Expresión diferencial (marginal)

Guillermo Ayala Gallego

2024-04-09

Table of contents

El problema

- ¿Hay diferencias de la expresión de genes entre dos (o más) grupos distintos?
- Dicho de otro modo, la expresión de un gen es distinta bajo distintos tratamientos.
- Se adopta la aproximación gen-a-gen, esto es, de momento solamente buscamos genes que se expresan diferencialmente sin atender al comportamiento conjunto que puedan tener.

Selección no específica o filtrado independiente

¿En qué consiste?

- La mayor parte de los genes no se expresan de un modo diferenciado entre condiciones.
- O ni se expresan en las distintas condiciones.
- Se puede realizar una preselección, un filtrado que no utilice información sobre las condiciones, utilizamos el perfil de expresión del gen pero no información sobre las muestras.
- A este filtrado se le denomina **filtrado independiente** o **selección no específica**.

Una posibilidad

- Consideramos el gen i-ésimo y tomamos una medida de localizacion como la mediana: u_i.
- Sea $q_{p_1}(u)$ sería el correspondiente percentil de orden p_1 de los valores u_i.
- Consideramos una medida de dispersión como el coeficiente intercuartílico: v_i.
- Sea $q_{p_2}(v)$ el correspondiente percentil de orden p_2 de los valores v_i.
- Nos quedamos con los genes tales que

$$\{i : i = 1, \dots, N; u_i \geq q_{p_1}(u); v_i \geq q_{p_2}(v)\},$$

- Lo lógico es usar bien **mediana y rango intercuartílico** o bien **media y desviación estándar**.
- Otra opción en lugar de localización: k-sobre-A: al menos k muestras tiene un nivel de expresión por encima de un valor mínimo A.

Datos ALL

- Los datos ALL son microarrays de 128 individuos distintos con leucemia linfoblástica aguda, ALL
- De estos individuos 95 corresponden a leucemia linfoblástica precursora aguda de células B y y 33 son leucemia linfoblástica precursora aguda de células T.
- Trabajaremos con las muestras de leucemia linfoblástica precursora aguda de células B.
- Los datos han sido preprocesados utilizando el método RMA.

Seleccionando algunas muestras en gse21942

- Es un `ExpressionSet`.

```
library(Biobase)
```

```
Loading required package: BiocGenerics
```

```
Attaching package: 'BiocGenerics'
```

```
The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs
```

```
The following objects are masked from 'package:base':

    anyDuplicated, aperm, append, as.data.frame, basename, cbind,
    colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
    get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
    match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
    Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
    table, tapply, union, unique, unsplit, which.max, which.min
```

```
Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.
```

- Leemos los datos.

```
data(gse21942,package="tamidata")
```

Atendiendo a la variabilidad

- Calculamos rango intercuartílico para cada gen.

```
gse21942.iqr = apply(exprs(gse21942),1,IQR)
```

- Calculamos un percentil (por ejemplo, mediana) de los rangos intercuartílicos.

```
sel.iqr = (gse21942.iqr > quantile(gse21942.iqr,0.5))
```

- Calculamos la mediana para cada gen.

```
gse21942.median = apply(exprs(gse21942),1,median)
```

- Y la mediana de las medianas.

```
sel.median = (gse21942.median > quantile(gse21942.median,0.5))
```

- ¿Y cómo nos quedamos nos estos genes?

```
gse219421 = gse21942[sel.iqr & sel.median,]
```

k sobre A

- Consideremos el siguiente criterio: si el nivel de expresión mínimo es `c` y tenemos n muestras podemos pedir que un gen determinado se considere activo si en al menos k muestras del total de n su nivel de expresión supere este nivel mínimo de actividad.

- ¿Qué c?

```
quantile(exprs(gse21942))
```

```
      0%       25%       50%       75%      100% 
1.994995  3.635526  5.035496  6.855258 15.011067 
```

```
c =  5.468801
```

- Determinamos qué niveles de expresión lo superan.

```
overc  = exprs(gse21942) > c
```

- Contamos el número de muestras que supera c.

```
count.c = apply(overc,1,sum)
```

- Y nos quedamos al menos 5 muestras superan c.

```
sel.c = count.c >= 5
```

- ¿Cuántos hay?

```
table(sel.c)
```

```
sel.c
FALSE  TRUE
28866 25809
```

kOverA

```
library(genefilter)
f1 = kOverA(5,5.468801)
ffun = filterfun(f1)
wh1 =  genefilter(exprs(gse21942), ffun)
```

nsFilter

- Supongamos que queremos aplicar la siguiente selección nos vamos a quedar con aquellas sondas tales que el rango intercuartílico (para todas las muestras) es mayor que la mediana de los rangos intercuartílicos.
- La mediana de la expresión es superior a la mediana de todas las medianas.
- Conocemos su anotación.

```
annotation(gse21942)
```

```
[1] "hgu133plus2"
```

- Cargamos la base de datos correspondiente.

```
library(hgu133plus2.db)
```

```
Loading required package: AnnotationDbi
```

```
Loading required package: stats4
```

```
Loading required package: IRanges
```

```
Loading required package: S4Vectors
```

```
Attaching package: 'S4Vectors'
```

```
The following object is masked from 'package:utils':
```

```
    findMatches
```

```
The following objects are masked from 'package:base':
```

```
    expand.grid, I, unname
```

```
Loading required package: org.Hs.eg.db
```

- Y filtramos.

```
gse21942.filt1 = nsFilter(gse21942,var.func=IQR,var.cutoff=0.5,
                         require.GOBP=TRUE)
gse21942_1 = gse21942.filt1$eset
```

- Tomamos la mediana por gen y nos quedamos con los genes cuya mediana sea mayor que la mediana de las medianas.

```
gse21942.filt2 = nsFilter(gse21942,var.func=median,var.cutoff=0.5,
  require.GOBP=TRUE)
gse21942_2 = gse21942.filt2$eset
```

- Consideramos simultáneamente las dos condiciones.

```
sel = intersect(featureNames(gse21942_1),featureNames(gse21942_2))
gse21942.sel = gse21942[sel,]
```

Log fold-change

El logaritmo del cociente de las medias

- Queremos comparar dos grupos.
- x_{ij} (y_{ij}) es la expresión de la i-ésima característica en la j-ésima muestra del primer grupo (segundo grupo).
- Tomamos los logaritmos (en base 2 o `log2`) de los valores originales:

$$u_{ij} = \log_2(x_{ij}),\ v_{ij} = \log_2(y_{ij}).$$

- Para cada gen, tendremos las expresiones medias para la i-ésima característica

$$\bar{x}_{i\cdot} = \sum_{j=1}^{n_1} \frac{x_{ij}}{n_1}$$

  $\bar{y}_{i\cdot}$, $\bar{u}_{i\cdot}$, $\bar{v}_{i\cdot}$.
- ¿Qué se entiende por **fold-change**?

- Dos son las interpretaciones **distintas** de este valor.

$$FC_i^{(1)} = \frac{\bar{x}_{i\cdot}}{\bar{y}_{i\cdot}}.$$

y

$$FC_i^{(2)} = \bar{u}_{i\cdot} - \bar{v}_{i\cdot},$$

- El log-fold change es el logaritmo en base 2 de los fold-change que acabamos de definir.
- Si el cociente anterior es mayor que `c` entonces diríamos que el gen se expresa de un modo diferencial en ambos grupos.
- Tenemos una expresión diferencial del gen i si

$$\left|\log_2\left(FC_i\right)\right| \geq c$$

Comparando dos grupos y un solo gen

Dos grupos

- Utilizamos los datos gse21942.

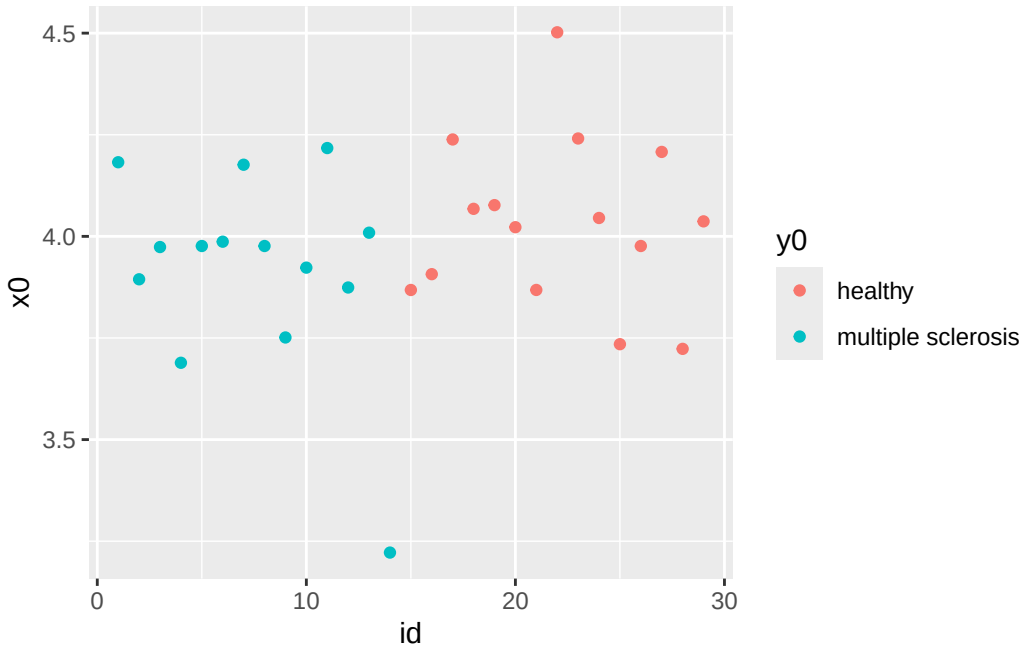
```
data(gse21942,package="tamidata")
```

- Nos fijamos en un gen.

```
x0 = exprs(gse21942)[2058,]
y0 = pData(gse21942)[,"FactorValue..DISEASE.STATE."]
```

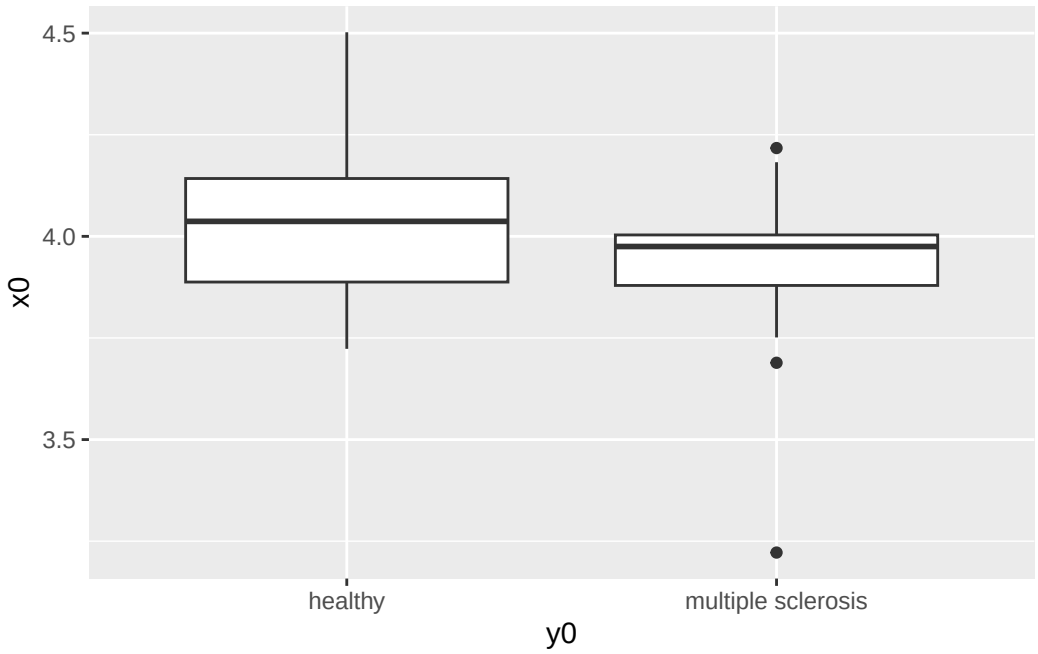
Perfil de expresión

```
pacman::p_load(ggplot2)
df = data.frame(id=1:length(x0),x0,y0)
ggplot(df,aes(x=id,y=x0)) + geom_point(aes(colour=y0))
```



Un diagrama de cajas

```
df = data.frame(id=1:length(x0),x0,y0)
ggplot(df,aes(x=y0,y=x0)) + geom_boxplot()
```



Comparamos dos condiciones

- Tenemos datos relativos a niveles de expresión bajo dos condiciones experimentales.
- Denotamos por X el nivel de expresión aleatorio que observamos bajo la primera condición y por Y lo mismo pero con la segunda condición.
- $X \sim N(\mu_X, \sigma_X^2)$
- $Y \sim N(\mu_Y, \sigma_Y^2)$
- Nos planteamos el contraste

$$H_0 : \mu_X = \mu_Y$$
$$H_1 : \mu_X \neq \mu_Y.$$

t-test con varianzas iguales

- Calculamos

$$T = \frac{\bar{X} - \bar{Y}}{S_p \sqrt{\frac{1}{n} + \frac{1}{m}}}$$

con

$$S_p = \hat{\sigma}^2 = \frac{(n-1)S_X^2 + (m-1)S_Y^2}{n+m-2}$$

siendo

$$S_X^2 = \sum_{i=1}^{n} \frac{(X_i - \bar{X})^2}{n-1}.$$

y S_Y^2 lo mismo sustituyendo X_i por Y_i.

- Rechazamos la hipótesis nula cuando

$$|T| > t_{\nu,1-\alpha/2}$$

- El p-valor viene dado por

$$p = P(|T| \geq t_0)$$

siendo t_0 el valor observado de T en cada caso.

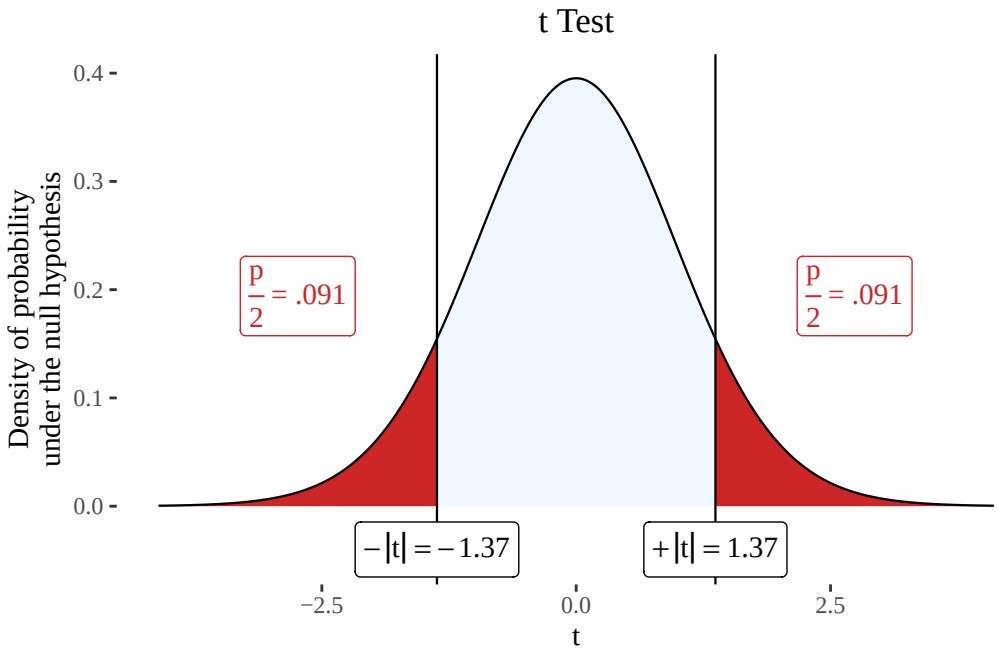
t.test

```
t.test(x0 ~ y0,var.equal=TRUE)
```

```
    Two Sample t-test

data:  x0 by y0
t = 1.3685, df = 27, p-value = 0.1824
alternative hypothesis: true difference in means between group healthy and group multiple sc
95 percent confidence interval:
 -0.0581131  0.2908712
sample estimates:
           mean in group healthy mean in group multiple sclerosis
                        4.034384                          3.918005
```

...



Comparación simultánea de todos los genes

Hacemos un t-test para todos los genes

```
library(genefilter)
tt = genefilter::rowttests(gse21942,pData(gse21942)[,"FactorValue..DISEASE.STATE."])
```

```
head(tt)
```

```
            statistic          dm     p.value
1007_s_at -2.29869931 -0.15981906 0.029492008
1053_at    3.44084102  0.20940361 0.001901206
117_at    -0.08505071 -0.01110444 0.932848609
121_at    -0.53362792 -0.02274832 0.597965094
1255_g_at -1.01536731 -0.04339509 0.318943716
1294_at   -1.05030996 -0.07873761 0.302886126
```

¿Cuándo tests mostrarían una expresión diferencial entre ambos grupos?

```r
table(tt$p.value < 0.01)
```

```
FALSE  TRUE
44014 10661
```

- Como vemos tenemos demasiados genes que muestran expresión diferencial.
- Y no parece muy razonable esto.
- El nivel de significación α es una cota superior del error tipo I cuando realizamos **un solo test**.
- Pero no estamos aplicando un solo test.
- Estamos rechazando muchas hipótesis nulas (no diferencia entre grupos para cada gen) cuando no deberíamos de hacerlo.

ANOVA: Comparando más de dos condiciones

- Supongamos que tenemos I condiciones distintas y en cada una de ellas n_i muestras de modo que $\sum_{i=1}^{I} n_i = n$.
- Y_i denota la expresión aleatoria de un gen bajo la condición i.

- Se asume que
$$Y_i \sim N(\mu_i, \sigma^2),$$
- **No existe expresión diferencial** es equivalente (bajo el modelo) a
$$H_0 : \mu_1 = ... = \mu_I,$$
$$H_1 : Existe\ i \neq j\ \mu_i \neq \mu_j.$$

- Supongamos que, para un gen dado, denotamos por y_{ij} la j-ésima muestra observada bajo la condición i ($i = 1, ..., I$ y $j = 1, ..., n_i$).
- Se consideran las **sumas de cuadrado intra**
$$SS(Intra) = \sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_{i\cdot})^2$$
y la **suma de cuadrados entre** como
$$SS(Entre) = \sum_{i=1}^{I} n_i (\bar{y}_{i\cdot} - \bar{y}_{\cdot\cdot})^2$$

- El estadístico para contrastar esta hipótesis nula es

$$F = \frac{SS(Entre)/(I-1)}{SS(Intra)/(n-I)}.$$

- Bajo la hipótesis nula de que todas las medias son la misma (y puesto que asumimos una misma varianza) tendríamos una distribución común bajo todas las condiciones.
- Asumiendo la hipótesis nula

$$F \sim F_{I-1,n-I}.$$

aov

- Leemos datos **gse20986** preprocesados con RMA.

```
data(gse20986,package="tamidata")
tejido = factor(c(1,2,2,1,2,1,3,3,3,4,4,4),levels = 1:4,
       labels=c("iris","retina","coroides","huvec"))
```

- Seleccionamos un gen cualquiera.

```
y = exprs(gse20986)[678,]
```

- Realizamos un análisis de la varianza.

```
summary(aov(y ~ tejido))
```

```
            Df   Sum Sq  Mean Sq F value Pr(>F)
tejido       3 0.005309 0.001770     0.7  0.578
Residuals    8 0.020212 0.002527
```

rowFtests

- Paquetes necesarios

```
library(multtest,genefilter())
```

- Calculamos los p-valores.

```
gse20986.aov = rowFtests(gse20986, tejido)
```

```
head(gse20986.aov,n=2)
```

```
          statistic     p.value
1007_s_at  9.782456 0.004715281
1053_at   14.144595 0.001455658
```