post01-jessica-li

Jessica Li

2017 October 31

The Use of ggplot2 and other R Packages in Bioengineering Research

The `ggplot2` package, which has been utilized extensively so far in Stat 133, produces declarative graphics for data representation. In class, we have specifically used `ggplot2` to evaulate data about NBA statistics. However, `ggplot2` and other packages are useful in bioengineering research. I will explore other packages and their applications, then focus on my own lab's use of `ggplot2` in cell tracking research.

Common R Packages in Research

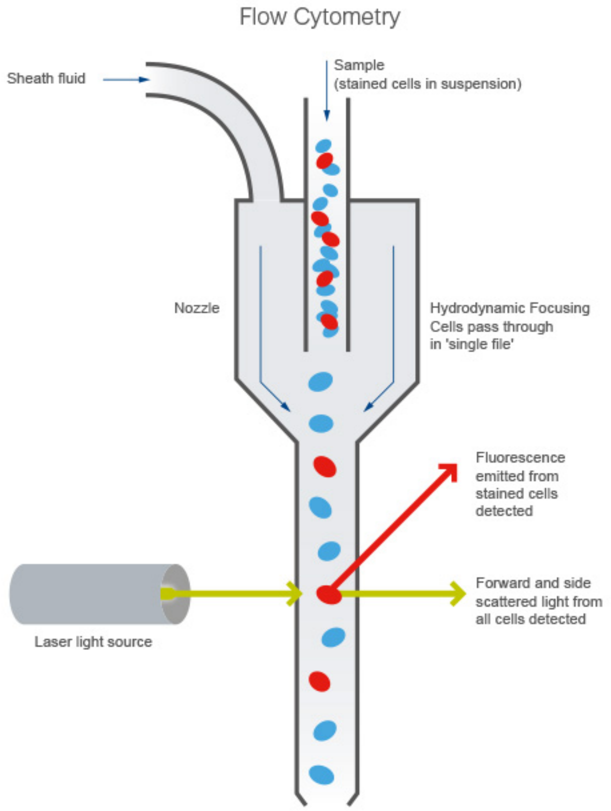
> This section describes common packages used in research as well as their applications. I do not personally use these packages in my research, but some of my fellow lab members do, so I drew on their experience and internet documentation to reflect on their application.

`tidyverse`

`tidyverse` is not a package unique to bioengineering research, but the few bioengineering researchers I spoke all use it. `tidyverse` consists of the discrete packages `ggplot2`, `dplyr`, `tidyr`, `readr`, `purr`, and `tibble`. As we have been using these in class, these packages are not unique to bioengineering research, but they are useful and commonly used, so I wanted to include them. The benefits of `tidyverse` include the ability to create tidy data, which streamlines the ability to cleanly represent data so researchers can focus on analytical questions, create tibbles, which correct a few frustrations with data frames, and other useful quick shortcuts. Since we have already discussed `tidyverse` in class, I will not more deeply analyze its use. Suffice it to say that many R users, including bioengineers, do use it.

`flowCore`

`flowCore` is a package used to analyze flow cytometry data. Flow cytometry is an experimental procedure that first stains cells with different fluorescent markers signifying various proteins or cell types then funnels the cells single-file past a laser beam, and counts the number of cells that emit each fluoresence to count the number of cells of each type.

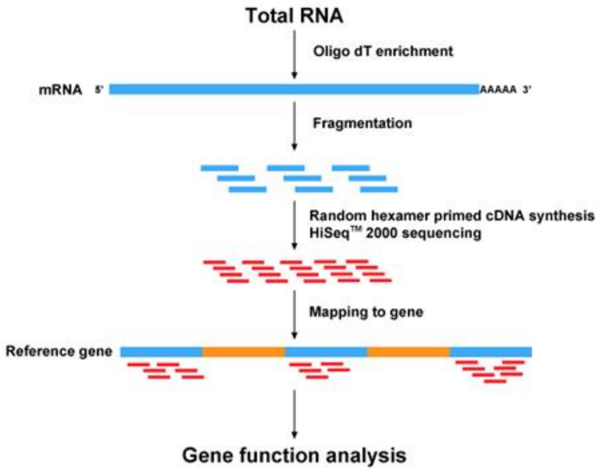


This image summarizes the process of flow cytometry as a easy-to-visualize schematic; the actual machine is much more complicated-looking than two grey lines.

`flowCore` enables researchers to evaulate flow cytometry data with relevant functions and data structures. The package includes many useful functions, which analyze both standard and specific cases of flow cytometry data. One of the most useful features of `flowCore` is the `compensatedParameter` class, which corrects for overlapping fluorescent signal between the different cell types, which cleanly differentiates different types of cells. Flow cytometry is an ubiquitous technique in most bioengineering labs that work with cells, so `flowCore` is a useful tool to analyze the resulting data.

`DESeq`

`DESeq` is a package used to analyze extensive sequencing results from procedures such as RNA sequencing. RNA sequencing takes in strands of mRNA, determines what nucleotides they're composed of, then determines to what genes they correspond.



This image summarizes the procedure of RNA sequencing in more detail than necessary; all that is necessary here is the fact that RNA sequencing takes in the researchers' RNA of interest and determines what genes correspond to that RNA.

`DESeq` enables researchers to systematically analyze the genes that correspond to the RNA of interest. Fundamentally, this package creates a statistical model of the data to analyze gene expression, AKA RNA.

My Lab's Research

> This section lays out a general protocol of how my lab creates cell counting diagrams in research.

Data import

Since many bioengineering machines, such as fluorescence plate microscopes and cell counters, export their data in Excel format, the package readxl allows the import of Excel files, preventing the need to modify the data file before analysis.

```
#load necessary libraries
library(ggplot2)
library(readxl)
excel_data <- read_excel("data/cell_counts_tile.xlsx")
```

Note: when not publishing code, read_excel will accept file.choose() as a parameter, which opens a pop-up window in which to choose the data file from a GUI of the machine. This is useful in research because the data is usually exported onto an external hard drive, and may be difficult to move locally.

Data can also be loaded from a .csv file, as we have learned in class:

```
data <- read.csv("data/cell_counts.csv")
#Note that this is a different file with different data than excel_data.
#This merely serves to reinforce the process of loading a .csv file.
```

In the remainder of this post, for authenticity, I will use `excel_data` as the main dataset.

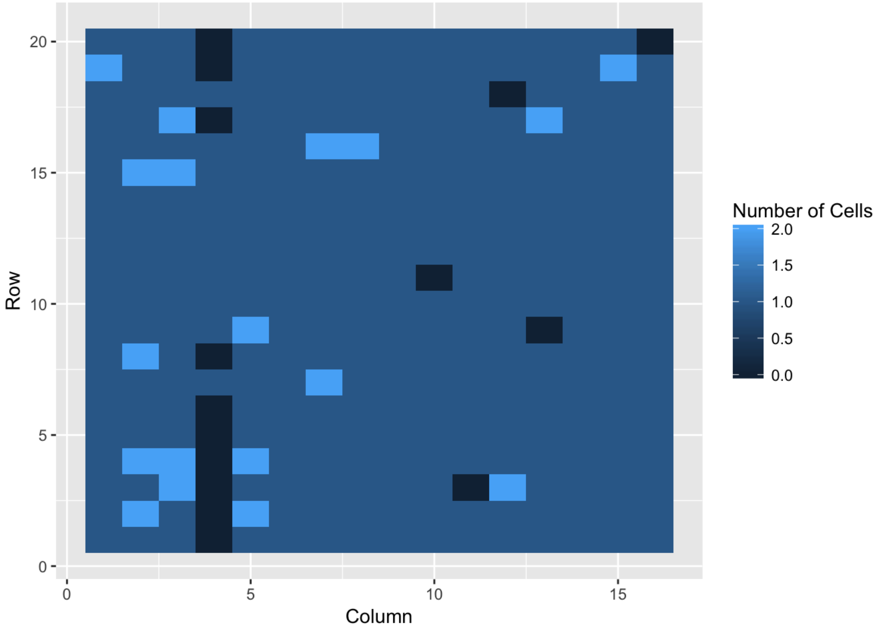
Summary of `ggplot` and introduction to `geom_tile`

The function `ggplot`, as we have learned and practiced, creates fully customizable representative graphs of datasets. Up until now, we have used the aesthetic `geom_line`, which creates line graphs suitable for tracing changes in a variable with respect to another, and `geom_point`, which notes relationships between two variables. Both of these can be used for `excel_data`.

The premise of the experiment from which `excel_data` was derived is that one slide was prepared with 320 tiny microislands, and cells were added to each microisland at 0 hours. After 2 hours, the slides were imaged and the cells in each microisland counted as a baseline measurement. Approximately every 12 hours after that, the slide was imaged again and the cells at that time point counted in order to track how fast cells grow.

`geom_line` can be used to represent the data by connecting the number of cells in each microisland at each time point. However, for a more graphical depiction of the data, the aesthetic `geom_tile` is useful.

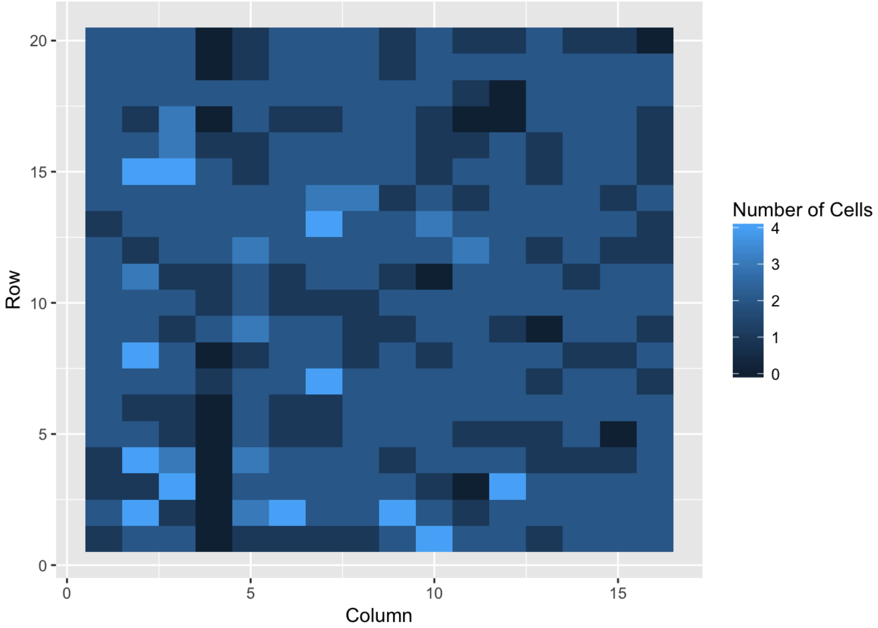
```
ggplot(data = excel_data,
       aes(x = excel_data$column, y = excel_data$row,
           fill = excel_data$'2hr')) +
  geom_tile() +
  labs(x = "Column", y = "Row",
       fill = "Number of Cells")
```



This plot is incredibly representative since it is a visual depiction of the slide itself; each tile represents a microisland, and the color of each tile represents the number of cells in that microisland after 2 hours.

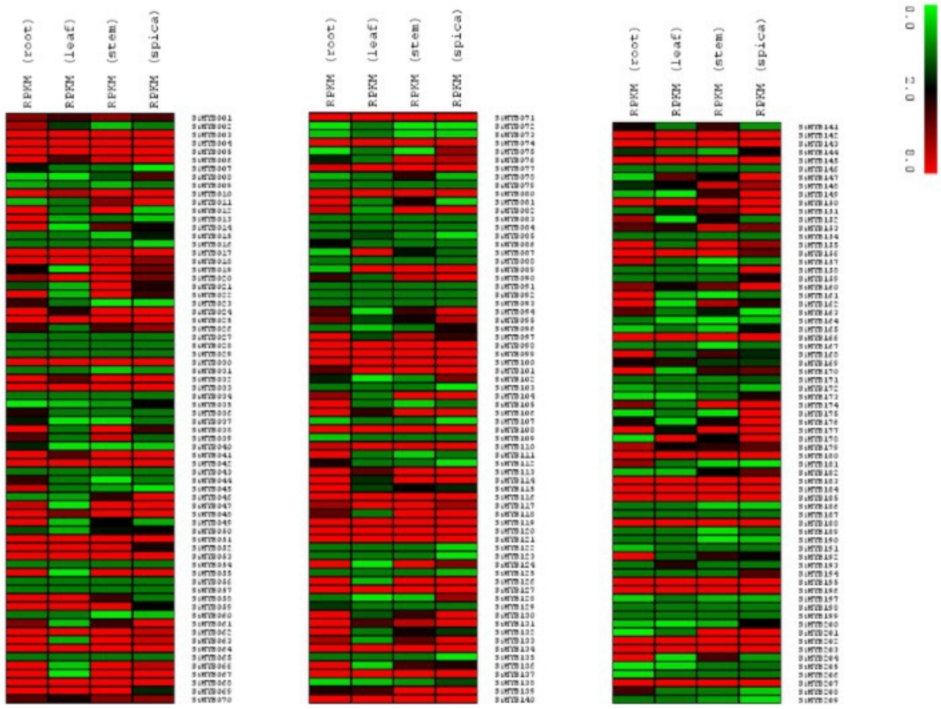
For cell tracking, the use of `geom_tile` is handy in comparing the appearance of slides at different time points. For example, this plot would give the appearance of the slide after 24 hours:

```
ggplot(data = excel_data,
       aes(x = excel_data$column, y = excel_data$row,
           fill = excel_data$'24hr')) +
  geom_tile() +
  labs(x = "Column", y = "Row",
       fill = "Number of Cells")
```



Between this plot and the previous one, the tile at each position represents the same microisland (at the same row/column coordinates), but the different color of the tile represents a different number of cells in the microisland after the elapsed time.

This is a brief introduction into how my lab uses the aesthetic `geom_tile` in the function `ggplot` of the package `ggplot2` to track cell growth in individual microislands. Other labs use `geom_tile` to represent RNA heat maps, which depicts the level of expression of different genes.



A typical RNA heat map looks like this. This particular one depicts different transcription factors for the plant foxtail.

To interpret this use of `geom_tile` in `ggplot`, each tile represents a certain gene in a certain type of cell. The color the tile represents the level of expression of that gene in the cell.

Conclusion

My lab uses R, in this post, specifically the novel-to-this-class aesthetic variable `geom_tile` to track cell numbers in experiments. However, many other scientists have created and provided packages, in this post, `flowCore` and `DESeq`, that allow for analysis of many types of bioengineering research data. R is a powerful tool for analysis of all types of data in a wide variety of fields.

References

- https://www.tidyverse.org/packages/
- http://r4ds.had.co.nz/tidy-data.html
- http://r4ds.had.co.nz/tibbles.html
- http://a.static-abcam.com/CmsMedia/Media/flowcytometry01472px.jpg
- https://www.rdocumentation.org/packages/flowCore/versions/1.38.2
- https://www.rdocumentation.org/packages/flowCore/versions/1.38.2/topics/compensatedParameter-class
- http://bio.lundberg.gu.se/courses/vt13/rna4.JPG.jpg
- http://bioconductor.org/packages/release/bioc/html/DESeq.html
- http://ggplot2.tidyverse.org/reference/ggplot.html
- https://www.researchgate.net/profile/Mehanathan_Muthamilarasan/publication/265395972/figure/fig5/AS:214225446072328@1428086734414/The-Illumina-RNA-seq-data-were-re-analyzed-and-heat-map-was-generated-Bar-at-the-top.png