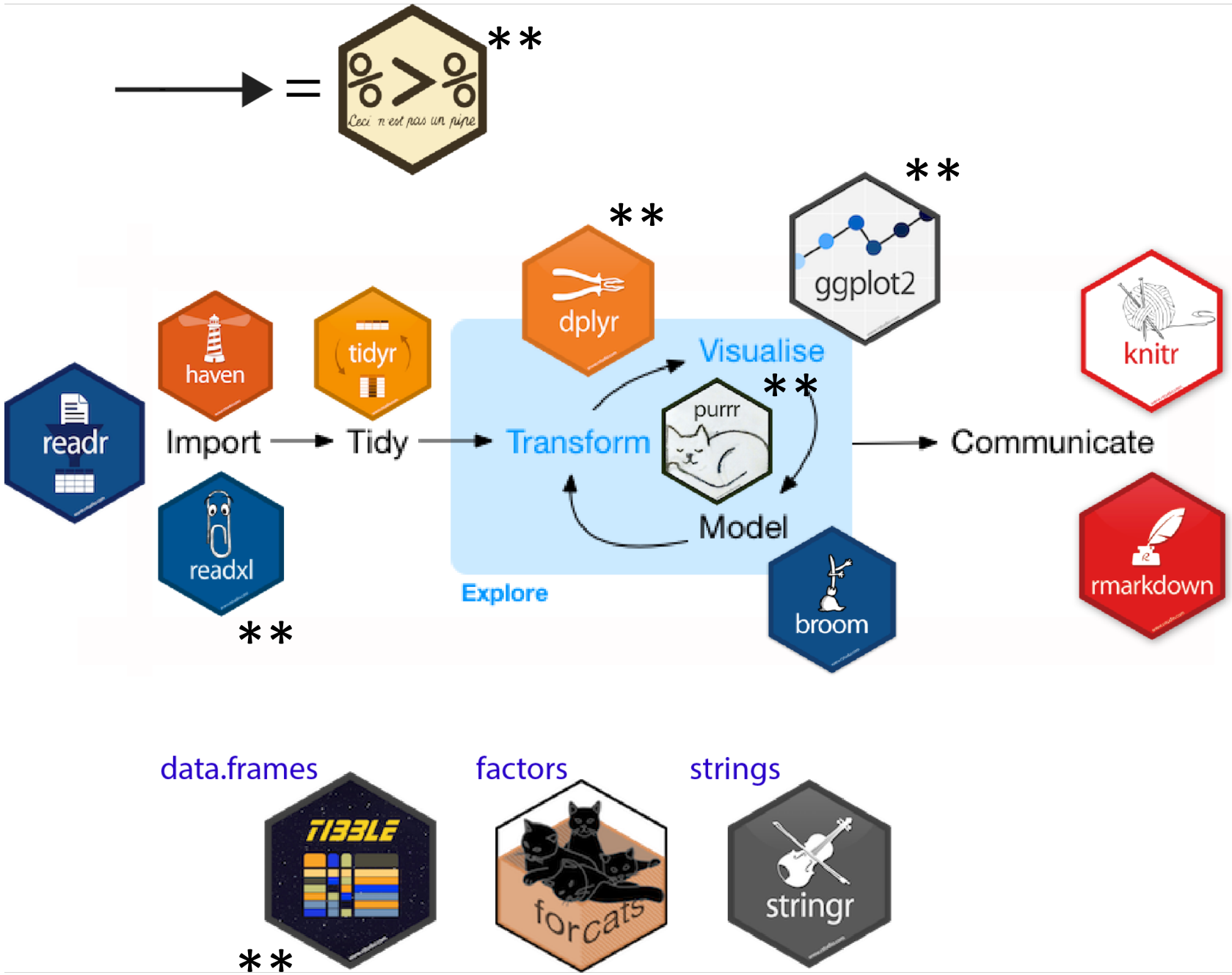
Confessions and Countermeasures:
Some of my not ideal R habits and how the Tidyverse resolved them

Rachael Workman

PhD student, BCMB



https://osf.io/69gub/

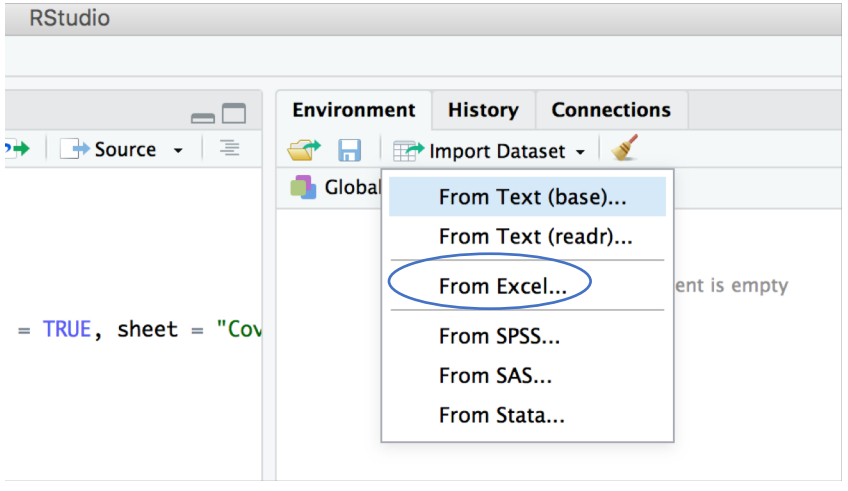
This year I…
Made data import harder than it had to be

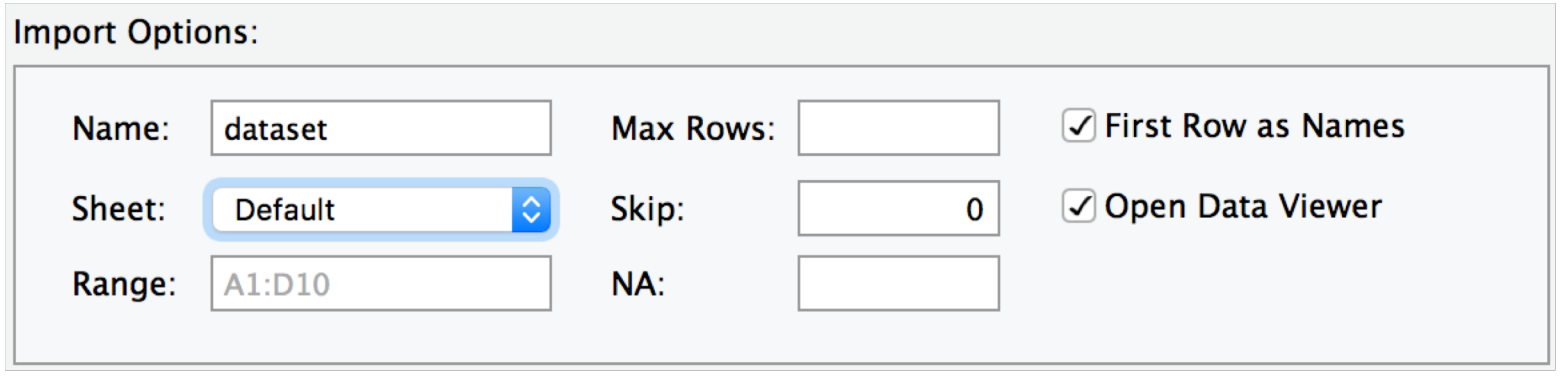
Excel with multiple sheets → Open, select sheet of interest → Save worksheet as CSV → Import using base R into dataframe

VS

```
library(tidyverse)
excel = readxl::read_excel("/path/to/my/xlsx", col_names = TRUE, sheet = “the_one”)
```

OR





This year I…
Made data import harder than it had to be

Benefits to tibbles over dataframes

1. Tibbles print nicely, they show the data type of each column, and if you subset one, it returns another tibble.

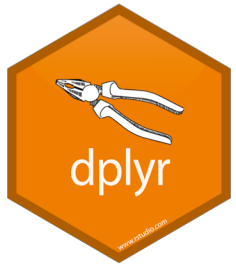
```
> plus1625.g
# A tibble: 63,451 x 8
   chrom                start   end    name score strand id    strain
   <chr>                <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>
 1 Newman_NC_009641.1.dna   639   669  109726    25 +      Plus  Newman
 2 Newman_NC_009641.1.dna   639   669  232055    25 +      Plus  Newman
 3 Newman_NC_009641.1.dna   639   669  274783    25 +      Plus  Newman
 4 Newman_NC_009641.1.dna   866   896   96673    25 +      Plus  Newman
 5 Newman_NC_009641.1.dna   866   896  180079    25 +      Plus  Newman
 6 Newman_NC_009641.1.dna   866   896  197887    25 +      Plus  Newman
 7 Newman_NC_009641.1.dna   959   989   46355    25 +      Plus  Newman
 8 Newman_NC_009641.1.dna  1310  1340   50368    25 -      Plus  Newman
 9 Newman_NC_009641.1.dna  1310  1340  112627    25 -      Plus  Newman
10 Newman_NC_009641.1.dna  1310  1340  139917    25 -      Plus  Newman
# … with 63,441 more rows
>
```

vs

```
> as.data.frame(plus1625.g)
                    chrom start   end    name score strand   id strain
1   Newman_NC_009641.1.dna   639   669  109726    25      + Plus Newman
2   Newman_NC_009641.1.dna   639   669  232055    25      + Plus Newman
3   Newman_NC_009641.1.dna   639   669  274783    25      + Plus Newman
4   Newman_NC_009641.1.dna   866   896   96673    25      + Plus Newman
5   Newman_NC_009641.1.dna   866   896  180079    25      + Plus Newman
6   Newman_NC_009641.1.dna   866   896  197887    25      + Plus Newman
7   Newman_NC_009641.1.dna   959   989   46355    25      + Plus Newman
8   Newman_NC_009641.1.dna  1310  1340   50368    25      - Plus Newman
9   Newman_NC_009641.1.dna  1310  1340  112627    25      - Plus Newman
10  Newman_NC_009641.1.dna  1310  1340  139917    25      - Plus Newman
```

This year I…

Did calculations in Excel and reimported my dataset



Excel with multiple sheets → Open, select sheet of interest → Save worksheet as CSV → Import using base R into dataframe→ Realized I needed to compute the sum of two columns → opened Excel file → calculated sum in Excel → resaved as CSV → reimported into R

VS

```
library(tidyverse)
excel = readxl::read_excel("/path/to/my/xlsx", col_names = TRUE, sheet = “the_one”) %>%
  mutate(newcol = col1 + col2)
```

Column name of new column

Two numerical columns to add together

This year I …
Saved too many intermediate objects

- The pipe operator is your friend

```
phiNM4 = read_tsv("190216_phage_116.spacers.fa.sam.sorted.bam.bed")
colnames(phiNM4) = c("chrom", "start", "end", "name", "score", "strand")
phage = "phiNM4"
phiNM4_2 = cbind(phiNM4, phage)
```

vs

```
phiNM4 = read_tsv("190216_phage_116.spacers.fa.sam.sorted.bam.bed") %>%
  `colnames<-`(c("chrom", "start", "end", "name", "score", "strand")) %>%
  mutate(phage="phiNM4")
```

This year I…

Read in a bunch of similar datasets one at a time


```
phi11 = read_tsv("190216_phage_68.spacers.fa.sam.sorted.bam.bed") %>%
  `colnames<-`(c("chrom", "start", "end", "name", "score", "strand")) %>%
  mutate(phage="phi11")

phiNM1 = read_tsv("190216_phage_79.spacers.fa.sam.sorted.bam.bed") %>%
  `colnames<-`(c("chrom", "start", "end", "name", "score", "strand")) %>%
  mutate(phage="phiNM1")

phiNM2 = read_tsv("190216_phage_121.spacers.fa.sam.sorted.bam.bed") %>%
  `colnames<-`(c("chrom", "start", "end", "name", "score", "strand")) %>%
  mutate(phage="phiNM2")

phiNM4 = read_tsv("190216_phage_116.spacers.fa.sam.sorted.bam.bed") %>%
  `colnames<-`(c("chrom", "start", "end", "name", "score", "strand")) %>%
  mutate(phage="phiNM4")
```

……for 12 files, which I then concatenated…

vs

```
read_plus <- function(flnm) {
  read_tsv(flnm) %>%
    mutate(filename = flnm) %>%
    `colnames<-`c("chrom", "start", "end", "name", "score", "strand"))
}

allruns <-
  list.files(pattern="*.bed",
             full.names = T) %>%
  map_df(~read_plus(.))
```

On that note - why care about reducing duplication?

- “It’s easier to see the intent of your code, because your eyes are drawn to what’s different, not what stays the same.
- It’s easier to respond to changes in requirements. As your needs change, you only need to make changes in one place, rather than remembering to change every place that you copied-and-pasted the code.
- You’re likely to have fewer bugs because each line of code is used in more places.”

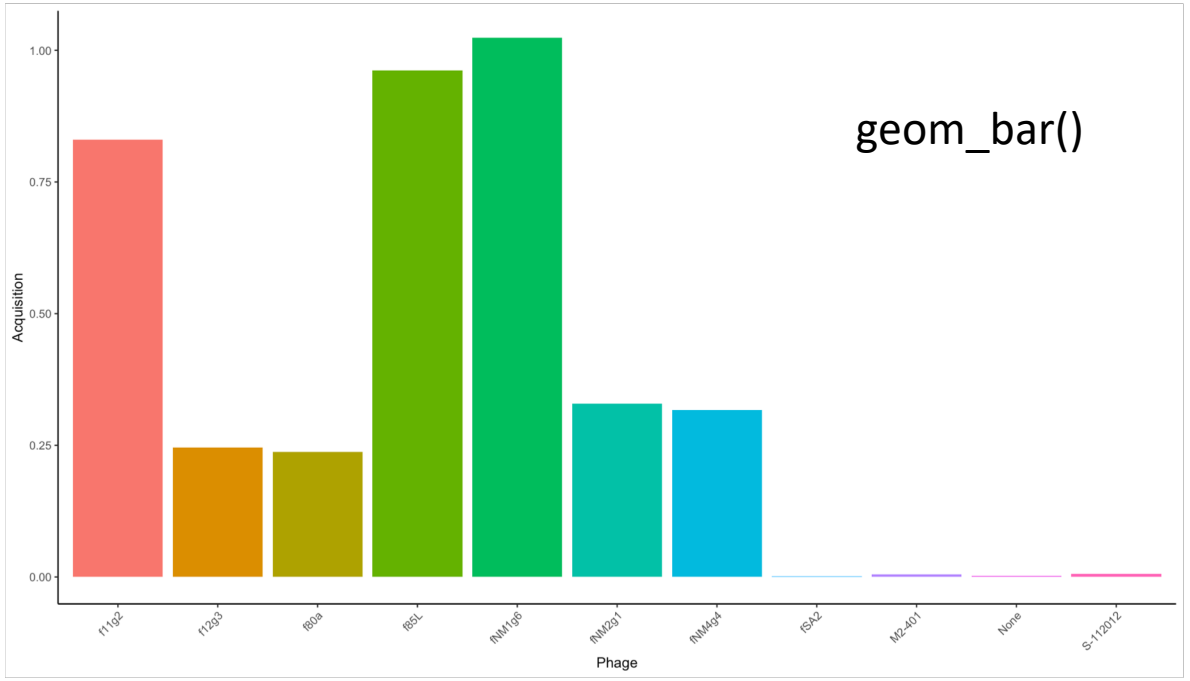
---R for Data Science, Grolemund and Wickham

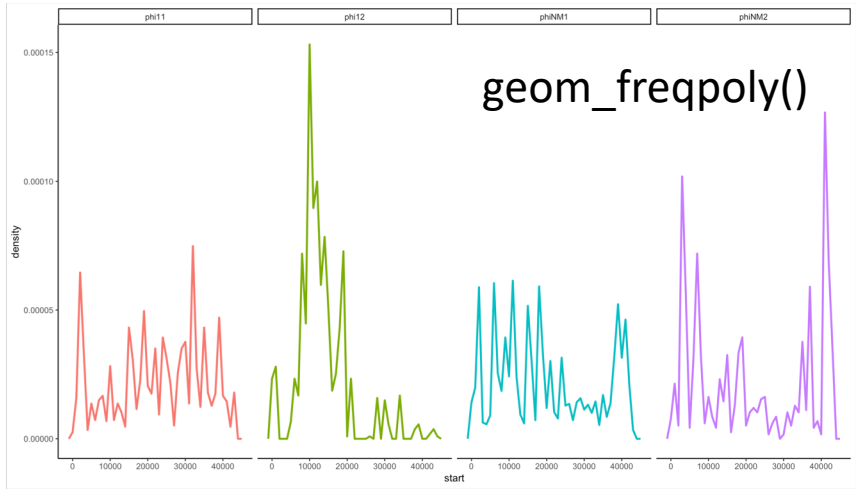
This year I…

Did a lot of plotting using default color schemes

ggplot color options – why go past default?

1. Colorblind-friendly graphs
2. Demonstrate a point
3. Just stand out





Make your own colorblind friendly palette for ggplot


```
# The palette with grey:
cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
# The palette with black:
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
# To use for fills, add
scale_fill_manual(values=cbPalette)
# To use for line and point colors, add
scale_colour_manual(values=cbPalette)
```

More palettes

RColorBrewer

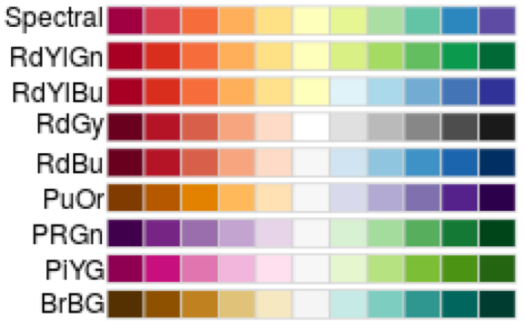

```
install.packages("wesanderson")
```

The Life Aquatic with Steve Zissou (2004)

```
wes_palette("Zissou1")
```

The Royal Tenenbaums (2001)

```
wes_palette("Royal1")
```




```
install.packages("devtools")
devtools::install_github('LaCroixColoR','johannesbjork')
```

```
lacroix_palette("Pamplemousse", n = 50, type = "continuous")
```


```
lacroix_palette("Pamplemousse", type = "discrete")
```