Enseignant : Sofiane MAAZI

# Introduction aux logiciels pour les statistiques

GIS2A3 / 2019-2020

R est à la fois un langage de programmation et un logiciel libre. Dans ce cours, nous ne ferons pas le distinguo entre les deux. R permet notamment d'analyser des données statistiques et de réaliser des graphiques. Force est de constater que l'interface de R est ni attrayante ni fonctionnelle. C'est pourquoi, il est recommandé d'installer Rstudio en complément de R.

## Scripts R

1. Une 1ère manière de travailler est de rentrer directement le code dans la console. Ouvrir le logiciel RStudio. Saisissez les commandes « 1+1 » puis « a<-1+1 » dans la console. Fermez ensuite RStudio. Ouvrir de nouveau Rstudio. Que remarquez-vous ?

   *En quittant la session, on peut sauvegarder une image de l'espace de travail sur le disque dur de l'ordinateur. Physiquement, elle prend la forme d'un fichier «. RData ». À la session suivante, on retrouve donc les objets R créés à la session précédente. Toutefois, le code utilisé pour créer les objets n'a pas été sauvegardé ! On ne sait donc plus l'utilité des objets R et surtout comment ils ont été construits !*

2. Une 2ème manière de travailler en R consiste à sauvegarder - dans des fichiers de script – le code qui a servi à créer les objets R. Ainsi les objets sont créés au besoin en exécutant le code des fichiers scripts. Donnez 2 avantages de cette méthode.

   *D'une part, il est possible de revenir sur le code ultérieurement. D'autre part, le code peut être transmis à quiconque notamment pour être retravaillé ou corrigé.*

3. Enregistrez dans un dossier « ViveR » un script R que vous nommerez « coursPolytech ». Fermez votre script et l'ouvrir de nouveau. Recopiez dans le script les lignes de commande du 1) et exécutez-les (CTRL+A suivi de CTRL+Entrée). Ouvrir de nouveau Rstudio. Que remarquez-vous ?

   *Tant que le script n'a pas été fermé, il est présent à chaque ouverture de Rstudio.*

4. Le répertoire de travail « workspace » est un dossier par défaut dans lequel R recherche fichiers de script ou de données et sauvegarde l'image de l'espace de travail. Il convient donc de préciser en début de script, le nom du dossier choisi pour être le répertoire par défaut. À l'aide la commande, « setwd() » , indiquez le dossier « ViveR » comme répertoire par défaut. Que doit respecter la chaîne de caractères passée en paramètre de la fonction setwd() ?

*La chaîne de caractère correspond au chemin d'accès vers le dossier cible. Elle n'accepte pas le symbole « \ ». Il doit être remplacé par « / » sous peine d'avoir un message d'erreur !*

5. Pour avoir un script aéré, il convient de le découper en plusieurs parties en utilisant le symbole « # » suivi d'une séquence d'un caractère comme « = » ou « - ». Quel est le rôle premier de « # » ?

   *Ce symbole sert à ajouter un commentaire à une ligne de code. Pour R, les lignes de code précédées de « # » et servant à séparer le code en plusieurs parties sont considérées comme étant des commentaires.*

# Table de données

1. Un grand magasin a relevé en 2019 à certains passages en caisse les articles achetés par des clients possédant une carte de fidélité. On suppose cet échantillon représentatif de l'ensemble de la population du magasin. Les données sont purement fictives. Placez le fichier « magasin.csv » dans le dossier « ViveR ». Ouvrez le fichier csv en vous utilisant le bouton « Import Dataset ». Est-il possible d'ouvrir le fichier csv d'une autre manière ? Le cas échéant, précisez les inconvénients qu'elle présente.

   *Il est possible d'ouvrir le fichier csv avec les fonctions read.csv () ou encore read.csv2(). Toutefois, elles requièrent chacune de passer des paramètres en argument avec des subtilités. Par exemple, le séparateur par défaut de read.csv2() est le point-virgule alors que pour la fonction read.csv() c'est la virgule. Néanmoins, elles permettent de forcer le type qu'impose R aux variables lors de l'importation. « Import Dataset » est plus simple d'utilisation car ce bouton permet d'importer le fichier csv via une interface. L'utilisateur peut notamment choisir le séparateur, changer le nom de la table importée, changer le type des variables, avoir un aperçu de la table et récupérer le code qu'il aurait fallu saisir pour importer la table. Il peut ainsi le copier-coller dans un script.*

   (a) Donnez la taille de l'échantillon et les individus statistiques étudiés.

   *La table comprend 170 observations. Autrement dit, on a relevé les informations pour 170 clients porteurs d'une carte de fidélité. La taille de l'échantillon est donc 170.*

   (b) Précisez si l'échantillon possède au moins une valeur manquante. Quel défaut majeur présente une valeur manquante en R comparé à d'autres logiciels de statistiques ? Illustrez.

   *Une valeur manquante est indiquée en R par « NA ». Il y a une valeur manquante à la première ligne et quatrième colonne (c'est d'ailleurs la seule !). Par défaut, R prend en considération les valeurs manquantes dans les calculs et les fonctions prédéfinie (somme, moyenne,...). Les résultats sont donc faussés sans qu'on s'en rend compte forcément. Une solution est d'enlever toutes les valeurs manquantes avant d'effectuer des calculs. Toutefois, on préferera utiliser l'expression « na.rm = TRUE ».*

   (c) Donnez le type que R attribue à chaque variable.

*La table comprend 2 variables de type caractère (RAYON, AVIS) et 4 variables numériques (ID_CARD, REF, PRIX, REDUCTION).*

(d) Vous disposez à présent de la signification de chaque variable :
- ID_CARD : numéro de la carte fidélité du client qui est passé en caisse.
- RAYON : rayon dans lequel l'article a été acheté.
- REF : référence magasin du produit.
- PRIX : prix de l'article.
- BON_REDUCTION : le client dispose d'un bon de réduction pour l'article.
- REDUCTION : montant de la réduction effectuées sur le prix de l'article.
- FREQUENCE : fréquence à laquelle le client achète l'article en 2019 (« Jamais», «Parfois», «Souvent», «Toujours»)

Donnez la nature de ces variables au sens du cours (quantitative, qualitative ordinale ou qualitative nominale). Les types du (c) sont-il cohérents avec ceux que vous venez de citer ? Est-ce toujours le cas ?

*La table comprend 2 variables qualitatives nominales (RAYON, BON_REDUCTION), 1 variable qualitative ordinale (FREQUENCE) et 4 variables numériques (ID_CARD, REF, PRIX, REDUCTION). . À noter que la nature de certaines variables est parfois discutable. Les types sont cohérents avec ceux du (b). Toutefois, il arrive que le type proposé par R ne soit pas celui le plus pertinent pour la variable. Par exemple, la variable RAYON aurait pu être de type facteur et pas de type caractère. L'utilisateur peut alors forcer le type en utilisant des fonctions de « conversion » (as.character(), as.numeric()...) ou le changer lors de l'importation.*

2. Pour explorer une table de données, il est courant de s'appuyer sur le package *dplyr*. La syntaxe de *dplyr* est construite selon le schéma :

$$Resultat <- verbe(nom\_de\_la\_table, action\_a\_realiser)$$

Les 5 principaux verbes possibles sont :

| Verbe | But |
|---|---|
| select | sélectionner des variables |
| filter | filtrer la table selon une ou plusieurs conditions |
| arrange | trier la table |
| summarise | calculer des indicateurs |
| mutate | créer une variable |

Comment stipuler à R que vous souhaitez utiliser le package dplyr ?

*Quand un utilisateur installe R sur son poste, R n'est pas installé intégralement. Seules les fonctionnalités de bases sont installées. Les fonctionnalités plus sophistiquées sont rangées par thème dans des packages sur le site du CRAN. Il faut alors identifier le package dont on a besoin (dplyr ici) est l'installer sur le poste de travail. L'appel au package devra être fait à chaque début de script à l'aide de l'instruction « library». À noter que les packages sont mis régulièrement à jour par la communauté R.*

(a) Sélectionnez la variables *PRIX*. Stockez le résultat dans une variable SELECTION.

*SELECTION<-select(MAGASIN,PRIX)*

(b) Filtrez les lignes de MAGASIN où *PRIX* est strictement supérieur à 5 euros et *REDUCTION* inférieure ou égale à 20%.

*MAGASIN<-filter(MAGASIN,PRIX>5,REDUCTION<=0.2)*

(c) Effectuez un tri croissant de la table précédente selon *PRIX* .

*MAGASIN<-arrange(MAGASIN,PRIX)*

(d) Créez dans la table précédente une variable *NEW_PRIX* égale au prix après réduction.

*MAGASIN<-mutate(MAGASIN,NEW_PRIX=round(PRIX*(1-REDUCTION),2))*

(e) Calculez la moyenne de *PRIX* par *RAYON* .

```
MAGASIN %>% group_by(RAYON) %>% summarise(MOYENNE_PRIX = mean(PRIX, na.rm=TRUE))
```

3. Les lignes de commande avec les verbes de dplyr peuvent être combinées. Par exemple, des colonnes peuvent être sélectionnées sur une table filtrée au préalable. Il est possible d'en faire de même avec la syntaxe de R sans recourir à dplyr. Expliquez.

*Une table de données en R est de type data.frame. Chaque colonne est donc de type vecteur. On peut donc sélectionner et filtrer la table par une approche « vectorielle ». Une data.frame possède également les propriétés d'une matrice. En particulier, une celulle est caractérisée par un couple d'identifiants ($i^{me}$ ligne et $j^{me}$ colonne). On peut donc aussi sélectionner et filtrer la table par une approche « matricielle ». À noter que l'approche vectorielle ( respectivement matricielle) est identifiable par « $ » (respectivement « [ » et « ] »). Supposons la $2^{me}$ colonne de la table MAGASIN intitulée RAYON. Supposons k et p des vecteurs d'entiers. le tableau ci-dessous donne les correspndances ( liste non exhaustive).*

| Sélection de | Vectorielle | Matrcielle |
|---|---|---|
| RAYON | MAGASIN$RAYON | MAGASIN[,2] |
| la $8^{eme}$ modalité de RAYON | MAGASIN$RAYON[8] | MAGASIN[8,2] |
| la $4^{eme}$ ligne de MAGASIN | FASTIDIEUX | MAGASIN[4,] |
| les $k^{eme}$ ligne de MAGASIN | FASTIDIEUX | MAGASIN[k,] |
| les $p^{eme}$ colonnes de MAGASIN | FASTIDIEUX | MAGASIN[,p] |

## Fusion de tables

Le directeur de l'enseigne souhaiterait disposer du libellé en clair de chaque produit acheté. Il possède une table « referentiel_produits.csc» qui contient 2 variables :

— REF : référence magasin du produit.
— LIBELLE : nom de l'article.

La variable REF est commune à « magasin.csv» et « referentiel_produits.csv». Proposez une nouvelle table répondant à la demande du directeur. Vous utiliserez au choix une fonction du package *dplyr* (left_join(), inner_join() et right_join()) ou la fonction merge(). Laquelle des fonctions proposées vous semble la plus appropriée ? Justifiez.

*C'est une situaton classique où il faut réaliser une fusion de table selon une colonne. Ici, il faut réaliser une fusion des tables « magasin.csv» et « referentiel_produits.csv» selon la variable REF. Les fonctions left_join() (respectivement right_join() permettent de réaliser une fusion en conservant les lignes ayant la même valeur de REF dans les 2 tables en plus des lignes de la table de gauche (respectivement de droite) qui n'auraient pas de correspondance dans la table de droite ( respectivement de gauche). La fonction inner_join() permet uniquement de réaliser une fusion en conservant les lignes ayant la même valeur de REF dans les 2 tables. Enfin, la fonction merge peut être paramétrée pour agir comme left_join(), inner_join() ou right_join(). Dans le cas présent, toutes les fonctions peuvent être utilisées indifféremment car chaque ligne a une correspondance dans l'autre table selon la variable REF.*

# Graphiques

Expliquez le rôle des paramètres « main »,« xlab », « ylab », « probability », « breaks » et « col » de la fonction *hist*.

*Les paramètres « main »,« xlab » et « ylab » permettent de personnaliser respectivement le titre du graphique et les étiquettes des axes. Quand le paramètre « probability » vaut TRUE alors l'histogramme indique la proportion des classes de valeurs au lieu des effectifs. Le paramètre « breaks » permet de contrôler les classes de valeurs. Il peut prendre - entre autres - comme valeur un chiffre indiquant le nombre de classes souhaité ou encore un vecteur indiquant les limites des différentes classes. Enfin, le paramètre « col » permet de choisir la couleur de l'histogramme. D'autres paramètres dans la documentation de la fonction permettent bien entendu de personnaliser l'histogramme.*