

TP2 : Lecture et affichage de tableaux avec R, illustrations de théorèmes classiques

Pour ce deuxième TP, nous allons dans un premier temps apprendre à gérer des tableaux de données sous R. Nous verrons ensuite deux exercices pour illustrer des théorèmes du cours.

1 Tableaux

1.1 Types de variables

Lors de ces travaux pratiques, on va manipuler des tables de données, décrivant souvent certains caractères pour plusieurs individus. On a en général un individu par ligne, et chaque colonne correspond à un caractère des individus, autrement dit à une *variable*. On peut distinguer au moins 3 types de variables :

variables numériques les valeurs sont des nombres réels, qu'on peut comparer, et telles que l'écart entre deux valeurs a un sens. Par exemple : taille ou poids d'un individu.

variables ordinales les valeurs sont des catégories, il y a un ordre entre ces catégories. Par exemple, une variable qui aurait pour valeurs possibles *tres grand, grand, moyen, petit, tres petit*.

variables nominales les valeurs sont des catégories, et on n'a pas d'ordre entre ces catégories. Par exemple, une variable dont les valeurs seraient les nationalités d'individus.

Avec R, les variables nominales et ordinales sont représentées par un même type, le type `factor`.

Télécharger les tables suivantes, repérer les individus et indiquer le type de chaque variable :

- `onu67_budget_temps.txt`
- `cesp92_budget_temps_multimedia.txt`
- `pbio.txt`
- `ronfle.txt`

1.2 Lecture de tableaux de données

`read.table` : c'est la fonction principale pour lire une table de données. Range les données dans un « `data.frame` ».

Quelques paramètres (avec leur valeur par défaut) :

`sep` (" ") le caractère qui sépare les colonnes; le défaut " " représente à la fois les espaces et les tabulations

`dec` (".") le caractère qui marque la séparation entre les parties entière et décimale des nombres

`comment.char("#")` le caractère qui indique une ligne qui ne doit pas être lue dans le fichier

`colClasses` un vecteur qui permet de spécifier les types des variables / colonnes (en l'absence de ce paramètre, R fait « au mieux »).

`header` un booléen qui indique si la première ligne contient les noms des colonnes

`row.names` un vecteur donnant les noms des lignes

`col.names` un vecteur donnant les noms des colonnes (si `header=FALSE`)

Remarque : si la première ligne contient un champ de moins que les autres, alors par défaut `header=TRUE`, la première ligne est utilisée pour les noms des colonnes, et la première colonne est utilisée pour les noms des lignes.

```
Tread.table("onu67_budget_temps.txt", colClasses = c("character",
rep("numeric", 10), rep("factor", 4)))
```

(La fonction `rep` reproduit son premier argument, par exemple `rep("factor", 4)` est équivalent à :

```
c("factor", "factor", "factor", "factor").)
```

Utiliser la fonction `read.table` pour lire les tables de l'exercice 1.1. On **vérifiera** que chaque table a été bien lue avec les commandes `summary` et `head`.

Manipulation des `data.frame` Si T est un tableau représenté par un `data.frame` (par exemple le résultat de `read.table`), on peut :

- avoir les noms des lignes et des colonnes avec les instructions `row.names(T)` et `names(T)`; on peut changer ces noms, en affectant à `row.names(T)` ou `names(T)` un vecteur de chaînes de caractères (`names(T)...`);
- extraire certaines lignes ou colonnes avec par exemple `T[, c(4, 8)]` (pour la 4ème et la 8ème colonne) ou encore `T[3,]` (la 3ème ligne);
- extraire certaines lignes qui vérifient certaines conditions, par exemple : `T[T$PAYS == 1,]` pour extraire les lignes pour lesquelles la variable `PAYS` vaut 1;
- rajouter des lignes et des colonnes avec les commandes `rbind` et `cbind`;

Quel est le résultat de l'instruction `T$PAYS == 1`, si T est la table du fichier `onu67_budget_temps.txt`?

À l'aide des fonctions `ifelse` et `cbind`, rajouter à la table T une colonne qui contient les noms des pays de chaque individu ("USA", "Yougoslavie", "Autre Ouest" ou "Autre Est").

2 Illustration des théorèmes du cours

2.1 Exercice 1

Dans tout l'exercice, pile sera associé à 1, et face à zéro. De plus, on notera θ la probabilité d'avoir pile.

1. Réalisez l'expérience aléatoire suivante : lancer $n = 15$ fois une pièce de monnaie équilibrée. Pour ce faire on simulera (à l'aide de) successivement

15 variables aléatoires de Bernoulli. On note x_i le résultat du i ème lancer, où $i = 1, \dots, 15$. Indiquer le résultat de l'expérience que vous venez de réaliser et donner la valeur de \bar{x} .

2. Faites 20 fois l'expérience aléatoire ci-dessus, et noter à chaque fois la valeur de \bar{x} obtenue.
 - (a) Imaginez un instant que vous ignorez le fait que la pièce est équilibrée. Que représente alors chaque \bar{x} ? Que constatez-vous? Commentez.
 - (b) Faites la moyenne de tous ces \bar{x} , puis comparer cette valeur à 0.5. On constate alors un écart entre ces deux valeurs : commentez cet écart relativement au fait que \bar{X} est un estimateur sans biais de θ .

2.2 Exercice 2

Soit X_1, \dots, X_n un échantillon de loi uniforme $\mathcal{U}([0, \theta])$. Rappeler la formule de l'estimateur du maximum de vraisemblance $\hat{\theta}_{EMV}$ de θ , puis de l'estimateur par la méthode des moments $\hat{\theta}_m$ de θ . Comparer à l'aide d'un plot la vitesse de convergence des deux estimateurs. Commenter.