Weekly Assignment

2024-10-15

```r
library(tidyverse)
```

0 - import libraries

```
## -- Attaching core tidyverse packages ---------------------- tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(ggplot2)
```

```r
num_genes <- 20
```

1 - Assume the cells have 20 genes.

```r
treated5 <- rpois(num_genes, lambda = 5)
treated6 <- rpois(num_genes, lambda = 6)
treated7 <- rpois(num_genes, lambda = 7)
```

2 - Use a Poisson distribution with means = 5, 6, 7 to generate 3 gene expression datasets representing treated

```r
untreat1 <- rpois(num_genes, lambda = 1)
untreat2 <- rpois(num_genes, lambda = 2)
untreat3 <- rpois(num_genes, lambda = 1)
```

3 - Repeat the above with means = 1, 2, 1 to generate 3 gene expression datasets representing untreated.

```r
data <- data.frame(
  Gene = paste("Gene", 1:num_genes, sep=""),
  ca1 = treated5,
  ca2 = treated6,
  ca3 = treated7,
  co1 = untreat1,
  co2 = untreat2,
  co3 = untreat3
)
print(data)
```

4 - Generate a data frame whose rows are genes (20 total) and whose columns are samples (ca1, ca2, ca3 representing cases and co1, co2, co3, representing controls)

```
##        Gene ca1 ca2 ca3 co1 co2 co3
## 1    Gene1   5  10   7   1   3   0
## 2    Gene2   1   5   5   1   2   3
## 3    Gene3   9   7  10   2   4   0
## 4    Gene4   7  11   6   2   1   2
## 5    Gene5   5   8   3   0   2   0
## 6    Gene6   3   3   7   1   1   0
## 7    Gene7   7   5   3   1   2   1
## 8    Gene8   4   3   2   1   0   0
## 9    Gene9   8   3   7   0   1   2
## 10  Gene10   4   9   5   0   3   2
## 11  Gene11   3   7   8   2   1   0
## 12  Gene12  10   5   4   3   1   0
## 13  Gene13   4   5   5   0   0   1
## 14  Gene14   7   2   4   1   3   2
## 15  Gene15   6   4   7   2   1   1
## 16  Gene16   5   2   5   0   5   2
## 17  Gene17   5   2   9   2   3   0
## 18  Gene18   4   8   2   1   0   3
## 19  Gene19   5   3   7   2   2   1
## 20  Gene20   0   9   7   0   1   2
```

```r
gene_expression <- data %>% pivot_longer(
    cols = -Gene,
    names_to = "Sample",
    values_to = "Expression"
)
print(gene_expression)
```

5 - Turn the data farme into long format

```
## # A tibble: 120 x 3
##    Gene  Sample Expression
##    <chr> <chr>       <int>
```

```
##  1 Gene1 ca1              5
##  2 Gene1 ca2             10
##  3 Gene1 ca3              7
##  4 Gene1 co1              1
##  5 Gene1 co2              3
##  6 Gene1 co3              0
##  7 Gene2 ca1              1
##  8 Gene2 ca2              5
##  9 Gene2 ca3              5
## 10 Gene2 co1              1
## # i 110 more rows
```

```r
ggplot(
  gene_expression,
  aes(
    x = Sample,
    y = Expression,
    fill = Sample
  )
) + (
  geom_boxplot()
) + (
# 7 - try to color-code the box-plots by sample type
# (blue representing case and red representing control).
  scale_fill_manual(
    values = c(   #
      "ca1" = "blue", "ca2" = "blue", "ca3" = "blue",
      "co1" = "red", "co2" = "red", "co3" = "red"
    )
  )
)
```

6 - Use ggplot to plot the boxplots for each.