

# Deconvolución de datos de melanoma con quantiseqr: Datos bulk RNA-seq: GSE54467

Elena Eyre Sánchez, PhD

2024-10-27

## Contents

1	Introducción y Objetivo	1
2	Paquetes y datos	1

## 1 Introducción y Objetivo

## 2 Paquetes y datos

Repositorio GitHub de : [https://github.com/Danko-Lab/quantiseqr/blob/main/tutorial\\_deconvolution.pdf](https://github.com/Danko-Lab/quantiseqr/blob/main/tutorial_deconvolution.pdf)

```
knitr::opts_chunk$set(warning=FALSE)
#if (!requireNamespace("BiocManager", quietly = TRUE)) {
#  install.packages("BiocManager")
#}
#BiocManager::install("quantiseqr")
package_to_load <- c("readr", "dplyr", "ggplot2", "tidyr", "dplyr", "RColorBrewer",
                     "Biobase", "quantiseqr", "gplots")
for (package in package_to_load) {
  require(package, character.only = T); packageVersion(package)
}
extra_to_load <- c("knitr", "stringr", "stringi", "ggrepel", "ggpubr", "ggbreak",
                  "reshape2", "ggfortify", "cowplot", "GEOquery", "Seurat", "data.table",
                  "limma", "illuminaHumanv4.db", "SummarizedExperiment", "tibble")
for (package in extra_to_load) {
  require(package, character.only = T); packageVersion(package)
}
rm(package_to_load, extra_to_load)
```

#Datos

#Deconvolución

En este análisis utilizo los datos del estudio GSE54467 descargados mediante la función `getGEO` des de la base de datos GEO, del NCBI: <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE54467>.

Las muestras consisten en 65 muestras analizadas con la plataformas GPL570, y con varios tratamientos: dabrafenib + trametinib.

Este estudio es de especial interés para el TFM debido a que los autores también proporcionan metadata la respuesta de los pacientes, cosa que permitirá estudiar posibles correlaciones con las poblaciones obtenidas de

la deconvolución.

```
setwd("~/Desktop/ELENA_UOC/TFM")

gset <- getGEO("GSE54467", GSEMatrix = TRUE, getGPL = FALSE)
if (length(gset) > 1) idx <- grep("GPL6884", attr(gset, "names")) else idx <- 1
gset <- gset[[idx]]

# Exploración de las condiciones clínicas de los datos
#table(gset$characteristics_ch1) # Age at primary diagnosis
#table(gset$characteristics_ch1.1) # gender
#table(gset$characteristics_ch1.2) # Age at sample banked
#table(gset$characteristics_ch1.3) # Survival from stage iii tumor banked
#table(gset$characteristics_ch1.4) # Survival from primary melanoma
#table(gset$characteristics_ch1.5) # Patient last status (OS)
#table(gset$characteristics_ch1.6) # number of primary melanomas
#table(gset$characteristics_ch1.7) # stage at primary diagnosis

# Debido a que los autores proporcionan los genes con la nomenclatura de Illumina, lo convierto a símbolos
x <- illuminaHumanv4SYMBOL # cargado con el paquete illuminaHumanv4.db
mapped_probes <- mappedkeys(x) # Para sacar los símbolos
xx <- as.list(x[mapped_probes]) # Lo paso a listado
my_genes <- as.data.frame(unlist(xx[(rownames(gset@assayData$exprs))])) # Lo convierto en tabla para poder usarlos
my_genes$gene <- rownames(my_genes)

bulk_metadata <- as.data.frame(gset@phenoData@data) # Paso la metadata disponible a una tabla

#dim(gset@assayData$exprs) # 26085 genes      79 muestras
# Para usar los símbolos en lugar de nombres de illumina, extraigo los datos de expresión:
bulk.mtx <- as.data.frame(gset@assayData$exprs) # Los datos de expresión
bulk.mtx$gene <- rownames(bulk.mtx) # La columna que usaré para integrar
bulk.mtx <- inner_join(my_genes, bulk.mtx, by = "gene") # Integración de ambas tablas
bulk.mtx$gene <- NULL # Elimino la columna con nombres de Illumina
colnames(bulk.mtx)[1] <- "symbols" # Cambio la columna de símbolos de los genes

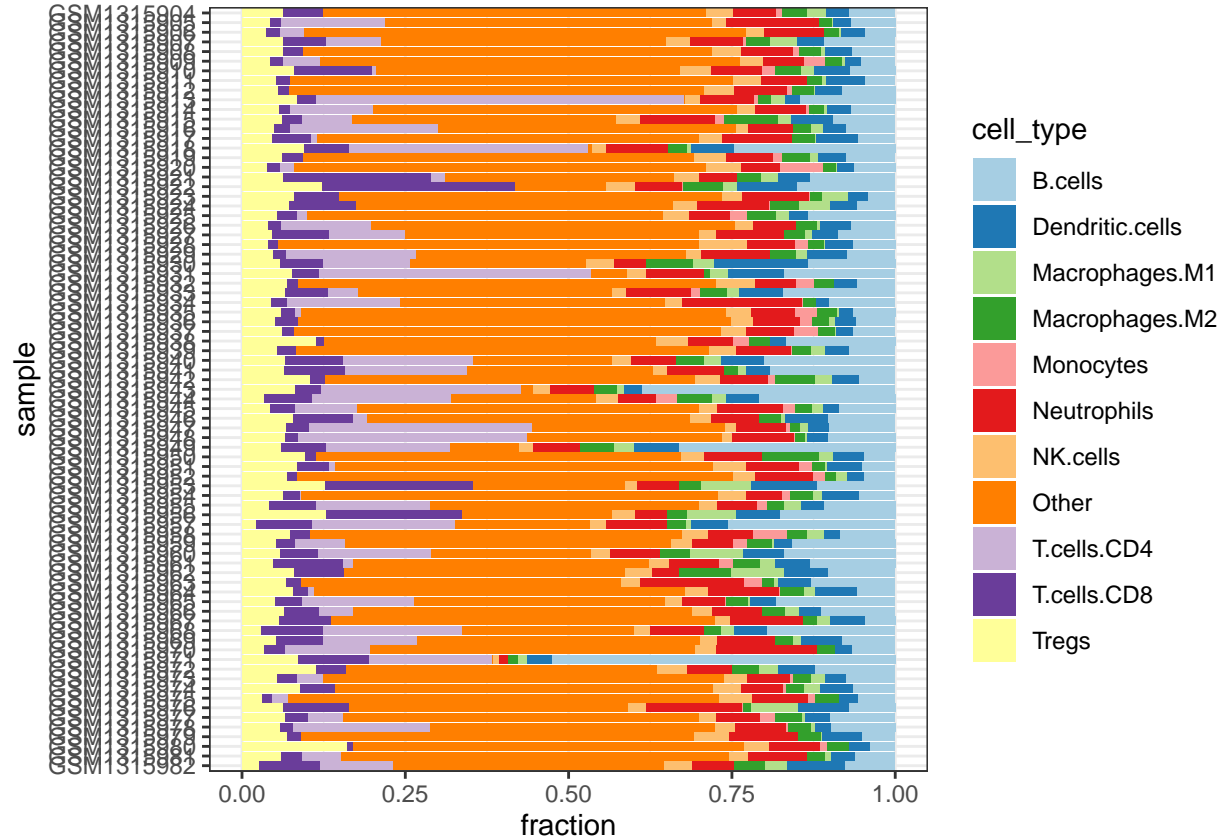
# Agrego los posibles duplicados calculando la media:
bulk.mtx <- aggregate(bulk.mtx, by = list(c(bulk.mtx$symbols)), mean) # Agregación
rownames(bulk.mtx) <- bulk.mtx$Group.1 # Los nombres de genes únicos sin duplicados sirven para dar nombre a las muestras
bulk.mtx <- bulk.mtx[, -c(1:2)] # Elimino las columnas usadas para conseguir los nombres

# Convertir los datos de expresión del bulk RNA-seq a objeto ExpressionSet:
bulk.eset <- Biobase::ExpressionSet(assayData = as.matrix(as.data.frame(bulk.mtx)))
bulk.eset

## ExpressionSet (storageMode: lockedEnvironment)
## assayData: 15354 features, 79 samples
## element names: exprs
## protocolData: none
## phenoData: none
## featureData: none
## experimentData: use 'experimentData(object)'
## Annotation:

ti_racle <- quantiseqr::run_quantiseq(
  expression_data = bulk.eset@assayData$exprs,
```

```
signature_matrix = "TIL10",
is_arraydata = FALSE,
is_tumordata = TRUE,
scale_mRNA = TRUE
)
quantipLOT(ti_racle)
```



Encontramos las proporciones del bulk RNA-seq en el apartado `ti_racle`, el cual puedo integrar en la metadata que ya tenía y almacenar en un archivo para posteriores análisis.

```
ref.based.estimates <- as.data.frame(ti_racle)
ref.based.estimates$geo_accession <- rownames(ref.based.estimates)
ref.based.estimates <- inner_join(ref.based.estimates, bulk_metadata, by = "geo_accession")
knitr::kable(head(ref.based.estimates[,1:7]), digits=2, caption = "Sección de las primeras muestras y c
```

Table 1: Sección de las primeras muestras y columnas como ejemplo del resultado

Sample	B.cells	Macrophages.M1	Macrophages.M2	Monocytes	Neutrophils	NK.cells
GSM1315904	0.07	0.03	0.04	0.01	0.07	0.04
GSM1315905	0.07	0.00	0.02	0.00	0.14	0.02
GSM1315906	0.05	0.00	0.02	0.00	0.09	0.03
GSM1315907	0.11	0.04	0.04	0.00	0.08	0.04
GSM1315908	0.07	0.01	0.03	0.01	0.08	0.05
GSM1315909	0.05	0.01	0.03	0.03	0.06	0.04

```
write.csv(ref.based.estimates, "./quantiseq_GSE54467.csv", row.names = FALSE)
```