

Intro_to_R_for-biologists.R

RetailAdmin

2020-07-14

```
# Load library
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --

## v ggplot2 3.3.2      v purrr  0.3.4
## v tibble  3.0.2      v dplyr  1.0.0
## v tidyr   1.1.0      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.5.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

# Read in breast cancer RNA-seq datacount
counts = read_csv("GSE60450_GeneLevel_Normalized(CPM.and.TMM)_data.csv")

## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:
## cols(
##   X1 = col_character(),
##   gene_symbol = col_character(),
##   GSM1480291 = col_double(),
##   GSM1480292 = col_double(),
##   GSM1480293 = col_double(),
##   GSM1480294 = col_double(),
##   GSM1480295 = col_double(),
##   GSM1480296 = col_double(),
##   GSM1480297 = col_double(),
##   GSM1480298 = col_double(),
##   GSM1480299 = col_double(),
##   GSM1480300 = col_double(),
##   GSM1480301 = col_double(),
##   GSM1480302 = col_double()
## )

sampleInfo = read_csv("GSE60450_filtered_metadata.csv")

## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_character(),
##   characteristics = col_character(),
##   immunophenotype = col_character(),
##   'developmental stage' = col_character()
## )
```

```
# view what is stored in variables
sampleInfo
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?
```

```
## # A tibble: 12 x 4
##   X1      characteristics      immunophenotype      'developmental st~
##   <chr>    <chr>                  <chr>                <chr>
## 1 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ virgin
## 2 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ virgin
## 3 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ 18.5 day pregnancy
## 4 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ 18.5 day pregnancy
## 5 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ 2 day lactation
## 6 GSM1480~ mammary gland, luminal cells,~ luminal cell popu~ 2 day lactation
## 7 GSM1480~ mammary gland, basal cells, v~ basal cell popula~ virgin
## 8 GSM1480~ mammary gland, basal cells, v~ basal cell popula~ virgin
## 9 GSM1480~ mammary gland, basal cells, 1~ basal cell popula~ 18.5 day pregnancy
## 10 GSM1480~ mammary gland, basal cells, 1~ basal cell popula~ 18.5 day pregnancy
## 11 GSM1480~ mammary gland, basal cells, 2~ basal cell popula~ 2 day lactation
## 12 GSM1480~ mammary gland, basal cells, 2~ basal cell popula~ 2 day lactation
```

```
counts
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?
```

```
## # A tibble: 23,735 x 14
##   X1      gene_symbol GSM1480291 GSM1480292 GSM1480293 GSM1480294 GSM1480295
##   <chr> <chr>          <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 ENSM~ Gnai3          243.    256.    240.    217.    84.7
## 2 ENSM~ Pbsn           0         0         0         0         0
## 3 ENSM~ Cdc45        11.2    13.8    11.6     4.27    8.35
## 4 ENSM~ H19         6.31     8.53     7.09    11.0    0.194
```

```
## 5 ENSM~ Scml2          2.19      4.66      2.80      2.50      1.24
## 6 ENSM~ Apoh           0.224     0.0840      0        0        0
## 7 ENSM~ Narf           11.3      14.7      26.2      18.8      14.7
## 8 ENSM~ Cav2           118.      113.      50.5      63.4      186.
## 9 ENSM~ Klf6           2036.     2230.     1903.     1960.     1094.
## 10 ENSM~ Scmh1         33.7      38.7      9.18      9.43      3.92
## # ... with 23,725 more rows, and 7 more variables: GSM1480296 <dbl>,
## #   GSM1480297 <dbl>, GSM1480298 <dbl>, GSM1480299 <dbl>, GSM1480300 <dbl>,
## #   GSM1480301 <dbl>, GSM1480302 <dbl>
```

```
# dimension of variables-> rows by columns
dim(sampleInfo)
```

```
## [1] 12  4
```

```
dim(counts)
```

```
## [1] 23735    14
```

```
# view the first 6 lines by default or specify more lines through Arg
head(sampleInfo)
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?
```

```
## # A tibble: 6 x 4
##   X1      characteristics      immunophenotype  'developmental st~
##   <chr>   <chr>                  <chr>            <chr>
## 1 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 2 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 3 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 4 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 5 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
## 6 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
```

```
?head # check Arg n
```

```
## starting httpd help server ...
```

```
## done
```

```
head(sampleInfo, 8)
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?

## # A tibble: 8 x 4
##   X1      characteristics      immunophenotype  'developmental st~
##   <chr>    <chr>                  <chr>            <chr>
## 1 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 2 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 3 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 4 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 5 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
## 6 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
## 7 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
## 8 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
```

```
# view the last 6 lines
tail(sampleInfo)
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?

## # A tibble: 6 x 4
##   X1      characteristics      immunophenotype  'developmental st~
##   <chr>    <chr>                  <chr>            <chr>
## 1 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
## 2 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
## 3 GSM1480~ mammary gland, basal cells, 18~ basal cell popula~ 18.5 day pregnancy
## 4 GSM1480~ mammary gland, basal cells, 18~ basal cell popula~ 18.5 day pregnancy
## 5 GSM1480~ mammary gland, basal cells, 2 ~ basal cell popula~ 2 day lactation
## 6 GSM1480~ mammary gland, basal cells, 2 ~ basal cell popula~ 2 day lactation
```

```
# view the whole variable
View(sampleInfo)# or just click on the variable in the Environment pane
View(counts)

# view column vectors
sampleInfo$X1
```

```
## [1] "GSM1480291" "GSM1480292" "GSM1480293" "GSM1480294" "GSM1480295"
## [6] "GSM1480296" "GSM1480297" "GSM1480298" "GSM1480299" "GSM1480300"
## [11] "GSM1480301" "GSM1480302"
```

```
sampleInfo$immunophenotype
```

```
## [1] "luminal cell population" "luminal cell population"
## [3] "luminal cell population" "luminal cell population"
## [5] "luminal cell population" "luminal cell population"
## [7] "basal cell population"    "basal cell population"
## [9] "basal cell population"    "basal cell population"
## [11] "basal cell population"    "basal cell population"
```

```
# view values from a to b [a:b] in a column vector
sampleInfo$X1[1:3]
```

```
## [1] "GSM1480291" "GSM1480292" "GSM1480293"
```

```
sampleInfo$X1[2:4]
```

```
## [1] "GSM1480292" "GSM1480293" "GSM1480294"
```

```
sampleInfo$immunophenotype[1:3]
```

```
## [1] "luminal cell population" "luminal cell population"
## [3] "luminal cell population"
```

```
# view the structure of the data
str(sampleInfo)
```

```
## tibble [12 x 4] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ X1 : chr [1:12] "GSM1480291" "GSM1480292" "GSM1480293" "GSM1480294" ...
## $ characteristics : chr [1:12] "mammary gland, luminal cells, virgin" "mammary gland, luminal ce
## $ immunophenotype : chr [1:12] "luminal cell population" "luminal cell population" "luminal cell
## $ developmental stage: chr [1:12] "virgin" "virgin" "18.5 day pregnancy" "18.5 day pregnancy" ...
## - attr(*, "spec")=
## .. cols(
## .. X1 = col_character(),
## .. characteristics = col_character(),
## .. immunophenotype = col_character(),
## .. 'developmental stage' = col_character()
## .. )
```

```
str(counts)
```

```
## tibble [23,735 x 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ X1 : chr [1:23735] "ENSMUSG000000000001" "ENSMUSG000000000003" "ENSMUSG000000000028" "ENSMUS
## $ gene_symbol: chr [1:23735] "Gnai3" "Pbsn" "Cdc45" "H19" ...
## $ GSM1480291 : num [1:23735] 243.29 0 11.18 6.31 2.19 ...
## $ GSM1480292 : num [1:23735] 255.66 0 13.78 8.53 4.66 ...
## $ GSM1480293 : num [1:23735] 239.74 0 11.6 7.09 2.8 ...
## $ GSM1480294 : num [1:23735] 217.1 0 4.27 11.04 2.5 ...
## $ GSM1480295 : num [1:23735] 84.744 0 8.35 0.194 1.243 ...
```

```
## $ GSM1480296 : num [1:23735] 84.599 0 8.199 0 0.855 ...
## $ GSM1480297 : num [1:23735] 175.04 0 12.11 2.12 5.79 ...
## $ GSM1480298 : num [1:23735] 187.49 0 11.1 1.19 8.8 ...
## $ GSM1480299 : num [1:23735] 176.66 0 7.53 1.55 9.81 ...
## $ GSM1480300 : num [1:23735] 169.094 0 7.099 0.867 7.47 ...
## $ GSM1480301 : num [1:23735] 158.45 0 1.98 10.83 7.57 ...
## $ GSM1480302 : num [1:23735] 133.59 0 2.88 5.77 9.88 ...
## - attr(*, "spec")=
## .. cols(
## ..   X1 = col_character(),
## ..   gene_symbol = col_character(),
## ..   GSM1480291 = col_double(),
## ..   GSM1480292 = col_double(),
## ..   GSM1480293 = col_double(),
## ..   GSM1480294 = col_double(),
## ..   GSM1480295 = col_double(),
## ..   GSM1480296 = col_double(),
## ..   GSM1480297 = col_double(),
## ..   GSM1480298 = col_double(),
## ..   GSM1480299 = col_double(),
## ..   GSM1480300 = col_double(),
## ..   GSM1480301 = col_double(),
## ..   GSM1480302 = col_double()
## .. )
```

summary of data: length of string vectors refers to num of coordinates
whereas for numerical vectors: min, max, 1st quartile, 2nd quartile(median), 3rd quartile
summary(counts)

	X1	gene_symbol	GSM1480291	GSM1480292
##	Length:23735	Length:23735	Min. : 0.000	Min. : 0.000
##	Class :character	Class :character	1st Qu.: 0.000	1st Qu.: 0.000
##	Mode :character	Mode :character	Median : 1.745	Median : 1.891
##			Mean : 42.132	Mean : 42.132
##			3rd Qu.: 29.840	3rd Qu.: 29.604
##			Max. :12525.066	Max. :12416.211
##	GSM1480293	GSM1480294	GSM1480295	GSM1480296
##	Min. : 0.00	Min. : 0.00	Min. : 0.00	Min. : 0.00
##	1st Qu.: 0.00	1st Qu.: 0.00	1st Qu.: 0.00	1st Qu.: 0.00
##	Median : 0.92	Median : 0.89	Median : 0.58	Median : 0.54
##	Mean : 42.13	Mean : 42.13	Mean : 42.13	Mean : 42.13
##	3rd Qu.: 21.91	3rd Qu.: 19.92	3rd Qu.: 12.27	3rd Qu.: 12.28
##	Max. :49191.15	Max. :55692.09	Max. :111850.87	Max. :108726.08
##	GSM1480297	GSM1480298	GSM1480299	
##	Min. : 0.000	Min. : 0.000	Min. : 0.000	
##	1st Qu.: 0.000	1st Qu.: 0.000	1st Qu.: 0.000	
##	Median : 2.158	Median : 2.254	Median : 1.854	
##	Mean : 42.132	Mean : 42.132	Mean : 42.132	
##	3rd Qu.: 27.414	3rd Qu.: 26.450	3rd Qu.: 24.860	
##	Max. :10489.311	Max. :10662.486	Max. :15194.048	
##	GSM1480300	GSM1480301	GSM1480302	
##	Min. : 0.000	Min. : 0.000	Min. : 0.000	
##	1st Qu.: 0.000	1st Qu.: 0.000	1st Qu.: 0.000	
##	Median : 1.816	Median : 1.629	Median : 1.749	

```
## Mean : 42.132 Mean : 42.132 Mean : 42.132
## 3rd Qu.: 23.443 3rd Qu.: 23.443 3rd Qu.: 24.818
## Max. :17434.935 Max. :19152.728 Max. :15997.193
```

```
summary(sampleInfo)
```

```
##      X1      characteristics immunophenotype developmental stage
## Length:12      Length:12      Length:12      Length:12
## Class :character Class :character Class :character Class :character
## Mode :character Mode :character Mode :character Mode :character
```

```
# Exercises 1-4
```

```
# 1.
```

```
?head
```

```
head(sampleInfo, n = 8)
```

```
## Warning: '...' is not empty.
```

```
##
```

```
## We detected these problematic arguments:
```

```
## * 'needs_dots'
```

```
##
```

```
## These dots only exist to allow future extensions and should be empty.
```

```
## Did you misspecify an argument?
```

```
## # A tibble: 8 x 4
```

```
##   X1      characteristics immunophenotype 'developmental st~
##   <chr> <chr>                <chr>      <chr>
## 1 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 2 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ virgin
## 3 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 4 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 18.5 day pregnancy
## 5 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
## 6 GSM1480~ mammary gland, luminal cells, ~ luminal cell popu~ 2 day lactation
## 7 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
## 8 GSM1480~ mammary gland, basal cells, vi~ basal cell popula~ virgin
```

```
# 2.
```

```
subsetCounts = head(counts, n = 20)
```

```
# 3.
```

```
subsetCounts$GSM1480291
```

```
## [1] 243.28596 0.00000 11.18453 6.30808 2.19217 0.22369
## [7] 11.27401 118.24288 2036.16657 33.68781 126.92208 0.67107
## [13] 0.04474 0.00000 0.26843 0.00000 0.67107 17.31366
## [19] 73.54949 75.74166
```

```
# 4.
```

```
mean(subsetCounts$GSM1480291)
```

```
## [1] 137.8874
```

```

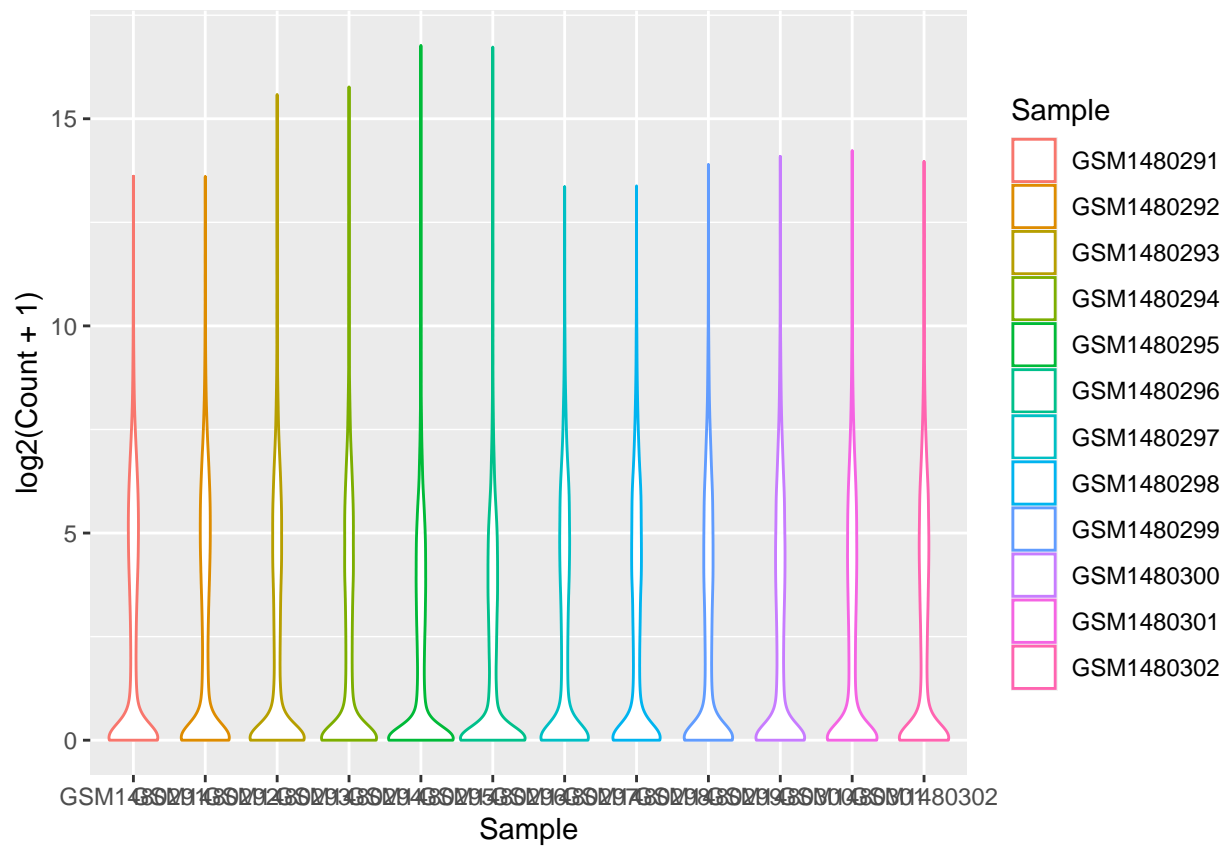
# subsetting

# Formatting the data
seqData = pivot_longer(counts, col = starts_with('GSM'), names_to = 'Sample', values_to = 'Count')
seqData = pivot_longer(counts, col = GSM1480291:GSM1480302, names_to = 'Sample', values_to = 'Count')
seqData = pivot_longer(counts, col = -c('X1', 'gene_symbol'), names_to = 'Sample', values_to = 'Count')

allInfo = full_join(seqData, sampleInfo, by = c('Sample' = 'X1'))

# Plot data
ggplot(allInfo, mapping = aes(x = Sample, y = log2(Count + 1), colour = Sample)) + geom_violin()

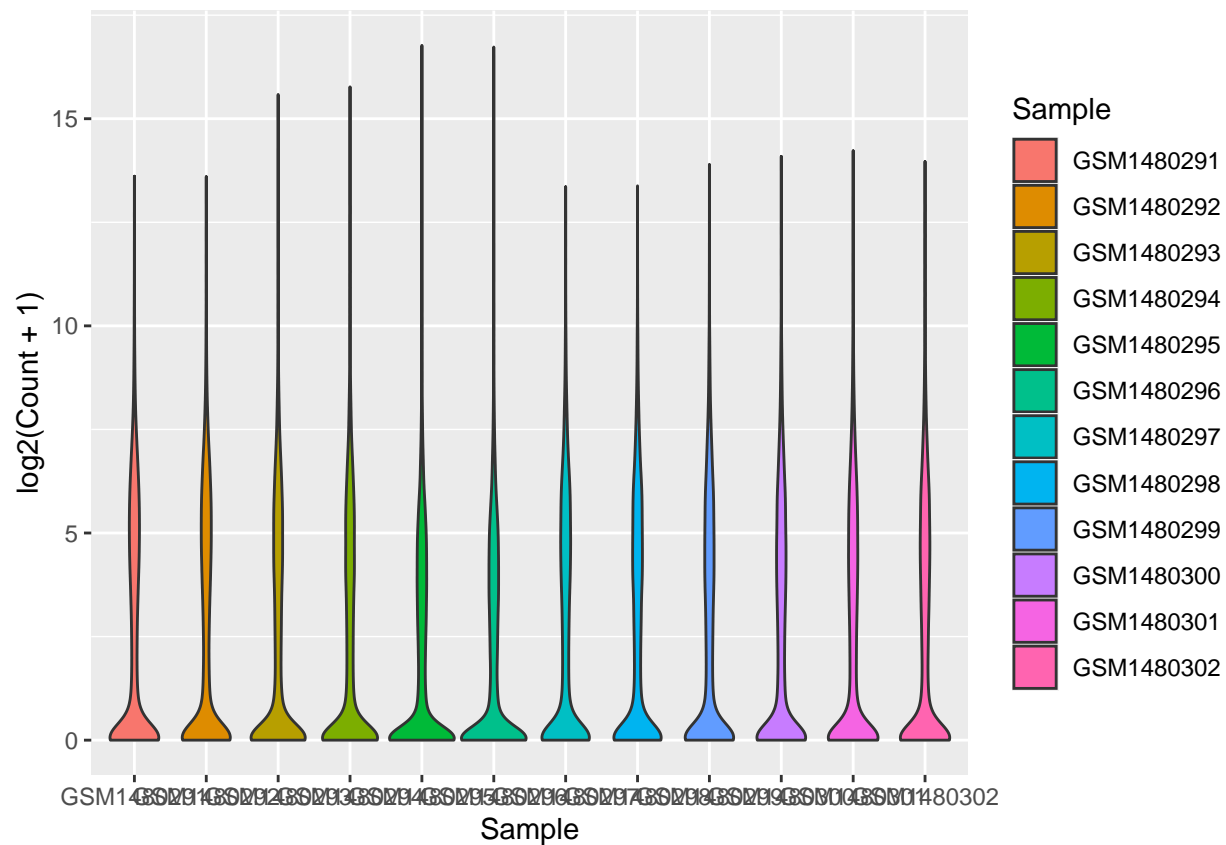
```



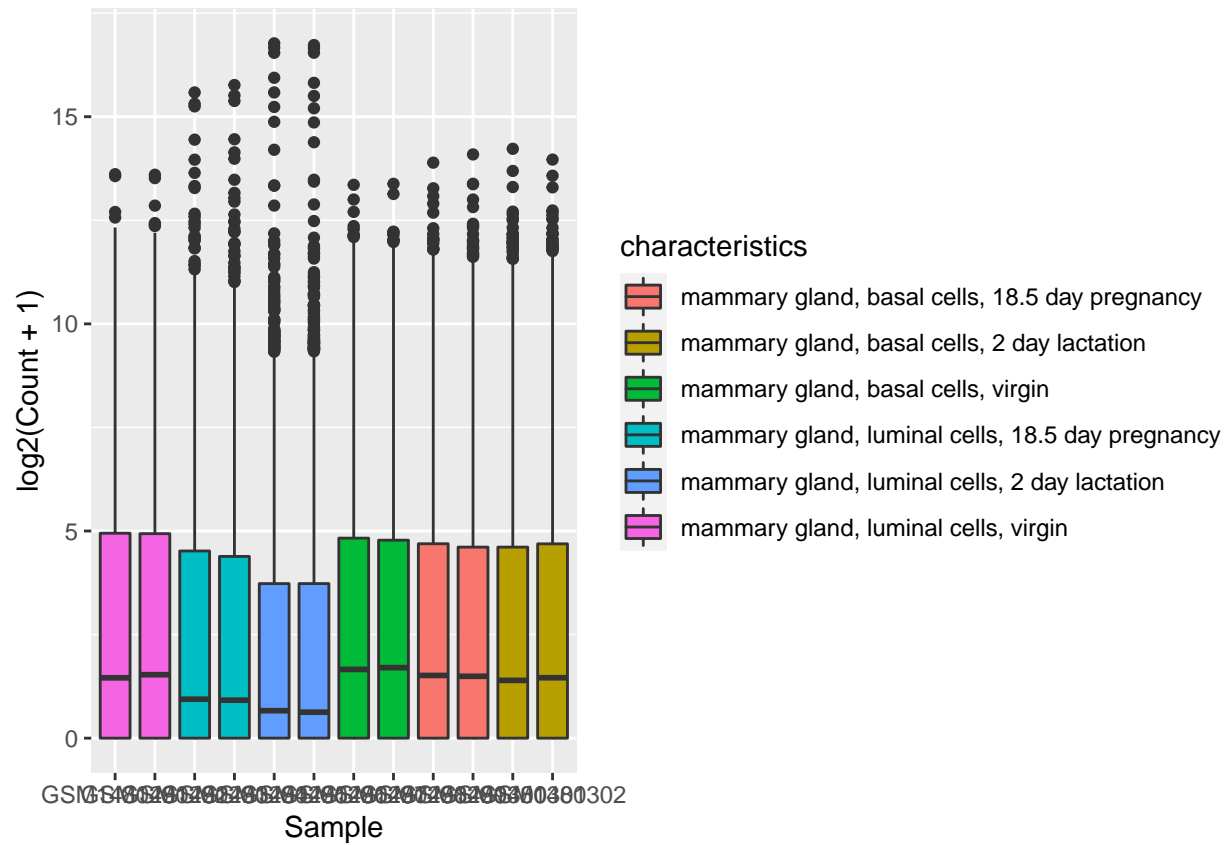
```

ggplot(allInfo, mapping = aes(x = Sample, y = log2(Count + 1), fill = Sample)) + geom_violin()

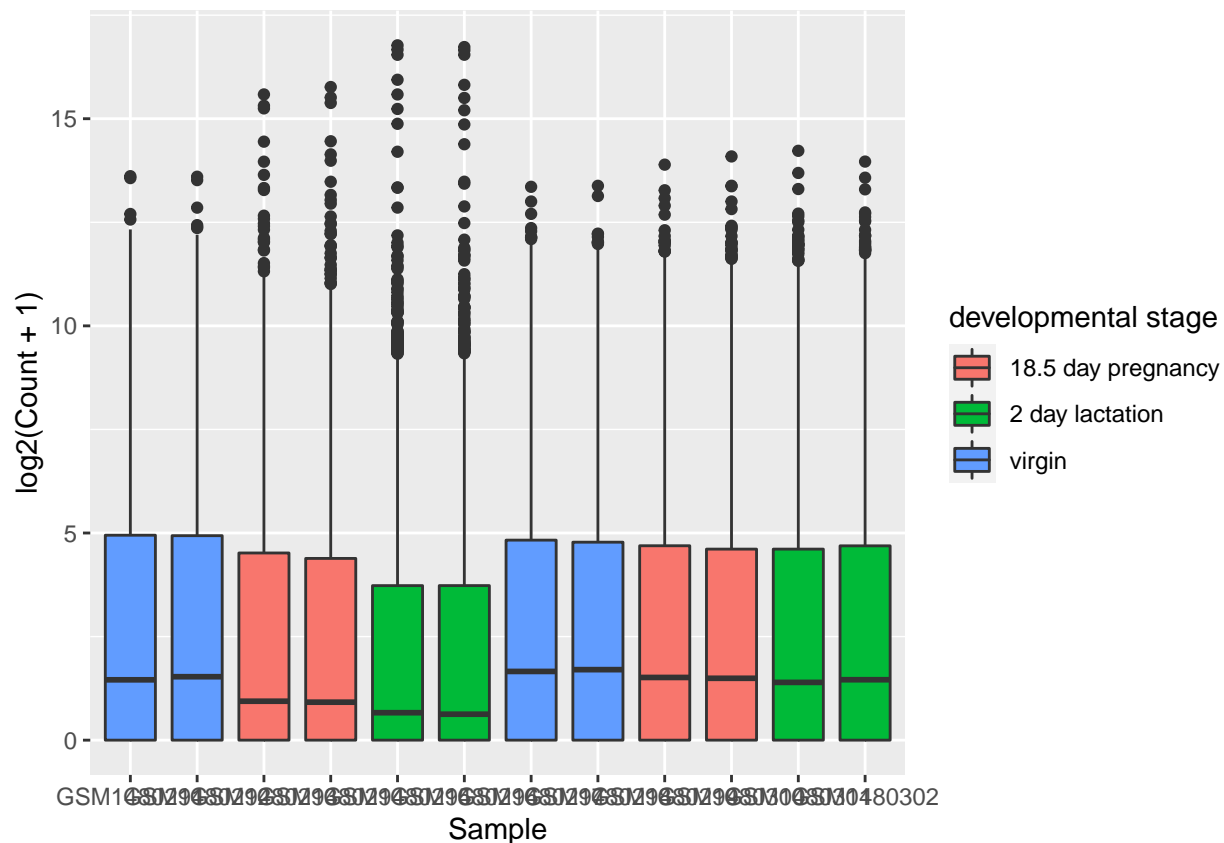
```

```
ggplot(allInfo, mapping = aes(x = Sample, y = log2(Count + 1), fill = characteristics)) + geom_boxplot()
```



```
ggplot(allInfo, mapping = aes(x = Sample, y = log2(Count + 1), fill = 'developmental stage')) + geom_boxplot()
```



```
# Shorten Category names
allInfo = mutate(allInfo, Group = case_when(
  str_detect(characteristics, 'basal.*virgin') ~ 'bvirg',
  str_detect(characteristics, 'basal.*preg') ~ 'bpreg',
  str_detect(characteristics, 'basal.*lact') ~ 'blact',
  str_detect(characteristics, 'luminal.*virgin') ~ 'lvirg',
  str_detect(characteristics, 'luminal.*preg') ~ 'lpreg',
  str_detect(characteristics, 'luminal.*lact') ~ 'llact',
))

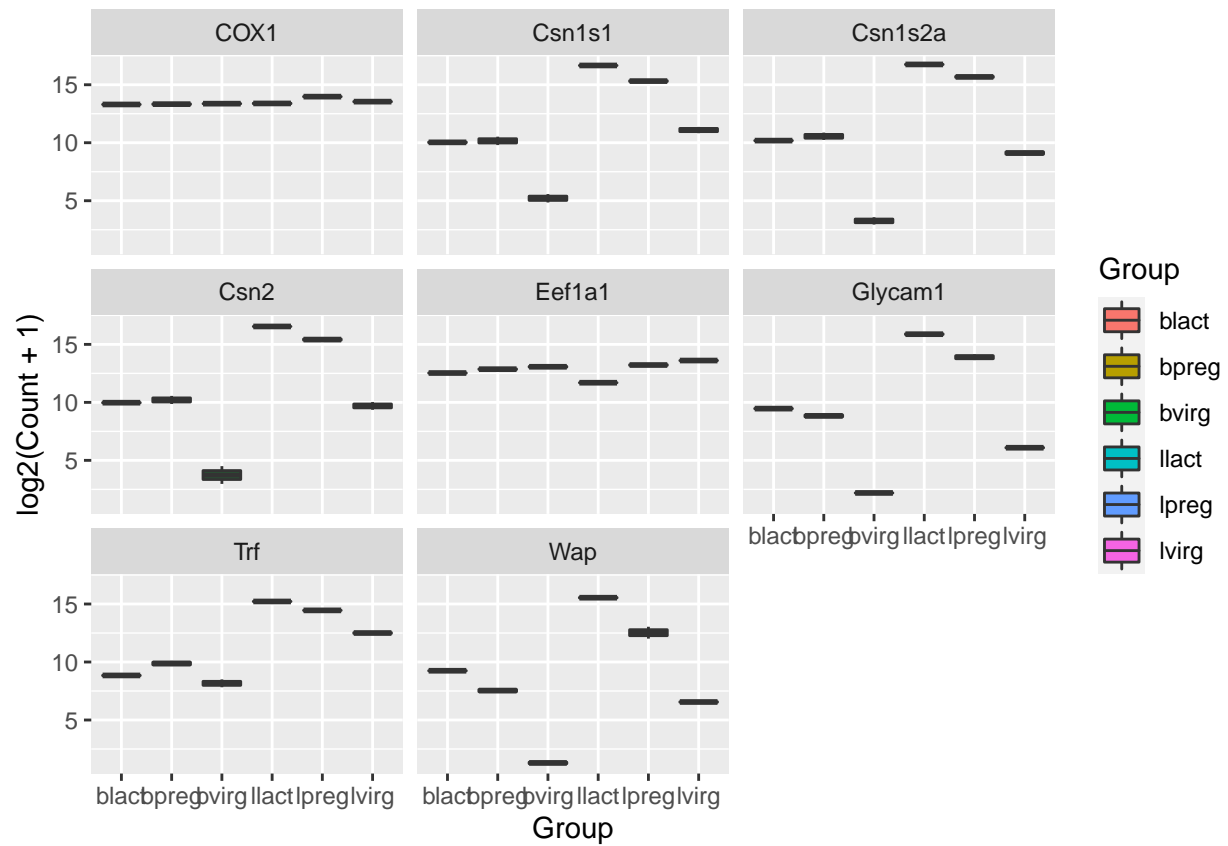
# Select 8 genes with the highest counts summed across all samples
myGenes = allInfo %>%
  group_by(gene_symbol) %>%
  summarise(Total_count = sum(Count)) %>% # remove repeated values
  arrange(desc(Total_count)) %>% # Arrange data into descending order
  head(n = 8) %>%
  pull(gene_symbol) # Pull out a single variable
```

```
## 'summarise()' ungrouping output (override with '.groups' argument)
```

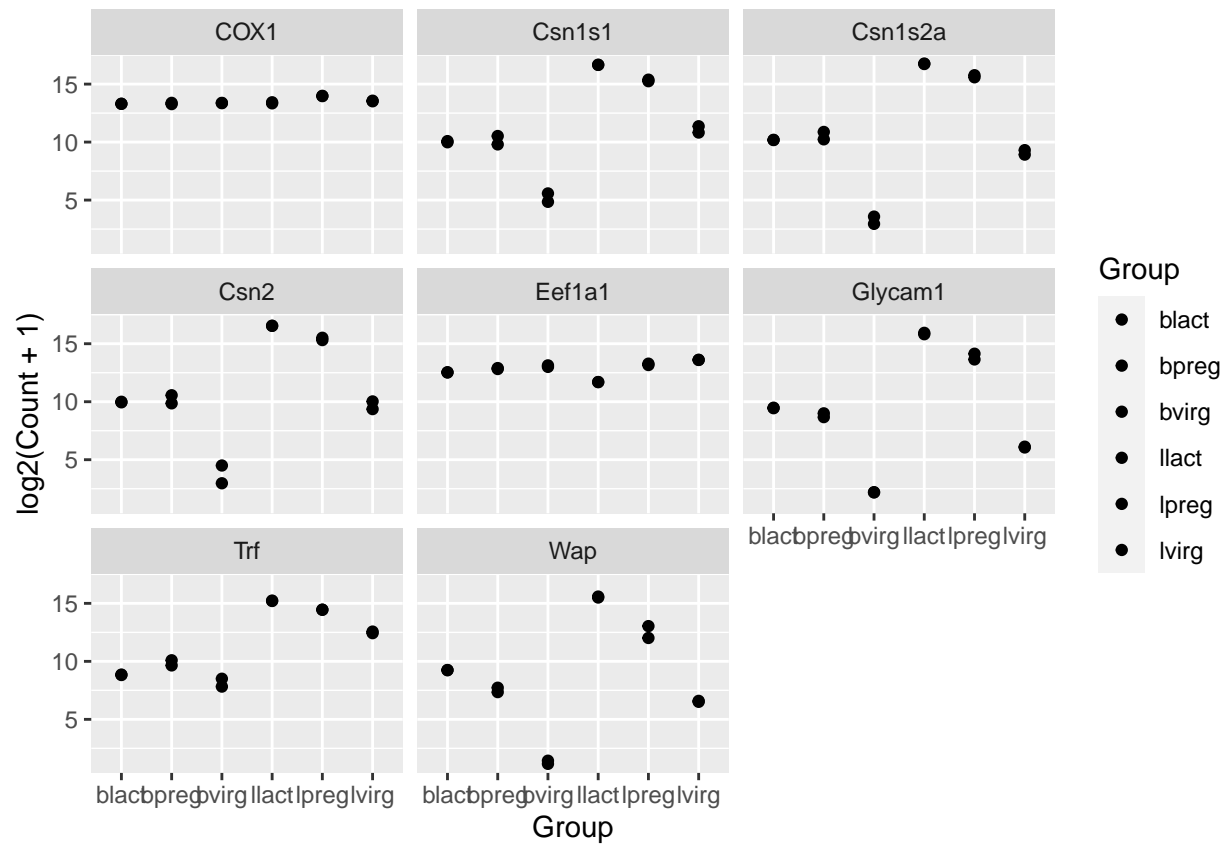
```
# Filter data
myGenesCounts = filter(allInfo, gene_symbol %in% myGenes)

# Create plot for each of the 8 genes
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), fill = Group)) +
```

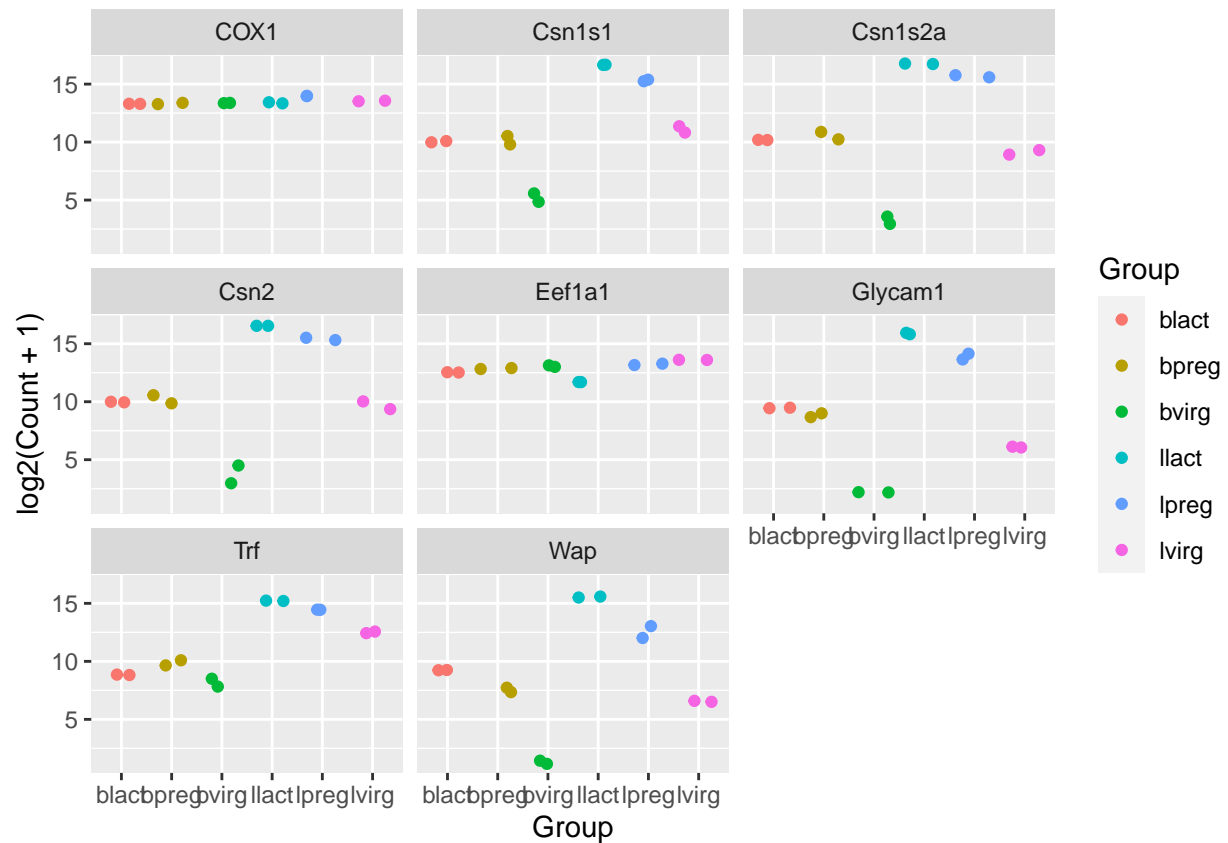
```
geom_boxplot() +  
facet_wrap(~ gene_symbol)
```



```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), fill = Group)) +  
geom_point() +  
facet_wrap(~ gene_symbol)
```

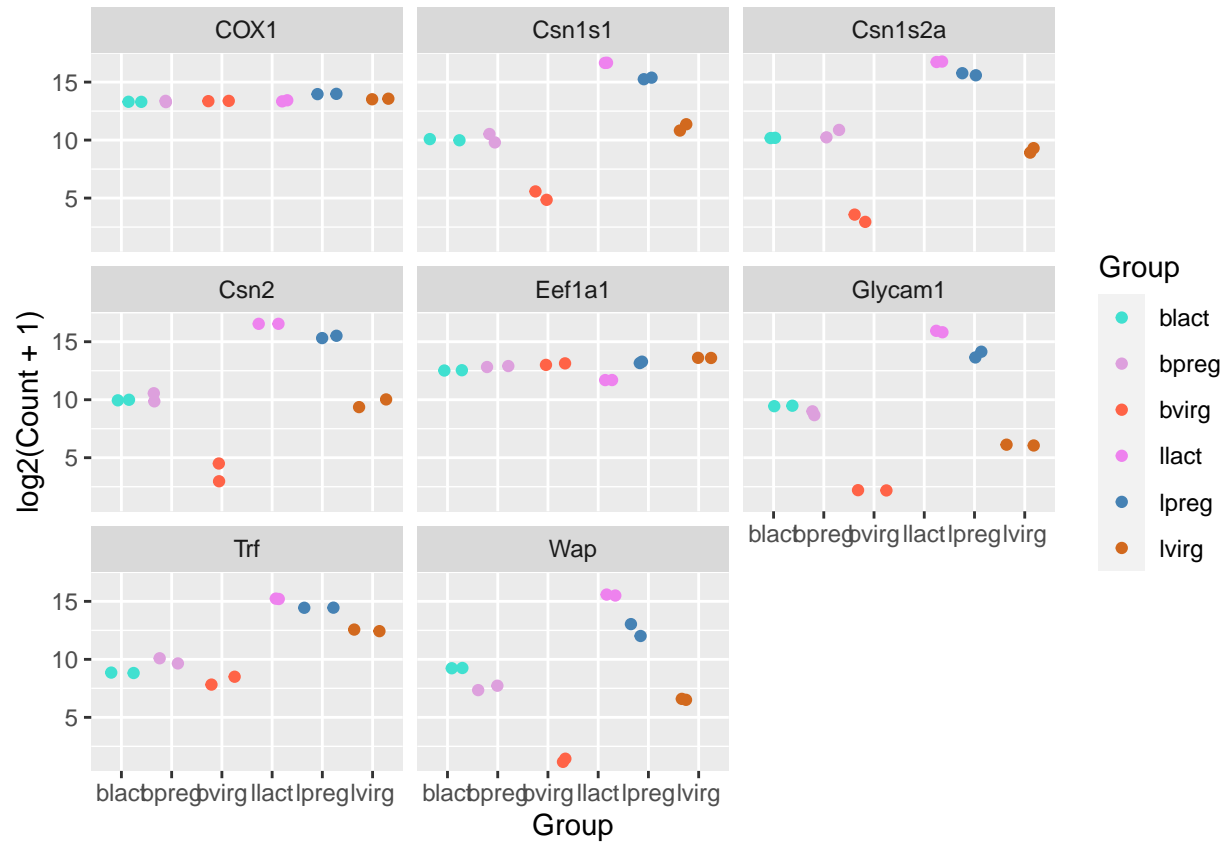


```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol)
```

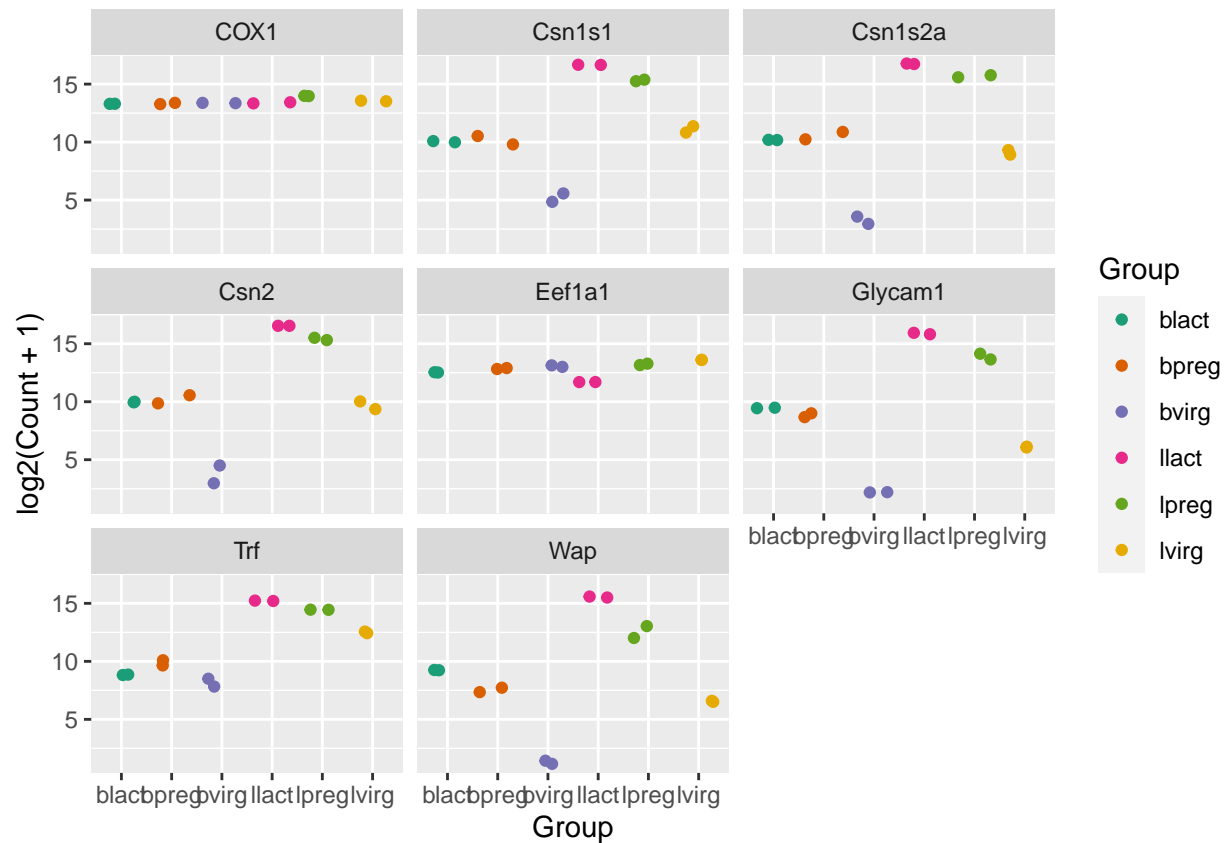


```
# customise plots
# colours
myColours = c('turquoise', 'plum', 'tomato', 'violet', 'steelblue', 'chocolate')

ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  scale_colour_manual(values = myColours)
```

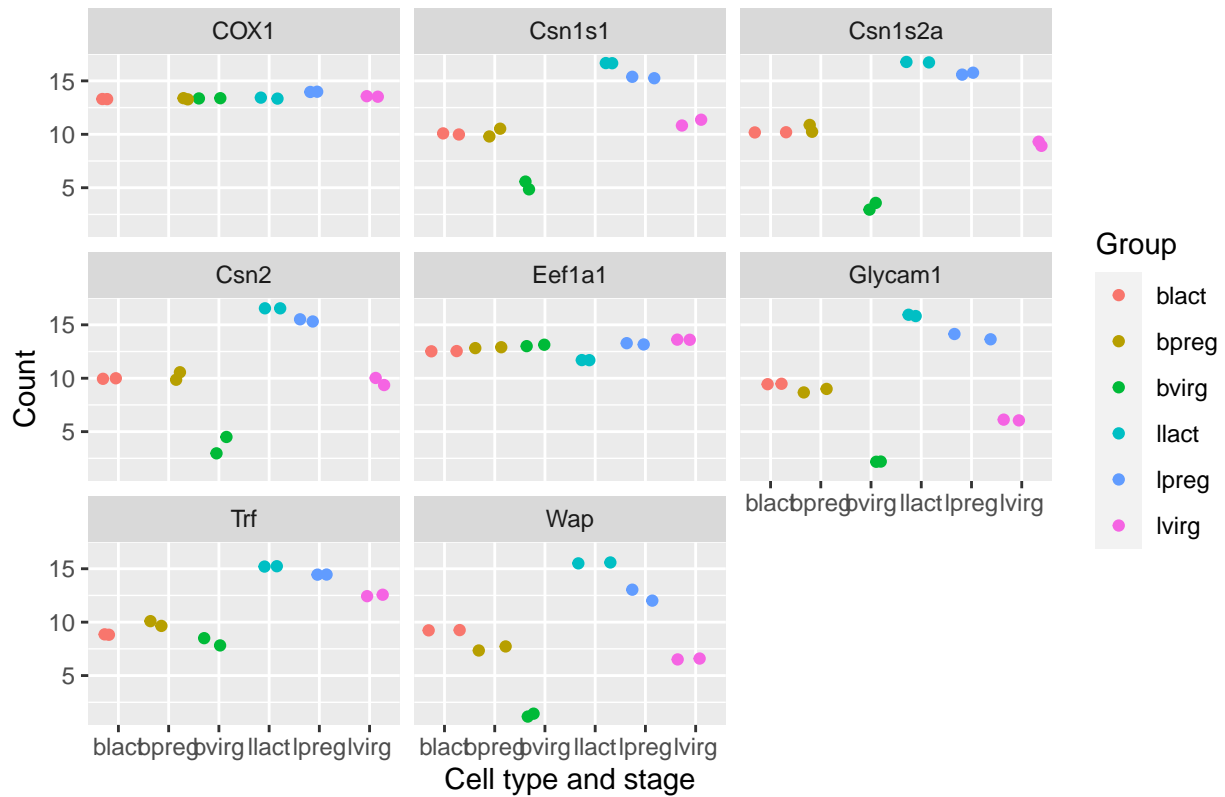


```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  scale_colour_brewer(palette = 'Dark2')
```



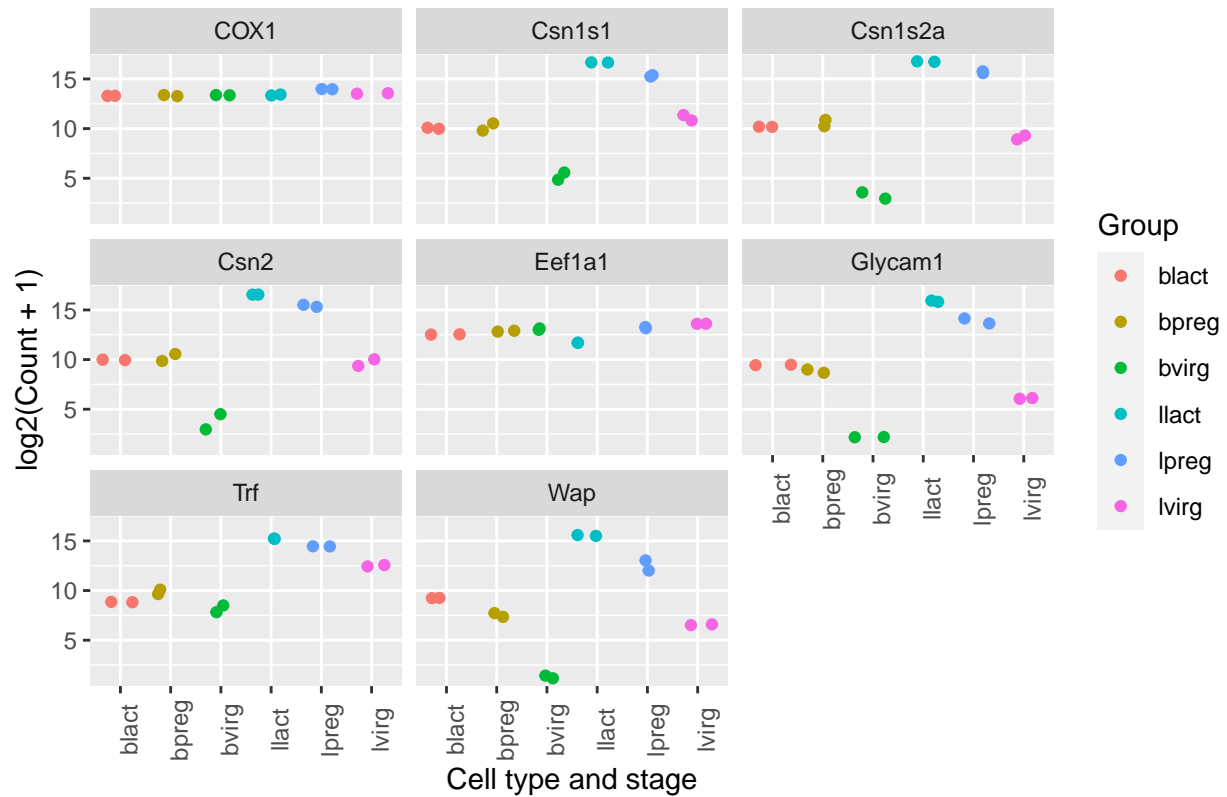
```
# axis
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'Count', title = 'Mammary gland RNA-seq data')
```


Mammary gland RNA-seq data



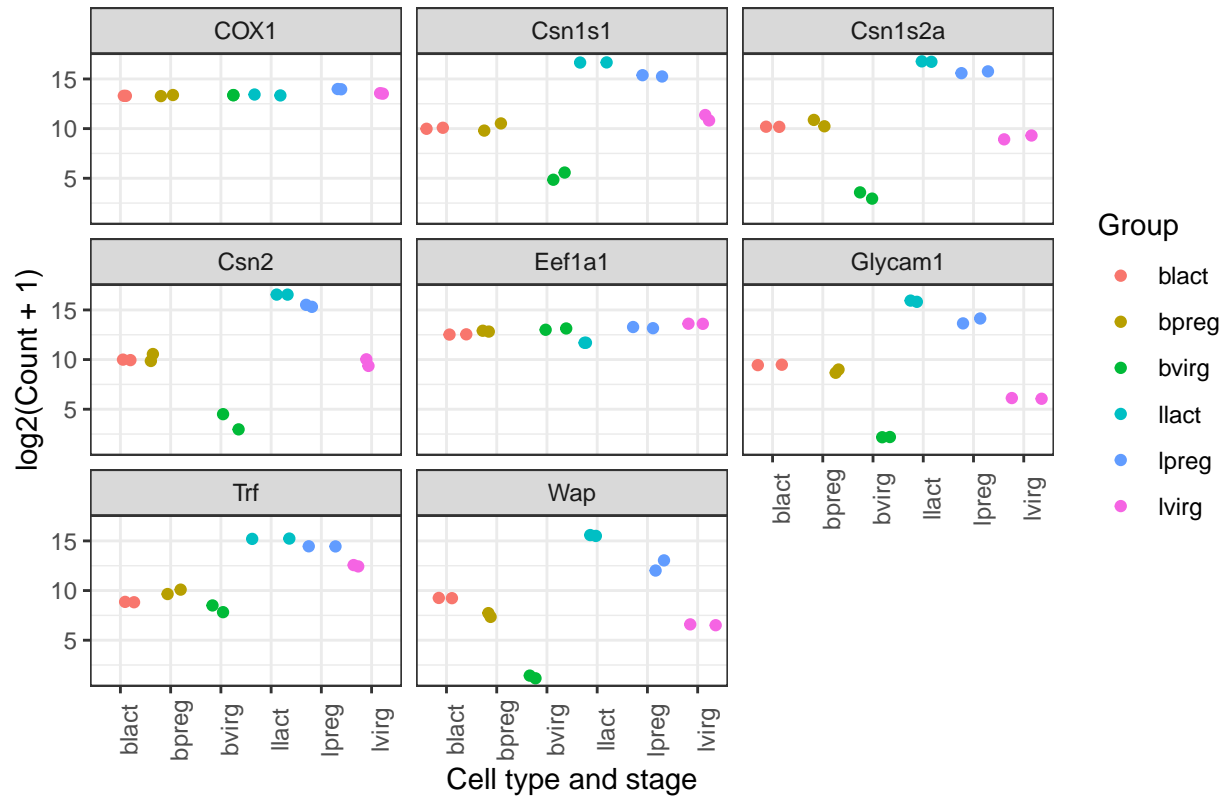
```
# theme
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme(axis.text.x = element_text(angle = 90))
```

Mammary gland RNA-seq data



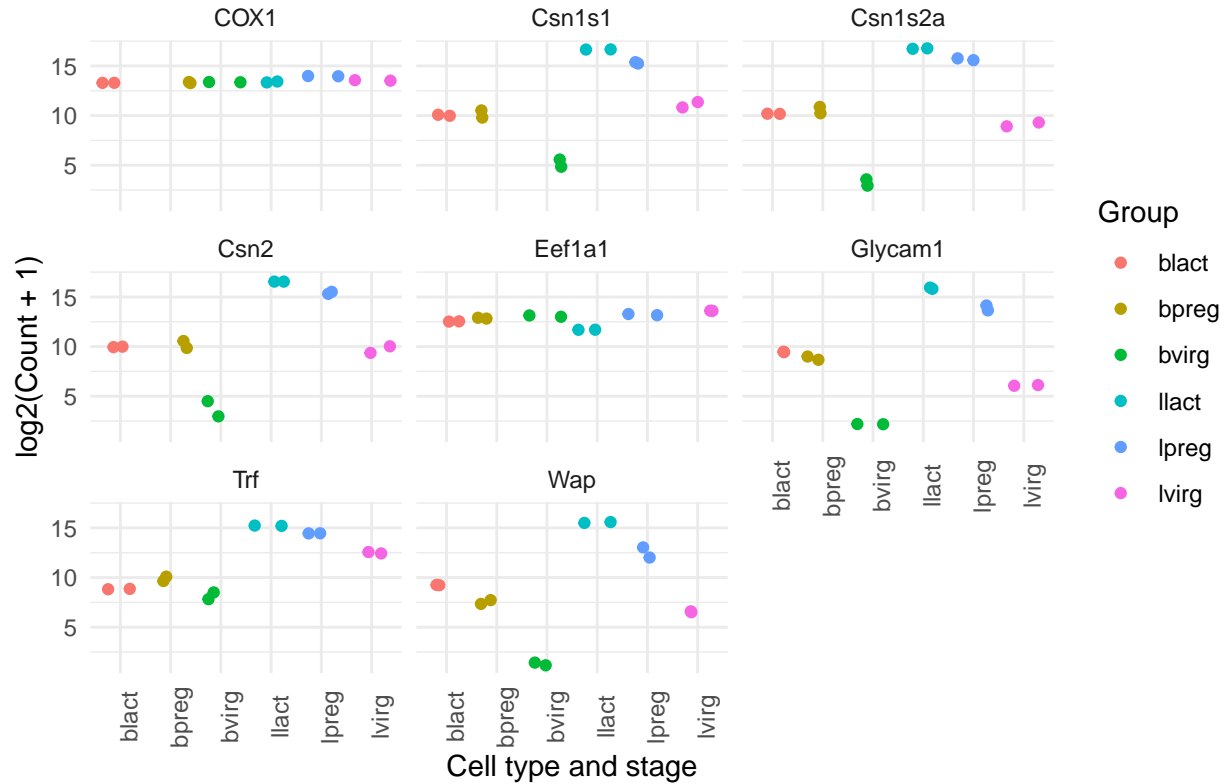
```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90))
```

Mammary gland RNA-seq data



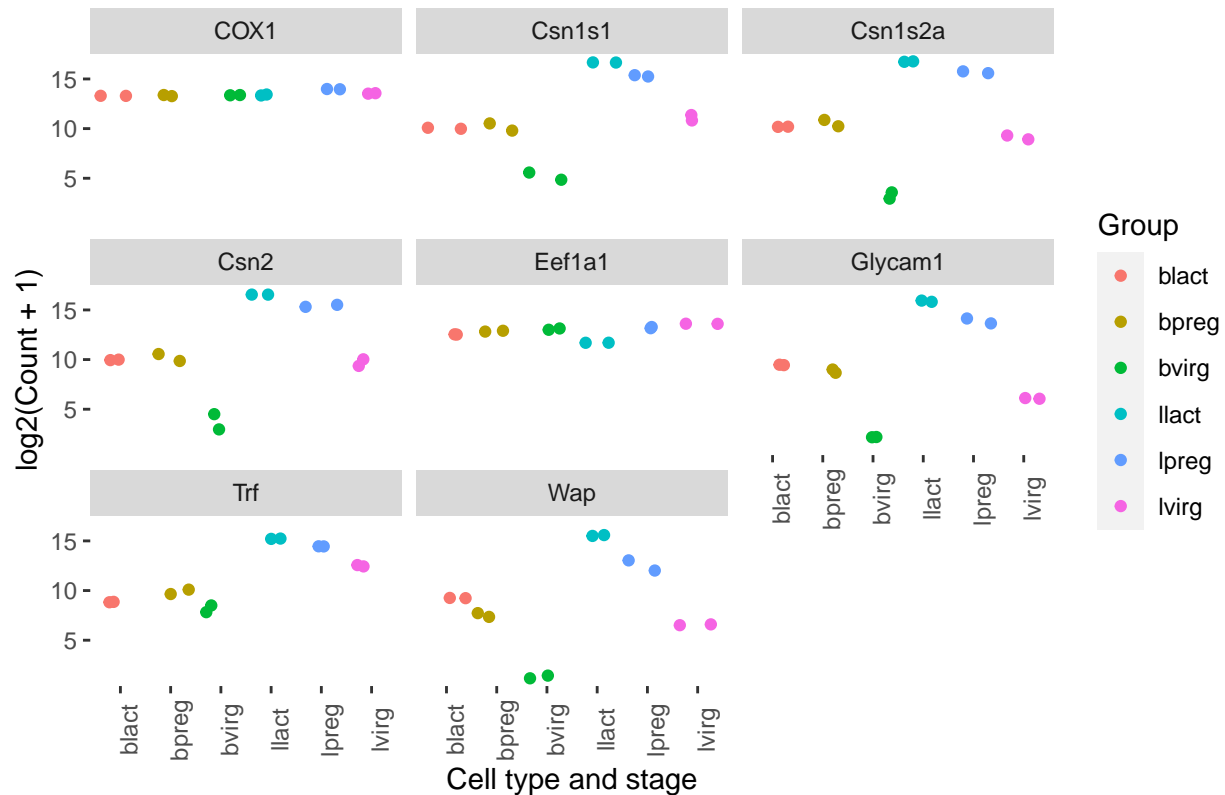
```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90))
```

Mammary gland RNA-seq data



```
ggplot(data = myGenesCounts, mapping = aes(x = Group, y = log2(Count + 1), colour = Group)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme(axis.text.x = element_text(angle = 90)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```

Mammary gland RNA-seq data



```
# order and categories(levels)
groupOrder = c('bvirg', 'bpreg', 'blact', 'lvirg', 'lpreg', 'llact')

myGenesCounts = mutate(myGenesCounts, Group_f = factor(Group, levels = groupOrder))

# selectively view data
myGenesCounts %>% select(X1, Group, Group_f)
```

```
## Warning: '...' is not empty.
##
## We detected these problematic arguments:
## * 'needs_dots'
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?
```

```
## # A tibble: 96 x 3
##   X1                Group Group_f
##   <chr>             <chr> <fct>
## 1 ENSMUSG00000000381 lvirg lvirg
## 2 ENSMUSG00000000381 lvirg lvirg
## 3 ENSMUSG00000000381 lpreg lpreg
## 4 ENSMUSG00000000381 lpreg lpreg
## 5 ENSMUSG00000000381 llact llact
## 6 ENSMUSG00000000381 llact llact
```

```
## 7 ENSMUSG00000000381 bvirg bvirg
## 8 ENSMUSG00000000381 bvirg bvirg
## 9 ENSMUSG00000000381 bpreg bpreg
## 10 ENSMUSG00000000381 bpreg bpreg
## # ... with 86 more rows
```

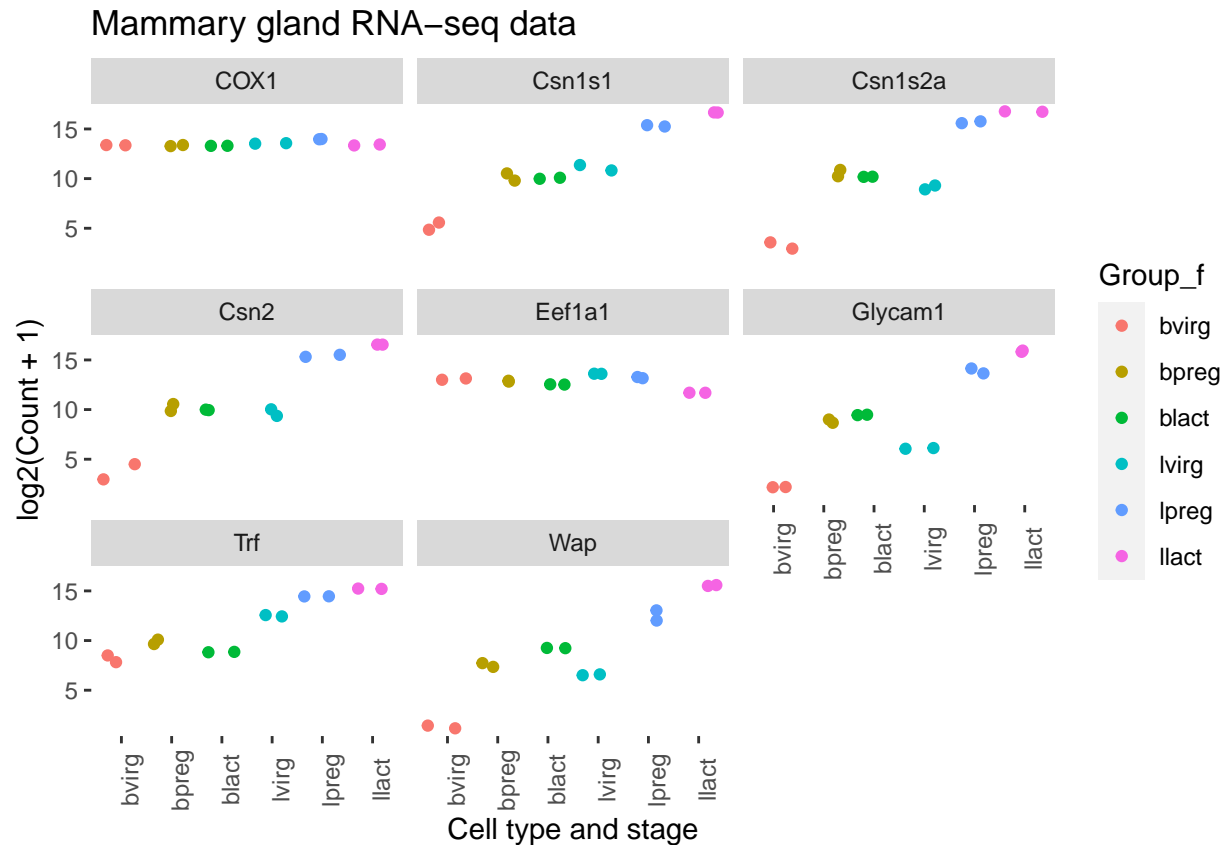
```
str(myGenesCounts)
```

```
## tibble [96 x 9] (S3: tbl_df/tbl/data.frame)
##  $ X1                : chr [1:96] "ENSMUSG00000000381" "ENSMUSG00000000381" "ENSMUSG00000000381" "E
##  $ gene_symbol        : chr [1:96] "Wap" "Wap" "Wap" "Wap" ...
##  $ Sample             : chr [1:96] "GSM1480291" "GSM1480292" "GSM1480293" "GSM1480294" ...
##  $ Count              : num [1:96] 90.2 95.6 4140.3 8414.4 49204.9 ...
##  $ characteristics    : chr [1:96] "mammary gland, luminal cells, virgin" "mammary gland, luminal ce
##  $ immunophenotype    : chr [1:96] "luminal cell population" "luminal cell population" "luminal cell
##  $ developmental stage: chr [1:96] "virgin" "virgin" "18.5 day pregnancy" "18.5 day pregnancy" ...
##  $ Group              : chr [1:96] "lvirg" "lvirg" "lpreg" "lpreg" ...
##  $ Group_f            : Factor w/ 6 levels "bvirg","bpreg",...: 4 4 5 5 6 6 1 1 2 2 ...
```

```
# check factor levels of a column
levels(myGenesCounts$Group_f)
```

```
## [1] "bvirg" "bpreg" "blact" "lvirg" "lpreg" "llact"
```

```
# alter the order of data
ggplot(myGenesCounts, mapping = aes(x = Group_f, y = log2(Count + 1), colour = Group_f)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme(axis.text.x = element_text(angle = 90)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```



```
# save the plot
pdf('myplot.pdf')

ggplot(myGenesCounts, mapping = aes(x = Group_f, y = log2(Count + 1), colour = Group_f)) +
  geom_jitter() +
  facet_wrap(~ gene_symbol) +
  labs(x = 'Cell type and stage', y = 'log2(Count + 1)', title = 'Mammary gland RNA-seq data') +
  theme(axis.text.x = element_text(angle = 90)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())

dev.off()
```

```
## pdf
## 2
```