



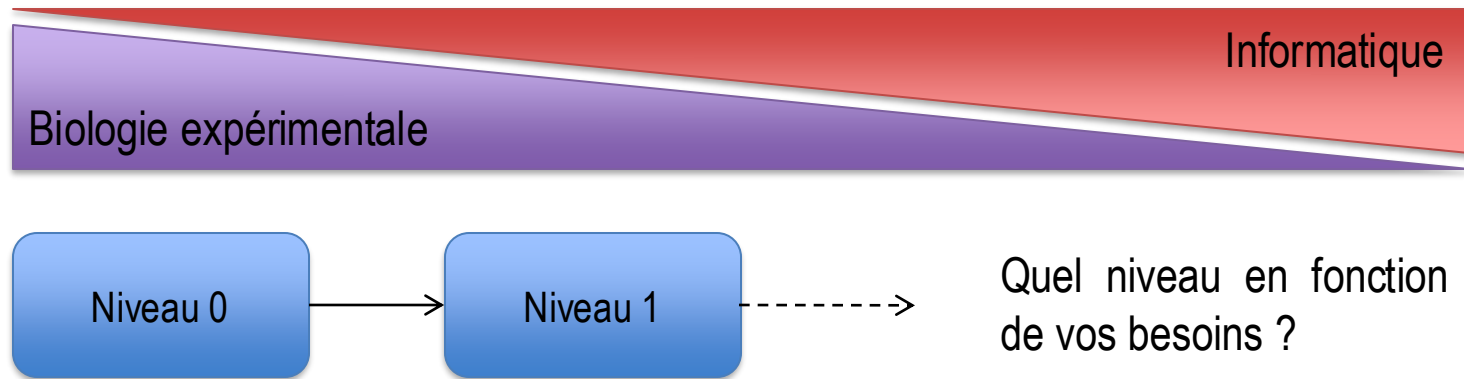
Premiers pas avec les logiciels R et RStudio

Gaëlle LELANDAIS

Version du document : 15/10/2024

Pourquoi apprendre un langage de programmation informatique ?

- Avoir la liberté d'analyser ses données **en autonomie**, d'**utiliser/modifier** des programmes (scripts) codés par des (bio)informaticiens, d'**écrire** des scripts originaux.



Quel(s) objectif(s) ?

- Utiliser l'ordinateur pour réaliser **sans erreur** des tâches (souvent) « simples » mais particulièrement « laborieuses »...

➤ Pour en savoir plus :

Ex :



Stocker



Trier



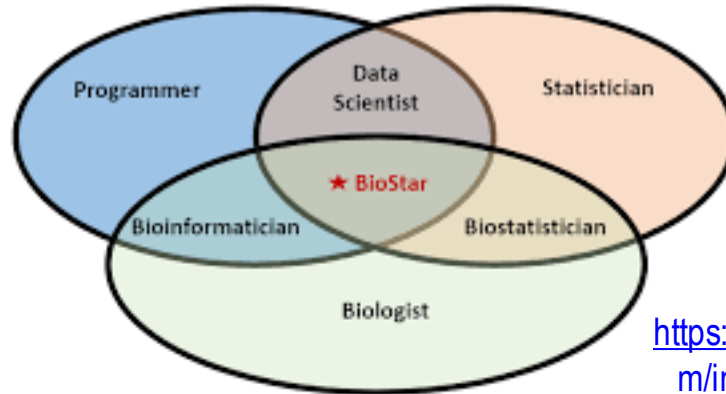
Répéter

Etc.

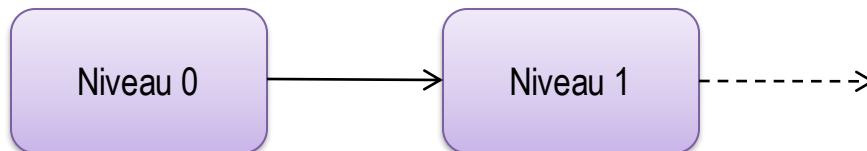
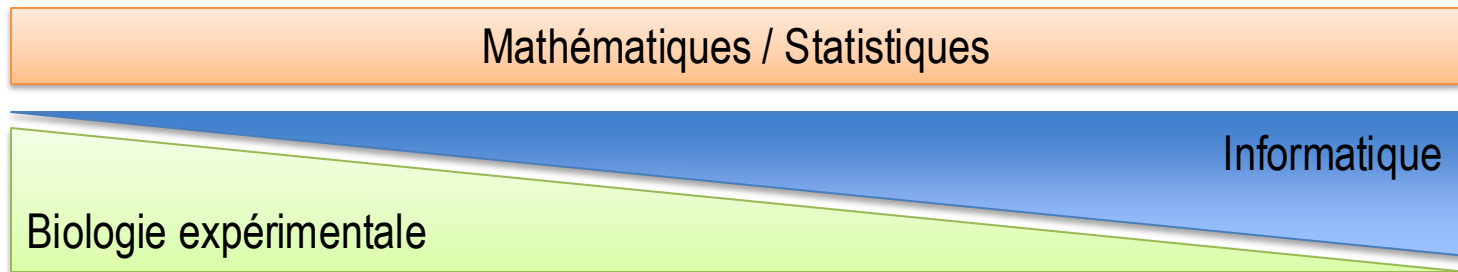
Vidéo d'introduction à la programmation (10 minutes) par Pierre Poulain :

<https://www.youtube.com/watch?v=N9URJ4yVuGA>

Vous, les élèves des filières interdisciplinaires...



<https://www.biostarhandbook.com/index.html#who-is-a-biostar>



Quel langage de programmation ?

- Un **éternel débat** dans les équipes pédagogiques ! Des avantages et des inconvénients pour chaque langage. Chaque enseignant(e) a son expertise qu'elle/il souhaite partager en priorité...

➤ Pour en savoir plus :



« En formation continue (professionnelle), nos stagiaires appartiennent à une communauté de chercheur(e)s « omiques », qui utilisent pour la très grande majorité d'entre eux R »

<https://bioinfo-fr.net/choisir-entre-r-et-python-en-bioinformatique-regards-croises-entre-collegues-enseignants-chercheurs>

Les logiciels R et Rstudio



- **Gratuits**
- **Très utilisés en bioinformatique pour l'analyse de données**
 - Fonctions mathématiques et statistiques,
 - Fonctions de simulation et de modélisation,
 - Représentations graphiques,
 - Etc.
- **De très nombreuses bibliothèques complémentaires**
 - Gestion des données « omiques »,
 - Interaction avec les bases de données publiques,
 - Etc.



Sites Web



<https://cran.rstudio.com/>

The screenshot shows the CRAN website with the following content:

- CRAN logo
- CRAN Mirrors
- What's new?
- Search
- CRAN Team
- About R
- R Homepage
- The R Journal
- Software
- R Sources
- R Binaries
- Packages
- Task Views
- Other
- Documentation
- Manuals
- FAQs
- Contributed

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora/Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2022-10-31, Innocent and Trusting) [R-4.2.2.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

<https://posit.co/download/rstudio-desktop/>

The screenshot shows the Posit website with the following content:

- posit logo
- DOWNLOAD
- RStudio Desktop
- Used by millions of people weekly, the RStudio integrated development environment (IDE) is a set of tools built to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that supports direct code execution. It also features tools for plotting, viewing history, debugging and managing your workspace.
- All Installers
- Step 1: Install R
- RStudio requires R 3.3.0+. Choose a version of R that matches your computer's operating system.

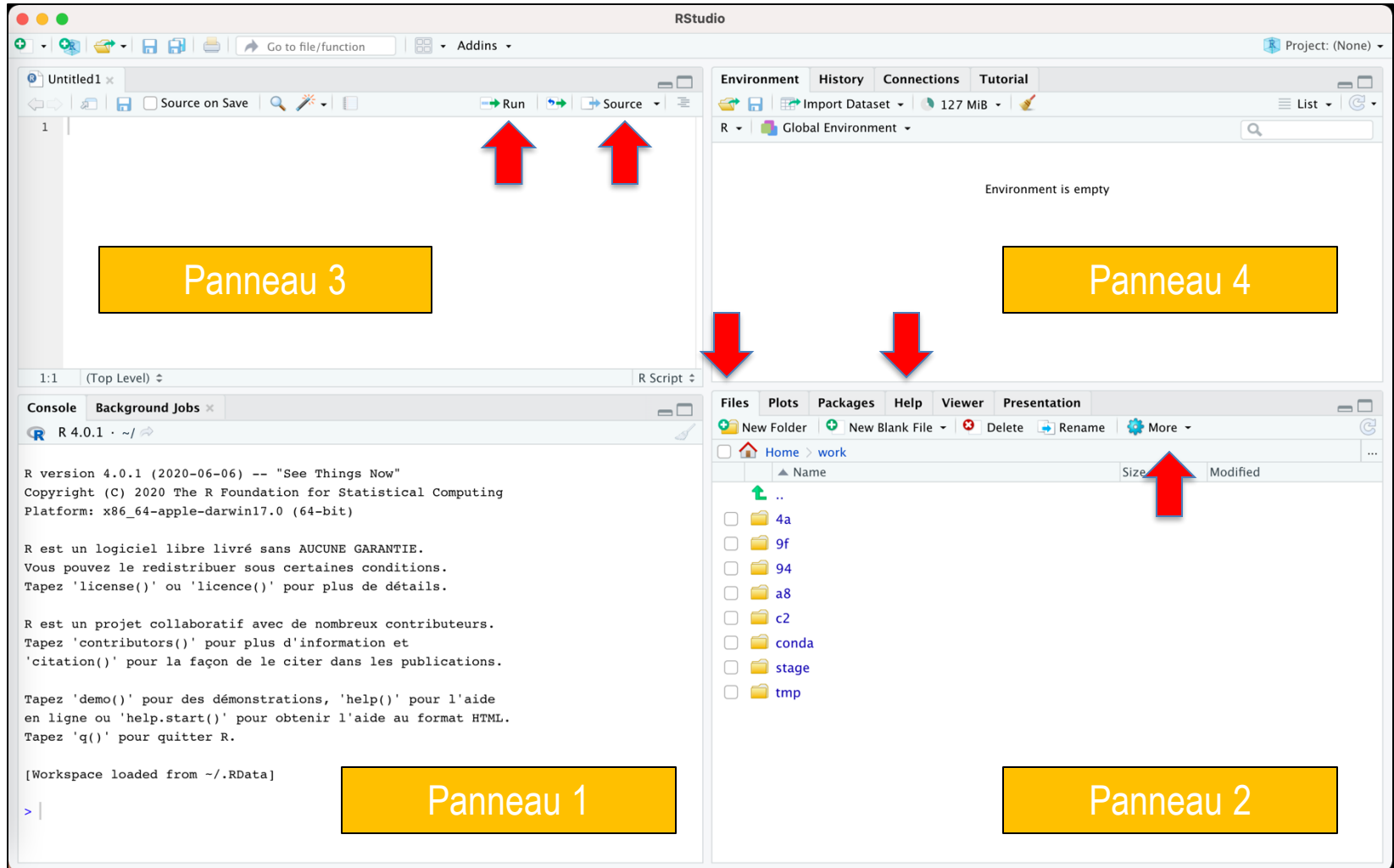
Vue d'ensemble



The image shows a screenshot of the RStudio interface with four yellow boxes highlighting specific areas:

- Panneau 1:** The Console window, showing the R version (4.0.1) and copyright information.
- Panneau 2:** The Files panel, showing a directory structure with folders like 4a, 9f, 94, a8, c2, conda, stage, and tmp.
- Panneau 3:** The Source editor, showing a script file named 'Untitled1'.
- Panneau 4:** The Environment panel, showing the Global Environment.

Vue d'ensemble

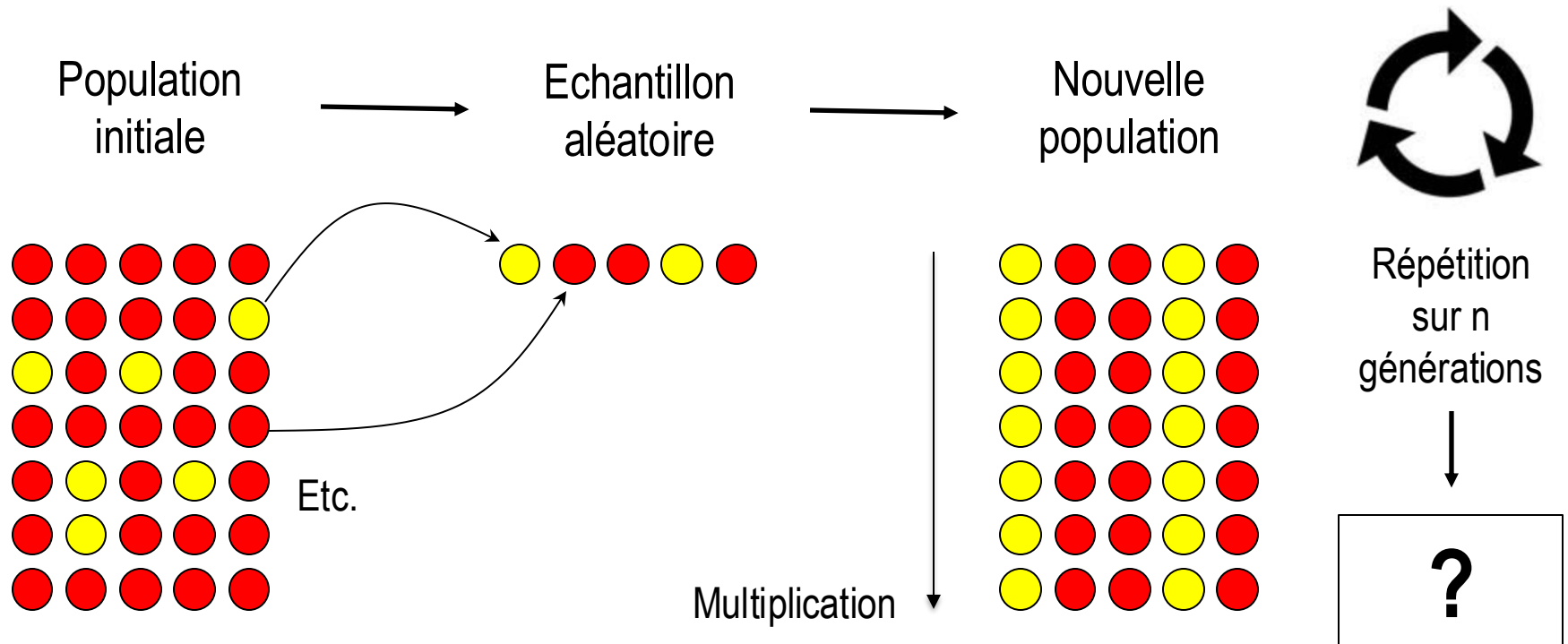


Stratégie d'apprentissage

- Etape 1 :
 - Utiliser sans comprendre,
 - Avec une assistance.
- Etape 2 :
 - Utiliser et comprendre (un peu puis moyennement),
 - Sans programmer.
- Etape 3 :
 - Utiliser et comprendre,
 - Programmer (modifier l'existant, écrire des commandes simples, puis des commandes de plus en plus complexes avec la pratique et l'expérience 😊).

Explication des simulations

- Dérive génétique : processus par lequel les **fréquences alléliques** changent dans **les populations** à cause de biais **aléatoires d'échantillonnage** lors de la transmission des allèles d'une **génération à l'autre** (<https://planet-vie.ens.fr/>).



Script R mis à votre disposition

```
premiers-pas_R-Rstudio_partie1.R x
18
19 #-----
20 # La dérive génétique
21 #-----
22 # Rappel : La dérive génétique est la modification aléatoire de la fréquence
23 # des allèles au sein d'une population au cours de générations
24 # successives.
25
26
27 # Partie 1-----
28 # Créer une population de taille n, comportant des individus de
29 # couleur "rouge" avec une fréquence f.
30 #
31 # ATTENTION : Par "fréquence" ici, il est question de probabilité d'observation
32 # du caractère dans la population
33 # --> Nombre d'individus de couleur "rouge" / Nombre total d'individus.
34 #-----
35
36 # Pour commencer, il est nécessaire de choisir une valeur de n et un valeur
37 # de fréquence f (cf. la définition de la fréquence ci dessus).
38 n = 10
39 f = 0.8
40 # Puis d'utiliser la fonction "sample" de R :
41 pop = sample(c("rouge", "autre"), size = n,
42             replace = T, prob = c(f, 1-f))
43 print(pop)
44
45 # Questions :
46 # Combien d'individus de couleur rouge composent votre population ?
47 # Ce résultat est-il identique à celui de la personne assise à côté de vous ?
48 # Essayer de créer des populations de taille 20, 50 puis 100.
49
50 # Partie 2 -----
51 # Compter le nombre d'individus de couleur rouge et réaliser
52 # une représentation graphique de type "barplot" (diagramme en baron).
53 #-----
54
55 # Il est utile de compter automatiquement le nombre d'individus de couleur
56 # rouge dans la population :
57 nbR = sum(pop == "rouge")
58 print(nbR)
59
124:3  Partie 5
R Script
```

premiers-pas_R-Rstudio_partie1.R



Au programme (1/2)

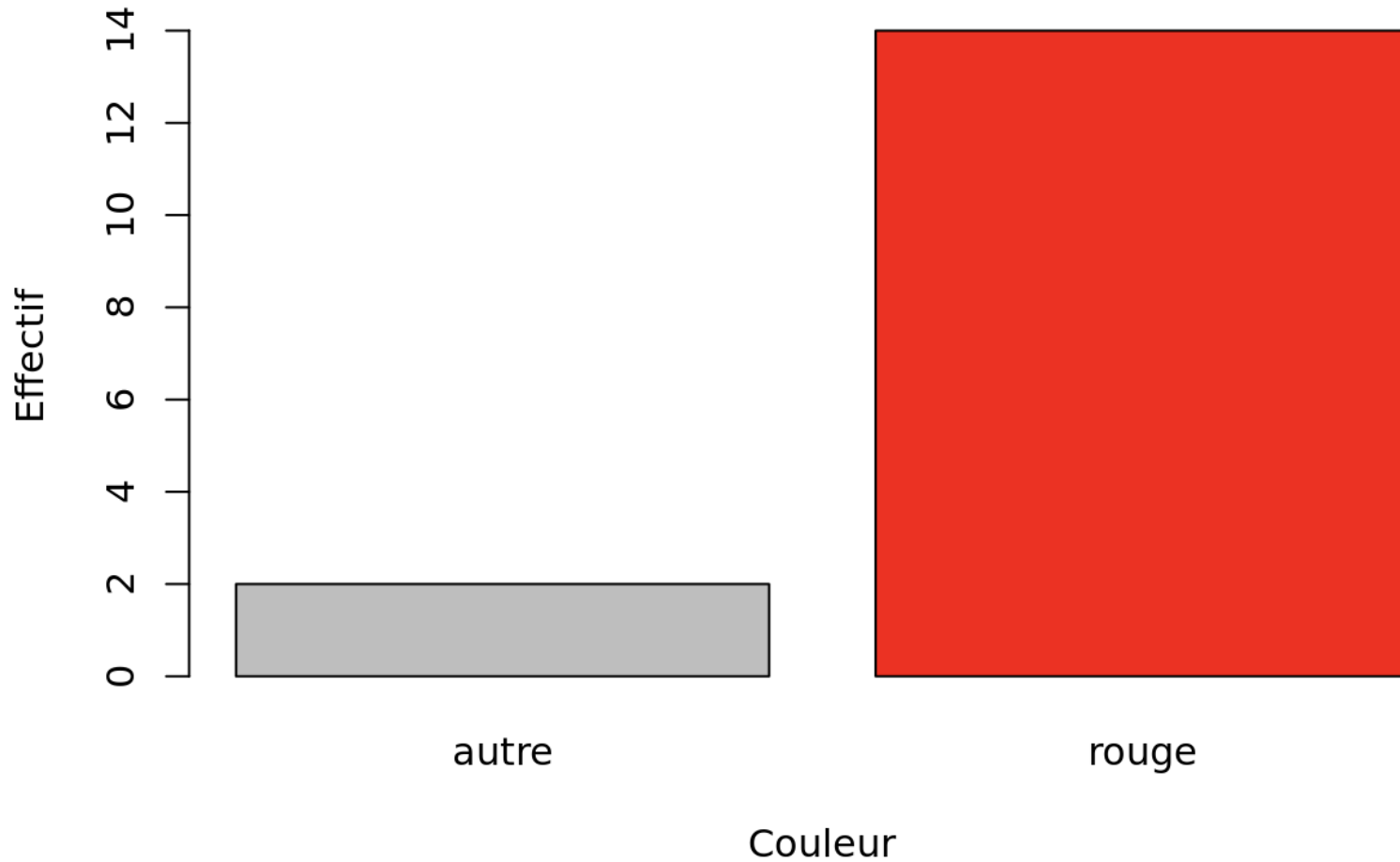
- Premiers pas avec les logiciels R et Rstudio
 - Quatre fenêtres,
 - Vigilance au répertoire de travail,
 - Exécution des commandes écrites dans un script (étape 1 de l'apprentissage d'un langage de programmation),
- Utiliser les logiciels R et Rstudio pour une première simulation en biologie
 - Dérive génétique

Au programme (2/2)

- Travail individuel
 - Réalisation de représentations graphiques,
 - Personnalisation de représentations graphiques.
- Les ressources à votre disposition
 - L'enseignante,
 - Vos notes, internet,
 - La personne qui est assise à côté de vous 😊.

Avez-vous réussi ?

Population de taille $9+7 = 16$ - Résultat de Gaëlle



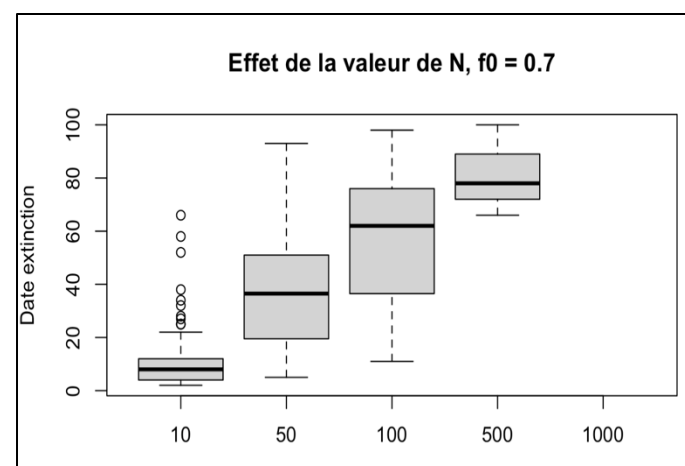
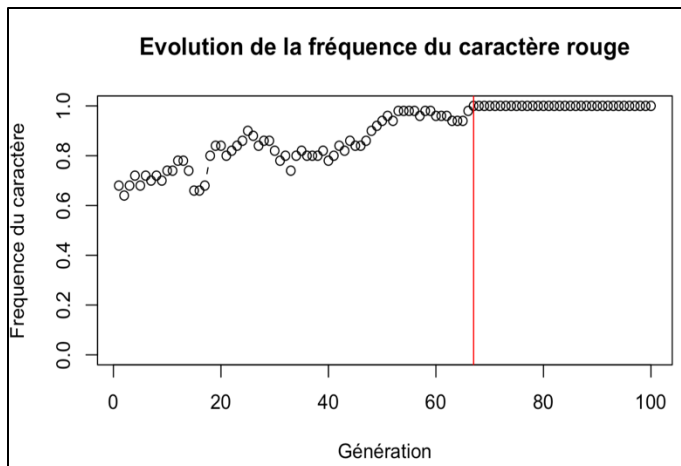
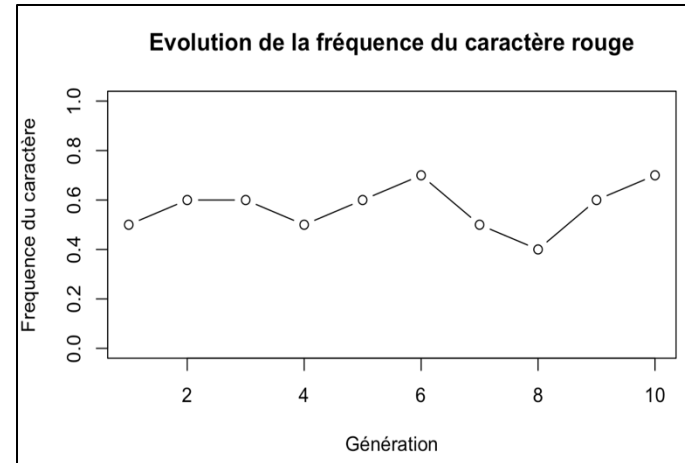
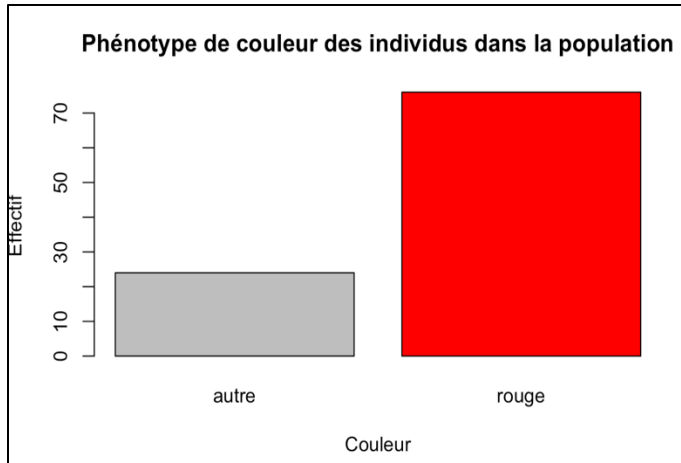
Script R mis à votre disposition (partie 2)

```
premiers-pas_R-Rstudio_partie1.R x
18
19 #-----
20 # La dérive génétique
21 #-----
22 # Rappel : La dérive génétique est la modification aléatoire de la fréquence
23 # des allèles au sein d'une population au cours de générations
24 # successives.
25
26
27 # Partie 1-----
28 # Créer une population de taille n, comportant des individus de
29 # couleur "rouge" avec une fréquence f.
30 #
31 # ATTENTION : Par "fréquence" ici, il est question de probabilité d'observation
32 # du caractère dans la population
33 # --> Nombre d'individus de couleur "rouge" / Nombre total d'individus.
34 #-----
35
36 # Pour commencer, il est nécessaire de choisir une valeur de n et un valeur
37 # de fréquence f (cf. la définition de la fréquence ci dessus).
38 n = 10
39 f = 0.8
40 # Puis d'utiliser la fonction "sample" de R :
41 pop = sample(c("rouge", "autre"), size = n,
42             replace = T, prob = c(f, 1-f))
43 print(pop)
44
45 # Questions :
46 # Combien d'individus de couleur rouge composent votre population ?
47 # Ce résultat est-il identique à celui de la personne assise à côté de vous ?
48 # Essayer de créer des populations de taille 20, 50 puis 100.
49
50 # Partie 2 -----
51 # Compter le nombre d'individus de couleur rouge et réaliser
52 # une représentation graphique de type "barplot" (diagramme en bâton).
53 #-----
54
55 # Il est utile de compter automatiquement le nombre d'individus de couleur
56 # rouge dans la population :
57 nbR = sum(pop == "rouge")
58 print(nbR)
59
124:3  Partie 5  R Script
```

premiers-pas_R-Rstudio_partie2.R



Résultats des simulations



Tutoriels - vidéos

- Vidéos « Premiers pas avec R et Rstudio » :
 - Exemple d'une simulation de la dérive génétique
 - Partie 1 (11 min) : <https://youtu.be/97mRN1IXtB4>
 - (Bonus) Exemple d'une création d'un graphe de co-expression
 - Partie 1 (12 min) : <https://youtu.be/FHVQFrtpw>
 - Partie 2 (7 min) : <https://youtu.be/a8Asps0-hg>

Apprendre à programmer avec R

- Ressources pédagogiques disponibles :

<https://gaellelelandais.github.io/teaching/initiation-R-base/>

Programmer avec le logiciel R

Résumé

Ensemble de ressources pour mieux comprendre les principes de la programmation informatique avec le logiciel R. La philosophie de programmation utilisée ici est de type `r-base`, à distinguer de la philosophie de type `tidyverse`.

Vidéos de cours (commentaires G. Lelandais)

- [Lien vers la vidéo - Cours 1](#) (30 minutes)
- [Lien vers la vidéo - Cours 2](#) (26 minutes)
- [Lien vers la vidéo - Cours 3](#) (14 minutes)

Documents pédagogiques

- [Fichiers PDF des cours](#)
- [Fichiers PDF des exercices](#)
- [Scripts de correction](#)

Ecrire du code informatique, à chacun(e) son style !

- Chaque langage de programmation informatique a son « **guide de style** ».
- Ce sont des règles qui énoncent :
 - la syntaxe :
 - espacements, indentations, opérateur d'assignation, etc.;
 - la façon de nommer les objets et fichiers.
 - l'organisation du code :
 - insertion de commentaires, ordre des éléments dans un programme, répartition du code entre plusieurs fichiers, etc.;
 - etc.
- Exemples de guide de style pour R :
 - <http://web.stanford.edu/class/cs109I/unrestricted/resources/google-style.html>
 - <https://google.github.io/styleguide/Rguide.html>



Sauf mention contraire, ce contenu est mis à disposition selon les termes de la licence Creative Commons Attribution - Partage dans les mêmes conditions 4.0 International (CC BY-SA 4.0)

Gaëlle LELANDAIS

Version du document : 15/10/2024