

Coding challenge 4 - R_markdown

Bibek

2025-02-27

Contents

Link	1
First level header	2
Second level header	3
Third level header	4
Fourth level header	5

Link

Link to the manuscript where the data are published

Noel, Z.A., Roze, L.V., Breunig, M., Trail, F. 2022. Endophytic fungi as promising bio-control agent to protect wheat from Fusarium graminearum head blight. Plant Disease. <https://doi.org/10.1094/PDIS-06-21-1253-RE>

Questions 1

a. YAML header

In R markdown, YAML header is a section at the top of the document that contains metadata about the file. It is enclosed between “—” lines and helps to tell how to format the output file.

For example, in this markdown i have YAML header with information like title, author, date which will appear in the rendered files. I have also specified output format to generate markdown file using gfm. Other than this i have also other output options which i can choose while knitting.

b. Literate programming

Literate programming means writing both the code and explanation in the same document which make analysis easy to understand and reproduce.

Question 2

Here i will use the relative path of the Mycotoxin.csv data file with na.strings option set to “na”. I will be using the codes from the coding challenge 3 assignment.

```
#Read .csv
#using relative path
DON.database <- read.csv("Sample_data/MycotoxinData.csv", na.strings = "na")
str(DON.database) #visualize the structure of the data frame.
```

```
## '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 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 ...
```

Now, I will separate code chunk for the figures plotting the DON data, 15ADON, and Seedmass, and one for the three combined using ggarrange.

First level header

This is for DON data.

```
library(ggplot2) # loading the ggplot2 package
library(ggpubr) ##integrating multiple statistics and plots

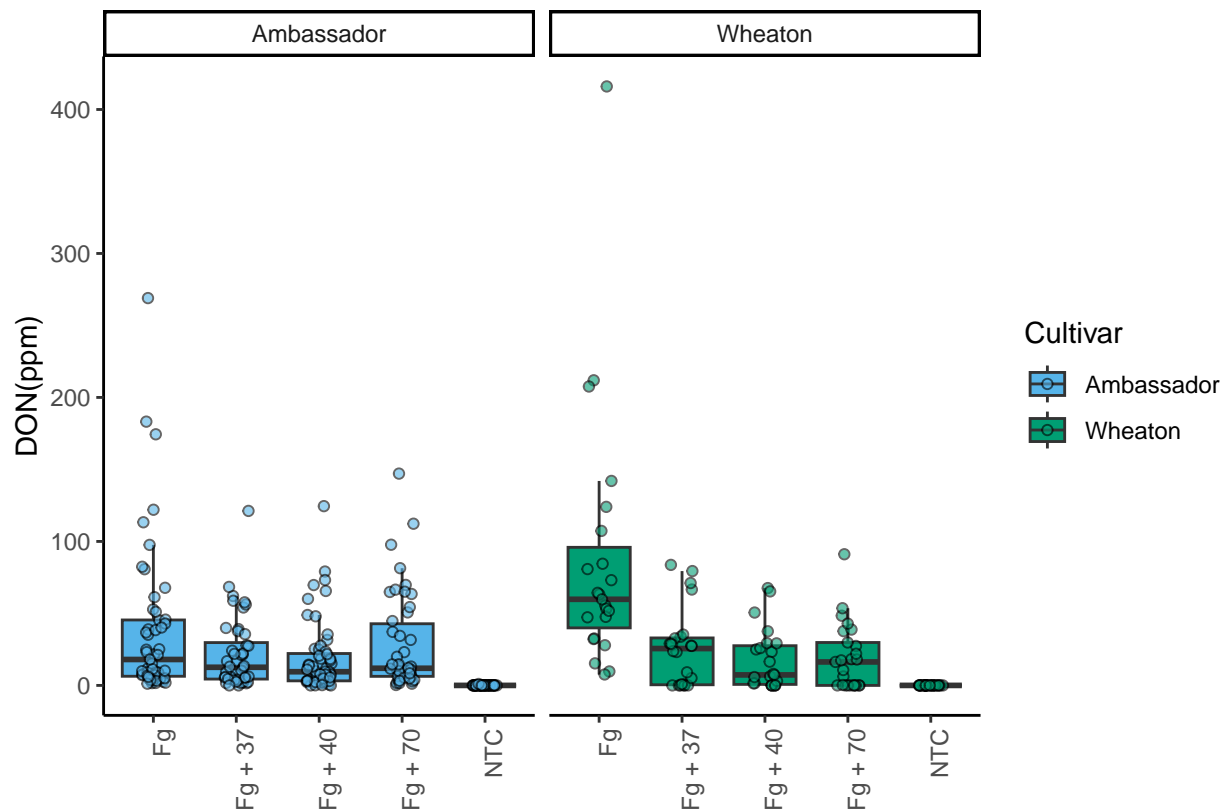
cbbPalette <-c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
myCols <- cbbPalette[c(3, 4)] ## choosing 3rd and 4th color palette from the cbbPalette.

##Question 1####
#Create a boxplot for DON (deoxynivalenol) values
DON <- ggplot(DON.database, aes(x = Treatment, y = DON, fill = Cultivar)) +
  geom_boxplot(position = position_dodge(0.85), outlier.shape = NA ) + # Create boxplots with dodge p
  geom_point(position = position_jitterdodge(0.3), shape = 21, color ="black", alpha =0.6) + #jittered
  scale_fill_manual(values = myCols) + ## Use custom fill colors for the Cultivar groups.
  ylab("DON(ppm)") + # Label the y-axis and remove x-axis label.
  xlab("") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  facet_wrap(~Cultivar) # Create separate panels for each Cultivar.

DON #Visual
```

```
## 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()').
```



```
#Reorder the factor order level
DON.database$Treatment <- factor(DON.database$Treatment, levels = c("NTC", "Fg", "Fg + 37", "Fg + 40", "Fg + 70"))

# Re-displaying the plot will now use the new Treatment order.
DON <- ggplot(DON.database, aes(x = Treatment, y = DON, fill = Cultivar)) +
  geom_boxplot(position = position_dodge(0.85), outlier.shape = NA) + # Create boxplots with dodge position
  geom_point(position = position_jitterdodge(0.3), shape = 21, color = "black", alpha = 0.6) + # jittered points
  scale_fill_manual(values = myCols) + ## Use custom fill colors for the Cultivar groups.
  ylab("DON(ppm)") + # Label the y-axis and remove x-axis label.
  xlab("") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  facet_wrap(~Cultivar) # Create separate panels for each Cultivar.
```

Second level header

This id for 15ADON plot.

```
#Change the y-variable to plot X15ADON and MassperSeed_mg.

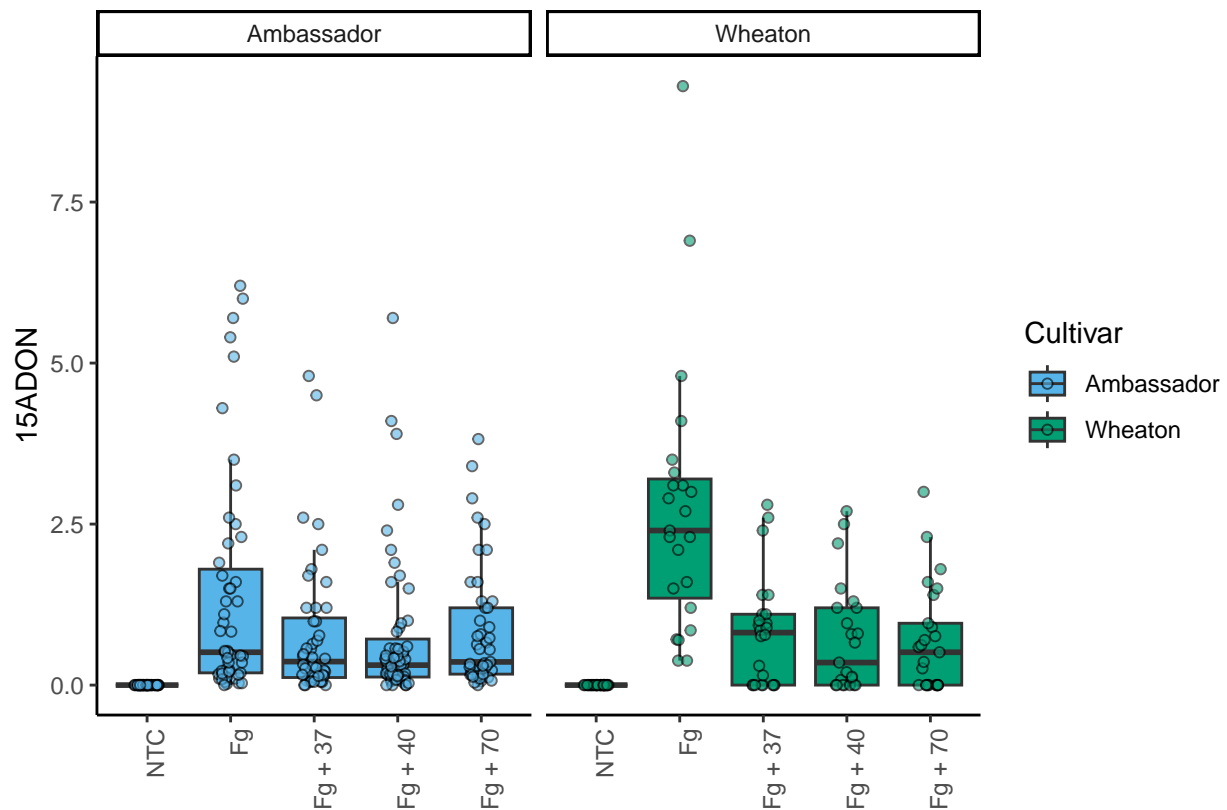
# Boxplot for X15ADON values
X15ADON <- ggplot(DON.database, aes(x = Treatment, y = X15ADON, fill = Cultivar)) +
  geom_boxplot(position = position_dodge(0.85), outlier.shape = NA) +
  geom_point(shape = 21, color = "black", position = position_jitterdodge(0.3), alpha = 0.6) +
```

```
scale_fill_manual(values = myCols) +
ylab("15ADON") + # y-axis label for X15ADON
xlab("") +
theme_classic() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
facet_wrap(~Cultivar)
```

X15ADON # Display the X15ADON plot

```
## Warning: Removed 10 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## ('geom_point()').
```



Third level header

This id for Seedmass plot.

```
# Boxplot for MassperSeed_mg (Seed Mass in mg)
```

```
MassperSeed_mg <- ggplot(DON.database, aes(x = Treatment, y = MassperSeed_mg, fill = Cultivar)) +
```

```

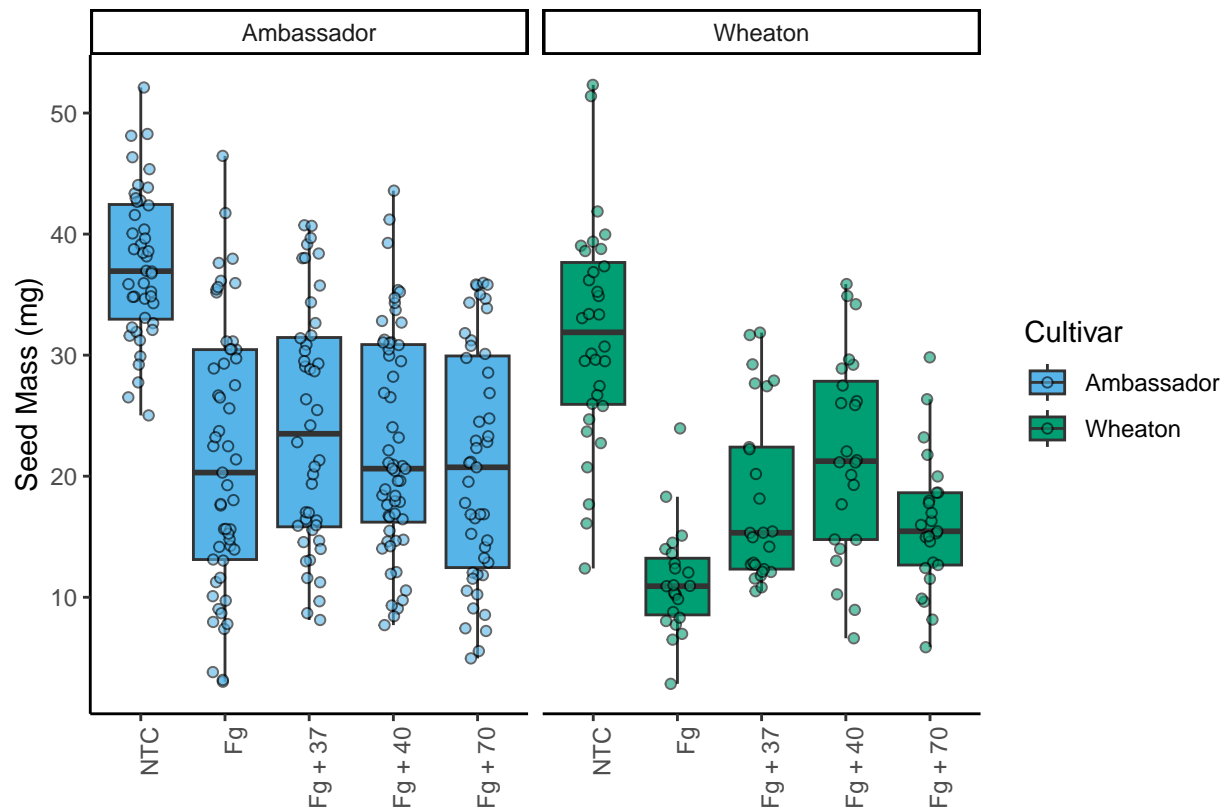
geom_boxplot(position = position_dodge(0.85), outlier.shape = NA) +
geom_point(shape = 21, color = "black", position = position_jitterdodge(0.3), alpha = 0.6) +
scale_fill_manual(values = myCols) +
ylab("Seed Mass (mg)") + # y-axis label for Seed Mass
xlab("") +
theme_classic() +
theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
facet_wrap(~Cultivar)

```

MassperSeed_mg *#Display the MassperSeed_mg plot*

Warning: Removed 2 rows containing non-finite outside the scale range
('stat_boxplot()').

Warning: Removed 2 rows containing missing values or values outside the scale range
('geom_point()').



Fourth level header

This is combined 3 plots using ggarrange.

```
# Combining all plots together
```

```
Fig_combo<-ggarrange(  ## Arrange the three plots (DON, X15ADON, MassperSeed_mg) side by side in one .  
  DON,  
  X15ADON,  
  MassperSeed_mg,  
  labels = "auto", # Add labels (a, b, c) to each plot  
  nrow = 1, # Arrange in one row  
  ncol = 3, # Three columns  
  legend = FALSE, # Do not display the legend in the combined plot  
  common.legend = T #If TRUE, a common unique legend will be created for arranged plots.  
)
```

```
## 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()').
```

```
## Warning: Removed 10 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range  
## ('geom_point()').
```

```
## Warning: Removed 2 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range  
## ('geom_point()').
```

```
## 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()').
```

```
## Warning: Removed 10 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range  
## ('geom_point()').
```

```
## Warning: Removed 2 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range  
## ('geom_point()').
```

Fig_combo

