G. Hunault
Licence Informatique

Angers, octobre 2001

Controle continu pour l'Option de Probabilités et Statistiques

Modalités du Controle

Le controle continu comprend deux petits projets à rendre, l'un concernant la partie Probabilités, l'autre la partie Statistiques. Chaque étudiant(e) doit rendre les deux projets. Chaque projet comporte une partie programmation et une partie rédaction **qui ne comptent pas à part égale** dans la note.

La rédaction se fait au choix mais exclusivement sous *Word* ou sous *Latex*. La programmation est effectuée au choix mais exclusivement en *Awk*, *Apl*, *Rexx*, *C*, *C++*, *Perl*, *Tcl*, *Java* ou *Pascal*. On pourra ne rendre qu'un seul texte qui contiendra les explications, algorithmes et programmes des deux projets. Les documents (explications, algorithmes et programmes) seront uniquement transmis par *mail* sous forme de fichier informatique. Ce fichier, unique, sera au format .zip ou .tar et les fichiers y seront archivés sans sous-répertoire. Ce fichier devra être remis au plus tard le

lundi 10 décembre, 12 h dernier délai

Pour éviter toute ambiguité, tous les noms de fichiers commenceront par les initiales de l'élève. Il sera possible de vérifier auprès de l'enseignant le lundi 17 décembre entre 14 et 18 h que l'archive a bien été reçue et que les fichiers ont bien été décompressés et qu'ils sont lisibles, compilables etc.

1. Projet de Probabilités

Une matrice d'identification est une matrice rectangulaire de n lignes et p colonnes contenant des nombres réels compris entre 0,01 et 0,99 qui relient l'appartenance de taxons (ou "classes") en ligne à des caractères (ou "variables") en colonnes. Un exemple réel est celui de la page 371 du **Handbook of New Bacterial Systematics** édité par Goodfellow et O'Donnell que nous reproduisons ici :

```
0,99   0,99   0,01
0,95   0,50   0,99
0,80   0,01   0,01
```

Soit V un vecteur binaire (c'est à dire ne contenant que des 0 et des 1) de taille p. On appelle valeur d'association probabiliste entre la valeur réelle y et le bit x la quantité y si x vaut 1 et $1 - y$ si x vaut 0. Le produit des valeurs d'association probabiliste terme à terme entre la ligne i de la matrice M et le vecteur V est nommé coefficient de vraisemblance de V pour la classe i. Par exemple, si V est défini par les valeurs `1 0 1` alors le coefficient de vraisemblance de V pour la classe 3 est `0,80` × `(1-0,01)` × `0,01` soit `0,00792`.

On appelle (coefficient de) vraisemblance maximum possible pour la classe i le coefficient de vraisemblance entre la ligne i de la matrice et son masque logique défini par le filtre "supérieur à 0,5" . Ainsi pour la ligne 1, le masque est `1 1 0` et donc la vraisemblance maximum possible pour la classe 1 est `0,99` × `0,99` × `(1-0,01)` soit `0,970299`.

On définit pour V vis à vis d'une classe i les deux probabilités numériques suivantes :

- la probabilité de $Wilcox$ (ou **score d'identification**) : c'est la division du coefficient de vraisemblance de V à la classe i par la somme des tous les coefficient de vraisemblance de V aux différentes classes ;
- la probabilité modale : c'est la division du coefficient de vraisemblance de V à la classe i par le coefficient de vraisemblance maximum possible pour la classe i.

On pourra vérifier que la probabilité de $Wilcox$ pour le vecteur V fourni en exemple est de `0,983233` pour la classe 2 (alors que les probabilités de $Wilcox$ de V vis à vis des autres classes sont très faibles) ce qui permet de supposer très fortement que le vecteur V peut être assimilé comme un élément de la classe 2.

Pour réaliser le projet de probabilités vous viendrez dans un premier temps écrire un algorithme qui lit un fichier-texte dont le nom est passé en paramètre, sachant que ce fichier qui contient $n + 1$ lignes et p colonnes de nombres séparés par des espaces, la matrice M correspondant aux n premières lignes du fichier et V à la dernière ligne ; cet algorithme doit ensuite effectuer les calculs de probabilité demandés en respectant scrupuleusement le format d'affichage suivant

```
Traitement du fichier : hnbs.dat

 Matrice A  :     0.9900    0.9900    0.0100
         B  :     0.9500    0.5000    0.9900
         C  :     0.8000    0.0100    0.0100

 Vecteur          1.0000    0.0000    1.0000

Coefficients de vraisemblance entre la matrice et le vecteur
         A  :   0.000099
         B  :   0.470250
         C  :   0.007920
      total :   0.478269  maximum atteint pour :  B

Coefficients de vraisemblance maximale pour la matrice
         A  :   0.970299
         B  :   0.470250
         C  :   0.784080     total :    2.224629

Probabilités de Wilcox
         A  :   0.000207    soit :    0.0  %
         B  :   0.983233    soit :   98.3  %
         C  :   0.016560    soit :    1.7  %

Probabilités modales
         A  :   0.000102    soit :    0.0  %
         B  :   1.000000    soit :  100.0  %
         C  :   0.010101    soit :    1.0  %
```

Il vous faudra en suite traduire l'algorithme dans un langage de votre choix parmi les langages cités puis rédiger la documentation de votre algorithme et de votre programme. Vous viendrez notamment indiquez quelles précautions doivent êtres prises ou quels "blindages" vous avez effectués (par exemple on refuse de calculer si toutes les lignes du fichier n'ont pas le même nombre de valeurs).

Vous fournirez également l'affichage obtenu si on exécute votre programme en lui passant comme paramètre *mdbc.dat*, ce paramètre correspondant à l'identificateur d'un fichier texte qui contient les valeurs

```
0.80  0.95  0.05
0.99  0.01  0.95
1     0     1
```

Ces valeurs correspondent à l'exemple réel traité page 47 du **Manuel de Bactériologie Clinique, volume 1** (d'où les initiales mdbc) de Freney, Renaud, Hansen et Bollet pour des colonnes nommées *catalase*, *oxydase* et $\beta - galactosidase$.

Enfin, pour conclure ce projet de probabilités, vous essaierez de répondre aux questions suivantes :

- pourquoi les nombres dans la matrice sont-ils compris entre 0,01 et 0,99 ? N'aurait-on pas pu prendre 0,0001 et 0,9999 ? Ou même encore 0 et 1 ?
- quelle est la sensibilité des calculs si on change un dixième à une valeur dans la matrice, par exemple si on utilise 0,94 au lieu de 0,95 ?
- y a-t-il des pages Web qui présentent les probabilités de $Wilcox$, qu'on nomme aussi scores d'identification, pourcentages d'identification normalisé de *Lapage* ?
- en cas de supposition forte d'identification du vecteur à une classe, y a-t-il moyen de savoir quelles colonnes ont contribué le plus à former la forte probabilité ?

2. Projet de Statistiques

Les statistiques traitent des données. Un affichage intelligent des données de base, par exemple un affichage trié, peut donner une idée de la répartition des valeurs. La présentation dite tige-et-feuille (**steam and leaf** en anglais) pour des valeurs entières est une façon particulière assez commode de présenter les données : la tige est formée par la partie haute des valeurs et les feuilles (toutes du même coté) par la partie basse des données. La tige n'est écrite qu'une seule fois, les feuilles autant de fois que de valeurs pour une même tige

Pour certains fichiers, la base de partie haute correspond à la dizaine et la valeur basse aux unités. Par exemple pour les valeurs

```
12 16 51 18 26 12
```

la présentation tige-et-feuille est

```
1   |   2 2 6 8
2   |   6
5   |   1
```

ce qui permet de voir qu'il y a plus de valeurs inférieures à 20 des valeurs supérieures à 20. Pour d'autres fichiers de données, la base de partie haute est la centaine, la partie basse la valeur hors centaine. Ainsi pour le fichier

```
1988 1990 2000 2001 2000 1988 2000
```

la présentation tige-et-feuille est

```
19  |  88 88 90
20  |  00 00 00 01
```

Comme plusieurs bases de partie haute sont possibles pour un même fichier de données, on admettra que la valeur haute est fournie par l'utilisateur, sous forme d'un entier qui donnera la puissance de 10 à utiliser comme valeur haute : ainsi pour 1 la base de partie haute sera la dizaine, pour 2 la base de partie haute sera la centaine etc.

Pour réaliser le projet de statistiques vous viendrez dans un premier temps écrire un algorithme qui lit un fichier-texte dont le nom est passé en paramètre, sachant que ce fichier contient un et un seul nombre par ligne. L'algorithme doit ensuite présenter les données sous forme d'un diagramme tige-et-feuille avec les nombres correctement formatés (unité sous unité, dizaine sous dizaine, zéro écrit en 00 pour des nombres à deux chiffres etc.) grâce à l'indication de base de partie haute fournie comme deuxième paramètre.

Vous n'oublierez pas de trier les feuilles par ordre croissant comme sur les exemples de la page précédente. On pourra admettre que les lignes d'affichages pour les feuilles peuvent être très longues dans un premier temps, mais il serait souhaitable que l'affichage soit cadré sur $LargeurPage$ caractères. Ainsi au lieu de voir (ou plutot de ne pas voir)

```
19   |   80 80 80 80 80 80 81 81 82 82 82 82 82 82 86 87 88 88 90
20   |   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 01 01
```

il serait plus agréable d'obtenir

```
19   |   80 80 80 80 80 80 81 81 82 82
     |   82 82 82 82 86 87 88 88 90
20   |   00 00 00 00 00 00 00 00 00 00
     |   00 00 00 00 00 00 00 00 01 01
     |   01 01 01 02 02 02 02 02 03 05
     |   06 09
```

Vous viendrez ensuite traduire l'algorithme dans un langage de votre choix parmi les langages cités puis rédiger la documentation de votre algorithme et de votre programme. Vous rédigerez ensuite une page entière sur le ou les inventeurs de ce type de présentation, les intérêts (cachés) de cet affichage, la disponibilité de cet affichage dans les logiciels statistiques usuels sans oublier de citer précisément vos sources. Celles-ci pourront être des URL.

Vous fournirez également l'affichage obtenu si on exécute votre programme en lui passant comme paramètres $elfage.dat$ et `1`, le premier paramètre correspondant à l'identificateur d'un fichier texte qui contient 99 valeurs (des ages, exprimés en années) extrait d'une Enquête Linguistique portant sur la Féminisation des noms de métiers (d'où les lettres `E`, `L` et `F`). Ce fichier est disponible à l'adresse http://www.info.univ-angers.fr/pub/gh/wstat/elfage.dat ainsi que sur le disque K: des machines de la Faculté des Sciences, dans le sous-répertoire Stat_Ad.

Remarques concernant la rédaction

La note sera principalement fonction de la qualité de l'analyse du travail effectué, de l'exposé de la méthode et de la clarté de la rédaction. La valeur attribuée au programme ne représentera que 25 % de la note. Aucune interface n'est demandée, les programmes devront être exécutables en ligne de commande. En cas de langage compilé, on fournira impérativement le programme-source et éventuellement l'exécutable produit. On précisera, si besoin est, la commande de compilation. En cas de langage interprété, on fournira impérativement le programme-source. Dans tous les cas de figures, on précisera la version du langage choisi, le type d'ordinateur et le système d'exploitation utilisés. Aucun langage autre que cité ne devra être utilisé, aucun traitement de textes autre que cité ne devra être utilisé. Tout fichier qui ne correspondra pas aux initiales de l'élève ne sera pas lu. En particulier, les fichiers nommés uniquement proba, algo, stat,projet, doc ne seront pas lus.

Un programme même incomplet mais commenté et bien indenté sera au moins autant apprécié qu'un programme qui "marche" mais sans commentaire et illisible quand aux structures de flot (boucles, tests etc.).

Les identificateurs auront tous au moins 4 caractères. Les algorithmes utiliseront la syntaxe de *Galg*. On fournira d'ailleurs non pas le texte original des algorithmes mais leurs résultats d'analyse par *Galg* avec le numéro de ligne, le numéro d'instruction, l'indication de profondeur etc. et le fichier des tableaux, variables et modules.

3. Références

Ce texte, de même que le cours de *Probabilités et Statistiques* pour la licence informatique, les T.D. correspondants (avec solution) et la procédure pour envoyer le mini-projet peut être récupéré à partir de l'adresse

`http://www.info.univ-angers.fr/pub/gh/wstat/`

Un manuel élémentaire d'*Algorithmiques raisonnées* et les références à *Galg* sont sur la page

`http://www.info.univ-angers.fr/pub/gh/wals/`

De plus, il est possible de tester si le projet a été correctement reçu grâce à la page

`http://www.info.univ-angers.fr/pub/gh/wstat/zipdirpli.php3`