Enhancing Scientific Inquiry with R for Data Manipulation and Visualization

Alexandre Paquette

2024-04-15

Table of Contents

# Introduction

Data visualization is a fundamental tool across a multitude of disciplines, serving as a conduit for conveying complex information in a visually accessible and palatable manner. In scientific research, where the exploration and interpretation of data are paramount, effective visualization techniques play a pivotal role in interpreting patterns, trends, and relationships within datasets. By transforming raw data into intuitive visual representations, researchers can gain deeper insights into their findings and communicate them with clarity and precision. In this report, we walk through the process of producing data visualizations for scientific research, exploring the methodologies and strategies that underline the creation of impactful visual narratives.

## Understanding Scientific Research

Scientific research constitutes a meticulous and systematic pursuit of knowledge, aiming to present phenomena, develop theories, and solve practical problems through rigorous methodologies via experimentation, observation, analysis, and interpretation. Central to this endeavor is the collection and processing of data, which serves as the cornerstone for empirical investigation. However, the mere accumulation of data is insufficient without an effective way to interpret and communicate. Data visualization emerges as a vital tool in this process, enabling researchers to distill complex datasets into intuitive visual representations that facilitate hypothesis confirmation or refutation. By harnessing the power of visualization, researchers can discern patterns, trends, and relationships within data, thereby advancing our understanding of the natural world and informing evidence-based decision-making.

## A Real-world Case Study

The real-world case study at the heart of our visualization endeavor revolves around an investigation into the potential benefits of utilizing a bio-priming agent comprising *Bacillus cereus* and *Pseudomonas alcaligenes* during the germination stage of plant development. The overarching objective of this study is to assess whether the application of this bio-priming agent contributes to the overall health and vigor of plants upon completion of the germination process.

The study aims to demonstrate the efficacy of the biopriming agent in enhancing plant growth, resilience, and productivity by leveraging the natural symbiotic relationship between *Bacillus cereus* and *Pseudomonas alcaligenes*. Through a series of controlled experiments and rigorous data collection protocols, researchers seek to evaluate various physiological parameters, including seed germination rates, seedling vigor, root development, and resistance to environmental stressors.

By employing a multidisciplinary approach that integrates microbiology, plant biology, and agronomy, the study endeavors to shed light on the potential applications of bio-priming techniques in sustainable agriculture practices. The findings from this research hold promise for informing agricultural strategies aimed at optimizing plant health and productivity while minimizing reliance on synthetic inputs and agrochemicals.

This real-world case study serves as the foundation for our exploration of data visualization techniques, providing a concrete context for the application of visual analytics in scientific research and innovation within the agricultural domain.

# Research Question

The focal point of our investigation revolves around a central research question: How do we effectively transform real-world data from scientific studies into meaningful visual representations that facilitate insight and understanding? This inquiry stems from the imperative to distill scientific data into actionable insights, particularly within the context of scientific research.

## Rationale Behind the Study

Our study aims to shed light on the process of taking real-world data from a scientific study, manipulating it to extract meaningful insights, and visualizing these insights for enhanced comprehension. Specifically, our interest lies in exploring the relationship between the application of a PGPB (Plant Growth Promoting Bacteria) consortium as a biopriming agent on the seeds of various plant species and the resulting health and vigor of plants during the germination growth stage of the plant life.

Data visualization serves as a powerful tool in this endeavor, enabling us to visually compare growth parameters and trends between experimental bioprimed seeds and control non-exposed seeds. Through visual representations, we seek to uncover patterns, trends, and correlations within the growth data, providing valuable insights into the efficacy of biopriming with PGPB as a plant growth promoter.

## Methodology

Our methodology revolves around leveraging programming tools, specifically R and relevant libraries, to manipulate and visualize the data obtained from the scientific study. By harnessing the capabilities of R programming and associated libraries, we aim to streamline the data processing and visualization workflows, facilitating efficient analysis and interpretation of the dataset.

The specific libraries and techniques employed will be determined based on the nature of the data and the visualization requirements. However, our methodology will prioritize clarity, accuracy, and reproducibility, ensuring that the visualizations produced effectively communicate the insights derived from the data.

This approach allows us to not only explore the intricacies of data manipulation and visualization within the context of scientific research but also lays the foundation for future research endeavors in data-driven inquiry.

# Data Manipulation

These are the libraries we’ll be using

# for data manipulation:  
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

library(tidyr)  
  
# for visualization  
library(ggplot2)

Load the data from the .csv and save it in a variable

bioData <- read.csv("rawData.csv")

## Data Cleanup

Replace all NA values with 0

bioData[is.na(bioData)] <- 0

Setting values to be all lower case for consistency (except for Group column)

bioData$Plant.Type <- tolower(bioData$Plant.Type)  
bioData$Control.Experimental <- tolower(bioData$Control.Experimental)  
bioData$Common.Name <- tolower(bioData$Common.Name)  
bioData$Scientific.Name <- tolower(bioData$Scientific.Name)

Converting the Group, Plant Type, and Control/Experimental columns to a categorical data type (factor)

bioData$Group <- factor(bioData$Group)  
bioData$Plant.Type <- factor(bioData$Plant.Type)  
bioData$Control.Experimental <- factor(bioData$Control.Experimental)

We will be renaming the columns to fix spelling mistake and apply a consistent format to our dataset.

bioData <- bioData %>%  
 rename(Common\_Name = Common.Name,  
 Scientific\_Name = Scientific.Name,  
 Plant\_Type = Plant.Type,  
 Controler\_Experimental = Control.Experimental,  
 Root\_Count = number.of.roots.over.all,  
 Sprout\_Count = number.of.sprouts.over.all,  
 Sprout\_Week\_1\_Change\_cm = Change.in.week.1..sprout.,  
 Sprout\_Week\_2\_Change\_cm = change.in.week.2..sprout.,  
 Sprout\_Week\_3\_Change\_cm = change.in.week.3..sprout.,  
 Sprout\_Week\_4\_Change\_cm = change.in.week.4..sprout.,  
 Seed\_Amount = seed.amount,  
 Seed\_Amount\_Variation = seed.amount.variation..,  
 Root\_Week\_1\_Change\_cm = Change.in.week.1..root.,  
 Root\_Week\_2\_Change\_cm = change.in.week.2..root.,  
 Root\_Week\_3\_Change\_cm = change.in.week.3..root.,  
 Root\_Week\_4\_Change\_cm = change.in.week.4..root.,  
 Root\_Lengths\_cm = Indavidual.lengrths..roots.,  
 Sprout\_Lengths\_cm = Indavidual.langth..sprouts.  
 )

We want to remove entries that didn’t show any growth for both sprout and roots. This means checking for rows that have empty columns for both root and sprout growth data.

bioData <- bioData %>% filter(Root\_Lengths\_cm != "" | Sprout\_Lengths\_cm != "")

Now we want to combine matching plant names by control/experimental to simplify our data further.

# remove leading and trailing commas  
remove\_commas <- function(x) {  
 gsub("^,|,$", "", x)  
}

bioData <- bioData %>%  
 group\_by(Scientific\_Name, Controler\_Experimental) %>%  
 summarize(  
 Common\_Name = first(Common\_Name),  
 Root\_Lengths\_cm = remove\_commas(paste(Root\_Lengths\_cm, collapse = ", ")),  
 Plant\_Type = first(Plant\_Type),  
 Root\_Count = sum(Root\_Count),  
 Sprout\_Count = sum(Sprout\_Count),  
 Sprout\_Week\_1\_Change\_cm = mean(Sprout\_Week\_1\_Change\_cm),  
 Sprout\_Week\_2\_Change\_cm = mean(Sprout\_Week\_2\_Change\_cm),  
 Sprout\_Week\_3\_Change\_cm = mean(Sprout\_Week\_3\_Change\_cm),  
 Sprout\_Week\_4\_Change\_cm = mean(Sprout\_Week\_4\_Change\_cm),  
 Seed\_Amount = sum(Seed\_Amount),  
 Seed\_Amount\_Variation = first(Seed\_Amount\_Variation),  
 Root\_Week\_1\_Change\_cm = mean(Root\_Week\_1\_Change\_cm),  
 Root\_Week\_2\_Change\_cm = mean(Root\_Week\_2\_Change\_cm),  
 Root\_Week\_3\_Change\_cm = mean(Root\_Week\_3\_Change\_cm),  
 Root\_Week\_4\_Change\_cm = mean(Root\_Week\_4\_Change\_cm),  
 Sprout\_Lengths\_cm = remove\_commas(paste(Sprout\_Lengths\_cm, collapse = ", "))  
 ) %>%  
 ungroup()

## `summarise()` has grouped output by 'Scientific\_Name'. You can override using  
## the `.groups` argument.

We will generate IDs for each entry so we can identify them uniquely.

bioData <- bioData %>% mutate(id = row\_number())  
bioData <- bioData %>% select(id, everything())

We are extracting the multiple values from the columns so that they can be read as individual numbers

rootLengthsData <- bioData %>% separate\_rows(Root\_Lengths\_cm, sep = ",") %>% select(id, Root\_Length\_cm = Root\_Lengths\_cm)

Next we want to convert our Root Length to numeric

rootLengthsData <- rootLengthsData %>% mutate(Root\_Length\_cm = as.numeric(Root\_Length\_cm))

We’ve now extracted all the individual Root Length for each plant. This will allow us to visualize this data more effectively if necessary. Regardless, this data needs to exist in this raw form for further investigation.

Now we want to repeat this step, but for the sprout length column. Now that we know the process, we can complete this in a single step.

sproutLengthsData <- bioData %>%  
 separate\_rows(Sprout\_Lengths\_cm, sep = ",") %>%  
 select(id, Sprout\_Length\_cm = Sprout\_Lengths\_cm) %>%  
 mutate(Sprout\_Length\_cm = as.numeric(Sprout\_Length\_cm))  
  
sproutLengthsData[is.na(sproutLengthsData)] <- 0

Finally, we can remove the old columns from the dataset

bioData <- bioData %>% select(-Root\_Lengths\_cm, -Sprout\_Lengths\_cm)

## Derive New Data

Next we need to extract insights from our data. We will be adding columns to our main dataset to calculate the following: - Average and median root/stem lengths - Average growth over time for root and sprout

### Calculate average and median

This method will be used to perform calculations on a given dataset and append the results to the main dataset. In this case, we’ll be passing bioData as our main dataset, and then rootLengthsData or sproutLengthsData depending on the calculation we want to make.

calculate\_summary <- function(main\_data, summary\_data, id\_col, summary\_col, summary\_name, summary\_method){  
 summary\_values <- summary\_data %>%  
 group\_by({{id\_col}}) %>%  
 summarize(  
 {{summary\_name}} := summary\_method({{summary\_col}})  
 )  
   
 result <- left\_join(main\_data, summary\_values, by = join\_by({{id\_col}}))  
 return(result)  
}

bioData <- calculate\_summary(bioData, rootLengthsData, id, Root\_Length\_cm, avg\_root\_length\_cm, mean) # average root length  
bioData <- calculate\_summary(bioData, sproutLengthsData, id, Sprout\_Length\_cm, avg\_sprout\_length\_cm, mean) # average sprout length  
bioData <- calculate\_summary(bioData, rootLengthsData, id, Root\_Length\_cm, median\_root\_length\_cm, median) # Median sprout length  
bioData <- calculate\_summary(bioData, sproutLengthsData, id, Sprout\_Length\_cm, median\_sprout\_length\_cm, median) # Median sprout length  
bioData <- calculate\_summary(bioData, rootLengthsData, id, Root\_Length\_cm, root\_length\_sd, sd) # root length standard deviation  
bioData <- calculate\_summary(bioData, sproutLengthsData, id, Sprout\_Length\_cm, sprout\_length\_sd, sd) # sprout length standard deviation  
  
# set all na values to 0  
bioData[is.na(bioData)] <- 0

### Calculate average growth over time

bioData <- cbind(bioData,   
 avg\_root\_growth\_per\_week\_cm =  
 rowSums(bioData %>%   
 select(Root\_Week\_1\_Change\_cm, Root\_Week\_2\_Change\_cm, Root\_Week\_3\_Change\_cm, Root\_Week\_4\_Change\_cm)))

bioData <- cbind(bioData,   
 avg\_sprout\_growth\_per\_week\_cm =  
 rowSums(bioData %>%   
 select(Sprout\_Week\_1\_Change\_cm, Sprout\_Week\_2\_Change\_cm, Sprout\_Week\_3\_Change\_cm, Sprout\_Week\_4\_Change\_cm)))

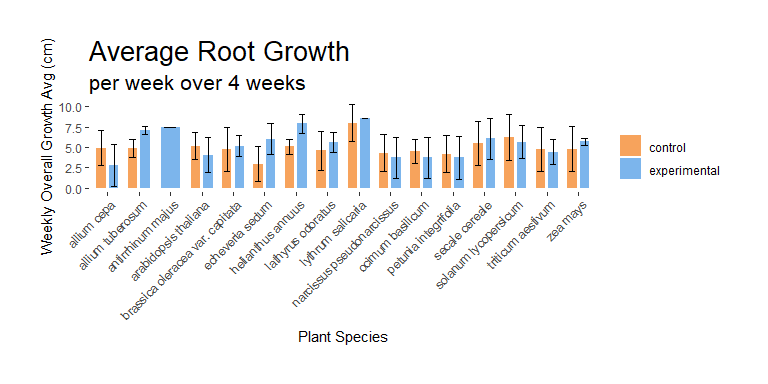
# Data Visualization

create\_bar\_plot <- function(data, x\_val, y\_val, fill\_val, sd = NULL, titleName, subtitleName = NULL, x\_title, y\_title, legend = TRUE, ...){  
 colors <- c("experimental" = "#7cb5ec", "control" = "#f7a35c")  
   
 chart <- ggplot(data, aes(x = {{x\_val}}, y = {{y\_val}}, fill = {{fill\_val}})) +  
 geom\_bar(stat = "identity", position = position\_dodge(width = 0.8), width = 0.6) +  
 scale\_fill\_manual(values = colors, name = "") +  
 labs(title = titleName, subtitle = subtitleName, x = x\_title, y = y\_title) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1),  
 plot.title = element\_text(size = 20),  
 plot.margin = margin(30, 30, 30, 30, "pt"),  
 plot.background = element\_rect(fill = "white"), # Set plot background color  
 panel.background = element\_rect(fill = "white")) # Rotate x-axis labels for better readability  
   
 if(!missing(sd)){  
 chart <- chart + geom\_errorbar(aes(ymin = pmax({{y\_val}} - {{sd}}, 0), ymax = {{y\_val}} + {{sd}}), width = 0.4, position = position\_dodge(width = 0.8))  
 }  
   
 if(!missing(subtitleName)){  
 chart <- chart + theme(plot.subtitle = element\_text(size = 16))  
 }  
   
 if(legend == FALSE){  
 chart <- chart + guides(fill = FALSE)  
 }  
   
 fileName <- paste("charts/",titleName,".png")  
 suppressMessages(ggsave(fileName, chart))  
 return(chart)  
}

What do we need to show? - growth differences for roots and sprouts between control/experimental

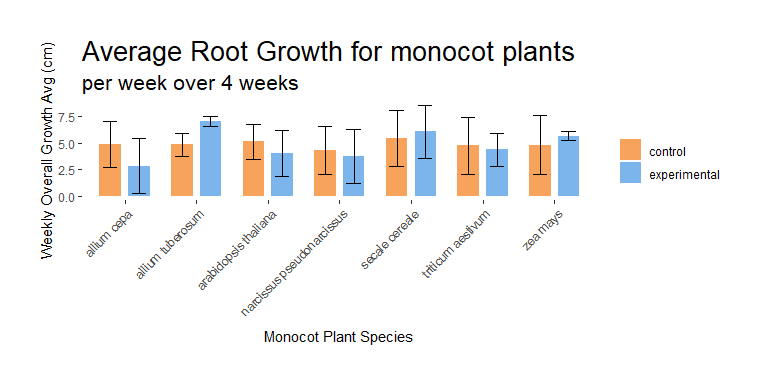
## Root Growth Average comparison

create\_bar\_plot(data = bioData,  
 x\_val = Scientific\_Name,  
 y\_val = avg\_root\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = root\_length\_sd,  
 titleName = "Average Root Growth",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")



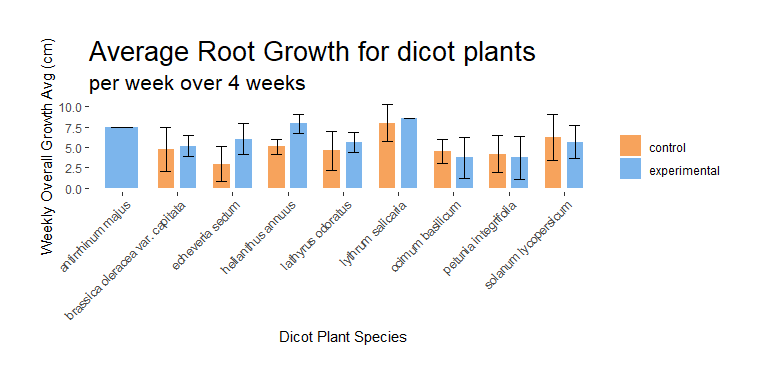
This chart illustrates the average root growth for monocot plants comparing control and experimental.

create\_bar\_plot(data = subset(bioData, Plant\_Type == "monocot"),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_root\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = root\_length\_sd,  
 titleName = "Average Root Growth for monocot plants",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Monocot Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")



This chart illustrates the average root growth for dicot plants comparing control and experimental.

create\_bar\_plot(data = subset(bioData, Plant\_Type == "dicot"),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_root\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = root\_length\_sd,  
 titleName = "Average Root Growth for dicot plants",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Dicot Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")

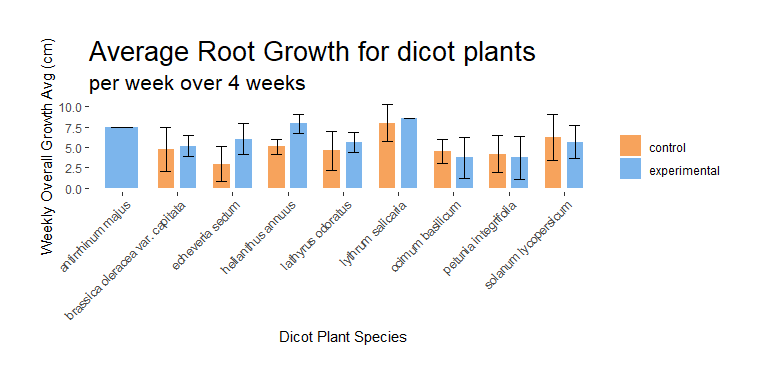


This generates a chart for each individual plant that compares the control and experimental sample for root growth.

unique\_plant\_names <- unique(bioData$Scientific\_Name)  
# Iterate through each unique plant name  
for(plant\_name in unique\_plant\_names){  
 # Generate a chart for the current plant  
 create\_bar\_plot(data = filter(bioData, Scientific\_Name == plant\_name),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_root\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = root\_length\_sd,  
 titleName = paste("Average Root Growth for",plant\_name),  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")  
}

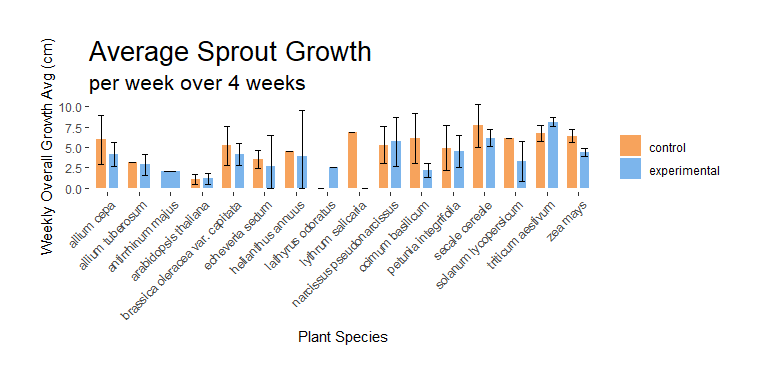
This generates a chart for the average root growth for all dicot plants comparing the control and experimental samples

create\_bar\_plot(data = subset(bioData, Plant\_Type == "dicot"),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_root\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = root\_length\_sd,  
 titleName = "Average Root Growth for dicot plants",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Dicot Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")



## Sprout Growth Average comparison

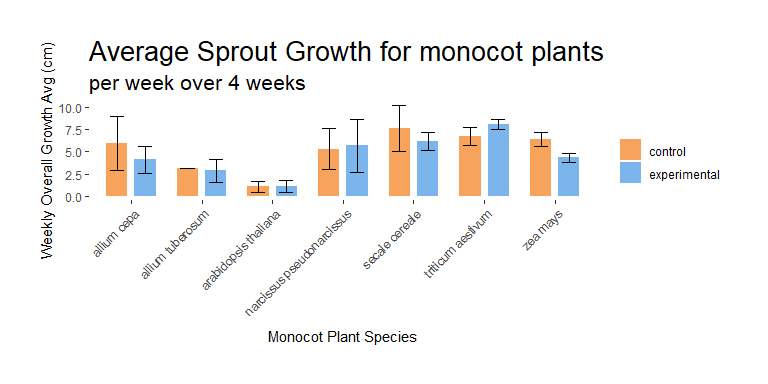
create\_bar\_plot(data = bioData,  
 x\_val = Scientific\_Name,  
 y\_val = avg\_sprout\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = sprout\_length\_sd,  
 titleName = "Average Sprout Growth",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")



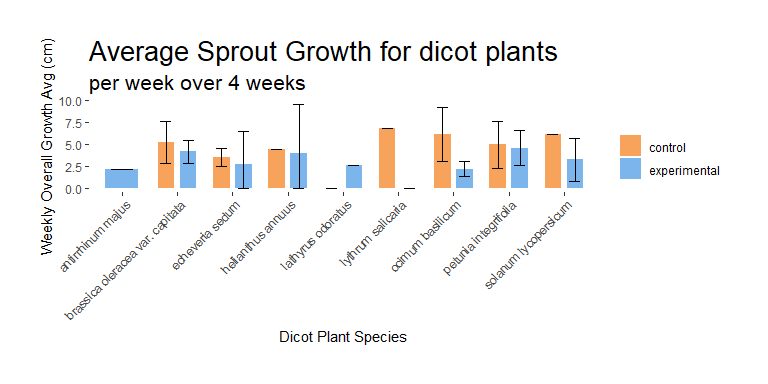
This generates a chart for each individual plant that compares the control and experimental sample for sprout growth.

# Iterate through each unique plant name  
for(plant\_name in unique\_plant\_names){  
 # Generate a chart for the current plant  
 create\_bar\_plot(data = filter(bioData, Scientific\_Name == plant\_name),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_sprout\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = sprout\_length\_sd,  
 titleName = paste("Average Sprout Growth for",plant\_name),  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")  
}

create\_bar\_plot(data = subset(bioData, Plant\_Type == "monocot"),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_sprout\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = sprout\_length\_sd,  
 titleName = "Average Sprout Growth for monocot plants",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Monocot Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")

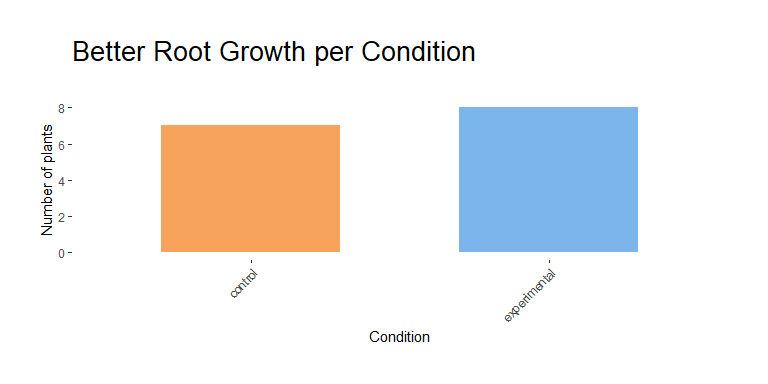


create\_bar\_plot(data = subset(bioData, Plant\_Type == "dicot"),  
 x\_val = Scientific\_Name,  
 y\_val = avg\_sprout\_growth\_per\_week\_cm,  
 fill\_val = Controler\_Experimental,  
 sd = sprout\_length\_sd,  
 titleName = "Average Sprout Growth for dicot plants",  
 subtitleName = "per week over 4 weeks",  
 x\_title = "Dicot Plant Species",  
 y\_title = "Weekly Overall Growth Avg (cm)")



* control best count
* experimental best count

better\_growth <- bioData %>%  
 group\_by(Scientific\_Name) %>%   
 summarise(  
 better\_growth\_control = sum(avg\_root\_growth\_per\_week\_cm[Controler\_Experimental == "control"] > avg\_root\_growth\_per\_week\_cm[Controler\_Experimental == "experimental"]),  
 better\_growth\_experimental = sum(avg\_root\_growth\_per\_week\_cm[Controler\_Experimental == "experimental"] > avg\_root\_growth\_per\_week\_cm[Controler\_Experimental == "control"])  
 ) %>%  
 summarise(  
 better\_growth\_control = sum(better\_growth\_control),  
 better\_growth\_experimental = sum(better\_growth\_experimental)  
 ) %>%  
 pivot\_longer(cols = starts\_with("better\_growth"), names\_to = "Category", values\_to = "total\_count") %>%  
 mutate(Category = gsub("better\_growth\_", "",Category)) %>%  
 mutate (sd = 0)  
  
chart <- create\_bar\_plot(data = better\_growth,  
 x\_val = Category,  
 y\_val = total\_count,  
 fill\_val = Category,  
 titleName = "Better Root Growth per Condition",  
 subtitleName = "",  
 x\_title = "Condition",  
 y\_title = "Number of plants",  
 legend = FALSE)  
chart



* growth over time chart

# Conclusion

# Glossary of Terms

Biopriming

The process of coating the seed with a plant-growth promoting bacteria consortium comprised of Basillus ceres and pusdomonas

Monocot plant

The seeds of these plants typically contain a single embryonic leaf

Dicot plant

A plant whose germinating seed contain two embryonic leaves

Embryonic leaf

The plant embryo, also known as cotyledon