Assignment Week 1

Aphrodite Demetriou

2023-03-13

#install.packages('plyr', repos = "http://cran.us.r-project.org")  
#install.packages("r package", repos = "http://cran.us.r-project.org")  
  
#Task 1 - Literature  
  
#2a. What is the medically relevant insight from the article?  
  
#In this article the authors proposed an approach for detecting communities from a genetic network constructed on the basis of coexpression properties.   
#They developed a novel metric, the exclusively expressed index (EEI), that identifies mutually exclusive gene pairs from sparse scRNA-seq data.   
#Findings showed that EEI is effective for detecting mutually exclusive gene sets, while maintaining robustness against a low sequencing depth and improve the sensitivity of the classification of single cells.   
#The results suggest that exclusive expression can be introduced to identify intercell-type heterogeneity based on the feature matrix.  
#Authors applied their methods to glioblastoma scRNA-seq data and found that gene communities were partially conserved after serum stimulation despite a considerable number of differentially expressed genes  
#Glioblastoma is an heterogeneous tumor, and since their methods improve the sensitivity of the identification of cell-to-cell heterogeneity, this can serve as a tool in multiple other contexts in the single-cell analysis field, complementing existing approaches and providing new biological insights, even for large, sparse dataset.  
  
#2b. Which genomics technology/ technologies were used?  
  
#The authors used scRNA-seq technology.  
#To evaluate the performance of EEI and the community detection of coexpression networks they applied their method on six scRNA-seq datasets.  
# They generated glioblastoma scRNA-seq data from stem-like cells that were collected at 0 and 12 hours before and after the addition of serum  
# They used 5 more already published scRNA-seq data from: Human ES progenitor, Human ESC-derived neuron, Mouse cortex, PBMC CELseq2, PBMC MARSseq   
  
#They evaluated the effectiveness of EEI compared to four existing methods: the Pearson correlation coefficient, minet, GENIE3 and PIDC.   
#They evaluated the performance of EEI on the basis of the area under the precision-recall curve (AUPR) and average precision. To validate the performance for clustering of single cells, they adopted the Adjusted Rand Index (ARI) and the silhouette coefficient.   
#ARI measures the similarity between predicted and true cluster labels and ranges from 0.0 to 1.0. The silhouette coefficient measures the cluster cohesion and separation and ranges from –1.0 to 1.0.  
  
#3a. List and explain at least three questions/ hypotheses you can think of that extend the analysis presented in the paper.  
  
#1 The authors' novel metric EEI was applied to a newly generated glioblastoma scRNA-seq dataset. To evaluate the performance it would be very interesting to be applied in multiple highly heterogeneous samples, for example human cortex from Alzheimers' patients (different disease stages - different inflammation levels).  
  
#2 The PBMC CELseq2, PBMC MARSseq and human ES datasets were generated by CELseq2, MARSseq and Smartseq2 protocols, not by a droplet-based protocol, they contained sufficient read counts.   
#In most cases, clustering with EEI displayed distinct clusters of cell types in these datasets. This means that EEI is effective not only for sparse scRNA-seq data but also for data with a sufficient sequencing depth.   
#Thus, EEI enables to capture the exclusive expression between two genes displaying intercelltype heterogeneity. It would be interesting to observe if this can be applied to check intracell-type heterogeneity.  
  
#3 The fact that the authors used scRNA-seq data from samples with different number of cells as well as different sequencing depth shows the robustness of this study and that the developed metric that can be aplied across multiple samples. It is still interesting if this tool can be used in clinical samples derived from patients and not cultured glioblastoma stem-like cells (GSCs). That would exclude any culturing-enriching bias.  
  
  
#if (!require("BiocManager", quietly = TRUE))  
 # install.packages("BiocManager")  
#BiocManager::install(version = "3.16")  
  
  
#Task 4 - R basic operations  
  
#1  
sqrt(10)

## [1] 3.162278

#2  
log2(32)

## [1] 5

#3  
#sum(1:1000)  
  
#4  
x= c(2:1000)  
#x  
a=x[lapply(x, "%%", 2) == 0]  
#a  
#sum(a)  
  
#5  
#The formula for the number of independent pairwise comparisons is k(k-1)/2, where k is the number of conditions  
  
100\*(100-1)/2

## [1] 4950

#6  
?combn  
triples= combn(100,3)  
sum(triples)

## [1] 24497550

#Task 5  
  
#1  
data(CO2)  
#CO2  
  
#2  
help(CO2)  
#CO2  
#Carbon Dioxide Uptake in Grass Plants  
#The CO2 data frame has 84 rows and 5 columns of data from an experiment on the cold tolerance of the grass species Echinochloa crus-galli.  
  
#3  
  
mean(CO2$uptake[CO2$Type=="Quebec"])

## [1] 33.54286

median(CO2$uptake[CO2$Type=="Quebec"])

## [1] 37.15

mean(CO2$uptake[CO2$Type=="Mississippi"])

## [1] 20.88333

median(CO2$uptake[CO2$Type=="Mississippi"])

## [1] 19.3

#Task 6  
  
#1  
vector = c(1, 2, 2, 4, 6, 9)  
mean(vector)/median(vector)

## [1] 1.333333

#2  
  
a=vector[!vector %in% c(min(vector), max(vector))]  
mean(a)

## [1] 3.5

#3  
#R pipes (represented by the %>% operator) are used to chain multiple operations in sequence  
#Don't use pipes when:  
#Pipes are long (create intermediate objects so you can easily check intermediate results)  
#You have multiple inputs or outputs  
#You have a complex dependency structure, pipes will yield confusing code since they are fundamentally linear  
  
#4  
#Takes care of repetitive tasks  
#Can often replace loops and are claimed to be faster  
#apply() Applies a function to each row or column of a data.frame, matrix (or any dimension of a array).  
#lapply() for lists…output as list  
#sapply() for lists…output simplified  
#tapply() for vectors (groupwise to a variable)  
  
#Task 7  
  
#install.packages("remotes")  
library(remotes)  
#install\_url("http://emotion.utu.fi/wp-content/uploads/2019/11/nummenmaa\_1.0.tar.gz",dependencies=TRUE)  
  
setwd("/Users/aphroditedemetriou/Library/CloudStorage/OneDrive-KarolinskaInstitutet/Mac/Documents/PhD/10. KI PhD Courses/Bioinformatics - UT")  
  
mg=read.csv("/Users/aphroditedemetriou/Library/CloudStorage/OneDrive-KarolinskaInstitutet/Mac/Documents/PhD/10. KI PhD Courses/Bioinformatics - UT/magic\_guys.csv")  
#mg  
  
#1  
  
mg\_only\_sith=mg[!mg$species=="jedi", ]  
#mg\_only\_sith  
  
mg\_only\_jedi=mg[!mg$species=="sith", ]  
#mg\_only\_jedi  
  
#histograms  
  
png(file = "histograms1.png", width=1600, height=1167) #save it as png, it is used best for figures  
par(mfrow=c(1,2)) #creates the two graphs next to each other  
hist(mg\_only\_jedi$length, breaks=25, main="Jedi height")  
hist(mg\_only\_sith$length, breaks=25, main="Sith height")  
title('Distribution of body heights',outer=TRUE)  
dev.off() #closes the active device (the png file) and returns to the default mode of presenting graphics on-screen

## quartz\_off\_screen   
## 2

pdf(file = "histograms2.pdf") #save it as pdf, it is used best for documents  
par(mfrow=c(1,2)) #creates the two graphs next to each other  
hist(mg\_only\_jedi$length, breaks=25, main="Jedi height")  
hist(mg\_only\_sith$length, breaks=25, main="Sith height")  
dev.off() #closes the active device (the pdf file)

## quartz\_off\_screen   
## 2

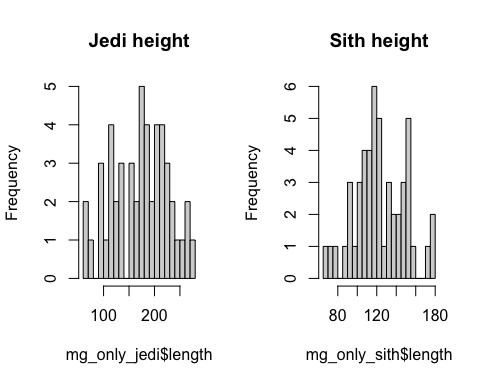
svg(file = "histograms3.svg") #save it as svg, svg files are used for web graphics like logos, illustrations, and charts

## Warning in grSoftVersion(): unable to load shared object '/Library/Frameworks/R.framework/Resources/modules//R\_X11.so':  
## dlopen(/Library/Frameworks/R.framework/Resources/modules//R\_X11.so, 0x0006): Library not loaded: /opt/X11/lib/libSM.6.dylib  
## Referenced from: <05451E21-B5F6-3B2F-9C0F-3EA08D57DC34> /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/modules/R\_X11.so  
## Reason: tried: '/opt/X11/lib/libSM.6.dylib' (no such file), '/System/Volumes/Preboot/Cryptexes/OS/opt/X11/lib/libSM.6.dylib' (no such file), '/opt/X11/lib/libSM.6.dylib' (no such file), '/Library/Frameworks/R.framework/Resources/lib/libSM.6.dylib' (no such file), '/Library/Java/JavaVirtualMachines/jdk-17.0.1+12/Contents/Home/lib/server/libSM.6.dylib' (no such file)

## Warning in cairoVersion(): unable to load shared object '/Library/Frameworks/R.framework/Resources/library/grDevices/libs//cairo.so':  
## dlopen(/Library/Frameworks/R.framework/Resources/library/grDevices/libs//cairo.so, 0x0006): Library not loaded: /opt/X11/lib/libXrender.1.dylib  
## Referenced from: <E7E5E590-1625-3923-977B-68091EA14521> /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/library/grDevices/libs/cairo.so  
## Reason: tried: '/opt/X11/lib/libXrender.1.dylib' (no such file), '/System/Volumes/Preboot/Cryptexes/OS/opt/X11/lib/libXrender.1.dylib' (no such file), '/opt/X11/lib/libXrender.1.dylib' (no such file), '/Library/Frameworks/R.framework/Resources/lib/libXrender.1.dylib' (no such file), '/Library/Java/JavaVirtualMachines/jdk-17.0.1+12/Contents/Home/lib/server/libXrender.1.dylib' (no such file)

## Warning in svg(file = "histograms3.svg"): failed to load cairo DLL

par(mfrow=c(1,2)) #creates the two graphs next to each other  
hist(mg\_only\_jedi$length, breaks=25, main="Jedi height")  
hist(mg\_only\_sith$length, breaks=25, main="Sith height")



dev.off() #closes the active device (the svg file)

## null device   
## 1

#install.packages("ggplot2")  
library(ggplot2)  
ggplot(mg, aes(x=length)) + geom\_histogram() #histogram using ggplot

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

#boxplots  
png(file = "boxplots.png", width=1600, height=1167) #creates a png file  
par(mfrow=c(1,2)) #creates the two graphs next to each other  
boxplot(mg\_only\_jedi$length,horizontal=TRUE, main="Jedi height")  
boxplot(mg\_only\_sith$length, horizontal=TRUE, main="Sith height")  
dev.off() #closes the active device (the png file)

## pdf   
## 2

ggplot(data = mg, mapping = aes(x = species, y = length)) +  
 geom\_boxplot() #creates the boxplot but with ggplot  
ggplot(data = mg, mapping = aes(x = species, y = length)) +  
 geom\_boxplot(alpha = 0) +  
 geom\_jitter(alpha = 0.2, color = "magenta") #example of different visualisation  
  
  
#2  
microarray=read.delim("/Users/aphroditedemetriou/Library/CloudStorage/OneDrive-KarolinskaInstitutet/Mac/Documents/PhD/10. KI PhD Courses/Bioinformatics - UT/microarray\_data.tab")  
#microarray  
#summary(microarray)  
  
#2a  
nrow(microarray) #553 rows

## [1] 553

ncol(microarray) #1000 columns

## [1] 1000

#2b  
sum(is.na(microarray)) #117826 missing values (NA)

## [1] 117826

#sapply(microarray, function(x) sum(length(which(is.na(x))))) #finds NA per column  
missing=sapply(microarray, function(x) sum(length(which(is.na(x)))))  
missing=as.matrix(missing)  
missing=cbind(rownames(missing), data.frame(missing, row.names=NULL)) #binds matrices together  
colnames(missing)=c("gene", "missing values")  
#missing  
  
hist(missing$"missing values", breaks=25, main="Missing values") # to visualize  
  
#2c  
percent\_missing\_ten=colMeans(is.na(missing))  
genes\_10\_percent\_missing=names(percent\_missing\_ten[percent\_missing\_ten > 0.1])  
genes\_10\_percent\_missing

## character(0)

percent\_missing\_twenty=colMeans(is.na(missing))  
genes\_20\_percent\_missing=names(percent\_missing\_twenty[percent\_missing\_twenty > 0.2])  
genes\_20\_percent\_missing

## character(0)

percent\_missing\_fifty=colMeans(is.na(missing))  
genes\_50\_percent\_missing=names(percent\_missing\_fifty[percent\_missing\_fifty > 0.5])  
genes\_50\_percent\_missing

## character(0)

#2d  
  
row\_means=rowMeans(microarray, na.rm=TRUE)  
#row\_means  
microarray\_data\_replaced=apply(missing, 2, function(x) {  
 x[is.na(x)] = row\_means  
 x  
})  
#microarray\_data\_replaced  
  
  
#3  
  
#CO2  
boxplot(uptake ~ Treatment, data=CO2, breaks=50, frame=F, col=c("blue", "green"),xlab="Plant", main="Uptake by treatment") # We observe the nonchilled plants having higher uptake of CO2 and greater variation on the chilled one   
boxplot(conc ~ Type, data=CO2, breaks=50, frame=F, col=c("blue", "green"),xlab="Plant", main="Conc by Area") # We observe that the Conc is not dependent by the area  
  
#Task 8  
#install.packages("tidybiology")  
#install.packages("devtools")  
#devtools::install\_github("hirscheylab/tidybiology")  
library(tidybiology)  
#install.packages("tidyverse")  
  
#a  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.0 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ lubridate 1.9.2 ✔ tibble 3.2.0  
## ✔ purrr 1.0.1 ✔ tidyr 1.3.0  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to force all conflicts to become errors

data(chromosome)  
#chromosome  
Summary\_statistics=summarise(chromosome,across(c(variations, protein\_codinggenes, mi\_rna),list(mean, median, max)))  
Summary\_statistics=as.data.frame(matrix(unlist(Summary\_statistics), ncol = 3, byrow = TRUE))  
colnames(Summary\_statistics)=c("Mean", "Median","Max")  
rownames(Summary\_statistics)=c("Variations", "Protein-coding genes","miRNA")  
Summary\_statistics

## Mean Median Max  
## Variations 6.484572e+06 6172346 12945965  
## Protein-coding genes 8.499583e+02 836 2058  
## miRNA 7.316667e+01 75 134

#8b  
library(ggplot2)  
  
Chromosomal\_size=ggplot(chromosome)+ aes(x =id , y = length\_mm, color = id, size=1) +   
 geom\_point(position=position\_jitter(w = 0.1,h = 5)) +  
 theme\_bw() +  
 ggtitle("Chromosomal size") +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1, size=12)) +  
 theme(axis.title.x = element\_text(size=10))+  
 theme(axis.title.y = element\_text(size=10))+  
 theme(plot.title = element\_text(hjust = 0.5, size=16))+  
 theme(legend.position = "none")  
Chromosomal\_size  
  
#8c  
Chromosomal\_num\_pcg=ggplot(chromosome)+ aes(x =protein\_codinggenes , y = length\_mm, color = id, size=1) +   
 geom\_point(position=position\_jitter(w = 0.1,h = 5)) +  
 theme\_bw() +  
 geom\_text(aes(label = id), hjust = 2, vjust = 0)+  
 ggtitle("Correlation chromosome size and protein-coding gene count") +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1, size=12)) +  
 theme(axis.title.x = element\_text(size=10))+  
 theme(axis.title.y = element\_text(size=10))+  
 theme(plot.title = element\_text(hjust = 0.5, size=16))+  
 theme(legend.position = "none") +  
 labs(x = "Number of protein-coding genes", y = "Length in mm")  
Chromosomal\_num\_pcg  
  
Chromosomal\_num\_miRNA=ggplot(chromosome)+ aes(x =mi\_rna , y = length\_mm, color = id, size=1) +   
 geom\_point(position=position\_jitter(w = 0.1,h = 5)) +  
 theme\_bw() +  
 geom\_text(aes(label = id), hjust = 2, vjust = 0)+  
 ggtitle("Correlation chromosome size and miRNA count") +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1, size=12)) +  
 theme(axis.title.x = element\_text(size=10))+  
 theme(axis.title.y = element\_text(size=10))+  
 theme(plot.title = element\_text(hjust = 0.5, size=16))+  
 theme(legend.position = "none") +  
 labs(x = "Number of miRNAs", y = "Length in mm")  
Chromosomal\_num\_miRNA  
  
#8d  
data(proteins)  
proteins

## # A tibble: 20,430 × 8  
## uniprot\_id gene\_name gene\_name\_alt protein\_…¹ prote…² seque…³ length mass  
## <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <dbl>  
## 1 P04217 A1BG <NA> "Alpha-1B… Alpha-… MSMLVV… 495 54254  
## 2 Q9NQ94 A1CF ACF ASP "APOBEC1 … APOBEC… MESNHK… 594 65202  
## 3 P01023 A2M CPAMD5 FWP007 "Alpha-2-… Alpha-… MGKNKL… 1474 163291  
## 4 A8K2U0 A2ML1 CPAMD9 "Alpha-2-… C3 and… MWAQLL… 1454 161107  
## 5 U3KPV4 A3GALT2 A3GALT2P IGBS3S "Alpha-1,… EC 2.4… MALKEG… 340 38754  
## 6 Q9NPC4 A4GALT A14GALT A4GALT1 "Lactosyl… EC 2.4… MSKPPD… 353 40499  
## 7 Q9UNA3 A4GNT <NA> "Alpha-1,… Alpha4… MRKELQ… 340 39497  
## 8 Q9NRG9 AAAS ADRACALA GL003 "Aladin " Adraca… MCSLGL… 546 59574  
## 9 Q86V21 AACS ACSF1 "Acetoace… EC 6.2… MSKEER… 672 75144  
## 10 P22760 AADAC DAC "Arylacet… EC 3.1… MGRKSL… 399 45734  
## # … with 20,420 more rows, and abbreviated variable names ¹​protein\_name,  
## # ²​protein\_name\_alt, ³​sequence

Protein\_mass\_length=ggplot(proteins)+ aes(x =mass , y = length, color = uniprot\_id, size=1) +   
 geom\_point(position=position\_jitter(w = 0.2,h = 5)) +  
 theme\_bw() +  
 geom\_text(aes(label = protein\_name), hjust = 1.4, vjust = 0)+  
 ggtitle("Correlation chromosome size and miRNA count") +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1, size=12)) +  
 theme(axis.title.x = element\_text(size=10))+  
 theme(axis.title.y = element\_text(size=10))+  
 theme(plot.title = element\_text(hjust = 0.5, size=16))+  
 theme(legend.position = "none") +  
 labs(x = "Protein mass", y = "Length")  
Protein\_mass\_length

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

## Including Plots

You can also embed plots, for example:

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.