IFT-1902: Dictionnaire en préparation de l’examen 2

Chapitre 7 (deuxième plus gros) : Tri et recherche (boucles)

Boucle:

for(‘variable’ in ‘suite’) ‘expression’

while(‘condition’) ‘expression’ (tant que TRUE)

repeat ‘expression’ (comporte TJS un test d’arrêt)

break : force la sortie de la boucle courante (pesqu’onbligatoire avec boucle repeat)

next : passer à la prochaine itération

\*\* Syndrome de la plaque à biscuits : lorsqu’on connaît la longueur de l’objet, on crée un contenant vide qu’on remplit \*\*

Tri :

sort : ordre croissant | décroissant

rank : rang des éléments dans l’ordre crosisant | décroissant

order : ordre d’extraction des éléments pour les placer en ordre croissant | décroissant (voir vidéo de la fonction)

rev : vecteur inversé

unique : éléments uniques d’un vecteur

Recherche :

Which : positions des éléments satisfaisant une condition

Which.min : position du minimum

Which.max : position du maximum

Match : position de la première occurrence d’un élément

%in% : appartenance d’une | plusieurs valeurs au vecteur (booléen)

Chapitre 8 : Débogage

Approche naïve :

Vérifier syntaxe et afficher résultats intermédiaires (manuellement)

Évaluation pas-à-pas :

Définir tous les arguments utilisés et Ctrl+Enter une ligne à la fois (sauter par-dessus les lignes « boucle »)

Navigateur d’environnements :

**traceback** et **browser** (compliqué, mais plus rapide et utile)

Commandes browser :

n : évaluer prochaine expression (pas d’entrée dans les appels de fonctions)

s : évaluer prochaine expression (entrée dans les appels de fonctions)

c : quitter le navigateur et poursuivre exécution normale

Q : quitter le navigateur et retour à l’invite de commande

Méthode du canard en plastique :

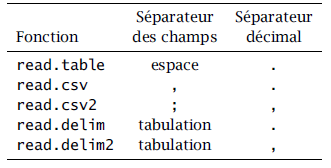
Sherlock Holmes avec son crâne.

Chapitre 9 : Import-Exportation de données

Importation de vecteurs de données :

Scan : importer des données brutes dans R (retourne un vecteur)

Importation de tableaux de données :



\*\* ajouter encoding = « UTF-8 » si accents présents \*\*

Exportation de données :

Cat : exportation d’un vecteur simple vers un fichier texte brut (write = tableau rectangulaire, par défaut = 5 colonnes)

Write.table : version exportation de read.table

Write.csv : version exportation de read.csv

Write.csv2 : version exportation de read.csv2

\*\* Voir script import-export.R pour une meilleur compréhension \*\*

Chapitre 10 : Bibliothèques et paquetages

Système de base :

Search() : liste des paquetages chargés

Library() : affiche le contenu de la bibliothèque de paquetages

Library(« ‘paquetage’ ») : charge les fonctionnalités d’un paquetage dans la session de travail

Création d’une bibliothèque personelle :

\*\* Lire les notes de cours et utiliser Git Bash \*\*

Installation de paquetages additionnels :

Install.packages(« ‘paquetage’ ») (ensuite, charger le paquetage avec library)

Chapitre 11 (le plus gros) : Analyse et contrôle de texte (expressions régulières)

Fonctions utiles :

Grep : recherche par correspondance à un | plusieurs motif(s)

Grepl : retourne valeur booléenne

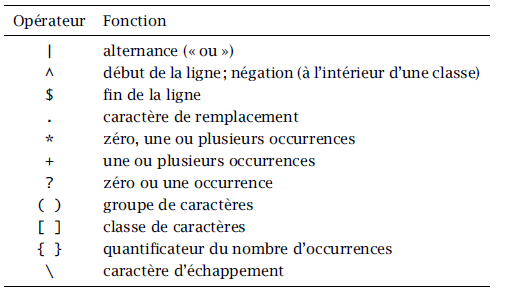
Sub : remplacement fait sur la première occurrence d’un motif (équivalent de sed) (dans chaque chaine)

Gsub : remplacement fait sur toutes les occurrences d’un motif (équivalent de sed) (dans chaque chaine)

Sed : recherche et remplace du texte sur une ligne (pas définie dans R, seulement Git Bash)

Awk : couteau suisse; base = traitement de texte naturellement séparé (pas dans R)

Expressions régulières :



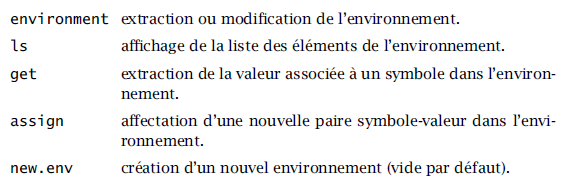
\*\* Lire 11.2, 11.4, 11.6 et 11.7 pour une meilleure compréhension \*\*

Chapitre 12 : Environnement et règles d’évaluation (court, mais complexe)

Environnement :

Formals(‘objet’) : arguments formels

Body(‘objet’) : corps de la fonction (‘expression’)



Ls.str : afficher en une seule commande la liste des objets d’un environnement et les valeurs de ceux-ci.

Environnement d’évaluation, portée lexicale, sumsq :

\*\* Lire les notes de cours et évaluer le script environnement.R \*\*