

---

## Introduction à R

Février 2020

---

### Contents

<b>1</b>	<b>R et RStudio</b>	<b>1</b>
<b>2</b>	<b>Prise en main de RStudio</b>	<b>2</b>
2.1	L'interface RStudio . . . . .	2
2.2	Utiliser R . . . . .	2
<b>3</b>	<b>Application 1 : importation d'une base et analyses statistiques</b>	<b>4</b>
<b>4</b>	<b>Application 2 : gestion de base de données et tests statistiques</b>	<b>4</b>

## 1 R et RStudio

R est à la fois un langage de programmation et un environnement pour exécuter des analyses statistiques et économétriques, et produire des graphiques. R se démarque des autres environnements d'analyses statistiques et économétriques de part la qualité des graphiques produits et la facilité de pouvoir contrôler chaque élément du graphique séparément.

R étant un langage avant tout, il est donc nécessaire de lui introduire des commandes à exécuter, ce qui peut se faire à l'aide d'un terminal de commandes. Cependant, il existe un environnement de développement intégré adapté à R : il s'agit de RStudio, que nous utiliserons tout au long de nos séances de travaux. Il faut avoir au préalable installé R avant de pouvoir utiliser RStudio.

### Installation de R et RStudio

Pour télécharger et installer R, suivez les étapes suivantes:

- Allez sur <https://cran.r-project.org/mirrors.html>
- Choisir le serveur le plus proche pour la Belgique (ici <https://www.freestatistics.org/cran/> ou <https://lib.ugent.be/CRAN/>)
- Selon le système d'exploitation que vous utilisez, cliquez sur le “**Download R for...**” correspondant
- Procédez à l'installation après téléchargement

Pour télécharger et installer RStudio, suivez les étapes suivantes:

- Allez sur <https://rstudio.com/products/rstudio/download/#download> pour télécharger la version ‘Bureau’ gratuitement
- Procédez à l'installation après téléchargement

## 2 Prise en main de RStudio

Démarrez le programme RStudio.

### 2.1 L'interface RStudio

A la première ouverture, 3 panneaux composent la fenêtre : le panneau à gauche est la *Console*, où les commandes peuvent être rentrées directement. Les résultats des commandes sont aussi affichés dans ce panneau. Les commandes s'exécutent dans ce panneau en appuyant **Entrée**. Essayez en tapant `<<2+2>>` et exécutez.

A droite de la fenêtre, 2 panneaux se superposent : en haut, le panneau *Environment/History/Connections* et en bas, *Files/Plots/Packages/Help/Viewer*. Dans le sous-panneau *Environment*, vous verrez apparaître les objets que vous avez créés, les données que vous avez importées... Le sous-panneau *History* contient l'historique des commandes exécutées. Vous aurez souvent recours au sous-panneau *Help* pour trouver de l'aide. Les graphiques et tableaux créés s'affichent dans *Plots*. Nous reviendrons plus tard sur les *Packages*.

Il existe un quatrième panneau, *Source* qui s'affichera en haut en gauche lorsque vous démarrez un nouveau ‘**Script**’. Il est en effet plus efficace d'écrire (et enregistrer) les commandes dans un Script que dans la Console. Pour ouvrir un nouveau Script, exécutez la combinaison Ctrl+Shift+N (vous pouvez également l'ouvrir à partir du menu **File**). Pour exécuter une commande à partir du Script, placez le curseur sur la ligne de la commande et appuyer sur Ctrl+Entrée, ou cliquez sur Run. Il est possible d'exécuter un bout de la commande en ne sélectionnant que la partie de la commande concernée.

**Astuce :** Pour ajouter des commentaires à votre code, commencez avec le symbole #, tout ce qui se trouvera après, et sur la même ligne est lu par R comme un commentaire.

### 2.2 Utiliser R

#### Les opérations basiques

R peut servir à effectuer plusieurs types de calculs et d'opérations. Les plus usuels sont avec les symboles pour les opérations arithmétiques (+, -, \*, /, ^) ou les opérations logiques (==, !=, <, >, <=, >=).

>, >=, ...).

Avant de commencer, il est plus judicieux de créer et enregistrer son travail dans un répertoire bien défini. Utilisez la commande **Save as...** du menu **File** pour cela.

**Astuce** : Pour gagner du temps, définissez votre répertoire de travail avec la commande:

```
1 setwd("C:/Mes Documents/Econometrie")
```

où vous remplacez 'C:/Mes Documents/Econometrie' par l'adresse de votre répertoire.

## Les packages

Pour pouvoir faire appel aux fonctions avancées de R, ou lorsque certaines opérations ne sont pas disponibles avec l'installation de base, il est possible d'“augmenter” R en lui ajoutant des **Packages** : il s'agit de bibliothèques de données ou de commandes, fournies par d'autres utilisateurs de la communauté de R.

Sous réserve que vous connaissez le nom du package dont vous avez besoin, vous pouvez l'installer à R en tapant dans la Console la commande: **install.packages()** (“Nom du package”). Une autre façon d'installer un Package est de passer par le sous-panneau Package, et cliquer sur le bouton **Install**. Vérifiez alors que la case **Install Dependencies** est cochée, afin d'installer automatiquement tout autre Package associé. A la fin de l'installation, il faudra charger le Package afin de pouvoir l'utiliser. La commande **library(Nom du package)** effectue cette opération.

## Importer des données

Afin de procéder à l'analyse de données avec R, il faudra charger ces données d'abord. Si votre base de données existe déjà, vous devez donc l'**importer**. Passez par le menu **File/Import Dataset**. En fonction du type de votre fichier (excel, texte, SAS, Stata ...), choisissez le menu adapté. Nous travaillerons le plus souvent avec des fichiers .xls ou .xlsx.

En sélectionnant alors **File/Import Dataset/From Excel...**, une fenêtre s'ouvrira. Indiquez le répertoire et le fichier Excel concerné avec le bouton **Browse**. Dans **Code Preview**, la commande correspondante est affichée.

### 3 Application 1 : importation d'une base et analyses statistiques

Nous allons, comme application, travailler avec le fichier **UKHP.xls**, à télécharger sur Webcampus. La première étape est d'importer ce fichier dans R. Étant un fichier excel, un package spécifique est nécessaire pour réaliser l'importation : **'readxl'**.

*Q1 : installez le package **'readxl'**. Après l'avoir chargé, importez la base UKHP.xls. Changez le type des colonnes en spécifiant "date" pour la première et "numeric" pour la seconde.*

*Q2 : effectuez une analyse de statistiques descriptives de la base UKHP à l'aide de la commande **summary()**. Concentrez maintenant l'analyse sur la variable **Average House Price***

*Q3 : calculez uniquement la moyenne de cette variable. Pour cela, utilisez l'opérateur d'extraction **\$**. Pour avoir le nom de toutes les variables de la base, tapez **names(UKHP)**. Après calcul de la moyenne, terminez en renommant cette variable **hp**.*

Afin d'analyser l'évolution du marché de l'immobilier résidentiel anglais, nous allons calculer le taux de croissance de la variable **Average House Price**.

*Q4 : créez une nouvelle variable **dhp** représentant ce taux de croissance. Pour cela, utiliser la fonction **diff()** qui calcule la différence entre une ligne *i* et la ligne précédente *i-1*, ainsi que la fonction **nrow()** qui compte le nombre total de lignes. Pour sélectionner une séquence d'éléments, utilisez le symbole : (**premier élément:dernier élément**). Multipliez la formule par 100 pour exprimer le résultat en pourcentage.*

*Une erreur apparaît à la suite de cette commande. En effet, calculer le taux de croissance d'une série implique que la première observation est 'perdue', et le nombre total de taux calculés est réduit de 1 par rapport à la série originale. Corrigez cette erreur en indiquant à R que la première observation est NA. La fonction **c()** permet de créer une collection d'éléments.*

Terminons notre analyse en créant un graphique de l'évolution des prix et un histogramme des rendements. La bibliothèque **graphics** inclut beaucoup de fonctions pour la construction de différents types de graphes. La commande **plot()** est utilisée pour la construction de graphiques tels qu'une courbe, un nuage de point...

*Q5 : tracez une courbe de l'évolution de **Average House Price**, ainsi qu'un histogramme des rendements **hp**. Sauvegardez votre travail à la fin.*

### 4 Application 2 : gestion de base de données et tests statistiques

Nous travaillons ici à partir de données financières, à récupérer sur Yahoo Finance. Nous privilégierons le package **quantmod** pour télécharger directement les données, à l'aide de sa fonction **?getSymbols**.

*Q1 : installez et chargez **quantmod**. Récupérez ensuite, en fréquence journalière et sur la période 01 Janvier 2005 - 31 Décembre 2015, les données pour les tickers suivant : ^GSPC, ^FTSE, ^HSI, ^GDAXI, ^FCHI et ^BFX (respectivement S&P500, FTSE100, HANG SENG INDEX, DAX PERFORMANCE INDEX, CAC40 et BEL20).*

Il est aussi possible que certaines observations soient manquantes. Dans ce cas, utilisez la commande **?na.omit** pour les exclure de la base.

*Q2 : tracez un histogramme de densité des prix à la clôture (colonne **.Close**) sur des graphiques différents (précisez l'option **probability = TRUE** pour calculer la densité), et ajoutez sur chaque graphique une courbe représentant la densité (commandes **?lines** et **density**). Pour les autres options, ajoutez un titre pour les axes et pour le graphique.*

*Optionnel: Modifiez aussi la couleur du contour et du contenu des bars à votre convenance (la liste des couleurs disponibles avec le package **grDevices** s'obtient via : **grDevices::colors()**).*

*Q3 : créez une nouvelle base **closing\_price** reprenant uniquement les prix à la clôture des indices. Supprimez les éventuelles observations manquantes et changez le type de la base en **data.frame**. Effectuez les analyses statistiques descriptives usuelles sur les séries.*

*Q4 : vérifiez l'hypothèse de normalité des séries (**?shapiro.test**).*

*Q5 : calculez les returns journaliers du S&P500 et du FTSE100. Procédez à un test de comparaison de moyennes des returns avec le t-test (**?t.test**).*

*Q6 : créez une nouvelle base de données **crise** avec la période 03 janvier 2007 - 30 Décembre 2010. La commande **?pmatch** vous permet de chercher un élément dans un vecteur et retourner la position de cet élément. Exportez cette nouvelle base au format **.txt** à l'aide de **?write.table**.*