

Statistical Consulting

Niloofer Khalajzadeh

2023-10-09

```
library(MASS)
library(dplyr)
library(lme4)
library(ordinal)
library(ggplot2)
library(tidyverse)
library(corrplot)
library(caret)
library(ggpubr)
library(e1071)
library(rpart)
library(reshape2)
library(Metrics)
library(pracma)
library(emmeans)
library(multcomp)
```

```
experiment_setup <- read.csv("experiment_setup.csv", sep = ";")
disease_score <- read.csv("disease_score.csv", sep = ";")
```

```
get_cultivar_label <- function(x) {
  experiment_setup[experiment_setup$Cultivar == x,][1,]$Sensitivity
}
```

```
disease_score$sensitivity <- unlist(lapply(disease_score$Cultivar, get_cultivar_label))
```

```
n <- dim(disease_score)[1]
disease_score <- disease_score[1:(n-3),]
```

```
disease_score$T9.RGR..cm. <- as.numeric(gsub(",", ".", disease_score$T9.RGR..cm.))
disease_score$T9.length..cm. <- as.numeric(gsub(",", ".", disease_score$T9.length..cm.))
disease_score <- disease_score %>%
  filter(!is.na(sensitivity), sensitivity != "")
disease_score <- disease_score %>% filter(T9.length..cm. >= 0, T9.RGR..cm. >= 0)
```

```
disease_score_clean <- subset(disease_score, Pathogen != "negative control")
disease_score_clean <- na.omit(disease_score_clean)
```

```
disease_score_clean$Treatment <- disease_score_clean$Treatment..1.34.
disease_score_clean$Block <- disease_score_clean$Block.nr.
```

```

disease_score_clean$Plant_nr <- disease_score_clean$Plant.nr.
disease_score_clean$T3 <- disease_score_clean$T3.disease..0.1.
disease_score_clean$T4 <- disease_score_clean$T4.disease...leaves.
disease_score_clean$T5_per <- disease_score_clean$T5.disease...
disease_score_clean$T5 <- disease_score_clean$T5.affected..0.1.
disease_score_clean$T6_per <- disease_score_clean$T6.disease...
disease_score_clean$T6 <- disease_score_clean$T6.affected..0.1.
disease_score_clean$T7_per <- disease_score_clean$T7.disease...
disease_score_clean$T7 <- disease_score_clean$T7.affected..0.1.
disease_score_clean$T8_per <- disease_score_clean$T8.disease...
disease_score_clean$T8 <- disease_score_clean$T8.affected..0.1.
disease_score_clean$T8_con <- disease_score_clean$T8.mate.disease..0.3.
disease_score_clean$T9_per <- disease_score_clean$T9.disease...
disease_score_clean$T9 <- disease_score_clean$T9.affected..0.1.
disease_score_clean$T9_con <- disease_score_clean$T9.mate.disease..0.3.
disease_score_clean$T5_length <- disease_score_clean$T5.length..cm.
disease_score_clean$T5_RGR <- disease_score_clean$T5.Relative.Growth.Rate.RGR..cm.
disease_score_clean$T9_length <- disease_score_clean$T9.length..cm.
disease_score_clean$T9_RGR <- disease_score_clean$T9.RGR..cm.
disease_score_clean$sensitivity_f <- disease_score_clean$sensitivity

disease_score_clean$time_to_first <- apply(disease_score_clean[,c("T3", "T5", "T6", "T7", "T8","T9")], 1,
  affected_week <- which(x == 1)
  if (length(affected_week) == 0) {
    return(70)
  } else {
    return(affected_week[1])
  }
})

new_dataframe <- data.frame()

new_dataframe <- disease_score_clean %>%
  select(sensitivity_f, Treatment, Soiltype, Cultivar, Pathogen, Plant_nr, Block, T3, T4, T5, T5_per, T6, T6_per, T7, T7_per, T8, T8_per, T8_con, T9, T9_per, T9_con, T9_length, T9_RGR)

new_dataframe$sensitivity_f <- factor(new_dataframe$sensitivity_f, ordered = TRUE,
  levels = c("Very sensitive", "Sensitive", "Reasonably sensitive",
    "intermediate with some damage", "Relatively tolerant",
    "highly tolerant", "Resistant"))

new_dataframe$T3 <- ifelse(new_dataframe$T3 == 0, 0, 1)
new_dataframe$T5 <- ifelse(new_dataframe$T5 == 0, 0, 1)
new_dataframe$T6 <- ifelse(new_dataframe$T6 == 0, 0, 1)
new_dataframe$T7 <- ifelse(new_dataframe$T7 == 0, 0, 1)
new_dataframe$T8 <- ifelse(new_dataframe$T8 == 0, 0, 1)
new_dataframe$T9 <- ifelse(new_dataframe$T9 == 0, 0, 1)

new_dataframe$Pathogen <- factor(disease_score_clean$Pathogen)
new_dataframe$Soiltype <- factor(disease_score_clean$Soiltype)
new_dataframe$Cultivar <- factor(disease_score_clean$Cultivar)
new_dataframe$Treatment <- factor(disease_score_clean$Treatment)

new_dataframe$T8_con <- factor(new_dataframe$T8_con, levels = 0:3)

```

```

new_dataframe$T9_con <- factor(new_dataframe$T9_con, levels = 0:3)
new_dataframe$T5_length <- as.numeric(sub(",", ".", new_dataframe$T5_length, fixed = TRUE))
new_dataframe$T9_length <- as.numeric(sub(",", ".", new_dataframe$T9_length, fixed = TRUE))
new_dataframe$T5_RGR <- as.numeric(sub(",", ".", new_dataframe$T5_RGR, fixed = TRUE))
new_dataframe$T9_RGR <- as.numeric(sub(",", ".", new_dataframe$T9_RGR, fixed = TRUE))

```

```
summary(new_dataframe)
```

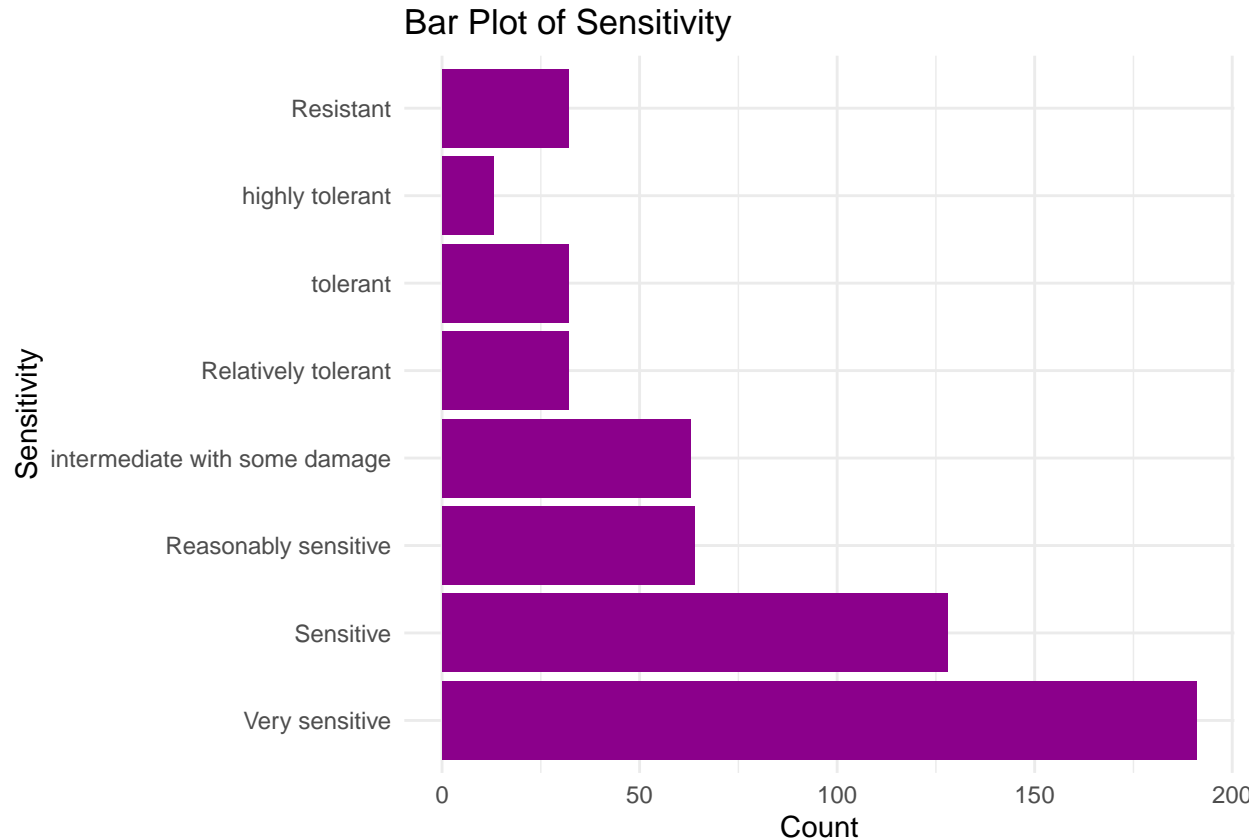
```

##              sensitivity_f Treatment      Soiltype
## Very sensitive      :191  3      : 32  Potting Soil:492
## Sensitive          :128  5      : 32  Soil      : 63
## Reasonably sensitive      : 64  6      : 32
## intermediate with some damage: 63  8      : 32
## Relatively tolerant      : 32 10      : 32
## tolerant            : 32 12      : 32
## (Other)              : 45 (Other):363
##      Cultivar      Pathogen      Plant_nr      Block
## Kennedy      :127  Conidia      :492  Min.      : 33.0  Min.      :1.000
## 130.857.000 : 32  Microsclerotia: 63  1st Qu.: 236.5  1st Qu.:2.000
## 91.023.000  : 32                      Median : 504.0  Median :3.000
## Abbey Purple: 32                      Mean   : 518.5  Mean   :2.526
## Alamos      : 32                      3rd Qu.: 802.5  3rd Qu.:4.000
## Antonov     : 32                      Max.    :1056.0  Max.    :4.000
## (Other)     :268
##      T3      T4      T5      T5_per
## Min.      :0.00000  Min.      : 0.0000  Min.      :0.000  Min.      : 0.00
## 1st Qu.:0.00000  1st Qu.: 0.0000  1st Qu.:0.000  1st Qu.: 0.00
## Median :0.00000  Median : 0.0000  Median :0.000  Median : 5.00
## Mean   :0.01622  Mean   : 0.8883  Mean   :0.436  Mean   : 19.41
## 3rd Qu.:0.00000  3rd Qu.: 0.0000  3rd Qu.:1.000  3rd Qu.: 30.00
## Max.    :1.00000  Max.    :17.0000  Max.    :1.000  Max.    :100.00
##
##      T6      T6_per      T7      T7_per
## Min.      :0.0000  Min.      : 0.00  Min.      :0.0000  Min.      : 0.00
## 1st Qu.:0.0000  1st Qu.: 0.00  1st Qu.:0.0000  1st Qu.: 5.00
## Median :1.0000  Median : 10.00  Median :1.0000  Median : 30.00
## Mean   :0.5459  Mean   : 24.09  Mean   :0.7027  Mean   : 35.13
## 3rd Qu.:1.0000  3rd Qu.: 45.00  3rd Qu.:1.0000  3rd Qu.: 60.00
## Max.    :1.0000  Max.    :100.00  Max.    :1.0000  Max.    :100.00
##
##      T8      T8_per      T9      T9_per
## Min.      :0.000  Min.      : 0.00  Min.      :0.0000  Min.      : 0.00
## 1st Qu.:1.000  1st Qu.: 10.00  1st Qu.:1.0000  1st Qu.: 10.00
## Median :1.000  Median : 50.00  Median :1.0000  Median : 60.00
## Mean   :0.764  Mean   : 42.97  Mean   :0.8036  Mean   : 50.04
## 3rd Qu.:1.000  3rd Qu.: 70.00  3rd Qu.:1.0000  3rd Qu.: 80.00
## Max.    :1.000  Max.    :100.00  Max.    :1.0000  Max.    :100.00
##
##      time_to_first      T5_length      T9_length      T5_RGR
## Min.      : 1.00  Min.      :23.70  Min.      :12.41  Min.      : 6.52
## 1st Qu.: 2.00  1st Qu.:55.68  1st Qu.:65.39  1st Qu.:43.70
## Median : 3.00  Median :62.79  Median :73.21  Median :51.05
## Mean   :15.85  Mean   :61.39  Mean   :72.00  Mean   :49.19

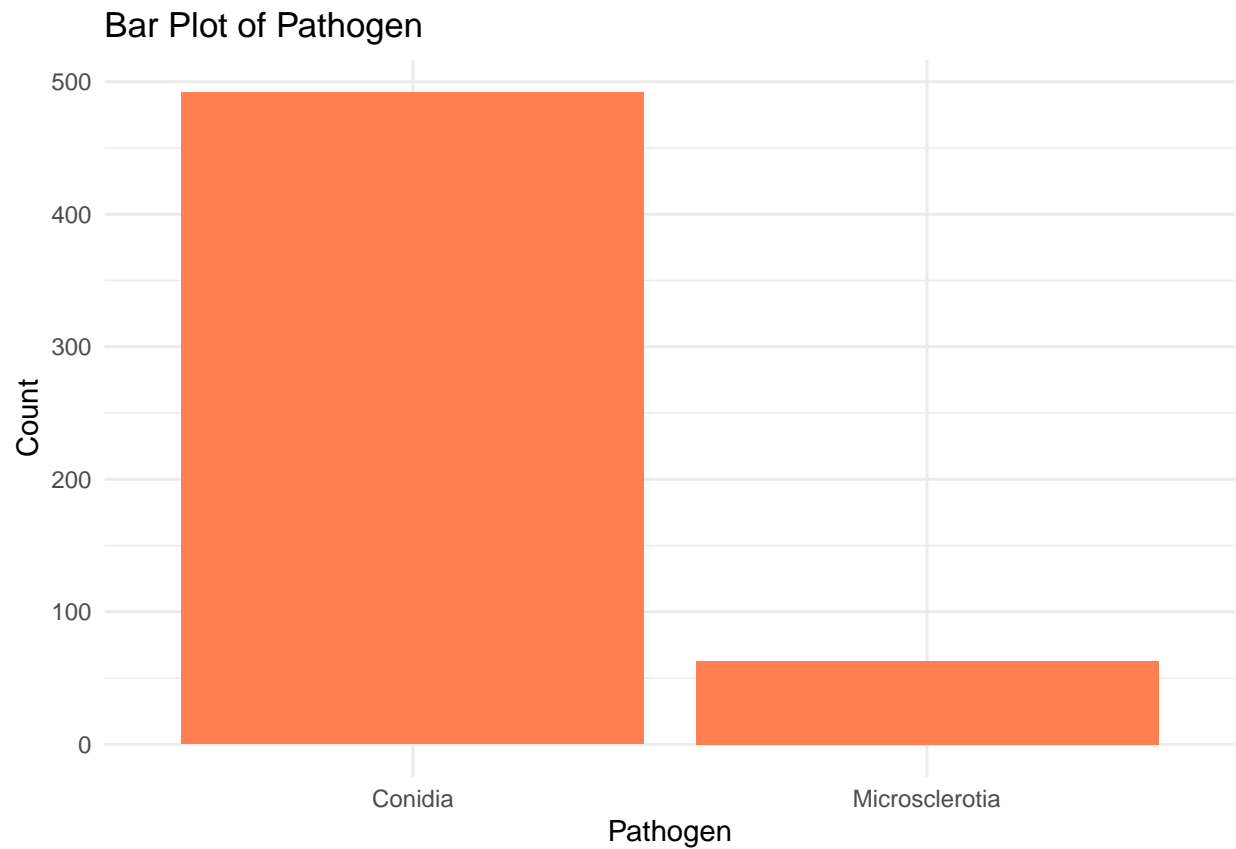
```

```
## 3rd Qu.: 5.00    3rd Qu.:68.16    3rd Qu.:80.63    3rd Qu.:55.96
## Max.      :70.00    Max.      :83.87    Max.      :98.30    Max.      :73.61
##              NA's    :1              NA's    :1
##      T9_RGR      T8_con  T9_con
## Min.      : 0.42    0: 78    0: 71
## 1st Qu.:53.19    1:154    1:110
## Median :62.04    2: 96    2: 98
## Mean      :59.83    3:227    3:276
## 3rd Qu.:68.97
## Max.      :87.11
##
```

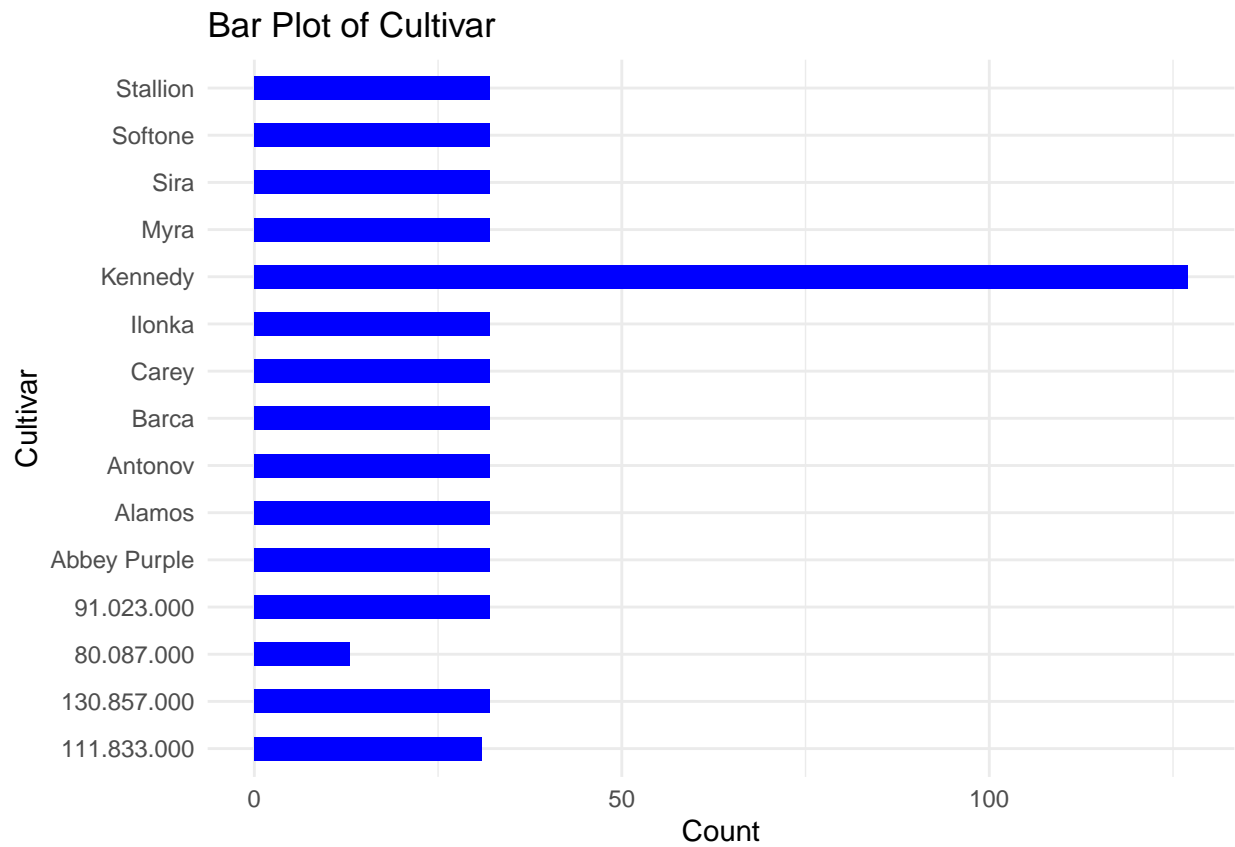
```
# Bar plot for Sensitivity_f
ggplot(new_dataframe) + aes(x = sensitivity_f) +
  geom_bar(fill = "darkmagenta") + coord_flip() +
  theme_minimal() +
  labs(title = "Bar Plot of Sensitivity", x = "Sensitivity", y = "Count")
```



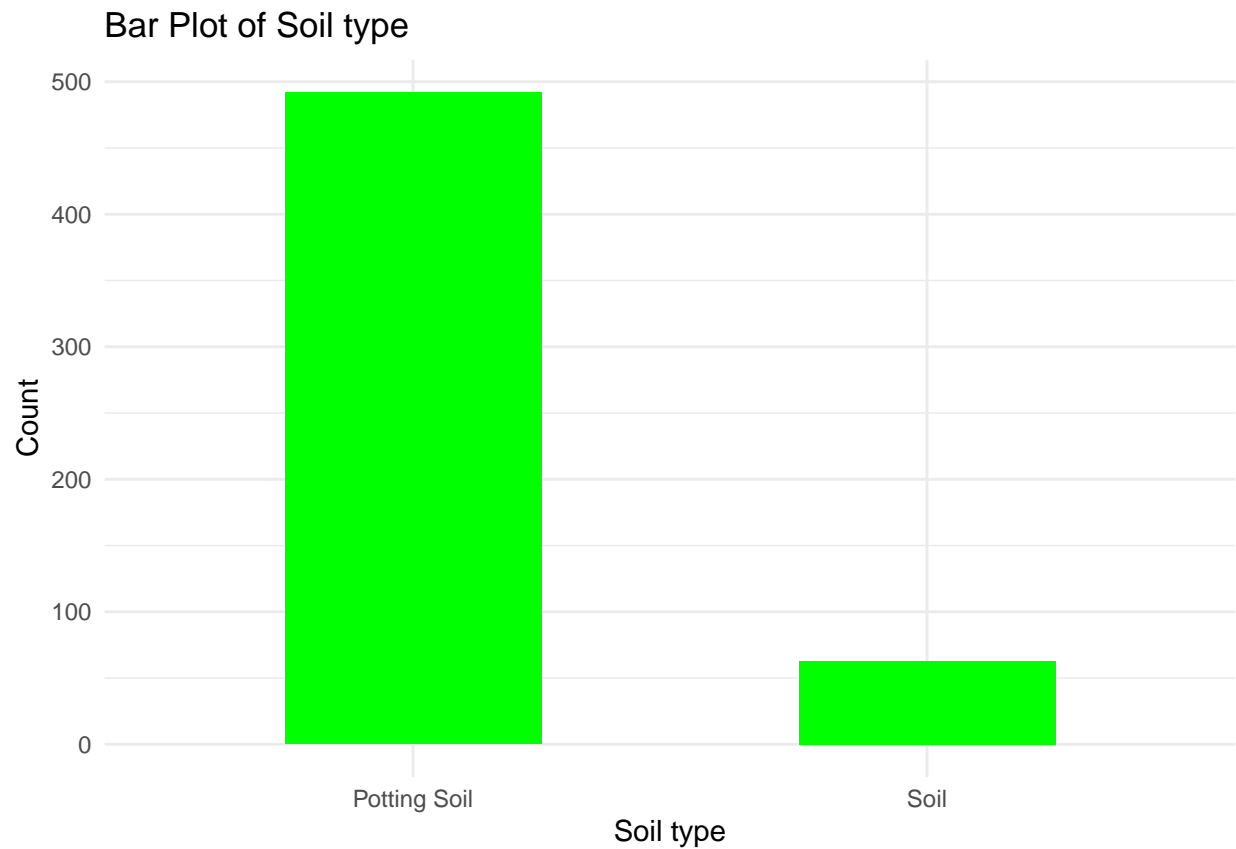
```
# Bar plot for Pathogen
ggplot(new_dataframe, aes(x = Pathogen)) +
  geom_bar(fill = "coral") +
  theme_minimal() +
  labs(title = "Bar Plot of Pathogen", x = "Pathogen", y = "Count")
```



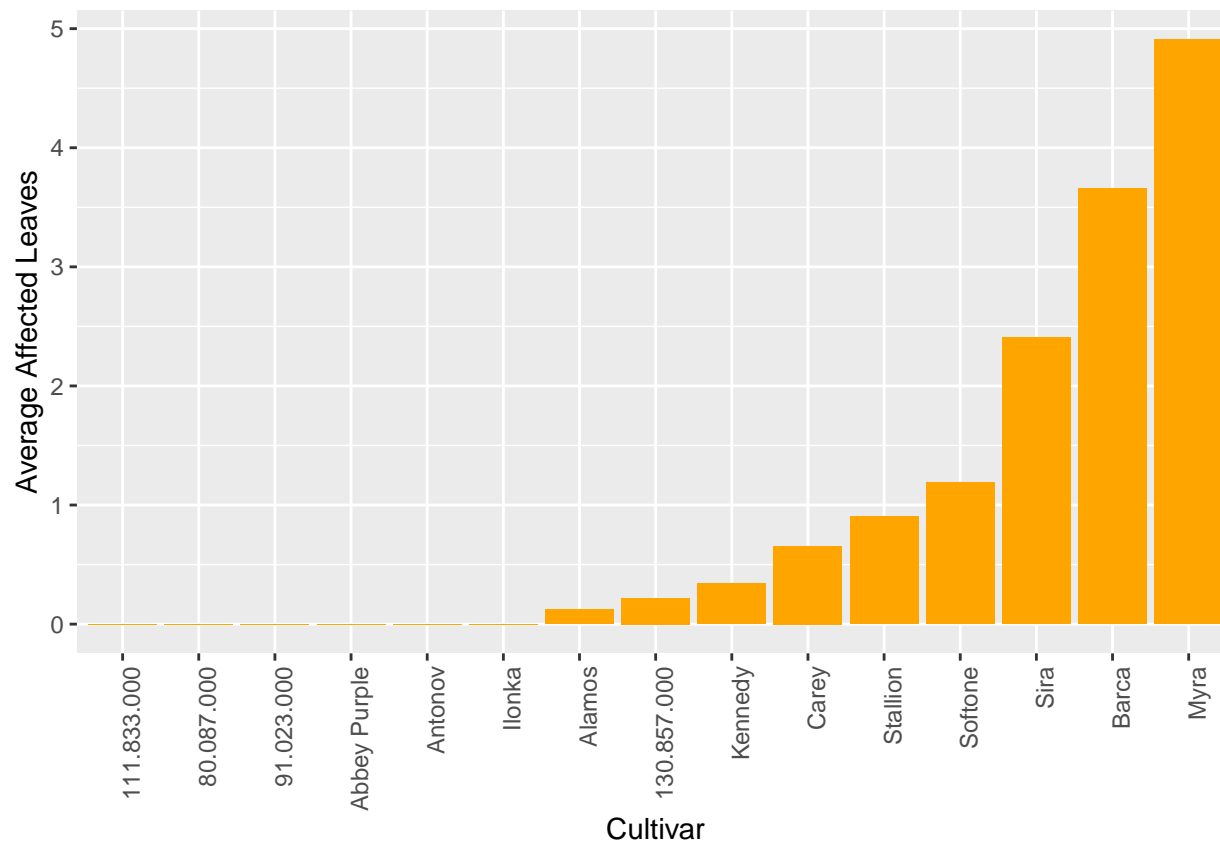
```
# Bar plot for Cultivar  
ggplot(new_dataframe, aes(x = Cultivar)) +  
  geom_bar(fill = "blue", width = 0.5) + coord_flip() +  
  theme_minimal() +  
  labs(title = "Bar Plot of Cultivar", x = "Cultivar", y = "Count")
```



```
# Bar plot for Soil type
ggplot(new_dataframe, aes(x = Soiltype)) +
  geom_bar(fill = "green", width = 0.5) +
  theme_minimal() +
  labs(title = "Bar Plot of Soil type", x = "Soil type", y = "Count")
```



```
average_affected_leaves <- aggregate(T4 ~ Cultivar, data = new_dataframe, mean)
average_affected_leaves_sorted <- average_affected_leaves %>%
  arrange(desc(T4))
ggplot(average_affected_leaves_sorted, aes(x = reorder(Cultivar, T4), y = T4)) +
  geom_bar(stat = "identity", fill = "orange") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(x = "Cultivar", y = "Average Affected Leaves")
```



```
by_cultivar <- split(new_dataframe, new_dataframe$Cultivar )
```

```
# plant's frequency
analysis_by_cultivar_freq <- function(cultivar, with_regression = FALSE) {
  cultivar_data <- by_cultivar[[cultivar]]

  combined_data <- list()

  for (block in unique(cultivar_data$Block)) {
    block_data <- cultivar_data[cultivar_data$Block == block, ]
    block_ag <- block_data %>%
      group_by(Pathogen) %>%
      summarise(
        f3 = mean(T3),
        f5 = mean(T5),
        f6 = mean(T6),
        f7 = mean(T7),
        f8 = mean(T8),
        f9 = mean(T9),
      ) %>%
      pivot_longer(cols = starts_with("f"), names_to = "TimePoint", values_to = "Freq")

    # Add block information
    block_ag$Block <- block
    combined_data[[block]] <- block_ag
  }
}
```



```

long_data <- bind_rows(combined_data)

long_data$TimePoint <- as.numeric(gsub("f", "", long_data$TimePoint))

r <- ggplot(long_data, aes(x = TimePoint, y = Freq, group = interaction(Block, Pathogen), color = Pathogen))
  geom_line() +
  geom_point() +
  facet_wrap(~Block) +
  theme_minimal() +
  labs(title = paste("Affected Plant Frequency over Time by Block and Pathogen for", cultivar),
       x = "Time Point",
       y = "Frequency of Affected Plants",
       color = "Pathogen")
}

```

```

# affected plants rate
analysis_by_cultivar_percentage <- function(cultivar, with_regression = TRUE) {

  cultivar_data <- by_cultivar[[cultivar]]

  combined_data <- list()

  for (block in unique(cultivar_data$Block)) {
    block_data <- cultivar_data[cultivar_data$Block == block, ]
    block_ag <- block_data %>%
      group_by(Pathogen) %>%
      summarise(
        t3 = 0,
        t5 = mean(T5_per),
        t6 = mean(T6_per),
        t7 = mean(T7_per),
        t8 = mean(T8_per),
        t9 = mean(T9_per),
      ) %>%
      pivot_longer(cols = starts_with("t"), names_to = "TimePoint", values_to = "Percent")

    # Add block information
    block_ag$Block <- block
    combined_data[[block]] <- block_ag
  }

  long_data <- bind_rows(combined_data)

  long_data$TimePoint <- as.numeric(gsub("t", "", long_data$TimePoint))

  r <- ggplot(long_data, aes(x = TimePoint, y = Percent, group = interaction(Block, Pathogen), color = Pathogen))
    geom_point() +
    geom_line() +
    facet_wrap(~Block) +
    theme_minimal() +
    labs(title = paste("Percentage of Affected Plant over Time by Block and Pathogen for", cultivar),
         x = "Time Point",
         y = "Percentage of Affected Plants",

```

```

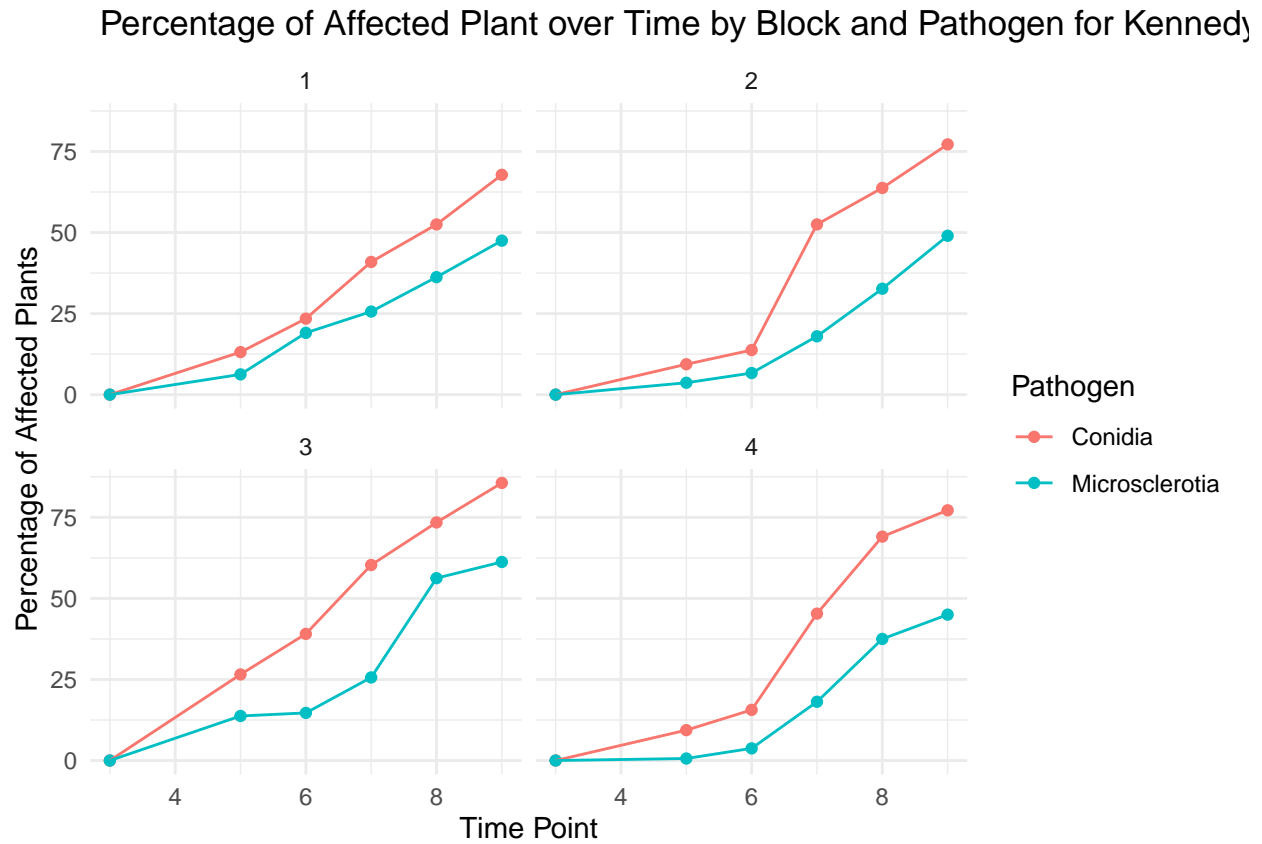
    color = "Pathogen")
}

```

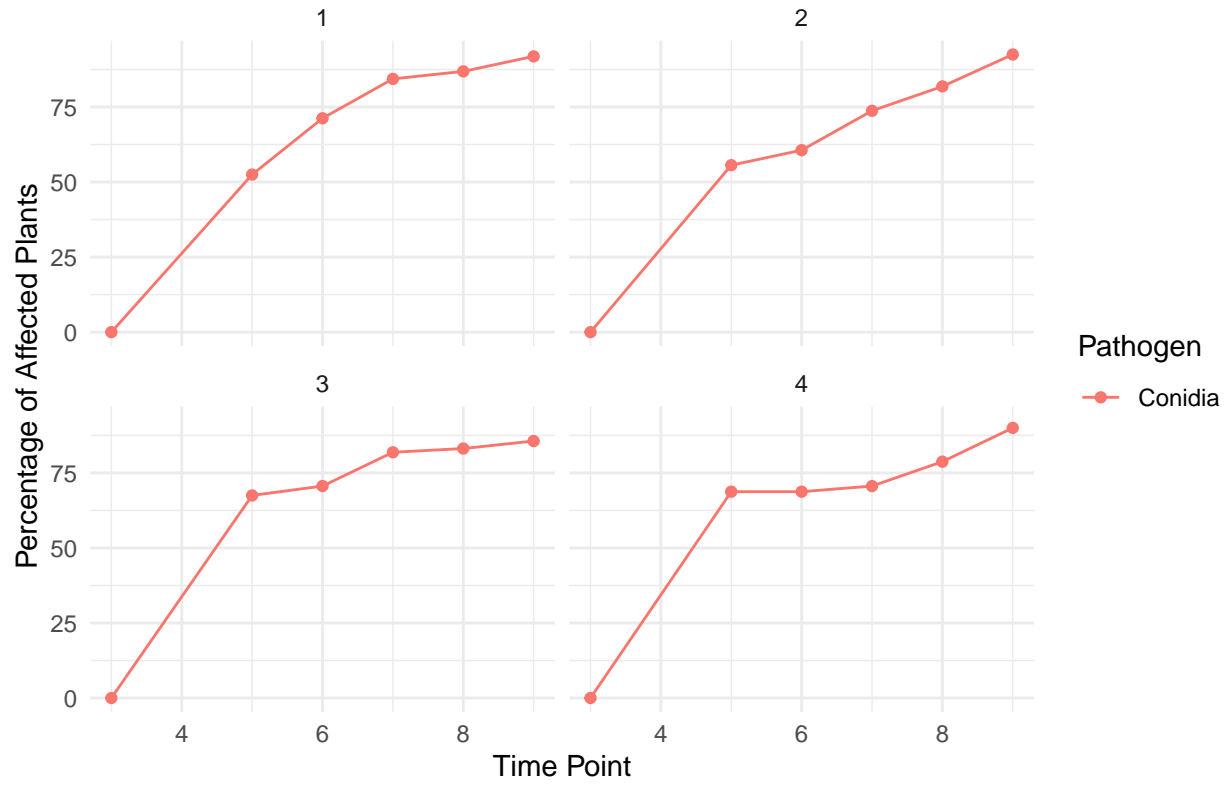
```

for (cultivar in unique(new_dataframe$Cultivar)) {
  print(analysis_by_cultivar_percentage(cultivar, TRUE))
}

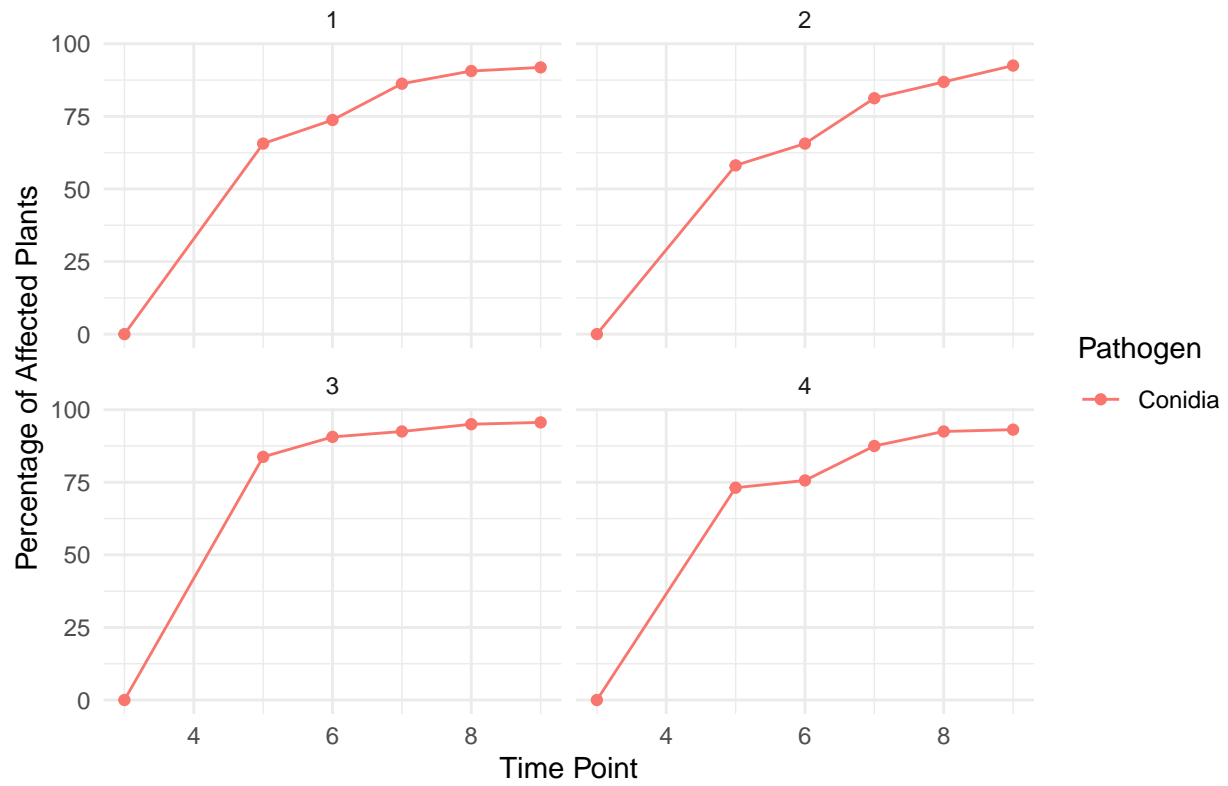
```



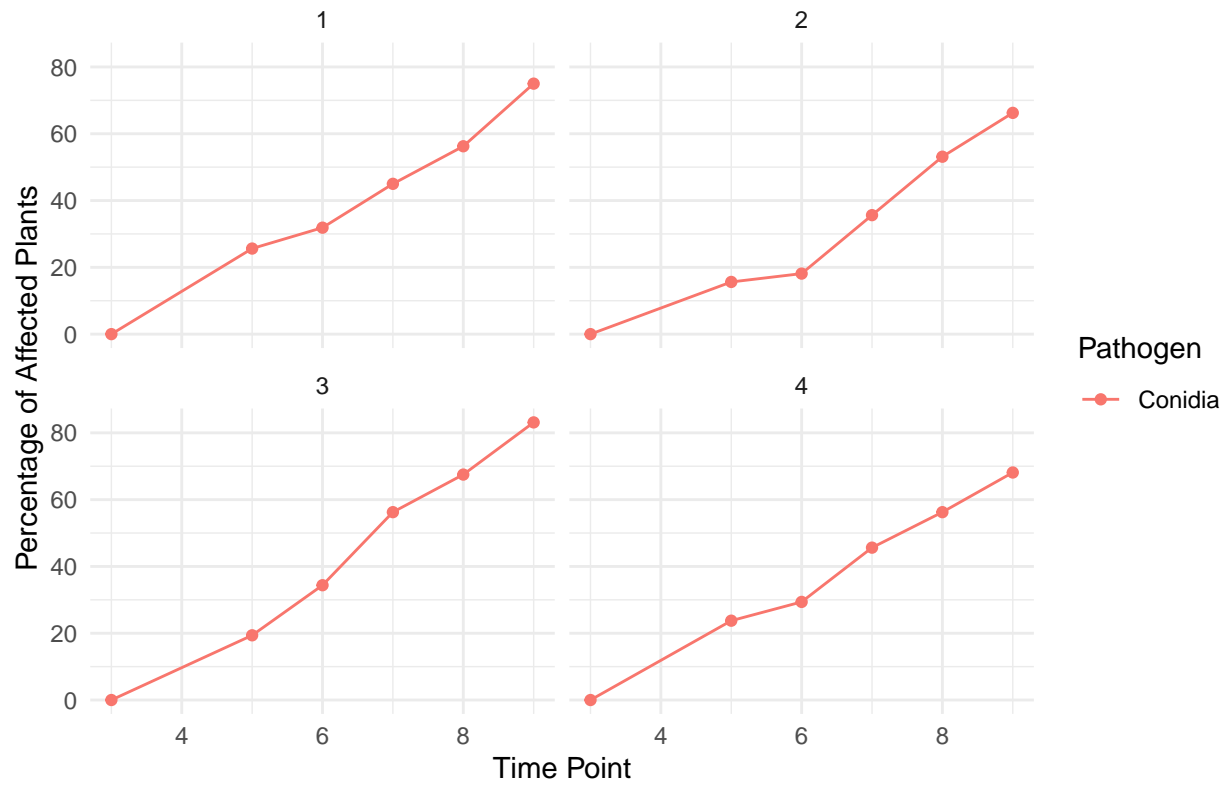
Percentage of Affected Plant over Time by Block and Pathogen for Myra



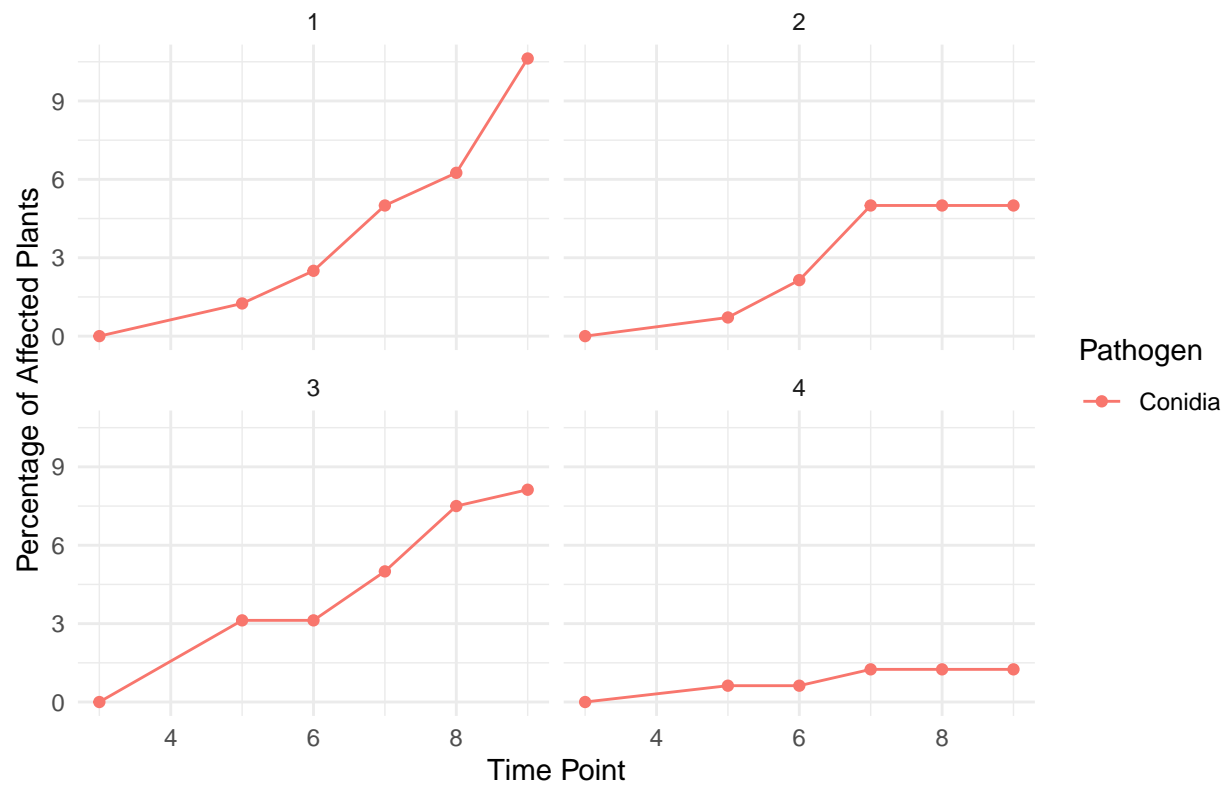
Percentage of Affected Plant over Time by Block and Pathogen for Sira



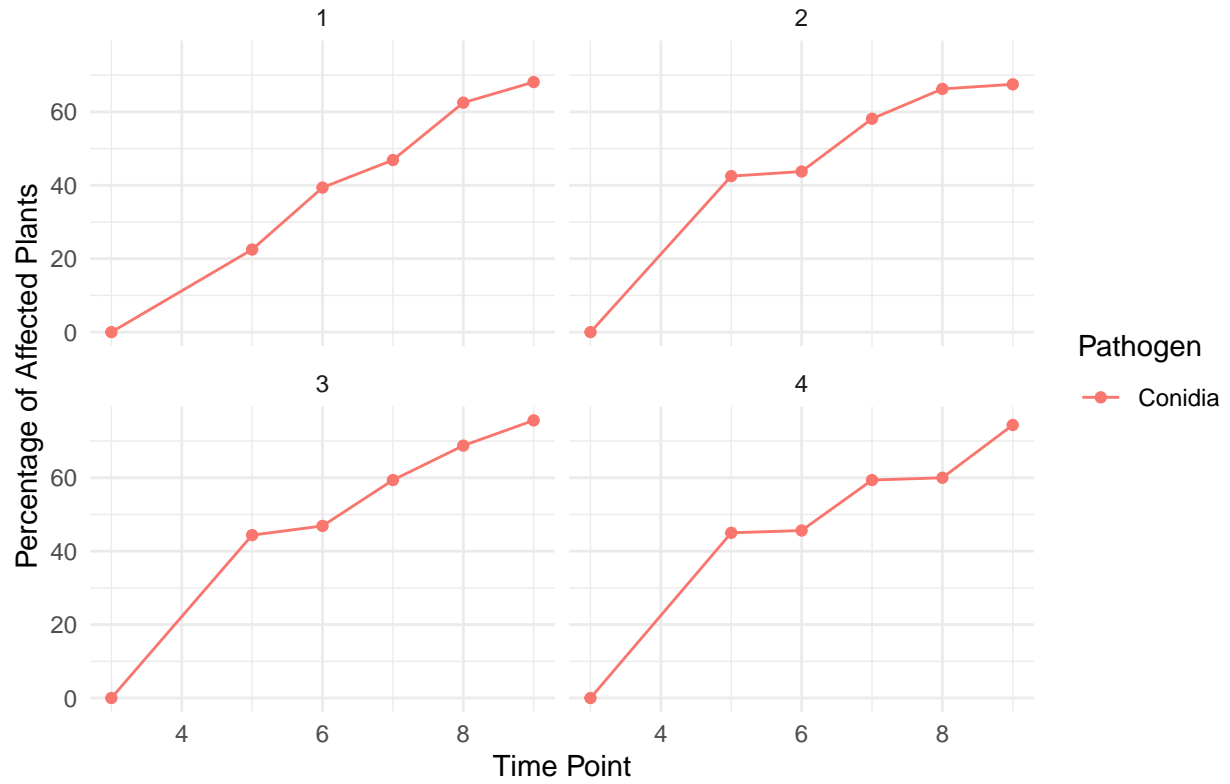
Percentage of Affected Plant over Time by Block and Pathogen for Softone



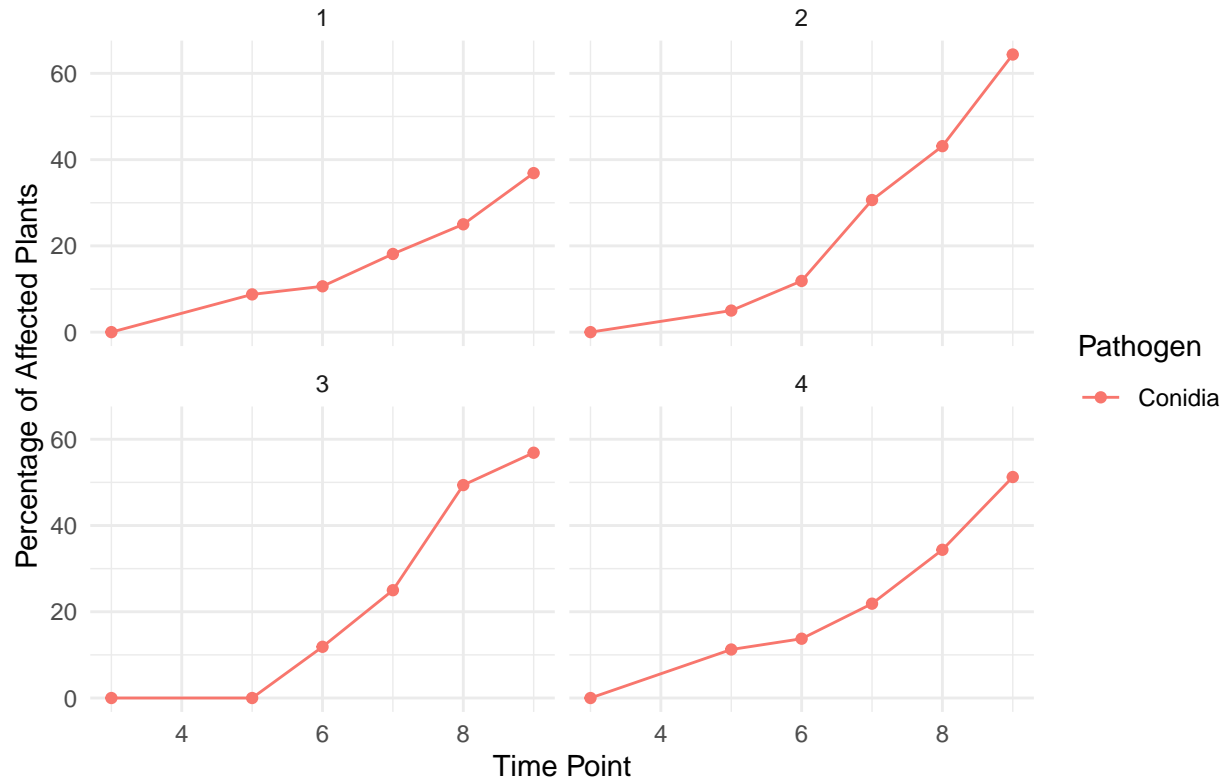
Percentage of Affected Plant over Time by Block and Pathogen for 111.833.i



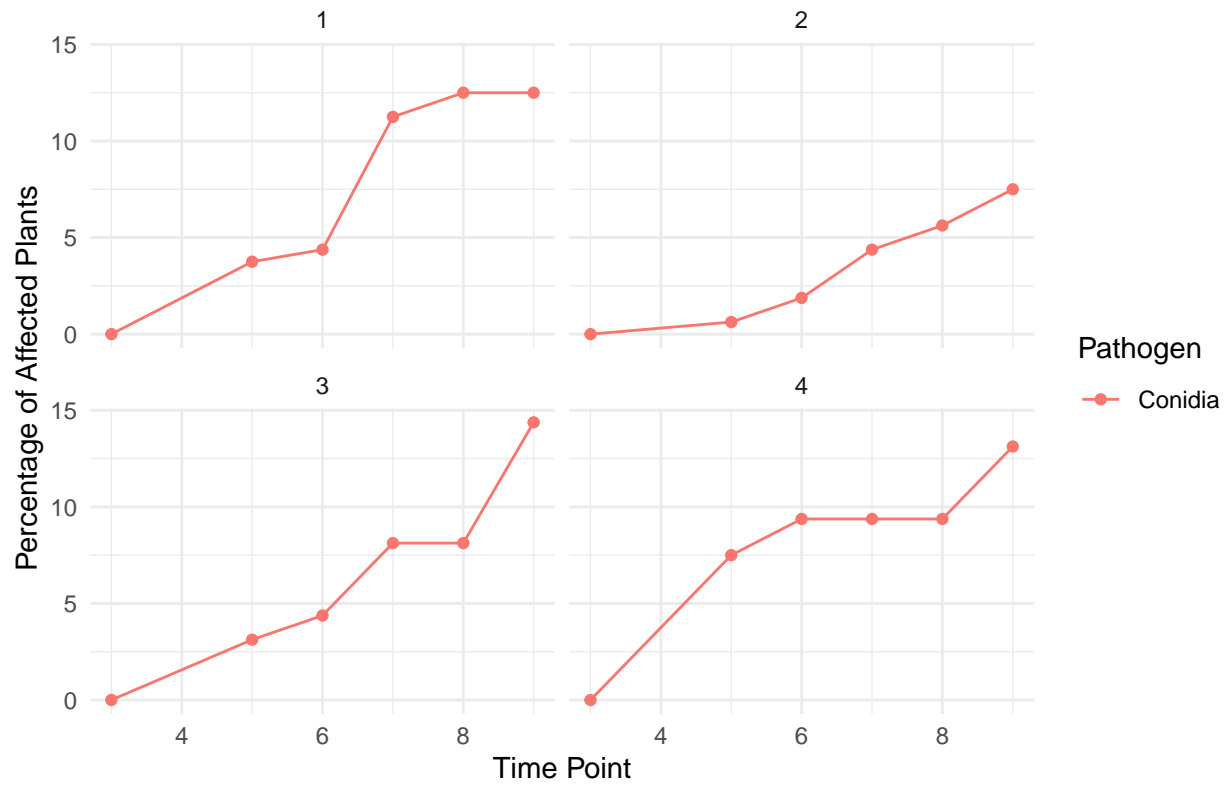
Percentage of Affected Plant over Time by Block and Pathogen for Stallion



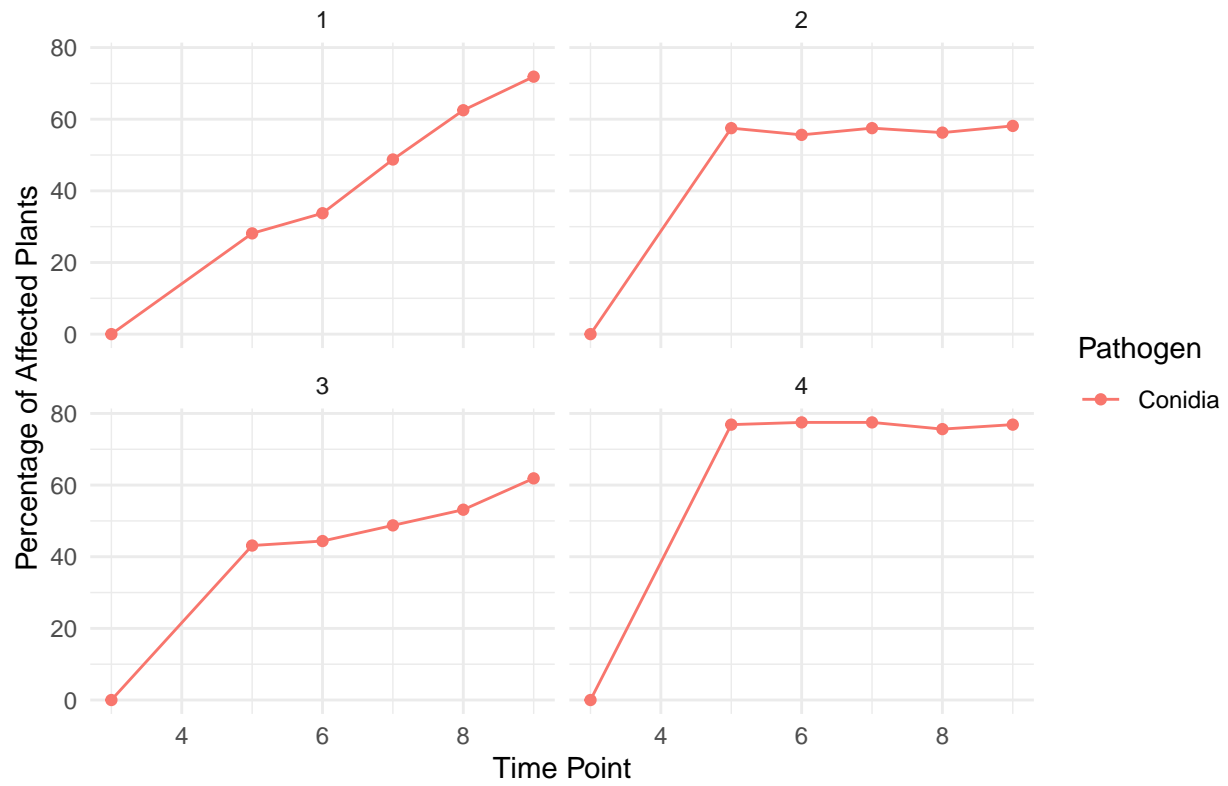
Percentage of Affected Plant over Time by Block and Pathogen for Ilonka



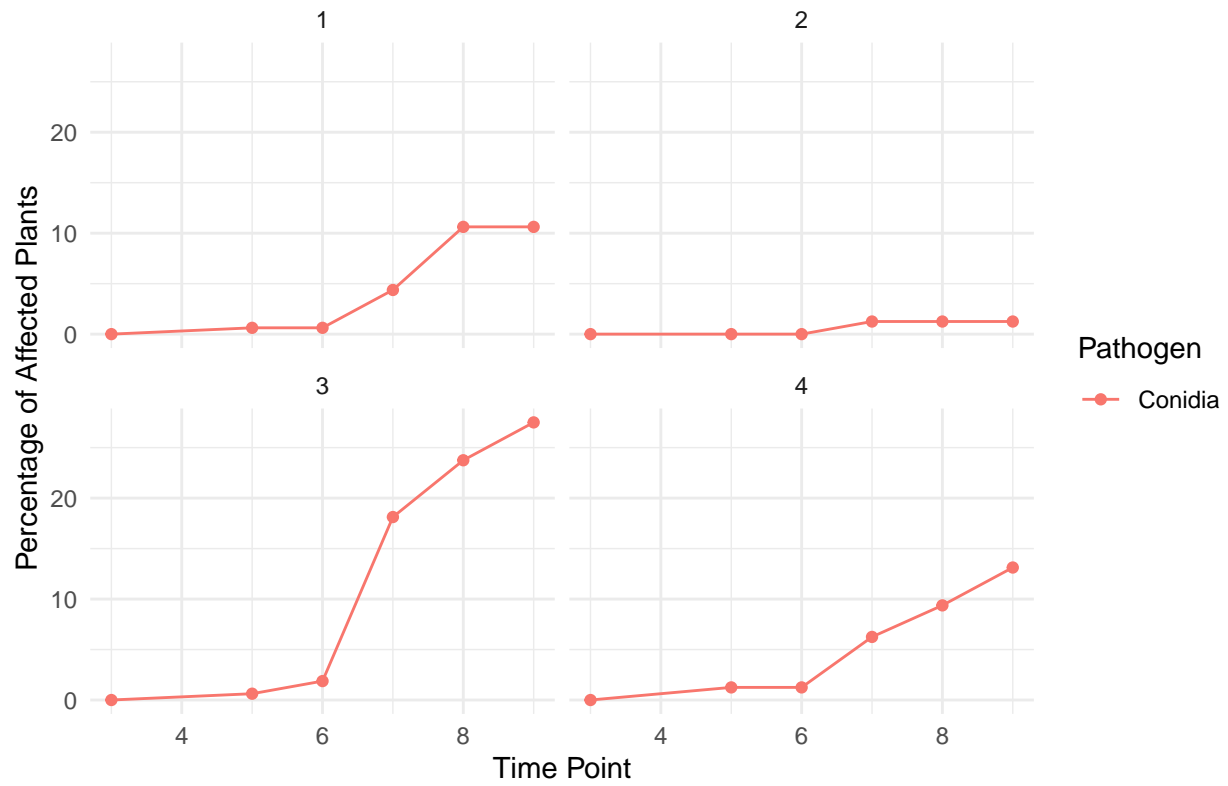
Percentage of Affected Plant over Time by Block and Pathogen for Alamos



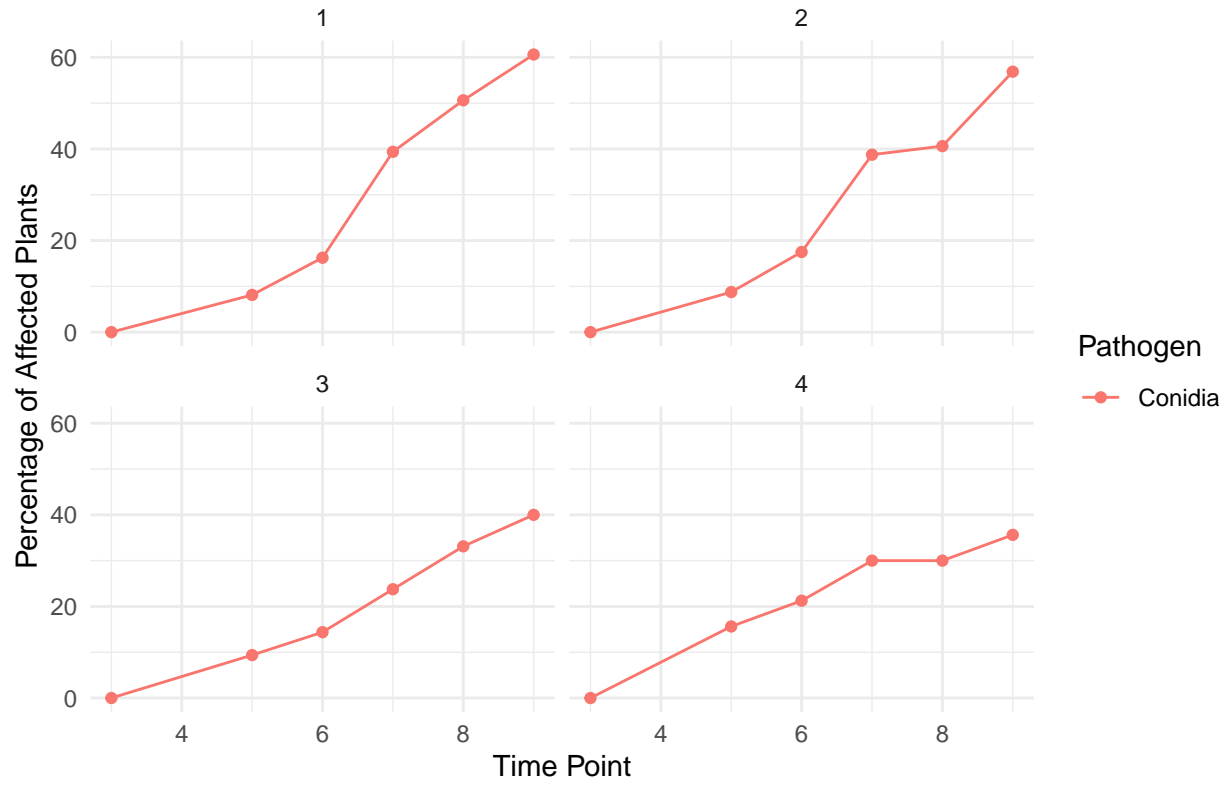
Percentage of Affected Plant over Time by Block and Pathogen for Barca



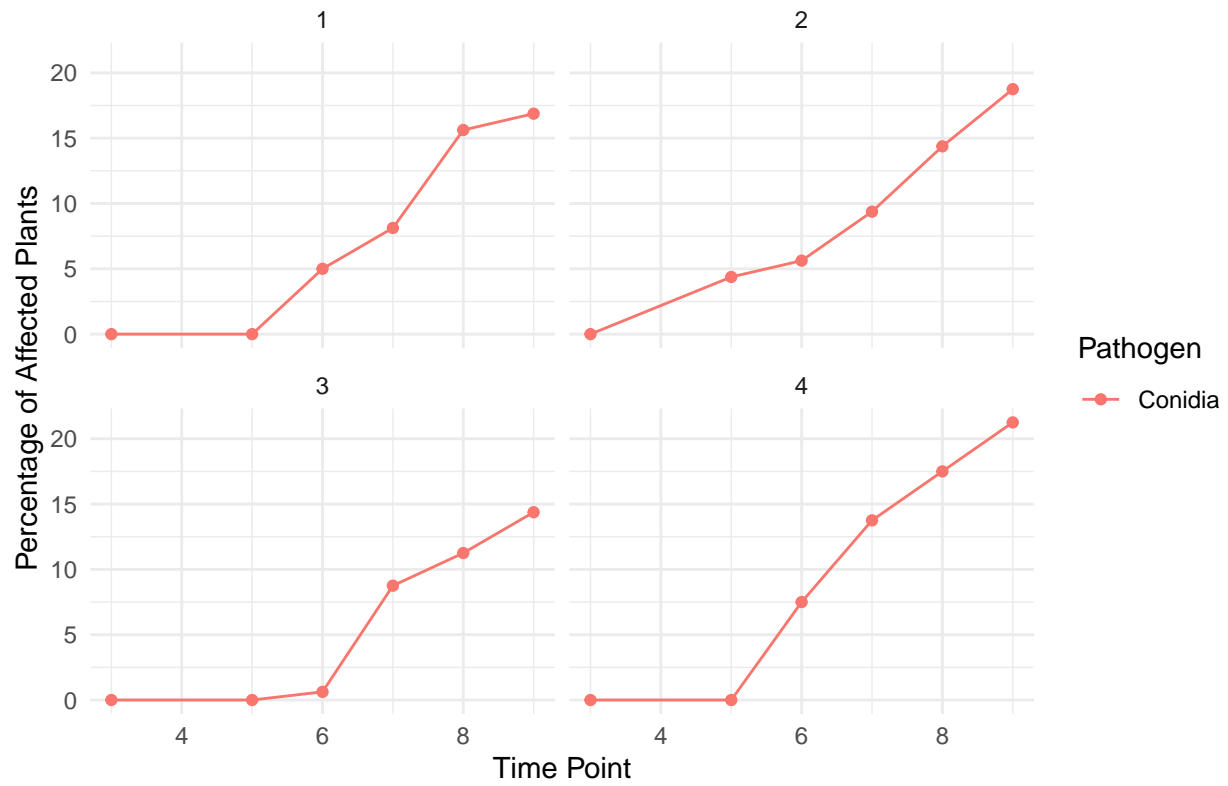
Percentage of Affected Plant over Time by Block and Pathogen for Abbey P

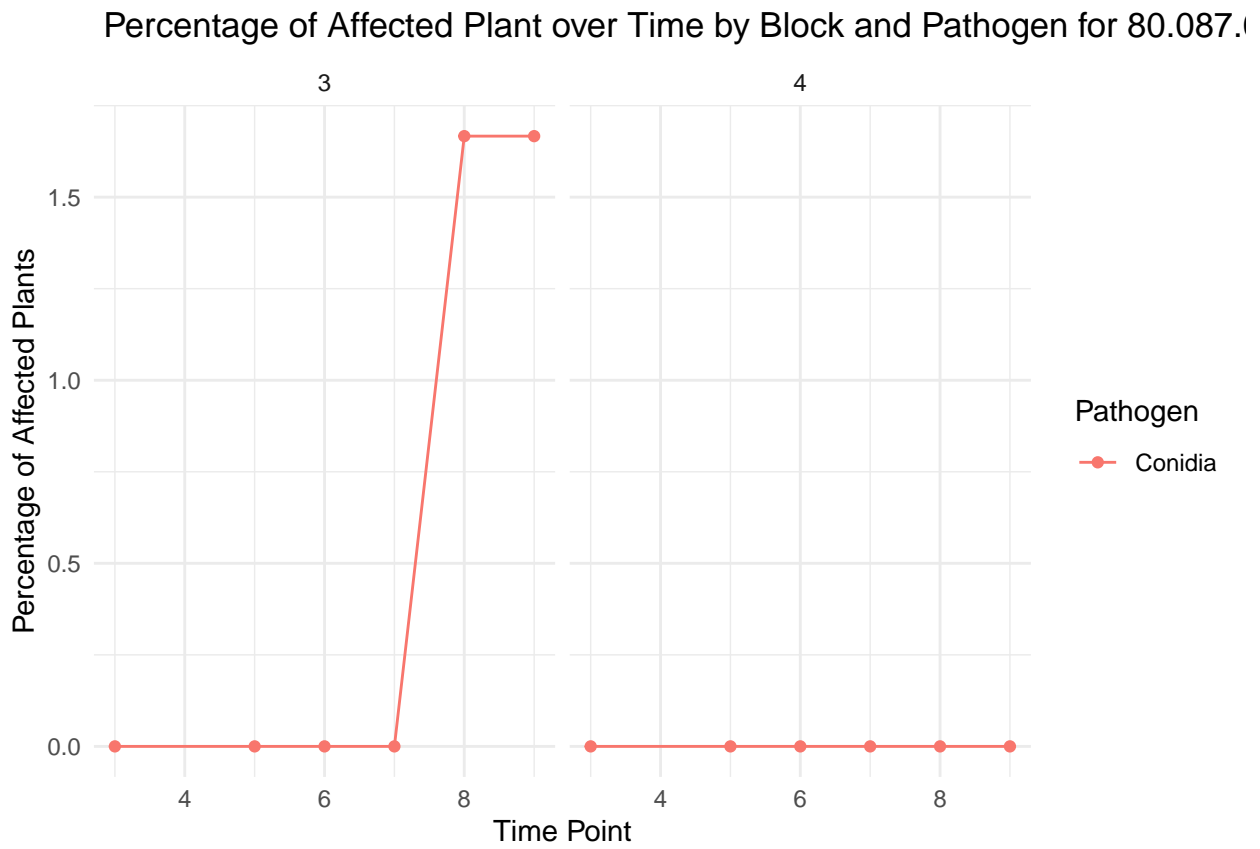


Percentage of Affected Plant over Time by Block and Pathogen for Carey

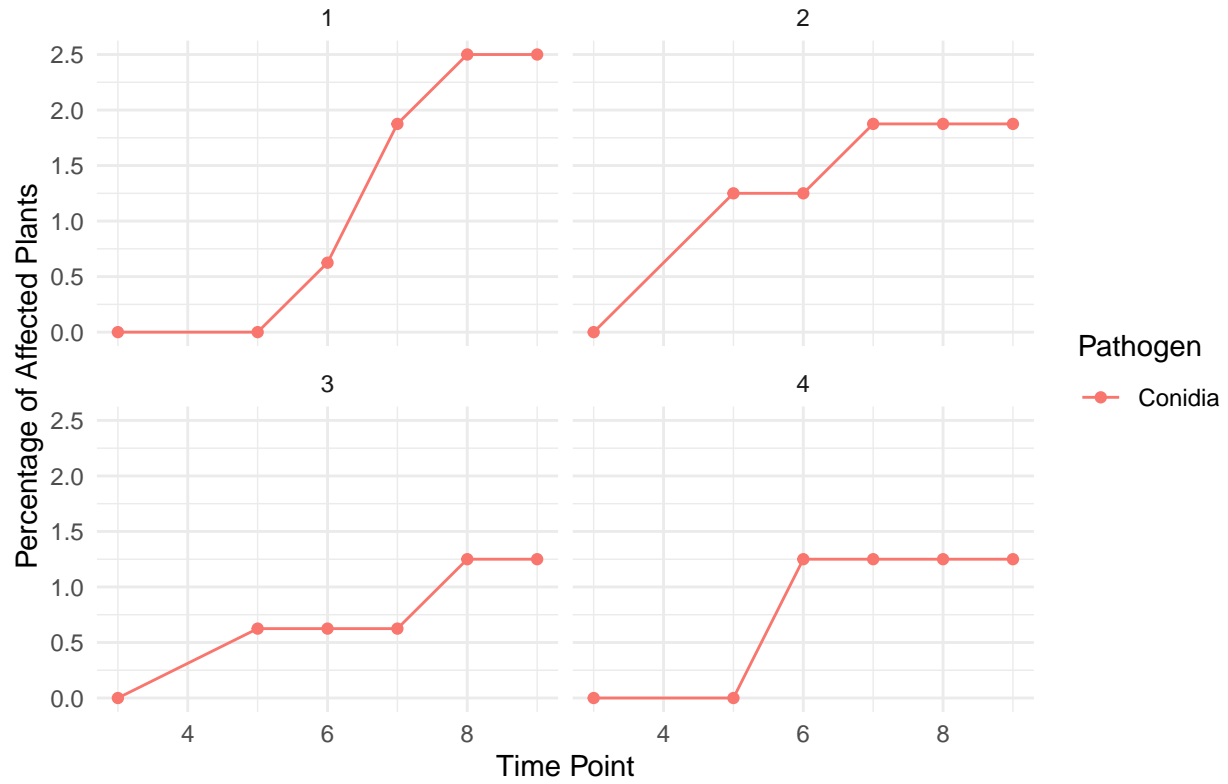


Percentage of Affected Plant over Time by Block and Pathogen for Antonov

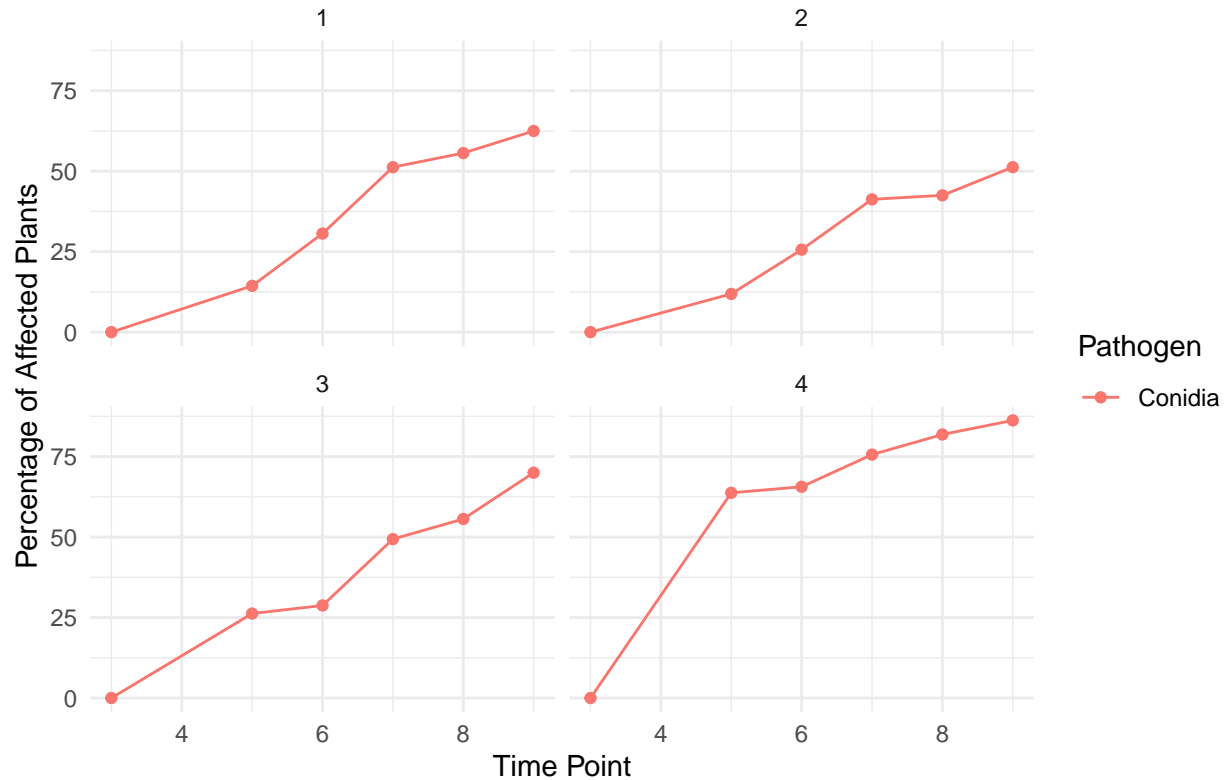




Percentage of Affected Plant over Time by Block and Pathogen for 91.023.



Percentage of Affected Plant over Time by Block and Pathogen for 130.857



```
# AUC
analysis_by_cultivar_percentage <- function(cultivar, filtered=FALSE) {
  cultivar_data <- by_cultivar[[cultivar]]

  combined_data <- list()
  auc_data <- data.frame(Block = integer(), Pathogen = character(), Soiltype = character(), AUC = numeric())

  if (filtered) {
    cultivar_data <- cultivar_data %>% filter(Pathogen != "Microsclerotia", Soiltype != "Soil")
  }

  for (block in unique(cultivar_data$Block)) {
    block_data <- cultivar_data[cultivar_data$Block == block, ]
    block_ag <- block_data %>%
      group_by(Pathogen, Soiltype) %>%
      summarise(
        t3 = 0,
        t5 = mean(T5_per),
        t6 = mean(T6_per),
        t7 = mean(T7_per),
        t8 = mean(T8_per),
        t9 = mean(T9_per),
        .groups = "drop"
      ) %>%
      pivot_longer(cols = starts_with("t"), names_to = "TimePoint", values_to = "Percent")
  }
}
```



```

block_ag$Block <- block
combined_data[[block]] <- block_ag

# Calculate AUC for each pathogen and soil type within the block
for (combo in unique(paste(block_ag$Pathogen, block_ag$Soiltype, sep = "_"))) {
  combo_data <- block_ag[block_ag$Pathogen == strsplit(combo, "_")[[1]][1] & block_ag$Soiltype == strsplit(combo, "_")[[2]][1]]
  time_points <- as.numeric(gsub("t", "", combo_data$TimePoint))
  percent_values <- combo_data$Percent
  auc_value <- trapz(time_points, percent_values)
  auc_data <- rbind(auc_data, data.frame(Block = block, Pathogen = strsplit(combo, "_")[[1]][1], Soiltype = strsplit(combo, "_")[[2]][1], AUC = auc_value))
}

return(auc_data)
}

```

```

auc_results <- list()
for (cultivar in unique(new_dataframe$Cultivar)) {
  auc_results[[cultivar]] <- analysis_by_cultivar_percentage(cultivar)
}

print(auc_results[["Kennedy"]])

```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	229.6875	Kennedy
## 2	1	Conidia	Soil	111.2500	Kennedy
## 3	1	Microsclerotia	Potting Soil	182.8125	Kennedy
## 4	1	Microsclerotia	Soil	45.3125	Kennedy
## 5	2	Conidia	Potting Soil	183.1250	Kennedy
## 6	2	Conidia	Soil	182.1875	Kennedy
## 7	2	Microsclerotia	Potting Soil	135.3125	Kennedy
## 8	2	Microsclerotia	Soil	32.5000	Kennedy
## 9	3	Conidia	Potting Soil	335.3125	Kennedy
## 10	3	Conidia	Soil	175.6250	Kennedy
## 11	3	Microsclerotia	Potting Soil	104.6875	Kennedy
## 12	3	Microsclerotia	Soil	190.9375	Kennedy
## 13	4	Conidia	Potting Soil	198.4375	Kennedy
## 14	4	Conidia	Soil	166.8750	Kennedy
## 15	4	Microsclerotia	Potting Soil	82.5000	Kennedy
## 16	4	Microsclerotia	Soil	83.1250	Kennedy

```
print(auc_results[["Myra"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	367.1875	Myra
## 2	2	Conidia	Potting Soil	345.9375	Myra
## 3	3	Conidia	Potting Soil	379.6875	Myra
## 4	4	Conidia	Potting Soil	366.2500	Myra

Cleaning up

Removing Soiltype="Soil" and Pathogen="Microsclerotia"

```

auc_results <- list()
for (cultivar in unique(new_dataframe$Cultivar)) {
  auc_results[[cultivar]] <- analysis_by_cultivar_percentage(cultivar, TRUE)
}

print(auc_results[["Kennedy"]])

```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 229.6875 Kennedy
## 2      2  Conidia Potting Soil 183.1250 Kennedy
## 3      3  Conidia Potting Soil 335.3125 Kennedy
## 4      4  Conidia Potting Soil 198.4375 Kennedy

```

```
print(auc_results[["Myra"]])
```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 367.1875 Myra
## 2      2  Conidia Potting Soil 345.9375 Myra
## 3      3  Conidia Potting Soil 379.6875 Myra
## 4      4  Conidia Potting Soil 366.2500 Myra

```

```
print(auc_results[["Barca"]])
```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 223.1250 Barca
## 2      2  Conidia Potting Soil 284.6875 Barca
## 3      3  Conidia Potting Soil 241.8750 Barca
## 4      4  Conidia Potting Soil 384.3750 Barca

```

```
print(auc_results[["Sira"]])
```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 395.0000 Sira
## 2      2  Conidia Potting Soil 367.1875 Sira
## 3      3  Conidia Potting Soil 451.5625 Sira
## 4      4  Conidia Potting Soil 411.8750 Sira

```

```
print(auc_results[["Softone"]])
```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 209.0625 Softone
## 2      2  Conidia Potting Soil 163.4375 Softone
## 3      3  Conidia Potting Soil 228.7500 Softone
## 4      4  Conidia Potting Soil 200.9375 Softone

```

```
print(auc_results[["Stallion"]])
```

```

##   Block Pathogen      Soiltype      AUC Cultivar
## 1      1  Conidia Potting Soil 216.5625 Stallion
## 2      2  Conidia Potting Soil 265.6250 Stallion
## 3      3  Conidia Potting Soil 279.3750 Stallion
## 4      4  Conidia Potting Soil 269.6875 Stallion

```

```
print(auc_results[["Carey"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	148.7500	Carey
## 2	2	Conidia	Potting Soil	138.4375	Carey
## 3	3	Conidia	Potting Soil	105.3125	Carey
## 4	4	Conidia	Potting Soil	122.5000	Carey

```
print(auc_results[["130.857.000"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	190.3125	130.857.000
## 2	2	Conidia	Potting Soil	152.8125	130.857.000
## 3	3	Conidia	Potting Soil	208.1250	130.857.000
## 4	4	Conidia	Potting Soil	361.8750	130.857.000

```
print(auc_results[["Alamos"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	40.0000	Alamos
## 2	2	Conidia	Potting Soil	16.5625	Alamos
## 3	3	Conidia	Potting Soil	32.5000	Alamos
## 4	4	Conidia	Potting Soil	45.9375	Alamos

```
print(auc_results[["Ilonka"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	85.3125	Ilonka
## 2	2	Conidia	Potting Soil	125.3125	Ilonka
## 3	3	Conidia	Potting Soil	114.6875	Ilonka
## 4	4	Conidia	Potting Soil	112.5000	Ilonka

```
print(auc_results[["Antonov"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	37.1875	Antonov
## 2	2	Conidia	Potting Soil	45.3125	Antonov
## 3	3	Conidia	Potting Soil	27.8125	Antonov
## 4	4	Conidia	Potting Soil	49.3750	Antonov

```
print(auc_results[["Abbey Purple"]])
```

##	Block	Pathogen	Soiltype	AUC	Cultivar
## 1	1	Conidia	Potting Soil	21.8750	Abbey Purple
## 2	2	Conidia	Potting Soil	3.1250	Abbey Purple
## 3	3	Conidia	Potting Soil	58.4375	Abbey Purple
## 4	4	Conidia	Potting Soil	25.3125	Abbey Purple

```
print(auc_results[["91.023.000"]])
```

```
##   Block Pathogen      Soiltype    AUC    Cultivar
## 1      1  Conidia Potting Soil 6.2500 91.023.000
## 2      2  Conidia Potting Soil 7.8125 91.023.000
## 3      3  Conidia Potting Soil 4.0625 91.023.000
## 4      4  Conidia Potting Soil 4.3750 91.023.000
```

```
print(auc_results[["80.087.000"]])
```

```
##   Block Pathogen      Soiltype    AUC    Cultivar
## 1      3  Conidia Potting Soil 2.5 80.087.000
## 2      4  Conidia Potting Soil 0.0 80.087.000
```

```
print(auc_results[["111.833.000"]])
```

```
##   Block Pathogen      Soiltype    AUC    Cultivar
## 1      1  Conidia Potting Soil 20.93750 111.833.000
## 2      2  Conidia Potting Soil 15.71429 111.833.000
## 3      3  Conidia Potting Soil 24.37500 111.833.000
## 4      4  Conidia Potting Soil  4.68750 111.833.000
```

```
df_anova <- bind_rows(auc_results)
df_anova$Block <- as.factor(df_anova$Block)

# ANOVA test
anova_result <- aov(AUC ~ Cultivar + Block, data = df_anova)
summary(anova_result)
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
## Cultivar   14 992782   70913  48.651 <2e-16 ***
## Block       3  10218    3406   2.337 0.0882 .
## Residuals  40  58303    1458
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# estimated marginal means
emmmeans_result <- emmeans(anova_result, specs = ~ Cultivar)

# cumulative link model
lsd_result <- cld(emmeans_result, adjust = "sidak")

# contrast function
pairwise_comparisons <- contrast(emmeans_result, method = "pairwise", ref = "Kennedy")
summary(pairwise_comparisons)
```

```
##   contrast                estimate    SE df t.ratio p.value
## 111.833.000 - 130.857.000   -211.85 27.0 40  -7.848 <.0001
## 111.833.000 - 80.087.000     28.46 33.5 40   0.851 0.9999
## 111.833.000 - 91.023.000     10.80 27.0 40   0.400 1.0000
```

##	111.833.000 - Abbey Purple	-10.76	27.0	40	-0.399	1.0000
##	111.833.000 - Alamos	-17.32	27.0	40	-0.642	1.0000
##	111.833.000 - Antonov	-23.49	27.0	40	-0.870	0.9999
##	111.833.000 - Barca	-267.09	27.0	40	-9.894	<.0001
##	111.833.000 - Carey	-112.32	27.0	40	-4.161	0.0116
##	111.833.000 - Ilonka	-93.02	27.0	40	-3.446	0.0745
##	111.833.000 - Kennedy	-220.21	27.0	40	-8.157	<.0001
##	111.833.000 - Myra	-348.34	27.0	40	-12.903	<.0001
##	111.833.000 - Sira	-389.98	27.0	40	-14.446	<.0001
##	111.833.000 - Softone	-184.12	27.0	40	-6.820	<.0001
##	111.833.000 - Stallion	-241.38	27.0	40	-8.941	<.0001
##	130.857.000 - 80.087.000	240.31	33.5	40	7.183	<.0001
##	130.857.000 - 91.023.000	222.66	27.0	40	8.248	<.0001
##	130.857.000 - Abbey Purple	201.09	27.0	40	7.449	<.0001
##	130.857.000 - Alamos	194.53	27.0	40	7.206	<.0001
##	130.857.000 - Antonov	188.36	27.0	40	6.977	<.0001
##	130.857.000 - Barca	-55.23	27.0	40	-2.046	0.7599
##	130.857.000 - Carey	99.53	27.0	40	3.687	0.0412
##	130.857.000 - Ilonka	118.83	27.0	40	4.402	0.0059
##	130.857.000 - Kennedy	-8.36	27.0	40	-0.310	1.0000
##	130.857.000 - Myra	-136.48	27.0	40	-5.056	0.0008
##	130.857.000 - Sira	-178.12	27.0	40	-6.598	<.0001
##	130.857.000 - Softone	27.73	27.0	40	1.027	0.9992
##	130.857.000 - Stallion	-29.53	27.0	40	-1.094	0.9984
##	80.087.000 - 91.023.000	-17.65	33.5	40	-0.528	1.0000
##	80.087.000 - Abbey Purple	-39.22	33.5	40	-1.172	0.9967
##	80.087.000 - Alamos	-45.78	33.5	40	-1.368	0.9857
##	80.087.000 - Antonov	-51.95	33.5	40	-1.553	0.9592
##	80.087.000 - Barca	-295.55	33.5	40	-8.834	<.0001
##	80.087.000 - Carey	-140.78	33.5	40	-4.208	0.0102
##	80.087.000 - Ilonka	-121.48	33.5	40	-3.631	0.0474
##	80.087.000 - Kennedy	-248.67	33.5	40	-7.433	<.0001
##	80.087.000 - Myra	-376.80	33.5	40	-11.263	<.0001
##	80.087.000 - Sira	-418.44	33.5	40	-12.508	<.0001
##	80.087.000 - Softone	-212.58	33.5	40	-6.354	<.0001
##	80.087.000 - Stallion	-269.84	33.5	40	-8.066	<.0001
##	91.023.000 - Abbey Purple	-21.56	27.0	40	-0.799	1.0000
##	91.023.000 - Alamos	-28.12	27.0	40	-1.042	0.9990
##	91.023.000 - Antonov	-34.30	27.0	40	-1.270	0.9928
##	91.023.000 - Barca	-277.89	27.0	40	-10.294	<.0001
##	91.023.000 - Carey	-123.12	27.0	40	-4.561	0.0037
##	91.023.000 - Ilonka	-103.83	27.0	40	-3.846	0.0273
##	91.023.000 - Kennedy	-231.02	27.0	40	-8.557	<.0001
##	91.023.000 - Myra	-359.14	27.0	40	-13.303	<.0001
##	91.023.000 - Sira	-400.78	27.0	40	-14.846	<.0001
##	91.023.000 - Softone	-194.92	27.0	40	-7.220	<.0001
##	91.023.000 - Stallion	-252.19	27.0	40	-9.342	<.0001
##	Abbey Purple - Alamos	-6.56	27.0	40	-0.243	1.0000
##	Abbey Purple - Antonov	-12.73	27.0	40	-0.472	1.0000
##	Abbey Purple - Barca	-256.33	27.0	40	-9.495	<.0001
##	Abbey Purple - Carey	-101.56	27.0	40	-3.762	0.0340
##	Abbey Purple - Ilonka	-82.27	27.0	40	-3.047	0.1796
##	Abbey Purple - Kennedy	-209.45	27.0	40	-7.759	<.0001
##	Abbey Purple - Myra	-337.58	27.0	40	-12.505	<.0001

## Abbey Purple - Sira	-379.22	27.0	40	-14.047	<.0001
## Abbey Purple - Softone	-173.36	27.0	40	-6.422	<.0001
## Abbey Purple - Stallion	-230.62	27.0	40	-8.543	<.0001
## Alamos - Antonov	-6.17	27.0	40	-0.229	1.0000
## Alamos - Barca	-249.77	27.0	40	-9.252	<.0001
## Alamos - Carey	-95.00	27.0	40	-3.519	0.0625
## Alamos - Ilonka	-75.70	27.0	40	-2.804	0.2853
## Alamos - Kennedy	-202.89	27.0	40	-7.516	<.0001
## Alamos - Myra	-331.02	27.0	40	-12.262	<.0001
## Alamos - Sira	-372.66	27.0	40	-13.804	<.0001
## Alamos - Softone	-166.80	27.0	40	-6.179	<.0001
## Alamos - Stallion	-224.06	27.0	40	-8.300	<.0001
## Antonov - Barca	-243.59	27.0	40	-9.023	<.0001
## Antonov - Carey	-88.83	27.0	40	-3.290	0.1068
## Antonov - Ilonka	-69.53	27.0	40	-2.576	0.4148
## Antonov - Kennedy	-196.72	27.0	40	-7.287	<.0001
## Antonov - Myra	-324.84	27.0	40	-12.033	<.0001
## Antonov - Sira	-366.48	27.0	40	-13.575	<.0001
## Antonov - Softone	-160.62	27.0	40	-5.950	0.0001
## Antonov - Stallion	-217.89	27.0	40	-8.071	<.0001
## Barca - Carey	154.77	27.0	40	5.733	0.0001
## Barca - Ilonka	174.06	27.0	40	6.448	<.0001
## Barca - Kennedy	46.88	27.0	40	1.736	0.9086
## Barca - Myra	-81.25	27.0	40	-3.010	0.1937
## Barca - Sira	-122.89	27.0	40	-4.552	0.0038
## Barca - Softone	82.97	27.0	40	3.073	0.1703
## Barca - Stallion	25.70	27.0	40	0.952	0.9996
## Carey - Ilonka	19.30	27.0	40	0.715	1.0000
## Carey - Kennedy	-107.89	27.0	40	-3.997	0.0183
## Carey - Myra	-236.02	27.0	40	-8.743	<.0001
## Carey - Sira	-277.66	27.0	40	-10.285	<.0001
## Carey - Softone	-71.80	27.0	40	-2.660	0.3642
## Carey - Stallion	-129.06	27.0	40	-4.781	0.0019
## Ilonka - Kennedy	-127.19	27.0	40	-4.711	0.0024
## Ilonka - Myra	-255.31	27.0	40	-9.457	<.0001
## Ilonka - Sira	-296.95	27.0	40	-11.000	<.0001
## Ilonka - Softone	-91.09	27.0	40	-3.374	0.0881
## Ilonka - Stallion	-148.36	27.0	40	-5.496	0.0002
## Kennedy - Myra	-128.12	27.0	40	-4.746	0.0021
## Kennedy - Sira	-169.77	27.0	40	-6.289	<.0001
## Kennedy - Softone	36.09	27.0	40	1.337	0.9884
## Kennedy - Stallion	-21.17	27.0	40	-0.784	1.0000
## Myra - Sira	-41.64	27.0	40	-1.542	0.9613
## Myra - Softone	164.22	27.0	40	6.083	<.0001
## Myra - Stallion	106.95	27.0	40	3.962	0.0201
## Sira - Softone	205.86	27.0	40	7.626	<.0001
## Sira - Stallion	148.59	27.0	40	5.504	0.0002
## Softone - Stallion	-57.27	27.0	40	-2.121	0.7138
##					
## Results are averaged over the levels of: Block					
## P value adjustment: tukey method for comparing a family of 15 estimates					

Pairwise T-test

```
pairwise.t.test(df_anova$AUC, df_anova$Cultivar, p.adjust.method = "bonferroni")
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: df_anova$AUC and df_anova$Cultivar
##
##      111.833.000 130.857.000 80.087.000 91.023.000 Abbey Purple Alamos
## 130.857.000 2.5e-07 - - - -
## 80.087.000 1.00000 5.7e-06 - - -
## 91.023.000 1.00000 7.2e-08 1.00000 - -
## Abbey Purple 1.00000 8.9e-07 1.00000 1.00000 - -
## Alamos 1.00000 1.9e-06 1.00000 1.00000 1.00000 -
## Antonov 1.00000 4.0e-06 1.00000 1.00000 1.00000 1.00000
## Barca 4.7e-10 1.00000 2.9e-08 1.5e-10 1.6e-09 3.2e-09
## Carey 0.02743 0.10671 0.06616 0.00832 0.08639 0.16977
## Ilonka 0.20722 0.01343 0.32956 0.06811 0.59180 1.00000
## Kennedy 9.5e-08 1.00000 2.6e-06 2.7e-08 3.3e-07 7.2e-07
## Myra 1.1e-13 0.00182 1.9e-11 3.8e-14 3.0e-13 5.8e-13
## Sira 2.1e-15 1.4e-05 5.9e-13 8.2e-16 5.7e-15 1.1e-14
## Softone 6.7e-06 1.00000 8.4e-05 1.9e-06 2.4e-05 5.2e-05
## Stallion 8.4e-09 1.00000 3.3e-07 2.5e-09 2.9e-08 6.1e-08
##
##      Antonov Barca Carey Ilonka Kennedy Myra Sira Softone
## 130.857.000 - - - - - - -
## 80.087.000 - - - - - - -
## 91.023.000 - - - - - - -
## Abbey Purple - - - - - - -
## Alamos - - - - - - -
## Antonov - - - - - - -
## Barca 6.5e-09 - - - - - -
## Carey 0.31438 0.00022 - - - - -
## Ilonka 1.00000 2.2e-05 1.00000 - - - -
## Kennedy 1.5e-06 1.00000 0.04424 0.00526 - - -
## Myra 1.1e-12 0.65121 1.5e-08 1.7e-09 0.00473 - -
## Sira 1.9e-14 0.00854 1.5e-10 1.9e-11 3.7e-05 1.00000 -
## Softone 0.00011 0.55367 1.00000 0.25132 1.00000 7.1e-05 5.1e-07 -
## Stallion 1.2e-07 1.00000 0.00425 0.00046 1.00000 0.04890 0.00045 1.00000
##
## P value adjustment method: bonferroni
```

```
# Simulate different number of blocks
```

```
redistribute_blocks <- function(df, num_new_blocks) {
  if (num_new_blocks <= 0) {
    stop("Number of blocks must be a positive integer.")
  }

  set.seed(123)
  shuffled_df <- df[sample(nrow(df)), ]
```

```

# Assign new block numbers
shuffled_df$Block <- as.factor(rep(1:num_new_blocks, length.out = nrow(shuffled_df)))

return(shuffled_df)
}

```

```

# Anova test with new number of blocks
ddf_anova <- redistribute_blocks(df_anova, 11)

anova_result <- aov(AUC ~ Cultivar + Block, data = ddf_anova)
summary(anova_result)

```

```

##              Df Sum Sq Mean Sq F value Pr(>F)
## Cultivar      14 992782   70913  54.966 <2e-16 ***
## Block         10  25948    2595   2.011 0.0644 .
## Residuals     33  42574    1290
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

# estimated marginal means
emmeans_result <- emmeans(anova_result, specs = ~ Cultivar)

# cumulative link model
lsd_result <- cld(emmeans_result, adjust = "sidak")

# contrast function
pairwise_comparisons <- contrast(emmeans_result, method = "pairwise", ref = "Kennedy")
summary(pairwise_comparisons)

```

```

## contrast              estimate    SE df t.ratio p.value
## 111.833.000 - 130.857.000   -237.63 32.3 33  -7.363  <.0001
## 111.833.000 - 80.087.000    -11.52 37.1 33  -0.311  1.0000
## 111.833.000 - 91.023.000     13.47 31.4 33   0.429  1.0000
## 111.833.000 - Abbey Purple   -28.48 31.2 33  -0.912  0.9997
## 111.833.000 - Alamos         -6.61 31.4 33  -0.210  1.0000
## 111.833.000 - Antonov       -49.28 29.7 33  -1.658  0.9313
## 111.833.000 - Barca        -287.85 27.3 33 -10.542  <.0001
## 111.833.000 - Carey        -130.46 27.8 33  -4.692  0.0034
## 111.833.000 - Ilonka       -76.46 28.3 33  -2.706  0.3442
## 111.833.000 - Kennedy      -227.07 29.0 33  -7.830  <.0001
## 111.833.000 - Myra        -375.27 30.3 33 -12.372  <.0001
## 111.833.000 - Sira        -402.19 29.4 33 -13.691  <.0001
## 111.833.000 - Softone     -177.35 26.1 33  -6.783  <.0001
## 111.833.000 - Stallion    -261.58 30.3 33  -8.624  <.0001
## 130.857.000 - 80.087.000    226.10 37.2 33   6.075  0.0001
## 130.857.000 - 91.023.000    251.09 28.8 33   8.728  <.0001
## 130.857.000 - Abbey Purple   209.14 26.2 33   7.983  <.0001
## 130.857.000 - Alamos       231.02 30.5 33   7.575  <.0001
## 130.857.000 - Antonov     188.35 27.1 33   6.944  <.0001
## 130.857.000 - Barca       -50.23 29.6 33  -1.696  0.9198
## 130.857.000 - Carey       107.17 30.4 33   3.522  0.0686
## 130.857.000 - Ilonka      161.17 29.9 33   5.398  0.0005

```


##	130.857.000 - Kennedy	10.55	30.8	33	0.343	1.0000
##	130.857.000 - Myra	-137.65	28.7	33	-4.795	0.0026
##	130.857.000 - Sira	-164.57	29.1	33	-5.658	0.0002
##	130.857.000 - Softone	60.27	32.2	33	1.869	0.8515
##	130.857.000 - Stallion	-23.96	31.6	33	-0.758	1.0000
##	80.087.000 - 91.023.000	24.99	34.9	33	0.715	1.0000
##	80.087.000 - Abbey Purple	-16.96	36.6	33	-0.464	1.0000
##	80.087.000 - Alamos	4.91	34.8	33	0.141	1.0000
##	80.087.000 - Antonov	-37.76	36.4	33	-1.038	0.9990
##	80.087.000 - Barca	-276.33	34.5	33	-8.012	<.0001
##	80.087.000 - Carey	-118.94	36.0	33	-3.308	0.1103
##	80.087.000 - Ilonka	-64.93	35.8	33	-1.814	0.8757
##	80.087.000 - Kennedy	-215.55	34.2	33	-6.302	<.0001
##	80.087.000 - Myra	-363.75	32.7	33	-11.139	<.0001
##	80.087.000 - Sira	-390.67	36.1	33	-10.813	<.0001
##	80.087.000 - Softone	-165.83	37.0	33	-4.483	0.0061
##	80.087.000 - Stallion	-250.06	31.8	33	-7.860	<.0001
##	91.023.000 - Abbey Purple	-41.95	27.2	33	-1.543	0.9595
##	91.023.000 - Alamos	-20.07	28.6	33	-0.701	1.0000
##	91.023.000 - Antonov	-62.75	28.6	33	-2.198	0.6643
##	91.023.000 - Barca	-301.32	29.5	33	-10.208	<.0001
##	91.023.000 - Carey	-143.93	28.9	33	-4.977	0.0016
##	91.023.000 - Ilonka	-89.92	28.2	33	-3.192	0.1408
##	91.023.000 - Kennedy	-240.54	28.8	33	-8.354	<.0001
##	91.023.000 - Myra	-388.74	28.5	33	-13.645	<.0001
##	91.023.000 - Sira	-415.66	28.5	33	-14.582	<.0001
##	91.023.000 - Softone	-190.82	31.3	33	-6.101	0.0001
##	91.023.000 - Stallion	-275.05	29.0	33	-9.474	<.0001
##	Abbey Purple - Alamos	21.87	29.6	33	0.740	1.0000
##	Abbey Purple - Antonov	-20.80	26.3	33	-0.792	1.0000
##	Abbey Purple - Barca	-259.37	28.7	33	-9.049	<.0001
##	Abbey Purple - Carey	-101.98	28.7	33	-3.551	0.0641
##	Abbey Purple - Ilonka	-47.98	28.9	33	-1.663	0.9299
##	Abbey Purple - Kennedy	-198.59	29.7	33	-6.683	<.0001
##	Abbey Purple - Myra	-346.79	28.6	33	-12.134	<.0001
##	Abbey Purple - Sira	-373.71	27.4	33	-13.650	<.0001
##	Abbey Purple - Softone	-148.87	31.3	33	-4.761	0.0028
##	Abbey Purple - Stallion	-233.10	30.7	33	-7.581	<.0001
##	Alamos - Antonov	-42.67	30.3	33	-1.408	0.9806
##	Alamos - Barca	-281.24	29.0	33	-9.698	<.0001
##	Alamos - Carey	-123.85	30.0	33	-4.131	0.0154
##	Alamos - Ilonka	-69.85	27.3	33	-2.562	0.4283
##	Alamos - Kennedy	-220.47	27.8	33	-7.927	<.0001
##	Alamos - Myra	-368.66	28.8	33	-12.801	<.0001
##	Alamos - Sira	-395.59	28.1	33	-14.060	<.0001
##	Alamos - Softone	-170.74	31.3	33	-5.448	0.0004
##	Alamos - Stallion	-254.97	29.8	33	-8.569	<.0001
##	Antonov - Barca	-238.57	28.2	33	-8.474	<.0001
##	Antonov - Carey	-81.18	27.5	33	-2.954	0.2245
##	Antonov - Ilonka	-27.18	29.2	33	-0.931	0.9997
##	Antonov - Kennedy	-177.79	29.4	33	-6.048	0.0001
##	Antonov - Myra	-325.99	28.4	33	-11.487	<.0001
##	Antonov - Sira	-352.92	27.3	33	-12.939	<.0001
##	Antonov - Softone	-128.07	29.8	33	-4.297	0.0100

```
## Antonov - Stallion      -212.30 29.7 33  -7.140 <.0001
## Barca - Carey           157.39 27.7 33   5.690 0.0002
## Barca - Ilonka          211.40 27.8 33   7.592 <.0001
## Barca - Kennedy         60.78 27.8 33   2.190 0.6692
## Barca - Myra            -87.42 28.0 33  -3.122 0.1624
## Barca - Sira            -114.34 27.2 33  -4.206 0.0127
## Barca - Softone         110.50 28.1 33   3.938 0.0252
## Barca - Stallion        26.27 28.8 33   0.911 0.9997
## Carey - Ilonka          54.00 27.8 33   1.940 0.8165
## Carey - Kennedy        -96.61 27.2 33  -3.556 0.0634
## Carey - Myra           -244.81 28.3 33  -8.656 <.0001
## Carey - Sira           -271.74 27.2 33  -9.985 <.0001
## Carey - Softone        -46.89 27.8 33  -1.684 0.9236
## Carey - Stallion      -131.12 29.1 33  -4.504 0.0057
## Ilonka - Kennedy       -150.62 26.9 33  -5.593 0.0003
## Ilonka - Myra         -298.82 28.2 33 -10.607 <.0001
## Ilonka - Sira         -325.74 27.9 33 -11.670 <.0001
## Ilonka - Softone      -100.90 27.5 33  -3.667 0.0489
## Ilonka - Stallion     -185.12 29.8 33  -6.213 <.0001
## Kennedy - Myra        -148.20 27.3 33  -5.433 0.0004
## Kennedy - Sira        -175.12 27.4 33  -6.401 <.0001
## Kennedy - Softone      49.72 28.9 33   1.718 0.9122
## Kennedy - Stallion    -34.51 28.1 33  -1.229 0.9943
## Myra - Sira           -26.92 29.0 33  -0.927 0.9997
## Myra - Softone        197.92 29.6 33   6.691 <.0001
## Myra - Stallion       113.69 27.0 33   4.208 0.0126
## Sira - Softone         224.84 30.1 33   7.461 <.0001
## Sira - Stallion       140.61 30.2 33   4.657 0.0038
## Softone - Stallion    -84.23 30.2 33  -2.789 0.3005
##
## Results are averaged over the levels of: Block
## P value adjustment: tukey method for comparing a family of 15 estimates
```

```
set.seed(42)

# Get means of each variable per each block
adjusted_dataframe <- new_dataframe %>%
  group_by(Cultivar, Block) %>%
  summarise(
    T9_per_mean = mean(T9_per, na.rm = TRUE),
    T5_per_mean = mean(T5_per, na.rm = TRUE),
    T6_per_mean = mean(T6_per, na.rm = TRUE),
    T7_per_mean = mean(T7_per, na.rm = TRUE),
    T8_per_mean = mean(T8_per, na.rm = TRUE),
    T5_length_mean = mean(T5_length, na.rm = TRUE),
    T5_RGR_mean = mean(T5_RGR, na.rm = TRUE),
  ) %>%
  ungroup()
```

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

```

data_size <- nrow(adjusted_dataframe)

train_indices <- sample(1:data_size, size = round(0.8 * data_size))

train_set <- adjusted_dataframe[train_indices, ]
test_set <- adjusted_dataframe[-train_indices, ]

#First model with all variables
model.1 <- lm(T9_per_mean ~ T5_per_mean + T6_per_mean + T7_per_mean + T8_per_mean + T5_length_mean + T5_RGR_mean, data = train_set)
summary(model.1)

```

```

##
## Call:
## lm(formula = T9_per_mean ~ T5_per_mean + T6_per_mean + T7_per_mean +
##      T8_per_mean + T5_length_mean + T5_RGR_mean, data = train_set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.0442 -3.0921 -0.6718  2.5502  9.5765
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.71763    7.71387  -0.352   0.7265
## T5_per_mean    0.08781    0.17225   0.510   0.6131
## T6_per_mean   -0.58342    0.24941  -2.339   0.0245 *
## T7_per_mean    0.34074    0.23407   1.456   0.1535
## T8_per_mean    1.16301    0.13718   8.478 2.21e-10 ***
## T5_length_mean -0.04025    0.47199  -0.085   0.9325
## T5_RGR_mean    0.12543    0.49259   0.255   0.8003
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.627 on 39 degrees of freedom
## Multiple R-squared:  0.9829, Adjusted R-squared:  0.9802
## F-statistic: 372.9 on 6 and 39 DF,  p-value: < 2.2e-16

```

```

# Second model with earlier time points
model.2 <- lm(T9_per_mean ~ T5_per_mean + T6_per_mean + T5_length_mean + T5_RGR_mean, data = train_set)
summary(model.2)

```

```

##
## Call:
## lm(formula = T9_per_mean ~ T5_per_mean + T6_per_mean + T5_length_mean +
##      T5_RGR_mean, data = train_set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.585 -12.202  -3.284   9.483  30.152
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.9604   24.4169   0.449 0.655878

```

```
## T5_per_mean      -1.6973      0.4705   -3.607 0.000833 ***
## T6_per_mean       2.6597      0.4560    5.832 7.49e-07 ***
## T5_length_mean    0.5463      1.5000    0.364 0.717571
## T5_RGR_mean      -0.5772      1.5699   -0.368 0.714996
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.9 on 41 degrees of freedom
## Multiple R-squared:  0.8133, Adjusted R-squared:  0.7951
## F-statistic: 44.65 on 4 and 41 DF,  p-value: 2.023e-14
```

```
predictions <- predict(model.1, newdata = test_set)

mse <- mean((test_set$T9_per_mean - predictions)^2)

rmse <- sqrt(mse)

print(paste("MSE:", mse))
```

```
## [1] "MSE: 10.0795876264494"
```

```
print(paste("RMSE:", rmse))
```

```
## [1] "RMSE: 3.1748366298834"
```

```
# Model with only week 5 data
model_w5 <- lm(T9_per_mean ~ T5_per_mean + T5_length_mean + T5_RGR_mean, data = train_set)
predictions_w5 <- predict(model_w5, newdata = test_set)
mse_w5 <- mean((test_set$T9_per_mean - predictions_w5)^2)
rmse_w5 <- sqrt(mse_w5)

# Model with only week 6 data
model_w6 <- lm(T9_per_mean ~ T6_per_mean + T5_length_mean + T5_RGR_mean, data = train_set)
predictions_w6 <- predict(model_w6, newdata = test_set)
mse_w6 <- mean((test_set$T9_per_mean - predictions_w6)^2)
rmse_w6 <- sqrt(mse_w6)

print(paste("Week 5 MSE:", mse_w5, "RMSE:", rmse_w5))
```

```
## [1] "Week 5 MSE: 294.548003979153 RMSE: 17.1624008803883"
```

```
print(paste("Week 6 MSE:", mse_w6, "RMSE:", rmse_w6))
```

```
## [1] "Week 6 MSE: 267.717854523651 RMSE: 16.3620858854747"
```

Splitting cultivars into two sets (Susceptible, Resistnat) based on Mean AUC (affected leaves rate)

```

# Assuming auc_results is a list of dataframes, each representing a cultivar
# Function to calculate the mean AUC for each cultivar
mean_auc_per_cultivar <- function(df) {
  mean_auc <- mean(df$AUC, na.rm = TRUE)
  data.frame(Cultivar = unique(df$Cultivar), Mean_AUC = mean_auc)
}

# Apply the function to each element in the list and combine the results
combined_auc_means <- do.call(rbind, lapply(auc_results, mean_auc_per_cultivar))

sorted_auc_means <- combined_auc_means[order(combined_auc_means$Mean_AUC, decreasing = TRUE), ]
split_index <- ceiling(nrow(sorted_auc_means) / 2)

# Splitting into two sets
susceptible_cultivars <- sorted_auc_means[1:split_index, ]
resistant_cultivars <- sorted_auc_means[(split_index + 1):nrow(sorted_auc_means), ]

# Splitting the adjusted_dataframe
resistant_data <- adjusted_dataframe[adjusted_dataframe$Cultivar %in% resistant_cultivars$Cultivar, ]
susceptible_data <- adjusted_dataframe[adjusted_dataframe$Cultivar %in% susceptible_cultivars$Cultivar, ]

# Adding the "Sensitivity" column
resistant_data <- resistant_data %>% mutate(Sensitivity = "Resistant")
susceptible_data <- susceptible_data %>% mutate(Sensitivity = "Susceptible")

T5_t_test_result <- t.test(T5_per_mean ~ Sensitivity, data = rbind(resistant_data, susceptible_data))
T6_t_test_result <- t.test(T6_per_mean ~ Sensitivity, data = rbind(resistant_data, susceptible_data))

print(T5_t_test_result)

##
## Welch Two Sample t-test
##
## data: T5_per_mean by Sensitivity
## t = -7.871, df = 32.166, p-value = 5.373e-09
## alternative hypothesis: true difference in means between group Resistant and group Susceptible is not equal to 0
## 95 percent confidence interval:
## -43.34479 -25.52544
## sample estimates:
## mean in group Resistant mean in group Susceptible
## 2.09478 36.52990

print(T6_t_test_result)

##
## Welch Two Sample t-test
##
## data: T6_per_mean by Sensitivity
## t = -9.2931, df = 33.563, p-value = 8.345e-11
## alternative hypothesis: true difference in means between group Resistant and group Susceptible is not equal to 0
## 95 percent confidence interval:
## -47.19571 -30.25126

```

```
## sample estimates:
##   mean in group Resistant mean in group Susceptible
##               3.95261                42.67610
```