Coding Practice 3
Advanced Data Visualization

PLPA-5820 (Spring 2025)

Gabriel A. A. Silva

2025-02-20

Instructions

Advanced Data Visualization– 25 pts

This assignment is designed to help you visualize data in R and practice your R coding skills. You may collaborate with a partner to enhance your learning experience. Please ensure the following:

- Collaboration: If you worked with a partner, include their name.
- Submission: Only one person should submit the assignment.
- Answers: Provide written answers to all questions in the current document or a rendered markdown file.
- Code Submission: Submit your R code in an .R or .Rmd file by sharing a link to your GitHub repository.

Data description

Here, we will analyze mycotoxin data collected from wheat heads grown in the greenhouse inoculated with Fusarium graminearum (Fg) with or without fungal endophytes to protect against Fg. The experiment was performed by inoculating the heads with Fg (positive control), Fg + 1 of 3 endophytes (Fg + 40, Fg + 70, or Fg + 37), and the non-treated control (NTC). Therefore, the experiment was five treatments with 6-10 technical replicates and three experimental replicates. The authors quantified the parts per million (ppm) of mycotoxins Deoxynivalenol (DON) and the variant 15ADON from the wheat grain, and this is your primary response variable. In other words, we are interested in knowing if the fungal endophytes could reduce the concentration of DON on wheat heads. This was done across two wheat cultivars, a winter wheat variety, Ambassador, and a spring wheat variety, Wheaton. These data are published here:

- Noel, Z.A., Roze, L.V., Breunig, M., Trail, F. 2022. Endophytic fungi as promising biocontrol agent to protect wheat from Fusarium graminearum head blight. Plant Disease. https://doi.org/10.1094/PDIS-06-21-1253-RE • The following github repository is associated with this paper: https://github.com/noelzach/EndophyteBiocontrol

Questions

```r
# Load libraries
library(tidyverse)
```

```
-- 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     1.0.4
-- 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 becom
```

```r
library(ggpubr)

# Read in the data
## Treatment, Cultivar, MassperSeed_mg, DON, 15ADON, BioRep
data <- read_csv("MycotoxinData.csv") %>%
            mutate(DON = as.numeric(DON),
                   MassperSeed_mg = as.numeric(MassperSeed_mg),
                   `15ADON` = as.numeric(`15ADON`),
                   BioRep = as.factor(BioRep)) %>%
            filter(!is.na(DON), !is.na(`15ADON`),
                   !is.na(MassperSeed_mg))
```

```
Rows: 375 Columns: 6
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (5): Treatment, Cultivar, MassperSeed_mg, DON, 15ADON
dbl (1): BioRep

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

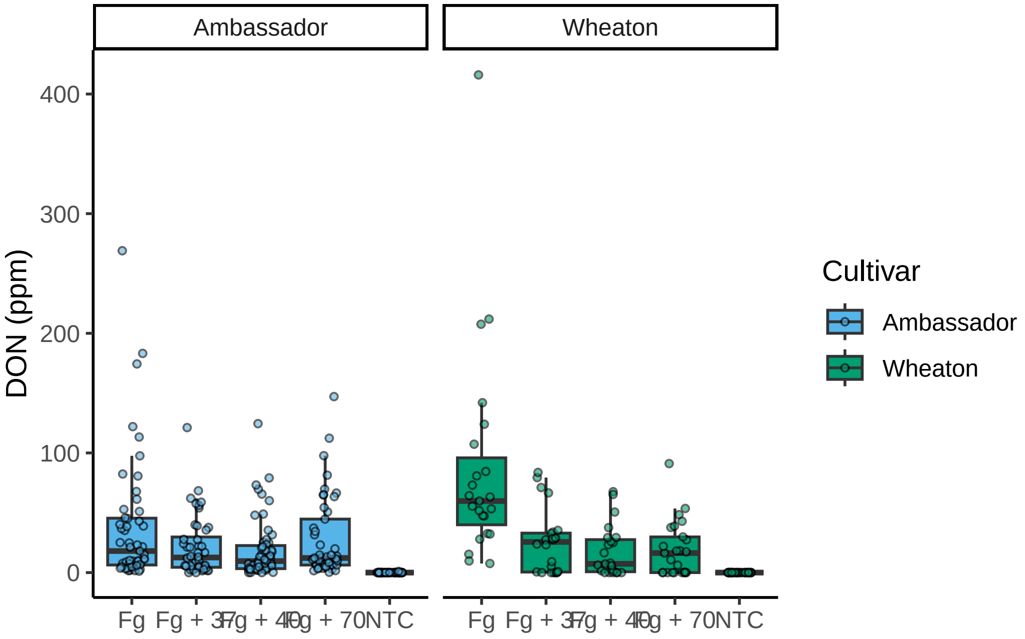
```
Warning: There were 3 warnings in `mutate()`.
The first warning was:
i In argument: `DON = as.numeric(DON)`.
Caused by warning:
! NAs introduced by coercion
i Run `dplyr::last_dplyr_warnings()` to see the 2 remaining warnings.
```

```
# Define color palette
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73",
                "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
```

1. 5pts. Using ggplot, create a boxplot of DON by Treatment so that the plot looks like the image below.

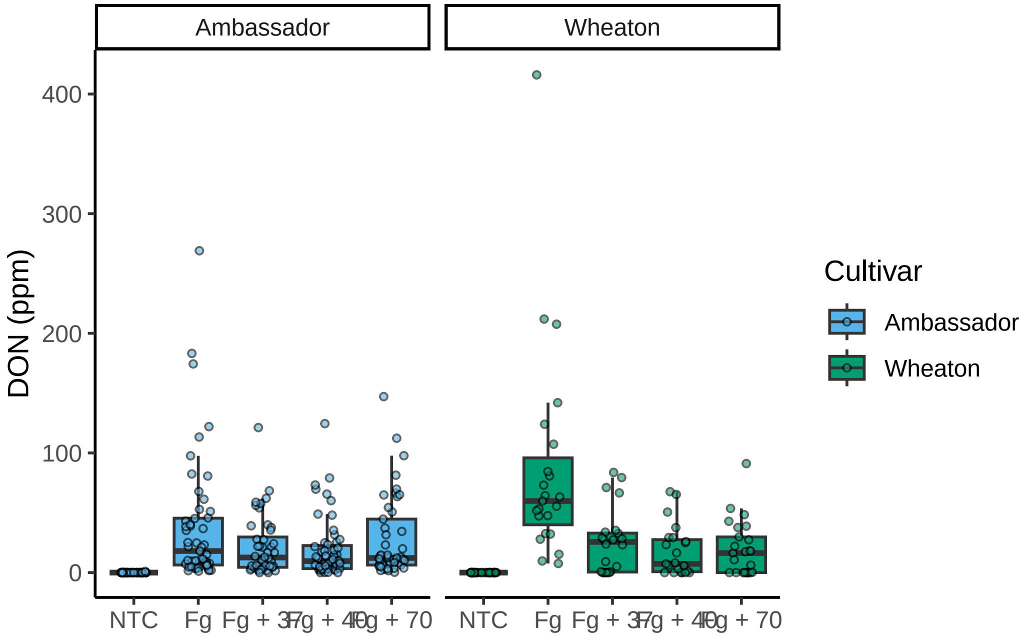
Jitter points over the boxplot and fill the points and boxplots Cultivar with two colors from the cbbPallete introduced last week. b. Change the transparency of the jittered points to 0.6. c. The y-axis should be labeled "DON (ppm)", and the x-axis should be left blank. d. The plot should use a classic theme e. The plot should also be faceted by Cultivar

```
# Boxplot
ggplot(data, aes(x = Treatment, y = DON, fill = Cultivar)) +
        geom_boxplot(outliers = FALSE) +
        geom_point(position = position_jitterdodge(), shape = 21,
                   color = "black", size = 1, alpha=0.6) +
        labs(y = "DON (ppm)", x = element_blank()) +
        scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                          labels = c("Ambassador", "Wheaton")) +
        theme_classic() +
        facet_wrap(~ Cultivar)
```



2. 4pts. Change the factor order level so that the treatment “NTC” is first, followed by “Fg”, “Fg + 37”, “Fg + 40”, and “Fg + 70.

```
# Change factor order
data$Treatment <- factor(data$Treatment, levels = c("NTC", "Fg", "Fg + 37",
                                                    "Fg + 40", "Fg + 70"))

# Boxplot
ggplot(data, aes(x = Treatment, y = DON, fill = Cultivar)) +
        geom_boxplot(outliers = FALSE) +
        geom_point(position = position_jitterdodge(), shape = 21,
                   color = "black", size = 1, alpha=0.6) +
        labs(y = "DON (ppm)", x = element_blank()) +
        scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                          labels = c("Ambassador", "Wheaton")) +
        theme_classic() +
        facet_wrap(~ Cultivar)
```



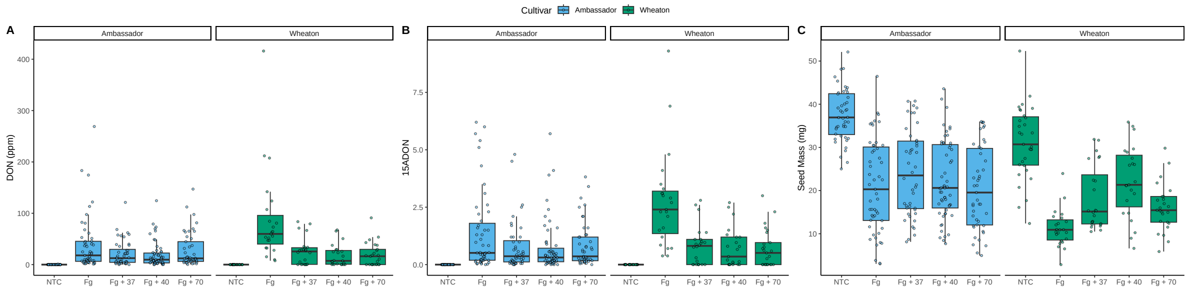
3. 5pts. Change the y-variable to plot X15ADON and MassperSeed_mg. The y-axis label should now be "15ADON" and "Seed Mass (mg)". Save plots made in questions 1 and 3 into three separate R objects.

```
# DON
plot1 <- ggplot(data, aes(x = Treatment, y = DON, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "DON (ppm)", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar)

# X15ADON
plot2 <- ggplot(data, aes(x = Treatment, y = `15ADON`, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "15ADON", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar)

# MassperSeed_mg
plot3 <- ggplot(data, aes(x = Treatment, y = MassperSeed_mg, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "Seed Mass (mg)", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar)
```

4. 5pts. Use ggarrange function to combine all three figures into one with three columns and one row. Set the labels for the subplots as A, B and C. Set the option common.legend = T within ggarage function. What did the common.legend option do?

a. HINT: I didn't specifically cover this in the tutorial, but you can go to the help page for the ggarange function to figure out what the common.legend option does and how to control it.

```
# Combine plots
ggarrange(plot1, plot2, plot3,
          labels = c("A", "B", "C"),ncol=3, common.legend = TRUE)
```

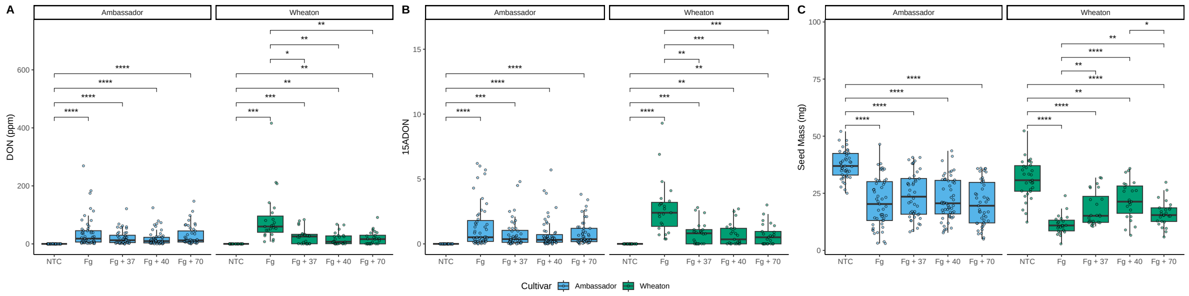


The common.legend option allows you to combine the legends of the three plots into one, saving space.

5. 5pts. Use geom_pwc() to add t.test pairwise comparisons to the three plots made above. Save each plot as a new R object, and combine them again with ggarange as you did in question 4.

```
# DON
stat1 <- ggplot(data, aes(x = Treatment, y = DON, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "DON (ppm)", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar) +
            geom_pwc(aes(group = Treatment),
                     method = "t_test", label = "p.adj.signif", hide.ns = TRUE)

# X15ADON
stat2 <- ggplot(data, aes(x = Treatment, y = `15ADON`, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "15ADON", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar) +
            geom_pwc(aes(group = Treatment),
                     method = "t_test", label = "p.adj.signif", hide.ns = TRUE)

# MassperSeed_mg
stat3 <- ggplot(data, aes(x = Treatment, y = MassperSeed_mg, fill = Cultivar)) +
            geom_boxplot(outliers = FALSE) +
            geom_point(position = position_jitterdodge(), shape = 21,
                       color = "black", size = 1, alpha=0.6) +
            labs(y = "Seed Mass (mg)", x = element_blank()) +
            scale_fill_manual(values = cbbPalette[3:4], name = "Cultivar",
                              labels = c("Ambassador", "Wheaton")) +
            theme_classic() +
            facet_wrap(~ Cultivar) +
            geom_pwc(aes(group = Treatment),
                     method = "t_test", label = "p.adj.signif", hide.ns = TRUE)
```

```
# Combine plots
ggarrange(stat1, stat2, stat3,
          labels = c("A", "B", "C"), ncol=3, common.legend = TRUE, legend = "bottom")
```



6. 1 pt. Annotate your code, push it to GitHub. Provide me a link to your GitHub

https://github.com/bielasilva/PLPA-5820-SP2025/tree/main/Coding%20Challenge%203