Coding\_Challenge4\_Markdown\_SK

Shakiba Kazemian

2025-02-26

Table of Contents

[Coding Challenge 4 - Markdown 1](#_Toc191480554)

[**Manuscript Reference** 1](#_Toc191480555)

[**1. Explain the Following** 1](#_Toc191480556)

[a. YAML Header 1](#_Toc191480557)

[b. Literate Programming 2](#_Toc191480558)

[**2. Data Preparation** 2](#_Toc191480559)

[**Read the Data** 2](#_Toc191480560)

[**Data Visualization** 3](#_Toc191480561)

[**Statistical Analysis: Pairwise t-tests** 3](#_Toc191480562)

[**Combining All Plots** 3](#_Toc191480563)

# Coding Challenge 4 - Markdown

## **Manuscript Reference**

[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)

## **1. Explain the Following**

### a. YAML Header

A **YAML (Yet Another Markup Language) header** in R Markdown is a section at the top of the file enclosed by triple dashes (---). It defines metadata for the document, such as the title, author, date, and output format.

### b. Literate Programming

Literate programming means **writing programs as documents**, where explanations guide the reader, and code is embedded seamlessly. **R Markdown is a perfect example of literate programming!**

## **2. Data Preparation**

### **Read the Data**

The dataset is loaded using a **relative file path**, ensuring portability. Missing values (NA) are explicitly defined.

# Load necessary libraries  
knitr::opts\_knit$set(root.dir = getwd()) # Ensure consistent working directory  
library(ggplot2)  
library(ggpubr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

# Read the data using a relative path  
data <- read.csv("R\_Markdown/Coding\_Challenge4/MycotoxinData.csv", na.strings = "na")  
  
#data <- read.csv("R\_Markdown/Coding\_Challenge4/MycotoxinData.csv", na.strings = "na")  
  
# Display the first few rows of the dataset  
head(data)

## Treatment Cultivar BioRep MassperSeed\_mg DON X15ADON  
## 1 Fg Wheaton 2 10.291304 107.3 3.00  
## 2 Fg Wheaton 2 12.803226 32.6 0.85  
## 3 Fg Wheaton 2 2.846667 416.0 3.50  
## 4 Fg Wheaton 2 6.500000 211.9 3.10  
## 5 Fg Wheaton 2 10.179167 124.0 4.80  
## 6 Fg Wheaton 2 12.044444 73.1 3.30

### **Data Visualization**

#### Plotting Individual Figures

Separate plots for DON data, 15ADON, and Seedmass using ggplot2.

# DON Plot  
DON\_plot <- ggplot(data, aes(x = Treatment, y = DON)) +  
 geom\_boxplot() +  
 theme\_minimal() +  
 ggtitle("DON Levels by Treatment")  
  
# 15ADON Plot  
ADON\_plot <- ggplot(data, aes(x = Treatment, y = X15ADON)) +  
 geom\_boxplot() +  
 theme\_minimal() +  
 ggtitle("15ADON Levels by Treatment")  
  
# Seedmass Plot  
SeedMass\_plot <- ggplot(data, aes(x = Treatment, y = MassperSeed\_mg)) +  
 geom\_boxplot() +  
 theme\_minimal() +  
 ggtitle("Seed Mass per mg by Treatment")

### **Statistical Analysis: Pairwise t-tests**

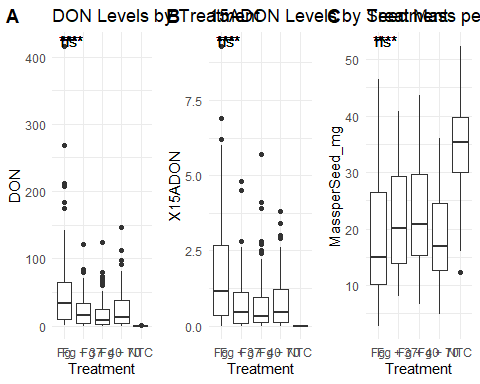
Adding pairwise t-tests to each plot using stat\_compare\_means().

# Add Pairwise t-tests to Each Plot  
DON\_plot\_ttest <- DON\_plot + stat\_compare\_means(method = "t.test", aes(group = Treatment), label = "p.signif")  
ADON\_plot\_ttest <- ADON\_plot + stat\_compare\_means(method = "t.test", aes(group = Treatment), label = "p.signif")  
SeedMass\_plot\_ttest <- SeedMass\_plot + stat\_compare\_means(method = "t.test", aes(group = Treatment), label = "p.signif")

### **Combining All Plots**

Using ggarrange() to create a single figure with all plots.

# Combine all updated plots into one figure  
Combined\_Plot\_ttest <- ggarrange(  
 DON\_plot\_ttest, ADON\_plot\_ttest, SeedMass\_plot\_ttest,   
 labels = c("A", "B", "C"),   
 ncol = 3, nrow = 1,   
 common.legend = TRUE,   
 legend = "right"  
)  
  
# Display the final combined plot with t-tests  
print(Combined\_Plot\_ttest)



# Save the final combined figure  
ggsave("Combined\_Plot\_ttest.png", Combined\_Plot\_ttest, width = 12, height = 5, dpi = 300)