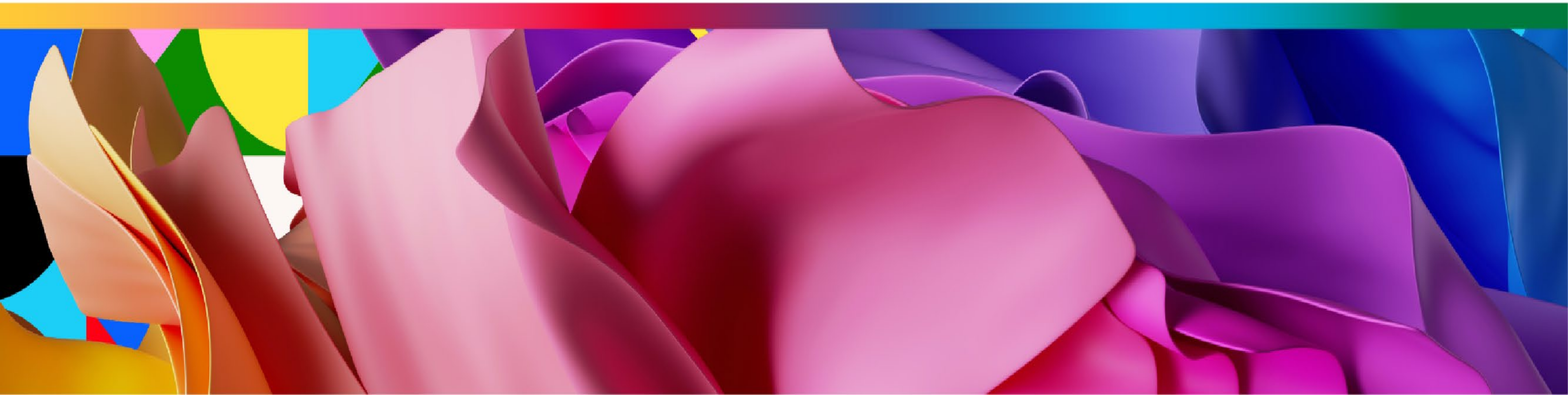
STI&HIV2025
WORLD CONGRESS

Data Visualisation for Beginners using R

Odile Harrison, Tatum Mortimer & Smritee Dabee

Wednesday 30th July 2025



Workshop Objectives:

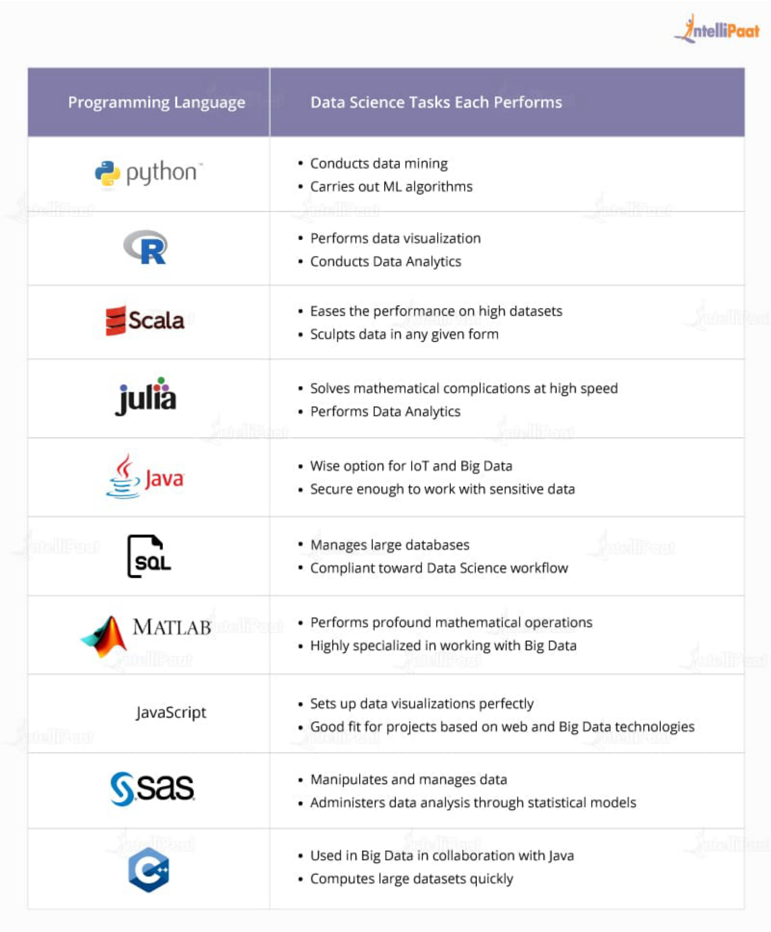
- Understand why Data Vizualisation is important
- What are good (and bad) vizualisation practices
- Gain hands-on experience in generating publication ready figures using R

Data Visualisation: Transforms non-visual big data in a visual form

- Allows information to be rapidly understood
- Detects patterns and relationships within and between data
- Ensures you get your message across

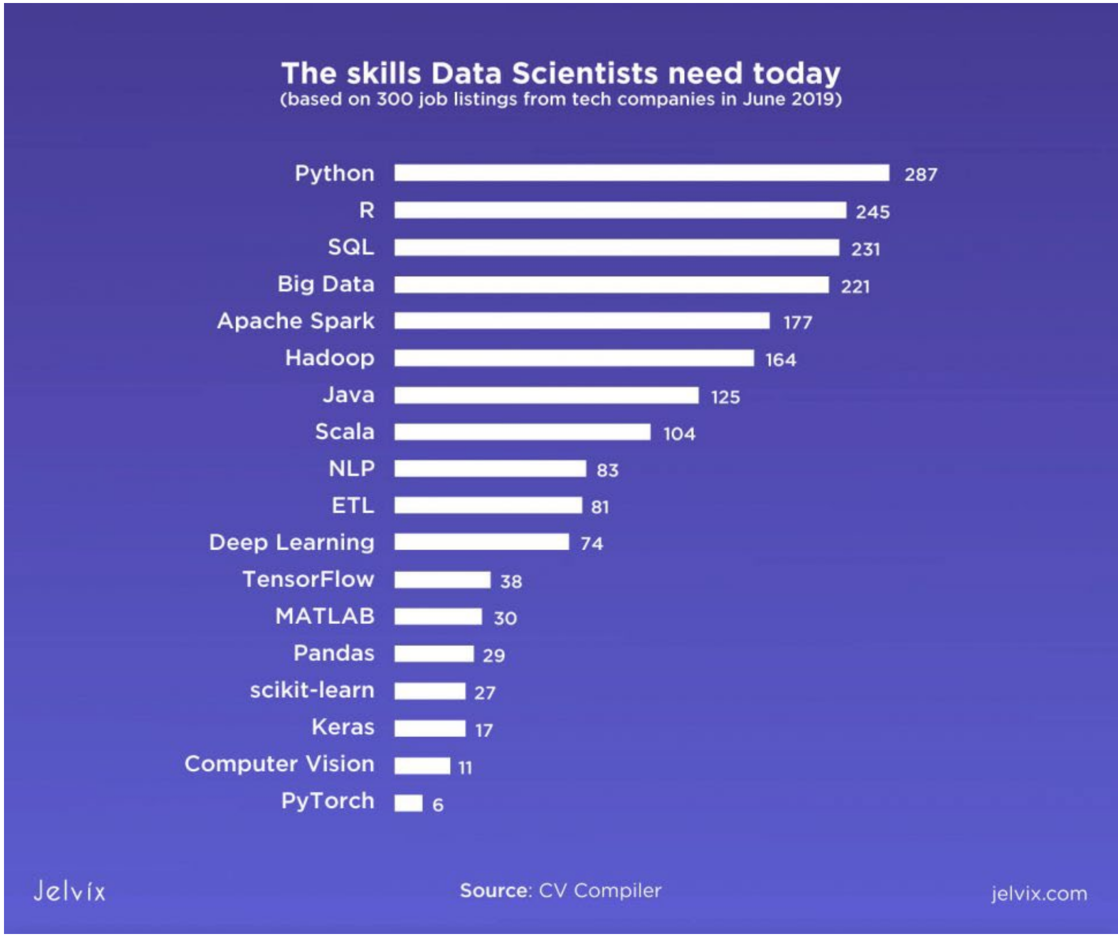
Top 5 Creative Data Visualization Examples for Data Analysis

Data Visualisation tools



IntelliPaat

Programming Language	Data Science Tasks Each Performs
python	• Conducts data mining • Carries out ML algorithms
R	• Performs data visualization • Conducts Data Analytics
Scala	• Eases the performance on high datasets • Sculpts data in any given form
julia	• Solves mathematical complications at high speed • Performs Data Analytics
Java	• Wise option for IoT and Big Data • Secure enough to work with sensitive data
SQL	• Manages large databases • Compliant toward Data Science workflow
MATLAB	• Performs profound mathematical operations • Highly specialized in working with Big Data
JavaScript	• Sets up data visualizations perfectly • Good fit for projects based on web and Big Data technologies
sas	• Manipulates and manages data • Administers data analysis through statistical models
C++	• Used in Big Data in collaboration with Java • Computes large datasets quickly



The skills Data Scientists need today
(based on 300 job listings from tech companies in June 2019)

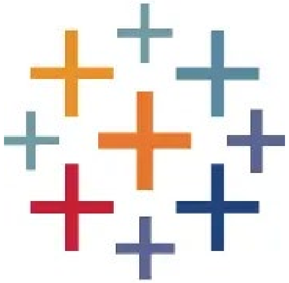
Jelvix

Source: CV Compiler

jelvix.com

Online and/or commercial tools

RAWGraphs (https://www.rawgraphs.io/)



Tableau

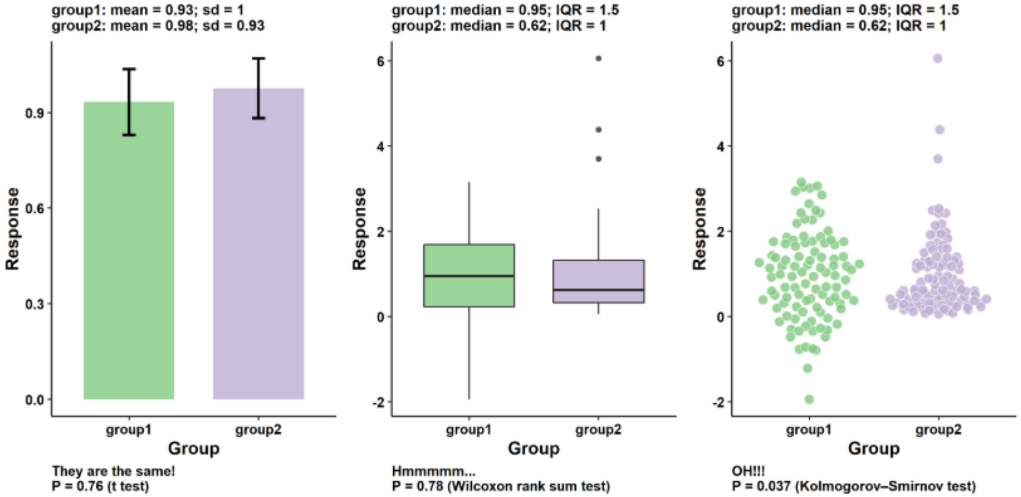
Microsoft 365

Good (and bad) visualisation

Good visualisation should be:

- Easy to interpret and read
- Highlight hidden insights to support your data
- Appeal to your audience
- Display massive insights using limited space

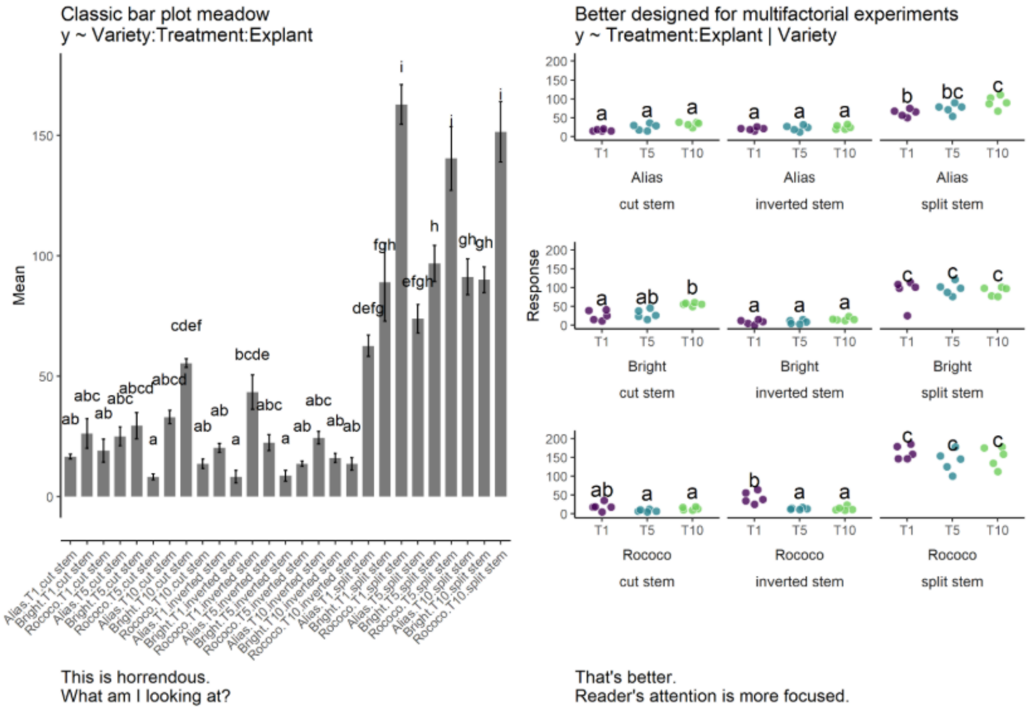
Two or more groups with multiple observations. The task of the visualization is to show the means and the spread (dispersion) of the data.



Weissgerber et al., 2015, PLOS Biology. Paper describes limitations of bar charts

Check this resource: GitHub - cxli233/FriendsDontLetFriends: Friends don't let friends make certain types of data visualization - What are they and why are they bad.

Good (and bad) visualisation


Classic bar plot meadow
y ~ Variety:Treatment:Explant

This is horrendous.
What am I looking at?

Better designed for multifactorial experiments
y ~ Treatment:Explant | Variety

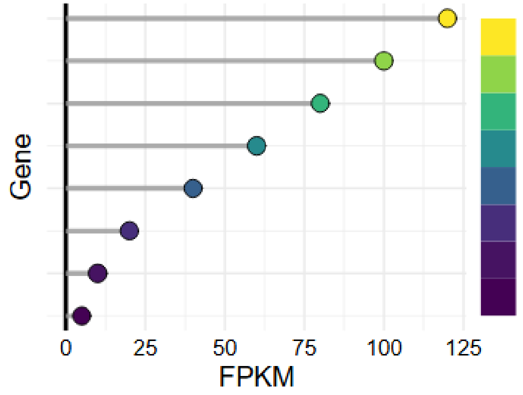
That's better.
Reader's attention is more focused.

Bar plot meadows are common for multi-factorial experiments. However, a bar plot meadow is poorly designed for its purpose. To communicate results of a multi-factorial experiment, it requires thoughtful designs regarding grouping/faceting by factors of interest.

Matand et al., 2020, BMC Plant Biology

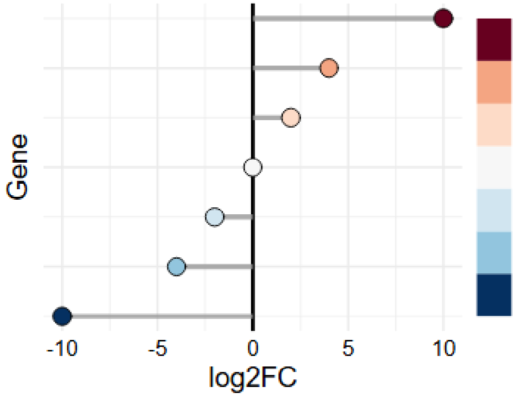
Good (and bad) visualisation

Darkest color = Min
Lightest color = Max



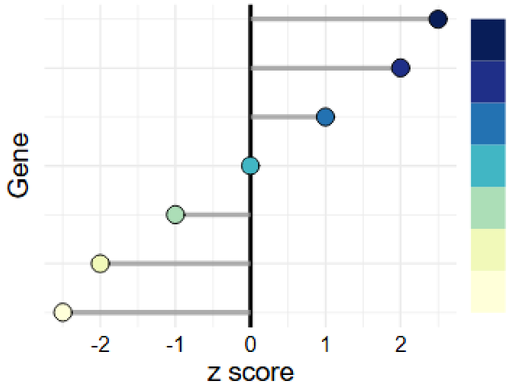
This is good.

Lightest color = 0
Darkest colors = Max absolutes



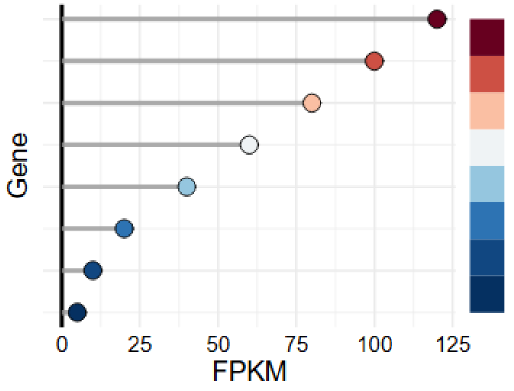
This is good.

Darkest color = Max
Lightest color = Min



This is good.

Lighest color means nothing
(neither mean nor median).



A data visualization sin

Think carefully about colour scales in figures. They should represent something meaningful.

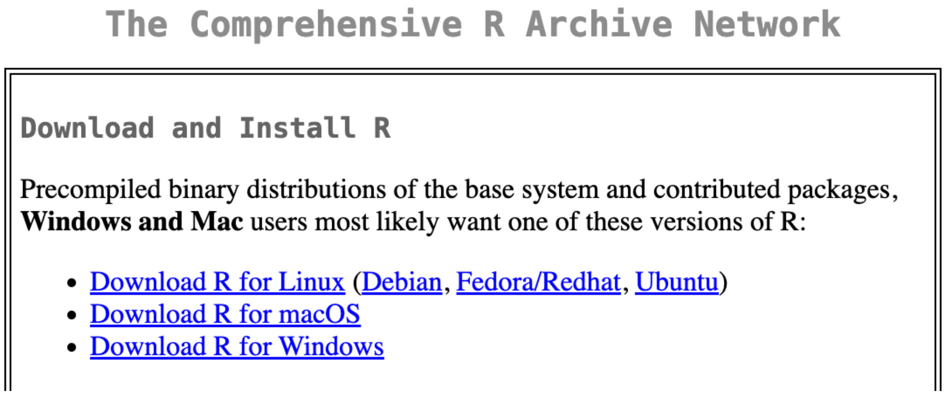
Common applications of R

- I have two datasets, and I want to merge them
- I have a very large file (Excel can’t open) how do I select a few columns to work with?
- Data transformations
- Data cleaning
- Statistical analysis
- Visualizations: how to I generate publication ready plots?
- Spatial analysis: creating maps
- Dashboards and interactive interfaces

STI&HIV2025 WORLD CONGRESS
July 26-30, 2025 • Montreal, Canada

How to get started

Step 1: Downloading & Install R (https://cran.r-project.org/)



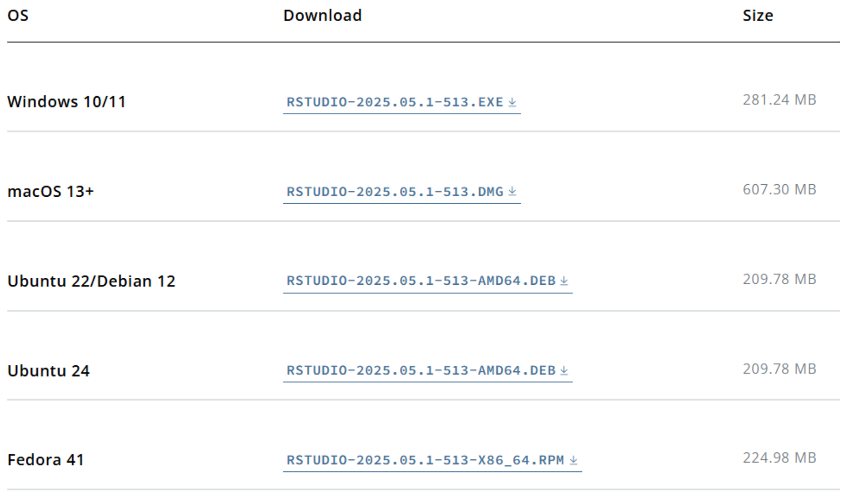
The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux (Debian, Fedora/Redhat, Ubuntu)
- Download R for macOS
- Download R for Windows

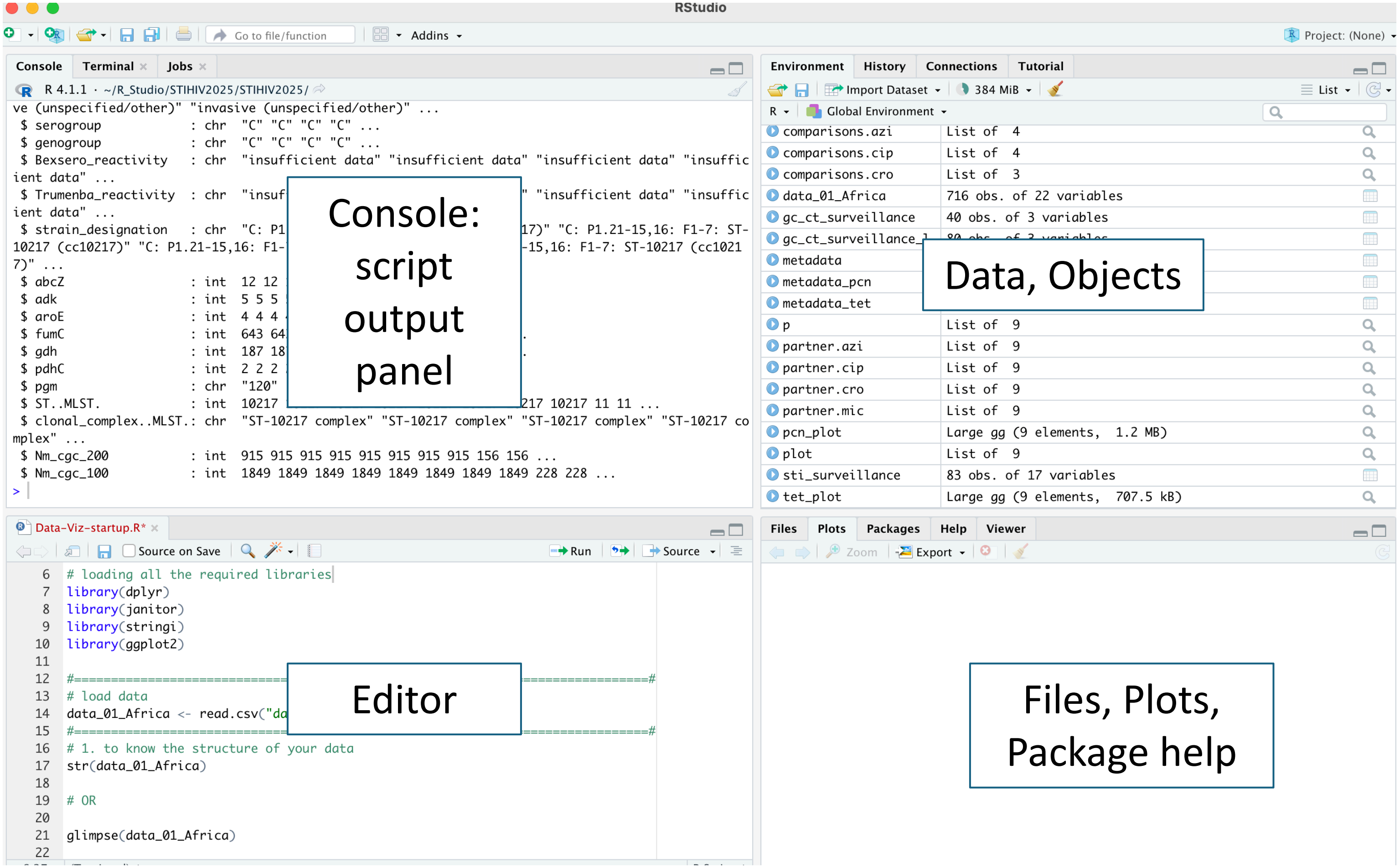
Step 2: Download and install R studio (user interface) (https://posit.co/download/rstudio-desktop/)



OS	Download	Size
Windows 10/11	RSTUDIO-2025.05.1-513.EXE ⤓	281.24 MB
macOS 13+	RSTUDIO-2025.05.1-513.DMG ⤓	607.30 MB
Ubuntu 22/Debian 12	RSTUDIO-2025.05.1-513-AMD64.DEB ⤓	209.78 MB
Ubuntu 24	RSTUDIO-2025.05.1-513-AMD64.DEB ⤓	209.78 MB
Fedora 41	RSTUDIO-2025.05.1-513-X86_64.RPM ⤓	224.98 MB

STI&HIV2025 WORLD CONGRESS
July 26-30, 2025 • Montreal, Canada

Navigating the R console



```
R 4.1.1 · ~/R_Studio/STIHIV2025/STIHIV2025/
ve (unspecified/other)" "invasive (unspecified/other)" ...
 $ serogroup             : chr  "C" "C" "C" "C" ...
 $ genogroup             : chr  "C" "C" "C" "C" ...
 $ Bexsero_reactivity    : chr  "insufficient data" "insufficient data" "insufficient data" "insuffic
ient data" ...
 $ Trumenba_reactivity   : chr  "insuf                              " "insufficient data" "insuffic
ient data" ...
 $ strain_designation    : chr  "C: P1                              17)" "C: P1.21-15,16: F1-7: ST-
10217 (cc10217)" "C: P1.21-15,16: F1-                              -15,16: F1-7: ST-10217 (cc1021
7)" ...
 $ abcZ                  : int  12 12
 $ adk                   : int  5 5 5
 $ aroE                  : int  4 4 4
 $ fumC                  : int  643 64                              .
 $ gdh                   : int  187 18                              .
 $ pdhC                  : int  2 2 2
 $ pgm                   : chr  "120"
 $ ST..MLST.             : int  10217                               217 10217 11 11 ...
 $ clonal_complex..MLST.: chr  "ST-10217 complex" "ST-10217 complex" "ST-10217 complex" "ST-10217 co
mplex" ...
 $ Nm_cgc_200            : int  915 915 915 915 915 915 915 915 156 156 ...
 $ Nm_cgc_100            : int  1849 1849 1849 1849 1849 1849 1849 1849 228 228 ...
> 
```

Console: script output panel

comparisons.azi	List of 4
comparisons.cip	List of 4
comparisons.cro	List of 3
data_01_Africa	716 obs. of 22 variables
gc_ct_surveillance	40 obs. of 3 variables
gc_ct_surveillance_1	80 obs. of 3 variables
metadata	
metadata_pcn	
metadata_tet	
p	List of 9
partner.azi	List of 9
partner.cip	List of 9
partner.cro	List of 9
partner.mic	List of 9
pcn_plot	Large gg (9 elements, 1.2 MB)
plot	List of 9
sti_surveillance	83 obs. of 17 variables
tet_plot	Large gg (9 elements, 707.5 kB)

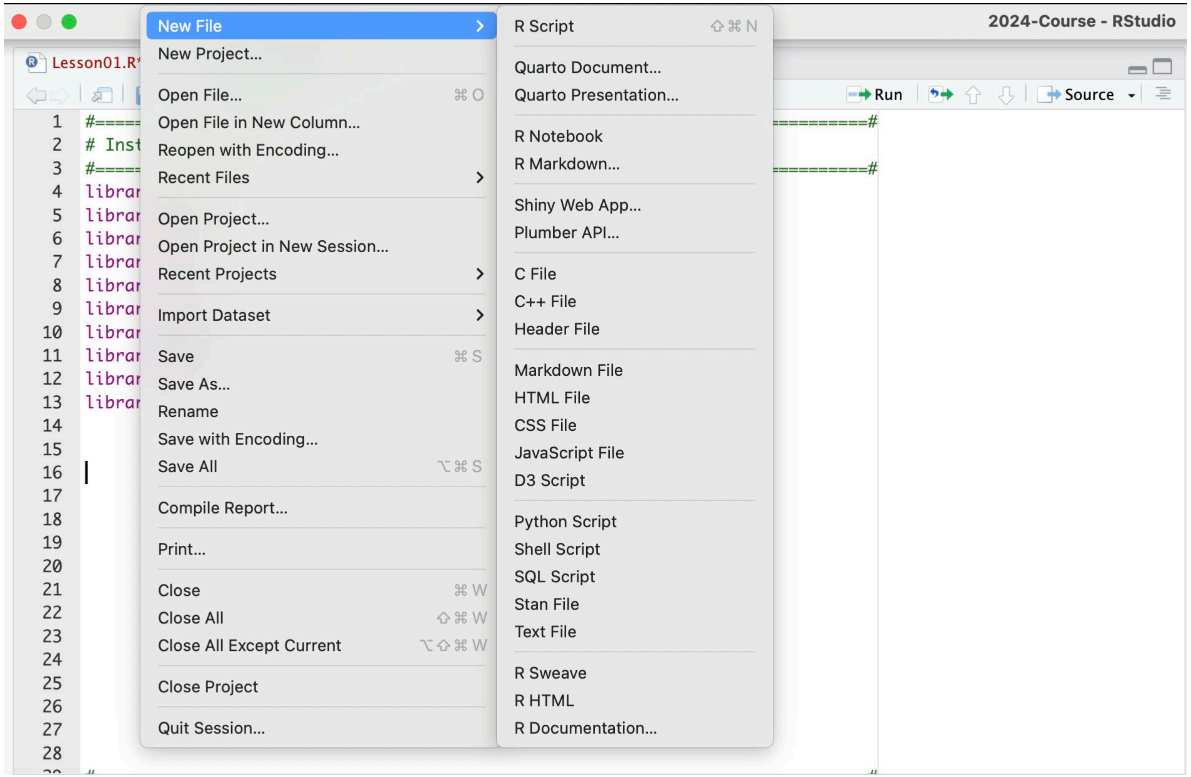
Data, Objects

```r
# loading all the required libraries
library(dplyr)
library(janitor)
library(stringi)
library(ggplot2)

#==============================                   ==================#
# load data
data_01_Africa <- read.csv("da
#==============================                   ==================#
# 1. to know the structure of your data
str(data_01_Africa)

# OR

glimpse(data_01_Africa)

```

Editor

Files, Plots, Package help

R studio



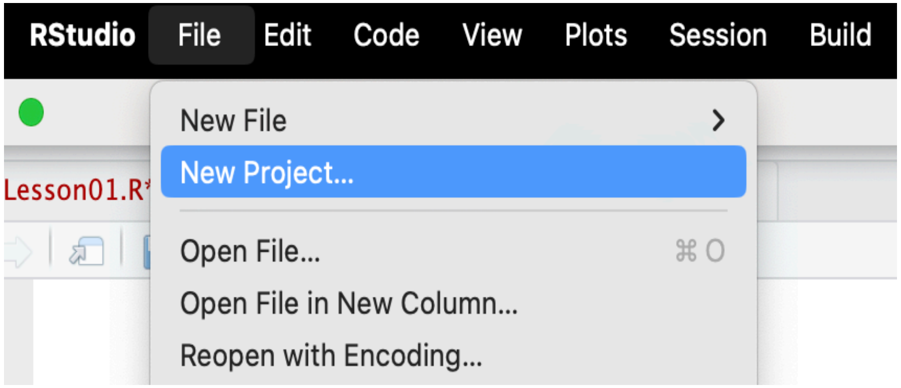
```
#=====                ==========#
# Inst
#=====                ==========#
librar
librar
librar
librar
librar
librar
librar
librar
librar
librar
```

R studio: User interface, Customizable interface, Code autocompletion, write code in other languages

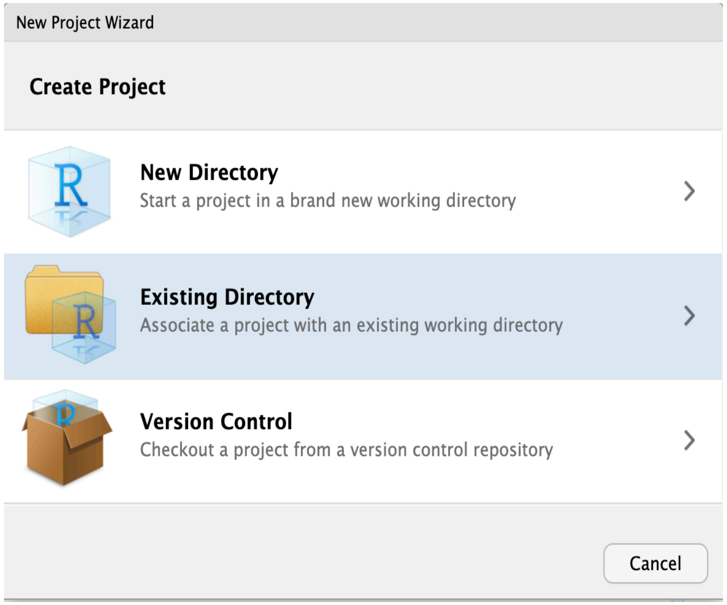
STI&HIV2025
WORLD CONGRESS
July 26-30, 2025 • Montreal, Canada

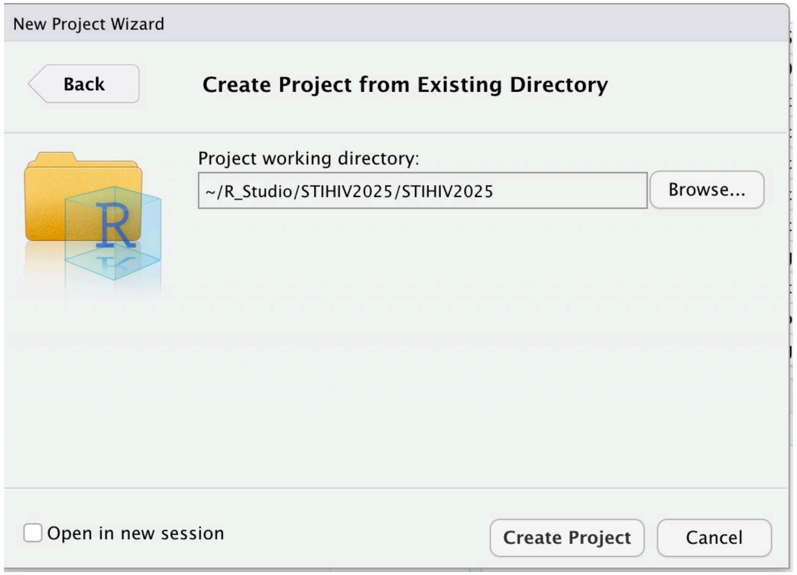
Creating an R project and/or saving your work

Step 1



Step 2


New Project Wizard
Create Project
New Directory
Start a project in a brand new working directory
Existing Directory
Associate a project with an existing working directory
Version Control
Checkout a project from a version control repository
Cancel



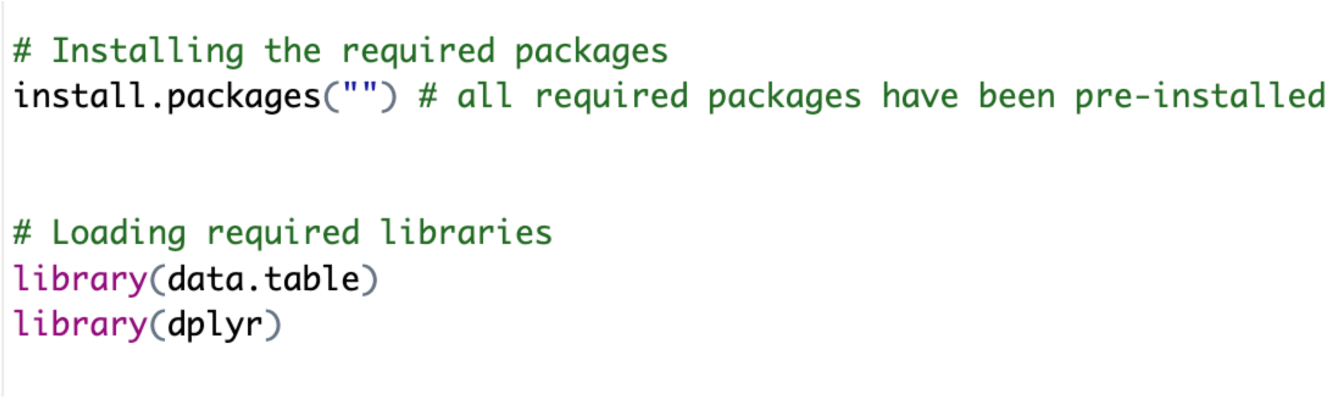
Project working directory: ~/R_Studio/STIHIV2025/STIHIV2025

☐ Open in new session

Basic but Important Commands

1. **#** - not a command, but used for commenting code

2. Installing and loading packages:
install.packages("name_of_package") then **library(name_of_package)**



```
# Installing the required packages
install.packages("") # all required packages have been pre-installed


# Loading required libraries
library(data.table)
library(dplyr)
```

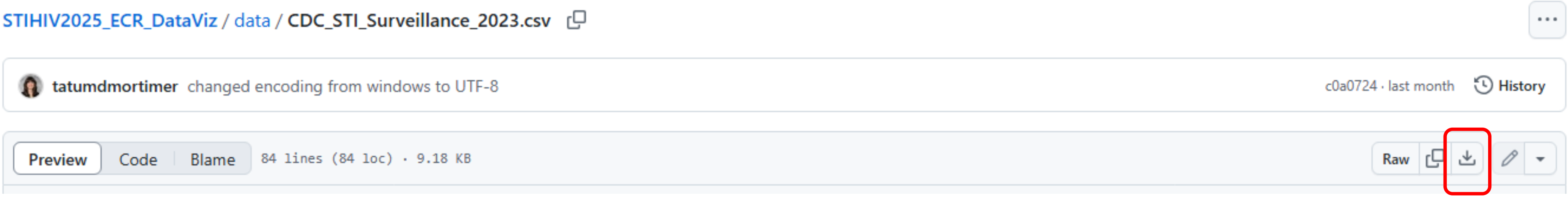
3. Use the function **search()** to know which packages have been loaded

4. Use **setwd()** & **getwd()** to set working directory etc

Files used in the course

[GitHub - mortimer-lab/STIHIV2025_ECR_DataViz: Resources for the Data Visualization Workshop at STI HIV 2025]

Download Course_data



STIHIV2025_ECR_DataViz / data / CDC_STI_Surveillance_2023.csv

tatumdmortimer changed encoding from windows to UTF-8

c0a0724 · last month

84 lines (84 loc) · 9.18 KB

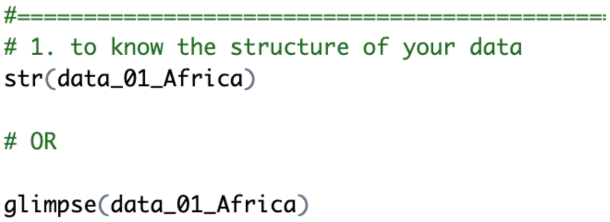
Exploring Data using R

GitHub - mortimer-lab/STIHIV2025_ECR_DataViz: Resources for the Data Visualization Workshop at STI HIV 2025

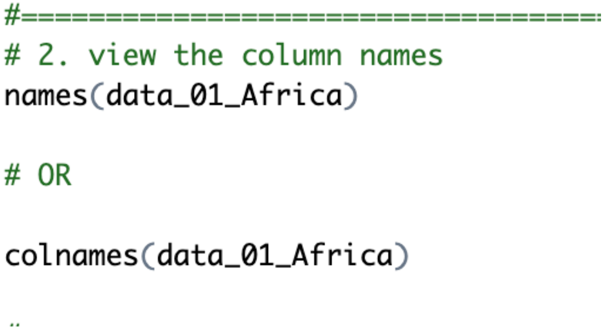
1. Download Course_data: African_meningococci.csv
2. Load the dataset in R:

```
data_01_africa <- read.csv(“data/African_meningococci.csv”)
```

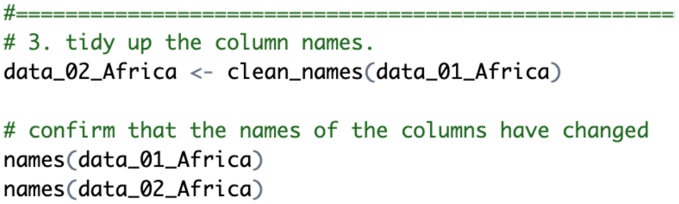
View and understand your data



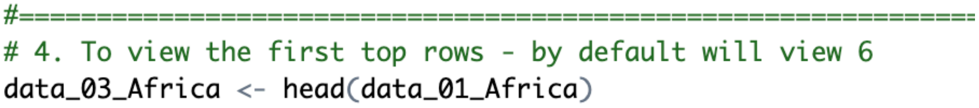
```
#=============================================
# 1. to know the structure of your data
str(data_01_Africa)

# OR

glimpse(data_01_Africa)
```



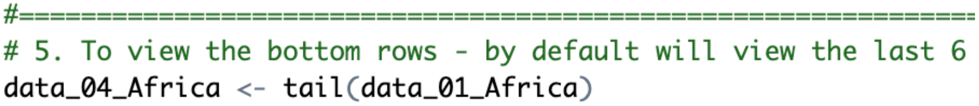
```
#====================================
# 2. view the column names
names(data_01_Africa)

# OR

colnames(data_01_Africa)

..
```



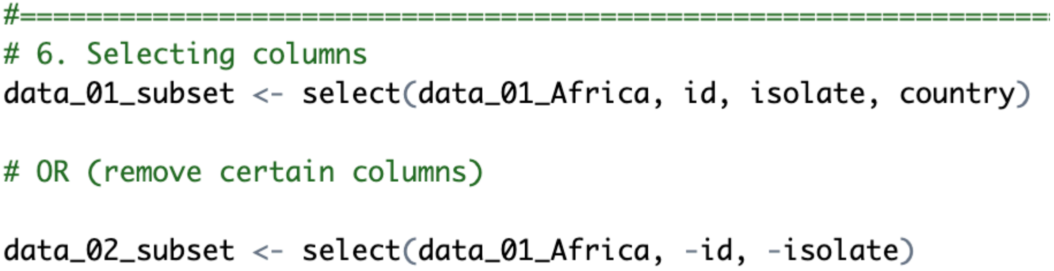
```
#=================================================
# 3. tidy up the column names.
data_02_Africa <- clean_names(data_01_Africa)

# confirm that the names of the columns have changed
names(data_01_Africa)
names(data_02_Africa)
```



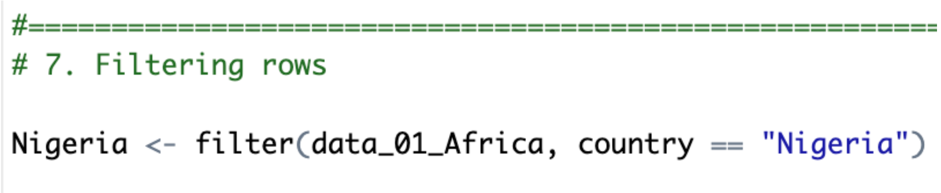
```
#==================================================
# 4. To view the first top rows - by default will view 6
data_03_Africa <- head(data_01_Africa)
```



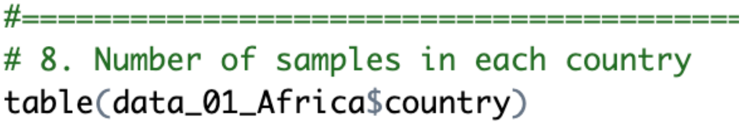
```
#==================================================
# 5. To view the bottom rows - by default will view the last 6
data_04_Africa <- tail(data_01_Africa)
```



```
#=====================================================
# 6. Selecting columns
data_01_subset <- select(data_01_Africa, id, isolate, country)

# OR (remove certain columns)

data_02_subset <- select(data_01_Africa, -id, -isolate)
```



```
#===========================================
# 7. Filtering rows

Nigeria <- filter(data_01_Africa, country == "Nigeria")
```



```
#==================================
# 8. Number of samples in each country
table(data_01_Africa$country)
```

Can you answer these?

1. The number of samples in each year? [hint: table]

2. The number of samples in each serogroup?

3. The number of samples collected in each year only in Togo?
[hint: filter then table]

Answers

1. The number of samples in each year? [hint: table]

```
> Table(data_01_Africa$year
```

2011	2012	2013	2014	2015	2016
115	272	60	26	158	85

2. The number of samples in each serogroup?

```
> Table(data_01_Africa$serogroup
```

A	C	NG	W	X	Y
90	124	8	431	61	2

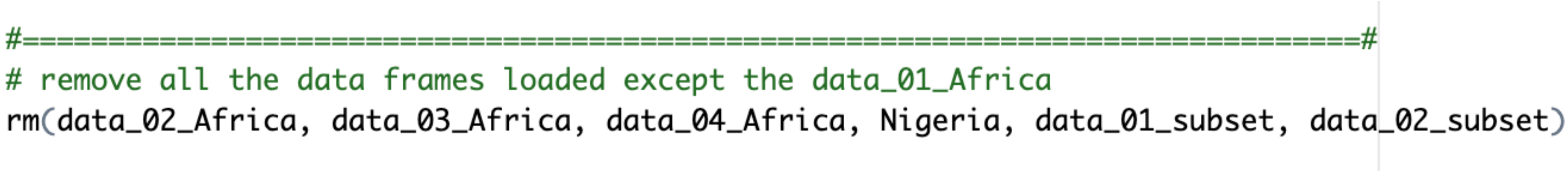
3. The number of samples collected in each year only in Togo? [hint: filter then table]

```
> TOGO <- filter(data_01_Africa, country == “Togo”)
> Table(TOGO$year)
```

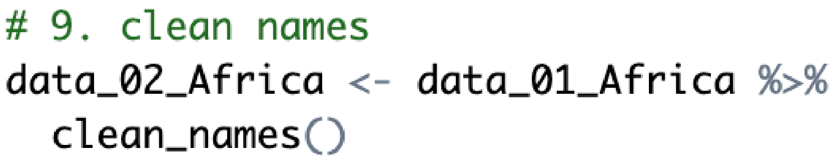
2014	2015	2016
16	12	42

Too many variables

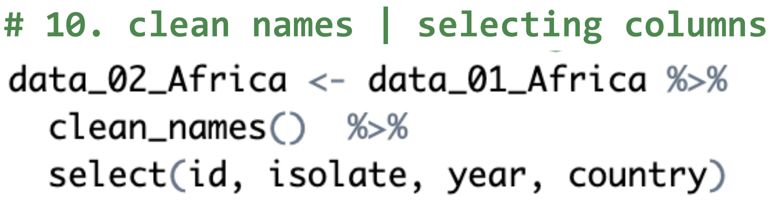
- Pipe operator to link commands: **%>%**
- Can I clean col names, select, filter all in one variable and run once?



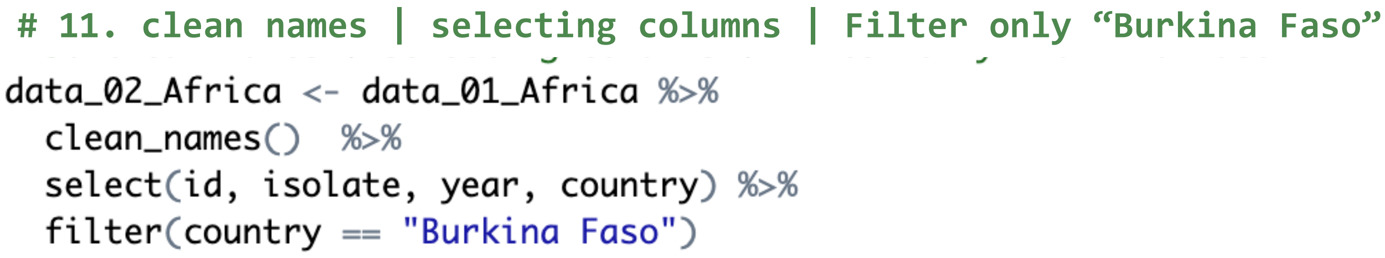
```
#===============================================================================#
# remove all the data frames loaded except the data_01_Africa
rm(data_02_Africa, data_03_Africa, data_04_Africa, Nigeria, data_01_subset, data_02_subset)
```



```
# 9. clean names
data_02_Africa <- data_01_Africa %>%
  clean_names()
```



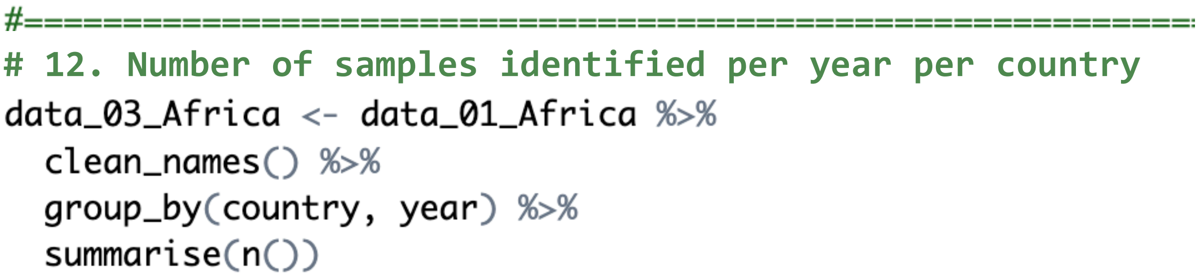
```
# 10. clean names | selecting columns
data_02_Africa <- data_01_Africa %>%
  clean_names()  %>%
  select(id, isolate, year, country)
```



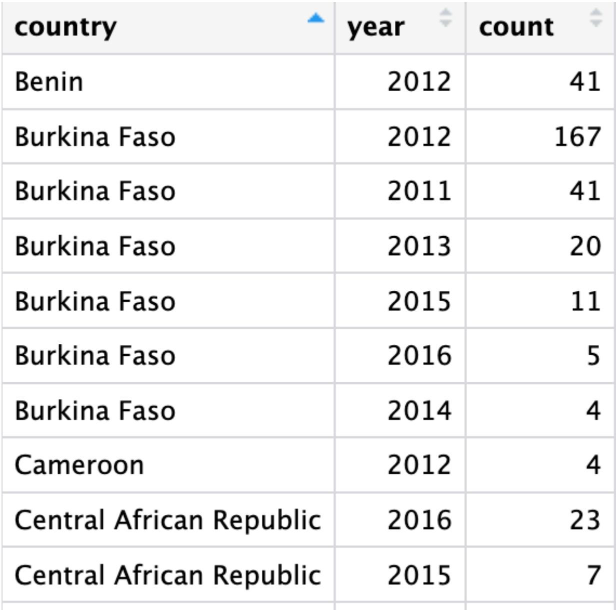
```
# 11. clean names | selecting columns | Filter only “Burkina Faso”
data_02_Africa <- data_01_Africa %>%
  clean_names()  %>%
  select(id, isolate, year, country) %>%
  filter(country == "Burkina Faso")
```

Grouping and summarizing data

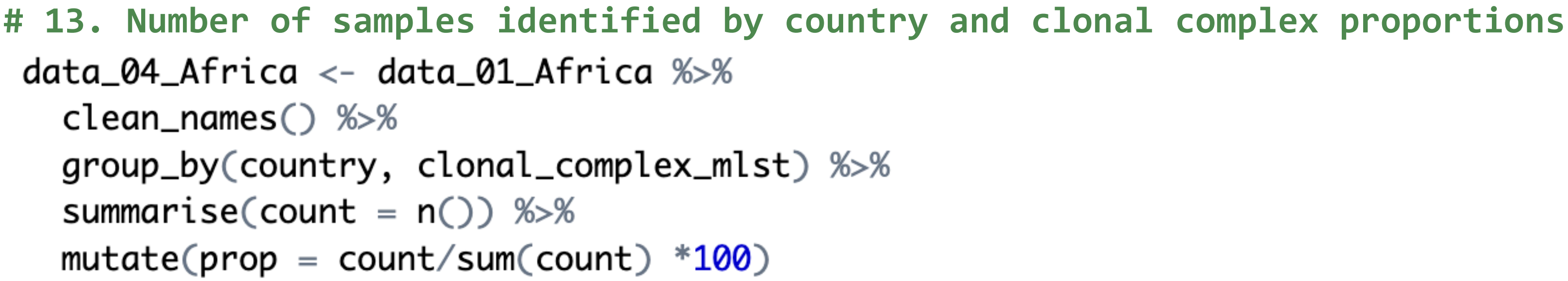
- **group_by** and **summarise** function



```
#==========================================================
# 12. Number of samples identified per year per country
data_03_Africa <- data_01_Africa %>%
  clean_names() %>%
  group_by(country, year) %>%
  summarise(n())
```



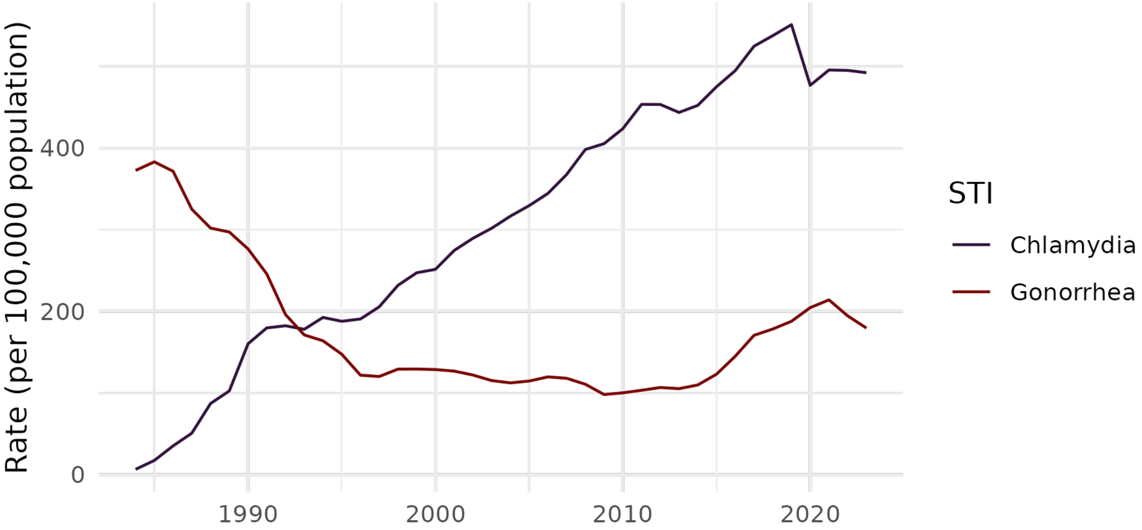
country	year	count
Benin	2012	41
Burkina Faso	2012	167
Burkina Faso	2011	41
Burkina Faso	2013	20
Burkina Faso	2015	11
Burkina Faso	2016	5
Burkina Faso	2014	4
Cameroon	2012	4
Central African Republic	2016	23
Central African Republic	2015	7



```
# 13. Number of samples identified by country and clonal complex proportions
 data_04_Africa <- data_01_Africa %>%
   clean_names() %>%
   group_by(country, clonal_complex_mlst) %>%
   summarise(count = n()) %>%
   mutate(prop = count/sum(count) *100)
```

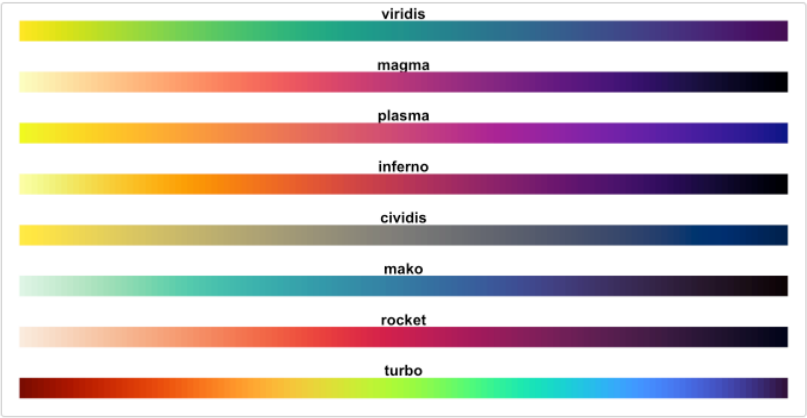
STI&HIV2025
WORLD CONGRESS

STI Surveillance: Line Plot



Packages used

- tidyverse: collection of packages includes dplyr and ggplot2
- janitor: cleaning
- viridis: colorblind-friendly color palettes



Input Data

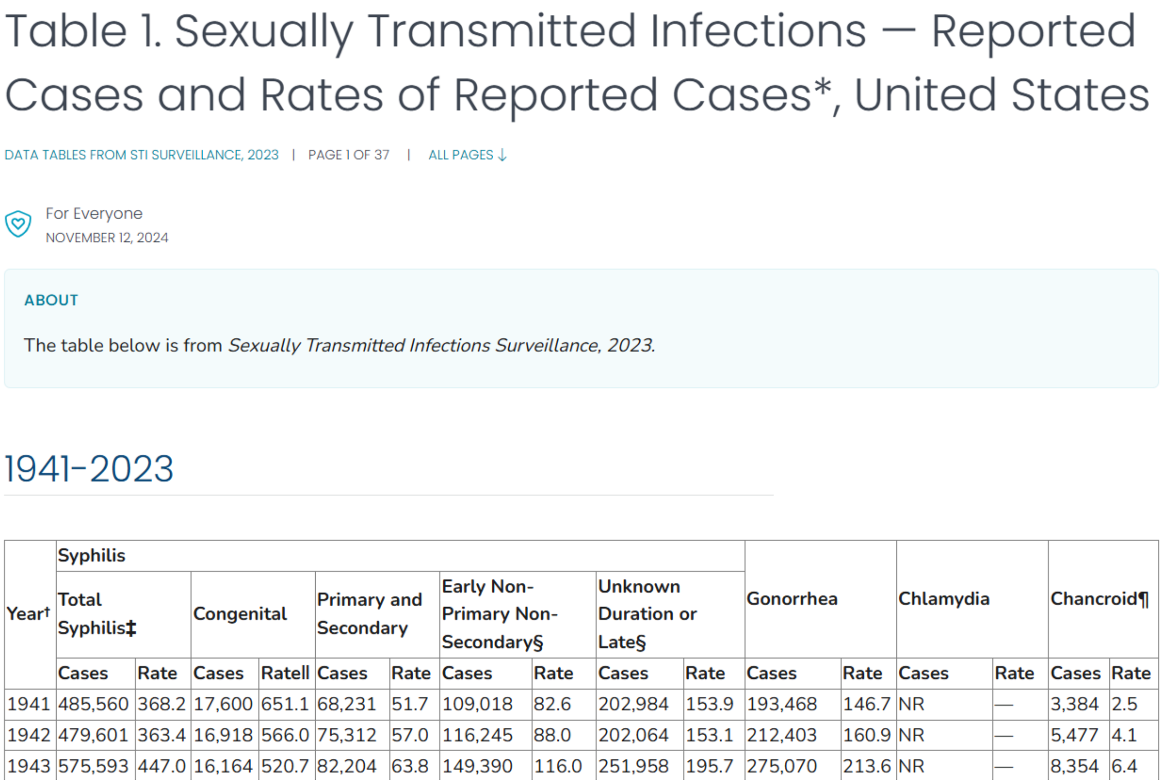


Table 1. Sexually Transmitted Infections — Reported Cases and Rates of Reported Cases*, United States

DATA TABLES FROM STI SURVEILLANCE, 2023 | PAGE 1 OF 37 | ALL PAGES ↓

For Everyone
NOVEMBER 12, 2024

ABOUT

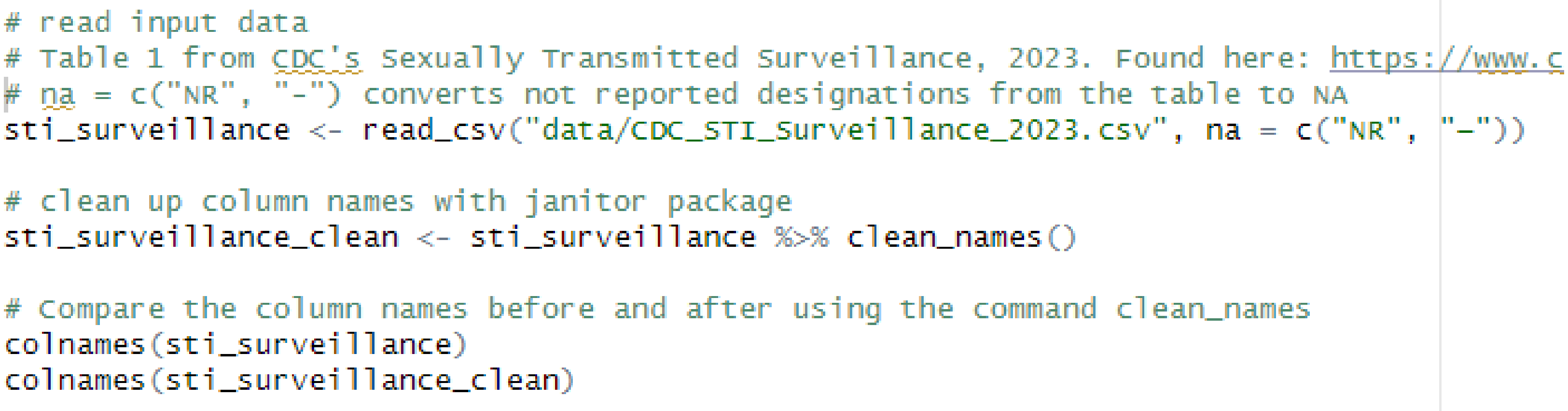
The table below is from *Sexually Transmitted Infections Surveillance, 2023.*

1941-2023

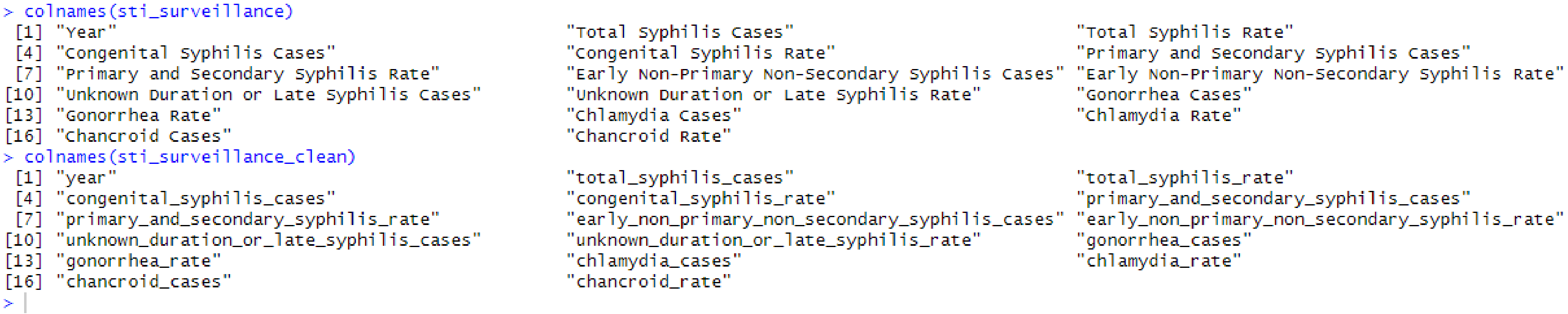
Year†	Syphilis										Gonorrhea		Chlamydia		Chancroid¶	
	Total Syphilis‡		Congenital		Primary and Secondary		Early Non-Primary Non-Secondary§		Unknown Duration or Late§							
	Cases	Rate	Cases	Rate‖	Cases	Rate	Cases	Rate	Cases	Rate	Cases	Rate	Cases	Rate	Cases	Rate
1941	485,560	368.2	17,600	651.1	68,231	51.7	109,018	82.6	202,984	153.9	193,468	146.7	NR	—	3,384	2.5
1942	479,601	363.4	16,918	566.0	75,312	57.0	116,245	88.0	202,064	153.1	212,403	160.9	NR	—	5,477	4.1
1943	575,593	447.0	16,164	520.7	82,204	63.8	149,390	116.0	251,958	195.7	275,070	213.6	NR	—	8,354	6.4

STI&HIV2025
WORLD CONGRESS
July 26-30, 2025 • Montreal, Canada

Reading and Cleaning Input Data

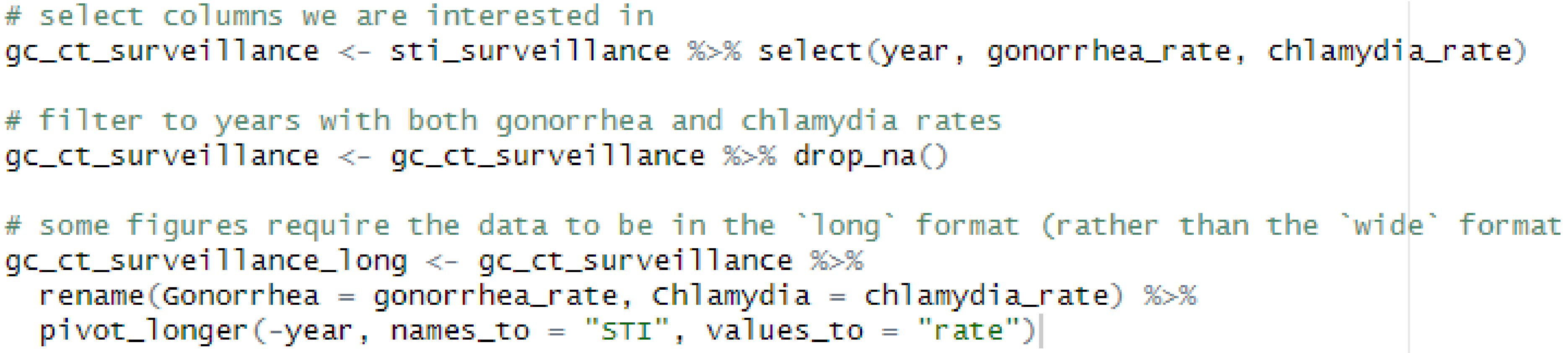


```r
# read input data
# Table 1 from CDC's Sexually Transmitted Surveillance, 2023. Found here: https://www.c
# na = c("NR", "-") converts not reported designations from the table to NA
sti_surveillance <- read_csv("data/CDC_STI_Surveillance_2023.csv", na = c("NR", "-"))

# clean up column names with janitor package
sti_surveillance_clean <- sti_surveillance %>% clean_names()

# Compare the column names before and after using the command clean_names
colnames(sti_surveillance)
colnames(sti_surveillance_clean)
```

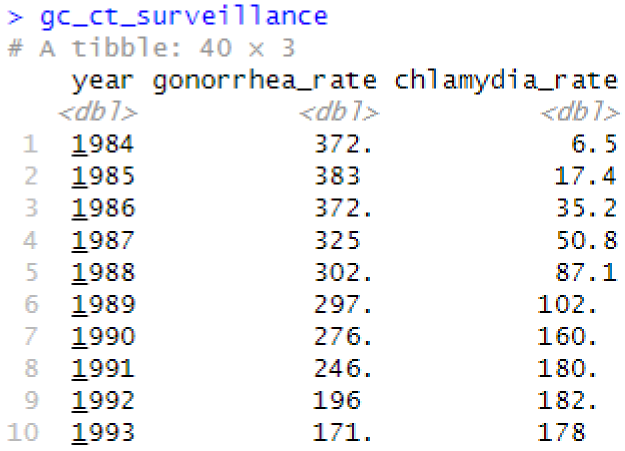


```
> colnames(sti_surveillance)
 [1] "Year"                                           "Total Syphilis Cases"                           "Total Syphilis Rate"
 [4] "Congenital Syphilis Cases"                      "Congenital Syphilis Rate"                       "Primary and Secondary Syphilis Cases"
 [7] "Primary and Secondary Syphilis Rate"            "Early Non-Primary Non-Secondary Syphilis Cases" "Early Non-Primary Non-Secondary Syphilis Rate"
[10] "Unknown Duration or Late Syphilis Cases"        "Unknown Duration or Late Syphilis Rate"         "Gonorrhea Cases"
[13] "Gonorrhea Rate"                                 "Chlamydia Cases"                                "Chlamydia Rate"
[16] "Chancroid Cases"                                "Chancroid Rate"
> colnames(sti_surveillance_clean)
 [1] "year"                                           "total_syphilis_cases"                           "total_syphilis_rate"
 [4] "congenital_syphilis_cases"                      "congenital_syphilis_rate"                       "primary_and_secondary_syphilis_cases"
 [7] "primary_and_secondary_syphilis_rate"            "early_non_primary_non_secondary_syphilis_cases" "early_non_primary_non_secondary_syphilis_rate"
[10] "unknown_duration_or_late_syphilis_cases"        "unknown_duration_or_late_syphilis_rate"         "gonorrhea_cases"
[13] "gonorrhea_rate"                                 "chlamydia_cases"                                "chlamydia_rate"
[16] "chancroid_cases"                                "chancroid_rate"
> 
```

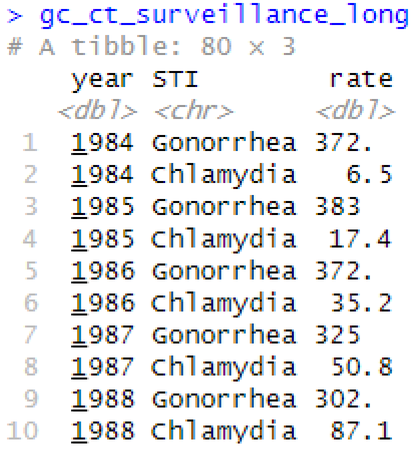
Reformatting Clean Data



```
# select columns we are interested in
gc_ct_surveillance <- sti_surveillance %>% select(year, gonorrhea_rate, chlamydia_rate)

# filter to years with both gonorrhea and chlamydia rates
gc_ct_surveillance <- gc_ct_surveillance %>% drop_na()

# some figures require the data to be in the `long` format (rather than the `wide` format
gc_ct_surveillance_long <- gc_ct_surveillance %>%
  rename(Gonorrhea = gonorrhea_rate, Chlamydia = chlamydia_rate) %>%
  pivot_longer(-year, names_to = "STI", values_to = "rate")
```



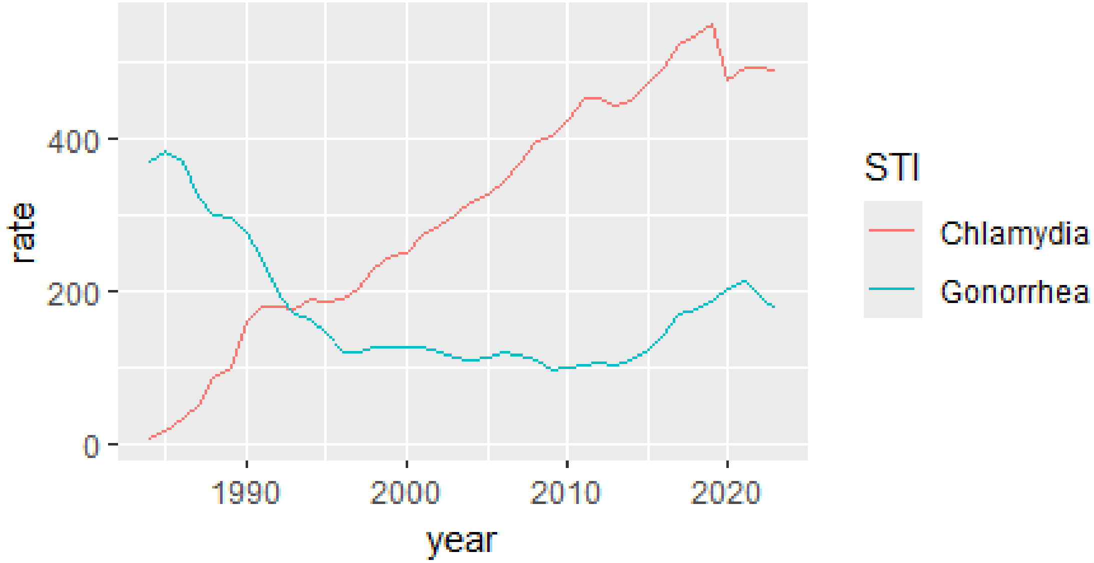
```
> gc_ct_surveillance
# A tibble: 40 × 3
    year gonorrhea_rate chlamydia_rate
   <dbl>          <dbl>          <dbl>
 1  1984           372.            6.5
 2  1985           383            17.4
 3  1986           372.           35.2
 4  1987           325            50.8
 5  1988           302.           87.1
 6  1989           297.          102.
 7  1990           276.          160.
 8  1991           246.          180.
 9  1992           196           182.
10  1993           171.          178
```



```
> gc_ct_surveillance_long
# A tibble: 80 × 3
    year STI        rate
   <dbl> <chr>     <dbl>
 1  1984 Gonorrhea 372.
 2  1984 Chlamydia   6.5
 3  1985 Gonorrhea 383
 4  1985 Chlamydia  17.4
 5  1986 Gonorrhea 372.
 6  1986 Chlamydia  35.2
 7  1987 Gonorrhea 325
 8  1987 Chlamydia  50.8
 9  1988 Gonorrhea 302.
10  1988 Chlamydia  87.1
```

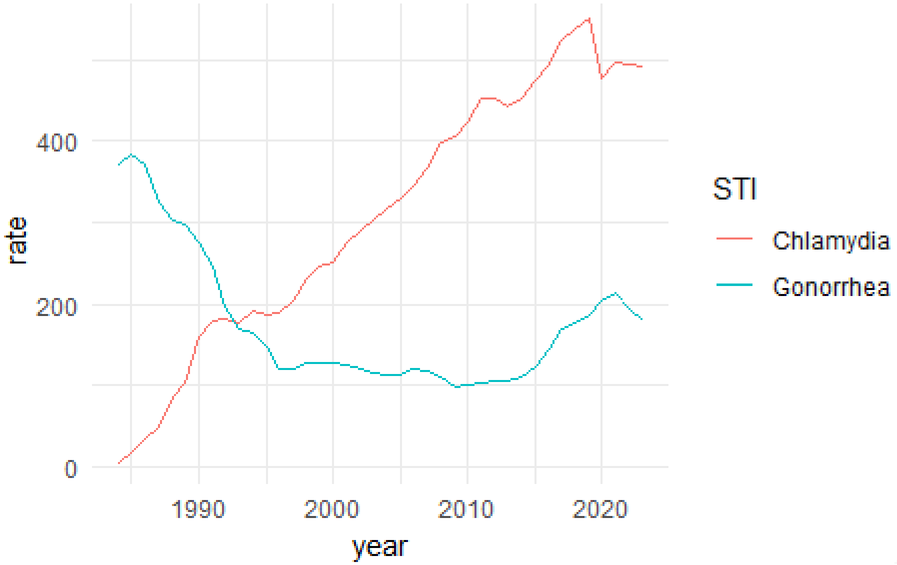
Minimal Plot

```
plot <- gc_ct_surveillance_long %>%
       ggplot(aes(x=year, y=rate, color=STI)) +
       geom_line()
```



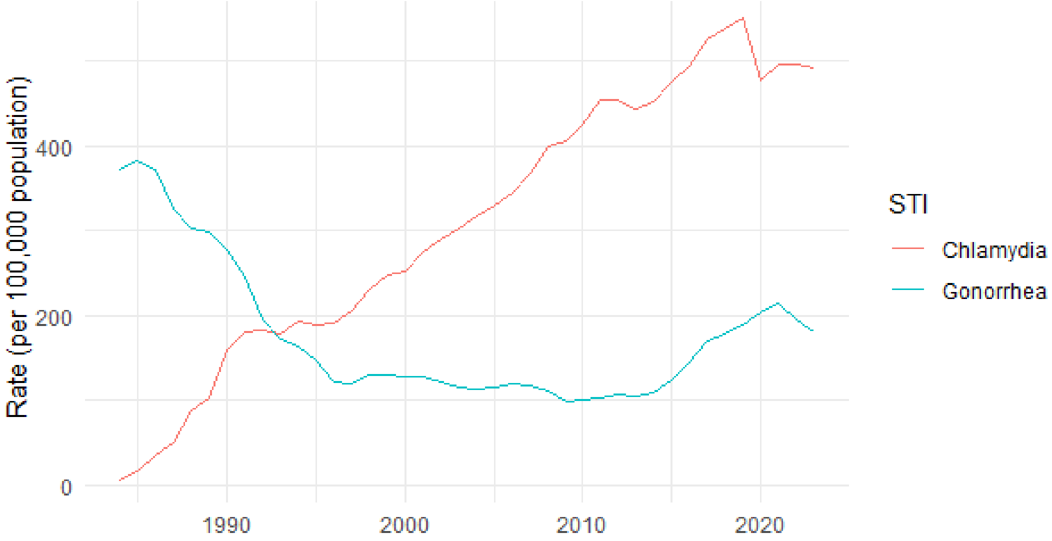
Adjusting Theme

```
plot <- gc_ct_surveillance_long %>%
        ggplot(aes(x=year, y=rate, color=STI)) +
        geom_line() + theme_minimal()
```



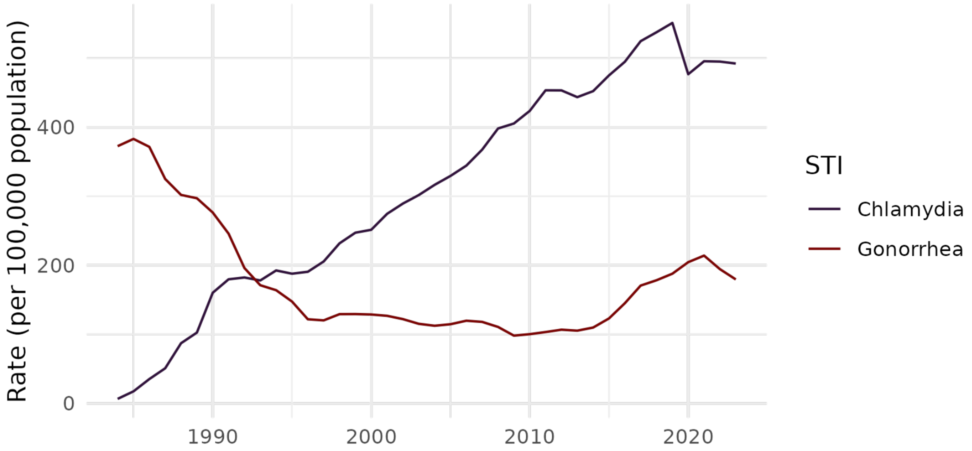
Changing Axis Labels

```
plot <- gc_ct_surveillance_long %>%
        ggplot(aes(x=year, y=rate, color=STI)) +
        geom_line() +
        theme_minimal() +
        ylab("Rate (per 100,000 population)") +
        xlab("")
```

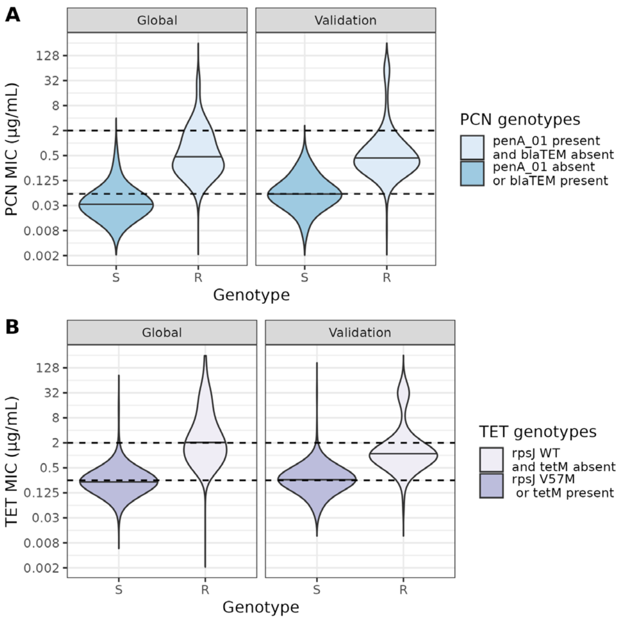


Changing Color Scheme

```
plot <- gc_ct_surveillance_long %>%
        ggplot(aes(x=year, y=rate, color=STI)) +
        geom_line() +
        theme_minimal() +
        ylab("Rate (per 100,000 population)") +
        xlab("") +
        scale_color_viridis(discrete = "T", option = "turbo")
```

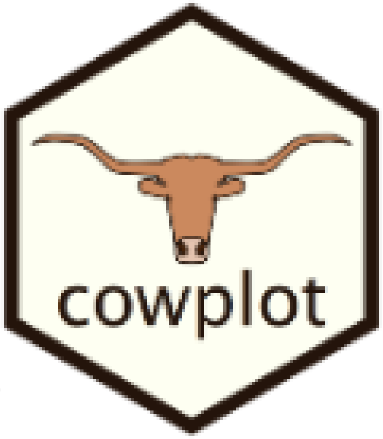


Antimicrobial Susceptibility: Violin Plot

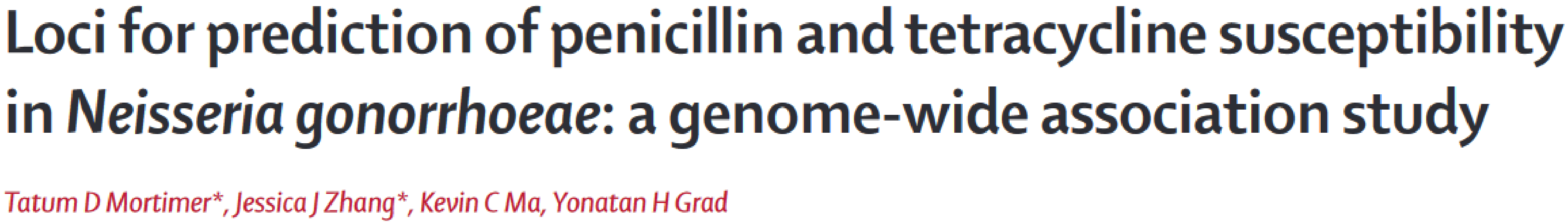


New Packages

- cowplot: plot annotation



Input Data

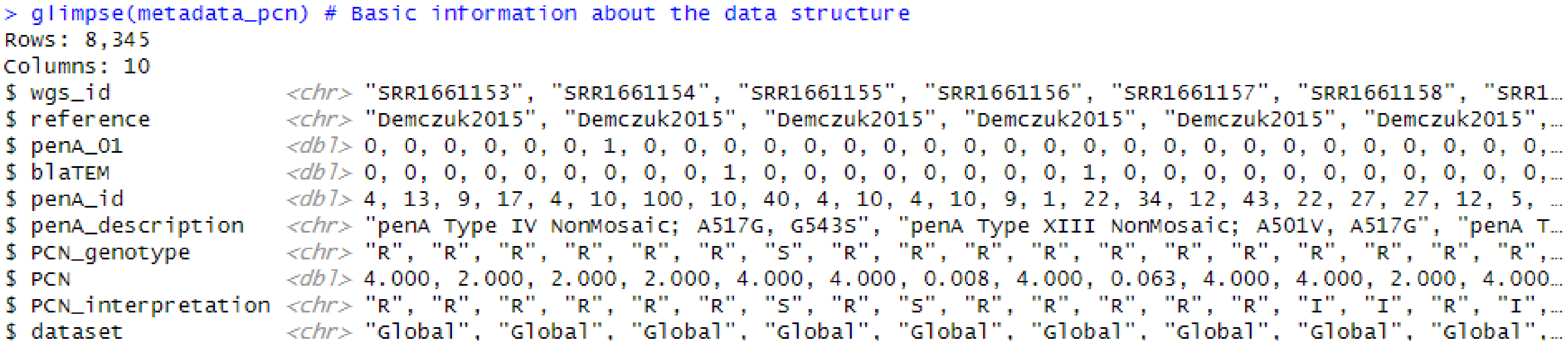


Loci for prediction of penicillin and tetracycline susceptibility in *Neisseria gonorrhoeae*: a genome-wide association study

Tatum D Mortimer*, Jessica J Zhang*, Kevin C Ma, Yonatan H Grad

Reading and Cleaning Input Data

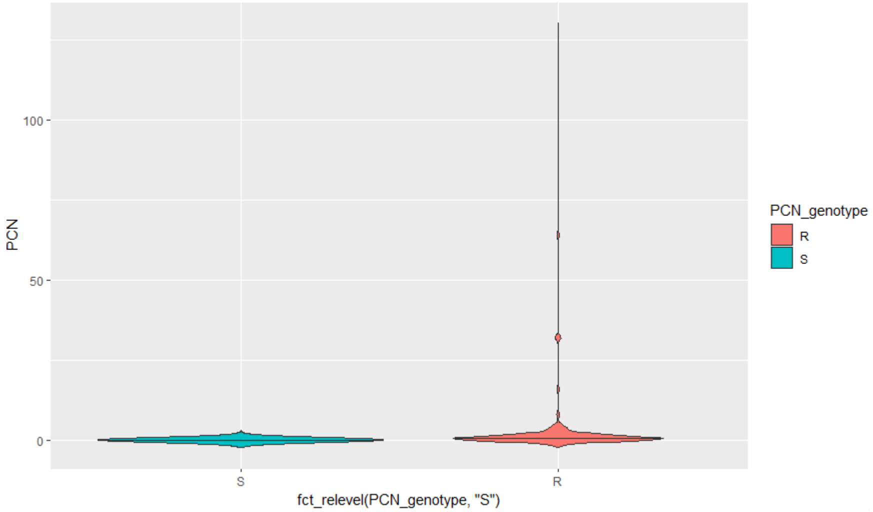
```
metadata_pcn <- read_csv("data/MortimerZhang2022_penicillin_genotypes_phenotypes.csv”)
metadata_tet <- read_csv("data/MortimerZhang2022_tetracycline_genotypes_phenotypes.csv")
view(metadata_pcn)
glimpse(metadata_pcn)
```



```
> glimpse(metadata_pcn) # Basic information about the data structure
Rows: 8,345
Columns: 10
$ wgs_id             <chr> "SRR1661153", "SRR1661154", "SRR1661155", "SRR1661156", "SRR1661157", "SRR1661158", "SRR1…
$ reference          <chr> "Demczuk2015", "Demczuk2015", "Demczuk2015", "Demczuk2015", "Demczuk2015", "Demczuk2015",…
$ penA_01            <dbl> 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
$ blaTEM             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
$ penA_id            <dbl> 4, 13, 9, 17, 4, 10, 100, 10, 40, 4, 10, 4, 10, 9, 1, 22, 34, 12, 43, 22, 27, 27, 12, 5, …
$ penA_description   <chr> "penA Type IV NonMosaic; A517G, G543S", "penA Type XIII NonMosaic; A501V, A517G", "penA T…
$ PCN_genotype       <chr> "R", "R", "R", "R", "R", "R", "S", "R", "R", "R", "R", "R", "R", "R", "R", "R", "R", "R",…
$ PCN                <dbl> 4.000, 2.000, 2.000, 2.000, 4.000, 4.000, 0.008, 4.000, 0.063, 4.000, 4.000, 2.000, 4.000…
$ PCN_interpretation <chr> "R", "R", "R", "R", "R", "R", "S", "R", "S", "R", "R", "R", "R", "R", "I", "I", "R", "I",…
$ dataset            <chr> "Global", "Global", "Global", "Global", "Global", "Global", "Global", "Global", "Global",…
```

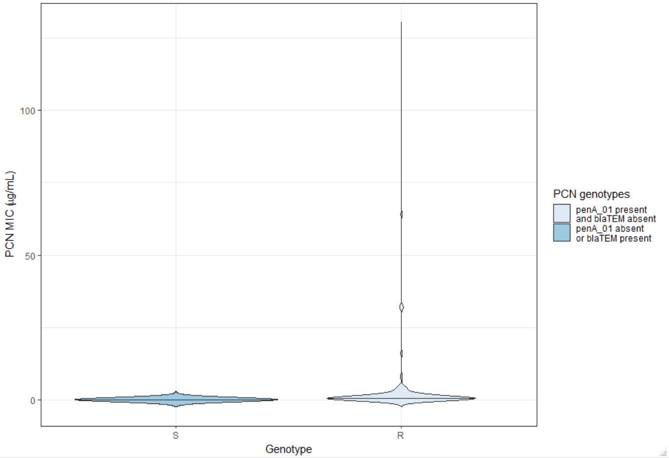
Minimal Plot

```
pcn_plot <- metadata_pcn %>%
    ggplot(aes(x = fct_relevel(PCN_genotype, "S"),
               y = PCN,
               fill = PCN_genotype)) +
    geom_violin(bw=0.8, draw_quantiles=c(0.5), trim=FALSE)
```

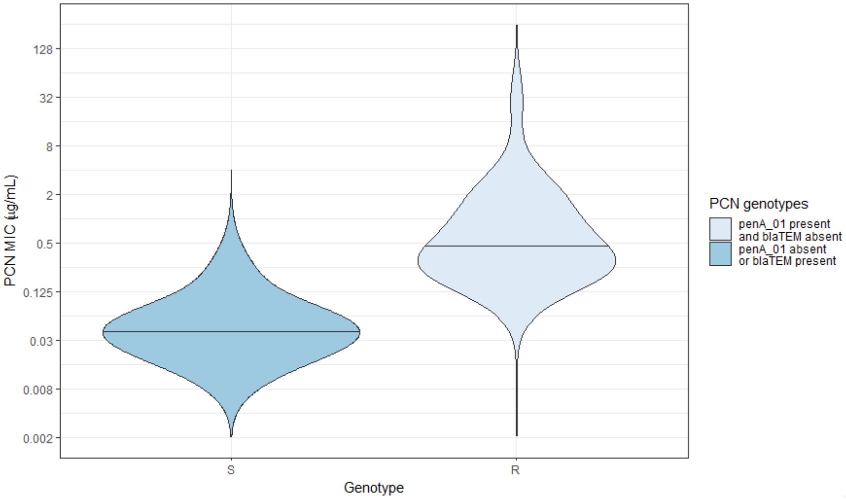


Adjusting Theme, Axis Labels, and Color Palette

```
pcn_plot <- metadata_pcn %>%
    ggplot(aes(x = fct_relevel(PCN_genotype, "S"),
               y = PCN,
               fill = PCN_genotype)) +
    geom_violin(bw=0.8, draw_quantiles=c(0.5), trim=FALSE) +
    theme_bw() +
    xlab("Genotype") +
    ylab(expression(paste("PCN MIC (", mu, "g/mL)", ))) +
    scale_fill_brewer(name = "PCN genotypes",
                      labels = c("penA_01 present \nand
blaTEM absent", "penA_01 absent \nor blaTEM present"))
```

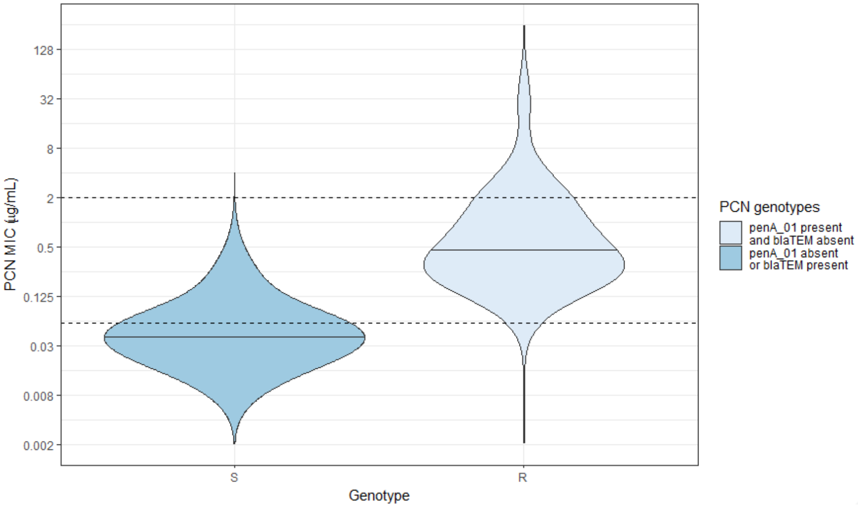


Changing Axis Scale



```
pcn_plot <- metadata_pcn %>%
    ggplot(aes(x = fct_relevel(PCN_genotype, "S"),
               y = PCN,
               fill = PCN_genotype)) +
    geom_violin(bw=0.8, draw_quantiles=c(0.5), trim=FALSE) +
    theme_bw() +
    xlab("Genotype") +
    ylab(expression(paste("PCN MIC (", mu, "g/mL)", ))) +
    scale_fill_brewer(name = "PCN genotypes",
                     labels = c("penA_01 present \nand blaTEM absent", "penA_01 absent \nor
blaTEM present")) +
   scale_y_continuous(trans = 'log2’,
        limits = c(0.002, 256),
        breaks = c(2^-9, 2^-7, 2^-5, 2^-3, 2^-1, 2^1, 2^3, 2^5, 2^7),
        labels = c(0.002, 0.008, 0.03, 0.125, 0.5, 2, 8, 32, 128))
```

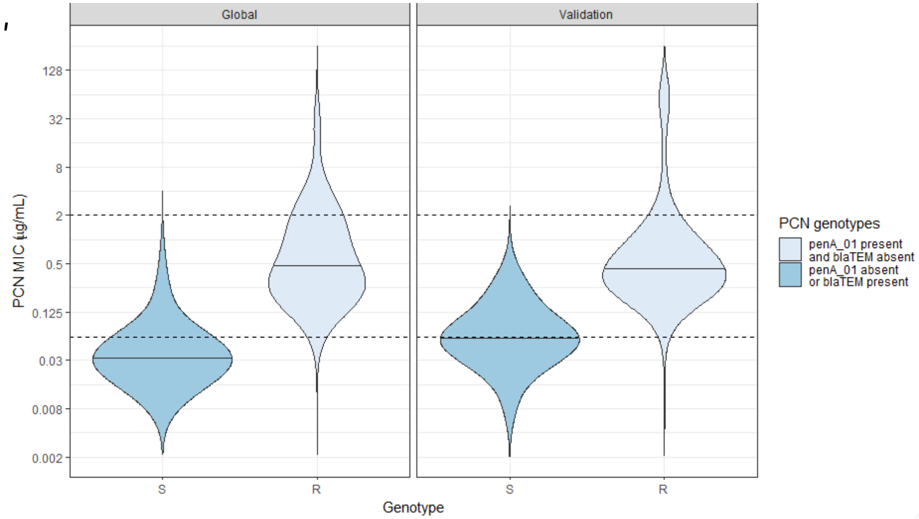
Adding Horizontal Lines



```
pcn_plot <- metadata_pcn %>%
    ggplot(aes(x = fct_relevel(PCN_genotype, "S"),
               y = PCN,
               fill = PCN_genotype)) +
    geom_violin(bw=0.8, draw_quantiles=c(0.5), trim=FALSE) +
    theme_bw() +
    xlab("Genotype") +
    ylab(expression(paste("PCN MIC (", mu, "g/mL)", ))) +
    scale_fill_brewer(name = "PCN genotypes",
                      labels = c("penA_01 present \nand blaTEM absent",
                                 "penA_01 absent \nor blaTEM present")) +
  scale_y_continuous(trans = 'log2’,
     limits = c(0.002, 256),
     breaks = c(2^-9, 2^-7, 2^-5, 2^-3, 2^-1, 2^1, 2^3, 2^5, 2^7),
     labels = c(0.002, 0.008, 0.03, 0.125, 0.5, 2, 8, 32, 128)) +
 geom_hline(yintercept = 0.06, linetype = "dashed") +
 geom_hline(yintercept = 2, linetype = "dashed")
```

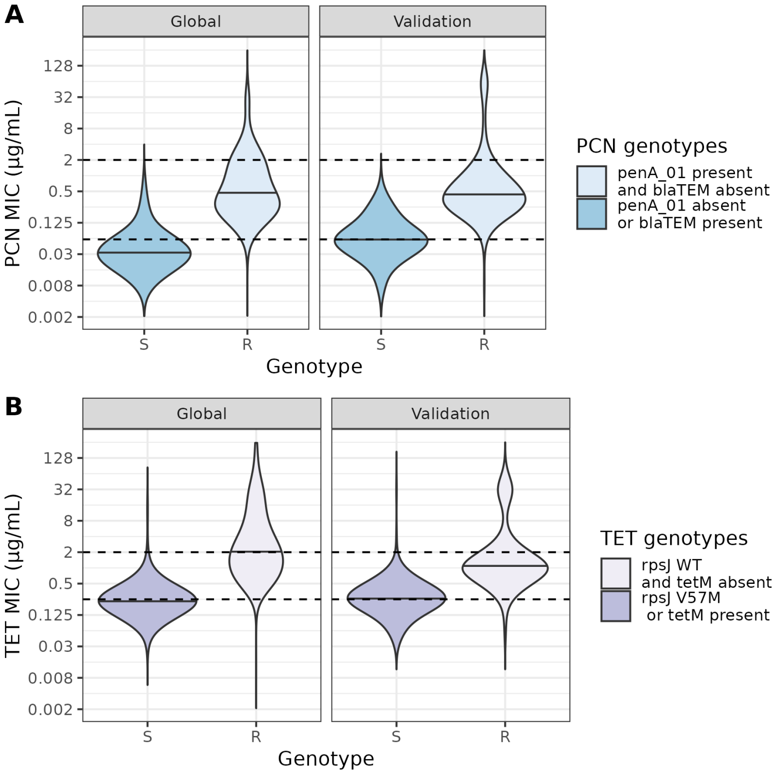
Facet by variable

```
pcn_plot <- metadata_pcn %>%
    ggplot(aes(x = fct_relevel(PCN_genotype, "S"),
               y = PCN,
               fill = PCN_genotype)) +
    geom_violin(bw=0.8, draw_quantiles=c(0.5), trim=FALSE) +
    theme_bw() +
    xlab("Genotype") +
    ylab(expression(paste("PCN MIC (", mu, "g/mL)", ))) +
    scale_fill_brewer(name = "PCN genotypes",
                     labels = c("penA_01 present \nand blaTEM absent",
                                "penA_01 absent \nor blaTEM present")) +
  scale_y_continuous(trans = 'log2’,
     limits = c(0.002, 256),
     breaks = c(2^-9, 2^-7, 2^-5, 2^-3, 2^-1, 2^1, 2^3, 2^5, 2^7),
     labels = c(0.002, 0.008, 0.03, 0.125, 0.5, 2, 8, 32, 128)) +
  geom_hline(yintercept = 0.06, linetype = "dashed") +
  geom_hline(yintercept = 2, linetype = "dashed") +
  facet_wrap(~dataset)
```



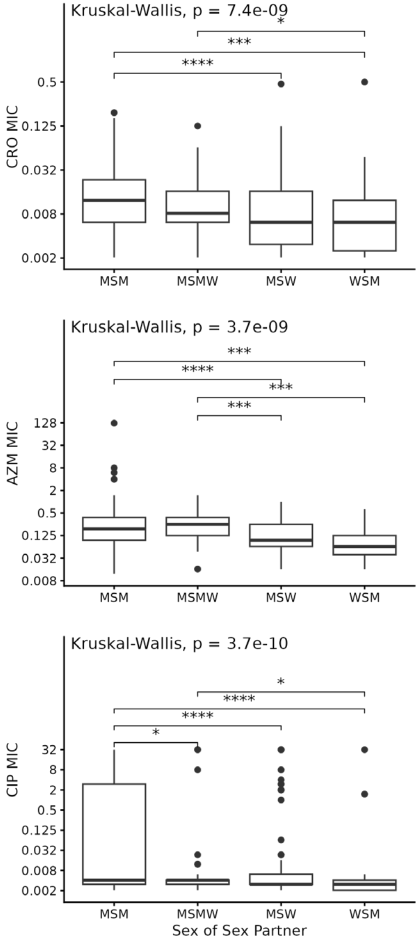
Making a multi-panel plot

```
p <- plot_grid(pcn_plot, tet_plot, labels = c("A","B"), nrow = 2)
```



STI&HIV2025
WORLD CONGRESS

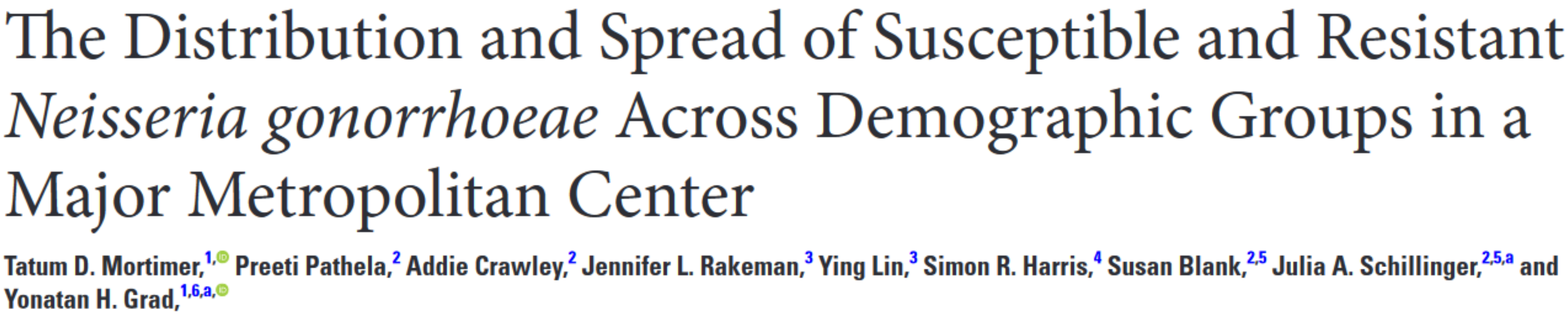
AMR and Sexual Behavior: Box Plot



New Packages

- ggpubr: annotate plot with statistical tests

Input Data

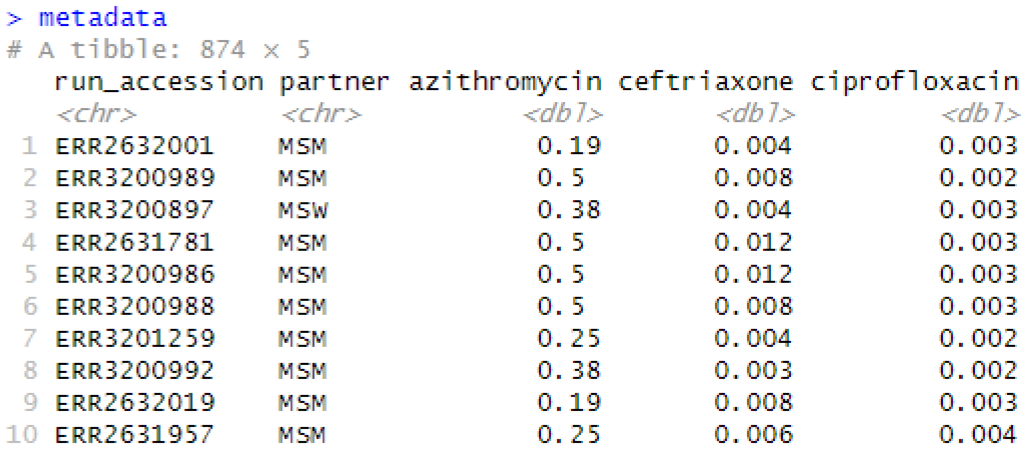


The Distribution and Spread of Susceptible and Resistant *Neisseria gonorrhoeae* Across Demographic Groups in a Major Metropolitan Center

Tatum D. Mortimer,[1] Preeti Pathela,[2] Addie Crawley,[2] Jennifer L. Rakeman,[3] Ying Lin,[3] Simon R. Harris,[4] Susan Blank,[2,5] Julia A. Schillinger,[2,5,a] and Yonatan H. Grad,[1,6,a]

Reading Input Data and Setting Theme

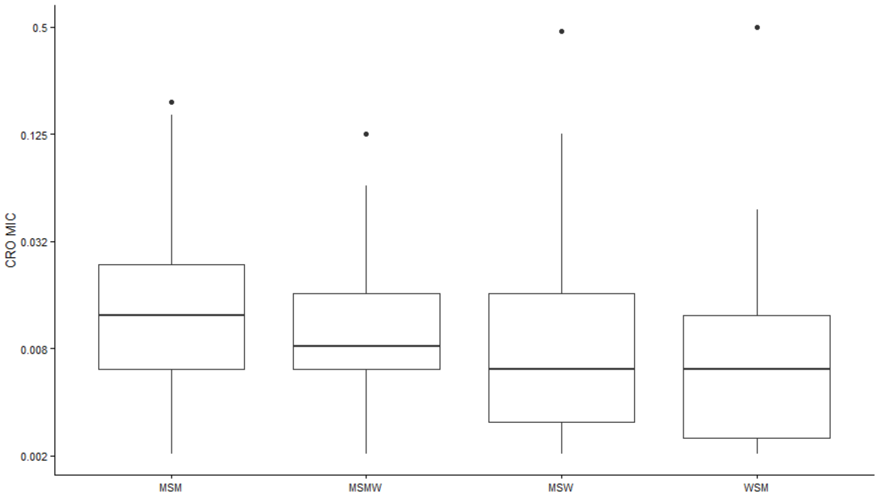
```
theme_set(theme_cowplot(font_size=10))

metadata <- read_csv("data/Mortimer2020_sexualbehavior_AMR.csv")
```



```
> metadata
# A tibble: 874 × 5
   run_accession partner azithromycin ceftriaxone ciprofloxacin
   <chr>         <chr>          <dbl>       <dbl>         <dbl>
 1 ERR2632001    MSM             0.19       0.004         0.003
 2 ERR3200989    MSM             0.5        0.008         0.002
 3 ERR3200897    MSW             0.38       0.004         0.003
 4 ERR2631781    MSM             0.5        0.012         0.003
 5 ERR3200986    MSM             0.5        0.012         0.003
 6 ERR3200988    MSM             0.5        0.008         0.003
 7 ERR3201259    MSM             0.25       0.004         0.002
 8 ERR3200992    MSM             0.38       0.003         0.002
 9 ERR2632019    MSM             0.19       0.008         0.003
10 ERR2631957    MSM             0.25       0.006         0.004
```

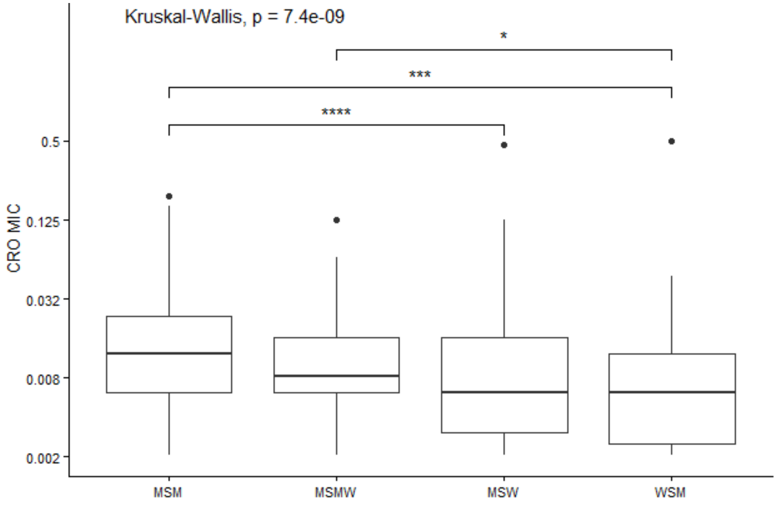
Base Plot

```
partner.cro <- metadata %>%  ggplot(aes(x = partner, y = ceftriaxone)) +
    geom_boxplot() +
    scale_y_continuous(trans='log2’,
        breaks = c(2^(-9), 2^(-7), 2^(-5), 2^(-3), 2^(-1)),
        labels = c("0.002", "0.008", "0.032", "0.125", "0.5")) +
        labs(x = "", y = "CRO MIC")
```



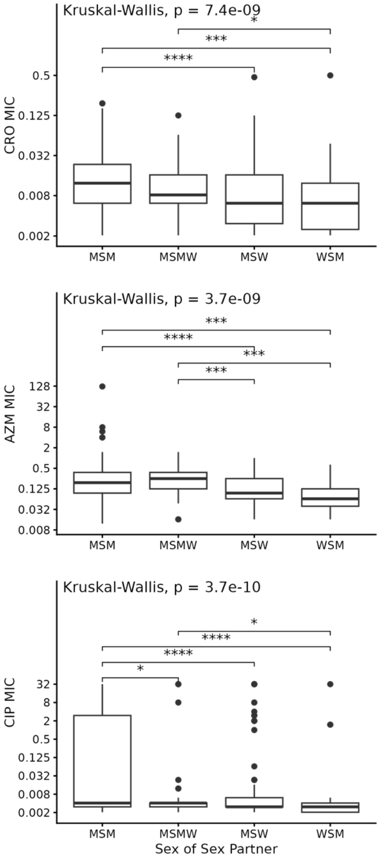
Adding Statistical Comparisons

```
comparisons.cro <- list(c("MSM", "MSW"), c("MSM", "WSM"), c("MSMW", "WSM"))

partner.cro <- metadata %>%  ggplot(aes(x = partner, y = ceftriaxone)) +
    geom_boxplot() +
    scale_y_continuous(trans='log2’,
        breaks = c(2^(-9), 2^(-7), 2^(-5), 2^(-3), 2^(-1)),
        labels = c("0.002", "0.008", "0.032", "0.125", "0.5")) +
        labs(x = "", y = "CRO MIC") +
    stat_compare_means(comparisons = comparisons.cro, label = "p.signif") +
    stat_compare_means(label.y = 2)
```



Combining Plots into Panels

```
partner.mic <- plot_grid(partner.cro,
                         partner.azi,
                         partner.cip,
                         ncol = 1,
                         align = "v",
                         axis = "tb")
```


Kruskal-Wallis, p = 7.4e-09
*
***
****
CRO MIC
MSM
MSMW
MSW
WSM
Kruskal-Wallis, p = 3.7e-09
***
****
***
***
AZM MIC
MSM
MSMW
MSW
WSM
Kruskal-Wallis, p = 3.7e-10
*
****
****
*
CIP MIC
MSM
MSMW
MSW
WSM
Sex of Sex Partner