Data Wrangling Homework

Bibek

2025-03-18

Tidyverse is a bunch of packages and fucntion for easiers use case of large datset in R. This assignments will cover follwing headings: -Data wrangling & manipulation -mutate() -select() -filter() -the pipe %>% -summarise() -group_by() -joining -pivoting -Integration with plotting

###Loading the dataset

```r
microbiome.fungi <- read.csv("Sample_data/Bull_richness.csv")
str(microbiome.fungi)
```

```
## 'data.frame':    287 obs. of  16 variables:
##  $ SampleID       : chr  "Corn2017LeafObjective2Collection1T1R1CAH2" "Corn2017LeafObjective2Collecti
##  $ Crop           : chr  "Corn" "Corn" "Corn" "Corn" ...
##  $ Objective      : chr  "Objective 2" "Objective 2" "Objective 2" "Objective 2" ...
##  $ Collection     : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Compartment    : chr  "Leaf" "Leaf" "Leaf" "Leaf" ...
##  $ DateSampled    : chr  "6/26/17" "6/26/17" "6/26/17" "6/26/17" ...
##  $ GrowthStage    : chr  "V6" "V6" "V6" "V6" ...
##  $ Treatment      : chr  "Conv." "Conv." "Conv." "Conv." ...
##  $ Rep            : chr  "R1" "R1" "R1" "R1" ...
##  $ Sample         : chr  "A" "B" "C" "A" ...
##  $ Fungicide      : chr  "C" "C" "C" "F" ...
##  $ Target_organism: chr  "Fungi" "Fungi" "Fungi" "Fungi" ...
##  $ Location       : chr  "Kellogg Biological Station" "Kellogg Biological Station" "Kellogg Biologica
##  $ Experiment     : chr  "LTER" "LTER" "LTER" "LTER" ...
##  $ Year           : int  2017 2017 2017 2017 2017 2017 2017 2017 2017 2017 ...
##  $ richness       : int  9 6 5 7 4 2 3 8 4 4 ...
```

```r
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.2
```

```
## Warning: package 'lubridate' was built under R version 4.4.2
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

#####Select() function- choose certain columns of th data.

```r
microbiome.fungi2 <- select(microbiome.fungi, SampleID, Crop, Compartment:Fungicide, richness)
```

filter() Function for sub-setting data easily.

```r
#Simple filtering
head(filter(microbiome.fungi2, Treatment == "Conv."))
```

```
##                                    SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1CCB3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R1FAC3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R1FBD3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R1FCE3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness
## 1          V6     Conv. R1       A         C        9
## 2          V6     Conv. R1       B         C        6
## 3          V6     Conv. R1       C         C        5
## 4          V6     Conv. R1       A         F        7
## 5          V6     Conv. R1       B         F        4
## 6          V6     Conv. R1       C         F        2
```

```r
#Complex filtering (&)
head(filter(microbiome.fungi2, Treatment == "Conv." & Fungicide == "C"))
```

```
##                                    SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1CCB3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R2CAF3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R2CBG3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R2CCH3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness
## 1          V6     Conv. R1       A         C        9
## 2          V6     Conv. R1       B         C        6
## 3          V6     Conv. R1       C         C        5
## 4          V6     Conv. R2       A         C        3
## 5          V6     Conv. R2       B         C        8
## 6          V6     Conv. R2       C         C        4
```

```r
# Another more complex (|)
head(filter(microbiome.fungi2, Sample == "A" | Sample == "B")) # samples A or B
```

```
##                                    SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1FAC3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R1FBD3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R2CAF3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R2CBG3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness
## 1          V6     Conv.  R1      A         C        9
## 2          V6     Conv.  R1      B         C        6
## 3          V6     Conv.  R1      A         F        7
## 4          V6     Conv.  R1      B         F        4
## 5          V6     Conv.  R2      A         C        3
## 6          V6     Conv.  R2      B         C        8
```

mutate() Helps to create new columns quickly.

```r
# Create a new column called logRich
head(mutate(microbiome.fungi2, logRich = log(richness)))
```

```
##                                    SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1CCB3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R1FAC3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R1FBD3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R1FCE3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness   logRich
## 1          V6     Conv.  R1      A         C        9 2.1972246
## 2          V6     Conv.  R1      B         C        6 1.7917595
## 3          V6     Conv.  R1      C         C        5 1.6094379
## 4          V6     Conv.  R1      A         F        7 1.9459101
## 5          V6     Conv.  R1      B         F        4 1.3862944
## 6          V6     Conv.  R1      C         F        2 0.6931472
```

```r
#Previous way to do it
#microbiome.fungi2$logRich <- log(microbiome.fungi2$richness)

# Creating a new column with combined Crop and Treatment
head(mutate(microbiome.fungi2, Crop_Treatment = paste(Crop, Treatment)))
```

```
##                                    SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1CCB3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R1FAC3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R1FBD3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R1FCE3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness Crop_Treatment
## 1          V6     Conv.  R1      A         C        9     Corn Conv.
## 2          V6     Conv.  R1      B         C        6     Corn Conv.
## 3          V6     Conv.  R1      C         C        5     Corn Conv.
## 4          V6     Conv.  R1      A         F        7     Corn Conv.
## 5          V6     Conv.  R1      B         F        4     Corn Conv.
## 6          V6     Conv.  R1      C         F        2     Corn Conv.
```

The pipe, %>% Helps to combine the various function together.

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>% # selecting columns
  filter(Treatment == "Conv.") %>% # sub-setting to only include the conventional treatment
  mutate(logRich = log(richness)) %>% # creating a new column log richness
  head() # displaying
```

```
##                                     SampleID Crop Compartment DateSampled
## 1 Corn2017LeafObjective2Collection1T1R1CAH2 Corn        Leaf     6/26/17
## 2 Corn2017LeafObjective2Collection1T1R1CBA3 Corn        Leaf     6/26/17
## 3 Corn2017LeafObjective2Collection1T1R1CCB3 Corn        Leaf     6/26/17
## 4 Corn2017LeafObjective2Collection1T1R1FAC3 Corn        Leaf     6/26/17
## 5 Corn2017LeafObjective2Collection1T1R1FBD3 Corn        Leaf     6/26/17
## 6 Corn2017LeafObjective2Collection1T1R1FCE3 Corn        Leaf     6/26/17
##   GrowthStage Treatment Rep Sample Fungicide richness   logRich
## 1          V6     Conv.  R1      A         C        9 2.1972246
## 2          V6     Conv.  R1      B         C        6 1.7917595
## 3          V6     Conv.  R1      C         C        5 1.6094379
## 4          V6     Conv.  R1      A         F        7 1.9459101
## 5          V6     Conv.  R1      B         F        4 1.3862944
## 6          V6     Conv.  R1      C         F        2 0.6931472
```

summarise() Helps to find out means and standard deviation or errors.

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>% # selecting columns
  filter(Treatment == "Conv.") %>% # sub-setting to only include the conventional treatment
  mutate(logRich = log(richness)) %>% # creating a new column log richness
  summarise(Mean.rich = mean(logRich)) # calculate overall mean log richness within the conventionally
```

```
##   Mean.rich
## 1  2.304395
```

```
# Can also connect multiple summary statistics
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>% # selecting columns
  filter(Treatment == "Conv.") %>% # sub-setting to only include the conventional treatment
  mutate(logRich = log(richness)) %>% # creating a new column log richness
  summarise(Mean.rich = mean(logRich),
            n= n(),  # calsulates the number of n
            sd.dev = sd(logRich)) %>%
  mutate(std.err = sd.dev/sqrt(n))
```

```
##   Mean.rich   n    sd.dev    std.err
## 1  2.304395 144 0.7024667 0.0585389
```

group_by() For obtaining the stat by group.

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>%
  group_by(Treatment, Fungicide) %>% # grouping by treatment and fungicide to later calculate summary s
  mutate(logRich = log(richness)) %>%
  summarise(Mean.rich = mean(logRich), # calculating the mean richness, stdeviation, and standard error
            n = n(),
            sd.dev = sd(logRich)) %>%
  mutate(std.err = sd.dev/sqrt(n))
```

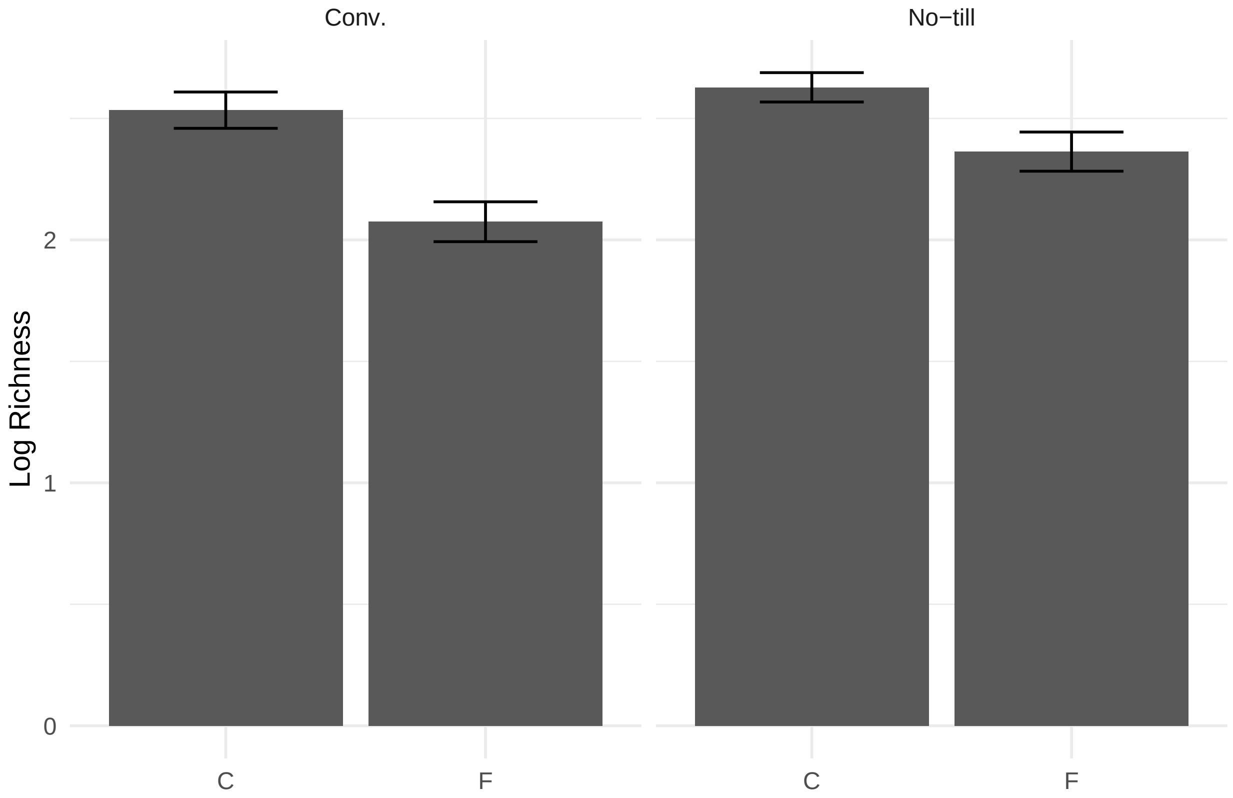
```
## `summarise()` has grouped output by 'Treatment'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 6
## # Groups:   Treatment [2]
##   Treatment Fungicide Mean.rich     n sd.dev std.err
##   <chr>     <chr>         <dbl> <int>  <dbl>   <dbl>
## 1 Conv.     C              2.53    72  0.635  0.0748
## 2 Conv.     F              2.07    72  0.696  0.0820
## 3 No-till   C              2.63    72  0.513  0.0604
## 4 No-till   F              2.36    71  0.680  0.0807
```

####adding ggplot to the previous using pipe

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>%
  group_by(Treatment, Fungicide) %>% # grouping by treatment and fungicide to later calculate summary s
  mutate(logRich = log(richness)) %>%
  summarise(Mean.rich = mean(logRich), # calculating the mean richness, stdeviation, and standard error
            n = n(),
            sd.dev = sd(logRich)) %>%
  mutate(std.err = sd.dev/sqrt(n)) %>%
  ggplot(aes(x = Fungicide, y = Mean.rich)) + # adding in a ggplot
  geom_bar(stat="identity") +
  geom_errorbar( aes(x=Fungicide, ymin=Mean.rich-std.err, ymax=Mean.rich+std.err), width=0.4) +
  theme_minimal() +
  xlab("") +
  ylab("Log Richness") +
  facet_wrap(~Treatment)
```

```
## `summarise()` has grouped output by 'Treatment'. You can override using the
## `.groups` argument.
```



#### Joining

Allow to combine multiple data set based on common set of variables.

They include: - left_join() - Keep all rows of X and add matching rows from Y. Any rows in Y that don't match X are excluded. - right_join() - reverse of left_join() - inner_join() - only keep rows that are common to both X AND Y, remove everything else. - full_join() - Keep any columns that are in either X or Y

```
# Examples
# selecting just the richness and sample ID
richness <- microbiome.fungi %>%
  select(SampleID, richness)

# selecting columns that don't include the richness
metadata <- microbiome.fungi %>%
  select(SampleID, Fungicide, Crop, Compartment, GrowthStage, Treatment, Rep, Sample)

head(richness)
```

```
##                                    SampleID richness
## 1 Corn2017LeafObjective2Collection1T1R1CAH2        9
## 2 Corn2017LeafObjective2Collection1T1R1CBA3        6
## 3 Corn2017LeafObjective2Collection1T1R1CCB3        5
## 4 Corn2017LeafObjective2Collection1T1R1FAC3        7
## 5 Corn2017LeafObjective2Collection1T1R1FBD3        4
## 6 Corn2017LeafObjective2Collection1T1R1FCE3        2
```

```
head(metadata)
```

```
##                                     SampleID Fungicide Crop Compartment
## 1 Corn2017LeafObjective2Collection1T1R1CAH2         C Corn        Leaf
## 2 Corn2017LeafObjective2Collection1T1R1CBA3         C Corn        Leaf
## 3 Corn2017LeafObjective2Collection1T1R1CCB3         C Corn        Leaf
## 4 Corn2017LeafObjective2Collection1T1R1FAC3         F Corn        Leaf
## 5 Corn2017LeafObjective2Collection1T1R1FBD3         F Corn        Leaf
## 6 Corn2017LeafObjective2Collection1T1R1FCE3         F Corn        Leaf
##   GrowthStage Treatment Rep Sample
## 1          V6     Conv.  R1      A
## 2          V6     Conv.  R1      B
## 3          V6     Conv.  R1      C
## 4          V6     Conv.  R1      A
## 5          V6     Conv.  R1      B
## 6          V6     Conv.  R1      C
```

```
# using leftjoin and adding the richness data to the metadata based on on the common column of sampleID
head(left_join(metadata, richness, by = "SampleID"))
```

```
##                                     SampleID Fungicide Crop Compartment
## 1 Corn2017LeafObjective2Collection1T1R1CAH2         C Corn        Leaf
## 2 Corn2017LeafObjective2Collection1T1R1CBA3         C Corn        Leaf
## 3 Corn2017LeafObjective2Collection1T1R1CCB3         C Corn        Leaf
## 4 Corn2017LeafObjective2Collection1T1R1FAC3         F Corn        Leaf
## 5 Corn2017LeafObjective2Collection1T1R1FBD3         F Corn        Leaf
## 6 Corn2017LeafObjective2Collection1T1R1FCE3         F Corn        Leaf
##   GrowthStage Treatment Rep Sample richness
## 1          V6     Conv.  R1      A        9
## 2          V6     Conv.  R1      B        6
## 3          V6     Conv.  R1      C        5
## 4          V6     Conv.  R1      A        7
## 5          V6     Conv.  R1      B        4
## 6          V6     Conv.  R1      C        2
```

Pivoting Used for converting from wide to long format and back again. we can do this using 'pivot_longer()' and 'pivot_wider()'.

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>% # selecting column
  group_by(Treatment, Fungicide) %>% # grouping by treatment and fungicide
  summarise(Mean = mean(richness)) %>% # calculate the mean
  pivot_wider(names_from = Fungicide, values_from = Mean) %>% # pivot to wide format
## can now easily calculate the difference between the mean between the fungicide and control groups.
  mutate(diff.fungicide = C - F)
```

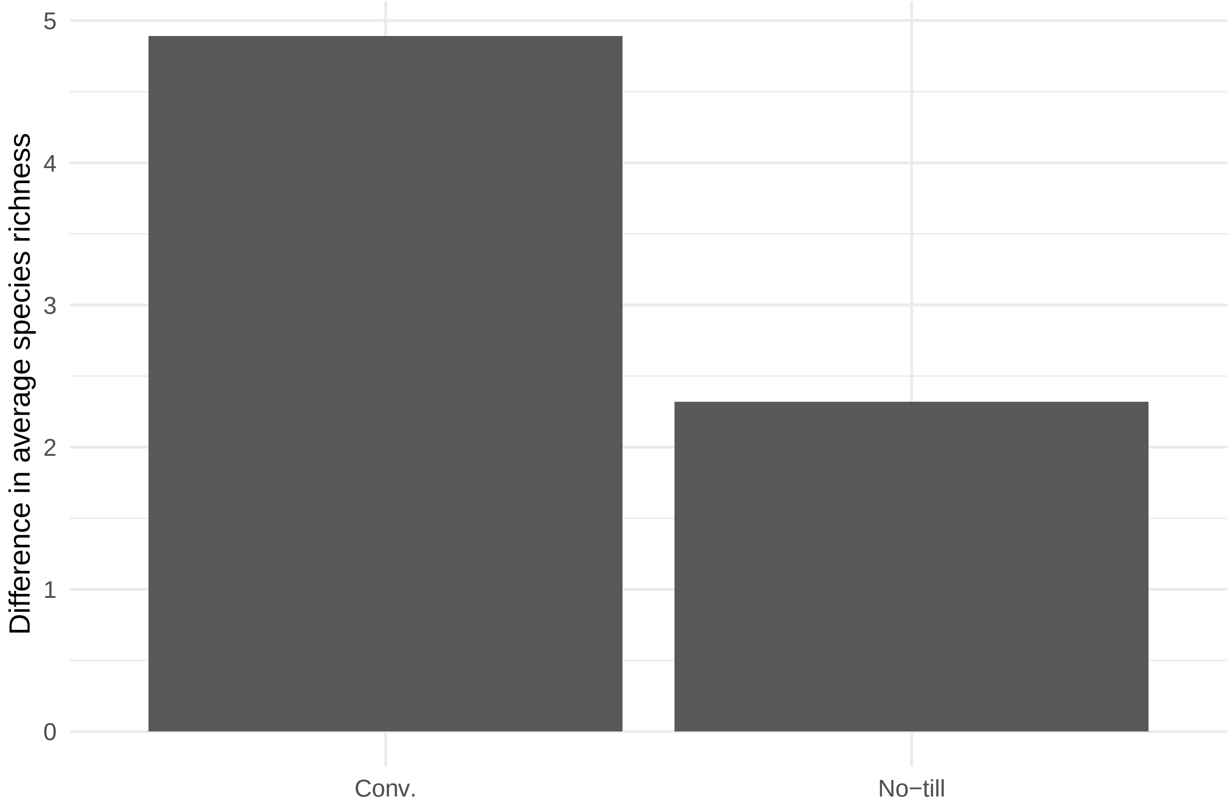
```
## 'summarise()' has grouped output by 'Treatment'. You can override using the
## '.groups' argument.
```

```
## # A tibble: 2 x 4
## # Groups:   Treatment [2]
##   Treatment     C     F diff.fungicide
##   <chr>     <dbl> <dbl>          <dbl>
## 1 Conv.      14.6  9.75           4.89
## 2 No-till    15.4 13.1            2.32
```

```
microbiome.fungi %>%
  select(SampleID, Crop, Compartment:Fungicide, richness) %>% # selecting column
  group_by(Treatment, Fungicide) %>% # grouping by treatment and fungicide
  summarise(Mean = mean(richness)) %>% # calculate the mean
  pivot_wider(names_from = Fungicide, values_from = Mean) %>% # pivot to wide format
## can now easily calculate the difference between the mean between the fungicide and control groups.
  mutate(diff.fungicide = C - F) %>%
  ggplot(aes(x= Treatment, y = diff.fungicide)) +
  geom_col() +
  theme_minimal() +
  xlab("") +
  ylab("Difference in average species richness")
```

adding a plot to above chunk

```
## ‘summarise()‘ has grouped output by ’Treatment’. You can override using the
## ‘.groups‘ argument.
```


Difference in average species richness
Conv.
No−till