Introduction à R

Ecole de Bioinformatique Aviesan-IFB 2018

Hugo Varet, Olivier Kirsh et Jacques van Helden

2019-01-15

Se connecter au serveur RStudio de Roscoff

http://r.sb-roscoff.fr/auth-sign-in

Aller dans son dossier de travail

Définir une variable qui indique le chemin du dossier de travail

```
work.dir <- "~/intro_R"
```

S'il n'existe pas encore, créer le dossier de travail. (Commande Unix équivalente: "mkdir -p ~/intro_R")

```
dir.create(work.dir, recursive = TRUE, showWarnings = FALSE
```

Aller dans ce dossier de travail. (Commande Unix équivalente: "cd ~/intro_R")

```
setwd(work.dir)
```

Où suis-je ? (Commande Unix équivalente: "pwd")

```
getwd()
```

R vu comme une calculatrice

```
2 + 3
4 * 5
6 / 4
```

Notion de variable/objet

```
a <- 2        ## Assigner une valeur à une variable
print(a)      ## Afficher la valeur de la variable a
```

```
b <- 3        ## Assigner une valeur à une seconde variable
c <- a + b    ## Effectuer un calcul avec 2 variables
print(c)      ## Afficher le contenu de la variable c
```

```
a <- 7        ## Changer la valeur de a
print(c)      ## Note: le contenu de c n'est pas modifié
```

Télécharger un fichier

La commande `download()` permet de télécharger un fichier à partir d'un serveur.

```
## download.file(url = "https://goo.gl/9QVAg6", destfile =
```

```
## download.file(url = "https://goo.gl/NQWnHg", destfile =
```

Chargement des données

Charger le contenu du fichier “expression.txt” dans une variable nommée “exprs”.

```
exprs <- read.table(file = "expression.txt", header = TRUE,
```

Accéder à l'aide d'une fonction

```
help(read.table)
```

Notation alternative

```
?read.table
```

Affichage de l'objet "exprs"

Imprimer toutes les valeurs.

```
print(exprs)
```

	id	WT1	WT2	KO1	KO2
1	ENSG00000034510	235960	94264	202381	91336
2	ENSG00000064201	116	71	64	56
3	ENSG00000065717	118	174	124	182
4	ENSG00000099958	450	655	301	472
5	ENSG00000104164	4736	5019	4845	4934
6	ENSG00000104783	9002	8623	7720	7142
7	ENSG00000105229	1295	2744	1113	2887
8	ENSG00000105723	3353	7449	3589	7202
9	ENSG00000116199	2044	4525	2604	4902
10	ENSG00000118939	7022	2526	6269	3068
11	ENSG00000119285	15783	17359	18591	20077

Affichage des premières lignes de l'objet

```
head(exprs)
```

	id	WT1	WT2	KO1	KO2
1	ENSG00000034510	235960	94264	202381	91336
2	ENSG00000064201	116	71	64	56
3	ENSG00000065717	118	174	124	182
4	ENSG00000099958	450	655	301	472
5	ENSG00000104164	4736	5019	4845	4934
6	ENSG00000104783	9002	8623	7720	7142

Un peu plus de lignes

```
head(exprs, n = 15)
```

	id	WT1	WT2	KO1	KO2
1	ENSG00000034510	235960	94264	202381	91336
2	ENSG00000064201	116	71	64	56
3	ENSG00000065717	118	174	124	182
4	ENSG00000099958	450	655	301	472
5	ENSG00000104164	4736	5019	4845	4934
6	ENSG00000104783	9002	8623	7720	7142
7	ENSG00000105229	1295	2744	1113	2887
8	ENSG00000105723	3353	7449	3589	7202
9	ENSG00000116199	2044	4525	2604	4902
10	ENSG00000118939	7022	2526	6269	3068
11	ENSG00000119285	15783	17359	18591	20077
12	ENSG00000121680	3133	2775	2045	2796
13	ENSG00000125384	1380	3079	869	2419

Caractéristiques d'un tableau

Dimensions

```
dim(exprs)     ## Dimensions
ncol(exprs)    ## Nombre de colonnes
nrow(exprs)    ## Nombre de lignes
```

Noms des lignes et colonnes

```
colnames(exprs)
rownames(exprs)
```

Résumé rapide des données par colonne

```
summary(exprs)
```

```
                 id             WT1                WT2              
 ENSG00000034510: 1   Min.   :    31   Min.   :   43.0   Mi
 ENSG00000064201: 1   1st Qu.:   264   1st Qu.:  203.2   1s
 ENSG00000065717: 1   Median :  1338   Median : 1903.0   Me
 ENSG00000099958: 1   Mean   :  9358   Mean   : 6498.6   Me
 ENSG00000104164: 1   3rd Qu.:  3730   3rd Qu.: 4727.2   3r
 ENSG00000104783: 1   Max.   :235960   Max.   :94264.0   Ma
 (Other)        :44
```

Sélection de colonnes d'un tableau

Valeurs stockées dans la colonne nommée "WT1"

```
exprs$WT1
```

Notation alternative

```
exprs[, "WT1"]  ## Sélection de la colonne WT1
```

Sélection de plusieurs colonnes.

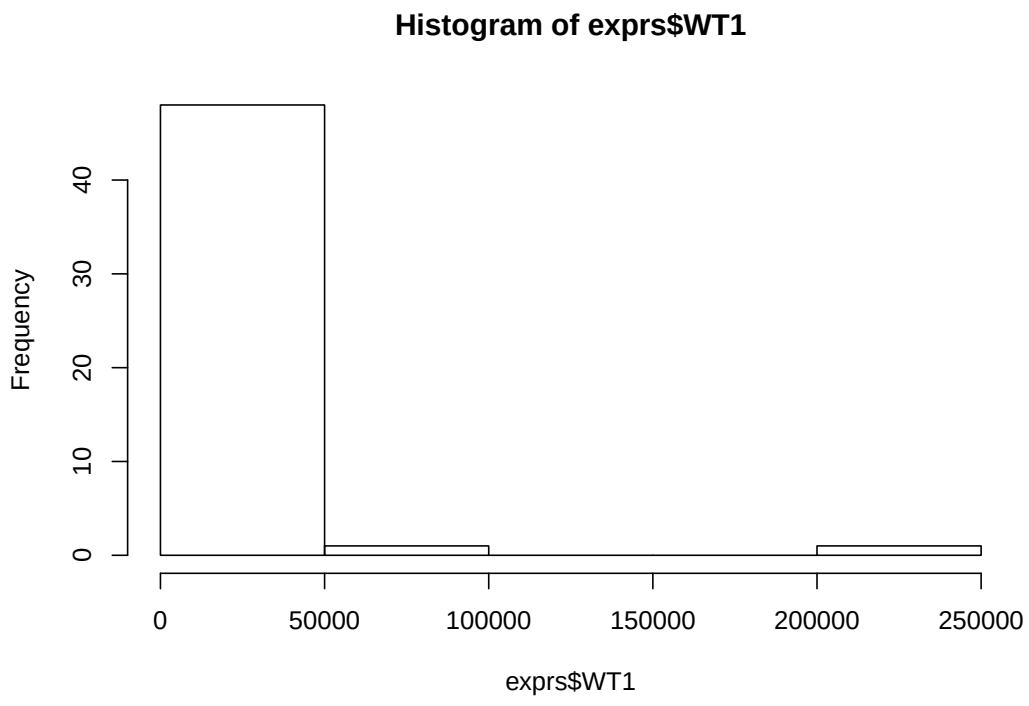
```
exprs[, c("WT1", "WT2")]
```

Sélection de colonnes par leur indice

```
exprs[, 2]
exprs[, c(2, 3)]
```

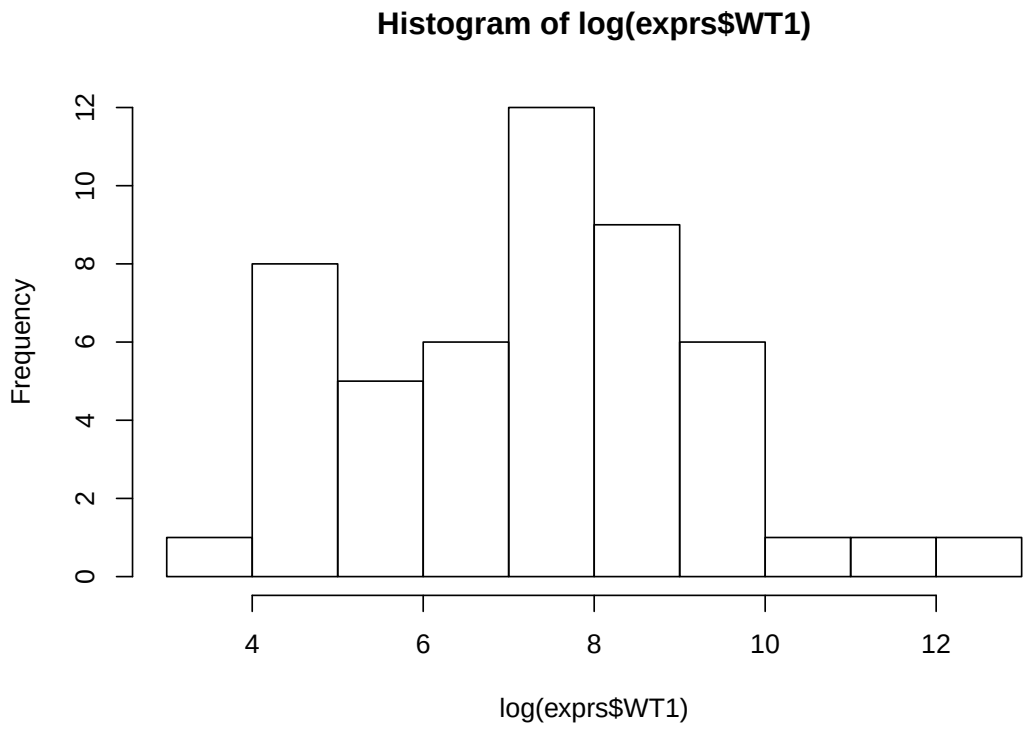
Histogramme des valeurs d'expression pour WT1

```
hist(exprs$WT1)
```



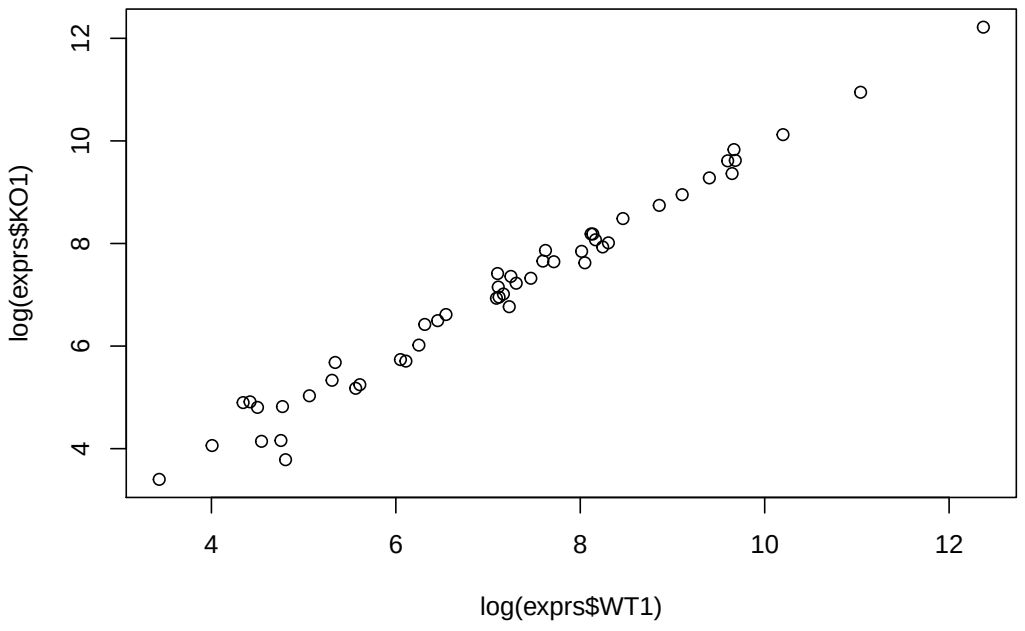
Histogramme du logarithme de ces valeurs

```
hist(log(exprs$WT1))
```



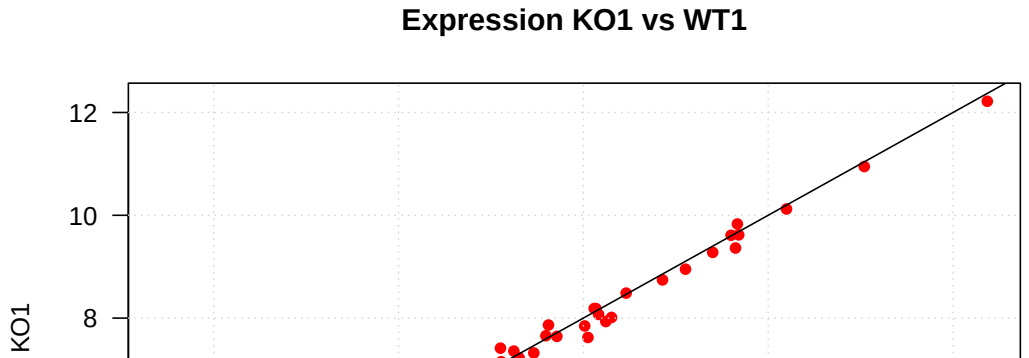
Nuages de points – Expressions KO1 vs WT1

```
plot(x = log(exprs$WT1), y = log(exprs$KO1))
```



Personnalisation des paramètres graphiques

```
plot(x = log(exprs$WT1),   ## données pour l'abscisse
     y = log(exprs$KO1),   ## données pour l'ordonnée
     main = "Expression KO1 vs WT1",  ## Titre principal
     xlab = "WT1",  ## légende de l'axe X
     ylab = "KO1",  ## légende de l'axe Y
     pch = 16,      ## caractère pour marquer les points
     las = 1,       ## écrire les échelles horizontalement
     col = "red")   ## couleur des points
grid()  ## Ajout d'une grille
abline(a = 0, b = 1)   ## Ajouter la droite X = Y (intercep
```



Sélection de lignes d'un tableau

Sélection des lignes 4 et 11 du tableau des expressions

```
exprs[c(4, 11), ]
```

Indices des lignes correspondant aux IDs ENSG00000253991 et ENSG00000099958

```
which(exprs$id %in% c("ENSG00000253991", "ENSG00000099958")
```

Afficher les lignes correspondantes

```
exprs[which(exprs$id %in% c("ENSG00000253991", "ENSG0000009
```

Calculs sur des colonnes

Calcul de moyennes par ligne (`rowMeans`) pour un sous-ensemble donné des colonnes (WT1 et WT2).

```
rowMeans(exprs[,c("WT1","WT2")])
```

Ajout de colonnes avec les expressions moyennes des WT et des KO.

```
exprs$meanWT <- rowMeans(exprs[,c("WT1","WT2")])
exprs$meanKO <- rowMeans(exprs[,c("KO1","KO2")])

head(exprs) ## Check the result
```

Fold-change KO vs WT

```
exprs$FC <- exprs$meanKO / exprs$meanWT
head(exprs) ## Check the result
```

MA-plot: log2FC vs intensité

M est le logarithme en base 2 du rapport d'expression.

$$M = log_2(\text{FC}) = log_2\left(\frac{\text{KO}}{\text{WT}}\right) = log_2(\text{KO}) - log_2(\text{WT})$$

```
exprs$M <- log2(exprs$FC)
```

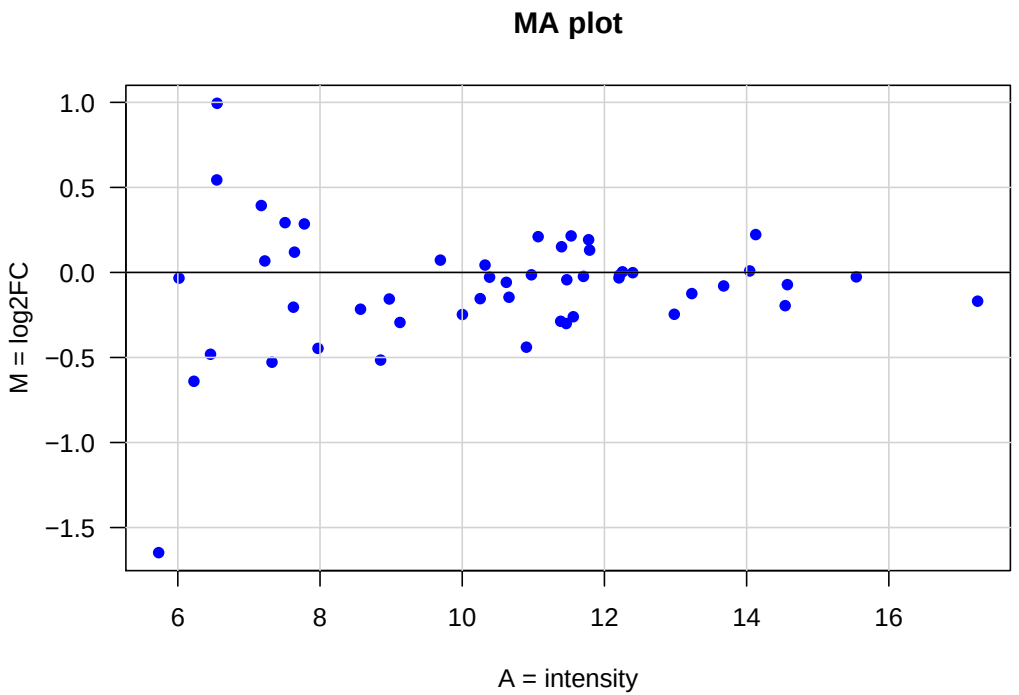
A (average intensity) est la moyenne des logarithmes des valeurs d'expression.

$$A = \frac{1}{2} log_2(\text{KO} \cdot \text{WT}) = \frac{1}{2}\left(log_2(\text{KO}) + log_2(\text{WT})\right)$$

```
exprs$A <- rowMeans(log2(exprs[,c("meanWT", "meanKO")]))
```

MA-plot: log2FC vs intensité

```
plot(x = exprs$A, y = exprs$M, main = "MA plot", las = 1,
     col = "blue", pch = 16, xlab = "A = intensity", ylab =
grid(lty = "solid", col = "lightgray")
abline(h = 0)
```



Charger les annotations des gènes

```
annot <- read.table(file = "annotation.csv", header = TRUE,
dim(annot)   ## Vérifier les dimensions
head(annot)  ## Afficher quelques lignes
```

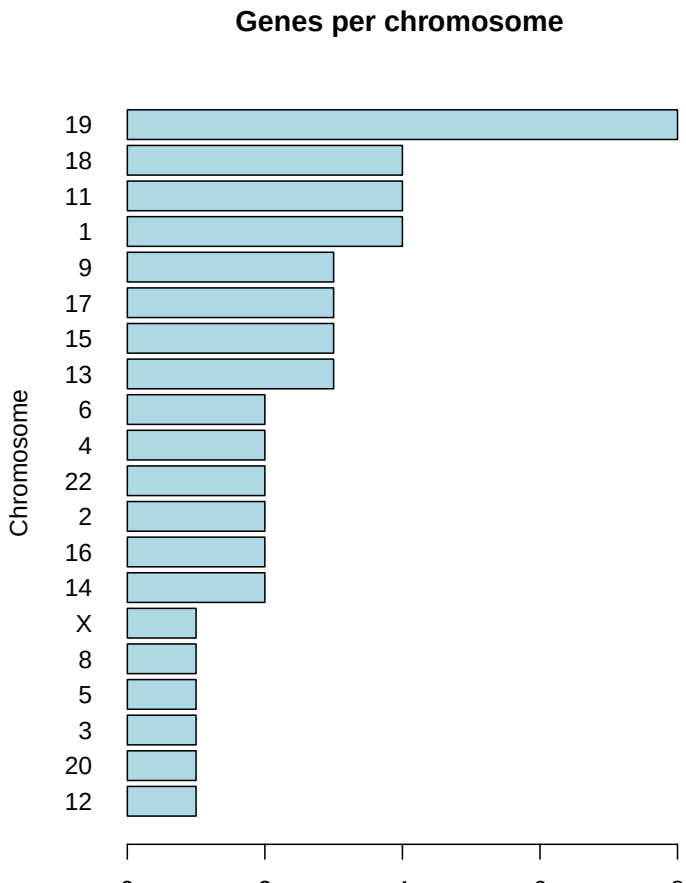
Combien de gènes par chromosome ?

```
table(annot$chr)
```

Question: combien de gènes sur le chromosome 8 ? Et sur le X ?

Diagramme en bâtons – gènes par chromosomes

```
barplot(sort(table(annot$chr)), horiz = TRUE, las = 1,
        main = "Genes per chromosome", ylab = "Chromosome",
        col = "lightblue", xlab = "Number of genes")
```



Sélectionner les données du chromosome 8

1ere étape: fusionner les deux tableaux exprs et annot

```
exprs.annot <- merge(exprs, annot, by = "id")
head(exprs.annot)
```

2eme étape: sous-ensemble des lignes pour lesquelles chr vaut 8

```
exprs8 <- exprs.annot[which(exprs.annot$chr == 8),]
print(exprs8)
```

Exporter exprs8 dans un fichier

```
write.table(x = exprs8, file = "exprs8.txt", sep = "\t",
            row.names = TRUE, col.names = NA)
```

A ajouter: ALLER PLUS LOIN (optionnel)

- charger un tableau complet de données (RNA-seq ou ChIP-seq et refaire quelques plots)
- hist(breaks = 100)
- apply
- sort, order
- lm() ?