



A  App to Analyze Bacterial Profiles and Aid the Discovery of New Antibiotics

Chase Clark
PhD Student
Murphy Lab

Department of Medicinal Chemistry and Pharmacognosy
Center for Biomolecular Sciences
University of Illinois at Chicago

```
# Run peakMatching pairwise
matched <- list(matched, lapply(steppedePairs, peakMatching))

### This version is much less RAM intensive, but we lose the groupings
peakMatching<-function(pairs)
{as.vector(unlist(lapply(lapply(seq_along(peakLists[pairs[1]][[1]]), function(x) {
    ppmVector <- ppmAcrossVector(peakLists[pairs[1]][[1]], 100)
    rtry<-which(peakLists[pairs[2]][[1]] <=
        peakLists[pairs[1]][[1]][x]+ppmVector[x] &
        peakLists[pairs[2]][[1]] >=
        peakLists[pairs[1]][[1]][x]-ppmVector[x])
    }
    ), function(z) length((z)))))}
```

add a comment

Questions Developer Jobs Tags Users Search...

Log In Sign Up

Spanning Apply() ...

Optimize Apply() While() in R

Related

How to sort a data frame by column(s)?

Using mapply() in R over rows, vs. columns

efficient way of coding for matching using



giphy.com

ChemicalData returns the correct result for "Sulfuric" but an empty list for "Sulfuric Acid". Why?

No space left, even though I have an untouched 990 GB partition

What is this methodology called?

What are good words to refer to the condition of objects?

If I generate a random symmetric matrix, what's the chance it is positive definite?

Dealing with aggressive student suspected to be cheating

During single-engine taxi, how is the asymmetric

txt = ' data1 data2 data3 data4
-0.710003 -0.714271 -0.709946 -0.713645
-0.710458 -0.715011 -0.710117 -0.714157
-0.71071 -0.714048 -0.710235 -0.713515
-0.710255 -0.713991 -0.709722 -0.713972'

Data_1 <- read.table(text=txt, header=TRUE)

txt = ' dataA dataB dataC dataD
-0.71097 -0.714059 -0.70928 -0.714059
-0.710343 -0.714576 -0.709338 -0.713644'

Data_2 <- read.table(text=txt, header=TRUE)

Code

```
check_inbetween <- function(x,y){  
  sapply(x, function(i) (i > y[1] & i < y[2]))  
}  
  
inbetween_matrix <- mapply(check_inbetween, Data_1, Data_2)  
  
inbetween_matrix  
# data1 data2 data3 data4  
# [1,] FALSE FALSE FALSE TRUE  
# [2,] TRUE FALSE FALSE FALSE  
# [3,] TRUE FALSE FALSE FALSE  
# [4,] FALSE FALSE FALSE TRUE
```

share improve this answer answered Jul 7 '17 at 1:04 Parfait 34k 7 23 55

Thanks @Parfait, I guess even `sweep()` can be used - Chetan Arvind Patil Jul 7 '17 at 1:18

Possibly, I never used `sweep()`. And that implementation has an interesting nested sweep. My attempt seems to not recycle the function but only runs on first columns of datasets: `sweep(as.matrix(Data_1), 1, as.matrix(Data_2), FUN = function(x,y) (x > y[1] & x < y[2]), check.margin = FALSE)` - Parfait Jul 7 '17 at 1:58

add a comment

Thanks Parfait!



answered Jul 7 '17 at 1:04



Parfait

34k ⚡ 7 ● 23 ● 55

Quick Outline

- Why we are trying to discover new antibiotics
- It's not easy!!
- A new pipeline to prioritize bacteria for antibiotic discovery
- What is Shiny?
- Brief overview of IDBac
- Some tips about Shiny

On the Precipice of a Pre-Antibiotic Era



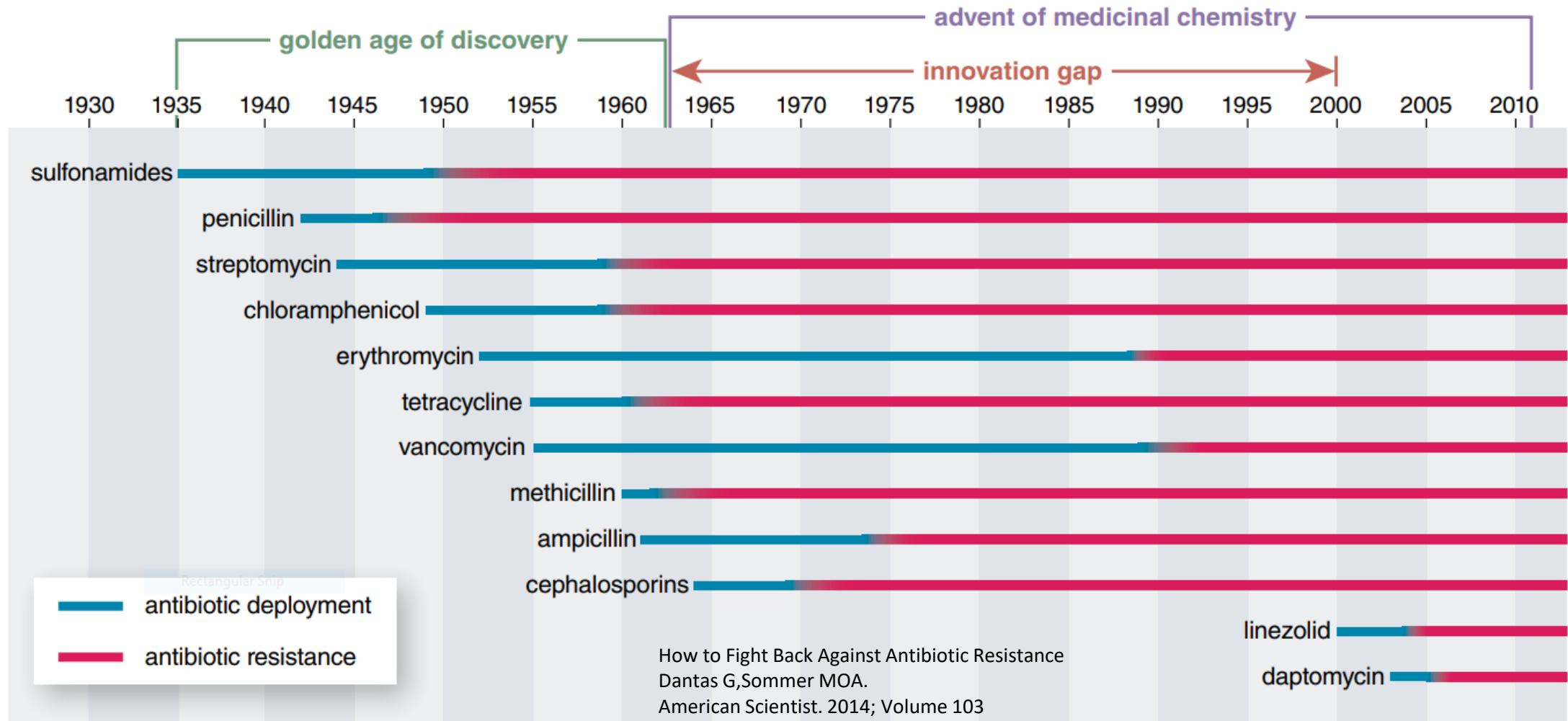
**The world is running out of antibiotics ,
WHO report confirms**
News release 20 September 2017 | Geneva



Antibiotics are Like R Packages...

New Packages are Needed to Solve Difficult Problems

But only a couple new packages since the 60's!!!!!!



Ironically, the most prolific source of antibiotics has been... bacteria!



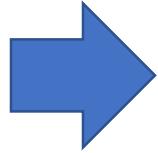
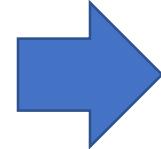
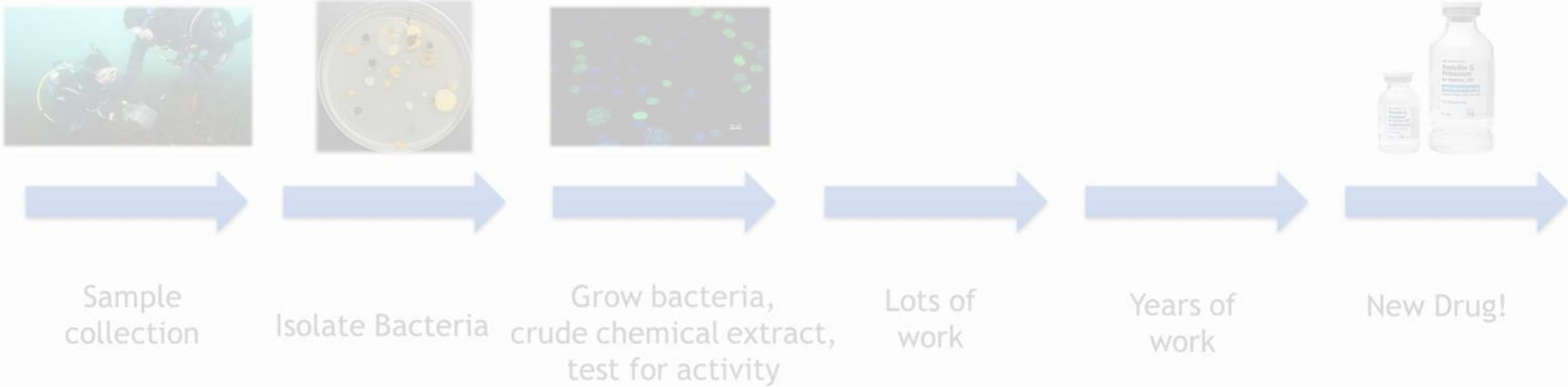
Fig. 1 – Key findings and dates of antibiotics. Highlights of the Streptomyces.

FROM THE LAB

SCIENCE
MUSEUM



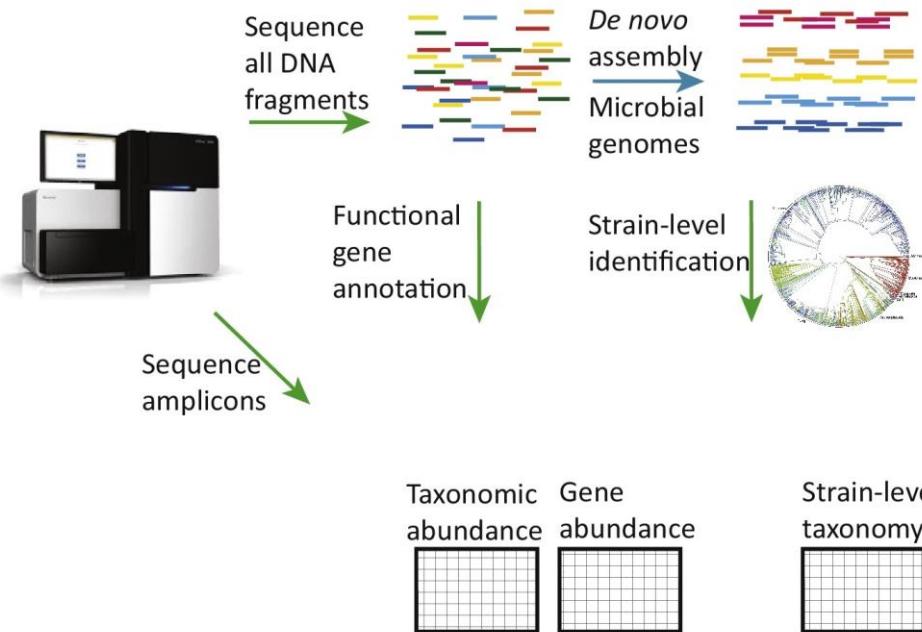
Current Methods: Are Like Finding a Needle in a Hay Stack... While Blindfolded



Each Bacterium Studied Requires Significant Time and Monetary Investment

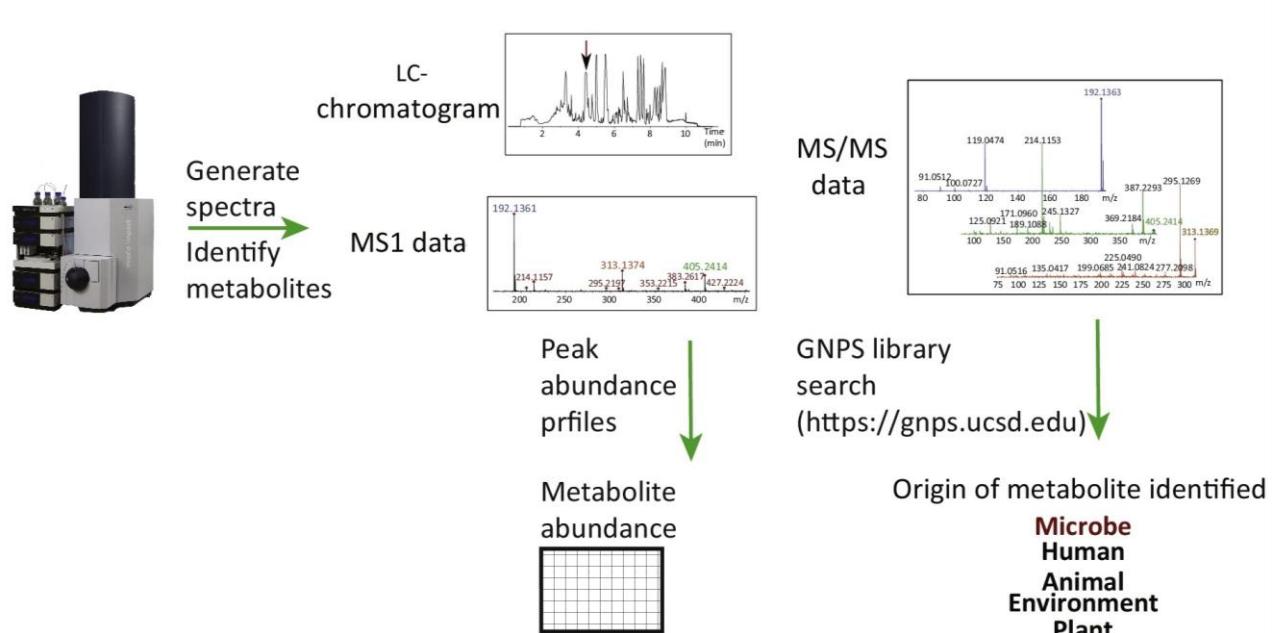
Genomics

Amplicon or shotgun metagenomic sequencing



Low-Throughput Metabolomics

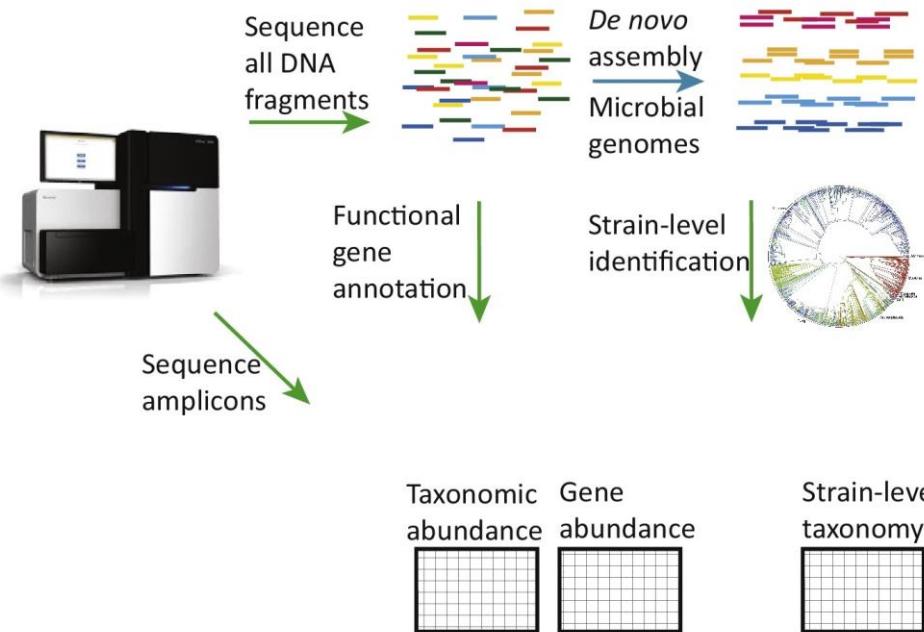
UPLC-Q-TOF MS/MS generation



Each Bacterium Studied Requires Significant Time and Monetary Investment

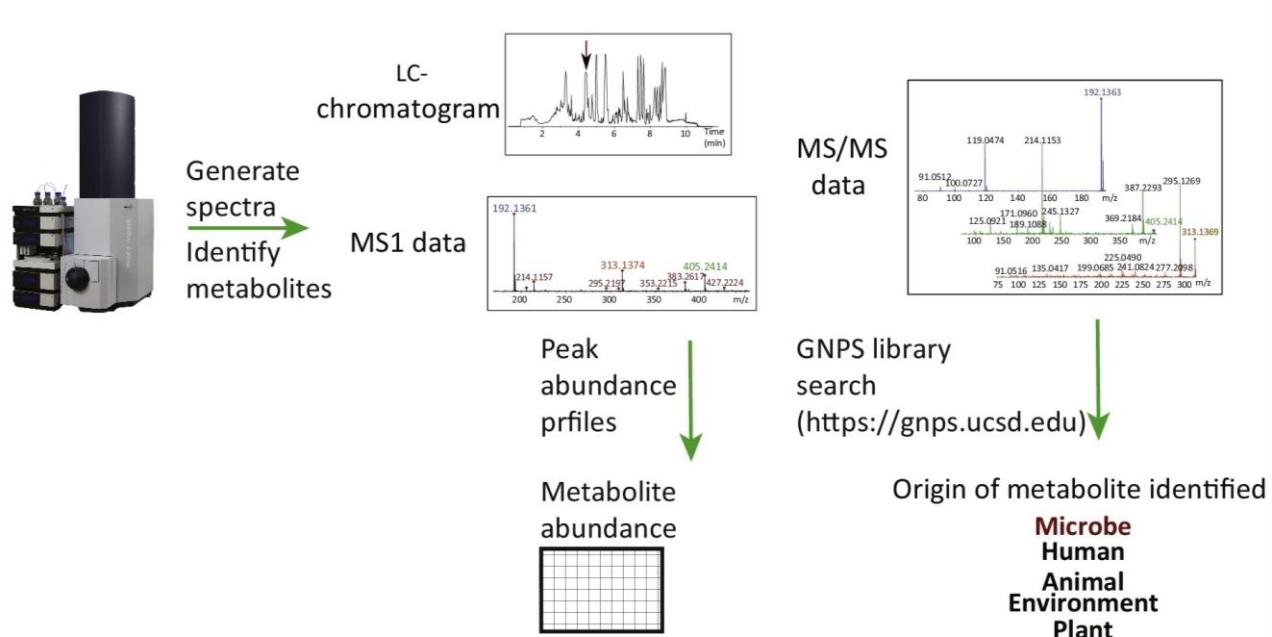
Genomics

Amplicon or shotgun metagenomic sequencing



Low-Throughput Metabolomics

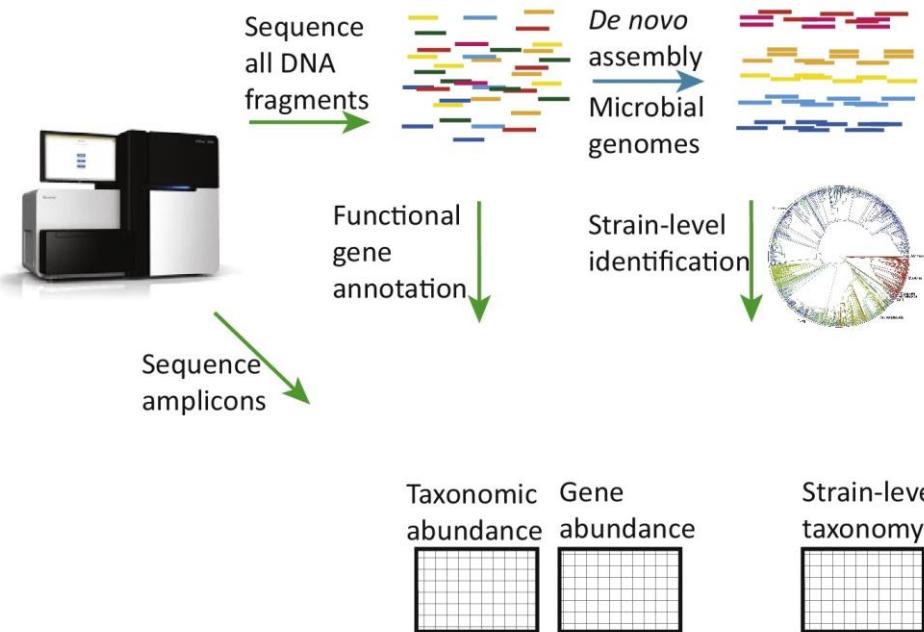
UPLC-Q-TOF MS/MS generation



Each Bacterium Studied Requires Significant Time and Monetary Investment

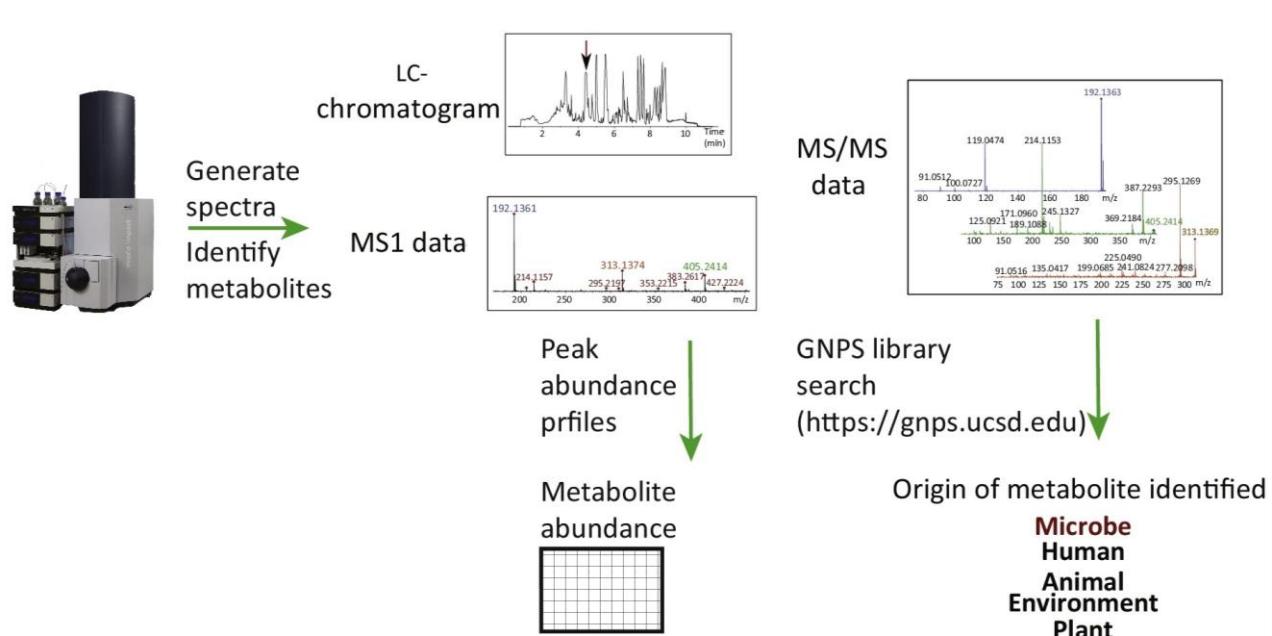
Genomics

Amplicon or shotgun metagenomic sequencing



Low-Throughput Metabolomics

UPLC-Q-TOF MS/MS generation



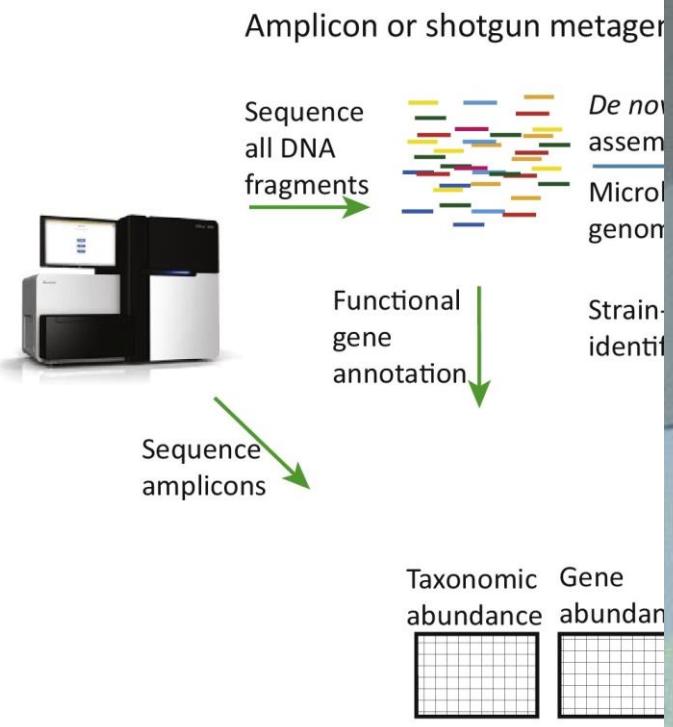
Metcalf, Jessica L. et al.

Trends in Biotechnology , Volume 35 , Issue 9 , 814 – 823

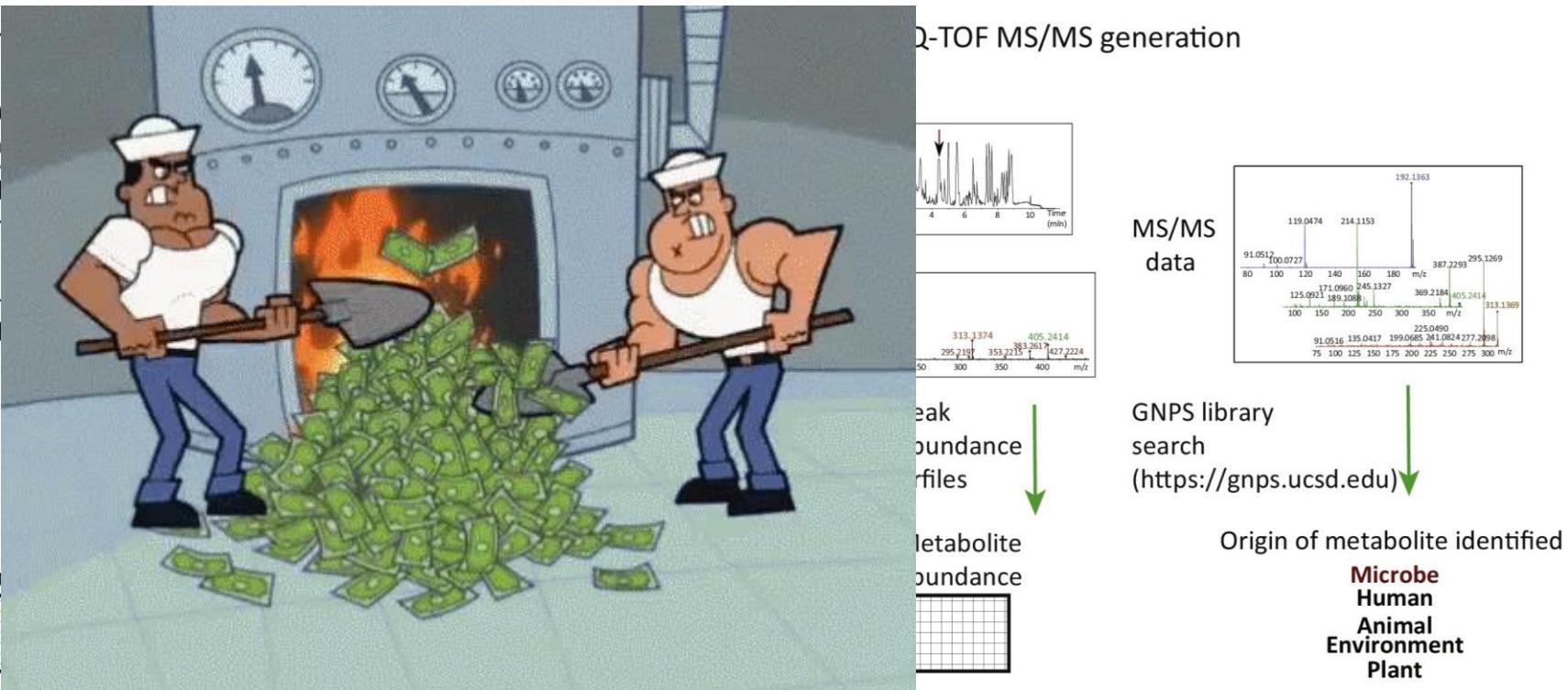
<http://dx.doi.org/10.1016/j.tibtech.2017.03.006>

Each Bacterium Studied Requires Significant Time and Monetary Investment

Genomics



Low-Throughput Metabolomics



Labs have trouble prioritizing which bacteria to study.

- Which orange colonies are actually the same?
 - Visual cues are not enough!
- Do they all produce the same chemistry?
 - Visual cues are not enough!



Our Lab Recently Developed IDBac: A High Throughput Method to Rapidly Distinguish Bacterial Phylogeny and Functional Metabolism

The screenshot shows a bioRxiv preprint page. At the top left is the CSHL logo and the bioRxiv beta logo with the tagline "THE PREPRINT SERVER FOR BIOLOGY". The top navigation bar includes links for HOME, ABOUT, SUBMIT, ALERTS / RSS, and CHANNELS. A search bar with a magnifying glass icon and an "Advanced Search" link are also present. The main content area displays a preprint titled "Coupling MALDI-TOF mass spectrometry protein and specialized metabolite analyses to rapidly discriminate bacterial function" by Chase M. Clark, Maria S. Costa, Laura M. Sanchez, Brian T. Murphy. The doi is listed as <https://doi.org/10.1101/215350>. Below the title, it says "This article is a preprint and has not been peer-reviewed [what does this mean?].". The abstract section is currently selected. Other tabs include Info/History, Metrics, and Supplementary material. There is a "Preview PDF" button. To the right of the abstract, there are download options (PDF, Email), sharing links (Twitter, Facebook, G+), and citation tools. The subject area is listed as Microbiology. A sidebar on the right lists Subject Areas: Animal Behavior and Cognition, Biochemistry, and Biocomputing.

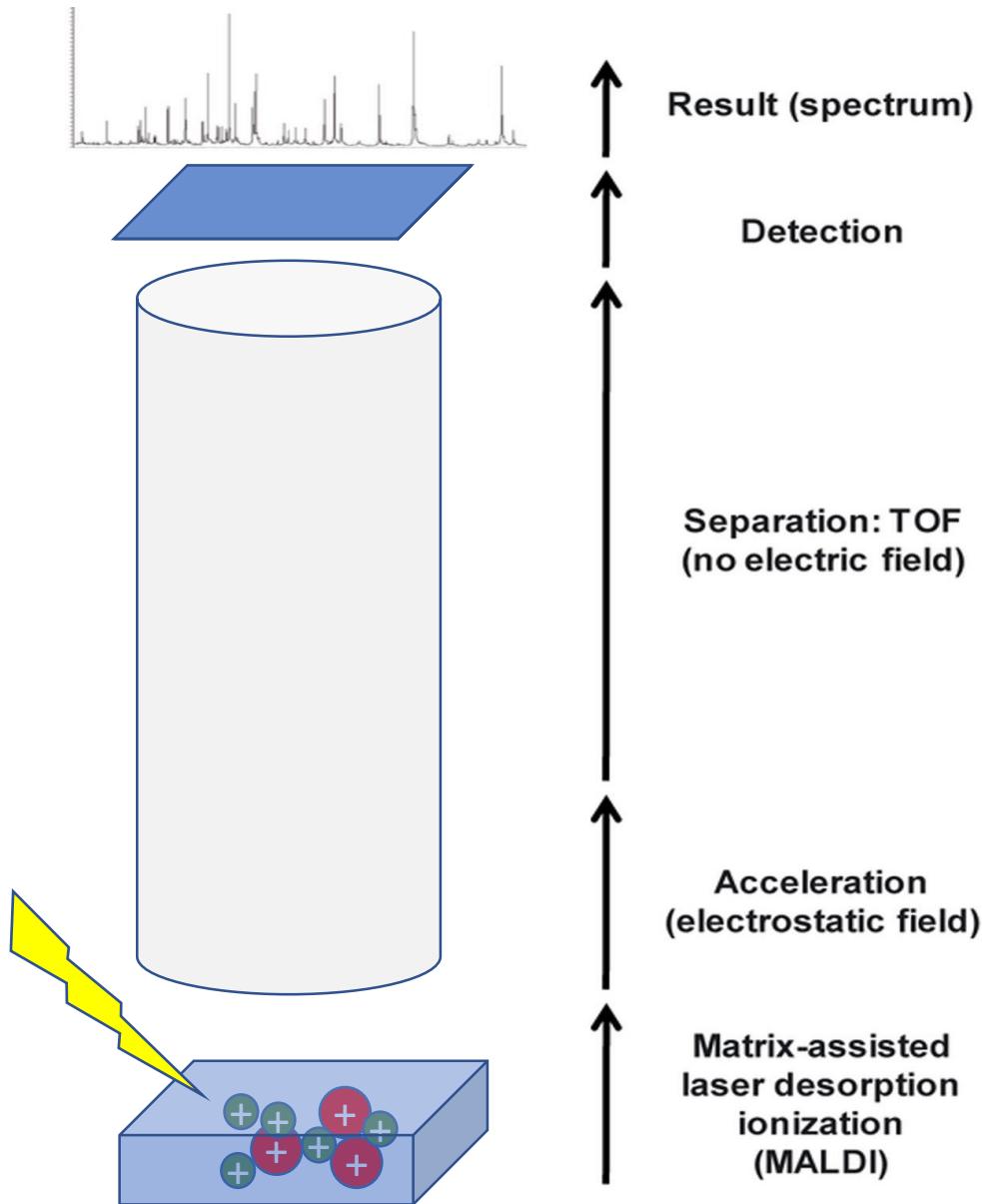
chasemc.github.io/IDBac

<https://github.com/chasemc/IDBac>

MALDI-TOF: A Rapid and Simple Way to Weigh Molecules From Bacteria



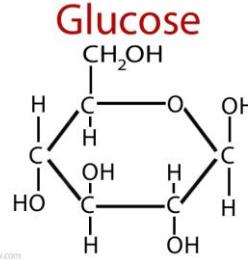
Up to 384 bacteria in one experiment



We're Just Weighing Molecules!

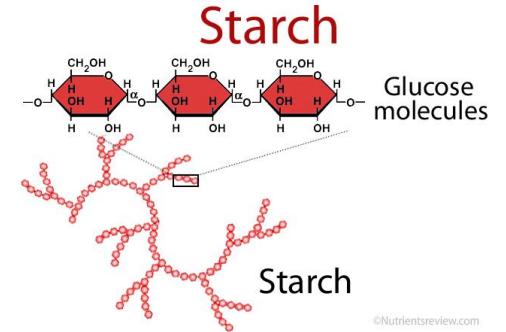


Light Molecules



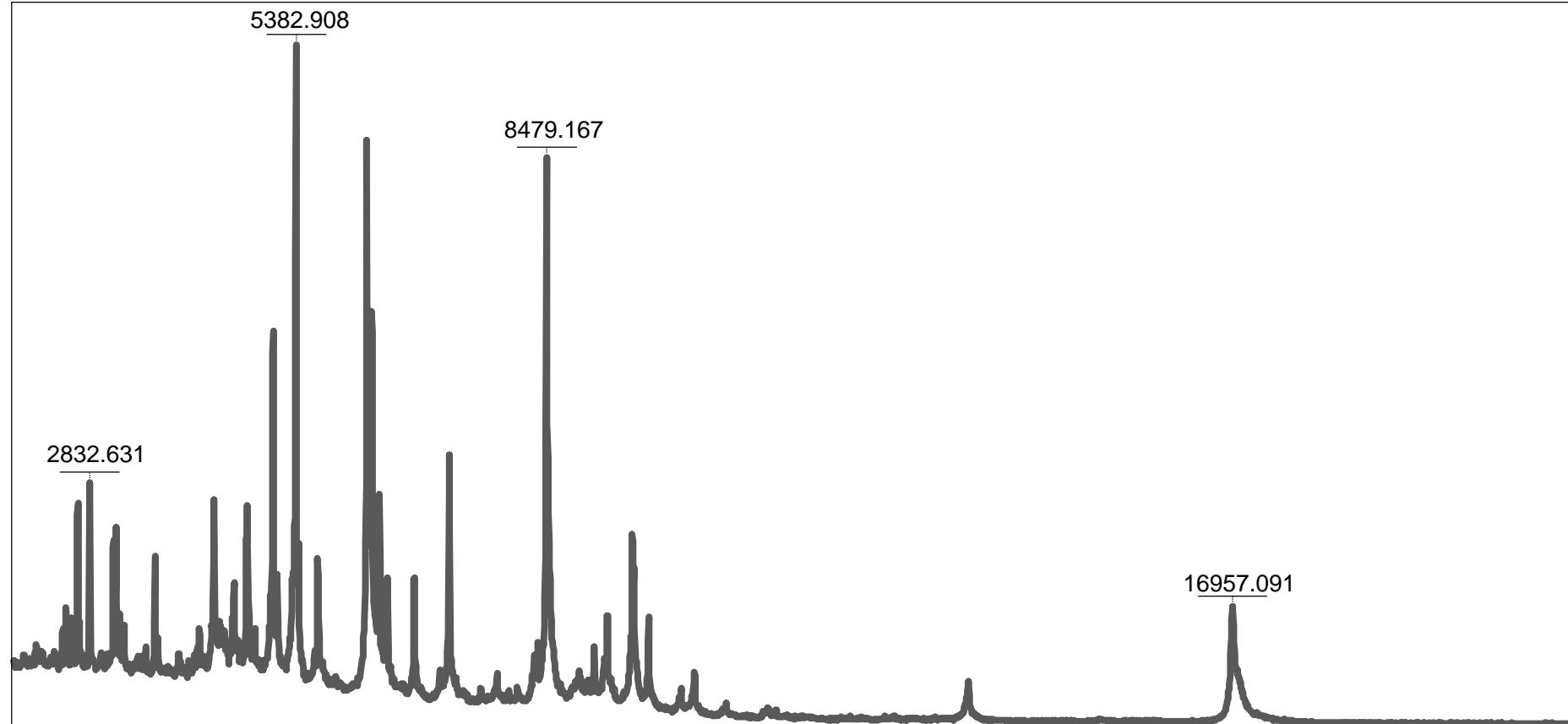
©Nutrientsreview.com

Heavy Molecules



©Nutrientsreview.com

We're Just Weighing Molecules!

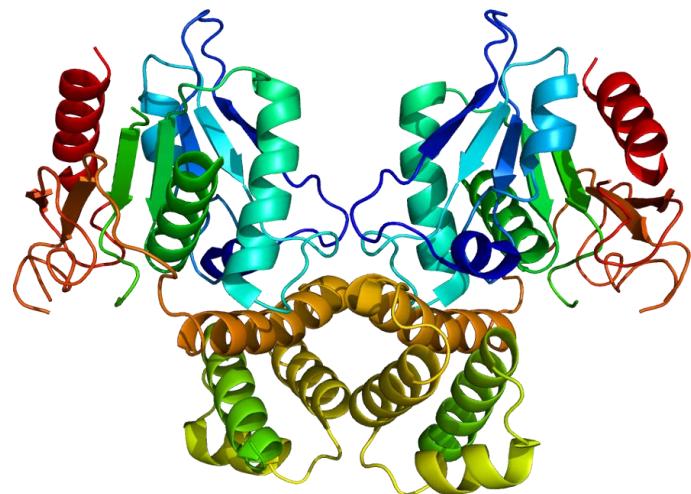


```
plot(readRDS(list.files(pattern="SummedProteinSpectra.rds")[[1]]), xlim=c(2000,12000))
```

We Weigh Two Types of Molecules = Two Spectra/Sample

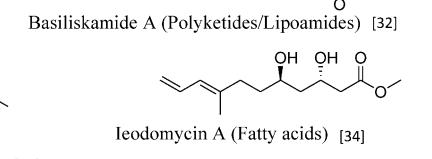
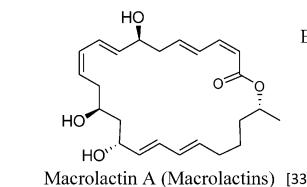
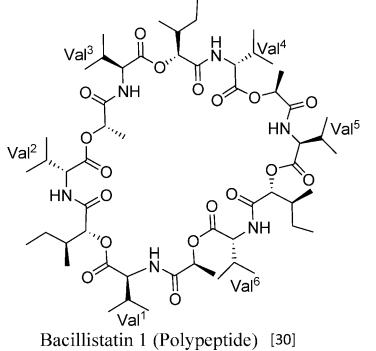
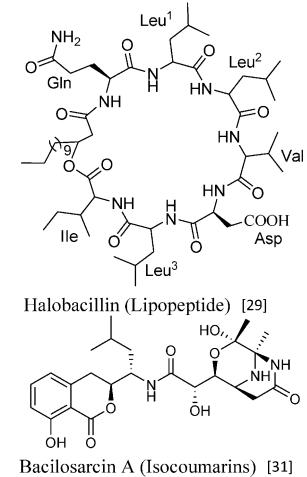
Protein Data (Identify Bacteria)

- Big molecules that are evolutionarily stable.



Specialized Metabolite Data (Potential Drugs)

- Small molecules that provide a selective advantage.

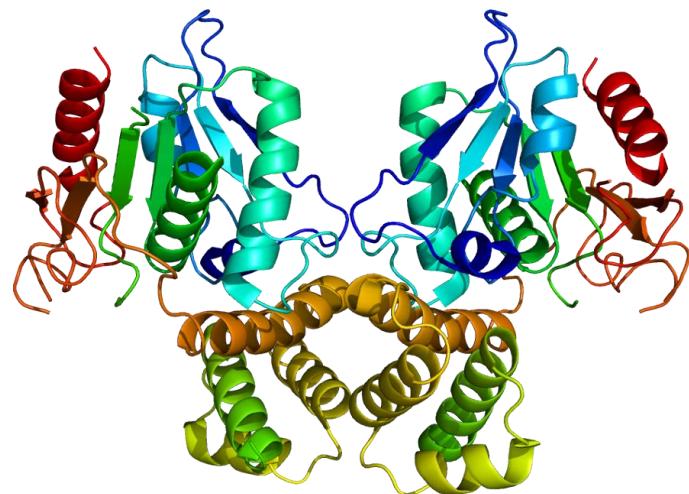


Basiliskamide A (Polyketides/Lipoamides) [32]

We Weigh Two Types of Molecules = Two Spectra/Sample

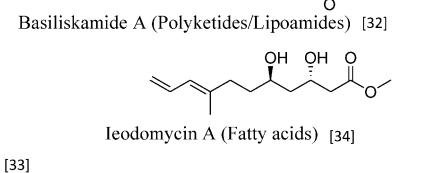
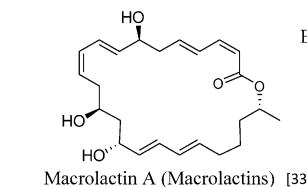
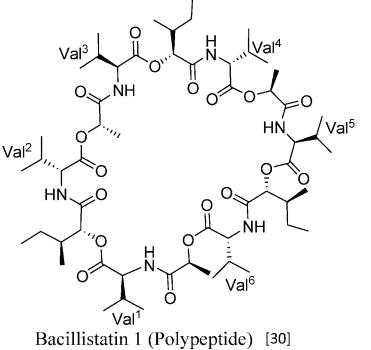
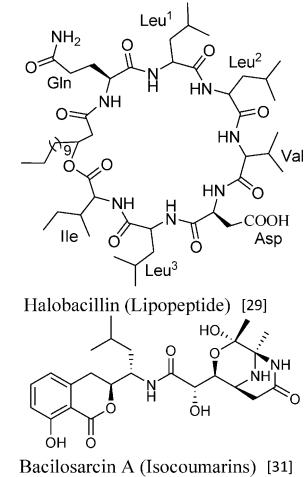
Protein Data (Identify Bacteria)

- Big molecules that are evolutionarily stable.



Specialized Metabolite Data (Potential Drugs)

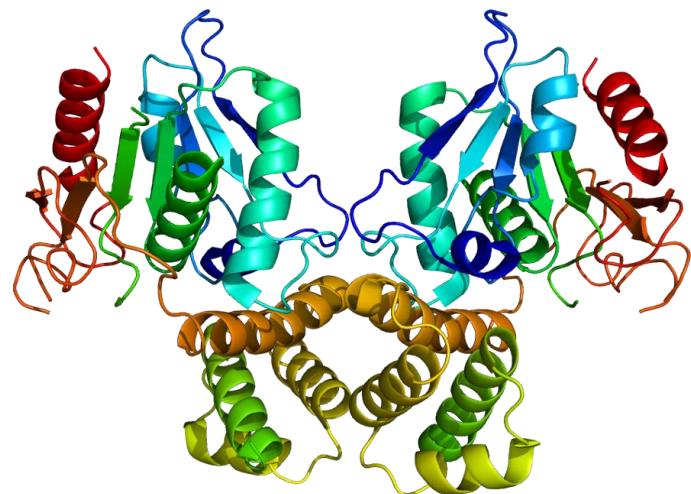
- Small molecules that provide a selective advantage.



We Weigh Two Types of Molecules = Two Spectra/Sample

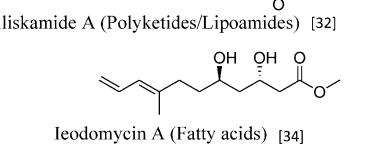
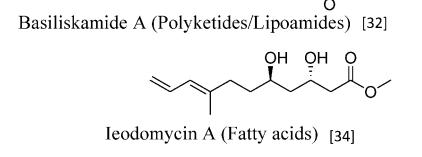
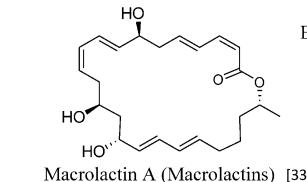
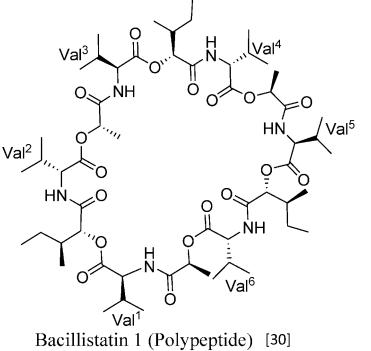
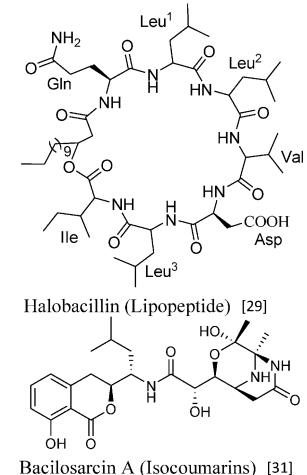
Protein Data (Identify Bacteria)

- Big molecules that are evolutionarily stable.



Specialized Metabolite Data (Potential Drugs)

- Small molecules that provide a selective advantage.



We Take Two Measurements:

- Big molecules
- Small Molecules



Data processing becomes
a major bottleneck

Not “Big Data”...But Not Small Data

$(384 \text{ spots} \times 1 \text{ Protein Data Set}) +$
 $(384 \text{ spots} \times 1 \text{ Small Molecule Data Set }) =$
768 Data Files per Run

	Small Mol Total	Protein Total
Instrument Data Format	774 MB	156 MB
mzML format	1.43 GB	251 MB

For context, we aren't dealing with just one diversity plate ->



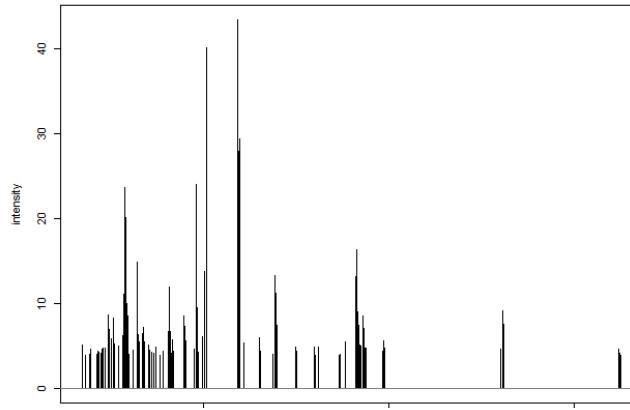
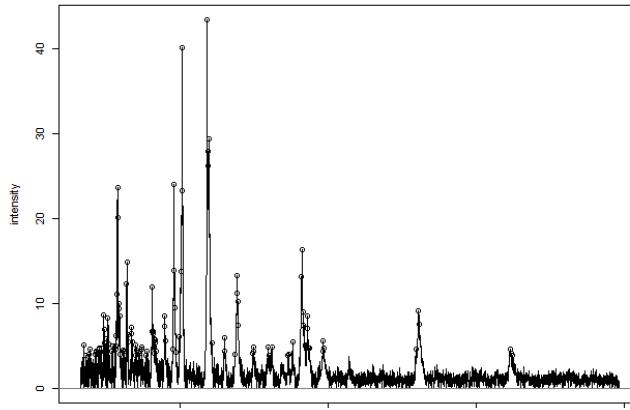
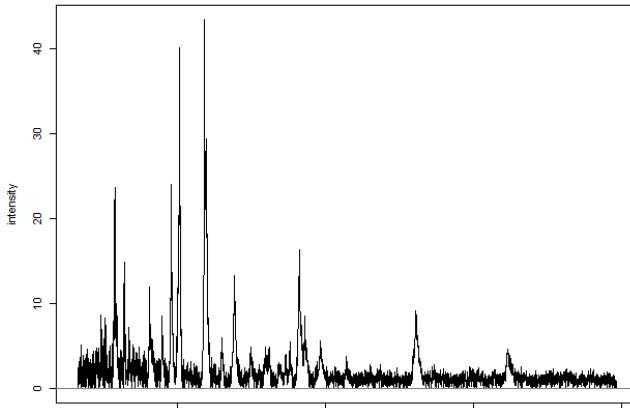
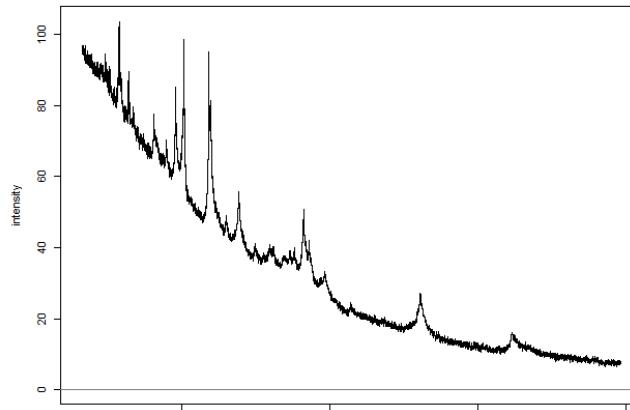
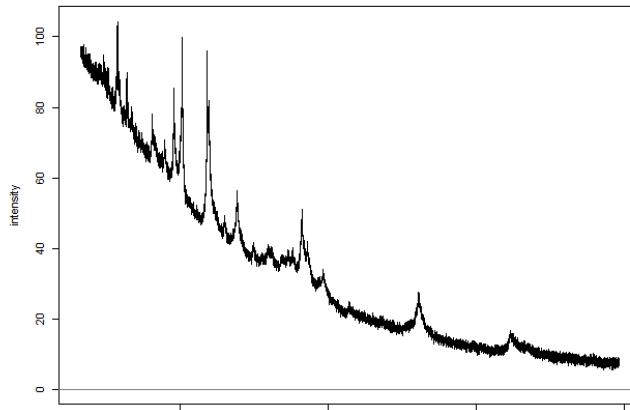
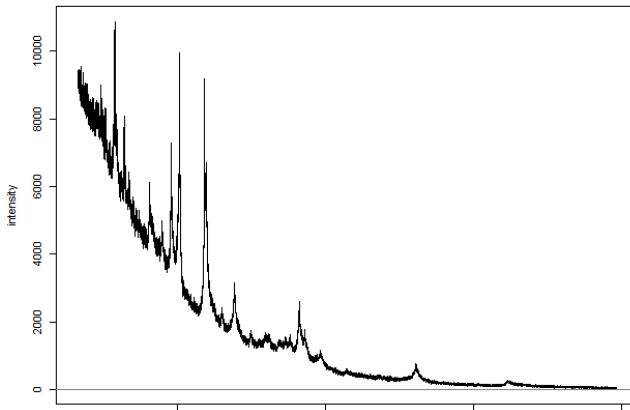
We are dealing with hundreds of plates. That means thousands of bacteria!



Automated Processing of MALDI Protein Spectra in R

Code here: <https://github.com/chasemc/Playground/blob/master/RshinyPres/README.md#spectra-processing>

1. Raw Spectrum
2. Intensity Normalization
3. Smoothing
4. Baseline Correction
5. Peak Detection
6. Extract Peak Data

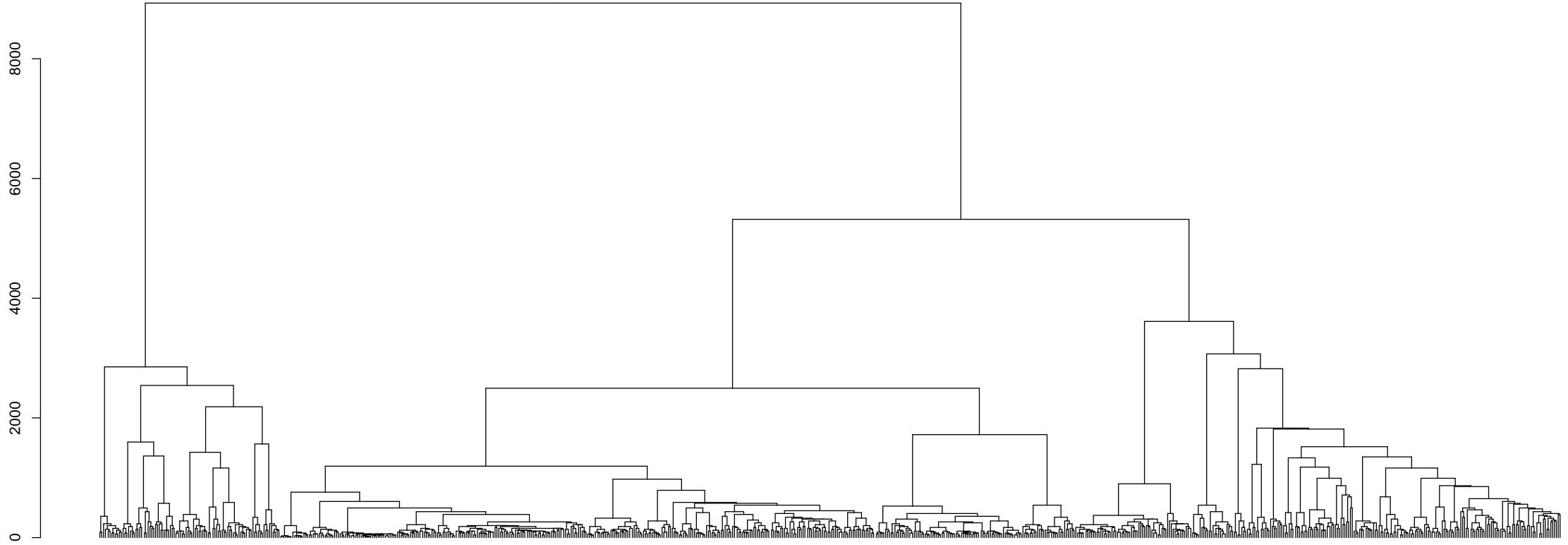


Most of these are from the MALDIquant package by Sebastian Gibb

Some of the Analyses Offered in the IDBac Shiny App... Real Data, Real Results

We're almost to the Shiny Part!

1) Hierarchical Clustering of Protein Data



Sub-sample of ~ 750 isolates

1) Hierarchical Clustering of Protein Data = Type of Bacteria



Sub-sample of ~ 750 isolates

Problem:

Same Species of Bacteria Can Produce Different Chemicals

Problem:

Same Species of Bacteria Can Produce Different Chemicals

That's why we collect two types of data!

Analyzing Small Molecule Data

Metabolite Association Network

Analyzing Small Molecule Data

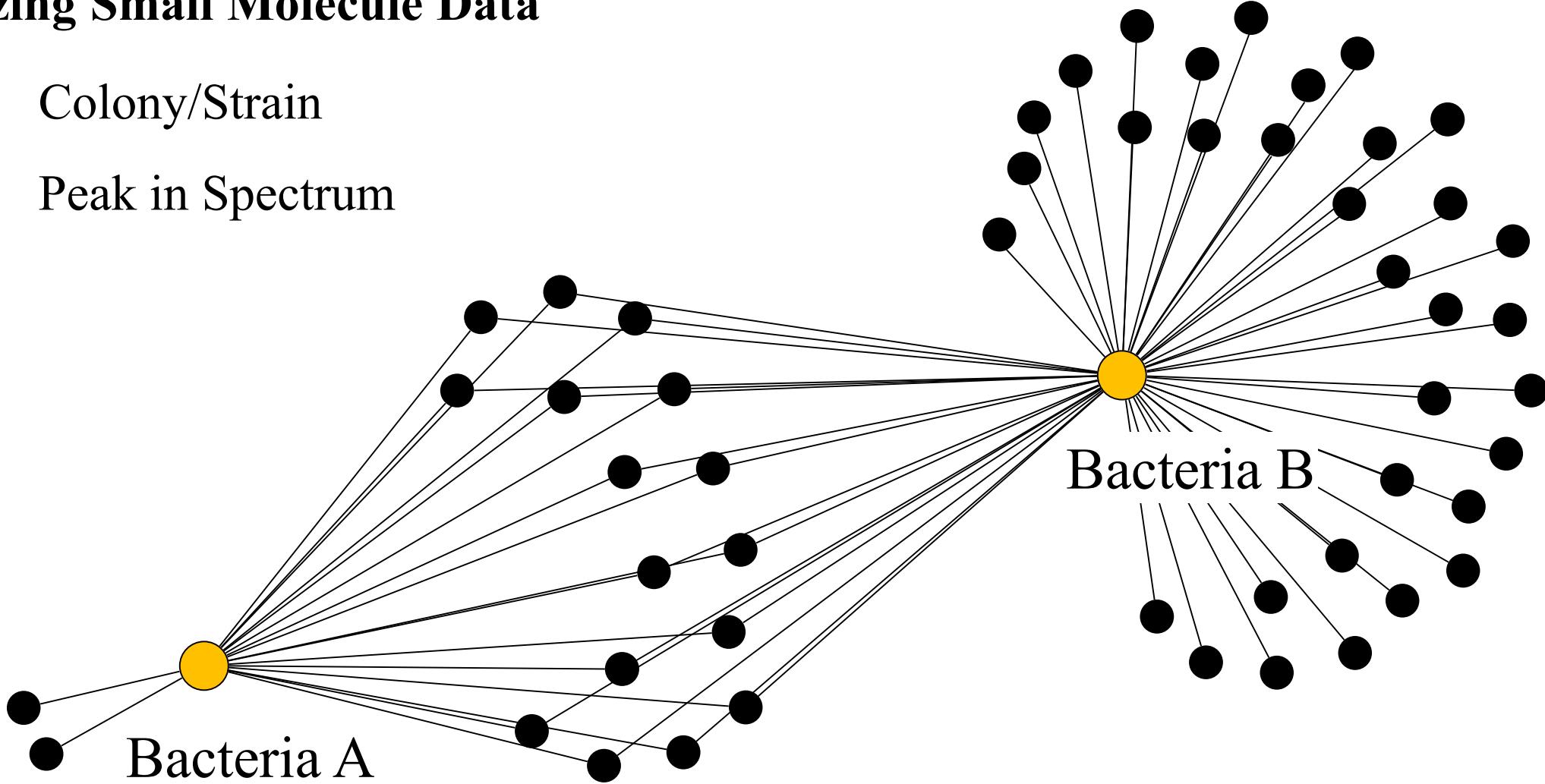
● Colony/Strain

● Peak in Spectrum

Metabolite Association Network

Analyzing Small Molecule Data

-  Colony/Strain
 -  Peak in Spectrum

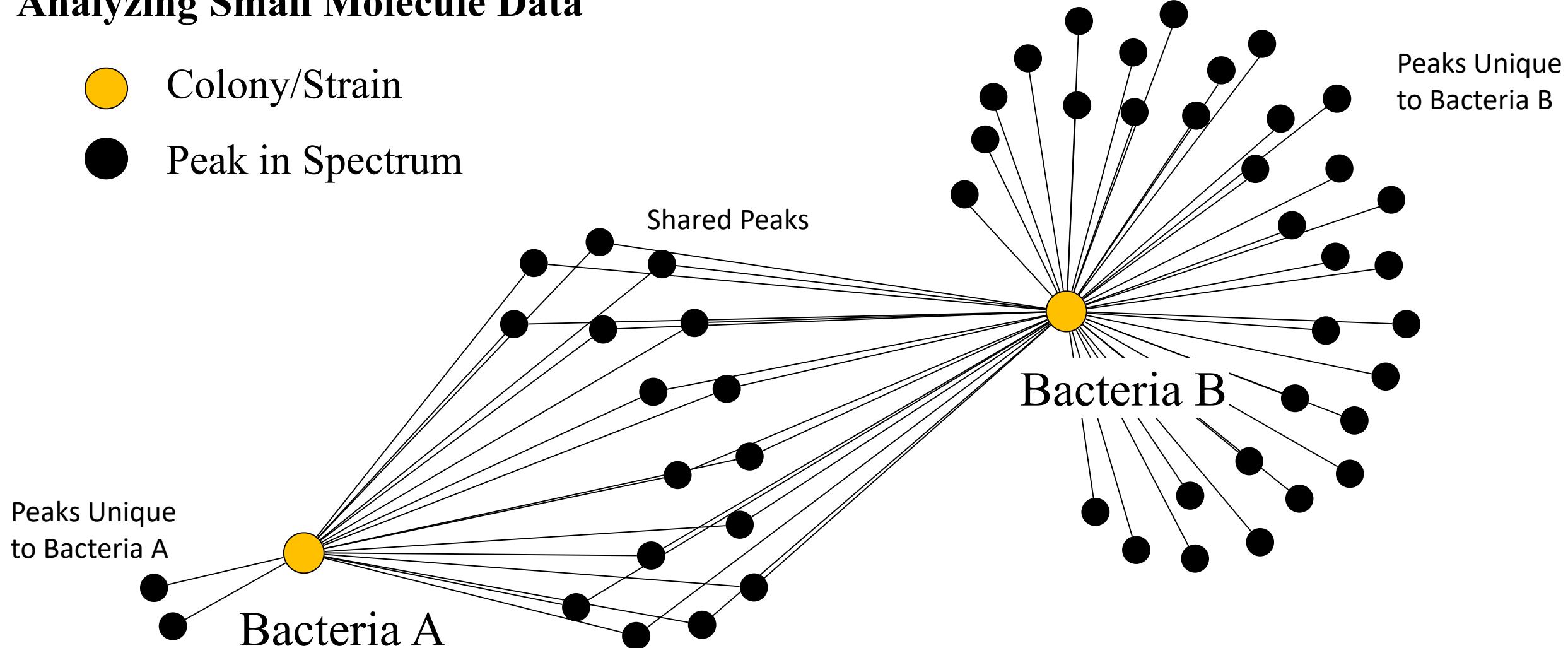


Metabolite Association Network

Analyzing Small Molecule Data

Colony/Strain

Peak in Spectrum

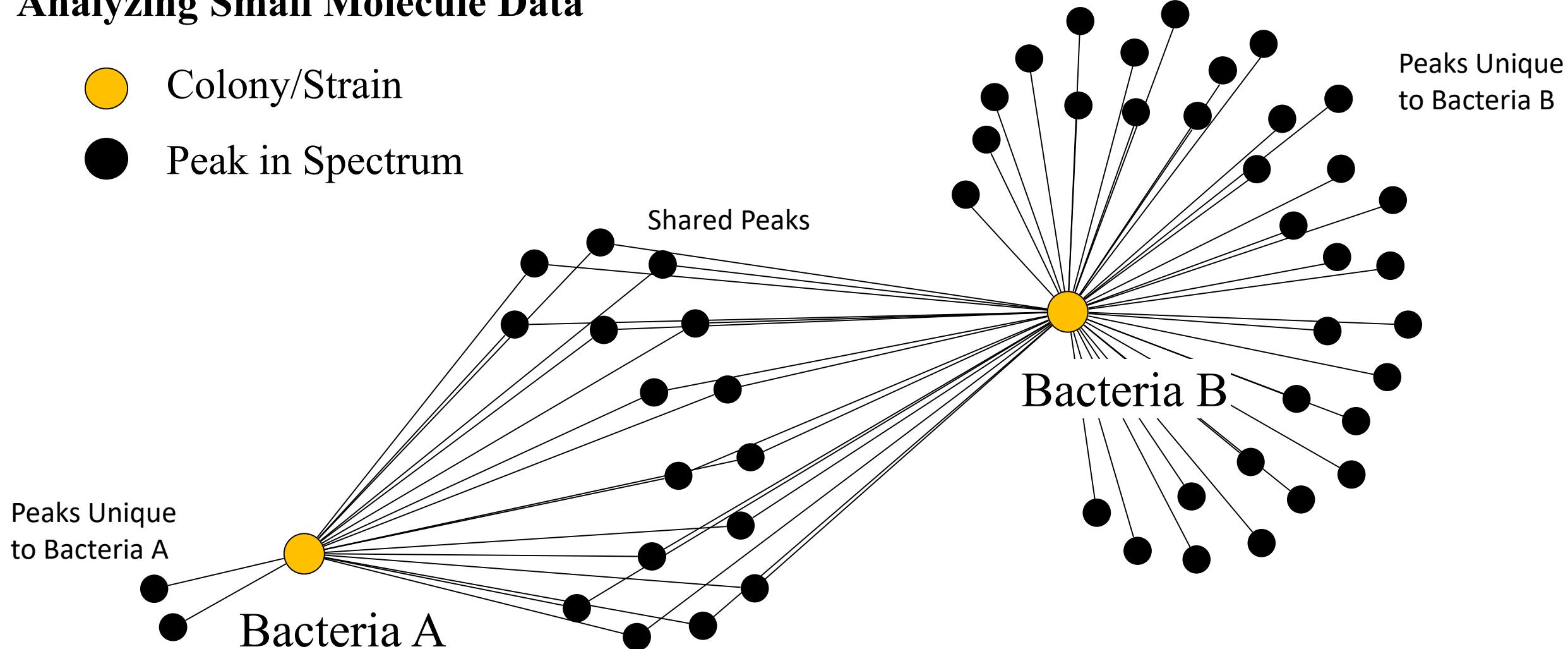


Metabolite Association Network

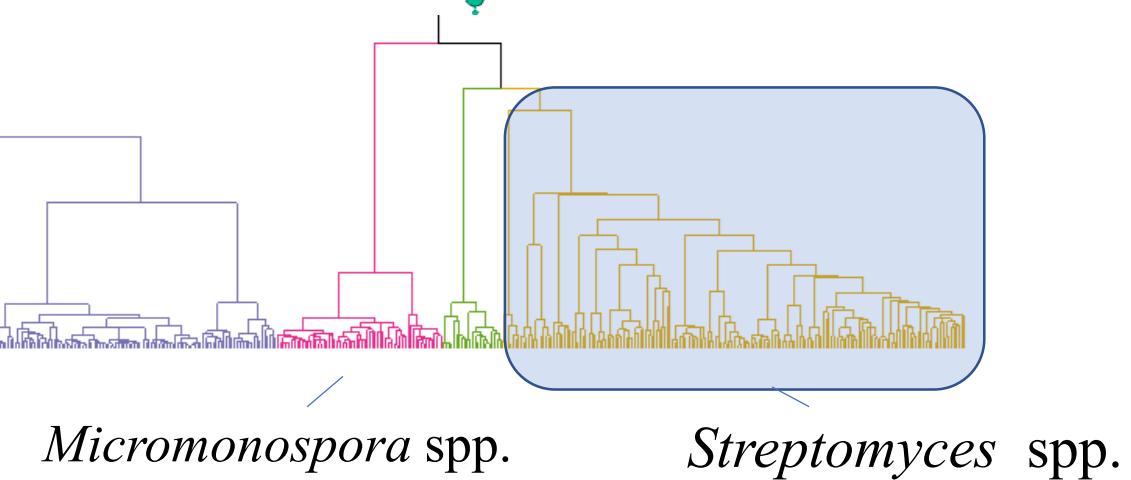
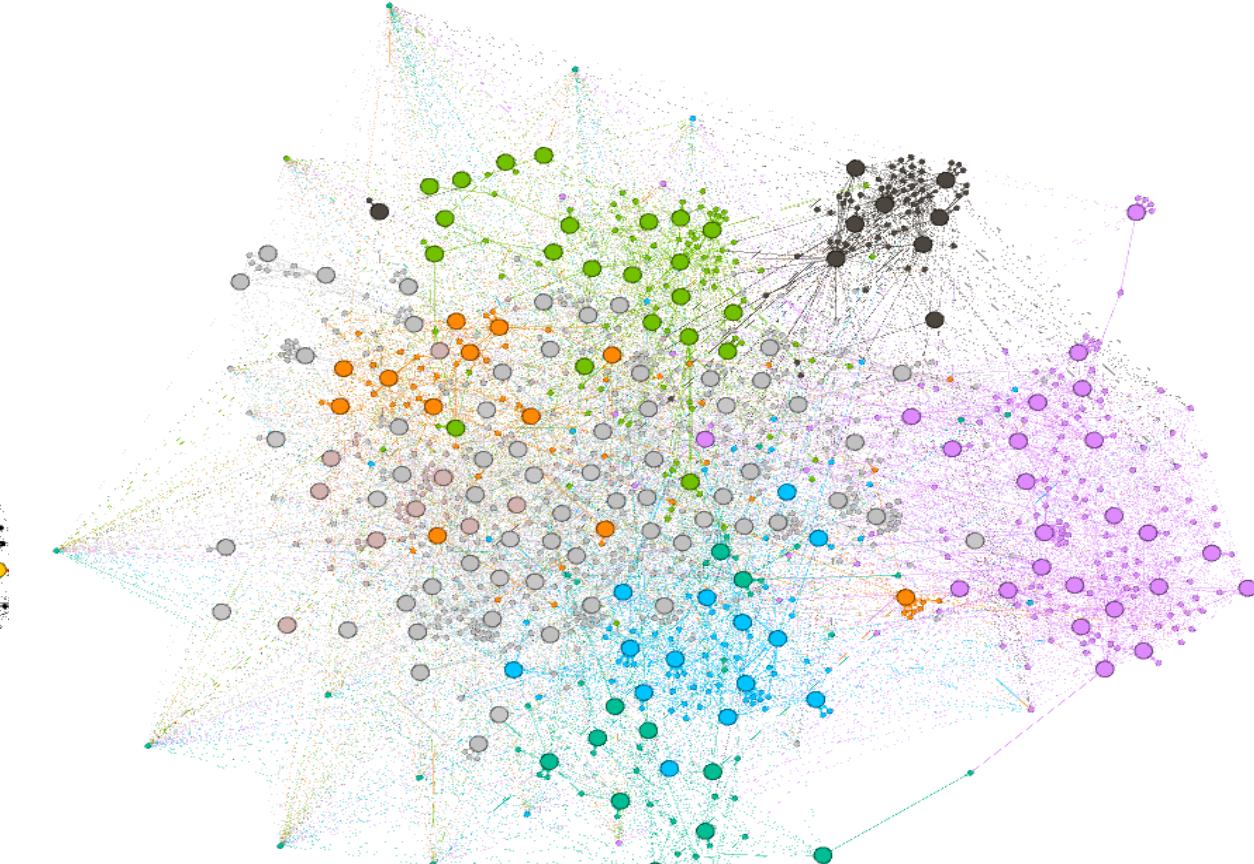
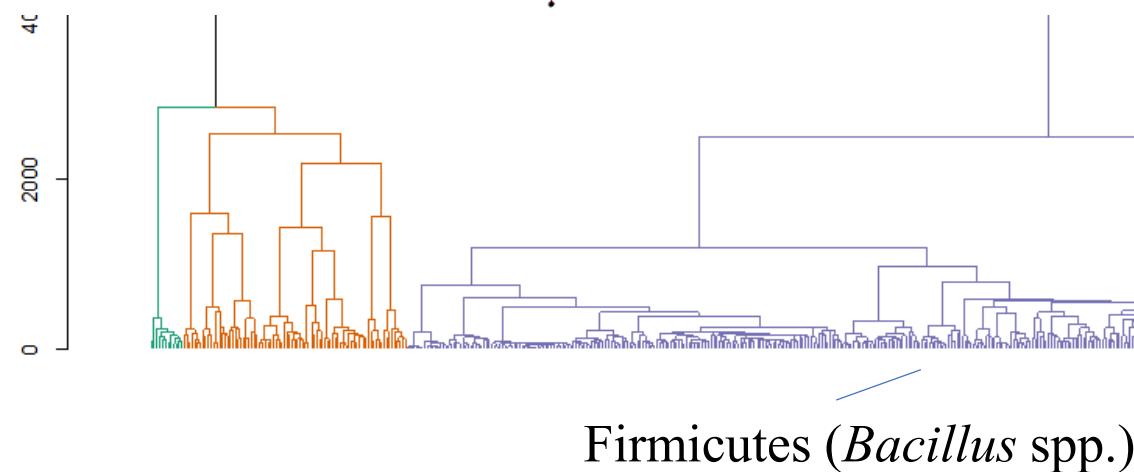
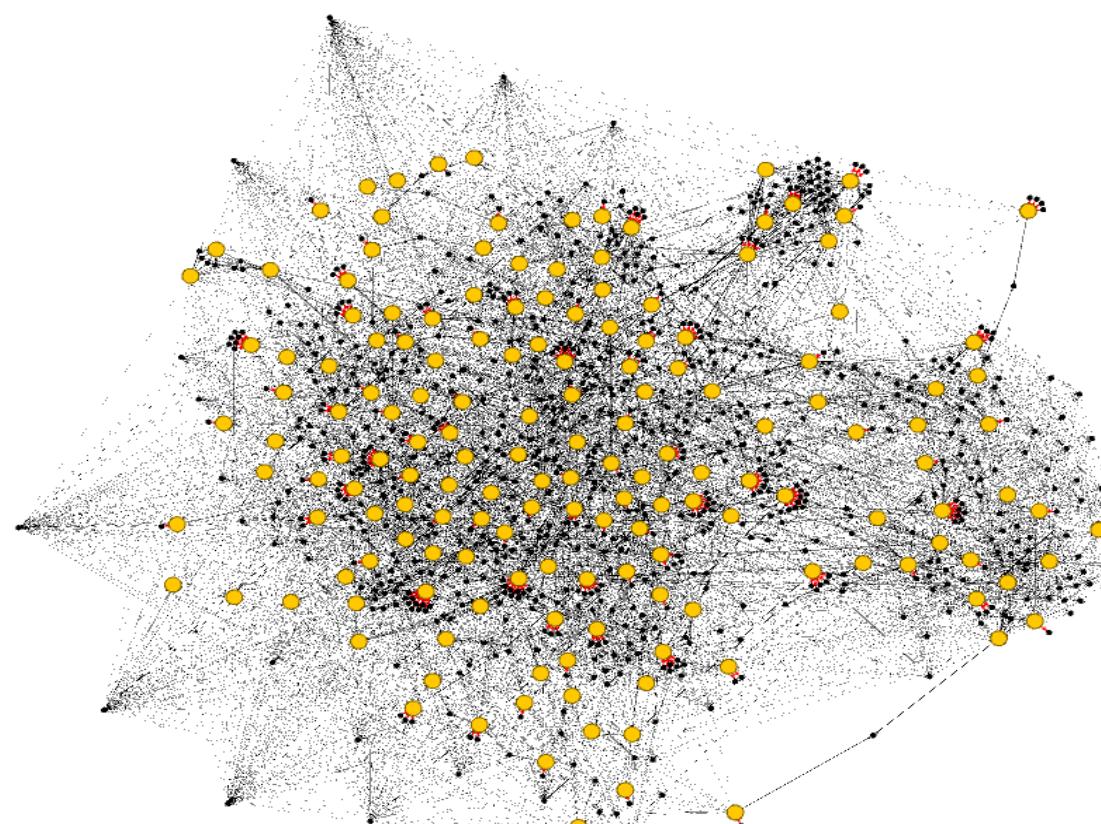
Analyzing Small Molecule Data

Colony/Strain

Peak in Spectrum



Metabolite Association Network



“So Chase,
I thought you used Shiny?”



Shiny was created and is currently maintained by Winston Chang/ RStudio

Maintainer: Winston Chang <winston@rstudio.com>

Winston Chang [aut, cre],
Joe Cheng [aut],
JJ Allaire [aut],
Yihui Xie [aut],
Jonathan McPherson [aut],
RStudio [cph],
jQuery Foundation [cph] (jQuery library and jQuery UI library),
jQuery contributors [ctb, cph] (jQuery library; authors listed in
inst/www/shared/jquery-AUTHORS.txt),
jQuery UI contributors [ctb, cph] (jQuery UI library; authors
listed in
inst/www/shared/jqueryui/AUTHORS.txt),
Mark Otto [ctb] (Bootstrap library),
Jacob Thornton [ctb] (Bootstrap library),
Bootstrap contributors [ctb] (Bootstrap library),
Twitter, Inc [cph] (Bootstrap library),
Alexander Farkas [ctb, cph] (html5shiv library)...

Scott Jehl [ctb, cph] (Respond.js library),
Stefan Petre [ctb, cph] (Bootstrap-datepicker library),
Andrew Rowls [ctb, cph] (Bootstrap-datepicker library),
Dave Gandy [ctb, cph] (Font Awesome font),
Brian Reavis [ctb, cph] (selectize.js library),
Kristopher Michael Kowal [ctb, cph] (es5-shim library),
es5-shim contributors [ctb, cph] (es5-shim library),
Denis Ineshin [ctb, cph] (ion.rangeSlider library),
Sami Samhuri [ctb, cph] (Javascript strftime library),
SpryMedia Limited [ctb, cph] (DataTables library),
John Fraser [ctb, cph] (showdown.js library),
John Gruber [ctb, cph] (showdown.js library),
Ivan Sagalaev [ctb, cph] (highlight.js library),
R Core Team [ctb, cph] (tar implementation from R)

A photograph of a man with light brown hair and sunglasses, wearing a white button-down shirt. He is looking upwards and slightly to the left with a thoughtful expression. His right hand is raised to his forehead, with his fingers partially covering his eyes. The background is dark and out of focus.

LET'S MAKE YOUR CODE

A LITTLE MORE SHINY

LET'S MAKE YOUR CODE

- Shiny apps are contained in a single script called app.R
- The script app.R lives in a directory and the app can be run with runApp("~/newdir")
- app.R has three components:

```
# A user interface object---
ui <- fluidPage( ) #Controls the layout

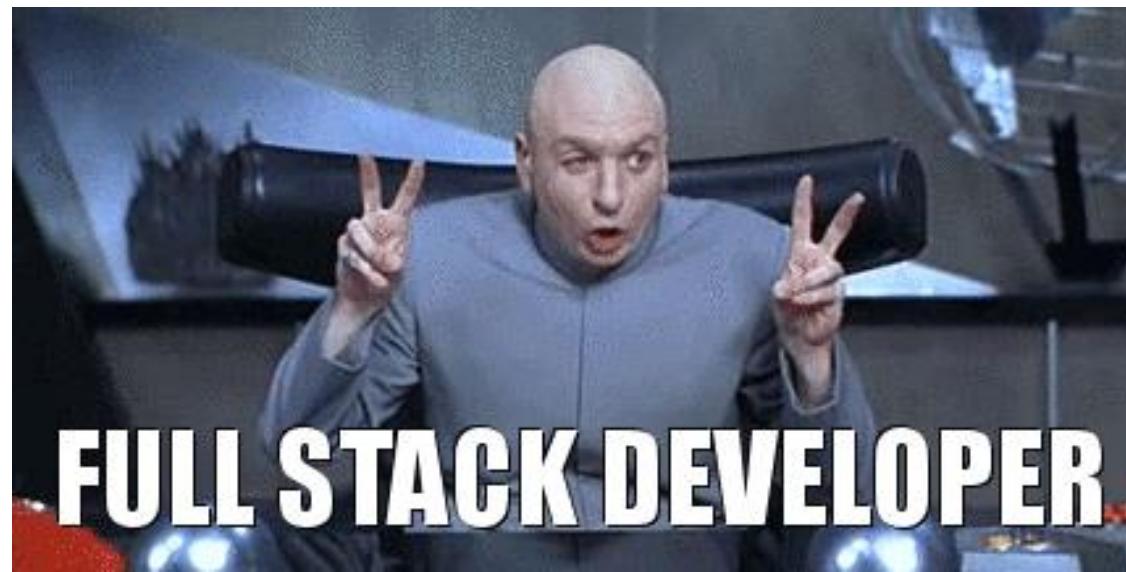
# A server logic function ----
server <- function(input, output) { }

# A call to run the app ----
shinyApp(ui = ui, server = server)
```

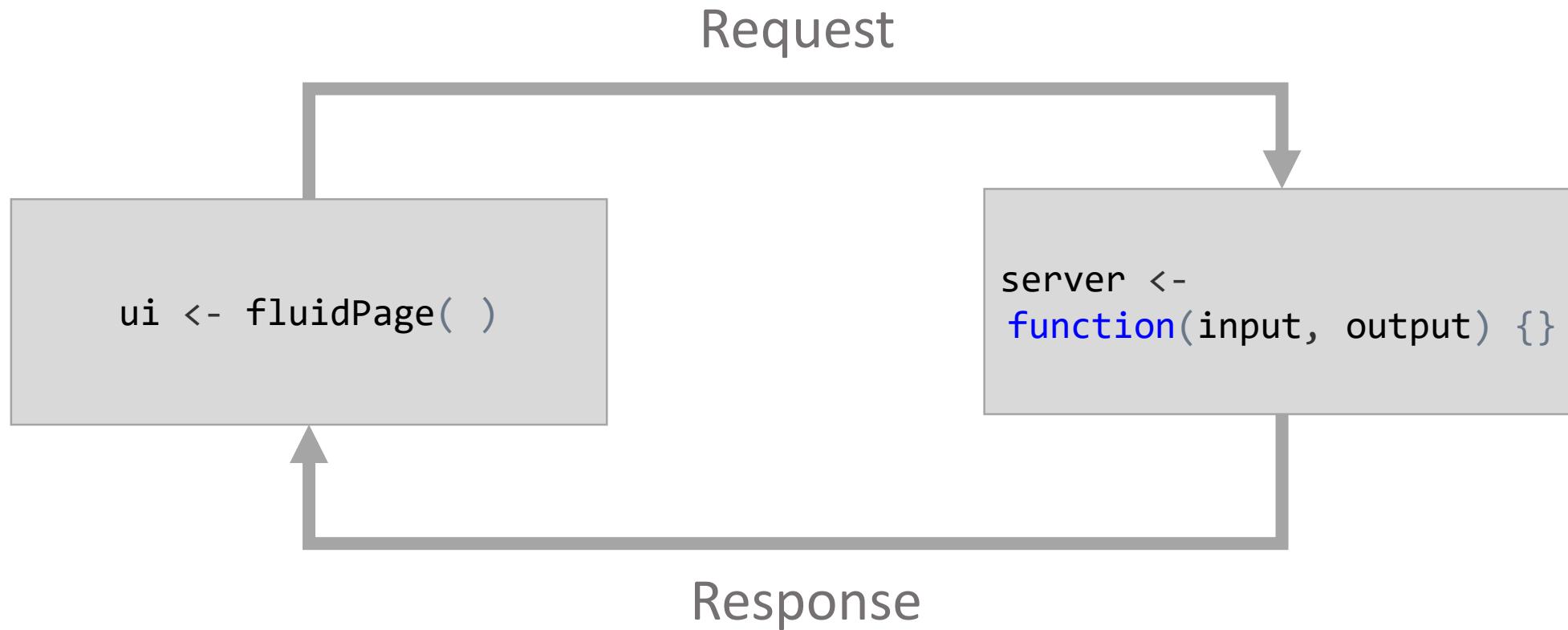
A LITTLE MORE SHINY

<https://shiny.rstudio.com/tutorial/>

Creating a Shiny App is like making a website.
Requires some understanding of front and backend design



Creating a Shiny App is like making a website.
Requires some understanding of front and backend design



Chase will do a quick walk-through of:
<https://shiny.rstudio.com/gallery/kmeans-example.html>

Some Tips to Make Coding Shiny Smoother



First Tip = Bad Tip

Unit Tests in Shiny... Not so Easy

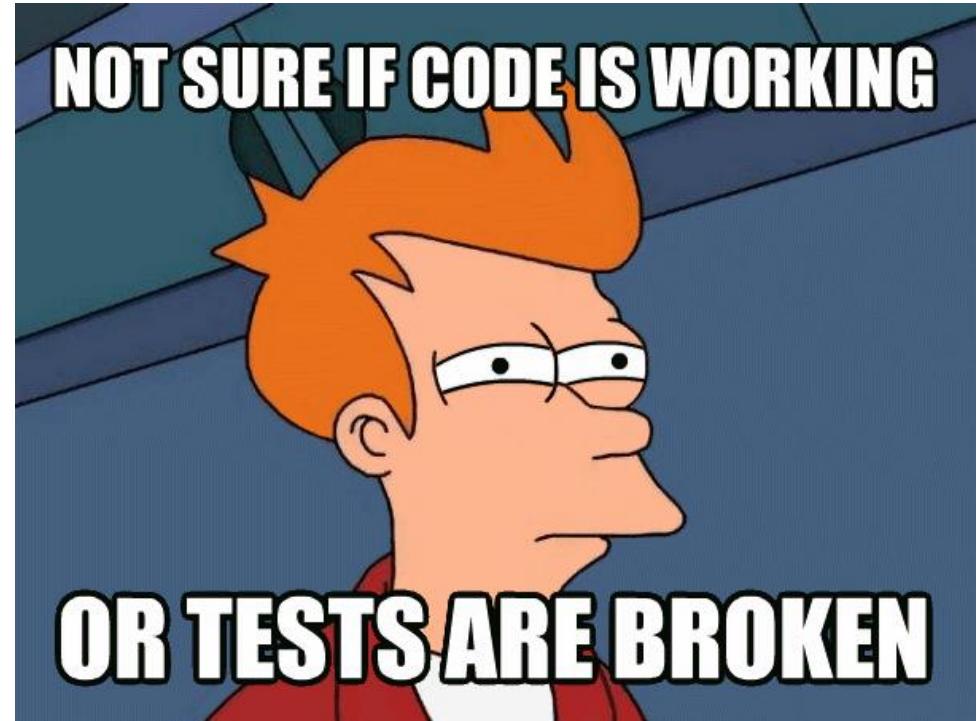
shinytest

Test Shiny Apps

build failing build unknown CRAN not published downloads 0/month

Installation

```
devtools::install_github("rstudio/shinytest")
```



<https://www.rstudio.com/resources/webinars/testing-shiny-applications-with-shinytest/>

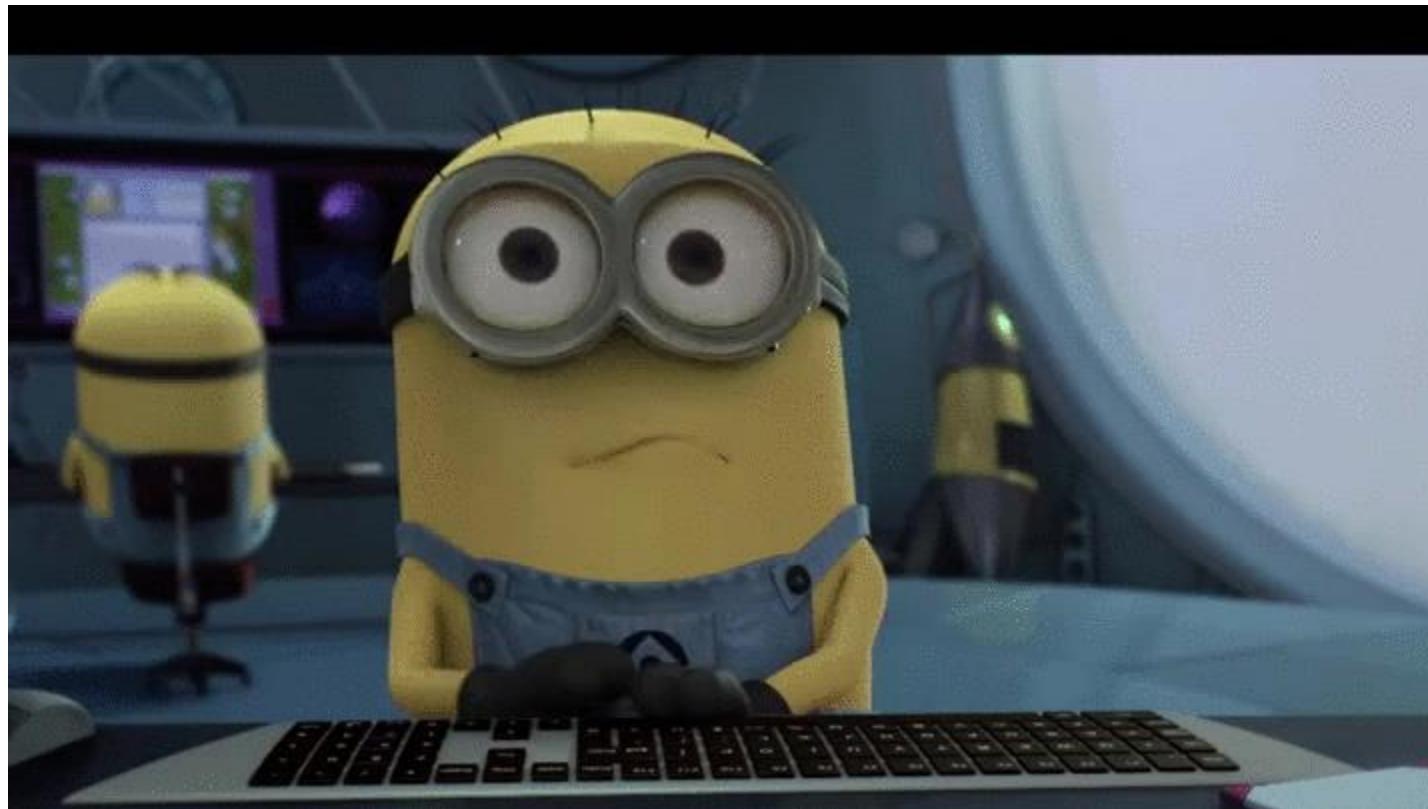
If you aren't using RStudio,
Why?....



Winston Chang created Shiny...
Winston Chang is a software engineer at RStudio...
Ergo:
RStudio works well with building Shiny apps



Distractions Happen



“Git” GitHub and “Commit” to it

Versioning is essential to reproducibility...and
keeping your sanity



GitHub

Username

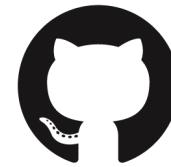
Email

Password

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.



GitHub

= Project Website
= Issue Tracking
= App Distribution

General Information:

- For more information about the method, as well as links to the full code, please visit chasemc.github.io/IDBac
- Bugs and suggestions may be reported on the [IDBac Issues Page on GitHub](#). 

Github +Zenodo = Automatic DOI Minting for Each Release

- Software:
 - For reproducibility, cite this version of IDBac with the version DOI: "10.5281/zenodo.1115619"
 - Past versions of IDBac can be found here: DOI [10.5281/zenodo.1115620](https://doi.org/10.5281/zenodo.1115620)



Tip: Make sure your users have needed packages!

```
# Function to Install and Load R Packages
Install_And_Load <- function(Required_Packages)
{ Remaining_Packages <- Required_Packages[!(Required_Packages %in% installed.packages()[, "Package"])]
  if (length(Remaining_Packages))
  { install.packages(Remaining_Packages)
  }
  for (package_name in Required_Packages)
  {
    library(package_name,
           character.only = TRUE,
           quietly = TRUE) }}
# Required packages to install and load
Required_Packages = c("parallel","shiny","MALDIquant","MALDIquantForeign","mzR","readxl",
"networkD3","factoextra","ggplot2","ape","FactoMineR","dendextend","networkD3","reshape2","dplyr",
"igraph","rgl")
# Install and Load Packages
Install_And_Load(Required_Packages)
```



Provide Your Users an Option to Install Shiny Apps via an Easy .exe



Tips and Tricks



RInno: Allows Users to Create Shiny Apps as a Shareable .exe Installer

- Not the easiest, but possible.
- Works... but needs work:
 - Continuous installation, `devtools::install_github()`
 - Package Dependencies (packrat?)



Why make users search for documentation?

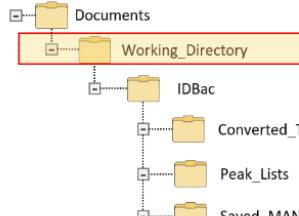
- Include directions in-app

Starting with a Single MALDI Plate of Raw Data

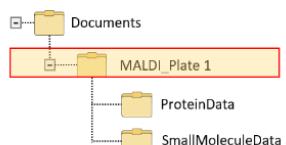
Instructions

1: Working Directory This directs where on your computer you would like to create an IDBac working directory.

In the folder you select, IDBac will create sub-folders within a main directory named "IDBac".



2: Raw Data Your RAW data is a single folder that contains: one subfolder containing protein data and one subfolder containing small-molecule data



*Note: Sometimes the browser window won't pop up, but will still appear in the application bar. See below:



Workflow Pane

1: Working Directory

Click to select your Working Directory

2: Raw Data

Click to select the location of your RAW data

No Folder Selected

3: Choose your Sample Map file, the excel sheet that IDBac will use to rename your files.

Browse... No file selected

4: Click "Convert to mzXML" to begin spectra conversion.

Convert to mzXML

Note: If you canceled out of the popup after spectra conversion completed, you can process your converted spectra using the button below: (but only after all files have been converted) This step is not necessary otherwise.

Process mzXML spectra

IDBac: Main Page

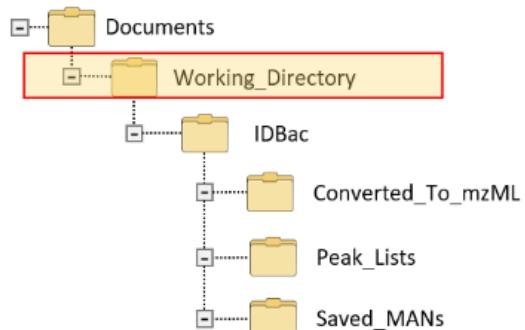
Why make users search for documentation?
- Include directions in-app

Starting with a Single MALDI Plate of Raw Data

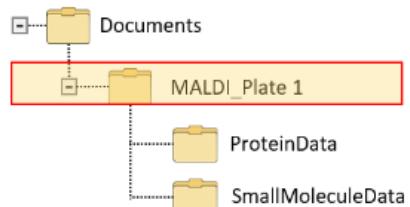
Instructions

1: Working Directory This directs where on your computer you would like to create an IDBac working directory.

In the folder you select, IDBac will create sub-folders within a main directory named "IDBac":



2: Raw Data Your RAW data is a single folder that contains: one subfolder containing protein data and one subfolder containing small-molecule data



*Note: Sometimes the browser window won't pop up, but will still appear in the application bar. See below:



Workflow Pane

1: Working Directory

Click to select your Working Directory

2: Raw Data

Click to select the location of your RAW data

No Folder Selected

3: Choose your Sample Map file, the excel sheet that IDBac will use to rename your files.

Browse... No file selected

4: Click "Convert to mzXML" to begin spectra conversion.

Convert to mzXML

Note: If you canceled out of the popup after spectra conversion completed, you can process your converted spectra using the button below: (but only after all files have been converted) This step is not necessary otherwise.

Process mzXML spectra

Tips and Tricks

Everyone Dislikes Popups. But I recommend their (judicious) use in Shiny

```
popup1<-reactive({  
  showModal(modalDialog(  
    title = "Important message",  
    "When file-conversions are complete this pop-up will be replaced by a summary  
    of the conversion.",br(),  
    "IDBac uses parallel processing to make these computations faster, unfortunately  
    this means we can't show a progress bar.",br(),  
    "This also means your computer might be slow during these file conversions.",br(),  
    "To check what has been converted you can navigate to:",  
    paste0(selectedDirectory(), "/",uniquifiedIDBac(),"/Converted_To_mzML"),  
    easyClose = FALSE, size="l",footer=""  
  ))  
})
```

IDBac

127.0.0.1:3785

Search

Select here if you have already converted data and just want to re-analyze it

Conversion Complete

586 files were converted into 100 open data format files.

To check what has been converted you can navigate to: C:\Users\chase\Desktop>IDBac-7/Converted_To_mzML

Click to continue with Peak Processing Close

Instructions

1: This directs where on your computer you would like to create an IDBac

In the folder you select- IDBac will create folders within a main directory named "IDBac".

Documents

Working_Directory

IDBac

Converted_To_mzML

Peak_Lists

Saved_MANs

2: Your RAW data should be one folder that contains: a folder containing protein data and folder containing small-molecule data

Click to select your Working Directory

[1] "C:\Users\chase\Desktop"

3: Choose your Sample Map file, the excel sheet which IDBac will use to rename your files.

Click to select the location of your RAW data

protein

small_molecule

4: Click "Convert to mzXML" to begin spectra conversion.

Convert to mzXML

Note: Sometimes the browser window won't pop up, but will still appear in the application bar. See below:

Type here to search

Process mzXML spectra

Spectra Processing Progress

Scoping Assignment for Debugging

I usually avoid using <<-

But... <<- works great for debugging complicated Shiny apps!

Insert **whatTheHellIsThis <<- shinyReactiveVariable()**

Resources Suggestions for R/Shiny Beginners

New to R

Where to start:

- swirlstats.com
- r4ds.had.co.nz
 - adv-r.had.co.nz

Where to find answers:

- stackexchange.com
- community.rstudio.com
- Web Search! (i.e. google.com)
 - E.g. (search: *convert list to data frame in R*)

New to Shiny

- shiny.rstudio.com/tutorial
- shiny.rstudio.com/gallery
- shiny.rstudio.com/gallery/widget-gallery
- deanattali.com/blog/building-shiny-apps-tutorial/

Active Community on Twitter- #rstats

Suggestions to follow (not inclusive):

- [@daattali](#) (shiny)
- [@hadleywickham](#)
- [@JennyBryan](#)
- [@drob](#)
- [@thomasp85](#)
- [@kierisi](#)
- [@dataandme](#)

Special Thanks



Dr. Laura Sanchez

**MEDICINAL
CHEMISTRY
AND
PHARMACOGNOSY
COLLEGE
OF PHARMACY**



Murphy Lab

Dr. Brian Murphy

Sofia Costa

Maryam Elfeki

Vanessa Nepomuceno

Jeong Ho Lee

Antonio Hernandez

Milan Patel



@ChaseClarkatUIC
murphylabuic.com
@Murphylabuic

Contribute

If you find IDBac useful and would like to make it even better,
contribute to the code or leave comments/suggestions at:

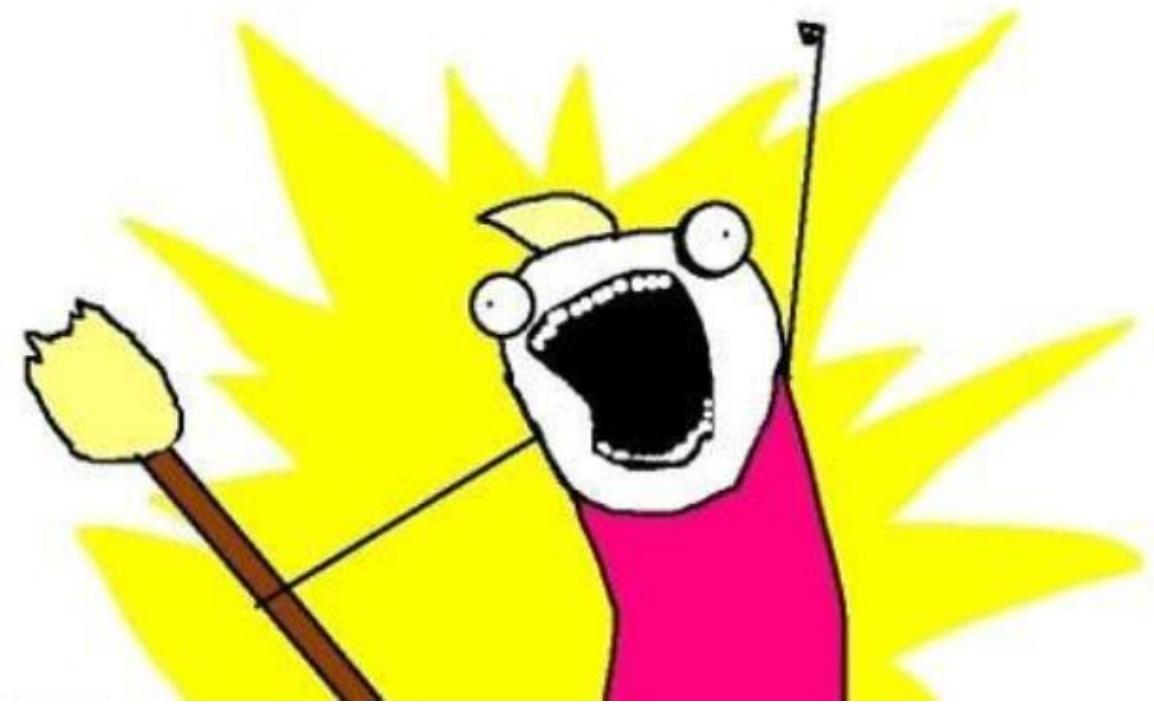
<https://github.com/chasemc/IDBac>



CODE ALL THE THINGS!



<https://goo.gl/forms/DWnAXBPhHOGtbzHs1>



```
library(meme)  
meme("https://imgflip.com/s/meme/X-All-The-Y.jpg", "code all the things!")
```