

Document partagé "Introduction au logiciel R - 2025"

Mercredi matin : rappels

Les versions de Rstudio

- Vincent : RStudio 2023.06.1+524 "Mountain Hydrangea" Release (547dcf861cac0253a8abb52c135e44e02ba407a1, 2023-07-07) for windows Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2023.06.1+524 Chrome/110.0.5481.208 Electron/23.3.0 Safari/537.36
- LBF: RStudio 2025.05.0+496 "Mariposa Orchid" Release (f0b76cc00df96fe7f0ee687d4bed0423bc3de1f8, 2025-05-04) for macOS Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2025.05.0+496 Chrome/132.0.6834.210 Electron/34.5.1 Safari/537.36, Quarto 1.6.42
- Dieynaba : RStudio 2025.05.0+496 "Mariposa Orchid" Release (f0b76cc00df96fe7f0ee687d4bed0423bc3de1f8, 2025-05-04) for macOS Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2025.05.0+496 Chrome/132.0.6834.210 Electron/34.5.1 Safari/537.36, Quarto 1.6.42
- Nithusha: RStudio 2024.12.1+563 "Kousa Dogwood" Release (27771613951643d8987af2b2fb0c752081a3a853, 2025-02-02) for windows Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2024.12.1+563 Chrome/126.0.6478.234 Electron/31.7.7 Safari/537.36, Quarto 1.5.57
- Ines; © 2009-2019 RStudio, Inc.
- Zohar: RStudio 2024.12.1+563 "Kousa Dogwood" Release (27771613951643d8987af2b2fb0c752081a3a853, 2025-02-02) for windows Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2024.12.1+563 Chrome/126.0.6478.234 Electron/31.7.7 Safari/537.36, Quarto 1.5.57
- Bahar: RStudio 2024.12.0+467 "Kousa Dogwood" Release (cf37a3e5488c937207f992226d255be71f5e3f41, 2024-12-11) for macOS Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2024.12.0+467 Chrome/126.0.6478.234 Electron/31.7.6 Safari/537.36, Quarto 1.5.57
- Bogol RStudio 2025.05.0+496 "Mariposa Orchid" Release (f0b76cc00df96fe7f0ee687d4bed0423bc3de1f8, 2025-05-04) for windows

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) RStudio/2025.05.0+496 Chrome/132.0.6834.210 Electron/34.5.1 Safari/537.36, Quarto 1.6.42

Installation du package du cours (intro2r)

```
WARNING: Rtools is required to build R packages but is not currently installed.
Please download and install the appropriate version of Rtools before proceeding:
### EXPLICATION = Rtools est utilisé sous Windows pour compiler des packages,
c'est une utilisation de niveau intermédiaire de R, donc attendez un peu avant
de vous y mettre en tant que débutant.

https://cran.rstudio.com/bin/windows/Rtools/
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.5/purrr_1.0.4.zip'
Content type 'application/zip' length 550490 bytes (537 KB)
downloaded 537 KB

package 'purrr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\gmartinez\AppData\Local\Temp\RtmpeCFGES\downloaded_packages

install.packages("purrr")
trying URL
'https://cran.rstudio.com/bin/macosx/big-sur-x86_64/contrib/4.4/purrr_1.0.4.tgz'
Content type 'application/x-gzip' length 565355 bytes (552 KB)
=====
downloaded 552 KB

The downloaded binary packages are in
/var/folders/mr/0pb421797qd69655_59b2s140000gn/T//RtmpZyU1nh/downloaded_packages
### EXPLICATION = tout a bien fonctionné, malgré ce message un peu étrange

LBF:
> install.packages("purrr")
also installing the dependency 'vctrs'

  There is a binary version available but the source version is later:
    binary source needs_compilation
purrr  1.0.2  1.0.4                TRUE

Do you want to install from sources the package which needs compilation?
(Yes/no/cancel)
### EXPLICATION = R a posé une question et attend la réponse (dans la console)

> install.packages("purrr")
trying URL
```

```
'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.5/purrr_1.0.4.tgz'
Content type 'application/x-gzip' length 570051 bytes (556 KB)
=====
downloaded 556 KB
```

The downloaded binary packages are in

```
/var/folders/y3/zn09ss79749_wlqtsxnj_jvh0000gn/T//RtmpdgALwm/downloaded_packages
```

```
> install.packages("purrr")
```

```
WARNING: Rtools is required to build R packages but is not currently installed.
Please download and install the appropriate version of Rtools before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/
```

Installation du package dans 'C:/Users/ICAP/AppData/Local/R/win-library/4.5'
(car 'lib' n'est pas spécifié)

essai de l'URL

```
'https://cran.rstudio.com/bin/windows/contrib/4.5/purrr_1.0.4.zip'
```

```
Content type 'application/zip' length 550490 bytes (537 KB)
```

```
downloaded 537 KB
```

Des versions binaires sont disponibles mais les versions des sources sont plus récentes:

	binary	source	needs_compilation
cli	3.6.2	3.6.5	TRUE
purrr	1.0.2	1.0.4	TRUE

Voulez-vous installer depuis les sources les packages qui nécessitent une compilation ? (Oui/non/annuler) no

EXPLICATION = R a posé une question et attend la réponse (dans la console)

```
remotes::install_github("vguillemot/intro2r")
```

```
Error in loadNamespace(x) : there is no package called 'remotes'
```

Explication : il faut installer remotes

EXPLICATION = Il faut installer le package remotes

```
Downloading GitHub repo vguillemot/intro2r@HEAD
```

```
Error: Failed to install 'intro2r' from GitHub:
```

```
input string 23 is invalid
```

```
In addition: Warning message:
```

```
In gsub("[^/]", "", file_list) :
```

```
unable to translate 'vguillemot-intro2r-6f03388/inst/extdata/Document
Partag+<ac> ILR 2022.pdf' to a wide string
```

EXPLICATION = il y avait un problème avec les caractères accentués dans le package sur Github

```
remotes::install_github("vguillemot/intro2r")
```

```
Downloading GitHub repo vguillemot/intro2r@HEAD
```

```
These packages have more recent versions available.
```

```
It is recommended to update all of them.
```

Which would you like to update?

```
1: All
2: CRAN packages only
3: None
4: withr      (2.5.0 -> 3.0.2 ) [CRAN]
5: glue       (1.6.2 -> 1.8.0 ) [CRAN]
6: cli        (3.6.2 -> 3.6.5 ) [CRAN]
7: utf8       (1.2.2 -> 1.2.5 ) [CRAN]
8: pillar     (1.8.1 -> 1.10.2) [CRAN]
9: fansi      (1.0.3 -> 1.0.6 ) [CRAN]
10: viridisLite (0.4.1 -> 0.4.2 ) [CRAN]
11: R6         (2.5.1 -> 2.6.1 ) [CRAN]
12: labeling   (0.4.2 -> 0.4.3 ) [CRAN]
13: tibble     (3.1.8 -> 3.2.1 ) [CRAN]
14: scales     (1.2.1 -> 1.4.0 ) [CRAN]
15: isoband    (0.2.5 -> 0.2.7 ) [CRAN]
16: gtable     (0.3.1 -> 0.3.6 ) [CRAN]
```

Enter one or more numbers, or an empty line to skip updates:

EXPLICATION = R a posé une question et attend la réponse (dans la console)

Pour charger le package, importer les données, et explorer les fonctionnalités du package :

```
# Chargement du package
library(intro2r)
# Exécution de la fonction "bonjour"
bonjour()
# Affichage de l'aide de la fonction "bonjour"
?bonjour
# Chargement en mémoire des données "fruits et légumes"
data("fruveg", package = "intro2r")
# Affichage des données dans la console
fruveg
```

Exercice

1. Extrayez la colonne de la teneur en sucres de la table des fruits et légumes... de deux façons différentes !
2. Créez un objet contenant la teneur en sucres : quelle est la classe de cet objet ?

Solution proposée

```
fruveg$sugar # pour extraire la teneur en sucre
teneur_en_sucres <- fruveg$sugar # enregistrer la teneur en sucre dans un objet
class(teneur_en_sucres) # demander la classe de l'objet
```

Problème rencontrés :

```
> data_fruveg <- fruveg::sugar
```

```

Error in loadNamespace(x) : there is no package called 'fruveg'
### EXPLICATION = le :: est réservé à la syntaxe package::function. C'est
le $ qui doit s'utiliser ici
> data_fruveg <- "fruveg"::sugar
Error in loadNamespace(x) : there is no package called 'fruveg'
### EXPLICATION = le :: est réservé à la syntaxe package::function. C'est
le $ qui doit s'utiliser ici
> sugar <- data_fruveg$sugar
Error: object 'data_fruveg' not found
### EXPLICATION = l'objet data_fruveg n'existe pas
> `class<-`(sugar)
Error in `class<-`(sugar) :
  1 argument passed to 'class<-' which requires 2
### EXPLICATION = la commande est class(objet)

```

D'autres solutions sont possibles : en utilisant les crochets carrés ou en empruntant des chemins de traverse. Attention, chaque solution a ses petites particularités : R est un langage qui permet de réaliser des tâches complexes donc parfois le résultat que l'on souhaite, même s'il est simplement exprimé, nécessite une suite d'étapes qui ne sont pas intuitives.

```

# Extraire la colonne "sugar"
fruveg.sugar <- fruveg[,c("sugar")]
class(fruveg.sugar)
[1] "tbl_df"      "tbl"          "data.frame"

# LBF:
fruveg$teneur<-c(1.43,1.79,3.15,2.48,1.70,3.26,6.76,4.20,1.60,2.76,4.46,0.9
9,6.70,1.10,2.79,1.59,0.47,1.00,2.90,4.00,2.11,6.00,9.68,6.30,5.40,6.63,6.1
0,10.10,8.10,11.00,9.92,16.50,4.17)
class(teneur)
Error: object 'teneur' not found
### EXPLICATION = une colonne "teneur" a été ajoutée au tableau, donc
l'objet "teneur" n'a pas été créé, par conséquent, on ne peut en déduire sa
classe

```

Mercredi après-midi : rappels

Exercice

Créez un vecteur groupe contenant les groupes de fruits et légumes. Donnez deux façons différentes d'extraire les dix premières valeurs de ce vecteur.

Solution proposée

```
groupe <- fruveg$group
groupe[1:10]
groupe[seq(1, 10, 1)]
```

Comment sélectionner des fruits et légumes ?

```
## Première option
fruveg[fruveg$water >= 90, ]
fruveg[
  fruveg$water >= 90 &
  fruveg$group == "raw vegetable", ]
## Deuxième option
groupe <- fruveg$group
eau <- fruveg$water
fruveg[eau >= 90, ]
fruveg[eau >= 90 &
  groupe == "raw vegetable", ]
```

Exercice

Construisez la sous-table contenant la teneur en protéines, en glucides et en lipides des fruits crus.

Solution proposée

```
desfruits = fruveg[[2]]
teneur_en_prot = fruveg[[5]]
teneur_en_gluc = fruveg[[6]]
teneur_en_lipi = fruveg[[7]]
df_teneurs <- data.frame(desfruits, teneur_en_gluc, teneur_en_lipi,
  teneur_en_prot)
df_teneurs = df_teneurs[23:33,]
```

Alternative

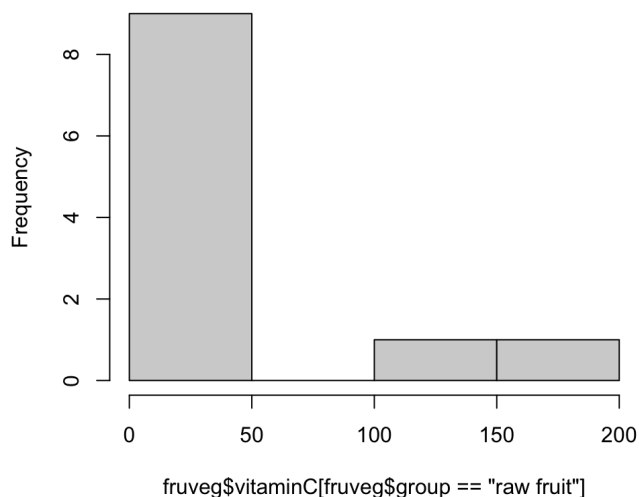
```
minifruitscrus <- fruveg[
  fruveg$group == "raw fruit",
  c("group", "proteins", "carbohydrates", "lipids")]
```

Exercice

Faire un histogramme de la teneur en Vitamine C des fruits crus.

```
hist(fruveg$vitaminC[fruveg$group == "raw fruit"])
```

Histogram of fruveg\$vitaminC[fruveg\$group == "raw fru



fruveg_sugar <- fruveg\$sugar Visualisation avec ggplot2

On commence par charger le package ggplot2, ainsi que les données

```
library(ggplot2)
data("fruveg", package = "intro2r")
```

Les fonctionnalités de ggplot2 permettent de créer des graphiques en suivant toujours la même "recette" : on définit le canevas vide à partir des données et des paramètres esthétiques, puis on demande d'utiliser un "pinceau" (la géométrie) spécial pour transformer les données en graphe.

```
## Mon premier ggplot
ggplot(
  data = fruveg,
  mapping = aes(
    x = sugar,
    y = energy,
    color = group)) +
  geom_point()
```

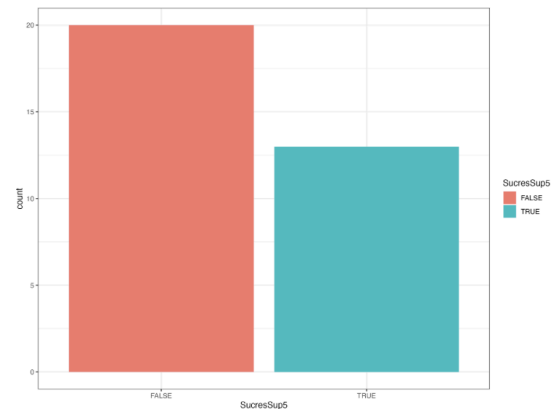
```
## La syntaxe suivante est équivalente
ggplot(fruveg, aes(sugar, energy, color = group)) +
  geom_point()
```

A vous !

À quoi servent les deux premières lignes ?
Comment corriger la commande suivante
pour obtenir le graphe à droite ?

```
fruveg$SucresSup5 <-
  fruveg$sugar > 5

ggplot(***,
  aes(***,
    fill = SucresSup5)) +
  geom_***()
```



Solution proposée

```
## On crée une nouvelle colonne
fruveg$SucresSup5 <-
  fruveg$sugar > 5

## On s'en sert pour compter et colorier
ggplot(
  data = fruveg,
  mapping = aes(
    x = SucresSup5,
    fill = SucresSup5)) +
  geom_bar()
```

Alternative

```
ggplot(
  fruveg,
  aes(sugar>5, fill = sugar>5)) +
  geom_bar()
```

Diagrammes de Tukey

```
ggplot(
  data = fruveg,
  mapping = aes(
    x = group,
    y = sugar,
```



```
    fill = group)) +  
  geom_boxplot()
```

Attention aux représentations qui sont parfois un peu trop synthétiques.

```
## On génère des données "bimodales" (càd avec deux pics observables  
dans la distribution)  
dat.bimodale <- data.frame(  
  x = c(  
    rnorm(100, -10),  
    rnorm(100, + 10))  
)  
  
ggplot( # on voit les deux pics sur un histogramme  
  data = dat.bimodale,  
  mapping = aes(x = x)) +  
  geom_histogram()  
ggplot( # les pics disparaissent sur le boxplot  
  data = dat.bimodale,  
  mapping = aes(x = x)) +  
  geom_boxplot()  
ggplot( # mais ils réapparaissent sur le diagramme en violon  
  data = dat.bimodale,  
  mapping = aes(x = x, y = 1)) +  
  geom_violin()  
ggplot( # on peut être créatif avec les boxplots  
  data = dat.bimodale,  
  mapping = aes(x = x, y = 1)) +  
  geom_boxplot() +  
  geom_jitter()
```

On peut personnaliser le graphe presque à l'infini !!

```
ggplot(  
  data = fruveg,  
  mapping = aes(  
    x = phosphorus,  
    y = calcium,  
    size = sugar,  
    color = group)) +  
  geom_point() +  
  theme_minimal() +  
  labs(  
    title = "Analyse des fruits et légumes",  
    x = "Teneur en Phospore (mg/100g)",  
    y = "Teneur en Calcium (mg/100g)",  
    color = "Groupe",
```

```
size = "Teneur en\nSucre (g/100g)",  
caption = "Fig. 1: le nuage de points des fruits et légumes")
```

Jeudi Matin

Révisions sur l'import des données et ggplot2

```
PK...  
1  
\\-\x9cf\xfb\xden\xe8*>\x81R\x86\xd7\036Jq\xc2%\f\xfb\020@\xe8\xd5\xfdp  
\xfbo,\x8e\017\b.Ik^\xc0D\xee\fi>H\xce\004MU\xd7{  
Warning messages:  
1: In read.table(file = file, header = header, sep = sep, quote = quote,  
:  
  line 1 appears to contain embedded nulls  
2: In read.table(file = file, header = header, sep = sep, quote = quote,  
:  
  line 2 appears to contain embedded nulls  
3: In read.table(file = file, header = header, sep = sep, quote = quote,  
:  
  incomplete final line found by readTableHeader on  
'fruits_and_veggies.xlsx'  
### EXPLICATION = read.csv ne peut pas lire de fichier Excel  
  
## Chargement des librairies  
library(ggplot2)  
library(readxl)  
## Chargement des données au format xlsx  
fruveg <- read_excel("fruits_and_veggies.xlsx")  
  
## Nuage de point avec personnalisation des titres et du thème  
ggplot(  
  data = fruveg,  
  mapping = aes(  
    x = sugar,  
    y = proteins,  
    color = group,  
    size = magnesium)) +  
  geom_point() +  
  labs(  
    x = "Teneur en Sucres",  
    y = "Teneur en Protéines",  
    color = "Groupe",  
    size = "Teneur en\nMagnesium") +
```

```
theme_bw()
```

Exercice

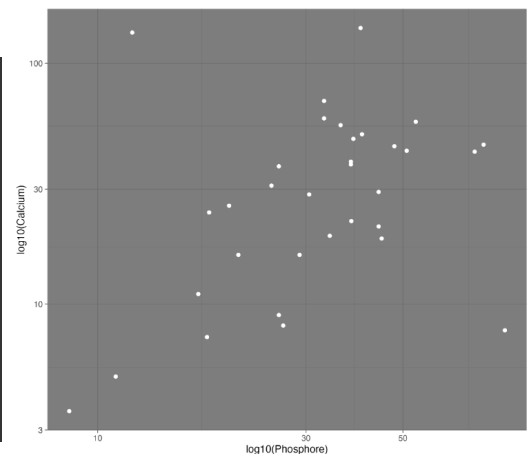
1. Consultez la page d'aide de la fonction `theme_bw` avec la commande `?theme_bw`
2. Choisissez le thème permettant de réaliser le graphe à droite en complétant la commande suivante.

```
ggplot(fruveg, aes(fibers)) +  
  geom_boxplot() +  
  theme_***()
```

A vous !

Complétez la commande suivante pour obtenir le graphe ci-contre.

```
ggplot(fruveg,  
       aes(phosphorus,  
           calcium)) +  
  geom_point(*** = "white") +  
  scale_***() +  
  scale_***() +  
  labs(x = "log10(Phosphore)",  
       y = "log10(Calcium)") +  
  theme_dark()
```



Solution proposée

```
ggplot(fruveg,  
       aes(phosphorus,  
           calcium)) +  
  geom_point(color = "white") +  
  scale_x_log10() +  
  scale_y_log10() +  
  labs(x = "log10(Phosphore)",  
       y = "log10(Calcium)") +  
  theme_dark()
```

Exemples de “facettes”

Quand on a une seule variable qualitative pour définir les facettes du graphe, on utilise le plus souvent la fonction `"facet_wrap"`, qui va les arranger en lignes (et en colonnes) en

fonction des modalités de cette variable.

```
ggplot(  
  data = fruveg,  
  mapping = aes(  
    x = phosphorus,  
    y = calcium,  
    color = group)) +  
  geom_point() +  
  facet_wrap(vars(sugar > 5)) +  
  theme_bw() +  
  theme(legend.position = "bottom")
```

Quand on a deux variables qualitatives sur lesquelles s'appuyer pour faire les facettes, on utilise la fonction `facet_grid`.

```
## On crée de nouvelles variables pour pouvoir contrôler les  
dénominations des catégories créées  
fruveg$sugarsup5 <- ifelse(  
  fruveg$sugar > 5,  
  "Sugar > 5",  
  "Sugar <= 5")  
fruveg$watersup90 <- ifelse(  
  fruveg$water > 90,  
  "Water > 90",  
  "Water <= 90")  
  
## On se sert de ces nouvelles variables pour les facettes  
ggplot(  
  data = fruveg,  
  mapping = aes(  
    x = phosphorus,  
    y = calcium,  
    color = group)) +  
  geom_point() +  
  facet_grid(  
    rows = vars(sugarsup5),  
    cols = vars(watersup90)) +  
  theme_bw() +  
  theme(legend.position = "bottom")
```

Exemple de courbes

```
## Chargement du package ggplot2
```

```
library(ggplot2)
```

```
#
```

Exemple de “petites” étoiles

```
## Chargement des packages
library(ggplot2)
library(ggpubr)

## Création d'un jeu de données aléatoire
df.etoile <- data.frame(
  class = rep(c("A", "B"), each = 3),
  expr = c(rnorm(3, 4), rnorm(3, 25))
)

## Boîtes à moustaches sans étoiles
ggplot(
  data = df.etoile,
  mapping = aes(
    x = class,
    y = expr,
    color = class)) +
  geom_boxplot() +
  theme_minimal()

## Boîtes à moustaches avec étoile
ggplot(
  data = df.etoile,
  mapping = aes(
    x = class,
    y = expr,
    color = class)) +
  geom_boxplot() +
  stat_compare_means(
    method = "wilcox.test",
    label = "p.signif",
    comparisons = list(c("A", "B")),
    label.y = 30) + # Ajuste la hauteur du texte
  theme_minimal()
```

Exemple de test de Student

```
## Chargement des librairies
library(ggplot2)
library(ggpubr)
library(openxlsx)

## Chargement des données
fruveg <- read.xlsx("fruits_and_veggies.xlsx")

## Création d'une variable binaire
fruveg$eauqual <- cut(fruveg$water, c(0, 90, 100))

## Visualisation
ggplot(
  data = fruveg,
  mapping = aes(
    x = eauqual,
    y = vitaminC)) +
  geom_boxplot() +
  stat_compare_means(
    method = "t.test",
    label = "p.signif",
    comparisons = list(c("(0,90]", "(90,100]"))) +
  theme_bw()

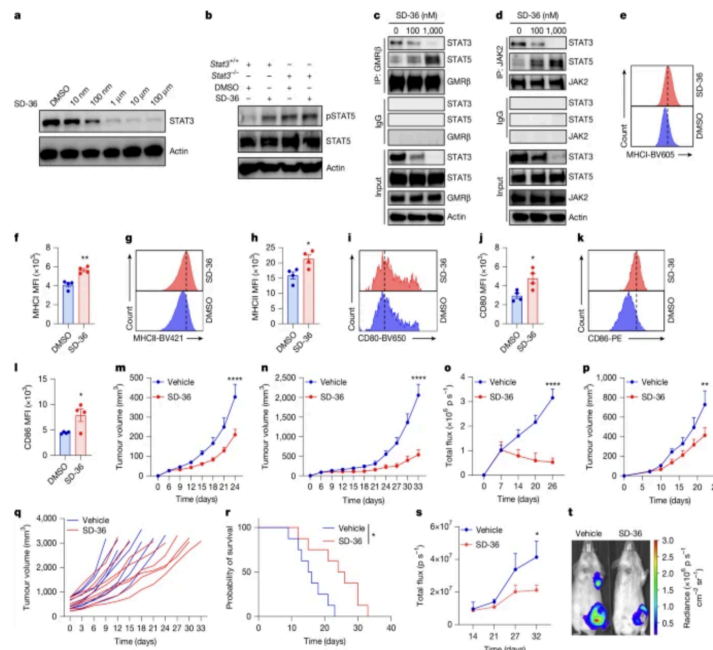
## Test de Student
t.test(vitaminC ~ eauqual, data = fruveg)
```

Travaux pratiques du jeudi après-midi

L'objectif de cette séance est de reproduire, en R et avec ggplot2, le maximum de la Figure 4 de l'article "Zhou, J., Tison, K., Zhou, H. *et al.* STAT5 and STAT3 balance shapes dendritic cell function and tumour immunity. *Nature* (2025)".

(<https://doi.org/10.1038/s41586-025-09000-3>)

Fig. 4: STAT3 degraders for treatment of advanced tumours.



Si vous le souhaitez, vous pouvez choisir une autre publication dont la thématique se rapproche plus de vos centres d'intérêts. Demandez-moi quand même avant de vous lancer pour que je puisse vous dire si c'est réalisable. Attention, parfois les données ne sont pas disponibles, ou bien sont dans un format difficile à importer !

Voici quelques étapes que vous pouvez suivre pour structurer votre travail :

1. Formez des petits groupe de travail et installez vous dans un coin de la salle (si besoin, trouvez une autre salle ou un endroit plus confortable pour travailler)
2. Allez sur la version en ligne de la publication et téléchargez les données associées à la figure
3. Lisez également rapidement l'article pour vous familiariser avec le contexte.
4. Reproduisez pendant le temps que vous avez quelques sous-figures de la Figure choisie.
5. Envoyez moi par mail (vincent.guillemot@pasteur.fr) le dossier compressé comprenant votre projet (avec tout ce qu'il y a à l'intérieur). Le projet doit...
 - a. s'appeler avec le nom du binôme ([p.ex.](#) : guillemot_madec)
 - b. contenir un ou plusieurs scripts **commentés**
 - c. contenir les données
 - d. contenir les figures produites au format PNG
6. On corrigera ensemble

Vous trouverez ci-dessous deux dossiers de projet compressés avec deux propositions pour reproduire la figure 4. La première correspond à ce que vous avez fait pendant la séance de TP, que nous avons ensuite retravaillé. La deuxième proposition est une proposition plus complète, mais qui utilise des packages que nous n'avons pas eu le temps de voir ensemble !

- [Proposition numéro 1 \(deux figures\)](#)
- [Proposition numéro 2 avec plus de figures](#)