

Fiche de révision : Commandes R

Préparation et manipulation des données

Importer un jeu de données

Rôle : charger un fichier externe dans l'environnement.

Syntaxe : `read.csv("fichier.csv", sep = ",")`

Exemple : `df <- read.csv("titanic.csv", sep = ";")`

Transformer en variable qualitative

Rôle : indiquer au logiciel qu'une variable chiffrée représente des catégories.

Syntaxe : `as.factor(tableau$variable)`

Exemple : `df$survie <- as.factor(df$survie)`

Vérifier la structure

Rôle : afficher le type de chaque colonne du tableau.

Syntaxe : `str(tableau)`

Exemple : `str(df)`

Vérifier la taille

Rôle : compter le nombre de lignes (individus) du tableau.

Syntaxe : `nrow(tableau)`

Exemple : `nrow(df)`

Extraire des valeurs précises

Rôle : cibler des données selon une condition spécifique avec les crochets.

Syntaxe : `tableau$variable[condition]`

Exemple : `df$age[df$survie == 1]`

Croisement qualitatif et qualitatif

Créer un tableau croisé

Rôle : compter les effectifs bruts entre deux variables catégorielles.

Syntaxe : `table(ligne, colonne)`

Exemple : `tab <- table(df$survie, df$genre)`

Calculer les pourcentages

Rôle : transformer les effectifs bruts en pourcentages (1 = ligne, 2 = colonne).

Syntaxe : `prop.table(tableau_brut, margin = 1 ou 2)`

Exemple : `prop.table(tab, margin = 2)`

Arrondir les résultats

Rôle : limiter le nombre de décimales pour faciliter la lecture.

Syntaxe : `round(valeur, nombre_decimales)`

Exemple : `round(14.567, 1)`

Croisement qualitatif et quantitatif

Calculer par sous-groupe

Rôle : calculer une moyenne ou un écart-type pour chaque catégorie.

Syntaxe : `aggregate(quantif ~ qualif, data = tableau, fun = mean ou sd)`

Exemple : `aggregate(prix ~ classe, data = df, fun = mean)`

Afficher une boîte à moustaches

Rôle : visualiser la dispersion d'une variable quantitative selon des catégories.

Syntaxe : `boxplot(quantif ~ qualif, data = tableau)`

Exemple : `boxplot(age ~ survie, data = df)`

Croisement quantitatif et quantitatif

Générer un nuage de points

Rôle : observer visuellement le lien entre deux variables numériques.

Syntaxe : `plot(x = variable1, y = variable2)`

Exemple : `plot(x = df$age, y = df$prix)`

Calculer la covariance

Rôle : indiquer le sens de variation commun (nécessite d'ignorer les valeurs manquantes).

Syntaxe : `cov(x, y, use = "complete.obs")`

Exemple : `cov(dfage, dfprix, use = "complete.obs")`

Calculer la corrélation

Rôle : évaluer la force du lien sur une échelle stricte de -1 à 1.

Syntaxe : `cor(x, y, use = "complete.obs")`

Exemple : `cor(dfage, dfprix, use = "complete.obs")`

Échantillonnage

Fixer le hasard

Rôle : bloquer le générateur aléatoire pour reproduire les mêmes résultats.

Syntaxe : `set.seed(nombre)`

Exemple : `set.seed(42)`

Tirage aléatoire simple

Rôle : piocher au hasard un nombre précis d'éléments.

Syntaxe : `sample(plage_elements, taille)`

Exemple : `lignes <- sample(1:nrow(df), 1000)`

Tirage stratifié par quotas

Rôle : extraire une proportion exacte pour chaque sous-groupe (nécessite le paquet dplyr).

Syntaxe : `tableau %>% group_by(categorie) %>% slice_sample(prop = decimal)`

Exemple : `df %>% group_by(region) %>% slice_sample(prop = 0.05)`

Tests statistiques et intervalles

Marge d'erreur statistique

Rôle : estimer mathématiquement la fourchette d'incertitude autour de la différence observée.

Syntaxe : `1.96 * sd * (1/sqrt(n1) + 1/sqrt(n0))`

Exemple : `1.96 * sd(df$age) * (1/sqrt(100) + 1/sqrt(150))`

Comparer deux moyennes (Test de Student)

Rôle : évaluer formellement si l'écart entre deux moyennes est lié au hasard.

Syntaxe : `t.test(quantitatif ~ qualitatif, data = tableau)`

Exemple : `t.test(age ~ survie, data = df)`

Tester l'indépendance (Khi-Deux)

Rôle : vérifier si deux variables qualitatives sont statistiquement liées.

Syntaxe : `chisq.test(tableau_croise)`

Exemple : `chisq.test(table(df$survie, df$genre))`