TP2

Justine Guégan - ICONICS

7 mars 2017

Exercice 1 : KEGG pathways

Cet exercice a pour but de vous faire:

- **installer un package Bioconductor**
- **réutiliser un code R créé par une personne tierce**

1. Installer le package Pathview de Bioconductor (https://bioconductor.org/packages/release/bioc/html/pathview.html)

```
source('http://www.bioconductor.org/biocLite.R')
biocLite("pathview")
```

NB : Ne mettez pas à jour les packages si on vous le propose car cela peut être long

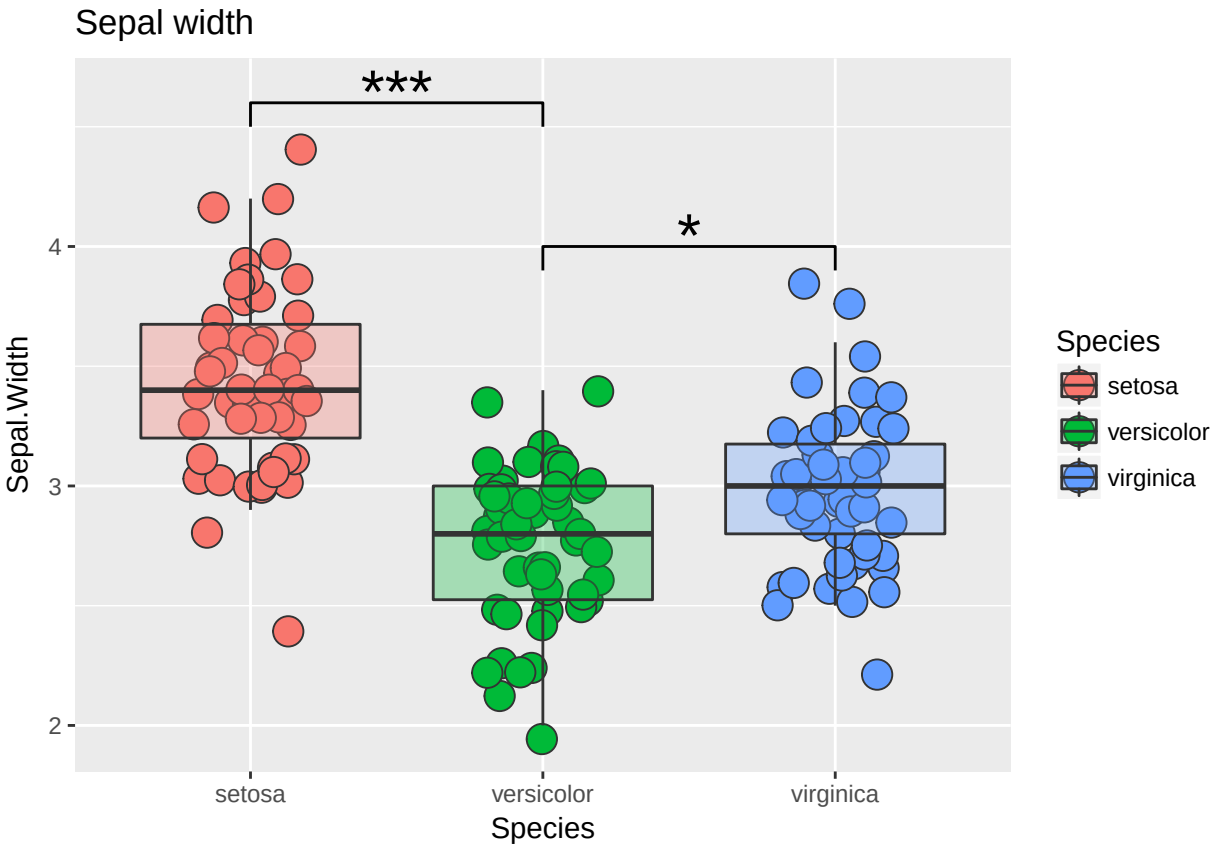
2. Exécuter le code R `pathview.R` et regarder le résultat.
3. Adapter ce code pour qu'il fonctionne sur le fichier d'entrée `myGenes.csv`, sur le pathway `hsa04012`. Pensez à :
   - Lire **correctement** le fichier `myGenes.csv`. Ajoutez l'argument `stringsAsFactor = FALSE` dans `read.table()`.
   - Imprimer le tableau lu, vérifier sa classe ...

Exercice 2 : From iris data to expression data

Cet exercice a pour but de vous faire:

- **"décrypter"" du code R**
- **installer des packages**
- **apréhender le package graphique `ggplot2`**
- **apréhender un test statistique simple**

Vous avez trouvé sur internet une figure que vous trouvez très intéressante et que vous aimeriez reproduire pour vos propres données.



Coup de chance ! vous trouvez avec le code R qui permet de générer cette figure ! `iris.R`

1. Essayez de comprendre ce que fait le code du script `iris.R`, notamment les lignes suivantes (n'y passez pas trop de temps non plus, on voit tout en détail par la suite) :

```
library(ggplot2)

res1 = t.test(iris$Sepal.Width[which(iris$Species == "setosa")], iris$Sepal.Width[which(iris$Species ==

pdf("sepalWidth.pdf")
print(p3)
dev.off()
```

2. Placez-vous à l'endroit où vous avez enregistré le script `iris.R` : Session –> Set working directory –> Choose directory.

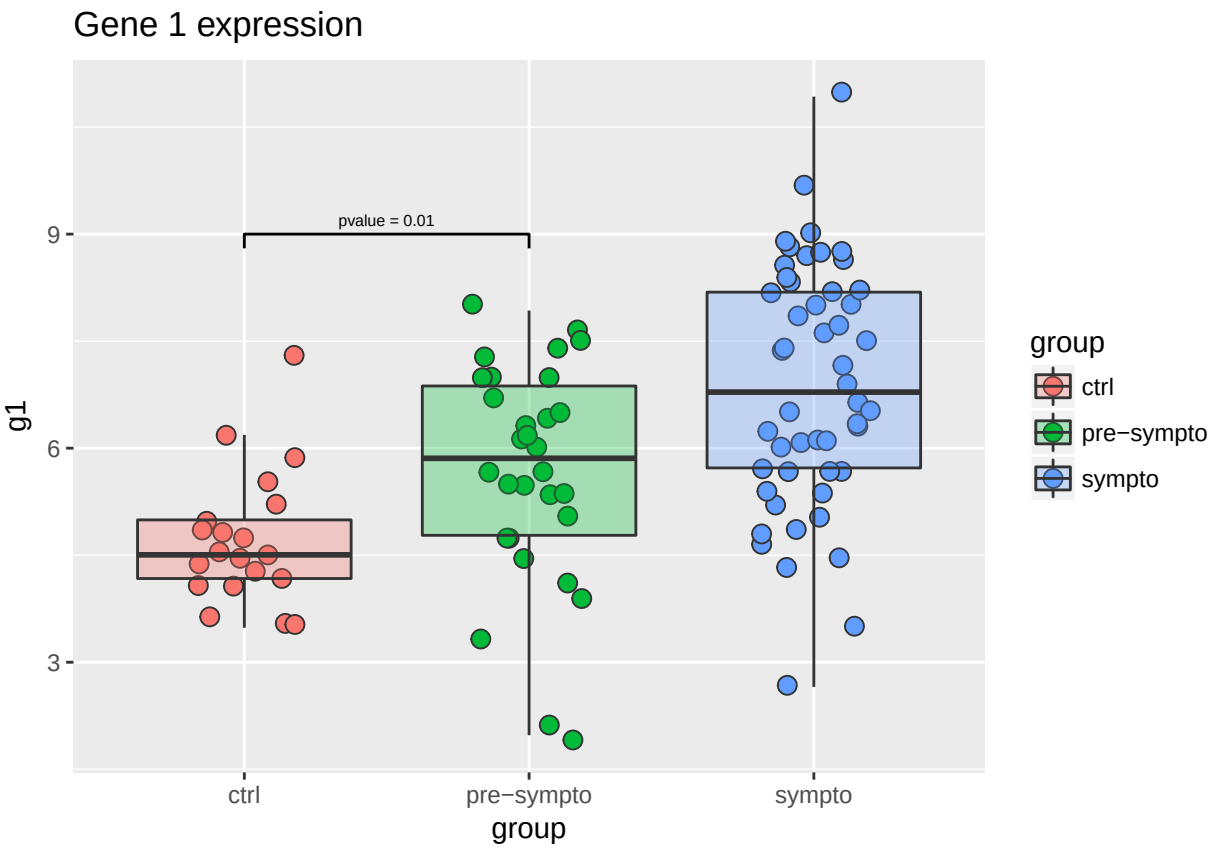
Exécutez à présent la ligne de commande suivante :

```
source(iris.R)
```

Que s'est-il passé ?

3. Vous trouvez la figure finale générée très intéressante et vous vous dites que vous pourriez vous en inspirer pour représenter vos données d'expression de gènes. Justement, vous avez screené votre gène fétiche appelé g1 chez 100 individus classés en 3 groupes : contrôles, pré-symptomatiques et symptomatiques. De plus, le t-test réalisé pourrait apporter une valeur statistique à vote étude.

L'idée est donc de produire la figure suivante, en indiquant la pvalue issue du t-test entre les contrôles et les pré-symptomatiques.



3.1 Lire le fichier `gene1.txt` (vous pouvez déjà le regarder sous excel ou notepad etc ... pour voir ce qu'il contient)

3.2 Comme pour le jeu de données `iris`, lancez quelques fonctions qui vous permettent d'avoir des infos sur le jeu de données (échantillonage, stats de base, dimensions du tableau ...)

3.3 Adaptez les lignes de commandes qui créent le premier graphique. Essayez de diminuer la taille des points, adaptez le titre ...

3.4 Lancez un t-test entre les contrôles et les pré-symptomatiques.

3.5 Adaptez les lignes de commandes qui créent le graphique final.

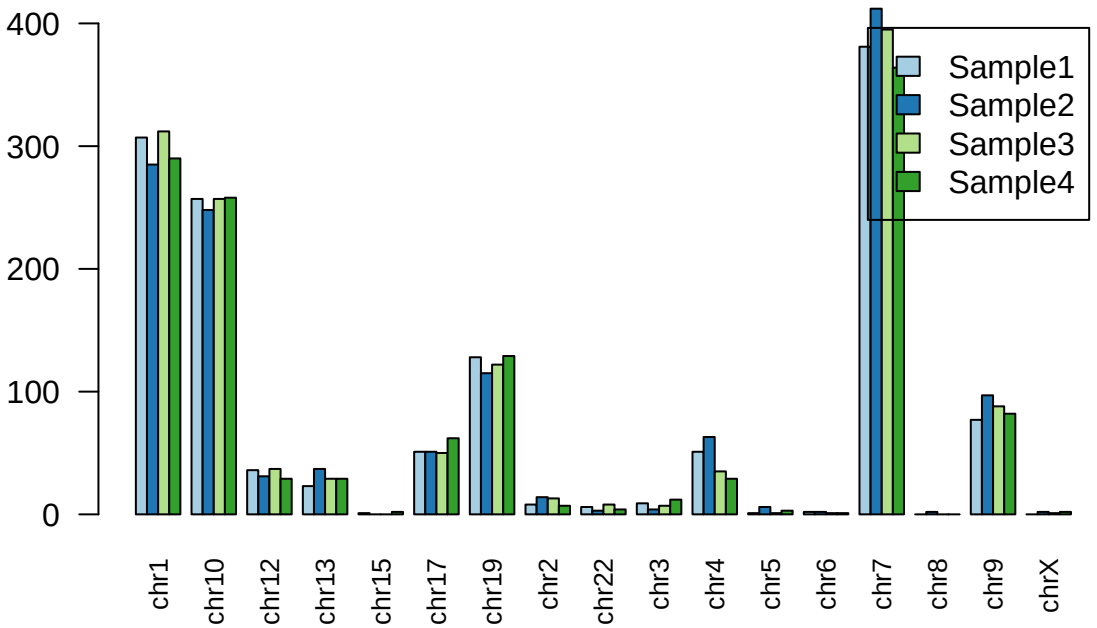
Exercice 3 : Manipulation de tableaux, lecture et écriture

Cet exercice a pour but de vous faire:

- **lire/écrire des fichiers excel**
- **filtrer des tableaux**
- **faire des représentations graphiques de base**

Le fichier `VariantsTable.xlsx` est un tableau contenant une liste de vraiations génomiques (snps/indels) issues d'une capture de gène faite par séquençage à haut débit, et ce, pour 4 patients : Sample1, Sample2, Sample3 et Sample4. Cette liste de variants est annotées avec différents base de données représentant les différentes colonne du fichier.

1. Chargez le package `gdata`
2. Lire le fichier VariantsTable.xlsx (ceci peut ête long)
3. Combien y-at-il de variants par échantillon ? Faites une représentation en bâtons (`barplot`) du nombre de variants par échantillon. Précisez un titre et colorer les barres en rouge et leur trait en rouge.
4. Représenter en barplot le nombre de variants par chromosome, par échantillon, de telle sorte que le graphique généré ressemble à celui ci-dessous :



4.1 Pour cela, vous pouvez combiner la fonction `with()` avec la fonction `table()`. Regarder l'aide de la fonction `table`, et isnpirez-vous des lignes d'exemple. L'idée est d'ovtenir un tableau de ce type

```
##          Chrom
## SampleID  chr1 chr10 chr12 chr13 chr15 chr17 chr19 chr2 chr22 chr3 chr4
##   Sample1  307   257    36    23     1    51   128    8     6    9   51
##   Sample2  285   248    31    37     0    51   115   14     3    4   63
##   Sample3  312   257    37    29     0    50   122   13     8    7   35
##   Sample4  290   258    29    29     2    62   129    7     4   12   29
##          Chrom
```

```
## SampleID  chr5 chr6 chr7 chr8 chr9 chrX
##   Sample1    1    2  381    0   77    0
##   Sample2    6    2  412    2   97    2
##   Sample3    1    1  395    0   88    1
##   Sample4    3    1  364    0   82    2
```

Exercice 4 : Intersecter des intervalles génomiques

Cet exercice a pour but de vous faire:

- **"décrypter"" du code R**
- **installer des packages**
- **découvrir le package GenomicRanges**
- **filtrer des matrices**
- **créer des graphiques de base**

Le package GenomicRanges permet de représenter et de manipuler des intervalles génomiques et des varaibles liées au génome. C'est un package Bioconductor, donc vous devez à présent savoir comment l'installer !

Dans cet exercice, on cherche à intersecter 2 fichiers contenant des positions génomiques : 1 fichier contenant une liste de transcrits, et 1 contenant une liste de snps. Inspirez-vous du script `findOverlaps.R` pour cet exercice.

1. Sourcez le script `findOverlaps.R`. Si des packages manquent, installez-les, soit via RStudio, soit via Bioconductor.
2. Essayer de comprendre les grandes lignes du script.
3. Créer un nouveau script pour répondre à cet exercice :

   3.1 Lire les fichiers `transcriptsTable.txt` et `snpsTable.txt`. Comme d'habitude, vérifier les dimensions, visualiser les données, les noms de colonnes etc ...

   3.2 En vous aidant du script `findOverlaps.R` et de l'aide du package GenomicRanges, créer 2 tableaux tab1 et tab2 contenant uniquement les snp dont la pvalue (*ie.* score) est inférieure à 0.01 et 0.005 respectivement ET qui overlappent un transcrit.