

DataVisualization_Challenge1

Somraj Shrestha

2026-02-12

1. 5 pts. Explain the following concepts about ggplot.
 - a. What three elements do you need to produce a ggplot?
 - dataframe (data), aesthetic mapping (x and y variables, shape, color etc.), and a geom (the type of plot).
 - b. What is a geom?
 - It is a geometric object which determines how the data are drawn on the plot.
 - c. What is a facet?
 - It helps splitting the plots into small panels based on categorical variable thus, allowing us to compare between the groups within a dataframe.
 - d. Explain the concept of layering.
 - Layering means building a plot layer by layer by adding components that we need for the plot using `+`. Each layer adds information to the plot.
 - e. Where do you add x and y variables and map different shapes, colors, and other attributes to the data?
 - We add x and y variables and different shapes, colors, and attributes inside `aes()` in the `ggplot()`. Or we can add them inside a specific geom too.
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. Change the y label to “DON (ppm)” and make the x label blank.

```
myco_data <- read.csv("Data/MycotoxinData.csv", na.strings = "na") #load the data
str(myco_data)
```

```
## 'data.frame':    375 obs. of  6 variables:
## $ Treatment     : chr  "Fg" "Fg" "Fg" "Fg" ...
## $ Cultivar      : chr  "Wheaton" "Wheaton" "Wheaton" "Wheaton" ...
## $ BioRep        : int  2 2 2 2 2 2 2 2 2 3 ...
## $ MassperSeed_mg: num  10.29 12.8 2.85 6.5 10.18 ...
## $ DON           : num  107.3 32.6 416 211.9 124 ...
## $ X15ADON       : num  3 0.85 3.5 3.1 4.8 3.3 6.9 2.9 2.1 0.71 ...
```

```

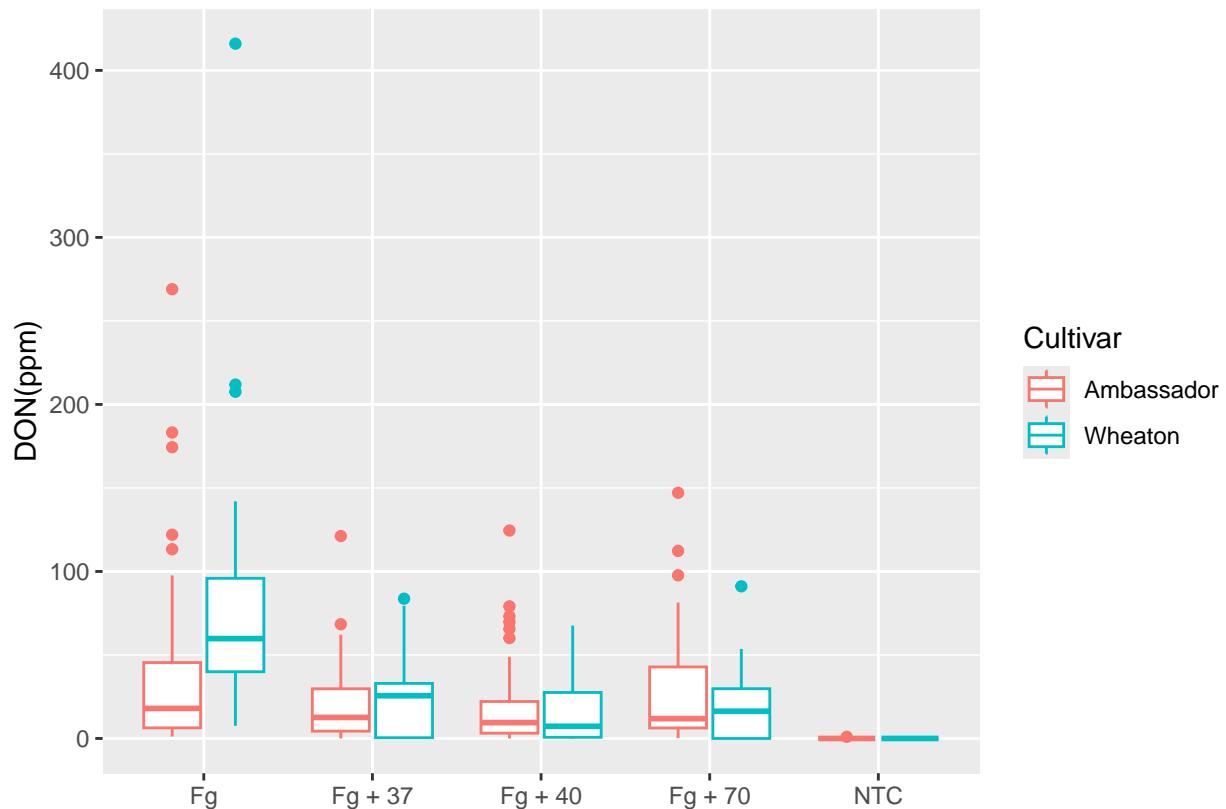
myco_data$Treatment <- as.factor(myco_data$Treatment)
myco_data$Cultivar <- as.factor(myco_data$Cultivar)
str(myco_data)

## 'data.frame': 375 obs. of 6 variables:
## $ Treatment : Factor w/ 5 levels "Fg","Fg + 37",...: 1 1 1 1 1 1 1 1 1 ...
## $ Cultivar   : Factor w/ 2 levels "Ambassador","Wheaton": 2 2 2 2 2 2 2 2 2 ...
## $ BioRep     : int 2 2 2 2 2 2 2 2 3 ...
## $ MassperSeed_mg: num 10.29 12.8 2.85 6.5 10.18 ...
## $ DON        : num 107.3 32.6 416 211.9 124 ...
## $ X15ADON    : num 3 0.85 3.5 3.1 4.8 3.3 6.9 2.9 2.1 0.71 ...

library(ggplot2)
plot1 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  geom_boxplot() +           #making a boxplot
  xlab("") +                 #labelling X-axis
  ylab("DON(ppm)")          #labelling Y-axis
plot1

```

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



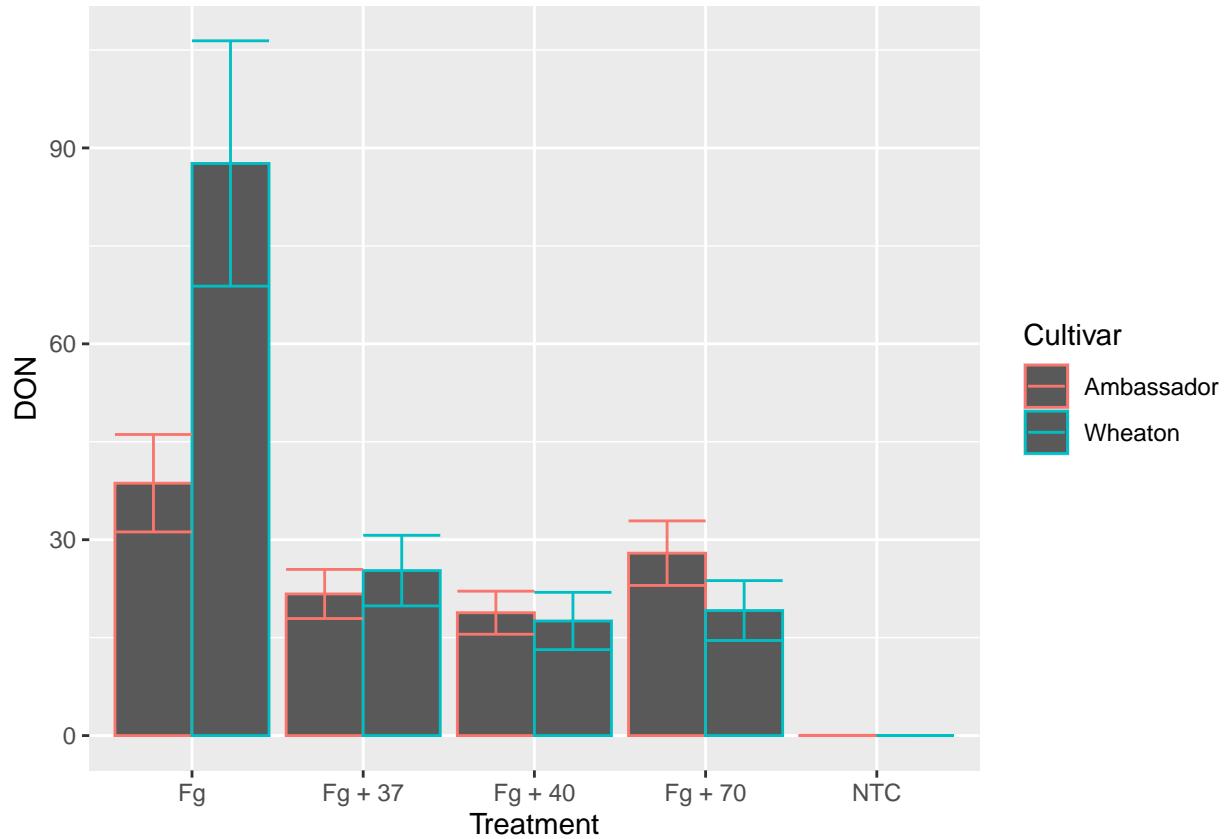
- 4 pts. Now convert this data into a bar chart with standard-error error bars using the stat_summary() command. (hint: use position = dodge)

```

plot2 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  stat_summary(fun = mean, geom = "bar", position = "dodge") + #generating barchart with mean
  stat_summary(fun.data = mean_se, geom = "errorbar", position = "dodge") #adding standard error bars
plot2

## 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. 3 pts. Add points to the foreground of the boxplot and bar chart you made in questions 2 & 3 that show the distribution of points. Set the shape = 21 and the outline color black (hint: use jitter_dodge).

```

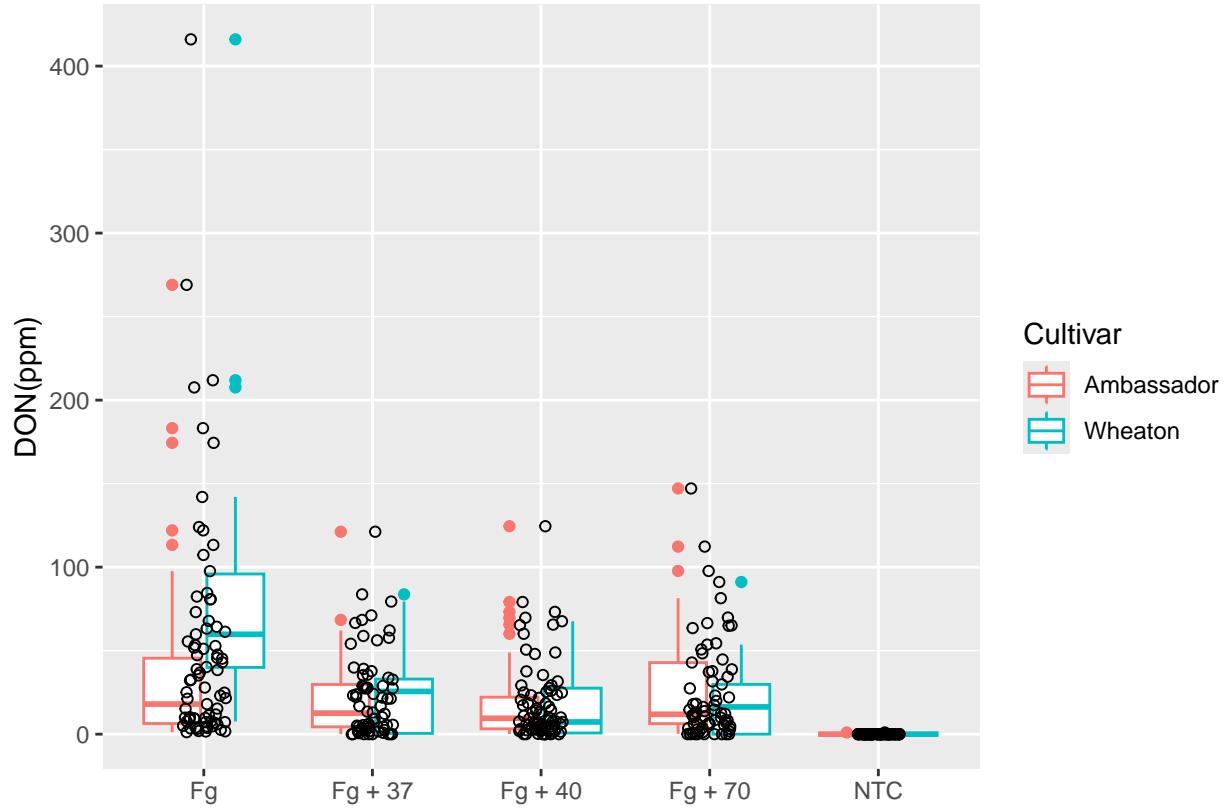
plot3 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  geom_boxplot() +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black") + #adding points on the boxplots
  xlab("") +
  ylab("DON(ppm)")

plot3

## 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()').

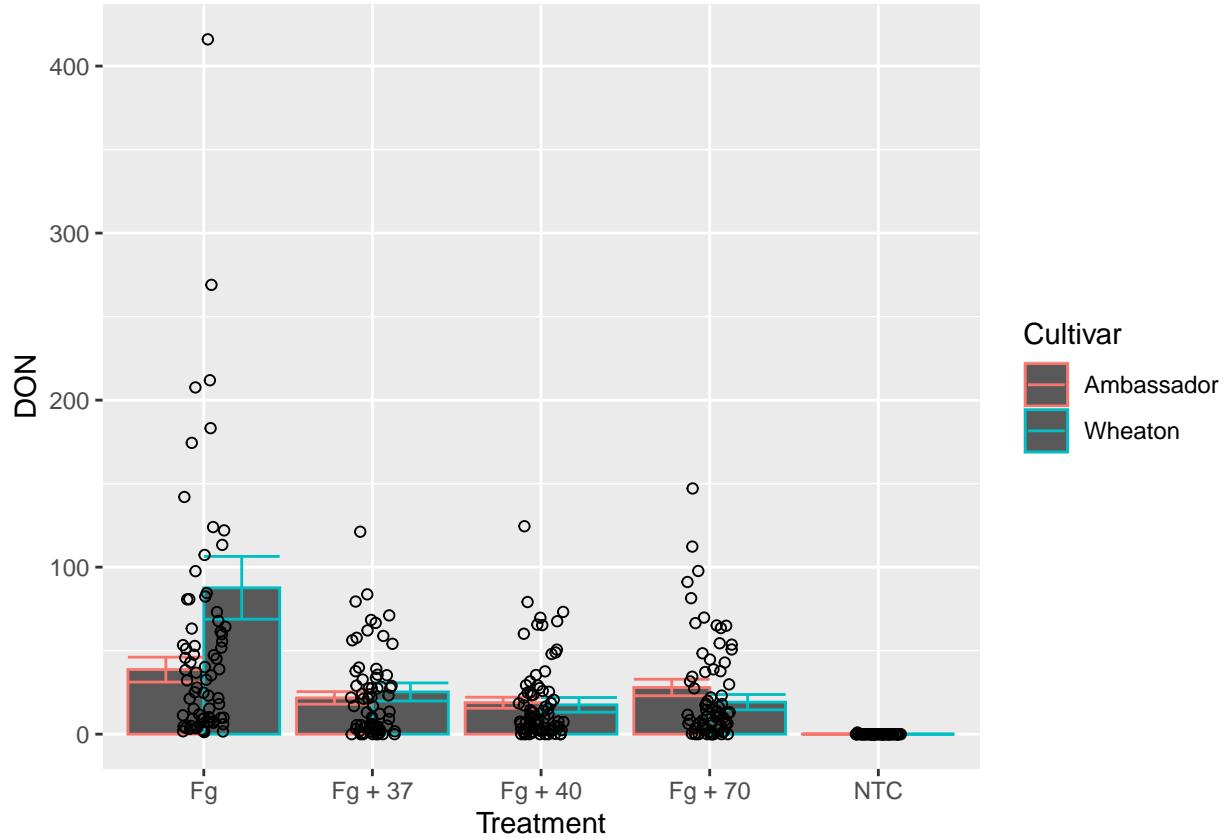
```



```
plot4 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  stat_summary(fun = mean, geom = "bar", position = "dodge") +
  stat_summary(fun.data = mean_se, geom = "errorbar", position = "dodge") +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black")
plot4
```

```
## 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()').

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

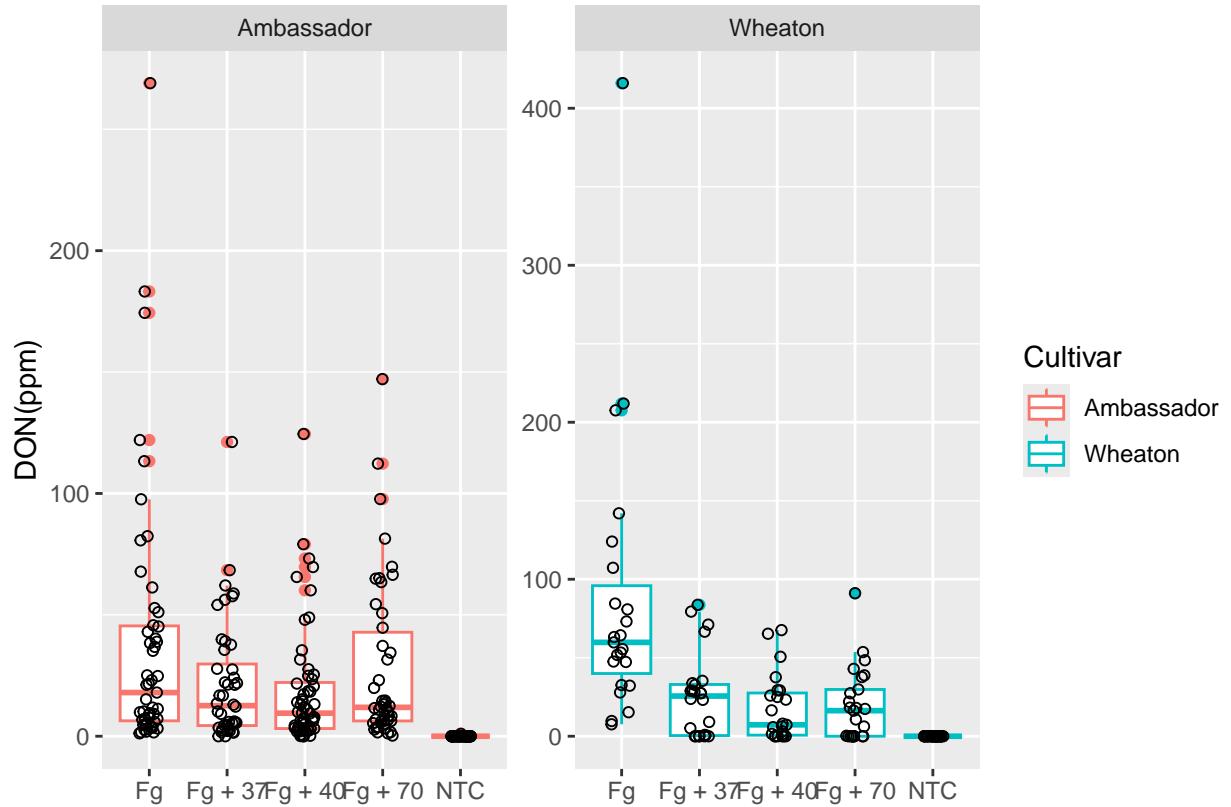


5. 2 pts. Add a facet to the plots based on cultivar.

```
plot5 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  geom_boxplot() +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black") +
  xlab("") +
  ylab("DON(ppm)") +
  facet_wrap(~Cultivar, scales = "free") #adding facet to divide the dataframe into sub-groups
plot5

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

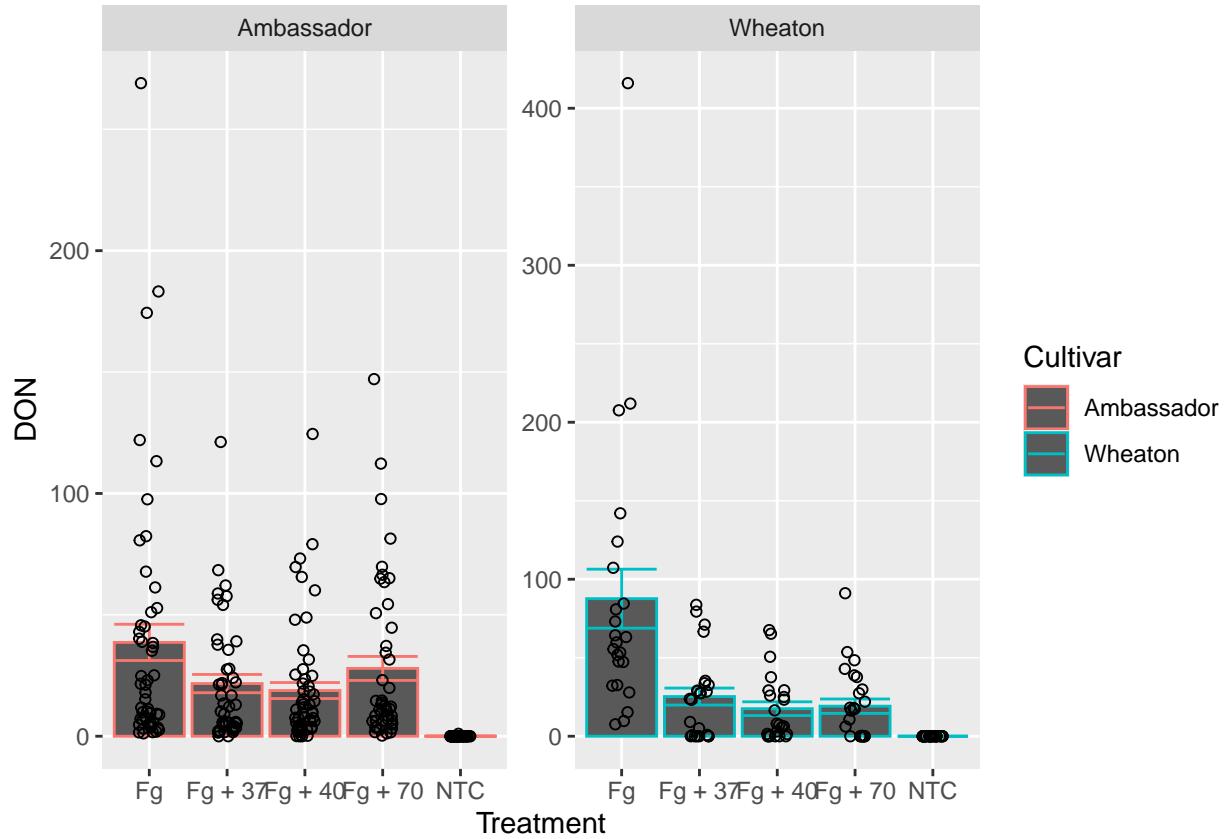
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



```
plot6 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  stat_summary(fun = mean, geom = "bar", position = "dodge") +
  stat_summary(fun.data = mean_se, geom = "errorbar", position = "dodge") +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black") +
  facet_wrap(~Cultivar, scales = "free")
plot6
```

```
## 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()').

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

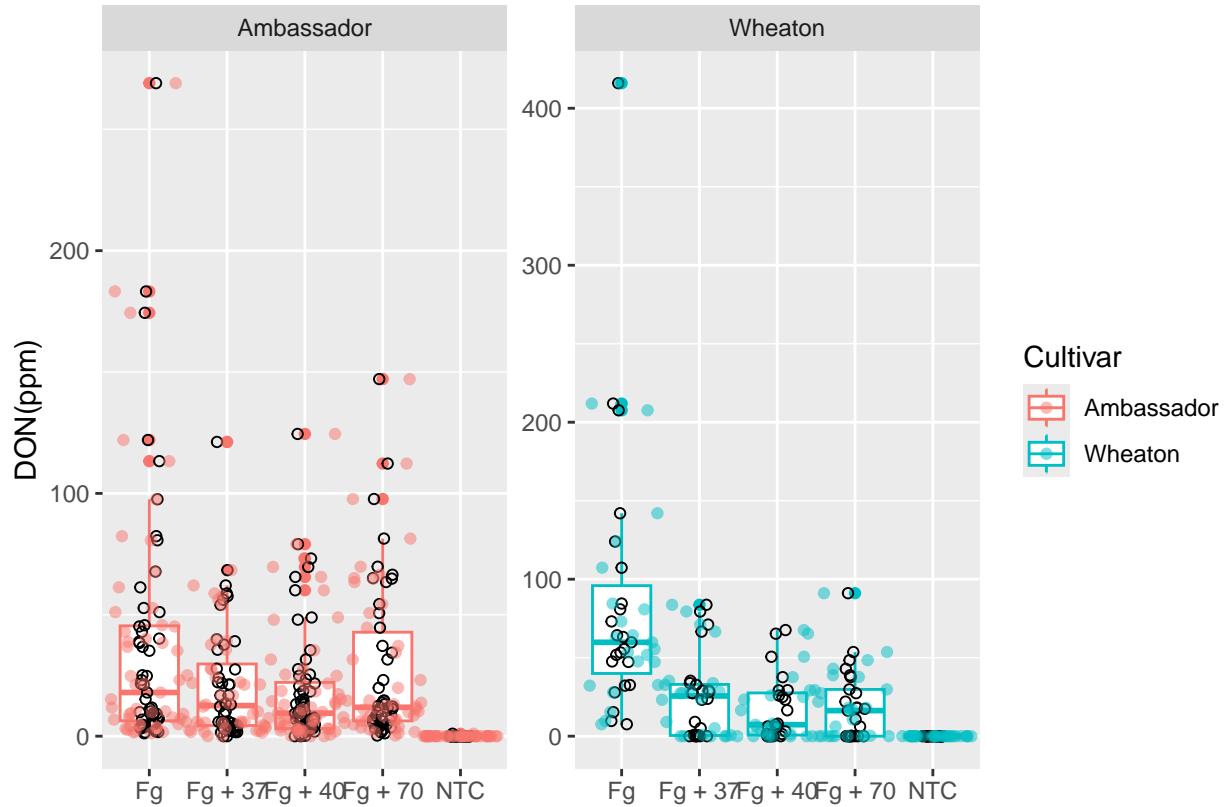


6. 3 pts. Add transparency to the points so you can still see the boxplot or bar in the background.

```
plot7 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  geom_boxplot() +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black") +
  geom_jitter(width = 0.5, alpha = 0.5) #adding transparency to the points
  xlab("") +
  ylab("DON(ppm)") +
  facet_wrap(~Cultivar, scales = "free")
plot7

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

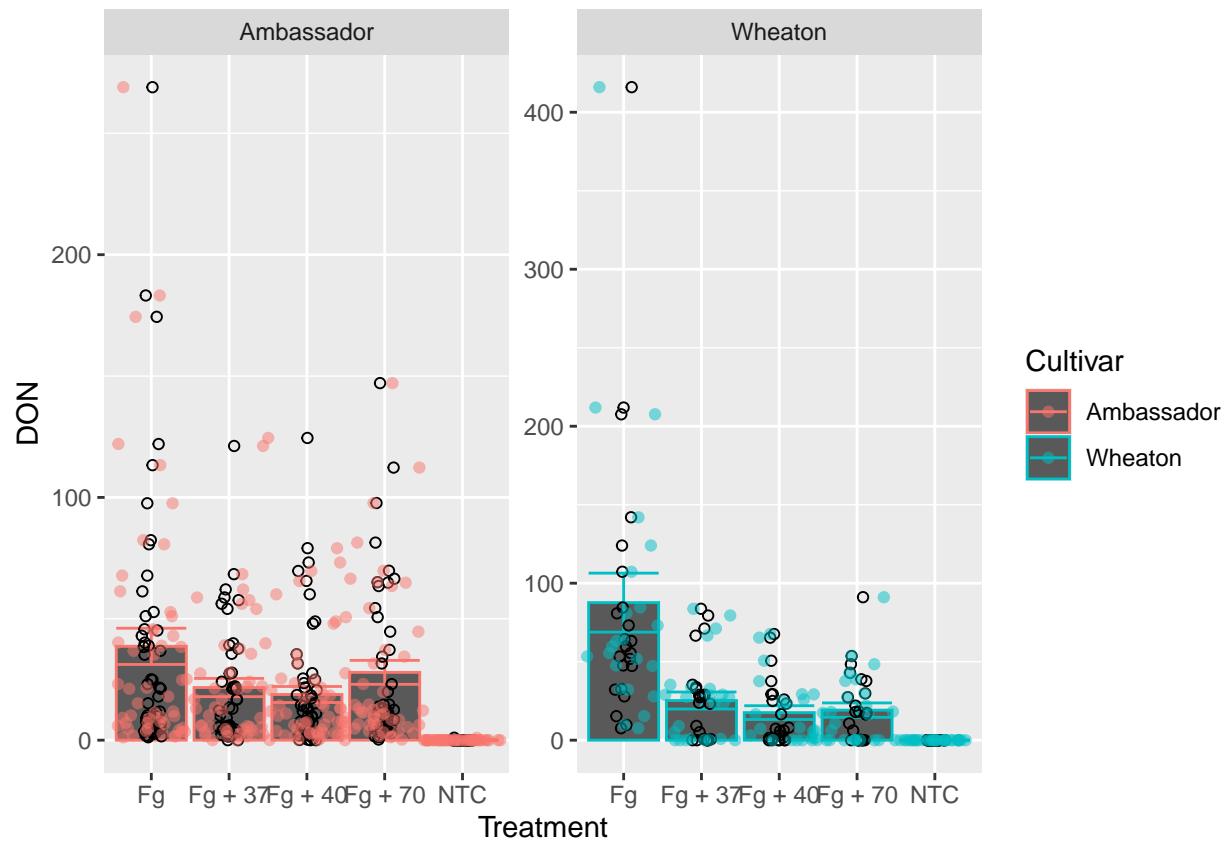
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
## Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```



```
plot8 <- ggplot(myco_data, aes(x = Treatment, y = DON, colour = Cultivar)) +
  stat_summary(fun = mean, geom = "bar", position = "dodge") +
  stat_summary(fun.data = mean_se, geom = "errorbar", position = "dodge") +
  geom_point(position = position_jitterdodge(), shape = 21, color = "black") +
  geom_jitter(width = 0.5, alpha = 0.5) + #adding transparency to the points
  facet_wrap(~Cultivar, scales = "free")
plot8
```

```
## 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()').

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



7. 4 pts. Annotate your code and push it to github. Code annotated.