

# Data Visualization 1

Mamata K C

2025-02-15

Introduction to visualization in R – 25 pts

PLEASE READ THIS BEFORE CONTINUING

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.

Partner 1: Mamata K C

Partner 2 to fork the repository: Sachida Pokhrel

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>

1. 5 pts. Explain the following concepts about ggplot

a. What three elements do you need to produce a ggplot?

The three elements we need to produce a ggplot are 1. data 2. ggplot2 package with ggplot function `ggplot()`, aesthetic mapping functions `aes()` 3. layer functions (geom) such as `geom_point()`, `geom_boxplot`, `geom_smooth` etc.

b. What is a geom?

Geom is a layer in ggplot used for visual representation of data.

c. What is a facet?

A facet is used for making multiple plots for same variables splitting by categorical variable.

d. Explain the concept of layering.

ggplot works with the concept of layering so to make a plot using ggplot function, we need to add layers by using function called geom such as `geom_point()`, `geom_smooth()`, `geom_boxplot` etc. Within layers, it has many options and also the order of layers can be changed according to which function is written first in the code.

e. Where do you add x and y variables and map different shapes, colors, and other attributes to the data?

For adding x and y variables, we use main aes function. If we add different shapes, colors and other attributes to the data within the main aes function then it applies to all the layers of the plot however, we can also specify aes separately for different layers.

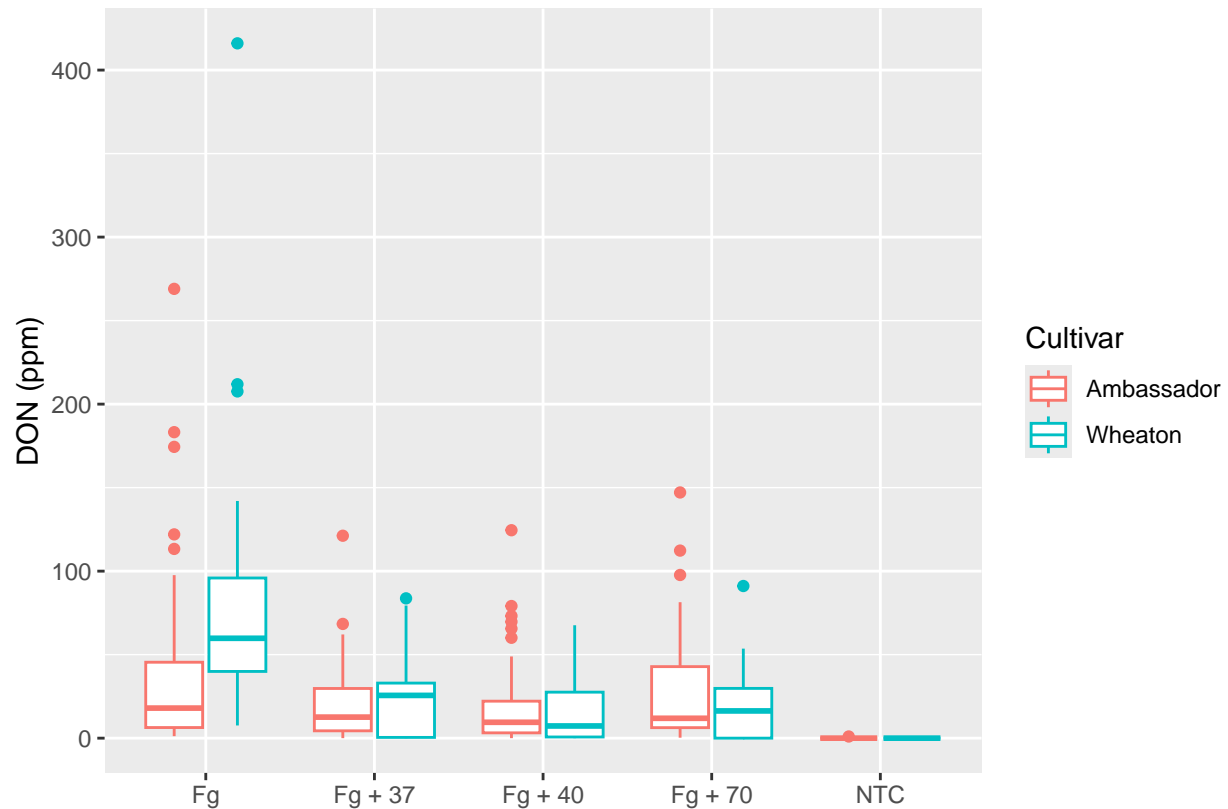
2. 4 pts. Make a boxplot using ggplot with DON as the y variable, treatment as the x variable, and color mapped to the wheat cultivar. Show the code you use to load the libraries you need to read in the data and make the plot. Change the y label to “DON (ppm)” and make the x label blank.

```
# loading data in R
datum <- read.csv("MycotoxinData.csv", na.strings = "na")

#loading library for ggplot
library(ggplot2)

#Creating boxplot
ggplot(datum, aes(x=Treatment, y=DON, color=Cultivar)) +
  geom_boxplot() +
  xlab("") +
  ylab("DON (ppm)")
```

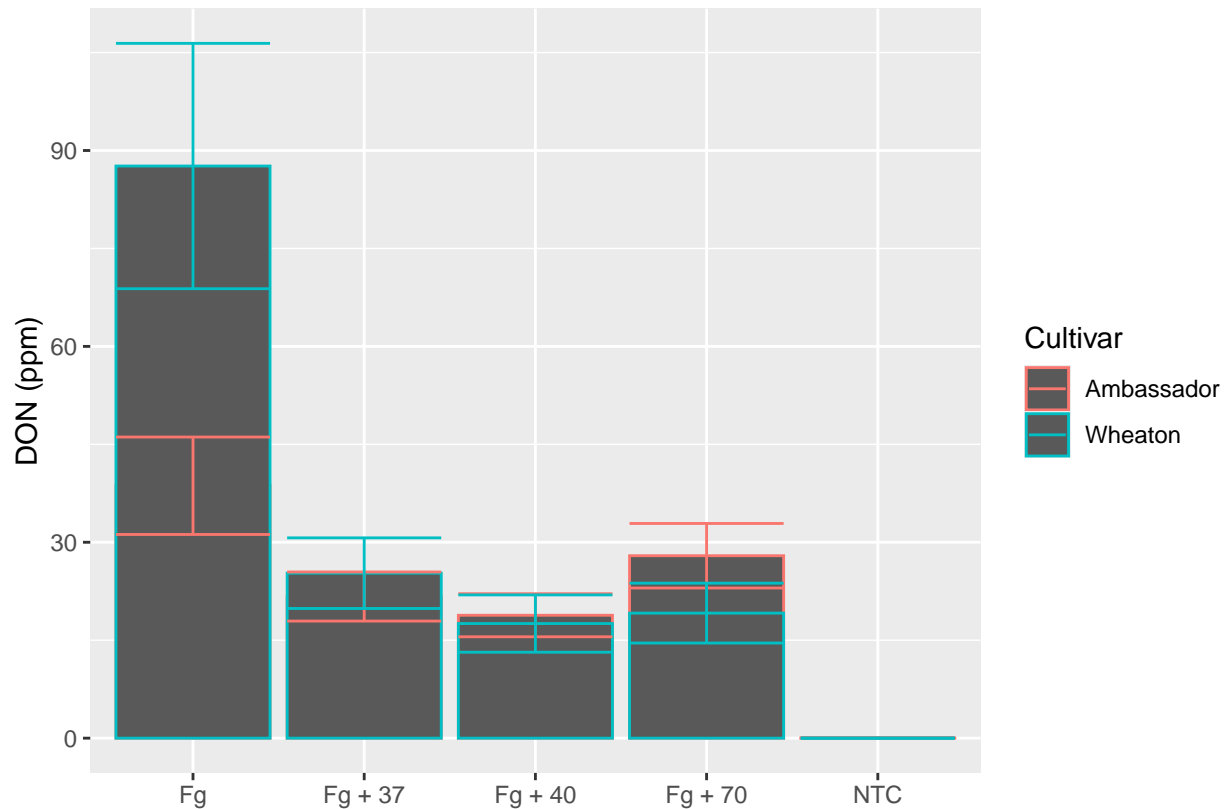
```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```



3.2 pts. Now convert this data into a bar chart with standard-error error bars using the `stat_summary()` command.

```
#creating bar chart with standard-error bars
ggplot(datum, aes(x=Treatment, y= DON, color= Cultivar)) +
  stat_summary(fun=mean, geom="bar") +
  stat_summary(fun.data = mean_se, geom = "errorbar") +
  xlab("") +
  ylab("DON (ppm)")
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
## Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

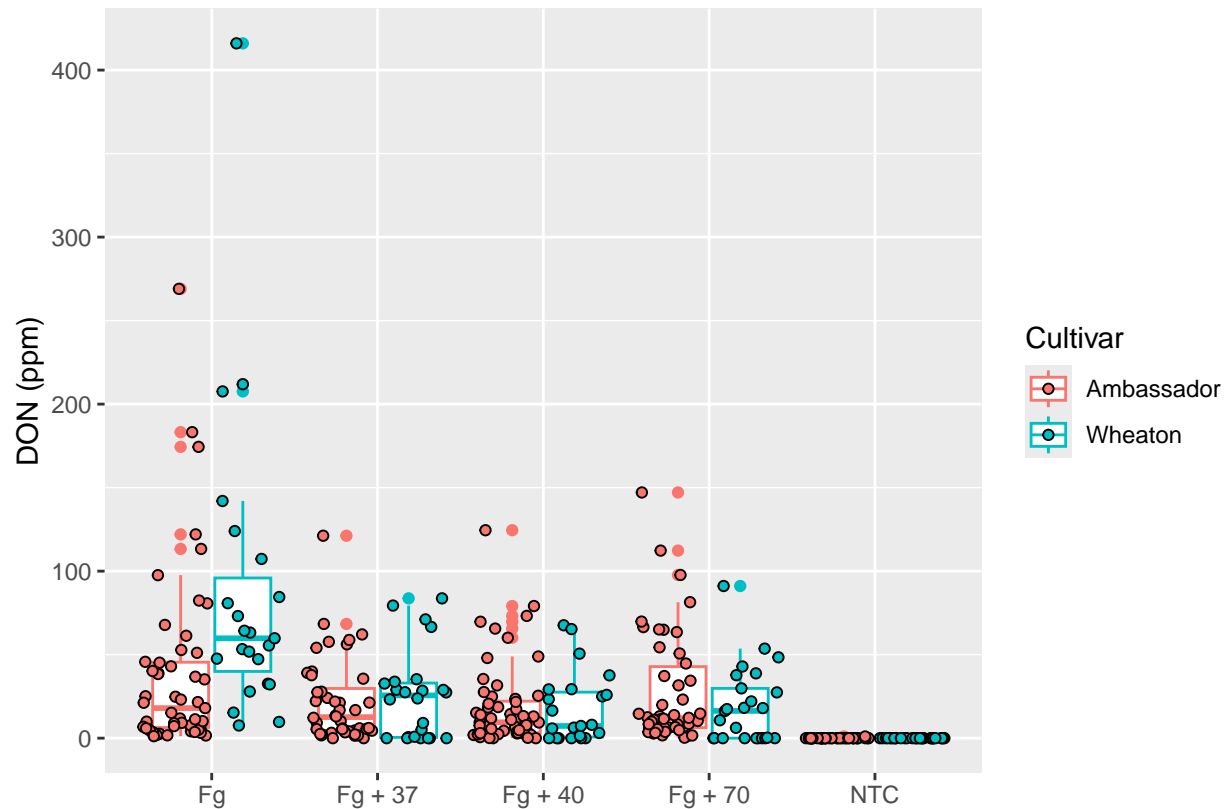


4. 2 pts. Add points to the foreground of the boxplot and bar chart you made in question 3 that show the distribution of points over the boxplots. Set the shape = 21 and the outline color black (hint: use jitter\_dodge).

```
#Adding points to the foreground of boxplot with distribution of points making shape 21 and outline color black
ggplot(datum, aes(x=Treatment, y=DON, color=Cultivar)) +
  geom_boxplot() +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black")
xlab("")+
ylab("DON (ppm)")
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

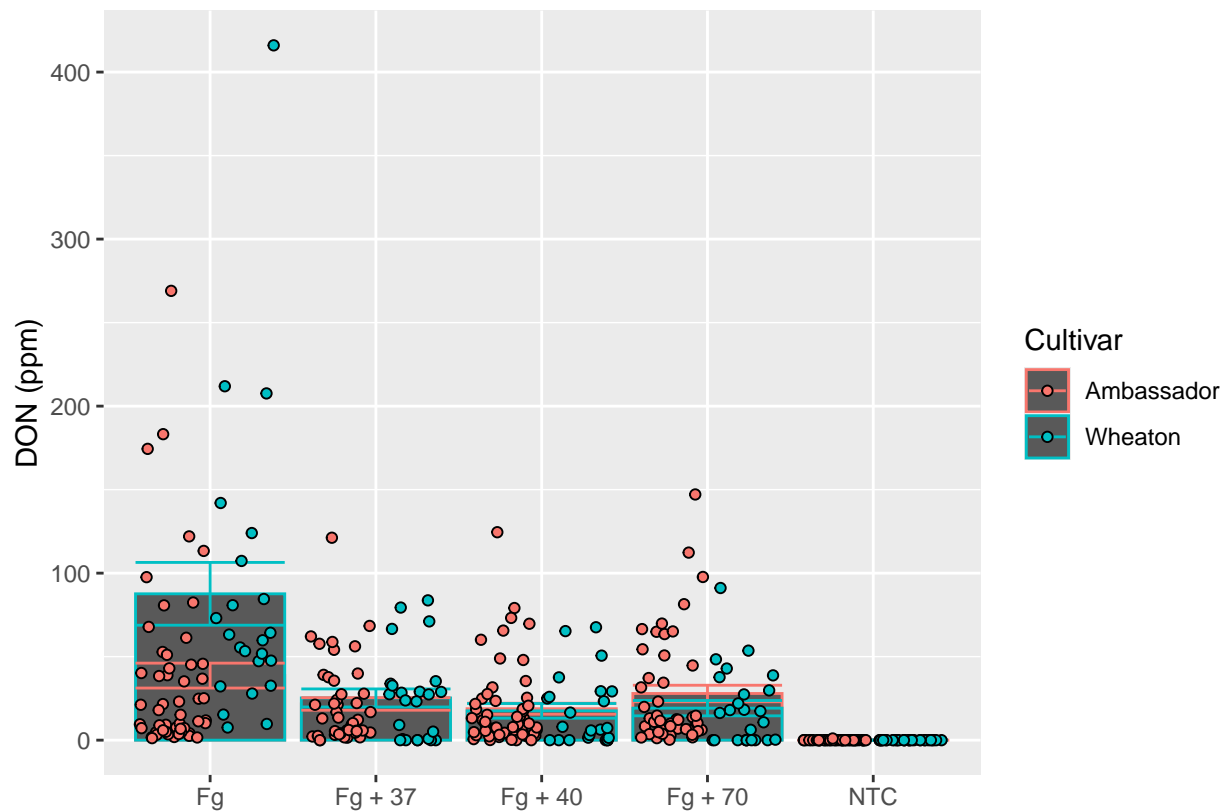


```
#Adding points to the foreground of barchart with distribution of points making shape 21 and outline color
ggplot(datum, aes(x=Treatment, y= DON, color= Cultivar)) +
  stat_summary(fun=mean, geom="bar") +
  stat_summary(fun.data = mean_se, geom = "errorbar") +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  xlab("") +
  ylab("DON (ppm)")
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



5. 2 pts. Change the fill color of the points and boxplots to match some colors in the following colorblind pallet.

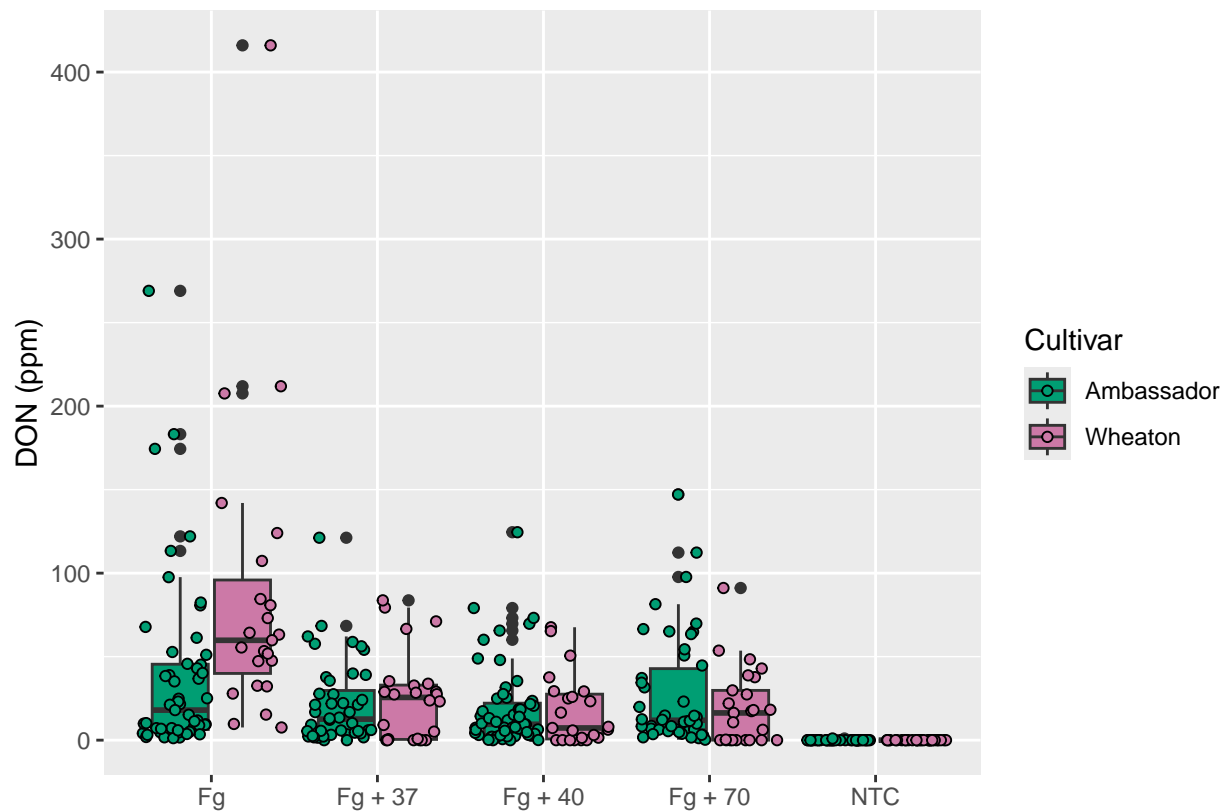
```
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00",
"#CC79A7")
```

```
#Loading colorblind palette
cbbPalette <- c( "#009E73", "#CC79A7")

#Changing fill color of points and boxplots to match with colorblind pallet
ggplot(datum, aes(x=Treatment, y=DON, fill=Cultivar)) +
  geom_boxplot() +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)")
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

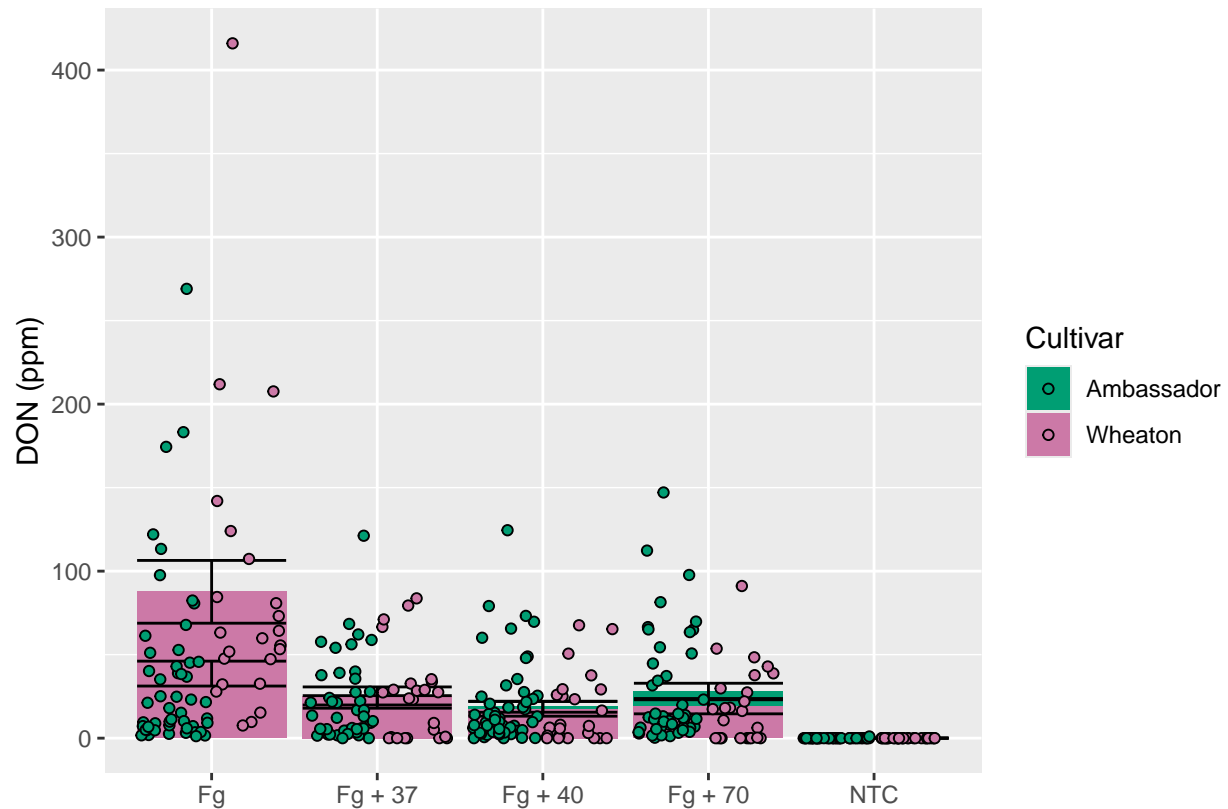


```
#Changing fill color of points and barchart to match with colorblind pallet
ggplot(datum, aes(x=Treatment, y= DON, fill=Cultivar)) +
  stat_summary(fun=mean, geom="bar") +
  stat_summary(fun.data = mean_se, geom = "errorbar") +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)")
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



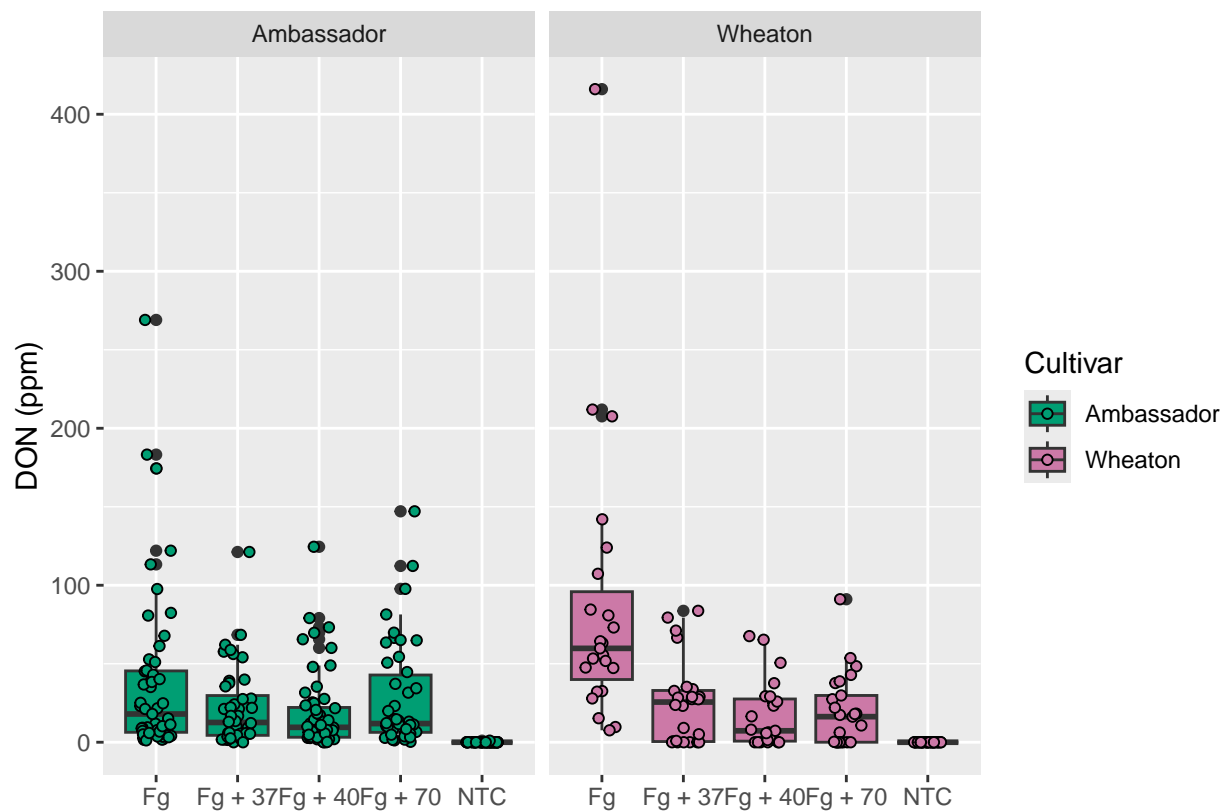
6. 2 pts. Add a facet to the plots based on cultivar.

```
#Adding a facet to the boxplot based on cultivar
cbbPalette <- c( "#009E73", "#CC79A7")
ggplot(datum, aes(x=Treatment, y=DON, fill=Cultivar)) +
  geom_boxplot() +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)") +
  facet_wrap(~Cultivar)
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



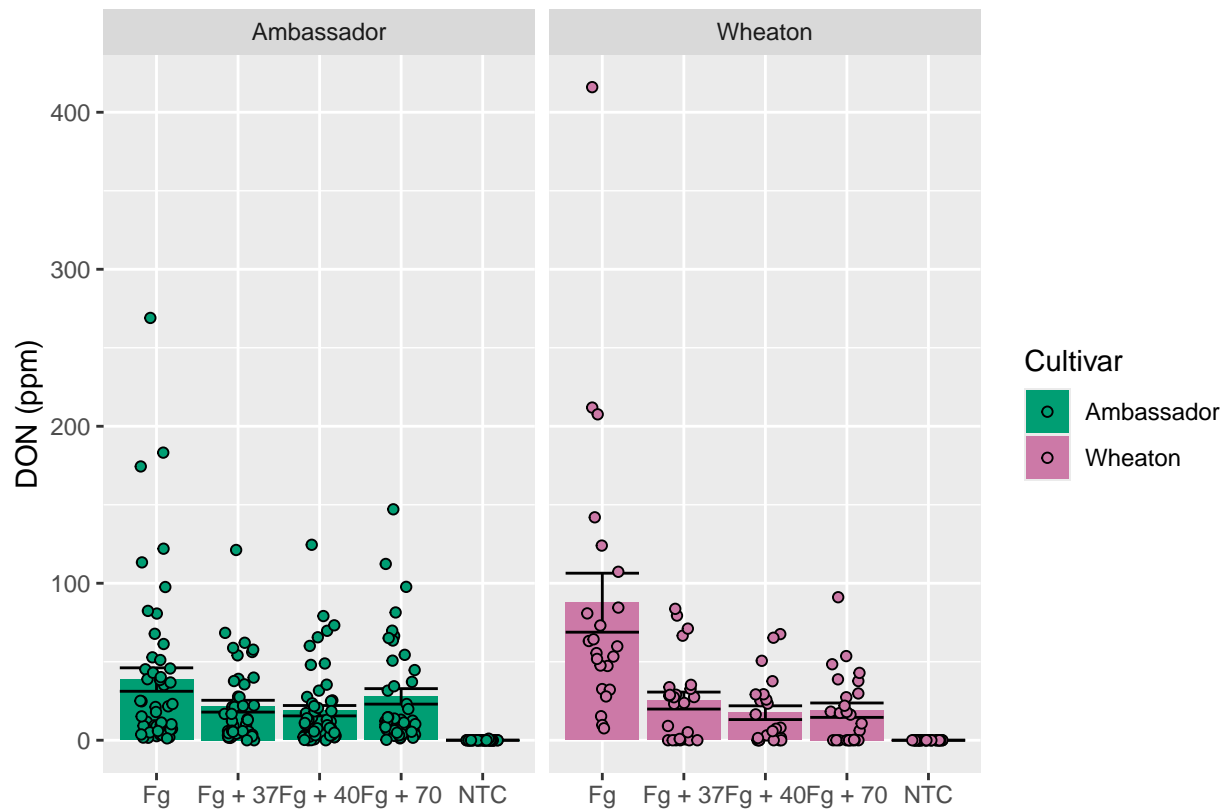


```
#Adding a facet to the barchart based on cultivar
ggplot(datum, aes(x=Treatment, y= DON, fill=Cultivar)) +
  stat_summary(fun=mean, geom="bar") +
  stat_summary(fun.data = mean_se, geom = "errorbar") +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)") +
  facet_wrap(~Cultivar)
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

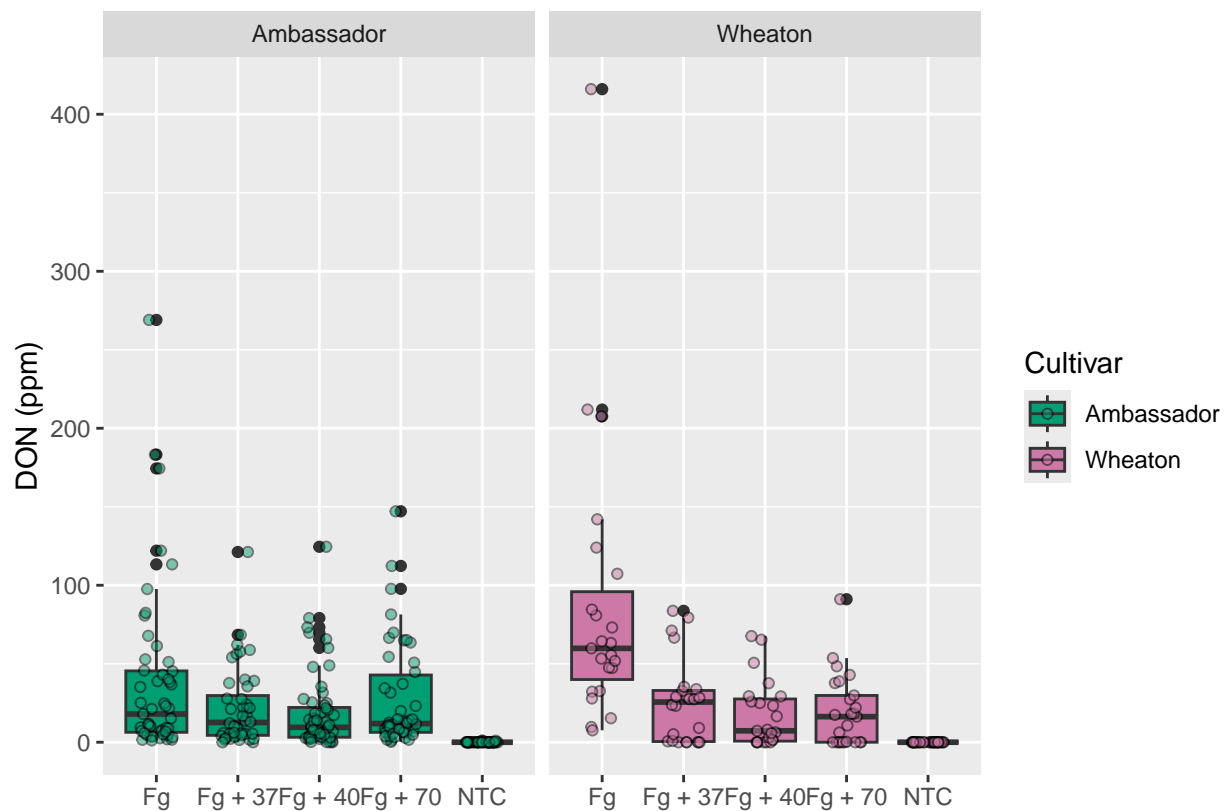


7. 2 pts. Add transparency to the points so you can still see the boxplot or bar in the background.

```
#Adding transparency to the point to make boxplot in the background visible
cbbPalette <- c( "#009E73", "#CC79A7")
ggplot(datum, aes(x=Treatment, y=DON, fill=Cultivar)) +
  geom_boxplot() +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black", alpha=0.5) +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)") +
  facet_wrap(~Cultivar)
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

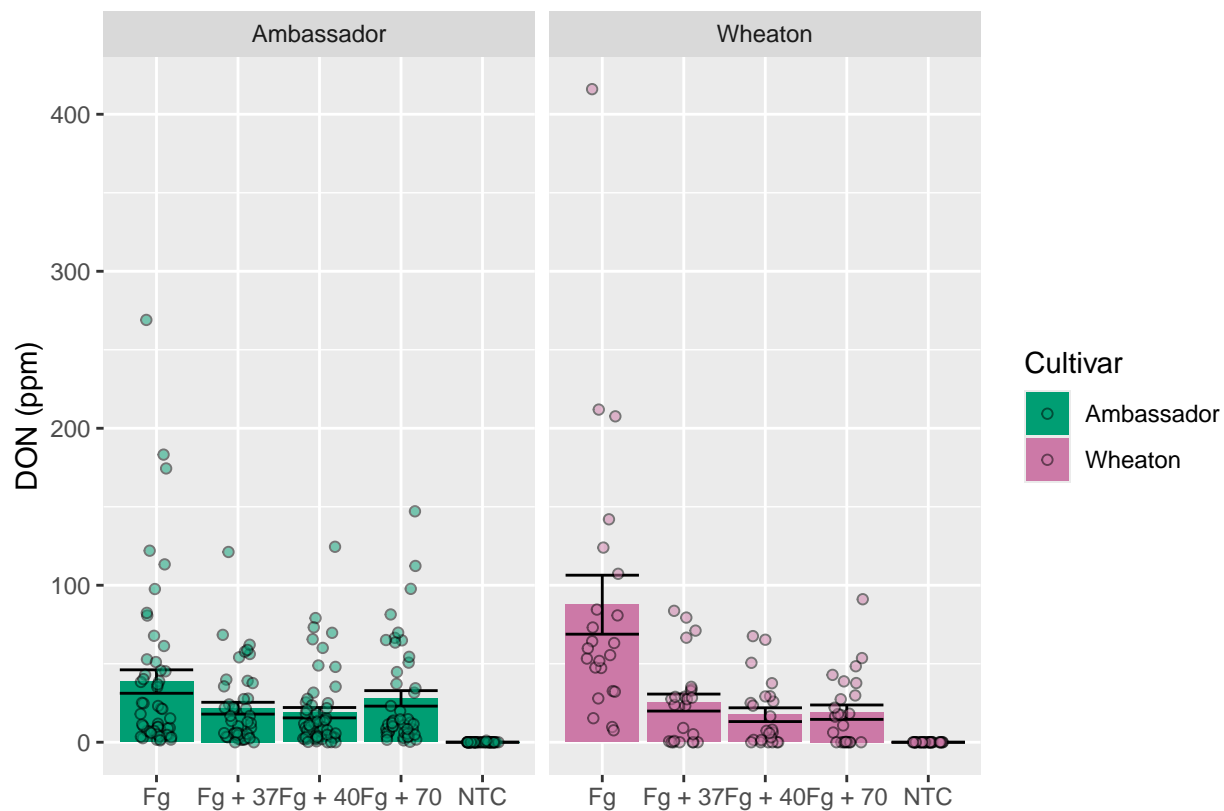


```
#Adding transparency to the point to make barchart in the background visible
ggplot(datum, aes(x=Treatment, y= DON,fill=Cultivar)) +
  stat_summary(fun=mean, geom="bar") +
  stat_summary(fun.data = mean_se, geom = "errorbar") +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)") +
  facet_wrap(~Cultivar)
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

```
## Warning: Removed 8 rows containing non-finite outside the scale range
## ('stat_summary()').
```

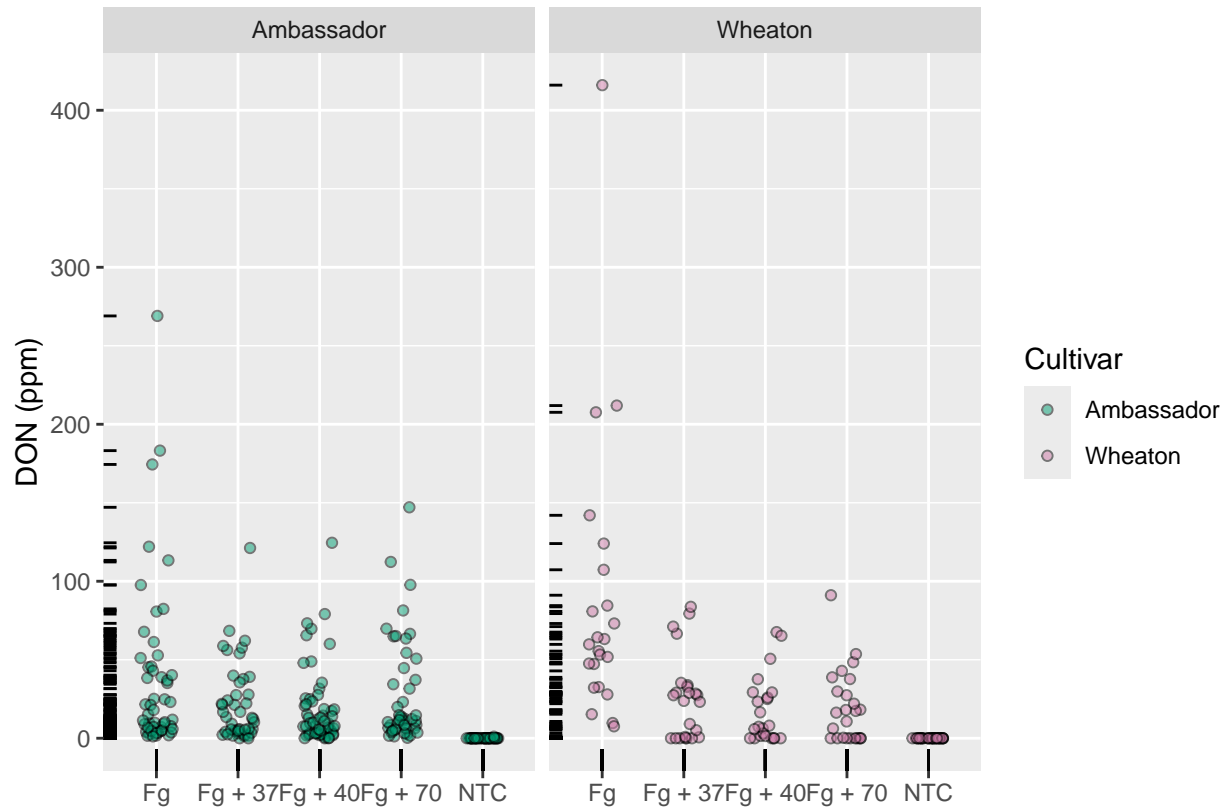
```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



8. 2 pts. Explore one other way to represent the same data <https://ggplot2.tidyverse.org/reference/> . Plot them and show the code here. Which one would you choose to represent your data and why?

```
#plotting rug plot
cbbPalette <- c( "#009E73", "#CC79A7")
ggplot(datum, aes(x=Treatment, y=DON, fill=Cultivar)) +
  geom_rug() +
  geom_point(position=position_jitterdodge(dodge.width = 0.9), aes(fill=Cultivar), shape=21, colour="black") +
  scale_fill_manual(values = cbbPalette) +
  scale_color_manual(values = cbbPalette) +
  xlab("") +
  ylab("DON (ppm)") +
  facet_wrap(~Cultivar)
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



I generated rug plot using `geom_rug()` function. It shows where the data are densely distributed and where they are sparsely distributed. However, I would choose boxplot for this data because with box plot, outliers are easily visible and box plot also gives better idea about the distribution of the data and helps us understand the central tendency of our data.

9. 4 pts. Annotate your code and push it to github. Now, find a partner if you don't already have one and have that person "fork" the repository containing the code you just pushed to github. Describe what just happened when you "forked" the repository. Please include the links (URLs) below to answer this question. Link to my github repository:

[https://github.com/mamata2423/ReproducibilityClass/blob/main/Data\\_visualization.R](https://github.com/mamata2423/ReproducibilityClass/blob/main/Data_visualization.R)

I forked the repository of one of my friend, and I observed that my github account has a copy of my friend's repository including all the files.

Link to my partner's repository:

[https://github.com/mamata2423/Class\\_Reproducibility.git](https://github.com/mamata2423/Class_Reproducibility.git)