Regany-Rosell-Paula-PEC1-Informe

2024-11-06

Carreguem les dades i les metadades i creem l'objecte SummarizedExperiment

```
# Carreguem les dades
dades <- read.csv("C:/Users/pregany/Desktop/MÀSTER/ADO/PAC_1/2018-MetabotypingPaper/DataValues_S013.csv", row.names = 1)
metadades <- read.csv("C:/Users/pregany/Desktop/MÀSTER/ADO/PAC_1/2018-MetabotypingPaper/DataInfo_S013.csv", row.names = 1)
```

Un cop carregades les dades procedim a encapsular-les dins un Summarized Experiment

```
SE <- SummarizedExperiment(
  assays = list(counts = as.matrix(dades)), # matriu amb les dades numèriques
  colData = metadades # metadades
)

SE
```

```
## class: SummarizedExperiment
## dim: 39 695
## metadata(0):
## assays(1): counts
## rownames(39): 1 2 ... 38 39
## rowData names(0):
## colnames(695): SUBJECTS SURGERY ... SM.C24.0_T5 SM.C24.1_T5
## colData names(3): VarName varTpe Description
```

```r
# Guardem el summarized experiment

save(SE, file = "dades_metabol_paper.Rda")

# Guardem exclusivament la matriu de dades en una variable per tal de treballar amb elles

assay_data <- assay(SE)

# Mirem l'estructura de les dades
str(assay_data)
```

```
##  chr [1:39, 1:695] " 1" " 2" " 3" " 4" " 5" " 6" " 7" " 8" " 9" "10" "11" ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:39] "1" "2" "3" "4" ...
##   ..$ : chr [1:695] "SUBJECTS" "SURGERY" "AGE" "GENDER" ...
```

```r
head(assay_data)
```

```
##   SUBJECTS SURGERY   AGE  GENDER Group MEDDM_T0 MEDCOL_T0 MEDINF_T0 MEDHTA_T0
## 1 " 1"     "by pass" "27" "F"    "1"   " 0"     " 0"      " 0"      " 1"
## 2 " 2"     "by pass" "19" "F"    "2"   " 0"     " 0"      " 0"      " 0"
## 3 " 3"     "by pass" "42" "F"    "1"   " 0"     " 0"      " 0"      " 0"
## 4 " 4"     "by pass" "37" "F"    "2"   " 0"     " 0"      " 0"      " 0"
## 5 " 5"     "tubular" "42" "F"    "1"   " 0"     " 0"      " 0"      " 0"
## 6 " 6"     "by pass" "24" "F"    "2"   " 0"     " 0"      " 0"
```

```
" 0"
[…]
## 4 "2.69"         "1.78"         "3.69"           "1.62"
## 5 "2.79"         "2.19"         "2.92"           "1.19"
## 6 "2.20"         "1.39"         "3.17"           "1.33"
##   SM..OH..C22.1_T5 SM..OH..C22.2_T5 SM..OH..C24.1_T5
SM.C16.0_T5 SM.C16.1_T5
## 1 " 2.44"          "3.93"           "0.24"           " 67.3"
"11.50"
## 2 " 2.60"          "3.76"           "0.27"           " 62.2"
"11.60"
## 3 " 2.77"          "3.70"           "0.40"           " 62.8"
" 8.85"
## 4 " 3.54"          "4.85"           "0.53"           " 76.6"
"12.60"
## 5 " 2.55"          "3.31"           "0.39"           " 74.0"
"12.60"
## 6 " 3.04"          "3.90"           "0.38"           " 67.0"
"11.80"
##   SM.C18.0_T5 SM.C18.1_T5 SM.C20.2_T5 SM.C24.0_T5 SM.C24.1_T5
## 1 "12.30"     " 8.17"     "0.19"      " 4.44"     "26.6"
## 2 " 9.90"     " 7.34"     "0.18"      " 4.75"     "24.7"
## 3 " 6.64"     " 4.33"     "0.17"      " 4.01"     "19.8"
## 4 " 9.51"     " 6.52"     "0.25"      " 5.90"     "32.1"
## 5 " 8.97"     " 6.62"     "0.16"      " 4.14"     "23.9"
## 6 " 9.82"     " 6.89"     "0.21"      " 5.62"     "28.6"

# Comprovem que no hi hagis NA a les dades
sum(is.na(assay_data))

## [1] 3390
```

```r
# Com que hi ha molts NAs anem a mirar com estan distribuits dins el dataset

na_count <- colSums(is.na(assay_data))
na_percentage <- (na_count / nrow(assay_data)) * 100

na_report <- data.frame(
  Column = colnames(assay_data),
  NA_Count = na_count,
  NA_Percentage = na_percentage
)

print(na_report)
```

Indiquem un llindar a partir del qual borrarem les columnes amb una alta presència de valors nuls.

```r
threshold <- 5
eliminar_cols <- na_report$Column[na_report$NA_Percentage > threshold]
print(eliminar_cols)
```

```
##   [1] "HBA1C_T0"          "HBA1C.mmol.mol_T0" "CC_T0"
##   [4] "CINT_T0"           "CAD_T0"            "TAD_T0"
##   [7] "TAS_T0"            "PCR_T0"            "LEP_T0"
##  [10] "ADIPO_T0"          "GGT_T0"            "URICO_T0"
##  [13] "TRANSF_T0"         "FERR_T0"           "MEDDM_T2"
##  [16] "MEDCOL_T2"         "MEDINF_T2"         "MEDHTA_T2"
##  [19] "GLU_T2"            "INS_T2"            "HOMA_T2"
##  [22] "HBA1C_T2"          "HBA1C.mmol.mol_T2" "PESO_T2"
##  [25] "bmi_T2"            "CC_T2"             "CINT_T2"
##  [28] "CAD_T2"            "TAD_T2"            "TAS_T2"
```

```
##  [31] "TG_T2"              "COL_T2"             "LDL_T2"
##  [34] "HDL_T2"             "VLDL_T2"            "PCR_T2"
##  [37] "LEP_T2"             "ADIPO_T2"           "GOT_T2"
##  [40] "GPT_T2"             "GGT_T2"             "URICO_T2"
##  [43] "CREAT_T2"           "UREA_T2"            "HIERRO_T2"
##  [46] "TRANSF_T2"          "FERR_T2"            "X"
##  [49] "MEDDM_T4"           "MEDCOL_T4"          "MEDINF_T4"
[…]
## [361] "PC.ae.C40.6_T5"    "PC.ae.C42.1_T5"    "PC.ae.C42.2_T5"
## [364] "PC.ae.C42.3_T5"    "PC.ae.C42.4_T5"    "PC.ae.C42.5_T5"
## [367] "PC.ae.C44.3_T5"    "PC.ae.C44.4_T5"    "PC.ae.C44.5_T5"
## [370] "PC.ae.C44.6_T5"    "SM..OH..C14.1_T5"
"SM..OH..C16.1_T5"
## [373] "SM..OH..C22.1_T5"  "SM..OH..C22.2_T5"
"SM..OH..C24.1_T5"
## [376] "SM.C16.0_T5"       "SM.C16.1_T5"       "SM.C18.0_T5"
## [379] "SM.C18.1_T5"       "SM.C20.2_T5"       "SM.C24.0_T5"
## [382] "SM.C24.1_T5"

# Com podem veure borrariem 382 del total de 695 variables
```

Utilitzant l'objecte Summarized Experiment, busquem la descripció de les variables per tal de saber com de relevants són per l'estudi i si les podem borrar.

```
# Accedim a les metadades de les columnes amb colData
metadades_info <- colData(SE)

# Filtrem la descripció i el tipus de les columnes a borrar
eliminar_cols_info <- metadades_info[eliminar_cols, c("varTpe",
"Description")]

print(eliminar_cols_info)
```

```
## DataFrame with 382 rows and 2 columns
##                         varTpe Description
##                    <character> <character>
## HBA1C_T0               numeric    dataDesc
## HBA1C.mmol.mol_T0      numeric    dataDesc
## CC_T0                  numeric    dataDesc
## CINT_T0                integer    dataDesc
## CAD_T0                 integer    dataDesc
## ...                        ...         ...
## SM.C18.0_T5            numeric    dataDesc
## SM.C18.1_T5            numeric    dataDesc
## SM.C20.2_T5            numeric    dataDesc
## SM.C24.0_T5            numeric    dataDesc
## SM.C24.1_T5            numeric    dataDesc
```

Malauradament el dataset de metadades no ens indica cap informació rellevant respecte la descripció de les columnes. Unicament podem saber que totes elles son numèriques excepte "X" que és lògica.

A continuació procedirem a una inspecció de les dades esborrant els NAs del dataset reduit sense columnes "perjudicials" i del dataset original.

```
# Primer filtrem les columnes perjudicials
assay_filtered <- assay_data[, !(colnames(assay_data) %in%
eliminar_cols)]


# Comprovem que hi ha diferència entre els dos datasets
dim(assay_data)

## [1]  39 695

dim(assay_filtered)

## [1]  39 313
```

Com podem veure s'han borrat correctament les 8 columnes "perilloses". Ara continuem amb la comparativa de dades restants després del borrat d'aquests NAs

```
# Eliminem les files amb NA del conjunt original
assay_data_dropped <- na.omit(assay_data)

# Eliminem files amb Na del conjunt de dades filtrat
assay_filtered_dropped <- na.omit(assay_filtered)
```

S'han realitzat diferents proves per veure quin era el llindar òptim de filtrat de dades per conservar la major quantitat de mostres perdent el mínim de columnes possible: - 50%: 3 variables, 8 columnes - 25%: 9 variables, 167 columnes - 10%: 28 variables, 347 columnes - 5%: 34 variables, 313 columnes

Podriem seguir baixant aquest llindar per obtenir més variables però ens arrisquem a quedar-nos sense columnes per fer l'estudi.

Cal mencionar que també es podrien aplicar mètodes de replenat de dades nul·les (com per exemple replenat amb el valor més freqüent), però per fer un anàlisi pur (sense augment de dades) s'ha decidit conservar el dataset cru en la mesura de lo possible.

URL repositori GitHub: https://github.com/paularegany/Regany-Rosell-Paula-PEC1