Building Shiny Apps Challenges and Responsibilities

Jessica Minnier

email: minnier@ohsu.edu

@datapointier

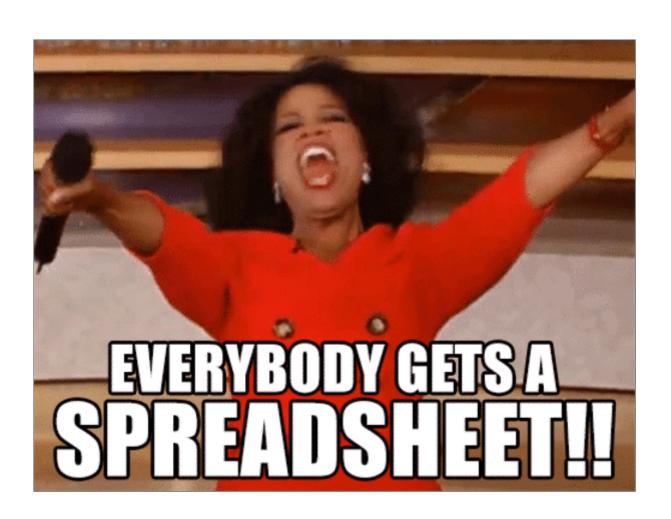
Saturday, January 27, 2018, Data/R Day Texas

Slides available at http://bit.ly/shiny-ddtx

Why Shiny?

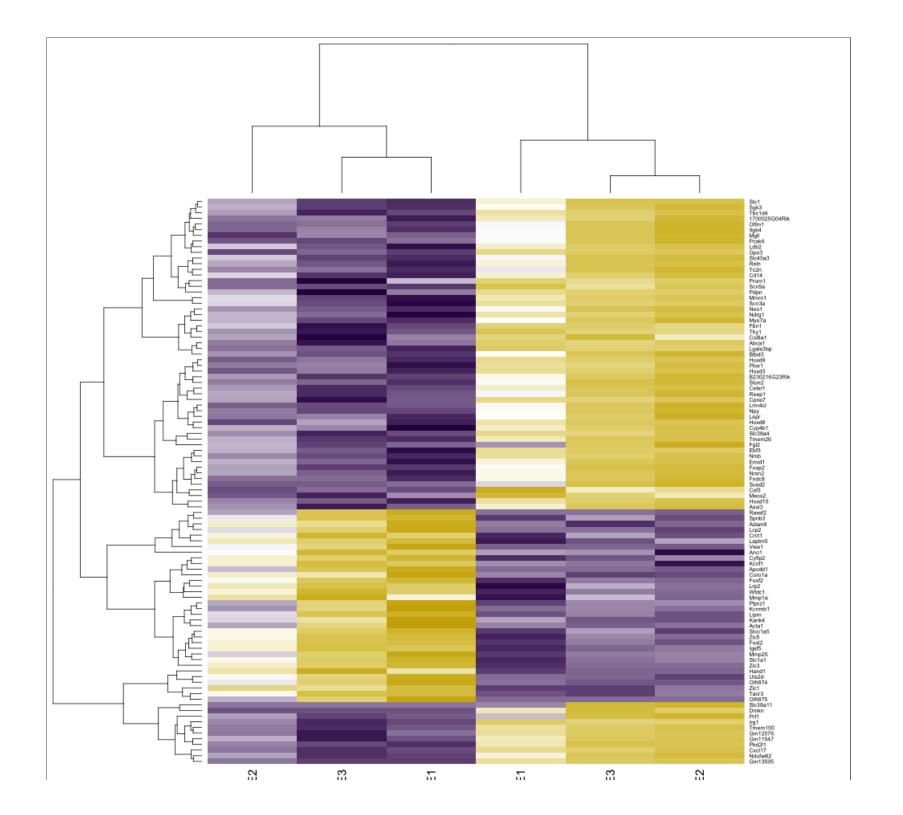
Motivation

Spreadsheets = (



Motivation

"Can you make me this plot? Ok now this plot? Wait one more plot..."



Queried Genes with FDR <= 0.01 (top 100 genes) from LRT test for difference across groups: BE vs. HE

Plot Monkey?

Enter Shiny



www.rstudio

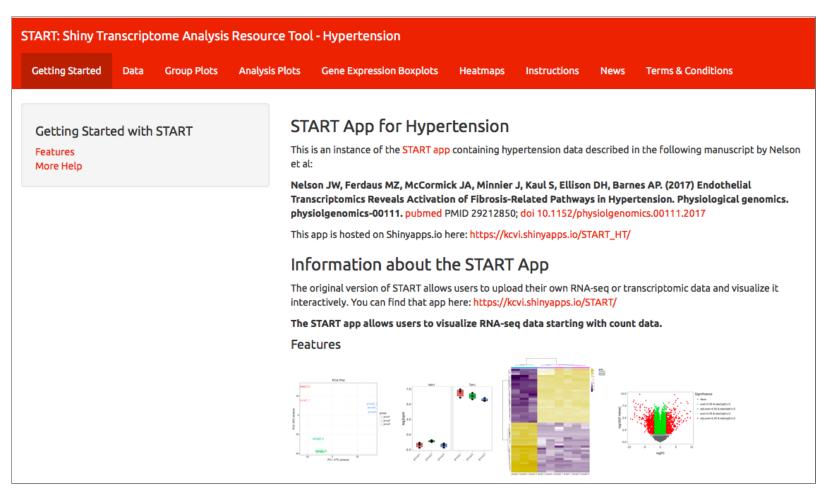
My history

- Started with Tableau
 - Tableau + R + Bioconductor = :-| (in 2013)
 - (Now you can **host a shiny app in Tableau??**)
- Realization: Shiny can do this!
- Made one site with one data set
- Created general site for input data set

Motivation For All

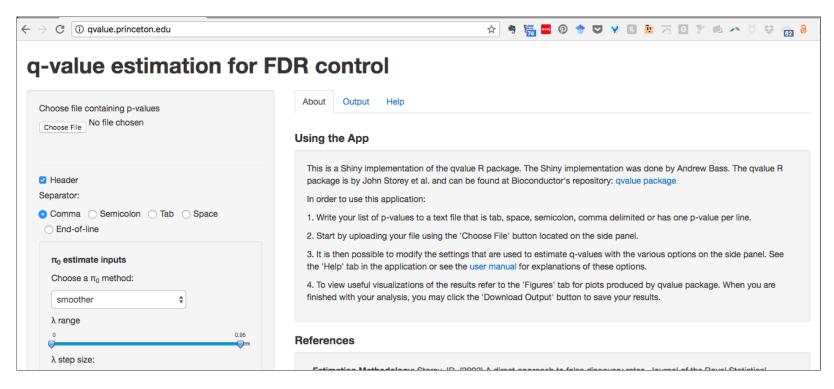
Distribute your hard work!

Collaborator: "How can I share my data?"



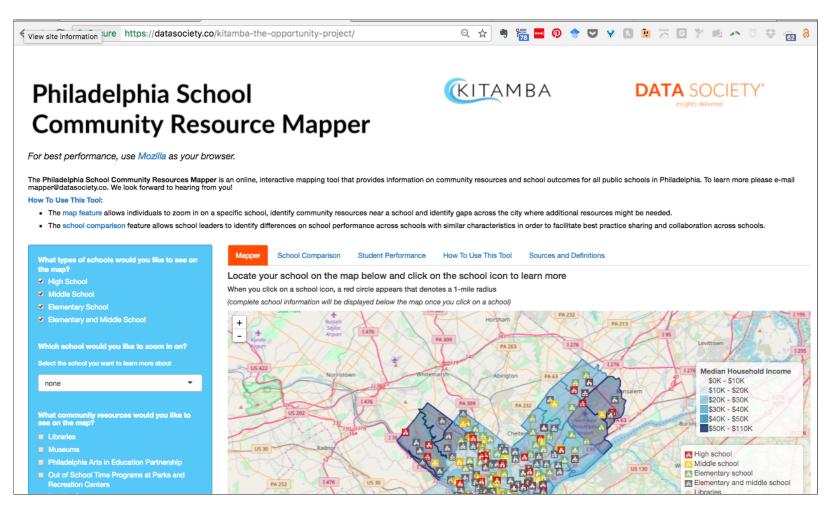
https://kcvi.shinyapps.io/START HT/

Developer/Statistician/Data Scientist: "How can I get people to use my new method/package?"



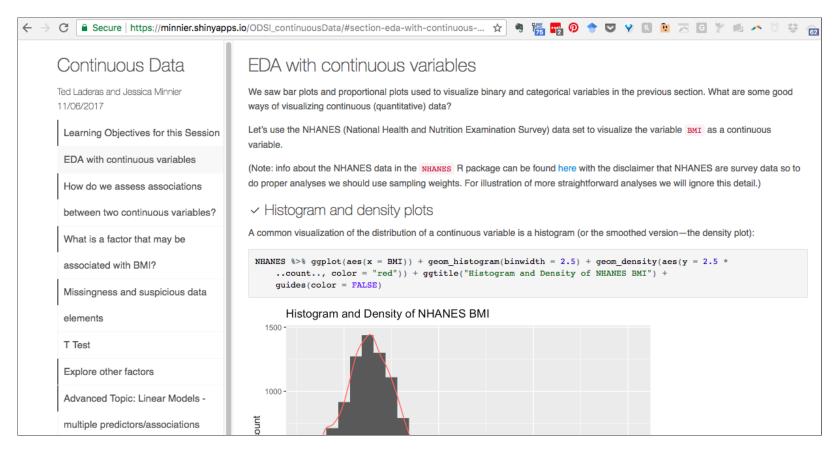
http://qvalue.princeton.edu/

Developer/Statistician/Data Scientist: Look at this analysis I did!



https://gallery.shinyapps.io/TSupplyDemand/

Everyone: Teach Some things! (learnr)



https://minnier.shinyapps.io/ODSI continuousData/

Shiny basics

```
# what you see, client side
ui <- fluidPage(
)</pre>
```

```
# computations here!
server <- function(input, output) {</pre>
```

```
# run the app!
shinyApp(ui = ui, server = server)
```

Shiny basics

```
# what you see, client side
ui <- fluidPage(
    # Title, panels
    plotOutput("myplot"), #from output$myplot
    selectInput("foo",choices=allthechoices)
)</pre>
```

```
# computations here!
server <- function(input, output) {
}</pre>
```

```
# run the app!
shinyApp(ui = ui, server = server)
```

Shiny basics

```
# what you see, client side
ui <- fluidPage(
    # Title, panels
    plotOutput("myplot"), #from output$myplot
    selectInput("foo",choices=allthechoices)
)</pre>
```

```
# computations here!
server <- function(input, output) {
    # reactive plots
    output$myplot <- renderPlot({ ggplot(yay) + geom_point() })
    # use input$foo here, maybe observe changes in UI
    observe({ dostuff(input$foo) })
}</pre>
```

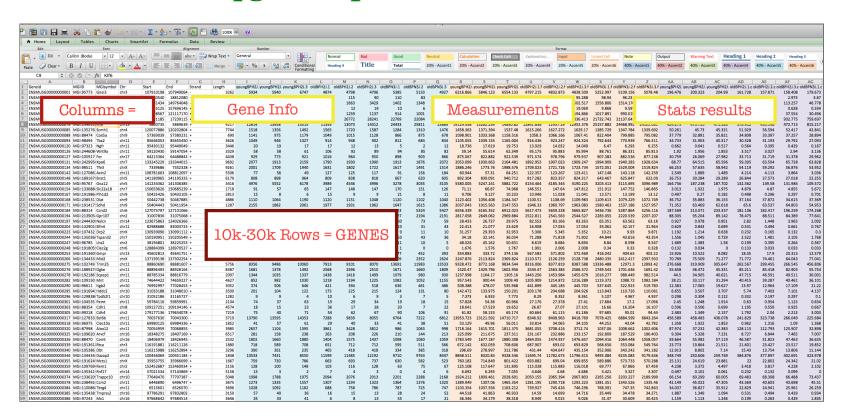
```
# run the app!
shinyApp(ui = ui, server = server)
```

An example: START

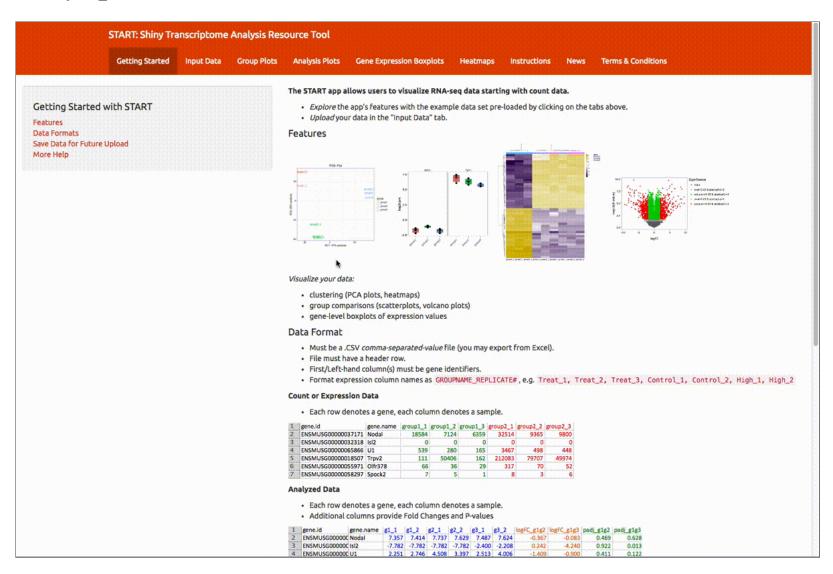
Shiny Transcriptome Analysis Resource Tool

Github: https://github.com/jminnier/STARTapp

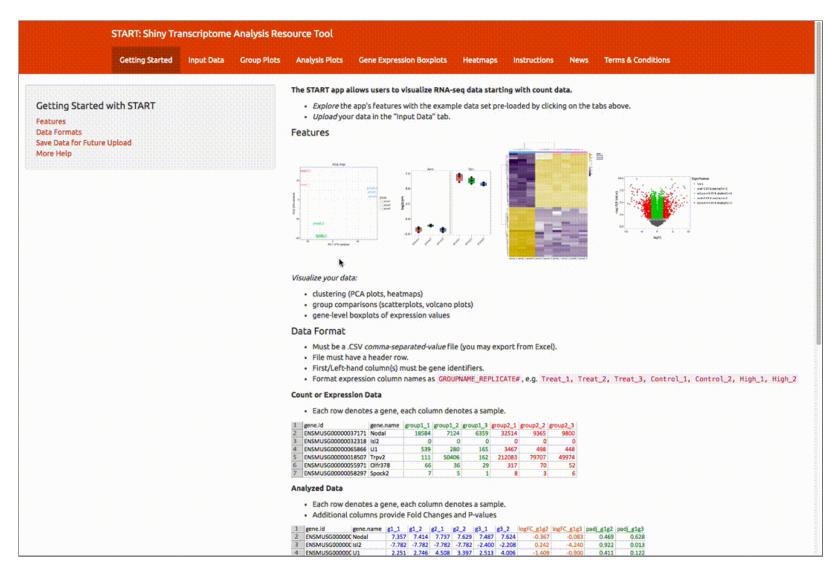
DATA = **RNA-seq gene expression**



Very quick tour



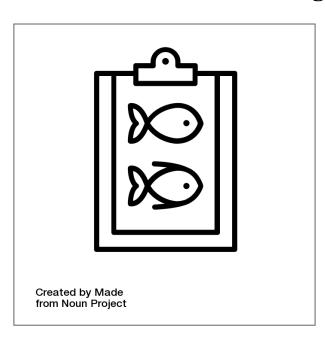
Very quick tour



Who is your audience?

Example: START app

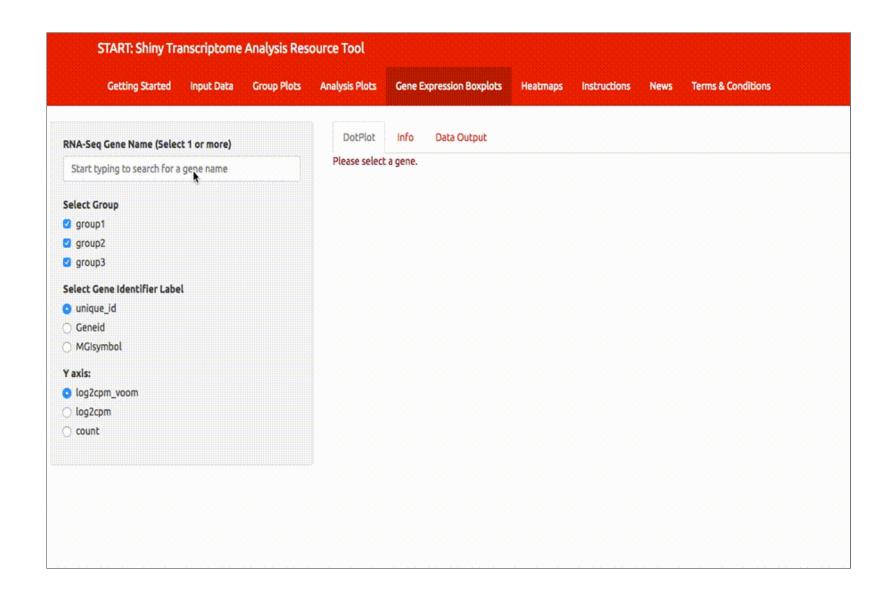
- likely biologists
- some not used to coding or large data sets



Challenges: *User*

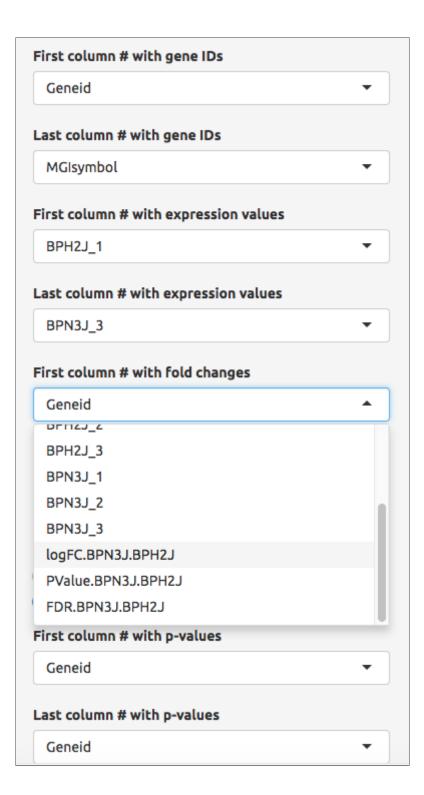
Data is larger than they are used to + Cannot query or share results easily

Interactive search boxes/filtering + Reactive visualization



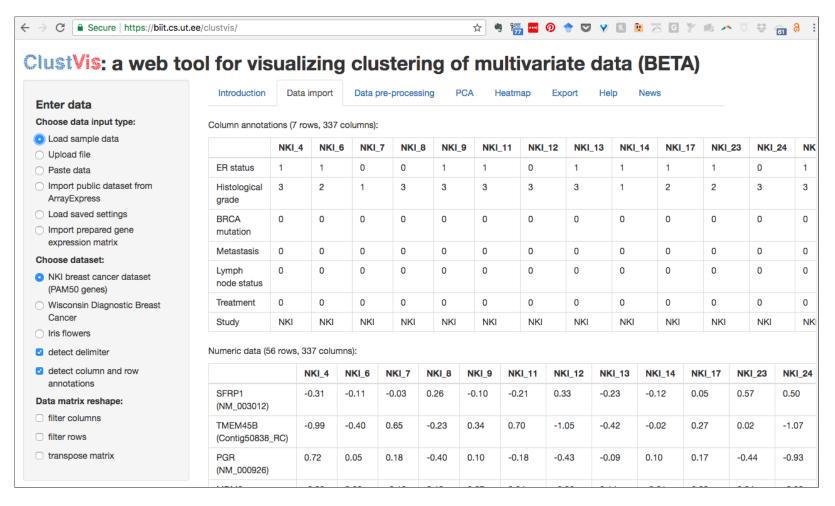
Challenges, Input/Ouput Data

- Input data ~ multiple formats: Special input formats
- Interface with other software? Export special formats



Challenges, Input/Ouput Data

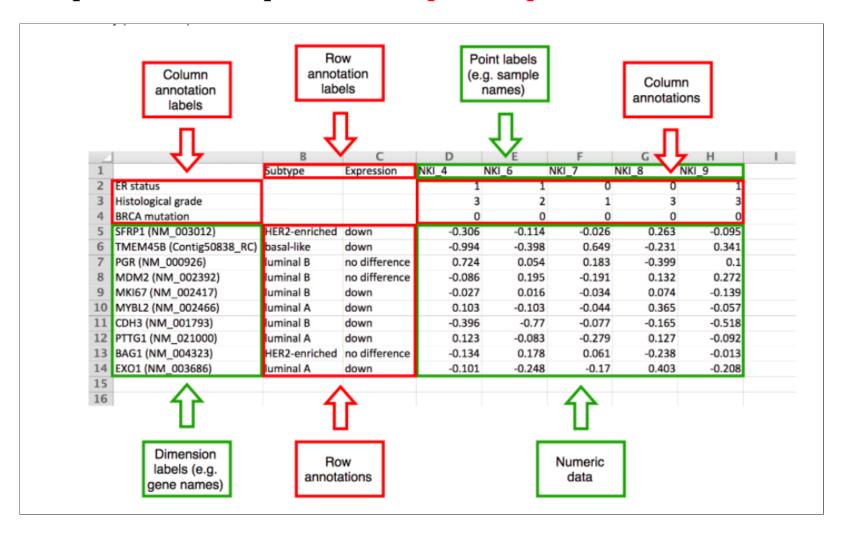
• Input data ~ multiple formats: Special input formats



https://biit.cs.ut.ee/clustvis/

Challenges, Input/Ouput Data

• Input data ~ multiple formats: Special input formats



https://biit.cs.ut.ee/clustvis/

Challenges, Analysis

Many possible ways to analyze data:

- Restrict analyses performed
- Allow for uploading of analysis results

Download Instructions (pdf)

Use example file or upload your own data

- Upload Data
- START RData file
- Example Data

Input Data Type:

- Expression data: Gene Counts or log-expression (log2cpms)
- Analyzed data: Expression Values, p-values, fold changes

Download Example Analysis Results File

1	gene.id gene.name	g1_1	41.2	g2_1	8.2	(1,1	43,2	logFC_g1g2	logIC_glg3	ped_glg2	pedj_glg3
2	ENSMUSG000000 Nodal	7.357	7.414	7,737	7.629	7,487	7.624	-0.367	-0.083	0.469	0.628
3	ENSWUSG000000 N/2	-7.782	-7.782	-7.782	-7.782	-2.400	-2.208	0.242	-4.240	0.922	0.013
4	ENSMUSG000000 U1	2,251	2.746	4,508	3.397	2.513	4.006	-1.409	-0.900	0.411	0.122
-	ENSAU ISGODOCCC Trav2	-0.024	10.137	10.441	10.719	0.951	1.535	-6.365	2.731	0.444	0.317

- · File must have a header row.
- Format expression column names as GROUPNAME_REPLICATE#: Group1_1, Group1_2, Group2_1, Group2_2...
- Number & order of fold changes must MATCH pvalue number & order.

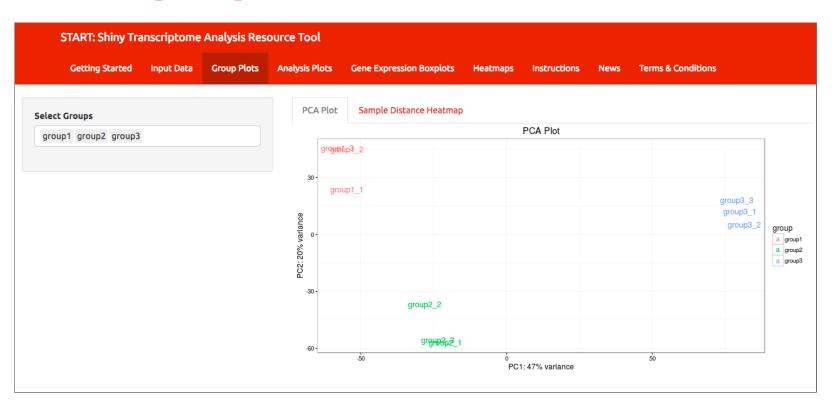
Choose File Containing Data (.CSV)

Choose File No file chosen

Challenges, Analysis

QA/QC

Present QA plots up front

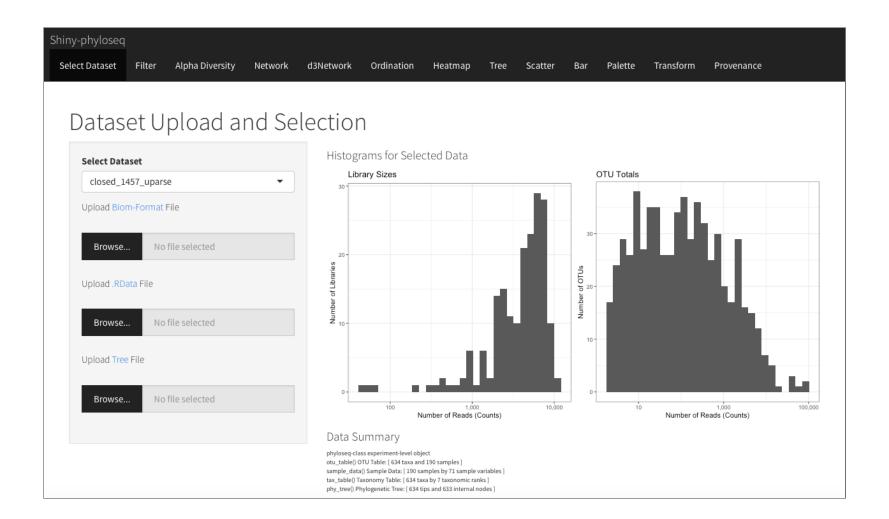


Challenges, Analysis

QA/QC

Present QA plots up front

https://github.com/joey711/shiny-phyloseq



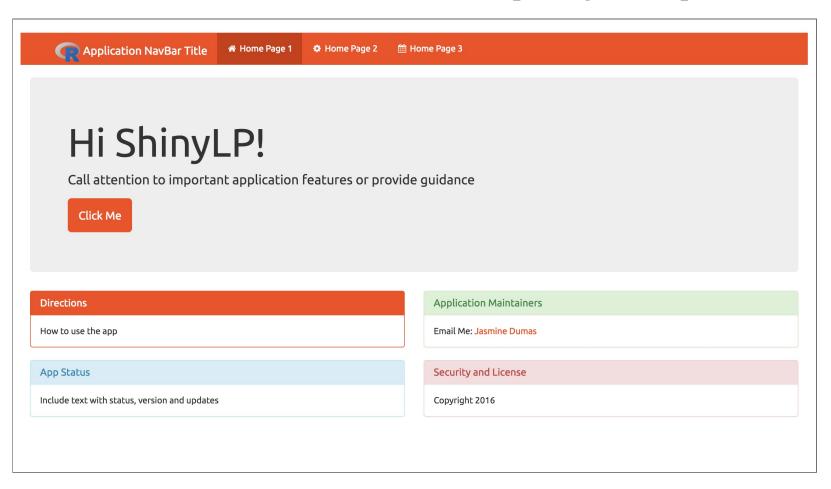
Responsibilities, Food for thought:

- Is resulting analysis correct?
- Are we encouraging "data fishing"?
- Misuse of p-values?
- Solutions? Use instructions and data inputs wisely.
- Other solutions? Avoid Hos, Bayesian posterior probabilities?

Landing page - Where am I?

- Show what the app is for/can do
- Point to instructions and guidelines

Bonus: Jasmine Dumas has a shinyLP package to help!



Instructions, Please

- So hard.
- Not just a vignette of a package (users may not be coders)
- Written instructions vs. video demos
- Gif instructions

No one will use it if they don't know **how** to use it!

Instructions

Tauno Metsalu and Jaak Vilo: "Clustvis"

ClustVis: a web tool for visualizing clustering of multivariate data (BETA)

Help

General
Data import
Public dataset from MEM
List of datasets available
List of pathways available
Annotations based filtering
Data pre-processing
PCA and heatmap
Interactivity of the plots
Interpreting the output
Mathematical basis

ClustVis editions

Introduction Data import Data pre-processing PCA Heatmap Export Help News

General

You can move through the analysis steps by going to each of the tabs from left to right. All tabs work in a similar way: you can choose settings from the left panel, image or table on the right will automatically renew after that. Sometimes, it can take seconds to load. When moving from one tab to another, settings are saved automatically.

In general, different tabs provide the following options:

- Data import choose input dataset, option to filter rows/colums based on annotations and transpose matrix
- Data pre-processing option to aggregate columns with similar annotations, remove rows/columns with missing values, row
 centering and scaling, method for calculating principal components and imputing missing values
- PCA options related to PCA plot
- Heatmap options related to heatmap
- · Export create link with current settings, download intermediate results

The idle timeout (the time when browser session ends if user is inactive) is set to 30 minutes from server side but this can be overridden by browser configuration. To save uploaded data and selected settings, you can use a button on the 'Export' tab, a link is given to recover the settings later. This can also be used to send a link to a collaborator to show the same view. There is no planned expiration time for the links, users can delete the settings if they are concerned about the privacy. Though, when version of ClustVis changes, old saved settings may not be fully compatible with the new version if e.g. there are some new features.

Data import

We aimed for a simple input data format. The numeric data matrix is situated in the bottom right corner, dimensions presented in rows and points in columns. Row labels and annotations are left from the matrix, column labels and annotations are above the matrix. Annotation labels are in the first row and column, respectively. Format of the input file is shown on the image below. Annotations are optional, data sets without annotations can be uploaded as well (on the example image, omitting rows 2-4 and/or columns B and C). When taking data from spreadsheet program (e.g. MS Excel), you can copy-paste the data to 'Paste data' box or export the data as delimited text file (ending with .csv or .tab) and then upload this file to ClustVis. Uploading Excel native files directly (.xls or .xlsx) doesn't work.

Column annotation labels Row annotation labels Point labels (e.g. sample names)

Column annotations

Instructions

Tauno Metsalu and Jaak Vilo: "Clustvis"

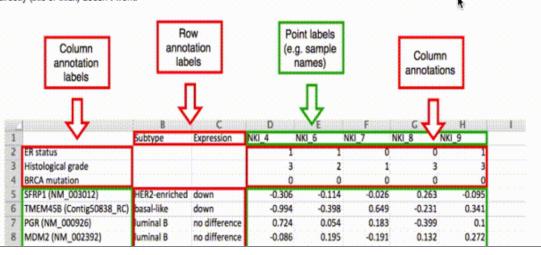
Interactivity of the plots Interpreting the output Mathematical basis ClustVis editions centering and scaling, method for calculating principal components and imputing missing values

- PCA options related to PCA plot
- · Heatmap options related to heatmap
- Export create link with current settings, download intermediate results

The idle timeout (the time when browser session ends if user is inactive) is set to 30 minutes from server side but this can be overridden by browser configuration. To save uploaded data and selected settings, you can use a button on the 'Export' tab, a link is given to recover the settings later. This can also be used to send a link to a collaborator to show the same view. There is no planned expiration time for the links, users can delete the settings if they are concerned about the privacy. Though, when version of ClustVis changes, old saved settings may not be fully compatible with the new version if e.g. there are some new features.

Data import

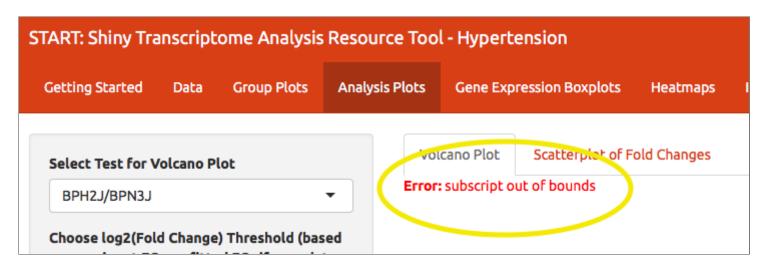
We aimed for a simple input data format. The numeric data matrix is situated in the bottom right corner, dimensions presented in rows and points in columns. Row labels and annotations are left from the matrix, column labels and annotations are above the matrix. Annotation labels are in the first row and column, respectively. Format of the input file is shown on the image below. Annotations are optional, data sets without annotations can be uploaded as well (on the example image, omitting rows 2-4 and/or columns B and C). When taking data from spreadsheet program (e.g. MS Excel), you can copy-paste the data to 'Paste data' box or export the data as delimited text file (ending with .csv or .tab) and then upload this file to ClustVis. Uploading Excel native files directly (.xls or .xlsx) doesn't work.



But Does It work?

Debugging Tips

Debugging is notoriously a pain here



Thanks, stack trace:

```
Warning in origRenderFunc():
 Ignoring explicitly provided widget ID "db8a6bc6ca6e"; Shiny doesn't use them
Warning in widgetFunc():
 renderDataTable ignores ... arguments when expr yields a datatable object; see ?re
nderDataTable
Joining, by = "unique_id"
Warning: Column `unique_id` joining character vector and factor, coercing into chara
cter vector
Warning: Error in [: subscript out of bounds
Stack trace (innermost first):
   85: rna_scatterplot [fun-analysisres.R#334]
   84: eval [server-analysisres.R#119]
    83: eval
    82: withProgress
   81: "plotly"::"ggplotly" [server-analysisres.R#115]
    80: func
    79: origRenderFunc
   78: output$scatterplot
    1: shiny::runApp
```

Did you update a package?

Good luck with that.

Old fashioned methods work best

Breakpoints and tracing:

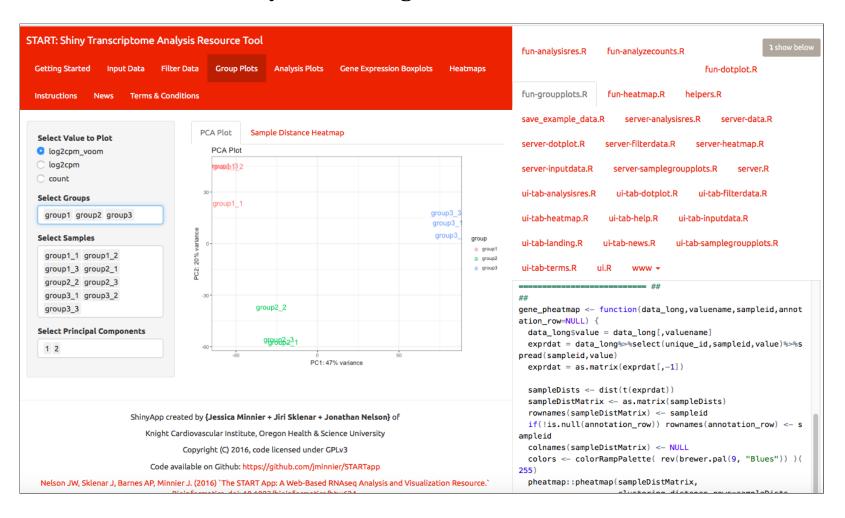
- browser() I didn't say it was fun
- print() every single function/observe statement gets a print

```
[1] "server-sampleplots-update-samples"
[1] "server-analysisres-update"
[1] "drawing volcano plot"
[1] "drawing scatterplot"
[1] "server-datafilter-update-filters"
[1] "server-sampleplots-update-samples"
[1] "server-datafilter-update-tests"
Warning in max(tmpfc, na.rm = T) :
 no non-missing arguments to max; returning -Inf
[1] "server-datafilter-update-expr"
Warning: Error in FUN: only defined on a data frame wi
les
Stack trace (innermost first):
   61: FUN
   60: lapply
   59: Summary.data.frame
   58: observerFunc [server-filterdata.R#76]
    1: shiny::runApp
ERROR: [on_request_read] connection reset by peer
ERROR: [on_request_read] connection reset by peer
ERROR: [on_request_read] connection reset by peer
```

New methods are also nice

runApp(..., display.mode="showcase")

But, doesn't show why something broke.

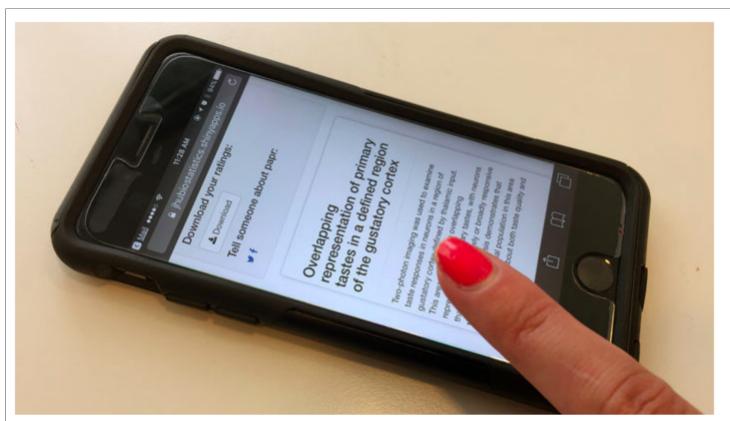


Random final tips:

- Test it first
 - Unit testing can save you so much headache
 - Run tests on example data
 - Create .Md file of test plots
- Security
 - Are users uploading sensitive data?
 - Security review of public apps vs commercial version
- Speed: Why so slow?
 - Rprofiler in Rstudio
 - Reduce start up time

Be Creative!

papr by Lucy D'Agostino McGowan, Nick Strayer, Jeff Leek



Papr lets you decide whether an abstract is "exciting," "boring," "probable," or "questionable."

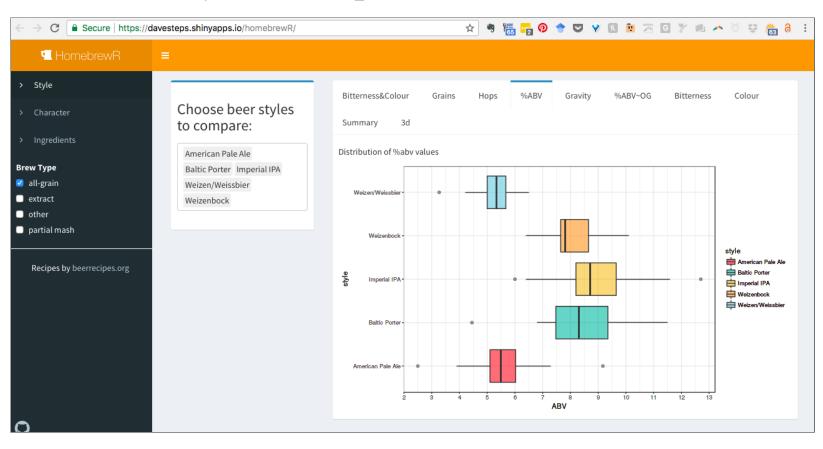
Science/AAAS

Great paper? Swipe right on the new 'Tinder for preprints' app

By Dalmeet Singh Chawla | Jun. 15, 2017, 5:00 PM

Have fun!

homebrewR by David Stephens



Thank you! Go forth and shiny!

Contact: @ datapointier

Github: https://github.com/jminnier/STARTapp

Slides available at http://bit.ly/shiny-ddtx

Code for slides available at

https://github.com/jminnier/presentation austin dataday201