Regresiones lineales

Class	Bioinformática
Created	@Feb 25, 2021 9:08 AM
Materials	
Profesor	Leonardo Collado
Reviewed	☐
Type	Lecture

Tiene una distrimubión normal con media cero y varianza

12.1 Linear regression example

For example, if we have a linear regression

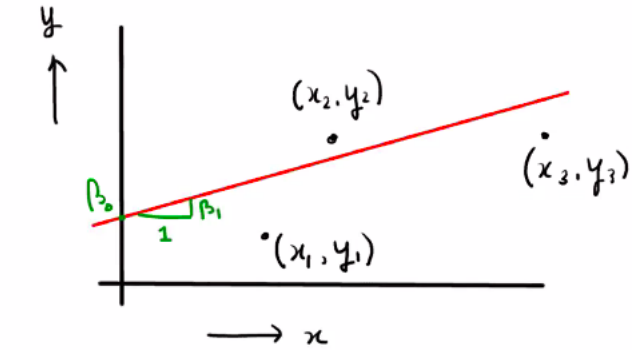
$$Y = \beta_0 + \beta_1 X + \epsilon$$

with

$$\epsilon \sim N(0, \sigma^2)$$

and we want to explain the meaning of the coefficient β_1 then we can explain it as:

- **Text**: β_1 is the average change in our outcome Y for a one unit increase in our explanatory variable X
  - The language here is quite precise and it can be hard to understand the implications of every word. This is the typical starting point for many questions: what does this mean?
- **Drawings**:



$\beta 1$ es la pendiente.

Dummy variables → funcionan como variables categóricas (para 2 categorías), e indician que categoría usando 1 o 0

Model.matrix se encarga de transformar los datos a 0 y 1

El signo "+" es una variable que se aumenta para evaluar con respecto a Y.

```
##
## Call:
## lm(formula = log(Volume) ~ log(Height) + log(Girth), data = trees)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.168561 -0.048488  0.002431  0.063637  0.129223
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.63162    0.79979  -8.292 5.06e-09 ***
## log(Height)  1.11712    0.20444   5.464 7.81e-06 ***
## log(Girth)   1.98265    0.07501  26.432  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08139 on 28 degrees of freedom
## Multiple R-squared:  0.9777, Adjusted R-squared:  0.9761
## F-statistic: 613.2 on 2 and 28 DF,  p-value: < 2.2e-16
```

Sí hay relación significativa entre la altura y volumen del árbol, ajustado a la circunferencia

Se lee en la columna **Pr(>|t|)**

ExploreModelMatrix

- Es un paquete de Bioconductor que nos ayuda a entender los modelos estadísticos que estamos usando gracias a visualizaciones http://www.bioconductor.org/packages/ExploreModelMatrix/ que está descrito en el siguiente artículo
- Revisaremos los ejemplos en http://www.bioconductor.org/packages/release/bioc/vignettes/ExploreModelMatrix/inst/doc/ExploreModelMatrix.htm

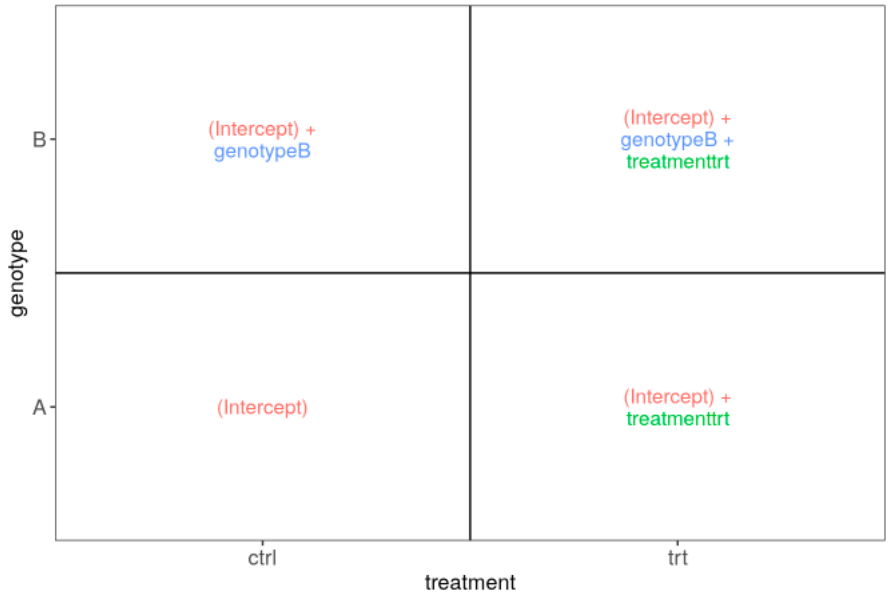
```
## Datos de ejemplo
(sampleData <- data.frame(
    genotype = rep(c("A", "B"), each = 4),
    treatment = rep(c("ctrl", "trt"), 4)
))
```

```
##   genotype treatment
## 1        A      ctrl
## 2        A       trt
## 3        A      ctrl
## 4        A       trt
## 5        B      ctrl
## 6        B       trt
## 7        B      ctrl
## 8        B       trt
```

```
## Creemos las imágenes usando ExploreModelMatrix
vd <- ExploreModelMatrix::VisualizeDesign(
    sampleData = sampleData,
    designFormula = ~ genotype + treatment,
    textSizeFitted = 4
)
```

```
## Veamos las imágenes
cowplot::plot_grid(plotlist = vd$plotlist)
```

en *designFormula = ~ genotype + treatment* no hay nada antes de la tilde porque eso cambia para cada gen.



B → 1

A → 0

De forma interactiva podemos correr el siguiente código:

```
## Usaremos shiny otra ves
app <- ExploreModelMatrix(
    sampleData = sampleData,
    designFormula = ~ genotype + treatment
)
if (interactive()) shiny::runApp(app)
```

Si restamos el cuadro inferior derecho al superior derecho, obtenemos el genotipo B. Lo mismo pasaría si lo hacemos del otro lado.

Creemos las imágenes usando ExploreModelMatrix

```
vd <- ExploreModelMatrix::VisualizeDesign(
sampleData = sampleData,
designFormula = ~ genotype + treatment,
textSizeFitted = 4
)
```

Veamos las imágenes

```
cowplot::plot_grid(plotlist = vd$plotlist)
```

Obtener la matriz

```
mod <- model.matrix(~ genotype + treatment, data = sampleData)
mod

      (Intercept) genotypeB treatmenttrt
1           1         0            0
2           1         0            1
3           1         0            0
4           1         0            1
5           1         1            0
6           1         1            1
7           1         1            0
8           1         1            1
attr(,"assign")[1] 0 1 2attr(,"contrasts")attr(,"contrasts")$genotype[1] "contr.treatment"attr(,"contrasts")$treatment[1] "contr.treatmen
t"
```

Se agrega un 0 después de la tilde cuando no queremos el valor del intercept

Normalización de datos

A scaling normalization method for differential expression analysis of RNA-seq data

The fine detail provided by sequencing-based transcriptome surveys suggests that RNA-seq is likely to become the platform of choice for interrogating steady state RNA. In order to discover biologically important changes in expression, we show that normalization continues to be an essential step in the

https://genomebiology.biomedcentral.com/articles/10.1186/gb-2010-11-3-r25#Sec2

Results and discussion

A hypothetical scenario

Estimated normalization factors should ensure that a gene with the same expression level in two samples is not detected as DE. To further highlight the need for more sophisticated normalization procedures in RNA-seq data, consider a simple thought experiment. Imagine we have a sequencing experiment comparing two RNA populations, A and B. In this hypothetical scenario, suppose every gene that is expressed in B is expressed in A with the same number of transcripts. However, assume that sample A also contains a set of genes equal in number and expression that are not expressed in B. Thus, sample A has twice as many total expressed genes as sample B, that is, its RNA production is twice the size of sample B. Suppose that each sample is then sequenced to the same depth. Without any additional adjustment, a gene expressed in both samples will have, on average, half the number of reads from sample A, since the reads are spread over twice as many genes. Therefore, the correct normalization would adjust sample A by a factor of 2.

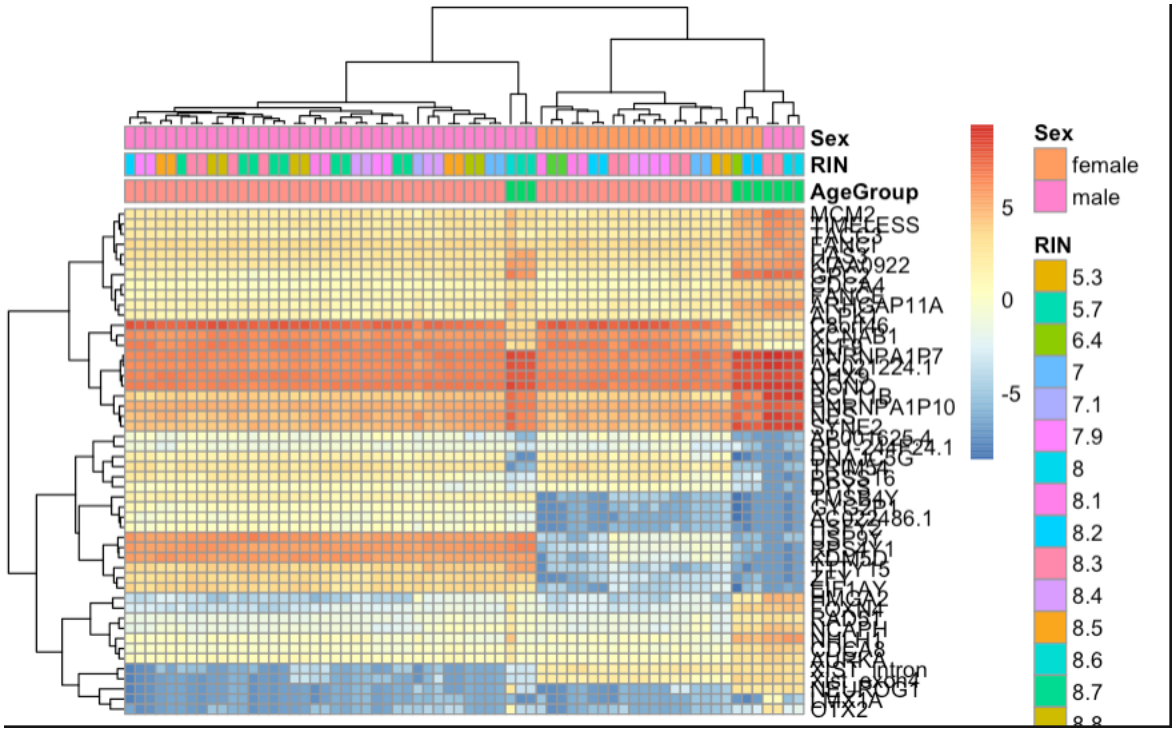
The hypothetical example above highlights the notion that the proportion of reads attributed to a given gene in a library depends on the expression properties of the whole sample rather than just the expression level of that gene. Obviously, the above example is artificial. However, there are biological and even technical situations where such a normalization is required. For example, if an RNA sample is contaminated, the reads that represent the contamination will take away reads from the true sample, thus dropping the number of reads of interest and offsetting the proportion for every gene. However, as we demonstrate, true biological differences in RNA composition between samples will be the main reason for normalization.

Método para normalizar ***composition bias.***

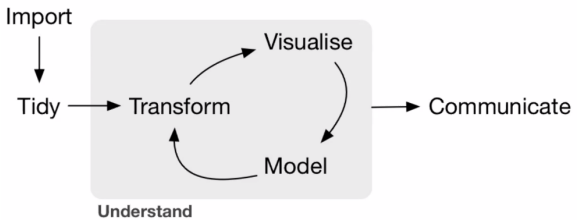
> 💡 volcanoplot y plotMA son gráficos básicos para modelos de expresión diferencial

Ejercicio

```
#Cambiar los rownames
rownames(exprs_heatmap)

#Obtener los índices de los id_gene en exprs_heatmap
our_match <- which(rowRanges(rse_gene_SRP045638)$gene_id %in% rownames(exprs_heatmap))

#Renombrar los rownames con los gene_name
rownames(exprs_heatmap) <- rowRanges(rse_gene_SRP045638)$gene_name[our_match ]

#Rehacer el heatmap
pheatmap(
  exprs_heatmap,
  cluster_rows = TRUE,
  cluster_cols = TRUE,
  show_rownames = TRUE,
  show_colnames = FALSE,
  annotation_col = df
)
```



Manejo de datos de RNA-seq



Normalización

Library size normalization

Toma en cuenta la suma de los niveles de expresión de todos los genes de todas las muestras y las compara. Normalizando de acuerdo a dicha comparación.

Librería `edgeR`

Tiene su propia clase de objetos, pero es fácil transformar un objeto `SummerizedExperiment` al objeto utilizado en `edgeR`

	Condition A	Condition B	A norm lib size	B norm lib size	A / B
Gene1	8	8	0.075471698	0.150943	0.5
Gene2	33	33	0.311320755	0.622642	0.5
Gene3	12	12	0.113207547	0.226415	0.5
Gene4	13	0	0.122641509	0	#DIV/0!
Gene5	6	0	0.056603774	0	#DIV/0!
Gene6	34	0	0.320754717	0	#DIV/0!
Total	106	53			