "grilles", on peut avant d'opter pour un nouveau choix (F ou G) détruire ou renommer le catalogue WORK.GSEG.
Il est donc possible de boucler indéfiniment sur choix : F ou G. Pour y mettre fin, il suffit de choisir: A suivi de plusieurs F3 pour revenir dans la fenêtre " PROGRAM EDITOR ".

V - L'ALGORITHME KACP

Algorithme KACP : Analyse en Composantes Principales version KOHONEN

Il s'agit d'une classification en classes liées par une structure de voisinage, qui utilise l'algorithme de Kohonen. On peut dire que cette analyse s'apparente à une analyse en composantes principales.

Pour illustrer le déroulement d'une session, nous utiliserons les données fournies par F.BLAYO qui concernent 53 pays en 1984 et qui sont présentées dans le Que sais-je ? vol 3042 "LES RESEAUX DE NEURONES ARTIFICIELS" de F. BLAYO et M. VERLEYSEN.

Pour chaque pays on dispose des valeurs de 7 variables :

1. *PAYS (nom du pays),*
2. *ANCRX (croissance annuelle de la population),*
3. *TXMORT (taux de mortalité infantile),*
4. *TXANAL (taux d'illettrisme),*
5. *SCOL2 (fréquentation scolaire du 2ème degré),*
6. *PIBH (PIB par habitant),*
7. *CRXPIB (croissance annuelle du PIB).*

Ces données sont dans une table sas (BLAYO.sas7bdat) qui est placée dans le répertoire c:\k8\aboul . La table C:\K8\ABOUL\BOPAY3.sas7bdat (96 pays en 1996) est une mise à jour plus récente de la table BLAYO.sas7bdat, avec introduction de nouvelles variables comme les taux de chômage et d'inflation, ainsi qu'une variable qualitative indiquant le niveau de l'IDH (Indice du Développement Humain).

DEBUT DE LA SESSION KACP

Charger et soumettre dans SAS : DKACP.sas qui se situe dans c:\k8.
Toute la suite est CONVERSATIONNELLE, des fenêtres s'ouvrent et guident l'utilisateur dans ses choix. Pour toutes les fenêtres, les choix retenus seront validés par F3.

Fenêtre n°1 : "CHEMIN".
L'utilisateur précise le chemin d'accès aux données (par défaut c:\k8\aboul).

Exemple : path: C:\K8\ABOUL suivi de F3.

Dans le cas où le répertoire proposé est vide ou inexistant, un message d'erreur apparaît dans la fenêtre LOG, la commande WINDOW permet le passage de la fenêtre CHEMIN à celle du LOG.

Fenêtre n°2 : "CHOIXDAT".
A partir des tables sas situées dans le répertoire choisi dans "CHEMIN", l'utilisateur sélectionne (par X) la table retenue pour l'analyse.
Exemple : X BLAYO suivi de F3.

Fenêtre n°3 : "CHOIXVAR".
A partir de la liste des variables de la table sélectionnée dans "CHOIXDAT", l'utilisateur indique (par C) la variable qui identifie les observations (cette variable sera traitée comme étant de type Caractère) et (par N) les variables Numériques à retenir pour l'analyse.

Exemple :

CHOIX	*VARIABLE*
C	*PAYS*
N	*ANCRX*
N	*TXMORT*
N	*TXANAL*
N	*SCOL2*
N	*PIBH*
N	*CRXPIB*

suivi de F3.

Remarque: En cas de données manquantes un message apparaît dans l'OUTPUT indiquant leur nombre, leurs positions (numéros d'observations dans la table) et le programme s'arrête en indiquant qu'il faut utiliser KACPX.

Fenêtre n°4 : "STRUCTUR".
L'utilisateur choisit son type de réseau : F pour Ficelle, G pour Grille.
Exemple : choix : G suivi de F3.

Fenêtre n°5 : "PARAM_FI" ou "PARAM_GR".
Pour une FIcelle, "PARAM_FI" demande sa taille (n) et le nombre maximum d'itérations (tmax).
Pour une GRille, "PARAM_GR" demande le nombre de lignes (m), le nombre de colonnes (n) et le nombre maximum d'itérations (tmax).
Les nu unités du réseau (nu = n pour une ficelle ou nu = m×n pour une grille) seront numérotées de 1 à nu. Par exemple si nu=12 pour la ficelle de n=12, on aura : 1 2 ..... 11 12.
Pour la grille de m = 3 et n = 4 on aura :
1 4 7 10
2 5 8 11
3 6 9 12
On peut prendre pour tmax une valeur de l'ordre de 5 à 6 fois le nombre d'observations; ceci revient à dire qu'en moyenne chaque observation sera présentée 5 à 6 fois durant l'apprentissage. Notons que ce rapport peut être diminué si le nombre d'observations est très élevé.

Dans "PARAM_FI" comme dans "PARAM_GR" deux choix supplémentaires sont offerts.
Le premier choix (O/N) permet d'initialiser le générateur de nombre au hasard.
O : le point de départ est fixe, ce qui rend les résultats reproductibles (valeur par défaut).
N: le point de départ est calé sur l'horloge de la machine, ce qui rend les résultats NON reproductibles.
Le deuxième choix (O/N) propose le calcul et la représentation éventuels de la fonction "énergie". Cette "énergie" ou "potentiel" généralise la notion de "variance intra" en l'étendant aux plus proches voisins. Passé les trois quarts de tmax (à zéro voisin) les deux notions coïncident. En fin d'itérations, l'énergie doit se situer sur un minimum (local).
Le choix négatif (N) est la valeur par défaut; si le nombre d'itérations est élevé (dès 2000), un choix affirmatif (O) peut s'avérer coûteux en temps de calcul, il implique la création d'un graphique de nom : NRJ stocké dans un catalogue graphique permanent (cf. la fenêtre n°6 : "INFORM").
Quel que soit le choix retenu le programme affiche dans l'OUTPUT la valeur finale de la variance intra.
Exemple :

m = 8
n = 8
tmax = 300
CHOIX : O
CHOIX : O suivi de F3.

Fenêtre n°6 : "INFORM".
Dans cette fenêtre, l'utilisateur doit renseigner le champ Nom (fixé par défaut à _TEMPOR_) et décider (choix O ou N) de l'éventuel visualisation et stockage d'un graphique de type "dx-dy" (cf. : la thèse de P. DEMARTINES dont la référence est donnée dans la section III).
Le champ Nom sera utilisé pour nommer des catalogues (fichier d'extension .sas7bcat) et des tables sas (fichier d'extension .sas7bdat). Il doit comporter au moins 5 caractères; dans le cas contraire, le programme complète la réponse par des X pour obtenir un champ Nom de 5 caractères.
Le choix : N est proposé par défaut; en cas de réponse positive (O), un graphique de nom DX_DY sera créé dans le catalogue graphique. C'est un nuage de points dans un carré de côté 1, où l'on compare les distances (normalisées à 1) théoriques entre les unités gagnantes avec les distances (normalisées à 1) euclidiennes entre les vecteurs poids associés. La situation idéale correspondant à une organisation parfaite serait celle où tous les points (représentés sur le graphique par des cercles) sont situés sur la diagonale du carré.

Exemple :

Nom : B8G3CPAY
choix : O suivi de F3.

Ces réponses impliquent la création de 5 tables sas et de 2 catalogues qui seront placés dans C:\K8\ABOUL (cf.: le champ path de la fenêtre n°1 : "CHEMIN").

Description des 5 tables sas :

1. *la table B8G3C_CL (valeur par défaut : _TEMP_CL) donne, pour chaque unité gagnante, le contenu de la classe correspondante.*
2. *la table B8G3C_UG (valeur par défaut : _TEMP_UG) donne les valeurs moyennes des variables brutes pour chaque unité gagnante.*
3. *La table B8G3C_WS (valeur par défaut : _TEMP_WS) donne pour chaque unité (gagnante ou pas) sa position dans la grille : ligne, colonne, son effectif (zéro si l'unité n'est pas gagnante) et son "vecteur poids final " ou "vecteur représentant ".*
4. *La table B8G3CPAY (valeur par défaut : _TEMPOR_) donne pour chaque modalité (ici PAYS) de la variable (C) qui identifie les observations (cf. la fenêtre n°3 : "CHOIXVAR") son unité gagnante (sa classe) : _codage_ , sa position dans la grille : ligne, colonne et ses valeurs pour les variables brutes ainsi que pour les variables éventuellement transformées (cf.: la fenêtre n°8 "PREPROC").*
5. *La table INFO_B8G3CPAY (valeur par défaut : INFO_ _TEMPOR_) qui renseigne sur la structure du réseau (ici GRILLE) et sur l'algorithme (ici KACP).*

Description des 2 catalogues :

1. *Le catalogue iml B8G3CPAY (valeur par défaut : _TEMPOR_) contient tous les intermédiaires de calcul (matrices de poids initiaux et finaux, liste des unités gagnantes , etc.) qui seront ultérieurement utilisés dans les traitements complémentaires (cf. le paragraphe VIII).*
2. *Le catalogue graphique GACPB8G3 (valeur par défaut : GACP_TEM) contient 10 graphiques (au plus). Chaque graphique possède son nom et sa description qui correspond aux 40 premiers caractères de son titre.*

Les 10 graphiques du catalogue graphique GACPB8G3 :

Nom	*Description*
CELL	*Contenu des 64 cellules*
CELLW	*Valeur Moyenne & Représentant des 64 cellules*
DX_DY	*dx_dy*
G_DIMA	*Distances (M) avec les plus proches voisins*
G_PAVAGE	*KACP : grille 8x8 et 300 itérations*
HIS_GRI	*Représentants des classes (Poids Finaux)*
HIS_GRI1	*Moyennes des classes (valeurs normalisées)*
LIN_GRI	*Représentants des classes (Poids Finaux)*
LIN_GRI1	*Moyennes des classes (valeurs normalisées)*
NRJ	*Variance intra étendue aux voisins*

Les graphiques, dans le catalogue, sont classés en ordre alphabétique alors qu'à l'affichage l'ordre sera pour une grille :
G_PAVAGE, CELL, CELLW, G_DIMA, LIN_GRI, HIS_GRI, LIN_GRI1, HIS_GRI1, DX_DY et NRJ
pour une ficelle :
F_PAVAGE, F_DIMA, LIN_FIC, HIS_FIC, LIN_FIC1, HIS_FIC1, CELL, CELLW, DX_DY et NRJ

Le graphique CELL présente sous forme de pavage le contenu des unités (les valeurs numériques des variables, éventuellement transformées selon le choix de la fenêtre " PREPROC " et qui correspondent aux individus associés à une même unité gagnante) du réseau (grille ou ficelle).

Le graphique CELLW présente sous forme de pavage le vecteur représentant (poids final) de l'unité du réseau (grille ou ficelle) ainsi que la moyenne des variables dans le cas ou l'unité est gagnante (classe non vide). Il permet de comparer l'allure du vecteur représentant d'une unité gagnante avec son vecteur moyen. (l'individu moyen de cette classe non vide).

G_DIMA pour une Grille (ou F_DIMA pour une Ficelle) permet d'apprécier pour chaque unité (classe) son effectif et les distances de ***Mahalanobis*** *normalisées avec ses plus proches voisins (au plus 8 pour une grille et au plus 2 pour une ficelle). A l'exception des unités des bords, si l'unité est très proche de ses (8 ou 2) voisins, son polygone sera très proche des bords du carré. (Adapté de M. Cottrell & E. De Bodt, ESANN'96).*

G_PAVAGE pour une Grille (ou F_PAVAGE pour une Ficelle) donne l'illustration graphique du contenu des classes (unités) du réseau (cf.: la table B8G3C_CL).

HIS_GRI pour une Grille (HIS_FIC pour une Ficelle) est un pavage d'histogrammes (diagramme en bâtons) des représentants des classes (poids finaux).
HIS_GRI1 pour une Grille (HIS_FIC1 pour une Ficelle) est un pavage d'histogrammes des moyennes normalisées pour les classes non vides.

LIN_GRI pour une Grille (LIN_FIC pour une Ficelle) est un pavage de courbes des représentants des classes (poids finaux).
LIN_GRI1 pour une Grille (LIN_FIC1 pour une Ficelle) est un pavage de courbes des moyennes normalisées pour les classes non vides.
Il n'y a pas de graphique de pavage pour une ficelle de plus de 50 unités

Fenêtre n°7 : "INIT0".
L'utilisateur indique la façon d'initialiser les vecteurs poids initiaux
E : s'ils sont pris au hasard entre le min et le max (choix par défaut).
H : s'ils sont pris au hasard parmi les entrées.
F : s'ils sont pris dans un maillage du premier plan factoriel.

Exemple : choix : E suivi de F3.

Fenêtre n°8 : "PREPROC".
L'utilisateur choisit un éventuel type de prétraitement de ses données :
A : pour Aucun
C : pour Centrer les variables
N : pour centrer et réduire les variables (Normer)
K : pour transformer les lignes de la matrice des données en pourcentages de somme 1 (profils lignes) et utiliser la distance du Khi deux sur ces profils.
Le choix N (normer) est proposé par défaut

Exemple : choix : N suivi de F3.

PATIENCE KACP travaille pour vous !!!!
Avant d'afficher les graphiques du catalogue graphique IML, les 6 autres fichiers permanents sont créés.
LA SESSION KACP EST TERMINEE

L'ALGORITHME KACPX

C'est la version pour données manquantes de KACP. Il faut noter que KACPX produit 2 tables sas de plus que KACP nommées par défaut : _TEMP_CMP.sas7bdat et _TEMP_WMQ.sas7bdat.

*La première est une table qui contient les **données complétées**, ainsi que le nombre et le taux de valeurs manquantes pour chaque observation. Elle peut servir d'entrée à KACP avec une autre initialisation des vecteurs poids initiaux.*

La seconde table donne pour chaque unité du réseau son effectif, le nombre et le pourcentage de données manquantes pour cette unité, ainsi que ses variances « horizontale » et « verticale », ceci permet d'apprécier, d'une part, la dispersion des individus d'une même classe (unité gagnante) autour de leur vecteur représentant (aspect « verticale »), d'autre part d'apprécier la dispersion longitudinale du représentant de l'unité (aspect « horizontale »).

L'ALGORITHME KFAST *: Simple Competitive Learning (SCL).*

C'est la version stochastique de l'algorithme de FORGY, il s'agit d'une classification qui utilise l'algorithme de KOHONEN à zéro voisin sur toutes les étapes du processus.

L'ALGORITHME KBATCH

C est la version déterministe de l'algorithme de KOHONEN.

Le déroulement d'une session KACPX, KFAST ou KBATCH est similaire à celle de KACP : il suffit de charger et soumettre dans SAS : DKACPX.sas, DKFAST.sas ou DKBATCH.sas qui se situe dans c:\k8.

VI - L'ALGORITHME KORRESP

Algorithme KORRESP : Version Kohonen de l'analyse d'un tableau de contingence

On analyse un tableau de contingence croisant deux variables qualitatives, au moyen d'un algorithme dérivé de celui de Kohonen. (M.Cottrell et P.Letrémy, dans les Actes de Neuro Nîmes, 1994).

Pour illustrer le déroulement d'une session, nous utiliserons un tableau de contingence qui croise les monuments historiques en France, suivant leur catégorie (au nombre de p=11) et leur type de propriétaire (au nombre de q=6).
Ce tableau est stocké dans la table sas de nom MONUMENT qui possède p=11 observations et q+1=7 variables soit :
Une variable caractère nommée MONU dont les valeurs sont les p=11 catégories et q=6 variables numériques qui ont pour nom les q=6 types de propriétaires.
Détaillons les p=11 catégories de monuments :
preh (antiquités préhistoriques),
hist (antiquités historiques),
chat (châteaux),
mili (architecture militaire),
cath (cathédrales),
egli (églises),
chap (chapelles),
mona (monastères),
ecpu (édifices civils publics),
ecpr (édifices civils privés)
div (divers).
et les q=6 types de propriétaires :
COMM (commune),
PRIV (privé),
ETAT (état),
DEPA (département),
ETPU (établissement public)
NDET (non déterminé).

L'allure de la table MONUMENT (placée dans le répertoire c:\k8\aboul) est la suivante:

```
MONU  COMM  PRIV .........  NDET
preh   244   790 .........   144
hist   246   166 .........    31
.................................
ecpr   224   909 ..............4
div    967   242                9
```

DEBUT DE LA SESSION KORRESP

Charger et soumettre dans SAS : DKORR.sas qui se trouve dans c:\k8.
Toute la suite est CONVERSATIONNELLE, des fenêtres s'ouvrent et guident l'utilisateur dans ses choix. Pour toutes les fenêtres, les choix retenus seront validés par F3.

Fenêtre n°1 : "CHEMIN".
L'utilisateur précise le chemin d'accès aux données (par défaut c:\k8\aboul).

Exemple : path: C:\K8\ABOUL suivi de F3.
Dans le cas où le répertoire proposé est vide ou inexistant, un message d'erreur apparaît dans la fenêtre LOG, la commande WINDOW permet le passage de la fenêtre CHEMIN à celle du LOG.

Fenêtre n°2 : "CHOIXDAT".
A partir des tables sas situées dans le répertoire choisi dans "CHEMIN", l'utilisateur sélectionne (par X) la table retenue pour l'analyse.

Exemple : X MONUMENT suivi de F3.

Fenêtre n°3 : "SELECT".
L'utilisateur doit sélectionner les éléments du tableau de contingence.

C : Pour l'identificateur des lignes du tableau de contingence.
N : Pour les colonnes (variables numériques) du tableau de contingence.

Exemple :

CHOIX	*VARIABLE*	
C	*MONU*	
N	*COMM*	
N	*PRIV*	
N	*ETAT*	
N	*DEPA*	
N	*ETPU*	
N	*NDET*	*suivi de F3.*

Remarque: En cas de données manquantes un message apparaît dans l'OUTPUT indiquant leur nombre, leurs positions (numéros d'observations dans la table) et le programme s'arrête.

En absence de donnée manquante, le programme calcule et affiche dans la fenêtre "OUTPUT" la statistique du khi_deux et sa "p-value". Si la p-value est > 5% alors un message d'avertissement apparaît dans la fenêtre "OUTPUT" qui explique en quoi l'analyse du tableau de contingence n'est pas très pertinente mais le programme KORRESP continue.

Fenêtre n°4 : "STRUCTUR".
L'utilisateur choisit son type de réseau : F pour Ficelle, G pour Grille.

Exemple : choix : G suivi de F3.

Fenêtre n°5 : "PARAM_FI" ou "PARAM_GR".
Pour une FIcelle, "PARAM_FI" demande sa taille (n) et le nombre maximum d'itérations (tmax).
Pour une GRille, "PARAM_GR" demande le nombre de lignes (m), le nombre de colonnes (n) et le nombre maximum d'itérations (tmax).
Les nu unités du réseau (nu = n pour une ficelle ou nu = m×n pour une grille) seront numérotées de 1 à nu. Par exemple si nu=12 pour la ficelle de n=12, on aura : 1 2 ..... 11 12.

Pour la grille de m = 3 et n = 4 on aura :
1 4 7 10
2 5 8 11
3 6 9 12

Dans "PARAM_FI" comme dans "PARAM_GR" deux choix supplémentaires sont offerts.
Le premier choix (O/N) permet d'initialiser le générateur de nombre au hasard.

O : le point de départ est fixe, ce qui rend les résultats reproductibles (valeur par défaut).

N: le point de départ est calé sur l'horloge de la machine, ce qui rend les résultats NON reproductibles.

Le deuxième choix (O/N) propose le calcul et la représentation éventuels de la fonction "énergie". Cette "énergie" ou "potentiel" généralise la notion de "variance intra" en l'étendant aux plus proches voisins (à zéro voisin les deux notions coïncident). En fin d'itérations, elle doit se situer sur un minimum (local).
Le choix négatif (N) est la valeur par défaut; si le nombre d'itérations est élevé (dès 1000), un choix affirmatif (O) peut s'avérer coûteux en temps de calcul, il implique la création de 2 graphiques (nommés NRJ et NRJ1) stockés dans un catalogue graphique permanent (cf. la fenêtre n°6 : "INFORM").
Quel que soit le choix retenu le programme affiche dans l'OUTPUT la valeur finale de la variance intra pour les profils lignes et colonnes.

Exemple :

m = 5
n = 5
tmax = 300
CHOIX : O
CHOIX : O suivi de F3.

Si le tableau de contingence a p×q modalités, on peut prendre pour :

- une ficelle, n au moins égal à max(p,q)
- une grille, m = n, où n est tel que n×n est immédiatement supérieur à 2 max(p,q).

Ici p=11 et q=6, on prend une grille 5×5 puisque 25 est le carré immédiatement supérieur à 22.

Fenêtre n°6 : "INFORM".
Dans cette fenêtre, l'utilisateur doit renseigner le champ Nom (fixé par défaut à _ TEMPOR _) et décider (choix O/N) de l'éventuel visualisation et stockage de deux graphiques de type "dx-dy" (cf. : la thèse de P. DEMARTINES dont la référence est donnée dans la section III).
Le champ Nom sera utilisé pour nommer des catalogues (fichier d'extension .sas7bcat) et des tables sas (fichier d'extension .sas7bdat). Il doit comporter au moins 5 caractères; dans le cas contraire, le programme complète la réponse par des X pour obtenir un champ Nom de 5 caractères.
Le choix : N est proposé par défaut; en cas de réponse positive (O), deux graphiques de noms DX_DY et DX_DY1 seront créés et stockés dans le catalogue graphique. Un graphique de type "dx-dy" est un nuage de points dans un carré de côté 1, où l'on compare les distances (normalisées à 1) théoriques entre les unités gagnantes avec les distances (normalisées à 1) euclidiennes entre les vecteurs poids associés.
La situation idéale correspondant à une organisation parfaite serait celle où tous les points (représentés sur le graphique par des cercles) sont situés sur la diagonale du carré.

Exemple :

Nom : MO3C5G
choix : O suivi de F3.

Ces réponses impliquent la création de 4 tables sas et de 2 catalogues qui seront placés dans C:\K8\ABOUL (cf.: le champ path de la fenêtre n°1 : "CHEMIN").

Description des 4 tables sas :

1. *La table MO3C5_CL (valeur par défaut : _TEMP_CL) donne, pour chaque unité gagnante, les modalités du croisement dont elle est la référence (contenu de la classe).*
2. *La table MO3C5_WS (valeur par défaut : _TEMP_WS) donne pour chaque unité (gagnante ou pas) sa position dans la grille : ligne, colonne, son effectif (zéro si l'unité n'est pas gagnante) et son "vecteur poids final concaténé " ou "vecteur représentant ".*
3. *La table MO3C5G (valeur par défaut : _TEMPOR_) donne pour chaque modalité du croisement son unité gagnante (sa classe) :_codage_ , sa position dans la grille : ligne, colonne. Dans le prolongement des modalités du croisement (ici 11+6=17 valeurs) se place la table de BURT.*
4. *La table INFO_MO3C5G (valeur par défaut : INFO_ _TEMPOR_) qui renseigne sur la structure du réseau (ici GRILLE) et sur l'algorithme (ici KORRESP).*

Description des 2 catalogues :

1. *Le catalogue iml MO3C5G (valeur par défaut : _TEMPOR_) contient tous les intermédiaires de calcul (matrices de poids initiaux et finaux, liste des unités gagnantes, etc.) qui seront ultérieurement utilisés dans les traitements complémentaires (cf. le paragraphe VIII).*
2. *Le catalogue graphique GKORMO3C (valeur par défaut : GKOR_TEM) contient 10 graphiques (au plus). Chaque graphique possède son nom et sa description qui correspond aux 40 premiers caractères de son titre.*

Les 10 graphiques du catalogue graphique GKORMO3C :

Nom	*Description*
CELLW	*Profil colonne Moyen & Représentant des 25 cellules*
CELLW1	*Profil ligne Moyen & Représentant des 25 cellules*
DX_DY	*Dx_dy pour les colonnes*
DX_DY1	*Dx_dy pour les lignes*
G_DIST	*Distances (E) avec les plus proches voisins*
G_PAVAGE	*KORRESP : grille 5x5 et 300 itérations*
LIN_GRI	*Représentants (Poids Finaux) pour les colonnes*
LIN_GRI1	*Représentants (Poids Finaux) pour les lignes*
NRJ	*Potentiel pour les colonnes*
NRJ1	*Potentiel pour les lignes*

Les graphiques, dans le catalogue, sont classés en ordre alphabétique alors qu'à l'affichage l'ordre sera
Pour une grille:
G_PAVAGE, LIN_GRI, CELLW, LIN_GRI1, CELLW1, G_DIST, DX_DY, DX_DY1, NRJ et NRJ1
Pour une ficelle:
F_PAVAGE, LIN_FIC, CELLW, LIN_FIC1, CELLW1, F_DIST, DX_DY, DX_DY1, NRJ et NRJ1

Les graphiques CELLW(1) présentent sous forme de pavage le vecteur représentant pour les profils colonnes(lignes) de l'unité du réseau (grille ou ficelle) ainsi que le profil colonne(ligne) moyen dans le cas ou l'unité est gagnante (classe non vide). Il permet de comparer l'allure du vecteur représentant d'une unité gagnante avec son profil moyen. (la modalité moyenne de cette classe non vide).

G_DIST pour une Grille (ou F_DIST pour une Ficelle) permet d'apprécier pour chaque unité (classe) son effectif et les distances euclidiennes normalisées avec ses plus proches voisins (8 pour une grille et 2 pour une ficelle). A l'exception des unités des bords, si l'unité est très proche de ses (8 ou 2) voisins, son polygone sera très proche des bords du carré (cf. la référence de M. Cottrell & E. De Bodt, ESANN'96).

G_PAVAGE pour une Grille (ou F_PAVAGE pour une Ficelle) donne l'illustration graphique du contenu des classes (unités) du réseau (cf. : la table MO3C5_CL).

LIN_GRI pour Grille (ou LIN_FIC pour une Ficelle) est un pavage de courbes des vecteurs "poids finaux" ou "représentants" associés aux profils colonnes.

LIN_GRI1 pour Grille (ou LIN_FIC1 pour une Ficelle) est un pavage de courbes des vecteurs "poids finaux" ou "représentants" associés aux profils lignes.

Il n'y a pas de graphique de pavage pour une ficelle de plus de 50 unités.

PATIENCE KORRESP travaille pour vous !!!!

Avant d'afficher les graphiques du catalogue graphique IML, les 5 autres fichiers permanents sont créés.

LA SESSION KORRESP EST TERMINEE

VII - LES ALGORITHMES KACM, KACM1 , KACM2 ET KDISJ

Algorithmes KACM(j) (j=1,2) et KDISJ
Versions Kohonen de l'Ana1yse des Correspondances Multiples.

Il s'agit d'une analyse des relations entre plusieurs variables qualitatives. cf. la références de M.Cottrell et S.Ibbou, de l'ESANN'95 pour KACM(j=1,2) et celle de M.Cottrell et P.Letrémy de l'ESANN'2003 pour KDISJ.

Pour illustrer le déroulement d'une session, nous utiliserons les données de BREFORT,1982 (cité dans l'ouvrage de G. SAPORTA "Probabilités, analyse des données et statistique").
*On dispose pour **27** races de chiens de 7 variables qualitatives :*
1. *TAILLE dont les 3 modalités sont : taille1, taille2 et taille3*
2. *POIDS dont les 3 modalités sont : poids1, poids2 et poids3*
3. *VELOCITE dont les 3 modalités sont : veloce1, veloce2 et veloce3*
4. *INTELLIG(ence) dont les 3 modalités sont : malin1, malin2 et malin3*
5. *AFFECTIO(n) dont les 2 modalités sont : affec1 et affec2*
6. *AGRESSIV(ité) dont les 2 modalités sont : agres1 et agres2*
7. *FONCTION dont les 3 modalités sont : chasse, compagnie et garde*

*(soit, au total, **19** modalités différentes).*

Ces données sont dans la table sas CHIENS qui est placée dans le répertoire 'c:\k8\aboul, et dont l'allure est la suivante:

```
RACE       TAILLE   POIDS   VELOCITE  INTELLIG  AFFECTIO AGRESSIV FONCTION
BEAUCERON  taille3  poids2  veloce3   malin3    affec2   agres2   garde
.....................................................................
```

Notons que le programme DTAIL.sas appliqué à la table CHIENS produit le tableau disjonctif complet (CHIEN_DJ.sas7bdat) et la table de burt (CHIEN_BT.sas7bdat)

DEBUT DE LA SESSION KACM(j=1,2) ou KDISJ

Charger et soumettre dans SAS : DKACM.sas qui se situe dans c:\k8.
Toute la suite est CONVERSATIONNELLE, des fenêtres s'ouvrent et guident l'utilisateur dans ses choix. Pour toutes les fenêtres, les choix retenus seront validés par F3.

Fenêtre n°1 : "CHEMIN".
L'utilisateur précise le chemin d'accès aux données (par défaut c:\k8\aboul).

Exemple : path: C:\K8\ABOUL suivi de F3.

Dans le cas où le répertoire proposé est vide ou inexistant, un message d'erreur apparaît dans la fenêtre LOG, la commande WINDOW permet le passage de la fenêtre CHEMIN à celle du LOG.

Fenêtre n°2 : "CHOIXDAT".
A partir des tables sas situées dans le répertoire choisi dans "CHEMIN", l'utilisateur sélectionne (par X) la table retenue pour l'analyse.

Exemple : X CHIENS suivi de F3.

Fenêtre n°3 : "CHOIX".
En premier lieu, l'utilisateur précise la nature des données :

- *Dans le cas où la table sas correspond à un tableau de Contingence.* *par : C*

(cette option permet de comparer les résultats obtenus avec KORRESP)

- *Dans le cas où la table sas correspond aux Réponses d'individus.* *par : R*
- *Dans le cas où la table sas correspond à un tableau Disjonctif complet.* *par : D*
- *Dans le cas où la table sas correspond à un tableau de Burt.* *par : B*

Ensuite, l'utilisateur précise le type des observations :

- *par : O* *si elles sont "anonymes" (valeur par défaut), seules les modalités seront prises en compte :* ***KACM***
- *par : N* *si les observations sont "NON anonymes", leurs identifiants seront traitées avec les modalités :* ***KACM1, KACM2*** *ou* ***KDISJ***

Dans les cas C (tableau de contingence) ou B (tableau de Burt), la question ne se posant pas, l'utilisateur est contraint de garder la réponse par défaut (O).

Enfin, l'utilisateur identifie la nature des variables :

- *par : K*
- *par : X*

Le choix de K est unique et il identifie une variable considérée comme étant de type caractère.
Dans les cas R (tableau de réponses) ou D (disjonctif complet), K correspond à l'identificateur des observations (individus, répondants).
Dans les cas C (contingence) ou B (Burt), K correspond à la variable qui identifie les lignes ou les modalités du tableau.
Plusieurs choix sont possibles pour X :
Dans les cas C (contingence), D (disjonctif) ou B (Burt), l'utilisateur est contraint de prendre toutes les variables restantes dans la liste.
Dans le cas R (tableau de réponses), le choix de 2 variables est l'option minimum.

Exemple :

CHOIX	*R*	
CHOIX	*N*	
CHOIX	*VARIABLE*	
K	*RACE*	
X	*TAILLE*	
X	*POIDS*	
X	*VELOCITE*	
X	*INTELLIG*	
X	*AFFECTIO*	
X	*AGRESSIV*	
X	*FONCTION*	*suivi de F3.*

Si les observations sont "NON anonymes".
La Fenêtre n°4 : "KELKACMJ".
Propose à l'utilisateur de choisir entre KACM1 (Choix : A), KACM2 (Choix : B) et KDISJ (Choix : C)

Pour KACM1 *l'apprentissage et le classement initial s'effectuent sur les individus à l'aide du tableau disjonctif complet "corrigé" puis les modalités sont présentées comme données supplémentaires à l'aide de la table de Burt "corrigée".*
Pour_KACM2 *l'apprentissage et le classement initial s'effectuent sur les modalités à l'aide de la table de Burt "corrigée" puis les individus sont présentés comme données supplémentaires à l'aide du tableau disjonctif "corrigé" (à l'instar de l'ACM les modalités sont classées de façon identique pour KACM et KACM2).*
Pour_KDISJ *l'apprentissage et le classement s'effectuent* ***simultanément*** *sur les modalités et les individus à l'aide du tableau disjonctif complet "corrigé".*

Exemple : choix : C suivi de F3.

Remarque: En cas de données manquantes un message apparaît dans l'OUTPUT indiquant leur nombre, leurs positions (numéros d'observations dans la table) et le programme s'arrête.

Comme pour KORRESP, si KACM traite un tableau de contingence (C), le programme calcule et affiche dans la fenêtre "OUTPUT" la statistique du khi2 et sa "p-value". Si la p-value est > 5% alors un message d'avertissement apparaît dans la fenêtre "OUTPUT" qui explique en quoi l'analyse du tableau de contingence n'est pas très pertinente mais le programme KACM continue.

Fenêtre n°5 : "STRUCTUR".
L'utilisateur choisit son type de réseau : F pour Ficelle, G pour Grille.
Exemple : choix : G suivi de F3.

Fenêtre n°6 : "PARAM_FI" ou "PARAM_GR".
Pour une FIcelle, "PARAM_FI" demande sa taille (n) et le nombre maximum d'itérations (tmax).
Pour une GRille "PARAM_GR" demande le nombre de lignes (m), le nombre de colonnes (n) et le nombre maximum d'itérations (tmax).
Les nu unités du réseau (nu = n pour une ficelle ou nu = m×n pour une grille) seront numérotées de 1 à nu. Par exemple si nu=12 pour la ficelle de n=12, on aura : 1 2 ..... 11 12.
Pour la grille de m = 3 et n = 4 on aura :

1 4 7 10
2 5 8 11
3 6 9 12

Dans "PARAM_FI" comme dans "PARAM_GR" deux choix supplémentaires sont offerts.
Le premier choix (O/N) permet d'initialiser le générateur de nombre au hasard.
O : le point de départ est fixe, ce qui rend les résultats reproductibles (valeur par défaut).
N: le point de départ est calé sur l'horloge de la machine, ce qui rend les résultats NON reproductibles.
Le deuxième choix (O/N) propose le calcul et la représentation éventuels de la fonction "énergie". Cette "énergie" ou "potentiel" généralise la notion de "variance intra" en l'étendant aux plus proches voisins (à zéro voisin les deux notions coïncident). En fin d'itérations, elle doit se situer sur un minimum (local).
Le choix négatif (N) est la valeur par défaut, si le nombre d'itérations est élevé (dès 1000), un choix affirmatif (O) peut s'avérer coûteux en temps de calcul, il implique l'affichage dans l'OUTPUT de la valeur finale de la variance intra ainsi que la création de 2 graphiques de nom : NRJ, NRJ1 stockés dans un catalogue graphique permanent (cf. la fenêtre n°6 : "INFORM").

Exemple : *m = 5*
n = 5
tmax = 700
CHOIX : O
CHOIX : O *suivi de F3.*

Si q est le nombre total de modalités et r le nombre d'individus, on peut prendre pour une ficelle, n de l'ordre de q/2 et pour une grille où m = n, n tel que n×n est immédiatement supérieur à q .
Pour tmax une valeur de l'ordre de 15 à 20 fois le nombre d'individus+le nombre de modalités

Ici q=19, r=27 donc on prend une grille 5×5, puisque 25 est le carré immédiatement supérieur à 19 et tmax = 700 > 15(27+19).*

Fenêtre n°7 : "INFORM".
Dans cette fenêtre, l'utilisateur doit renseigner le champ Nom (fixé par défaut à _ TEMPOR _) et décider (choix O ou N) de l'éventuel visualisation et stockage d'un graphique de type "dx-dy" (cf. : la thèse de P. DEMARTINES dont la référence est donnée dans la section III).

Le champ Nom sera utilisé pour nommer des catalogues (fichier d'extension .sas7bcat) et des tables sas (fichier d'extension .sas7bdat). Il doit comporter au moins 5 caractères; dans le cas contraire le programme complète la réponse par des X pour obtenir un champ Nom de 5 caractères.

Le choix : N est proposé par défaut; en cas de réponse positive (O), un graphique de nom DX_DY sera créé dans le catalogue graphique. C'est un nuage de points dans un carré de côté 1, où l'on compare les distances (normalisées à 1) théoriques entre les unités gagnantes avec les distances (normalisées à 1) euclidiennes entre les vecteurs poids associés. La situation idéale correspondant à une organisation parfaite serait celle où tous les points (représentés sur le graphique par des cercles) sont situés sur la diagonale du carré.

Exemple : *Nom : D7C5G*
choix : O *suivi de F3.*

Ces réponses impliquent la création de 6 tables sas et de 2 catalogues qui seront placés dans C:\K8\ABOUL (cf.: le champ path de la fenêtre n°1 : "CHEMIN").

Description des 6 tables sas :

1. *La table D7C5G _CL (valeur par défaut : _TEMP_CL) donne, pour chaque unité gagnante, les modalités (caractéristiques des chiens) ainsi que les identifiants des individus (race des chiens) dont elle est la référence, ceci dans le cas d'un tableau de Réponses ou d'un tableau Disjonctif portant sur des individus Non anonymes (CHOIX R ou D et N dans la fenêtre n°3 : "CHOIX").*
2. *La table D7C5G _WS (valeur par défaut : _TEMP_WS) donne pour chaque unité (gagnante ou pas) sa position dans la grille : ligne, colonne, son effectif (zéro si l'unité n'est pas gagnante) et son "vecteur poids final " ou "vecteur représentant ".*
3. *La table D7C5G (valeur par défaut : _TEMPOR_) donne pour chaque modalité son unité gagnante (sa classe) :_codage_ , sa position dans la grille : ligne, colonne. Dans le prolongement des modalités (ici 19 valeurs) se place la table de BURT. Dans le cas d'un tableau de Réponses ou d'un tableau Disjonctif portant sur des individus Non anonymes, la table donne en plus, pour chaque modalité identifiant un individu son unité gagnante (sa classe) :_codage_, sa position dans la grille : ligne, colonne. Dans le prolongement des identifiants des individus se place le tableau disjonctif complet.*
4. *La table D7C5G_IN (valeur par défaut : _TEMP_IN) donne pour chaque modalité identifiant un individu son unité gagnante (sa classe) :_codage_, sa position dans la grille : ligne, colonne. Dans le*

*prolongement des identifiants des individus (ici 27 races de chiens) se place le tableau disjonctif complet **corrigé** qui sert ici d'entrée (_IN) à l'algorithme KDISJ.*

5. *La table D7C5G _NB (valeur par défaut : _TEMP_NB) donne, pour chaque unité de la structure (grille ou ficelle) le nombre d'individus déclarés non anonymes et le nombre de modalités ainsi que leur total.*
6. *La table INFO_D7C5G (valeur par défaut : INFO_ _TEMPOR_) qui renseigne sur la structure du réseau (ici GRILLE) et sur l'algorithme (ici KDISJ).*

Description des 2 catalogues :

1. *Le catalogue iml D7C5G (valeur par défaut : _TEMPOR_) contient tous les intermédiaires de calcul (matrices de poids initiaux et finaux, liste des unités gagnantes, etc.) qui seront ultérieurement utilisés dans les traitements complémentaires (cf. le paragraphe VIII).*
2. *Le catalogue graphique GACMD7C5 (valeur par défaut : GACM_TEM) contient 12 graphiques (au plus). Chaque graphique possède son nom et sa description qui correspond aux 40 premiers caractères de son titre.*

Les 12 graphiques du catalogue graphique GACMD6C5 :

Nom	*Description*
CELLW	*Individu Moyen et Représentant des 25 cellules*
CELLW1	*Modalité Moyenne et Représentant des 25 cellules*
DX_DY	*dx_dy pour les individus*
DX_DY1	*dx_dy pour les modalités*
G_DIMA	*Distances (M) avec les plus proches voisins*
G_DIST	*Distances (E) avec les plus proches voisins*
G_PAVAG1	*KDISJ : grille 5x5 et 300 itérations*
G_PAVAGE	*KDISJ : grille 5x5 et 300 itérations*
LIN_GRI	*Représentants (Poids Finaux) pour les individus*
LIN_GRI1	*Représentants (Poids Finaux) pour les modalités*
NRJ	*Variance intra étendue aux voisins pour les individus*
NRJ1	*Variance intra étendue aux voisins pour les modalités*

Les graphiques, dans le catalogue, sont classés en ordre alphabétique alors qu'à l'affichage l'ordre sera
Pour une grille :
G_PAVAGE(1), LIN_GRI, CELLW, LIN_GRI1, CELLW1, G_DIMA, G_DIST, DX_DY(1), NRJ(1)
Pour une ficelle :
F_PAVAGE(1), LIN_FIC, CELLW, LIN_FIC1, CELLW1, F_DIMA, F_DIST, DX_DY(1), NRJ(1)

Les graphiques CELLW(1) présentent sous forme de pavage le vecteur représentant pour les individus(modalités) de l'unité du réseau (grille ou ficelle) ainsi que l'individu moyen(la modalité moyenne) dans le cas ou l'unité est gagnante (classe non vide). Il permet de contrôler la qualité de la convergence de l'algorithme.

*G_DIMA pour une Grille (ou F_DIMA pour une Ficelle) permet d'apprécier pour chaque unité (classe) son effectif et les distances de **Mahalanobis** normalisées avec ses plus proches voisins (au plus 8 pour une grille et au plus 2 pour une ficelle). A l'exception des unités des bords, si l'unité est très proche de ses (8 ou 2) voisins, son polygone sera très proche des bords du carré.*

G_DIST pour une Grille (ou F_DIST pour une Ficelle) permet d'apprécier pour chaque unité (classe) son effectif et les distances euclidiennes normalisées avec ses plus proches voisins (8 pour une grille et 2 pour une ficelle). A l'exception des unités des bords, si l'unité est très proche de ses (8 ou 2) voisins, son polygone sera très proche des bords du carré.

G_PAVAGE pour une Grille (ou F_PAVAGE pour une Ficelle) donne l'illustration graphique du contenu des classes (les modalités pour KACM et les individus en plus pour KACM1, KACM2 et KDISJ) du réseau (cf. D7C5G _CL).
G_PAVAG1 pour une Grille (ou F_PAVAG1 pour une Ficelle) donne l'illustration graphique des seules modalités.

LIN_GRI pour une Grille (LIN_FIC pour une Ficelle) est un pavage de courbes des représentants (poids finaux) pour les individus

LIN_GRI1 pour une Grille (LIN_FIC1 pour une Ficelle) est un pavage de courbes des représentants (poids finaux) pour les modalités

Il n'y a pas de graphique de pavage pour une ficelle de plus de 50 unités.

PATIENCE KACM travaille pour vous !!!!

Avant d'afficher les graphiques du catalogue graphique IML, les 7 autres fichiers permanents sont créés.

LA SESSION KACM EST TERMINEE

VIII - TRAITEMENTS COMPLEMENTAIRES

A) GENERALITES CONCERNANT LES MACROS

Les 20 macros (MKLUG.sas, MKLUF1.sas, MKLUF2.sas, MSTAT.sas, MCONT.sas, MKLOBS.sas, MKOC.sas, MSCAL.sas, MSCOL.sas, MSUPP.sas, MSUPPX.sas, MGVAR.sas, MVC3D.sas, MKAMG.sas, MKAMF1.sas, MKAMF2.sas, MDIMA.sas, MDIST.sas, MFMOD.sas, MFIND.sas) sont des traitements qui peuvent être envisagés à la suite des programmes KACP, KACPX, KFAST, KBATCH, KORRESP, KACM(j=1,2) ou KDISJ .

***A chaque session**, la soumission de KOMAC.sas, situé dans C:\K8, permet de les compiler et de les stocker dans le catalogue permanent SASMACR.sas7bcat placé dans C:\K8\AKOR.*

*L'appel d'une macro s'obtient en soumettant l'instruction **%nom_de_la_macro** dans la fenêtre PROGRAM EDITOR*

exemple : %MKLUG

Chaque macro est conversationnelle et s'ouvre sur une fenêtre où est indiqué son domaine d'application: COMPLEMENT(S) TOUT PROGRAMME signifie qu'elle est applicable aux sorties des 9 programmes KACP, KACPX, KFAST, KBATCH, KORRESP, KACM(j=1,2)) ou KDISJ .

C'est par le biais du champ " type du catalogue" que l'utilisateur précise le type de programme (gacp pour KACP, KACPX, KFAST, et KBATCH, gkor pour KORRESP et gacm pour KACM(j=1,2) et KDISJ).

Le vocable : TOUTE STRUCTURE signifie que le traitement peut s'appliquer aussi bien à une ficelle qu'à une grille.

Par exemple pour la macro MKLUG, il est indiqué dans la fenêtre " SURCLASS " :
COMPLEMENTS TOUT PROGRAMME POUR UNE GRILLE

Suite au domaine d'application, il est donné une brève description de la nature du traitement.
Par exemple pour la macro MKLUG il est indiqué :
=>>> PAVAGE EN SUPER CLASSES (CLUSTERS)

Deux champs sont communs à toutes ces macros : " librairie " et " nom commun ".

La valeur par défaut de " librairie " est C:\K8\ABOUL, ce champ est équivalent au champ "path " de la fenêtre " CHEMIN "; il correspond au chemin d'accès des données et de certains catalogues (comme le catalogue graphique).
La valeur par défaut de " nom commun " est _TEMPOR_; ce champ est équivalent au champ " Nom " de la fenêtre " INFORM "; il correspond au nom utilisé pour nommer des tables sas et le catalogue graphique.

Pour abandonner l'exécution de la macro, il suffit de taper O dans le champ " STOP ".
En bas de la fenêtre de chaque macro il est stipulé que la touche F3 valide les choix et qu'en cas de problèmes de cohérence dans la saisie des réponses, l'utilisateur reste dans la fenêtre de la macro avec les réponses par défaut, il doit consulter les messages d'erreurs de la fenêtre LOG avant de reprendre la saisie dans la fenêtre de la macro.

B) DESCRIPTION ET EXEMPLE D'APPLICATION POUR CHAQUE MACRO

MKLUG : *(fenêtre " SURCLASS ", domaine :* ***tout programme pour une grille*** *).*

Création de Super-Classes via une Classification Ascendante Hiérarchique (cf. PROC CLUSTER avec la méthode de Ward) appliquée aux vecteurs poids finaux (représentants) des unités de la grille.

Dans le cas de KORRESP, KACM, KACM1, KACM2 et KDISJ on tient compte dans la CAH des effectifs dans les unités de la grille et on adjoint aux représentants leurs coordonnées dans la grille.

Cette macro produit au plus 7 graphiques rajoutés au catalogue graphique déjà existant (cf.: le champ "nom commun ") et une table sas placée dans le répertoire des données (cf.: le champ "librairie").

Description des 7 graphiques :

Le graphique nommé TREE, qui représente le dendrogramme associé à la classification hiérarchique ascendante, est précédé de l'historique de la classification qui apparaît dans la fenêtre OUTPUT et permet de confirmer a posteriori le choix du nombre de super-classes.

Le graphique nommé CLUSPAV représente la grille, où dans chaque unité figure le vecteur poids final (le représentant) avec le numéro de son cluster

Les 5 autres graphiques nommés : CLUSCOUL, CLUSCOLH, CLUSCOL1, CLUSDIMA et CLUSDIST ne seront produits que dans la mesure où l'utilisateur a choisi (cf. le champ " nombre de super-classes " de la fenêtre " SURCLASS ") un nombre de clusters qui ne dépasse pas 22, qui est le nombre de couleurs disponibles (cf. le programme KOLCLUS.sas dans C:\K8).

Le graphique nommé CLUSCOUL est la version couleur de CLUSPAV .
Dans les graphique nommés CLUSCOLT et (ou) CLUSCOL1, figurent les "valeurs" associées aux unités de la grille ainsi que les couleurs des clusters.

Par "valeurs" on entend :

- *les individus pour KACP, KACPX, KFAST ou KBATCH*
- *les modalités pour KORRESP et KACM*
- *les modalités et les individus pour KACM1, KACM2 et KDISJ*

Ainsi, pour KACM1, KACM2 et KDISJ, nous aurons deux graphiques du même type (CLUSCOLH et CLUSCOL1) le premier dédié aux seules modalités et le second comprenant modalités et individus.

Dans le graphique nommé CLUSDIMA, on colorie le graphique G_DIMA avec les couleurs associées aux clusters (super-classes).
Dans le graphique nommé CLUSDIST, on colorie le graphique G_DIST avec les couleurs associées aux clusters (super-classes).

La table sas nommée CLUSTn (où n correspond au nombre de clusters retenu par l'utilisateur) donne, pour chaque unité de la grille, son numéro de cluster ainsi que son vecteur poids final. Cette table sas sera utilisée par les macros MSTAT, MCONT, MKLOBS, MKOC, MSCOL, MVC3D, MKAMG, MFIND et MFMOD, qui effectuent des représentations graphiques ou des calculs impliquant les clusters.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
type du catalogue :	*GACP*	
nombre de super-classes :	*5*	*suivi de F3*

Ces réponses impliquent la création du fichier C:\K8\ABOUL\CLUST5.sas7bdat, et portent à 13 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MKLUF1 : *(fenêtre " SURFIC10 ", domaine :* ***tout programme pour une ficelle (<=10)****).*

MKLUF2 : *(fenêtre " SURFICEL ", domaine :* ***tout programme pour une ficelle (>10 et <=50)*** *).*

Ces deux macros effectuent sur une ***ficelle*** *un traitement qui est équivalent à celui de MKLUG pour une grille.*

Dans le cas de KORRESP, KACM, KACM1, KACM2 et KDISJ on tient compte dans la CAH des effectifs dans les unités de la ficelle et on adjoint aux représentants leurs positions dans la ficelle.

Les graphiques :
CLUSPAV, CLUSCOUL, CLUSTCOLH, CLUSTCOL1, CLUSDIMA, CLUSDIST.
Deviennent respectivement :

Pour MKLUF1:
CLUFPA10, CLUFCO10, CLUFCT10, CLUFCT11, CLUFDM10, CLUFDE10.

Pour MKLUF2:
CLUFPAV, CLUFCOUL, CLUFCTXT, CLUFCTX1, CLUFDMAH, CLUFDEUC.

La table sas nommé CLUSFn (l'équivalent de la table CLUSTn) donne pour chaque unité de la ficelle son numéro de cluster ainsi que son vecteur poids final. Cette table sera utilisée par les macros MSTAT, MCONT, MKLOBS, MKOC, MSCOL, MVC3D, MKAMFj (j=1,2), MFIND et MFMOD qui effectuent des représentations graphiques ou des calculs impliquant les clusters.

MSTAT : *(fenêtre " STATCLUS ", domaine :* ***KACP, KACPX, KFAST, KBATCH et toute structure*** *).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2), elle considère les super-classes (la variable cluster) comme une variable de classification et affiche dans la fenêtre OUTPUT les moyennes conditionnelles des variables brutes (cf. la sélection de la fenêtre " CHOIXVAR " dans KACP(X), KFAST, KBATCH).

Pour chaque variable figure dans L'OUTPUT la décomposition de la variance, la statistique de Fisher et la P-value associée. Enfin, y apparaissent les tests multidimensionnels de Wilks et Hotelling avec leur valeur approchée en terme de khi-deux ainsi que leur p-value.

La macro MSTAT produit 4 tables sas qui sont placées dans le répertoire des données (cf.: le champ "librairie").

La table B8G3_MOY (valeur par défaut : _TEM_MOY) où sont stockées les moyennes conditionnelles de la fenêtre OUTPUT.
La table B8G3_ANO (valeur par défaut : _TEM_ANO) où sont stockées les décompositions de la variance de la fenêtre OUTPUT.
La table B8G3_MAN (valeur par défaut : _TEM_MAN) où sont stockés les tests multidimensionnels de la fenêtre OUTPUT

Enfin dans le cas d'une Grille (respectivement d'une Ficelle) la table sas nommée CLASGn (respectivement CLASFn), où n correspond au nombre de clusters retenu dans MKLUG (respectivement dans MKLUFj (j=1,2)).

La table CLASGn (respectivement CLASFn) fournit pour chaque observation (ici chaque PAYS) ses numéros de cluster et de classe (unité gagnante) ainsi que ses valeurs pour les variables brutes et les variables "éventuellement prétraitées" qui servent d'entrées à KACP, KFAST ou KBATCH.

Dans le cas de KACPX ce sont les ***données complétées*** *qui sont prises, ainsi que le nombre et le taux de valeurs manquantes de chaque observation.*

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Outre les sorties dans la fenêtre OUTPUT, ces réponses impliquent la création de 4 fichiers: CLASG5.sas7bdat, B8G3_MOY.sas7bdat, B8G3_ANO.sas7bdat et B8G3_MAN.sas7bdat, placés dans le répertoire C:\K8\ABOUL\.

A noter que, dans le cas où tous les individus seraient placés dans la même superclasse, un message s'affiche dans l'output et la macro MSTAT s'arrête.

MCONT : *(fenêtre " REPCLUST ", domaine :* ***tout programme et toute structure****).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2); elle ajoute un graphique nommé REPCLU au catalogue graphique déjà existant.

Ce graphique représente sous forme de pavage les vecteurs représentants (poids finaux) des unités du réseau (grille ou ficelle) associés à un même cluster (les REPrésentants des CLUsters). Il permet d'évaluer l'homogénéité des clusters (super-classes) au niveau de ses représentants.

Si le nombre de clusters, choisi par l'utilisateur, ne dépasse pas le nombre de couleurs disponibles, le graphique REPCLU est colorié (une même couleur par contenu de cluster).

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses portent à 15 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MKLOBS : *(fenêtre " OBSCLUST ", domaine :* ***KACP, KACPX, KFAST, KBATCH et toute structure****).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2); elle ajoute un graphique nommé OBSCLU au catalogue graphique déjà existant

.

Ce graphique représente sous forme de pavage les valeurs (éventuellement prétraitées) des observations (individus) qui servent d'entrées à KACP, KFAST ou KBATCH et qui sont associées à un même cluster (les OBServations des CLUsters).

Dans le cas de KACPX ce sont les ***données complétées*** *qui servent à la construction du graphique.*

Ce graphique permet d'évaluer l'homogénéité des clusters (super-classes) au niveau de ses observations.

Si le nombre de clusters, choisi par l'utilisateur, ne dépasse pas le nombre de couleurs disponibles, le graphique OBSCLU est colorié (une même couleur par contenu de cluster).

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses portent à 16 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MKOC : *(fenêtre " KOCS ", domaine :* ***KACP, KACPX, KFAST, KBATCH et toute structure****).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2); elle ajoute un graphique nommé CCOL au catalogue graphique déjà existant.

Ce graphique représente sous forme de pavage les valeurs (éventuellement prétraitées) des observations (individus) qui servent d'entrées à KACP, KFAST ou KBATCH et qui sont associées à une même unité gagnante, chaque unité gagnante étant coloriée au couleur de son cluster.

Dans le cas de KACPX ce sont les ***données complétées*** *qui servent à la construction du graphique.*

Ce graphique permet d'apprécier l'homogénéité des unités gagnantes à l'intérieur d'un même cluster.
Si le nombre de clusters, choisi par l'utilisateur, dépasse le nombre de couleurs disponibles, le graphique CCOL n'est pas produit.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses portent à 18 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MSCAL : *(fenêtre " ESCALING ", domaine :* ***tout programme et toute structure****).*

L'utilisation de la technique Multi Dimensional Scaling (cf. PROC MDS qui fournit une représentation graphique des unités du réseau (grille ou ficelle) à partir des distances ***euclidiennes*** *entre les différents représentants (poids finaux) de ces unités, permet de valider la structure du réseau a posteriori.*

Cette macro produit un graphique nommé ESCALING qui est rajouté au catalogue graphique déjà existant et une table sas nommée par défaut _TEMP_DE.sas7bdat qui est placée dans le répertoire des données. Cette table sas fournit les distances euclidiennes entre les vecteurs poids finaux des unités du réseau.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
type du catalogue :	*GACP*	*suivi de F3*

Ces réponses impliquent la création du fichier C:\K8\ABOUL\B8G3C_DE.sas7bdat, et portent à 19 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MSCOL : *(fenêtre " ESCALCLU ", domaine :* ***tout programme et toute structure****).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2); elle effectue le même traitement que la macro MSCAL en ajoutant à chaque unité la couleur associée à son cluster.

Cette macro produit un graphique nommé ESCALCLU (version couleur de ESCALING) qui est rajouté au catalogue graphique déjà existant et une table sas nommée par défaut _TEMP_DE.sas7bdat qui est placée dans le répertoire des données.

Cette table sas fournit les distances euclidiennes entre les vecteurs poids finaux des unités du réseau.

Si le nombre de clusters dépasse 22 il faut utiliser la macro MSCAL.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
type du catalogue :	*GACP*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses portent à 20 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MSUPP : *(fenêtre " EN_PLUS ", domaine :* ***KACP, KFAST, KBATCH et toute structure****).*

Cette macro effectue le traitement des données (observations) supplémentaires (avec éventuellement des valeurs manquantes).

Elle produit 2 graphiques nommés G_PAVAG1 et G_PAVAG2 ou F_PAVAG1 et F_PAVAG2 (selon que le réseau est une grille ou une ficelle) qui sont rajoutés au catalogue graphique déjà existant.

Le graphique G_PAVAG1 pour une grille (ou F_PAVAG1 pour une ficelle) représente la position des données (individus) supplémentaires sur le réseau.

Le graphique G_PAVAG2 pour une grille (ou F_PAVAG2 pour une ficelle) représente la position des données supplémentaires (écrites en italique rouge) sur le réseau, ainsi que celles des données d'apprentissage (d'origine) écrites en bleu.

Elle produit aussi deux tables sas, le nom de la première table est construit à partir du nom de la table contenant les données supplémentaires suivies de "_SG" ou "_SF" (selon que le réseau est une grille ou une ficelle), pour la deuxième table son nom est aussi construit à partir du nom de la table des données supplémentaires suivies de "_PROBG" ou "_PROBF" (selon que le réseau est une grille ou une ficelle). Ces deux tables sont placées dans le répertoire des données

La table dont le nom se termine par de "_SG" ou "_SF" donne pour chaque modalité de la variable qui identifie les données (observations) supplémentaires son numéro d'unité gagnante (sa classe), sa position dans le réseau et ses valeurs (éventuellement manquantes) pour les variables éventuellement transformées par le traitement initial.

La table dont le nom se termine par de "_PROBG" ou "_PROBF" donne pour chaque modalité de la variable qui identifie les données (observations) supplémentaires son numéro d'unité gagnante, son

« résidu », sa probabilité d'appartenir à l'unité gagnante, ainsi que les probabilités d'appartenir aux autres unité du réseau.

Elle produit enfin un catalogue iml des résultats intermédiaires de même nom que celui de la première table sas précédemment créée.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	*suivi de F3*

Suite à cette soumission, s'ouvre la fenêtre " WHAT_NEW " qui permet de sélectionner la table qui contient les données supplémentaires.

Exemple :	*CHOIX*	*Nom de la Table*	
	X	*BLAYO3*	*suivi de F3.*

Puis s'ouvre la fenêtre " QUEL_VAR " qui permet la sélection par C de la variable qui identifie les observations supplémentaires et par N des variables numériques déjà retenues dans l'analyse.

Exemple :	*CHOIX*	*VARIABLE*	
	C	*PAYS*	
	N	*ANCRX*	
	N	*TXMORT*	
	N	*TXANAL*	
	N	*SCOL2*	
	N	*PIBH*	
	N	*CRXPIB*	*suivi de F3.*

Ces réponses impliquent la création des tables sas BLAYO3_SG.sas7bdat et BLAYO3_PROBG.sas7bdat ainsi que du catalogue iml BLAYO3_SG.sas7bcat et portent à 22 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MSUPPX : *(fenêtre " ENPLUX ", domaine :* ***KACPX pour toute structure****).*
C'est la version de la macro MSUPP pour KACPX.

MGVAR : *(fenêtre " GRAF_VAR ", domaine :* ***tout programme et toute structure****).*

Représentations graphiques (profils) des variables ou des modalités (<u>mais pas des individus)</u> à travers la structure du réseau (grille ou ficelle).

Pour éviter d'alourdir le contenu du catalogue graphique déjà existant, cette macro crée son propre catalogue graphique (par défaut GVAR_TEM.sas7bcat) qui est placé dans le répertoire des données et contient un nombre total de graphiques égal à deux fois le nombre de variables (ou de modalités) retenues dans l'étude.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	*suivi de F3*

Ces réponses impliquent la création du catalogue graphique GVARB8G3.sas7bcat placé dans C:\K8\ABOUL; ce nouveau catalogue graphique contient 2x6=12 graphiques.

MVC3D : *(fenêtre " G3DVACLU ", domaine :* ***tout programme et toute structure****).*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2). Comme la macro MGVAR, elle présente l'influence des variables ou des modalités (mais pas des individus) à travers le réseau (grille ou ficelle), mais en plus, elle tient compte de la répartition en super-classes (clusters) en faisant apparaître les couleurs des différents clusters.

Si le nombre de clusters dépasse 22 il faut utiliser la macro MGVAR.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses rajoutent 6 graphiques au catalogue graphique GVARB8G3.sas7bcat placé dans C:\K8\ABOUL et portent à 18 son nombre total de graphiques.

MKAMG : *(fenêtre " CAMGRID ", domaine :* ***KACP, KACPX, KFAST, KBATCH pour une grille*** *).*

Il s'agit d'une méthode qui permet de croiser une variable qualitative avec les variables quantitatives utilisées dans le cadre de KACP, KACPX, KFAST ou KBATCH (cf. la thèse P. ROUSSET dont la référence est donnée dans la section III).

Cette macro doit faire suite à MKLUG, elle produit au plus 4 graphiques rajoutés au catalogue graphique déjà existant.

Description et chronologie des 4 graphiques :

Le graphique nommé CLUSPAC représente sur la grille, pour chaque unité gagnante, la répartition sous forme sectorielle (camembert) des modalités d'une variable qualitative .

Suit le graphique nommé CLUSCAM; ici on colorie CLUSPAC en remplaçant les valeurs numériques (modalités) de la variable qualitative par des couleurs.

Puis vient le graphique nommé CANCLU qui produit un camembert de la variable qualitative pour chaque cluster.

Arrive enfin, le graphique nommé CAMCLU; là encore on colorie le graphique précédent (CANCLU).

Les graphiques, en couleurs, nommés CLUSCAM et CAMCLU ne seront produits que dans la mesure où le nombre de modalités de la variable qualitative et le nombre de clusters ne dépassent pas respectivement 13 et 22 (cf. : les programmes KOLCLUS.sas et KOLCAM.sas dans C:\K8).

Dans le cas où la variable qualitative n'est pas numérique, le programme la recode et affiche dans la fenêtre OUTPUT le tableau de conversion.

La table temporaire sas WORK.GG fournit, pour chaque observation, son unité gagnante (variable _codage_), sa super-classe (variable cluster) et la valeur de la variable qualitative (variable var_qual).

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
nombre de clusters :	*5*	*suivi de F3*

Suite à cette soumission, s'ouvre la fenêtre " QUELDSN " qui permet de sélectionner la table qui contient la variable qualitative.

Exemple :	*CHOIX*	*Nom de la Table*	
	X	*BLAYO2*	*suivi de F3.*

*Puis, s'ouvre la fenêtre " QUEL_VAR " qui permet la sélection par **V** de la variable qualitative et par **O** de la variable qui identifie les observations.*

Exemple :	*CHOIX*	*VARIABLE*	
	O	*PAYS*	
	V	*IDH*	*suivi de F3.*

IDH correspond à 6 niveaux (codés de 1 à 6) de l'Indice du Développement Humain.
Ces réponses portent à 26 le nombre de graphiques du catalogue graphique GACPB8G3.sas7bcat placé dans C:\K8\ABOUL.

MKAMF1 : *(fenêtre " CAMFIC10 ", domaine : **KACP(X), KFAST, KBATCH pour une ficelle (<=10)**.*

MKAMF2 :*(fenêtre " CAMFICEL ", domaine : **KACP(X), KFAST, KBATCH pour une ficelle]10,50]**.*

*Ces deux macros doivent faire suite à MKLUFj (j=1,2). Elles réalisent sur une **ficelle** un traitement qui est équivalent à celui de MKAMG pour une grille.*

Les graphiques CLUSPAC, CLUSCAM, CANCLU et CAMCLU deviennent respectivement :

Pour MKAMF1: CANBFI10, CACOFI10, CANCLU et CAMCLU
Pour MKAMF2: CANBFIC, CACOLFIC, CANCLU et CAMCLU

La table temporaire sas WORK.GG (pour MKAMG) devient la table temporaire WORK.F1 (pour MKAMF1) et la table temporaire WORK.F2 (pour MKAMF2).

MDIMA :*(fenêtre " DIMAWS ", domaine :* ***tout programme et toute structure****).*

Cette macro produit une table sas nommée par défaut _TEMP_DM.sas7bdat qui est placée dans le répertoire des données.

Elle fournit les distances de Mahalanobis entre tous les vecteurs poids finaux des unités du réseau (grille ou ficelle).

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*B8G3CPAY*	
type du catalogue :	*GACP*	*suivi de F3*

Ces réponses impliquent la création du fichier C:\K8\ABOUL\B8G3C_DM.sas7bdat

MDIST : *(fenêtre " DISTWS ", domaine :* ***tout programme et toute structure****).*

Cette macro produit une table sas nommée par défaut _TEMP_DE.sas7bdat qui est placée dans le répertoire des données.

Elle fournit les distances euclidiennes entre tous les vecteurs représentants (poids finaux) des unités du réseau (grille ou ficelle).

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*D7C5G*	*suivi de F3*

Ces réponses impliquent la création du fichier C:\K8\ABOUL\D7C5G_DE.sas7Bdat.

MFMOD : *(fenêtre "FILEMOD", domaine :* ***KORRESP, KACM(j=1,2), KDISJ pour toute structure****.*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2); elle produit deux tables sas nommées MODALn.sas7bdat et MODALTRIn.sas7bdat où n correspond au nombre de clusters retenu dans MKLUG ou dans MKLUFj (j=1,2). Ces fichiers seront placés dans le répertoire des données.

La table MODALn fournit pour chaque modalité (variable idobs) son numéro de super-classes (variable cluster) et son unité gagnante (variable _codage_) et sa position (variables: ligne, colonne) si le réseau est une grille.

La table MODALTRIn est la version triée sur la clef=cluster de la précédente table.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*D7C5G*	
type du catalogue :	*GACM*	
nombre de clusters :	*5*	*suivi de F3*

Ces réponses impliquent la création des fichiers MODAL5.sas7bdat et MODALTRI5.sas7bdat dans le répertoire C:\K8\ABOUL.

MFIND : *(fenêtre " FICKACMj ", domaine :* ***KACMj (j=1,2), KDISJ pour toute structure****.*

Cette macro doit faire suite à MKLUG ou MKLUFj (j=1,2), elle produit une table sas nommée INDIVn.sas7bdat où n correspond au nombre de clusters retenu dans MKLUG ou dans MKLUFj (j=1,2).

La table INDIVn, placée dans le répertoire des données, ne sera créée que si les individus ont été déclarés non anonymes (KACMj pour j=1,2 ou KDISJ); elle fournit pour chaque individu (variable idobs) son numéro de super-classes (variable cluster), son unité gagnante (variable _codage_) et ses modalités de réponses (les autres variables de la table).

Supposons que l'on ait soumis au préalable la macro MKLUG avec les renseignements suivants :

librairie : C:\K8\ABOUL
nom commun : D7C5G
type du catalogue : GACM
nombre de super-classes : 5.

Exemple :

librairie :	*C:\K8\ABOUL*	
nom commun :	*D7C5G*	
nombre de clusters :	*5*	*suivi de F3*

Suite à cette soumission, s'ouvre la fenêtre "CHOIXREP" qui permet de sélectionner le nom de la table qui contient les réponses des individus aux questions posées.

Exemple :	*CHOIX*	*Nom de la Table*	
	X	*CHIENS*	*suivi de F3.*

Puis, s'ouvre la fenêtre " CHOIXQST " qui permet de sélectionner par I la variable identifiant les individus et Q les variables qui correspondent aux questions posées.

Exemple :

CHOIX	*QUESTION*	
I	*RACE*	
Q	*TAILLE*	
Q	*POIDS*	
Q	*VELOCITE*	
Q	*INTELLIG*	
Q	*AFFECTIO*	
Q	*AGRESSIV*	
Q	*FONCTION*	*suivi de F3.*

Ces réponses impliquent la création du fichier INDIV5.sas7bdat dans le répertoire C:\K8\ABOUL.