Fecha de publicación del enunciado: 24/10/2024

Fecha límite para presentar la PEC: 6/11/20241

Descripción de la PEC El objetivo de esta PEC es que planifiquéis y ejecutéis una versión simplificada del proceso de análisis de datos ómicos, a la vez que practicáis con algunas de las herramientas y métodos que hemos trabajado.

En concreto lo que tendréis que hacer es:

1. **Seleccionar un dataset de metabolómica que podéis obtener de**
    a. **Este repositorio de github: https://github.com/nutrimetabolomics/metaboData/**
    b. **Si lo preferís podéis usar algún dataset del repositorio metabolomicsWorkbench**

Selecciono el dataset "2024-Cachexia" que se encuentra en el repositorio github, arriba mencionado.

2. **Una vez descargados los datos cread un contenedor del tipo *SummarizedExperiment* que contenga los datos y los metadatos (información acerca del dataset, las filas y las columnas). La clase *SummarizedExperiment* es una extensión de *ExpressionSet* y muchas aplicaciones o bases de datos (como metabolomicsWorkbench) lo utilizan en vez de usar expressionSet.**

Ejecuto el siguiente código para instalar *SummarizedExperiment* y otras dependencias:

```
> if (!requireNamespace("SummarizedExperiment", quietly = TRUE)) {
+     install.packages("BiocManager")
+     BiocManager::install("SummarizedExperiment")
+ }
```

Cargo el dataset en R a partir del archivo que he cargado en mi repositorio de github:

```
> dataset <- read.csv("https://raw.githubusercontent.com/aramonarr/RAMON-ARRUFAT-ALBE
RT-PEC1/main/human_cachexia.csv", row.names = 1)
```

Una vez cargado el dataset, creo un objeto *SummarizedExperiment* utilizando los datos del estudio y sus met adatos:

```
> library(SummarizedExperiment)
> colData <- DataFrame(SampleID = colnames(dataset), Condition = "CondiciónEjemplo")
> rowData <- DataFrame(FeatureID = rownames(dataset), Description =
"DescripciónEjemplo")
> se <- SummarizedExperiment(assays = list(counts = as.matrix(dataset)), rowData =
rowData, colData = colData)
```

```
> colData
DataFrame with 64 rows and 2 columns
             SampleID       Condition
          <character>     <character>
1          Muscle.loss CondiciónEjemplo
2  X1.6.Anhydro.beta.D... CondiciónEjemplo
3   X1.Methylnicotinamide CondiciónEjemplo
4       X2.Aminobutyrate CondiciónEjemplo
5   X2.Hydroxyisobutyrate CondiciónEjemplo
...               ...           ...
60         cis.Aconitate CondiciónEjemplo
61          myo.Inositol CondiciónEjemplo
62       trans.Aconitate CondiciónEjemplo
63     pi.Methylhistidine CondiciónEjemplo
64    tau.Methylhistidine CondiciónEjemplo


> rowData
DataFrame with 77 rows and 2 columns
     FeatureID       Description
    <character>      <character>
1       PIF_178 DescripciónEjemplo
2       PIF_087 DescripciónEjemplo
3       PIF_090 DescripciónEjemplo
4    NETL_005_V1 DescripciónEjemplo
5       PIF_115 DescripciónEjemplo
...         ...           ...
73  NETCR_019_V2 DescripciónEjemplo
74   NETL_012_V1 DescripciónEjemplo
75   NETL_012_V2 DescripciónEjemplo
76   NETL_003_V1 DescripciónEjemplo
77   NETL_003_V2 DescripciónEjemplo

> se
class: SummarizedExperiment
dim: 77 64
metadata(0):
assays(1): counts
rownames(77): PIF_178 PIF_087 ...
  NETL_003_V1 NETL_003_V2
rowData names(2): FeatureID Description
colnames(64): Muscle.loss
  X1.6.Anhydro.beta.D.glucose ...
  pi.Methylhistidine tau.Methylhistidine
colData names(2): SampleID Condition
```

3. **Llevad a cabo una exploración del dataset que os proporcione una visión general del mismo en la línea de lo que hemos visto en las actividades**

La estructura del dataset que he seleccionado muestra lo siguiente:

- **Número de muestras**: 77 (cada fila corresponde a una muestra con un ID único en la columna Patient ID).
- **Número de características/metabolitos**: 63 (variables cuantitativas), además de las columnas de identificación (Patient ID) y la condición (Muscle loss).
- **Condición**: La columna Muscle loss parece indicar el estado de cada muestra (por ejemplo, cachexic).

Realizo también un resumen estadístico del dataset estudiado:

Primero verifico que solo selecciono columnas numéricas para el resumen:

```
> numeric_data <- dataset[, sapply(dataset, is.numeric)]
> summary(numeric_data)
```

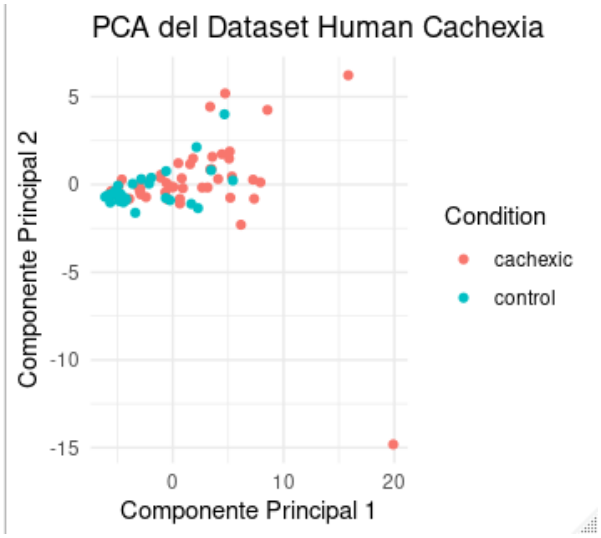
```
X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide
 Min.   :  4.71              Min.   :   6.42
 1st Qu.: 28.79              1st Qu.:  15.80
 Median : 45.60              Median :  36.60
 Mean   :105.63              Mean   :  71.57
 3rd Qu.:141.17              3rd Qu.:  73.70
 Max.   :685.40              Max.   :1032.77
 X2.Aminobutyrate X2.Hydroxyisobutyrate
 Min.   :  1.28   Min.   : 4.85
 1st Qu.:  5.26   1st Qu.:15.80
 Median : 10.49   Median :32.46
 Mean   : 18.16   Mean   :37.25
 3rd Qu.: 19.49   3rd Qu.:54.60
 Max.   :172.43   Max.   :93.69
 X2.Oxoglutarate   X3.Aminoisobutyrate
 Min.   :   5.53   Min.   :   2.61
 1st Qu.:  22.42   1st Qu.:  11.70
 Median :  55.15   Median :  22.65
 Mean   : 145.09   Mean   :  76.76
 3rd Qu.:  92.76   3rd Qu.:  56.26
 Max.   :2465.13   Max.   :1480.30
.....

.....
```

A continuación, realizamos el análisis de componentes más importantes y creamos un gráfico de dispersión para visualizar las muestras en los primeros dos componentes principales.

```
> pca <- prcomp(numeric_data, scale. = TRUE)
```

Y creamos un data frame con los resultados del PCA y graficamos:

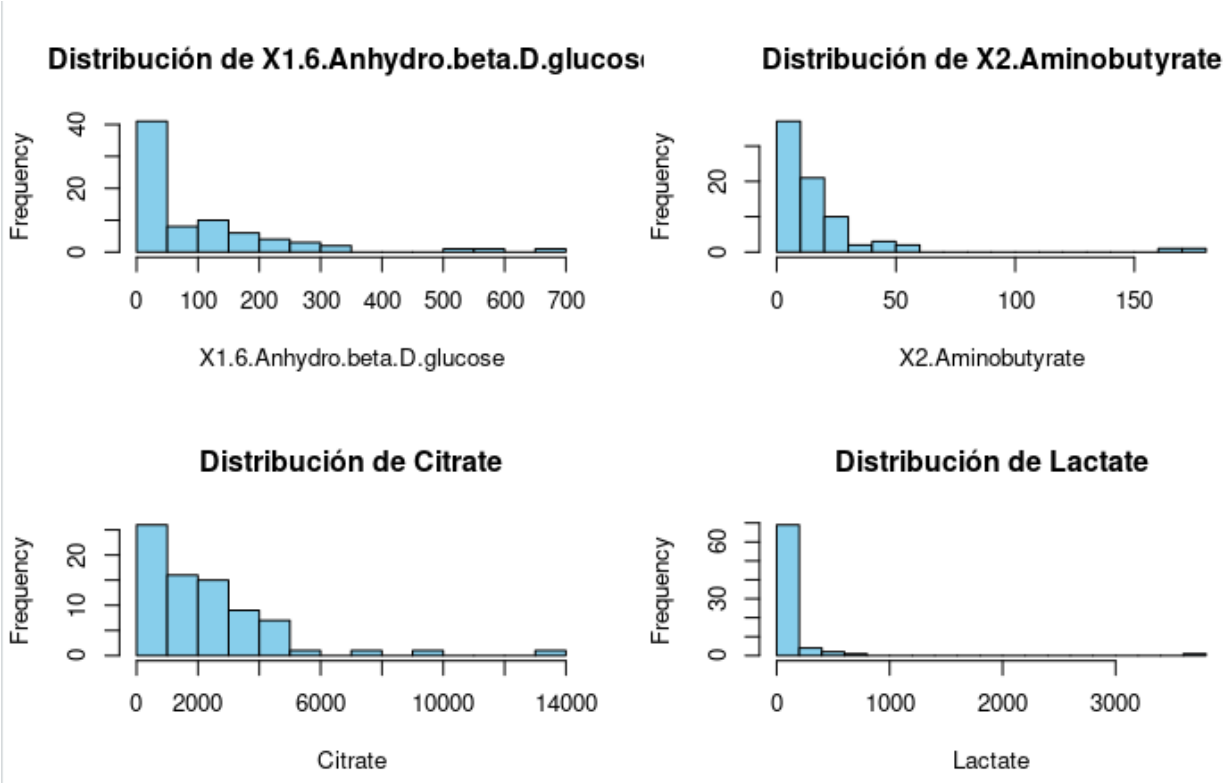
```
> pca_df <- data.frame(PC1 = pca$x[, 1], PC2 = pca$x[, 2], Condition = dataset$Muscle
.loss)
> install.packages("ggplot2")
> library(ggplot2)
 > ggplot(pca_df, aes(x = PC1, y = PC2, color = Condition)) +
 +     geom_point() +
 +     labs(title = "PCA del Dataset Human Cachexia", x = "Componente Principal
 1", y = "Componente Principal 2") +
 +     theme_minimal()
```



El gráfico muestra una separación entre las muestras de las condiciones cachexic y control en los componentes principales, lo cual sugiere variabilidad en los datos entre estas dos condiciones.

Realizamos también de las variables seleccionadas los histogramas correspondientes:

```
> selected_vars <- c("X1.6.Anhydro.beta.D.glucose", "X2.Aminobutyrate", "Citrate"
, "Lactate")
> par(mfrow = c(2, 2))
> for (var in selected_vars) {
+     hist(numeric_data[[var]], main = paste("Distribución de", var), xlab = var,
col = "skyblue", breaks = 15)
+ }
> par(mfrow = c(1, 1))
>
```



Los gráficos muestran claramente la dispersión de valores para los metabolitos X1.6-Anhydro-beta-D-glucose, X2-Aminobutyrate, Citrate, y Lactate, con una distribución sesgada hacia valores más bajos en la mayoría de los casos, lo que es común en datos de metabolómica.

4. **Elaborad un informe que describa el proceso que habéis realizado, incluyendo la descarga de los datos, la creación del contenedor, la exploración de los datos y la reposición de los datos en github. El nombre del repositorio tiene que ser el siguiente: APELLIDO1-Apellido2-Nombre-PEC1. Por ejemplo, en mi caso el repositorio se llamaría: "Sanchez-Pla-Alex-PEC1"**

Realizado

1. **Cread un repositorio de github que contenga: aramonarr/RAMON-ARRUFAT-ALBERT-PEC1: PEC1:**

    a. **el informe**
    b. **el objeto contenedor con los datos y los metadatos en formato binario (.Rda)**
    c. **el código R para la exploración de los datos o los datos en formato texto y**
    d. **los metadatos acerca del dataset en un archivo markdown.**

La dirección (url) del repositorio deberá estar incluida en la última sección del informe de forma clara.

1. **Repositorio:** [aramonarr/RAMON-ARRUFAT-ALBERT-PEC1: PEC1]**:**