FZJ Data Analysis

Elise

2024-06-09

Set the right working directory.

```
setwd("C:/Users/elise/Documents/Mémoire/Main/Data/Templates/FZJ")
```

Data importation

Import the data sets extracted from the Data Preparation R Markdown.

```
list.files()
```

```
## [1] "endpoint.txt"     "plant_info.txt"   "S_timeseries.txt" "T_timeseries.txt"
## [5] "timeseries.txt"
```

```
plant_info <- read.table("plant_info.txt", header = TRUE, sep = "\t")
endpoint <- read.table("endpoint.txt", header = TRUE, sep = "\t")
timeseries <- read.table("timeseries.txt", header = TRUE, sep = "\t")
```

Convert the columns to factor and date formats.

```
# plant_info
plant_info <- lapply(plant_info, factor)

# endpoint
matching_cols <- intersect(names(endpoint), names(plant_info))
endpoint[, matching_cols] <- lapply(endpoint[, matching_cols], factor)
endpoint$Date <- date(endpoint$Date)
endpoint$Timestamp <- NA

# timeseries
matching_cols <- intersect(names(timeseries), names(plant_info))
timeseries[, matching_cols] <- lapply(timeseries[, matching_cols], factor)
timeseries$Timestamp <- as.POSIXct(timeseries$Timestamp, format = "%Y-%m-%d %H:%M:%S")
timeseries$Date <- date(timeseries$Date)
```

Collect the variables of every data template and print the names of the variables. This serves as a double check.

```
platform <- "FZJ"

# endpoint
df <- endpoint[,colSums(is.na(endpoint))<nrow(endpoint)]
genotype_index <- which(colnames(df) == "Genotype")
variables <- colnames(df[, c(3:(genotype_index - 1))]) # We remove the 3 first columns that are "Uni
t.ID" and "Date" etc

# timeseries
df_timeseries <- timeseries[,colSums(is.na(timeseries))<nrow(timeseries)]
genotype_index <- which(colnames(df_timeseries) == "Genotype")
variables_t <- colnames(df_timeseries[, c(3:(genotype_index - 1))]) # We remove the three first colum
ns that are "Unit.ID","Time" and "Date"


print(paste(platform, ": The variables for endpoint are", paste(variables, collapse = ", "), sep = "
"))
```

```
## [1] "FZJ : The variables for endpoint are DW_shoot_g, FW_shoot_g, DW_root_g, Root_length_cm, Root_
number, Root_angle"
```

```
print(paste(platform, ": The variables for timeseries are", paste(variables_t, collapse = ", "), sep
= " "))
```

```
## [1] "FZJ : The variables for timeseries are Manual_Plant_height_cm, Leaf_number"
```

Add a column Plant_type with three levels, H L and T. This variable is useful to test for heterosis effects.

```
endpoint$Plant_type <- substr(endpoint$Genotype, nchar(as.character(endpoint$Genotype)), nchar(as.cha
racter(endpoint$Genotype)))
timeseries$Plant_type <- substr(timeseries$Genotype, nchar(as.character(timeseries$Genotype)), nchar
(as.character(timeseries$Genotype)))
```

1. Endpoint dataframe

A. Exploration of data

Exploration tables using the rstatix, janitor and skimr packages

```
endpoint %>%
  count(Genotype)
```

```
##   Genotype n
## 1  EPPN1_H 8
## 2  EPPN1_L 8
## 3  EPPN2_H 8
## 4  EPPN2_L 8
## 5 EPPN20_T 8
## 6  EPPN3_H 8
## 7  EPPN3_L 8
## 8  EPPN4_H 8
## 9  EPPN4_L 8
```

```
endpoint %>%
  tabyl(Genotype, Column) %>%
  adorn_totals("row") %>%
  adorn_percentages("row") %>%
  adorn_pct_formatting() %>%
  adorn_ns() %>%
  adorn_title("combined")
```

```
##  Genotype/Column          1          2
##          EPPN1_H 50.0%  (4) 50.0%  (4)
##          EPPN1_L 50.0%  (4) 50.0%  (4)
##          EPPN2_H 50.0%  (4) 50.0%  (4)
##          EPPN2_L 50.0%  (4) 50.0%  (4)
##         EPPN20_T 50.0%  (4) 50.0%  (4)
##          EPPN3_H 50.0%  (4) 50.0%  (4)
##          EPPN3_L 50.0%  (4) 50.0%  (4)
##          EPPN4_H 50.0%  (4) 50.0%  (4)
##          EPPN4_L 50.0%  (4) 50.0%  (4)
##            Total 50.0% (36) 50.0% (36)
```

```
endpoint %>%
  tabyl(Genotype, Row) %>%
  adorn_totals("row") %>%
  adorn_percentages("row") %>%
  adorn_pct_formatting() %>%
  adorn_ns() %>%
  adorn_title("combined")
```

```
##  Genotype/Row         1         2         3         4         5         6
##       EPPN1_H  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##       EPPN1_L  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##       EPPN2_H 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##       EPPN2_L 12.5% (1)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##      EPPN20_T  0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)
##       EPPN3_H  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)
##       EPPN3_L  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##       EPPN4_H  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##       EPPN4_L  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##         Total  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)
##          7         8         9        10        11        12        13
##  12.5% (1)  0.0% (0) 12.5% (1) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)
##  12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)
##         14        15        16        17        18        19        20
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0) 12.5% (1)
##  12.5% (1)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##  12.5% (1)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)
##         21        22        23        24        25        26        27
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)
##  12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0) 12.5% (1)
##   0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##  12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)
##         28        29        30        31        32        33        34
##   0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)
##  12.5% (1)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0) 12.5% (1)
##  12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)
##   0.0% (0) 12.5% (1) 12.5% (1)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0)  0.0% (0) 12.5% (1)  0.0% (0)
##   2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)  2.8% (2)
##         35        36
##   0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)
##   0.0% (0)  0.0% (0)
##  12.5% (1)  0.0% (0)
##   0.0% (0) 25.0% (2)
##   0.0% (0)  0.0% (0)
```

```
##   0.0% (0)  0.0% (0)
##  12.5% (1)  0.0% (0)
##   2.8% (2)  2.8% (2)
```

```
get_summary_stats(data = endpoint,
                  variables,
                  type = "common")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## ℹ Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(variables)
##
##   # Now:
##   data %>% select(all_of(variables))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## # A tibble: 6 × 10
##   variable      n     min     max  median     iqr    mean      sd     se      ci
##   <fct>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
## 1 DW_shoot…    71   0.134 1.03e+0 4.78e-1   0.311 5.28e-1   0.202  0.024   0.048
## 2 FW_shoot…    69   1.95  1.44e+1 6.73e+0   4.63  7.16e+0   2.94   0.354   0.707
## 3 DW_root_g    72   0.018 1.61e-1 8.2 e-2   0.039 8.4 e-2   0.03   0.003   0.007
## 4 Root_len…    72 275.    4.42e+3 1.70e+3 940.    1.82e+3 817.    96.3   192.
## 5 Root_num…    67   4     1.7 e+1 1   e+1   2     9.79e+0   2.35   0.287   0.572
## 6 Root_ang…    42  11     1.3 e+2 8.65e+1  40.8   8.33e+1  29.5    4.55    9.19
```

```
skim(endpoint[variables])
```

Data summary

Name	endpoint[variables]
Number of rows	72
Number of columns	6
_____	
Column type frequency:	
numeric	6
_____	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
DW_shoot_g	1	0.99	0.53	0.20	0.13	0.37	0.48	0.68	1.03	
FW_shoot_g	3	0.96	7.16	2.94	1.95	4.35	6.73	8.98	14.37	
DW_root_g	0	1.00	0.08	0.03	0.02	0.06	0.08	0.10	0.16	
Root_length_cm	0	1.00	1824.79	816.81	275.45	1248.03	1701.37	2188.20	4418.41	
Root_number	5	0.93	9.79	2.35	4.00	9.00	10.00	11.00	17.00	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Root_angle	30	0.58	83.26	29.50	11.00	64.00	86.50	104.75	130.00	▂▅▆▇▅

Data visualization

Using several functions that are located in the functions.R script

Boxplots

```
create_boxplots(endpoint, variables, "Genotype")
```

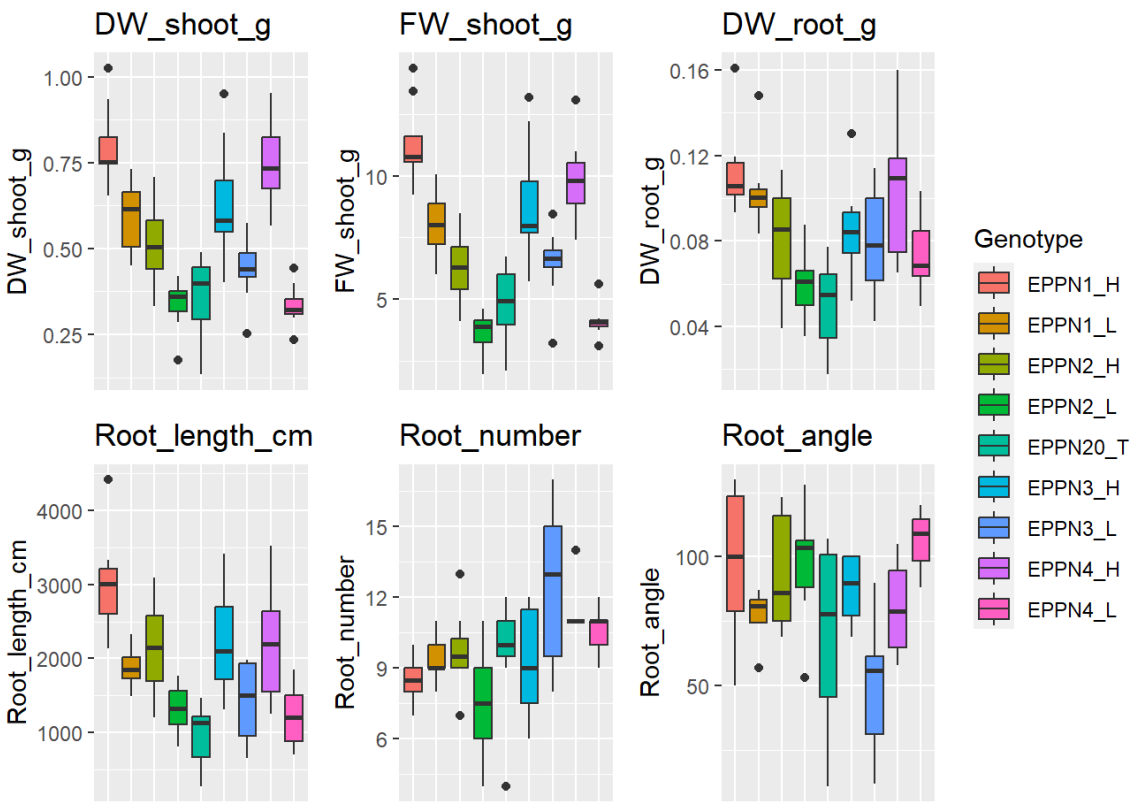
```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## ℹ Please use tidy evaluation idioms with `aes()`.
## ℹ See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 30 rows containing non-finite values (`stat_boxplot()`).
```



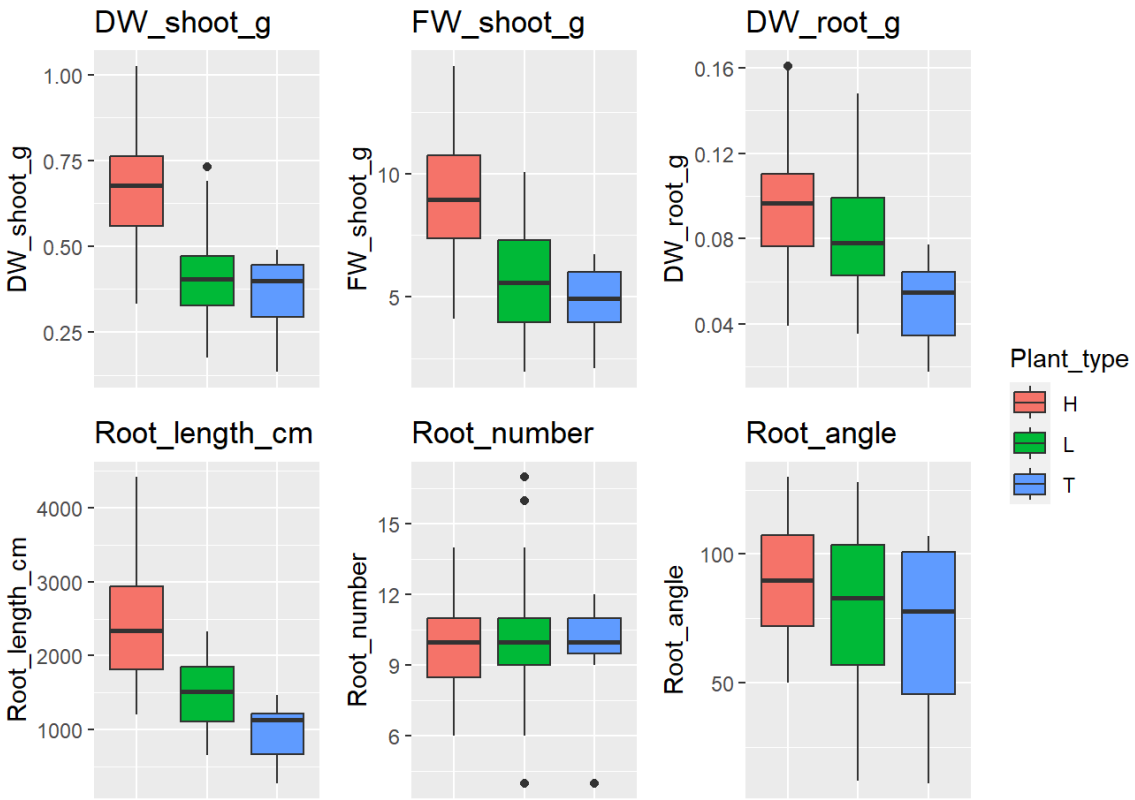

```
create_boxplots(endpoint, variables, "Plant_type")
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 30 rows containing non-finite values (`stat_boxplot()`).
```



Correlation plots

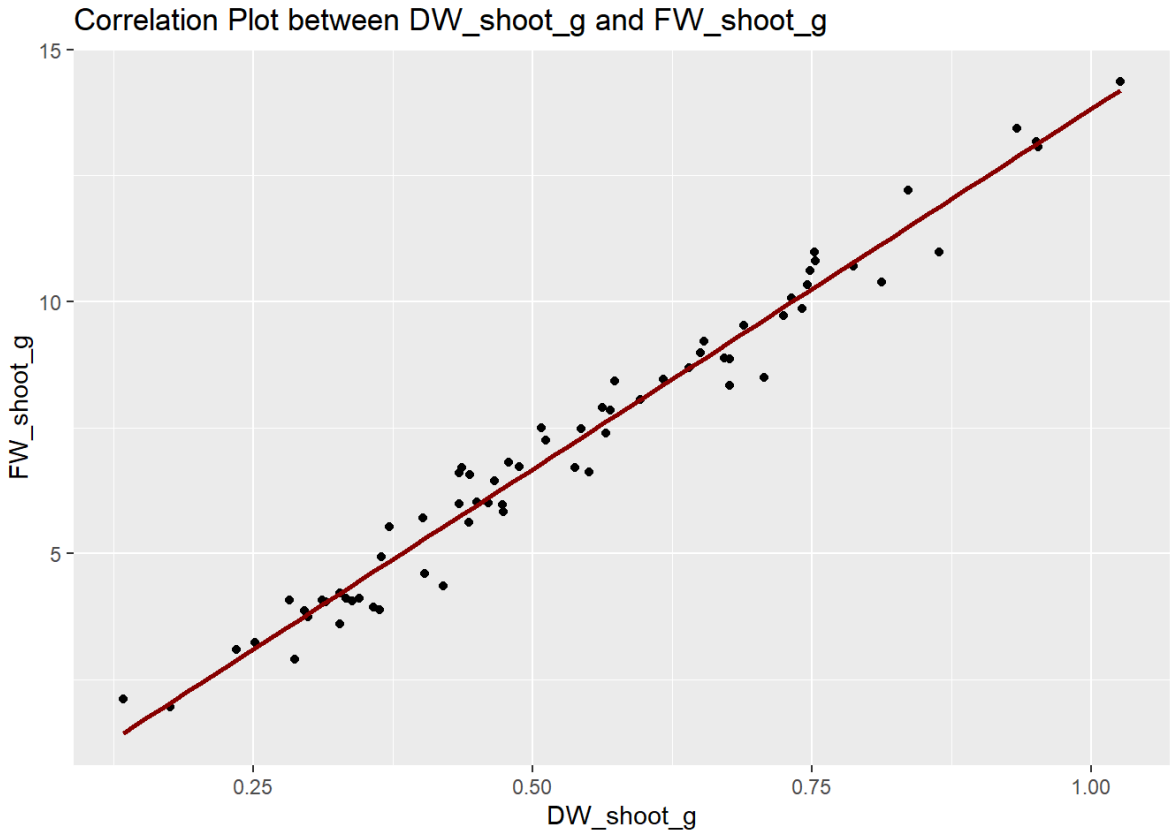
```
for (i in 1:(length(variables) - 1)) {
  for (j in (i + 1):length(variables)) {
    calculate_correlation_plot(endpoint, variables[i], variables[j])
  }
}
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 4 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 4 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

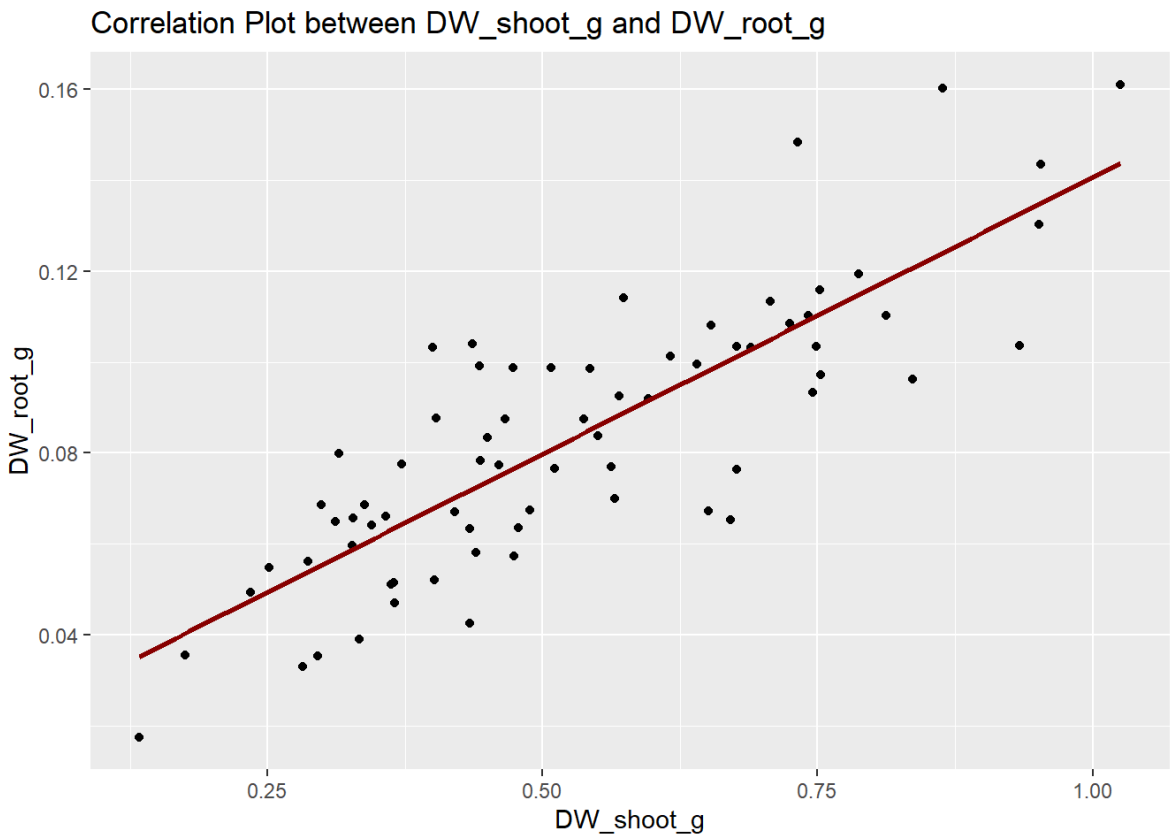



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

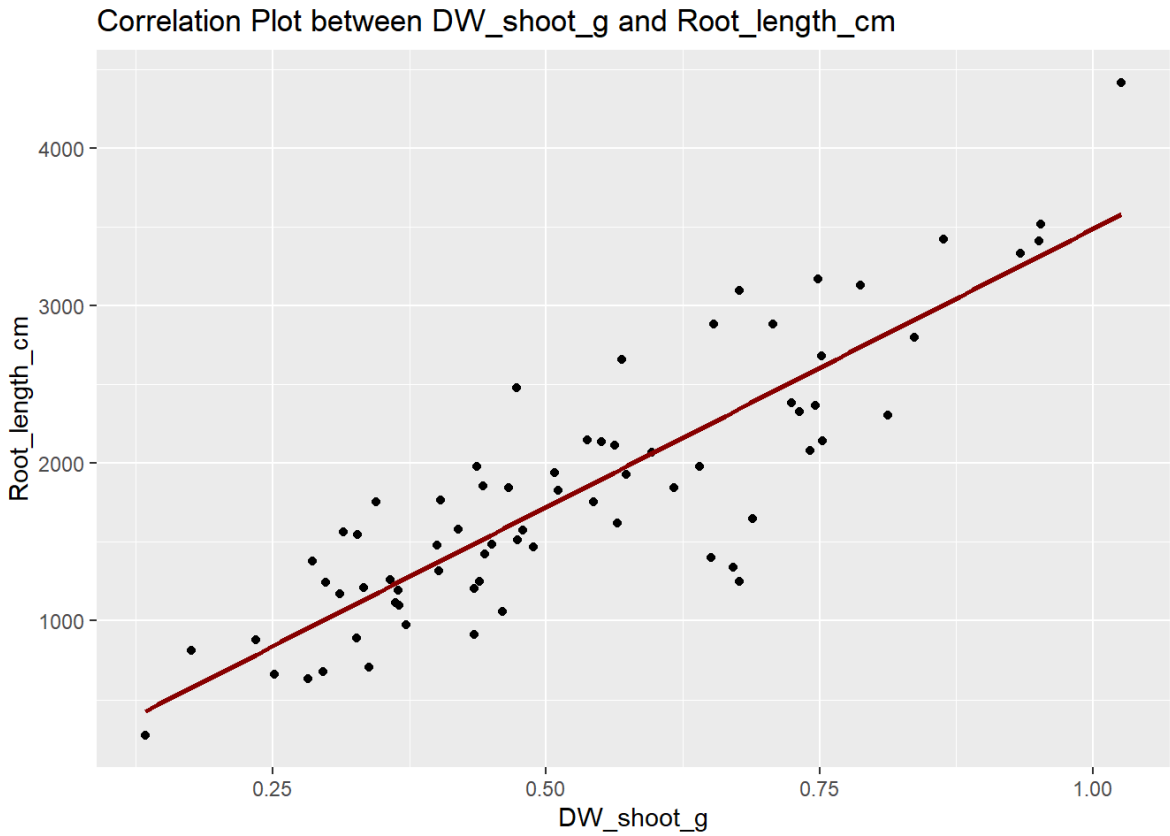


```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

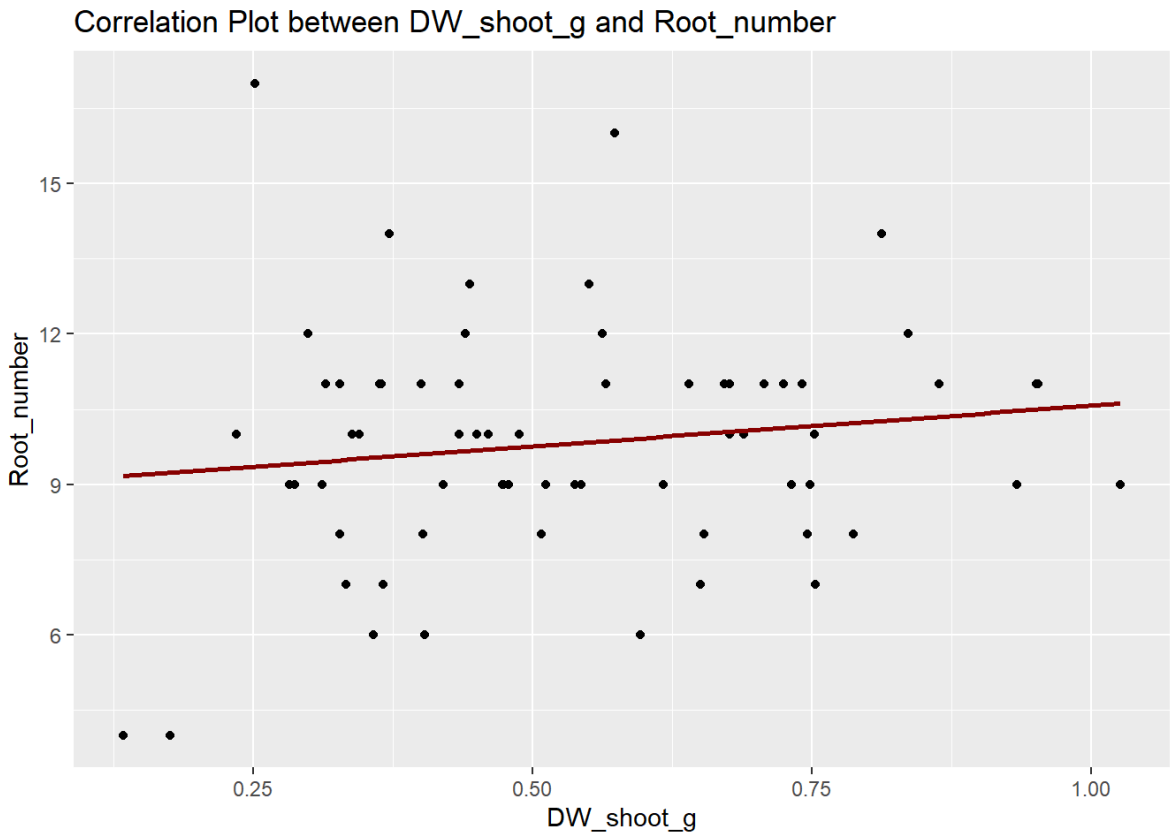



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 6 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 6 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

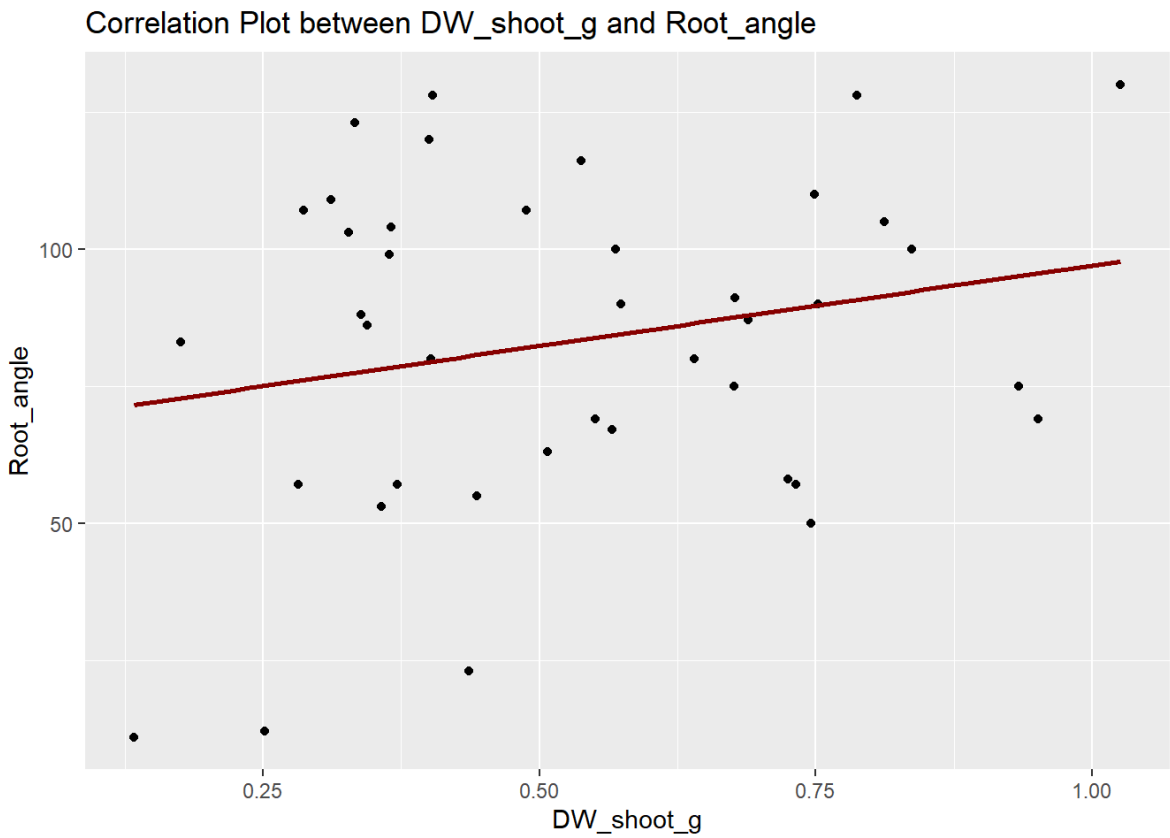



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 31 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 31 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

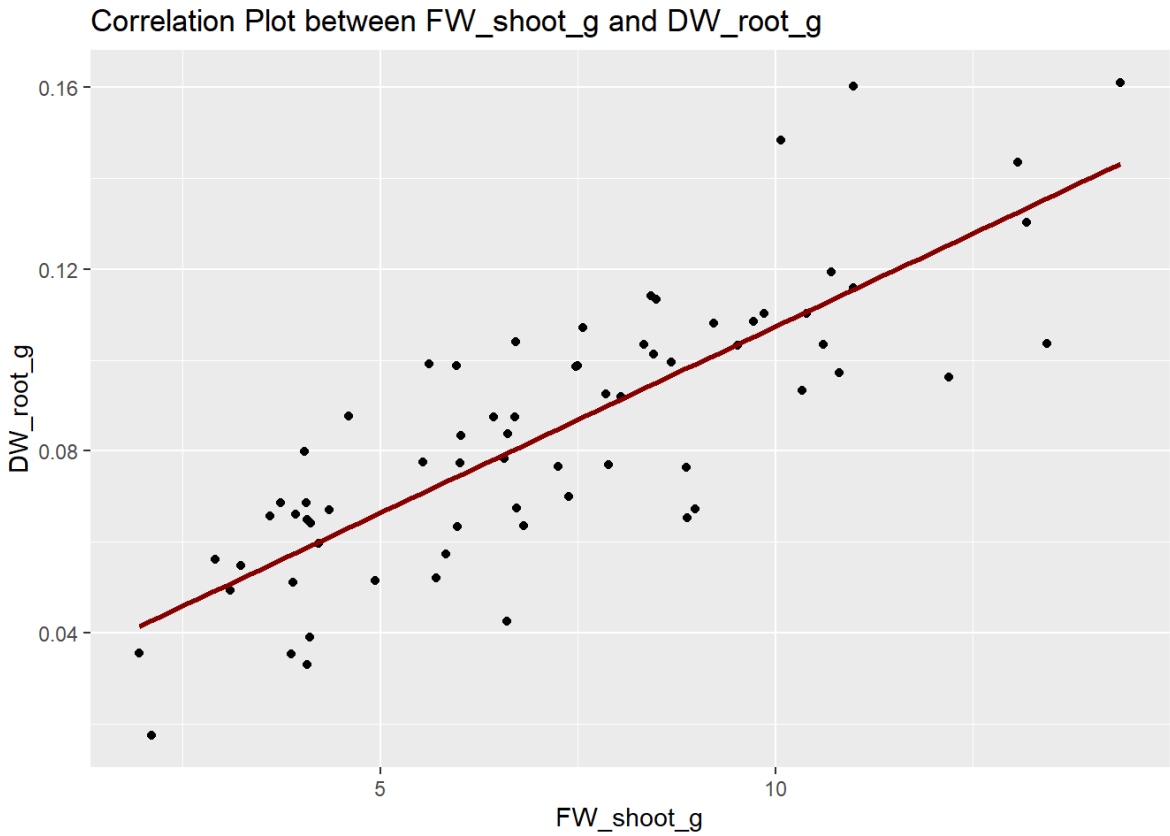


```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 3 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

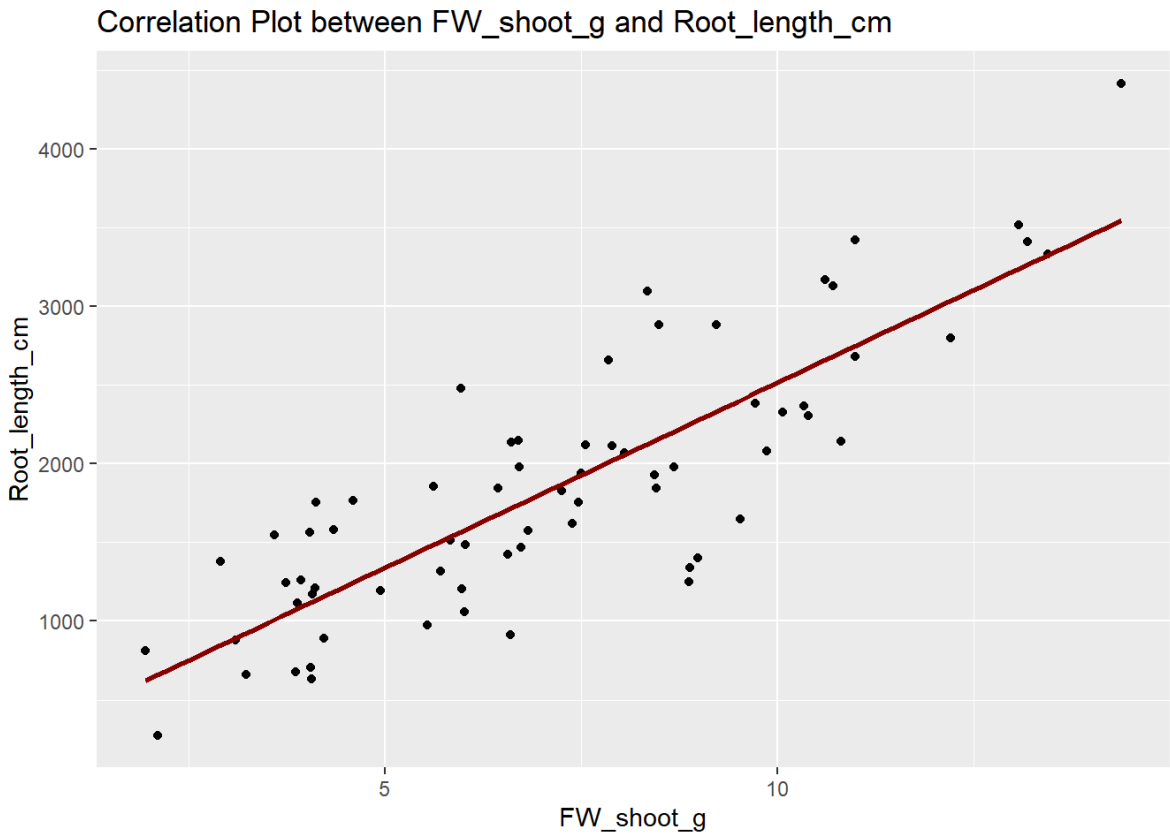



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 3 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

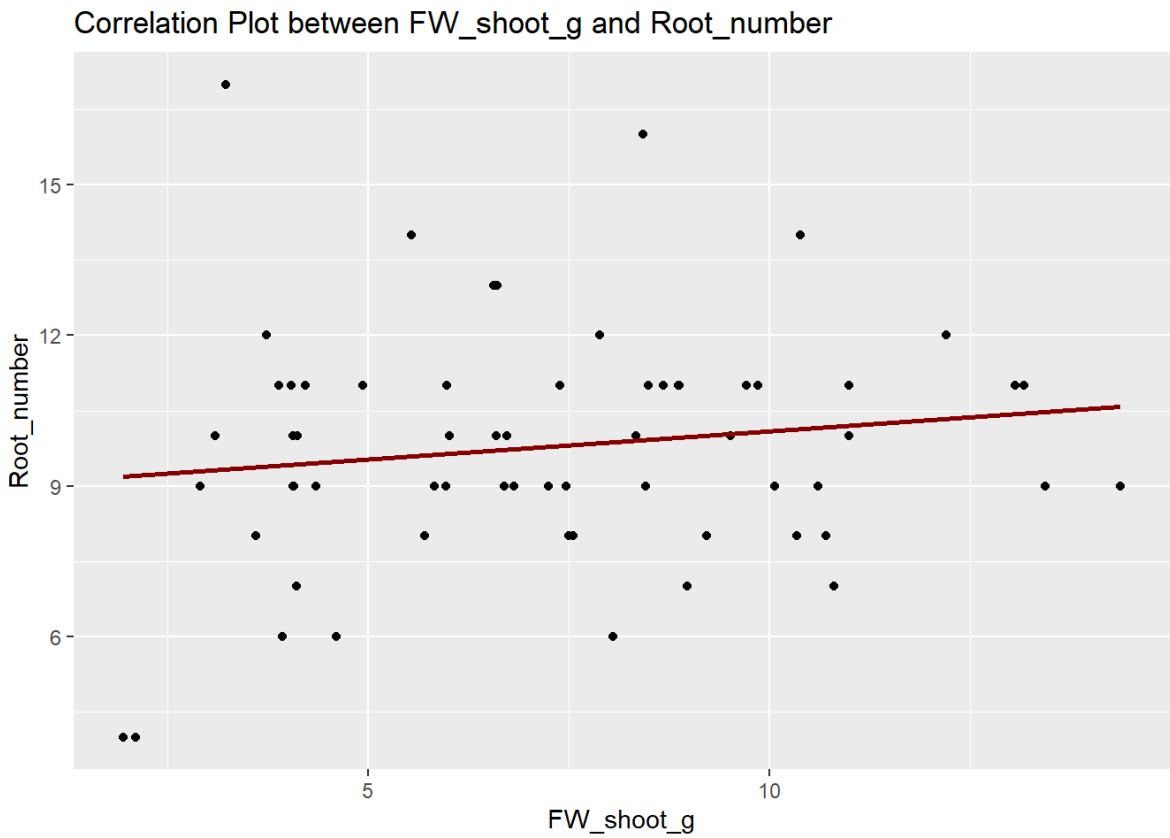



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 8 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 8 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

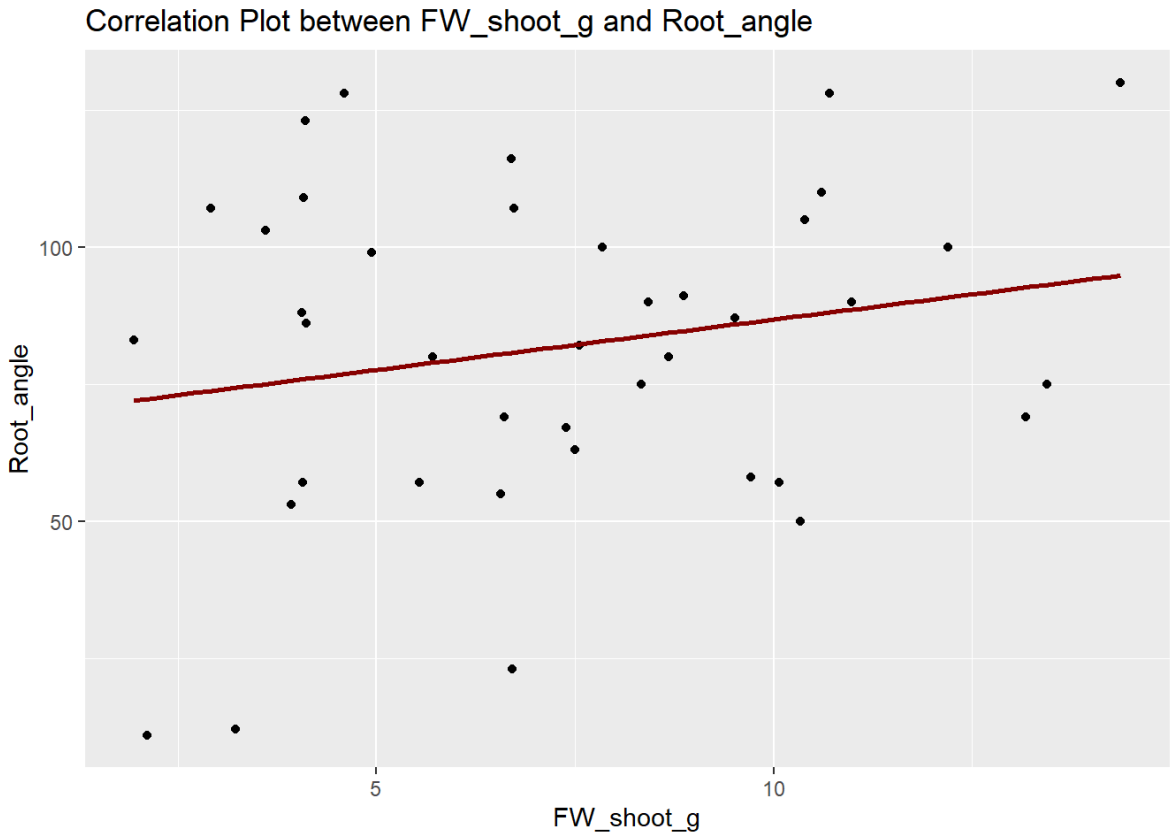


```
## `geom_smooth()` using formula = 'y ~ x'
```

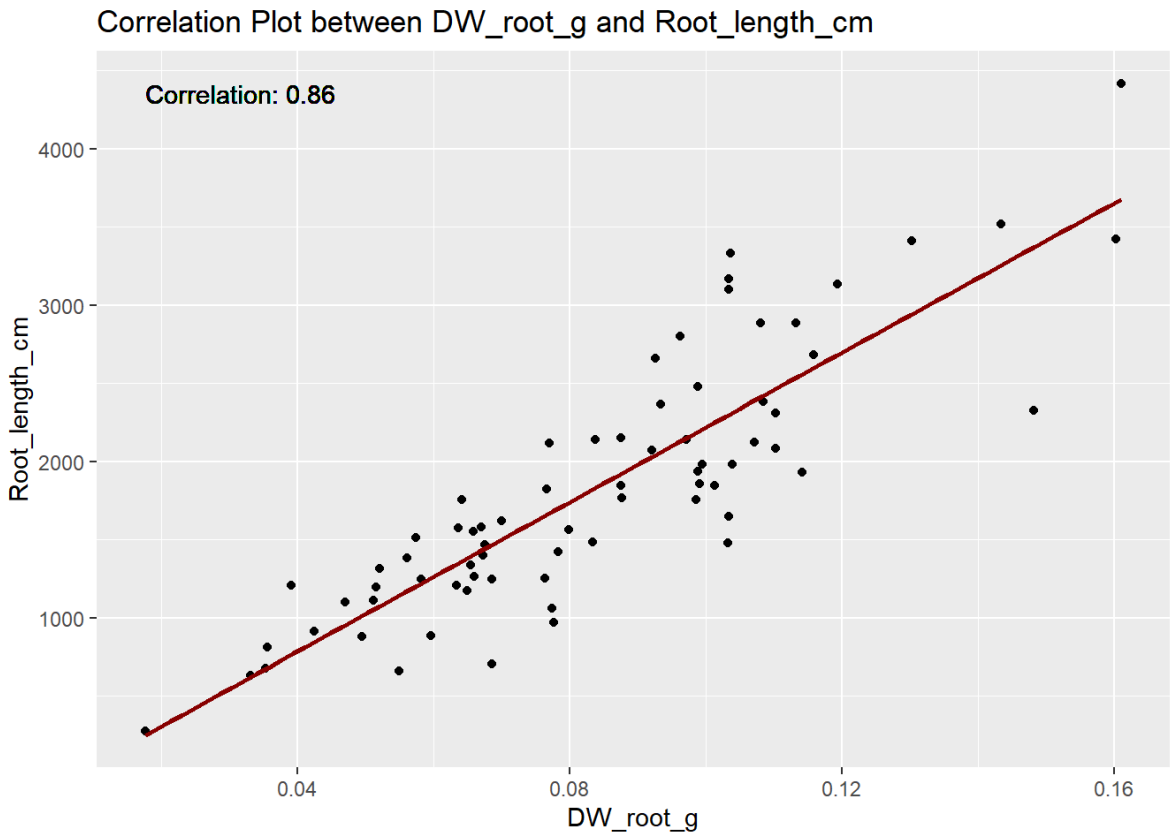
```
## Warning: Removed 32 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 32 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```




```
## `geom_smooth()` using formula = 'y ~ x'
```

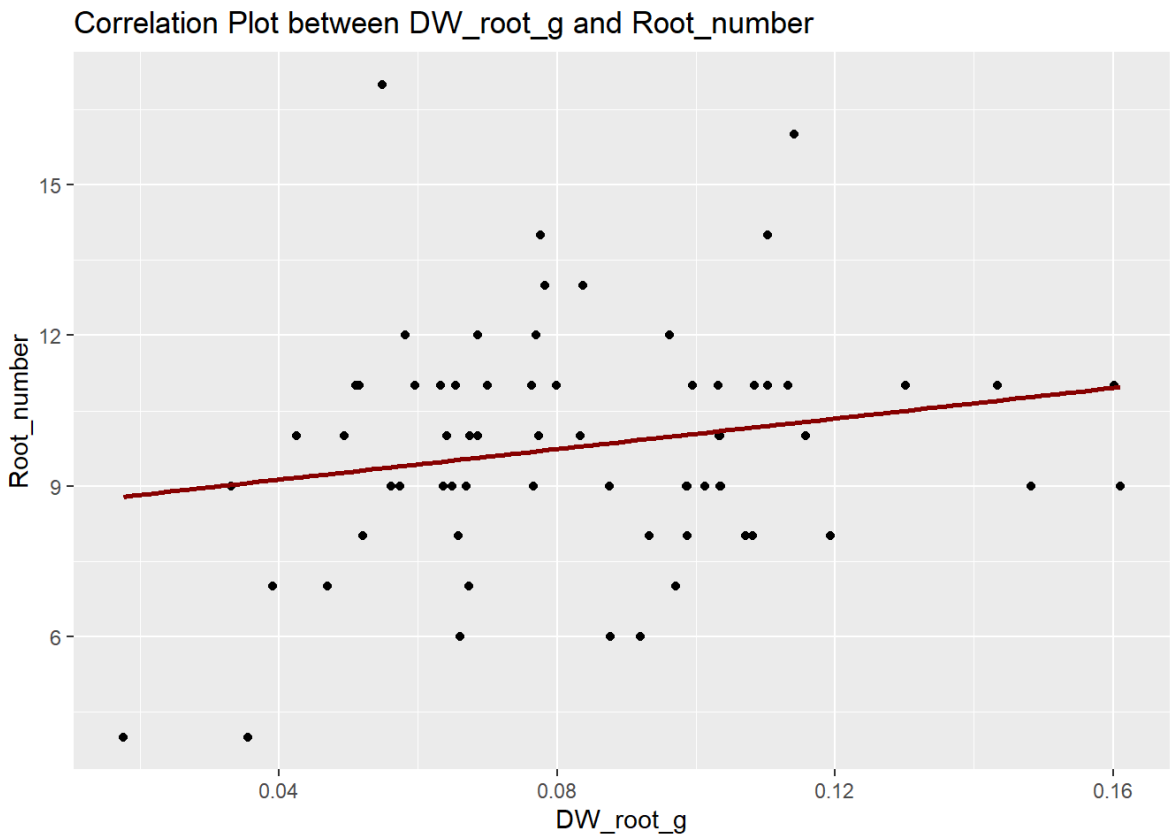



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 5 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

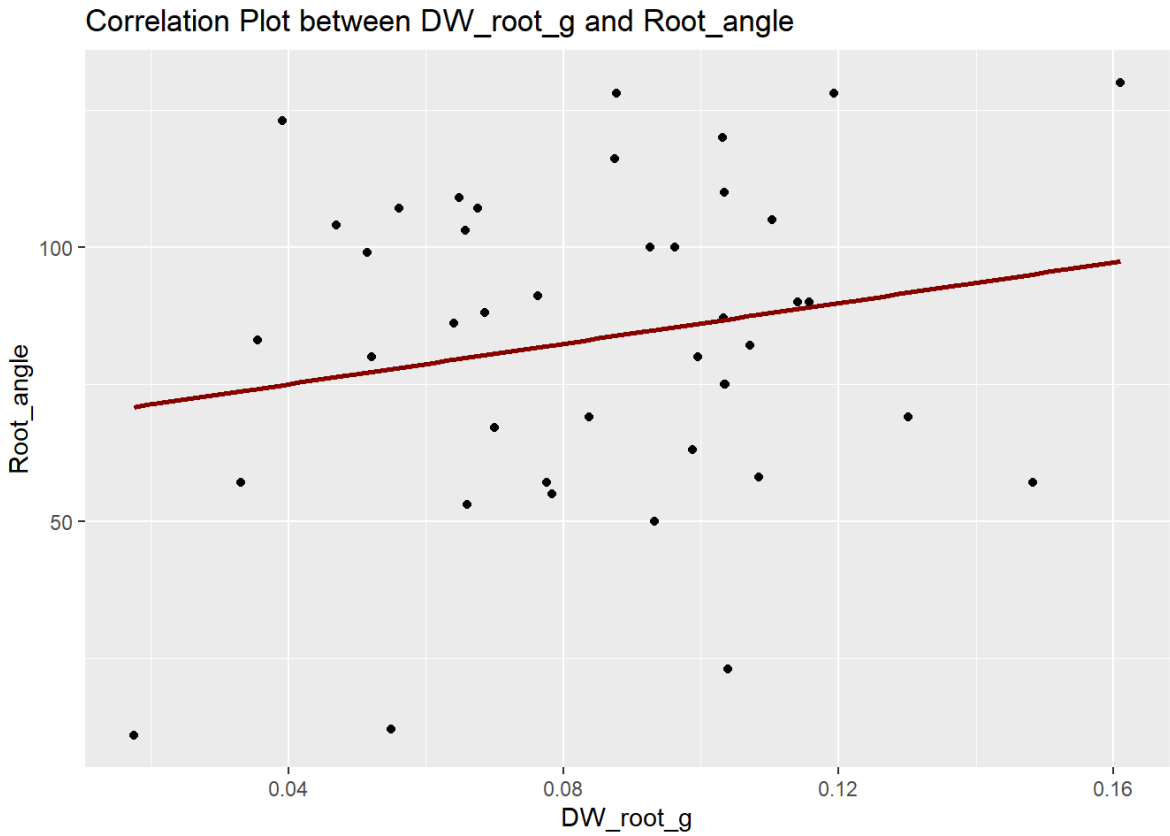


```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 30 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 30 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

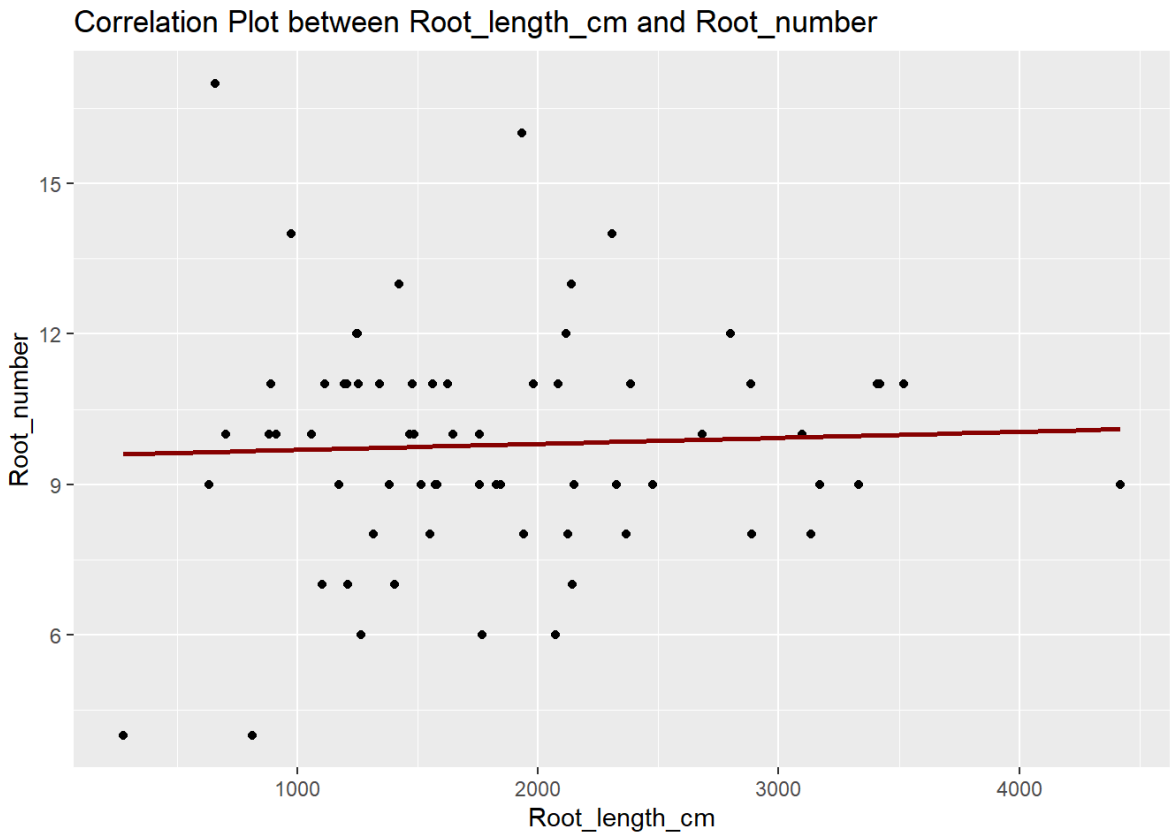



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 5 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

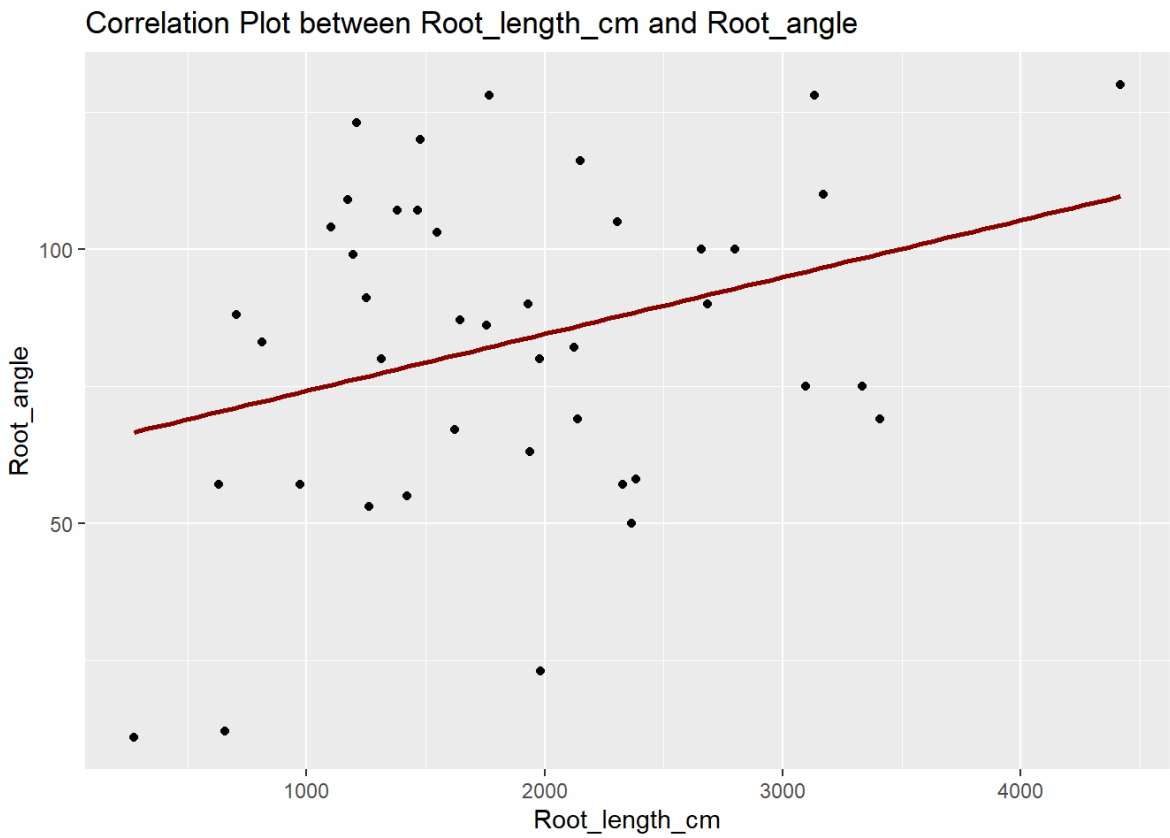



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 30 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 30 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

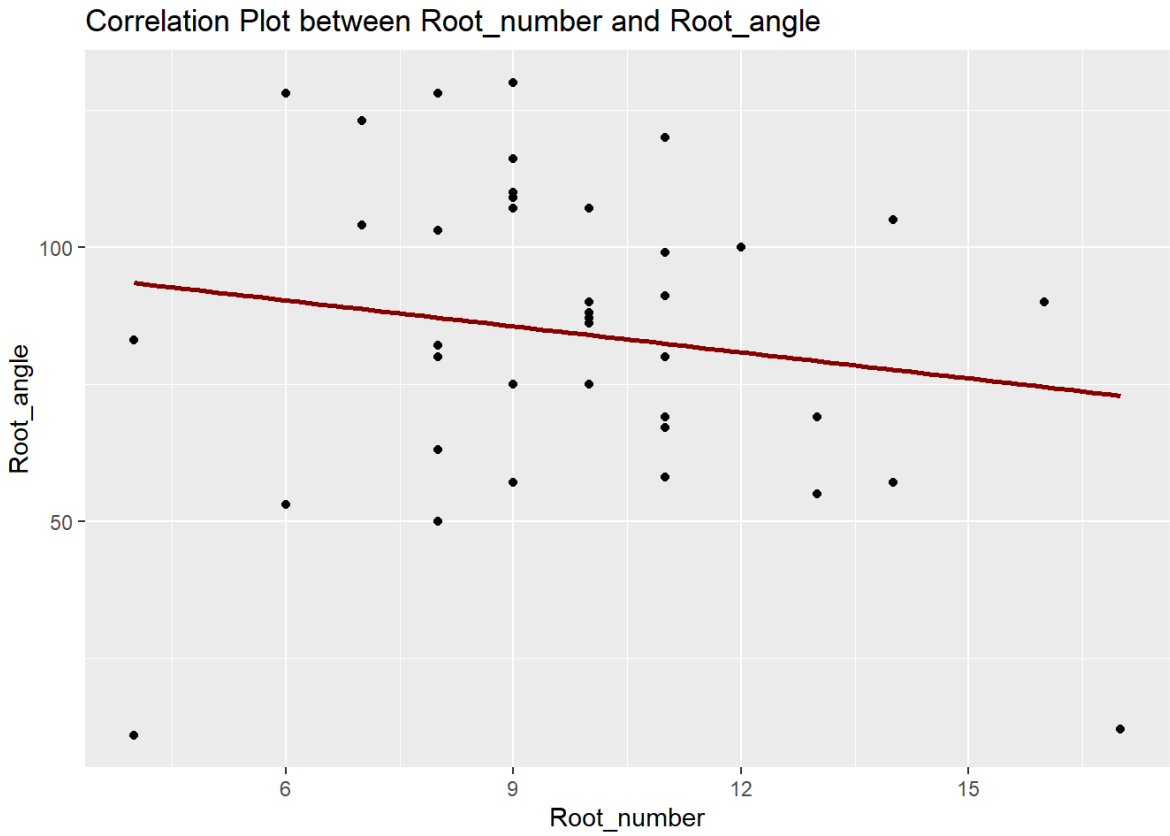


```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 32 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 72 rows containing missing values (`geom_text()`).
```

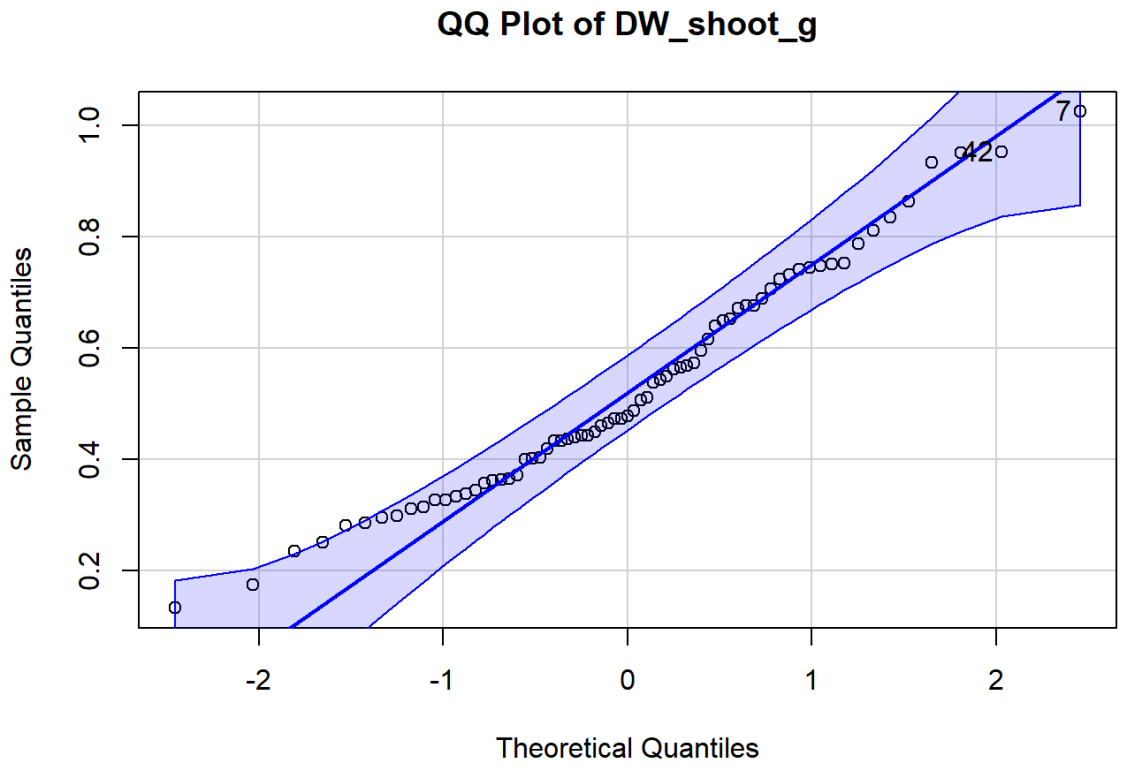


B. Normality hypothesis and outlier detection

Test for normality hypothesis and plot density histogram. The red curve is the normal distribution, the blue dotted curve is the data density curve.

```
normality_results <- normality_test_histogram(endpoint)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

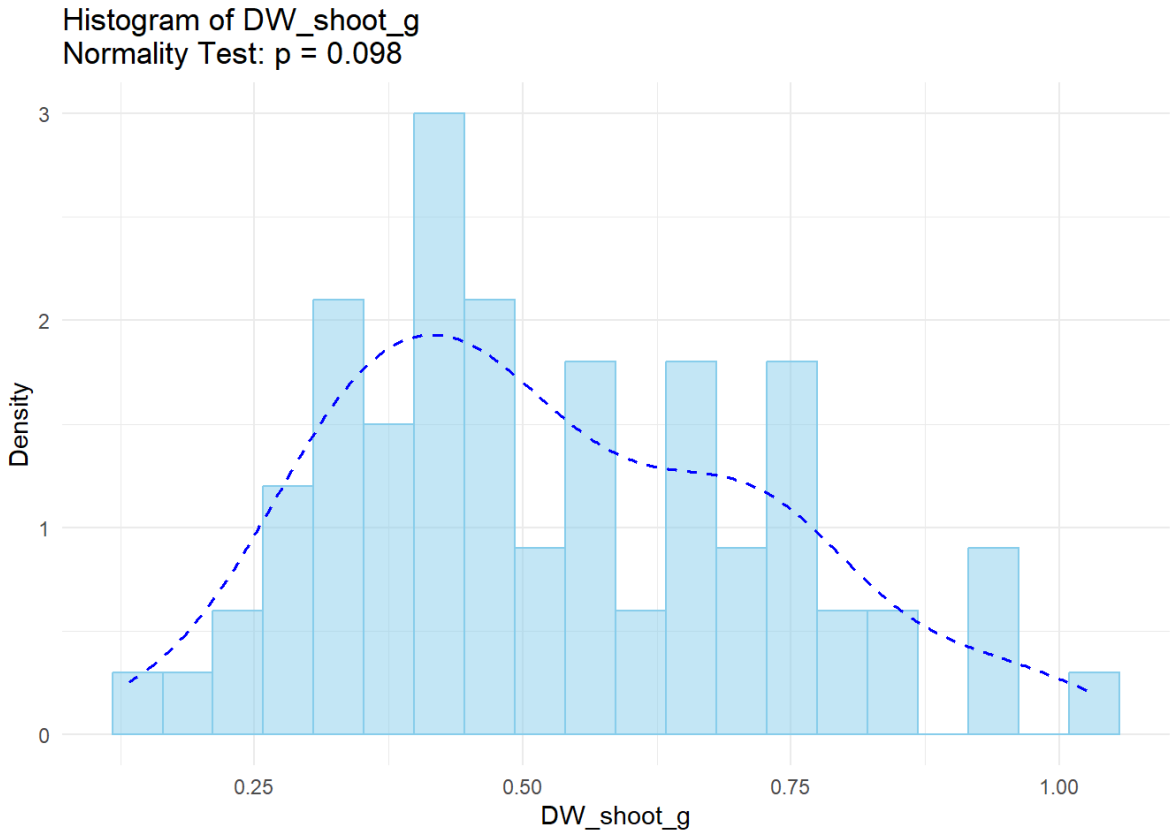



```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

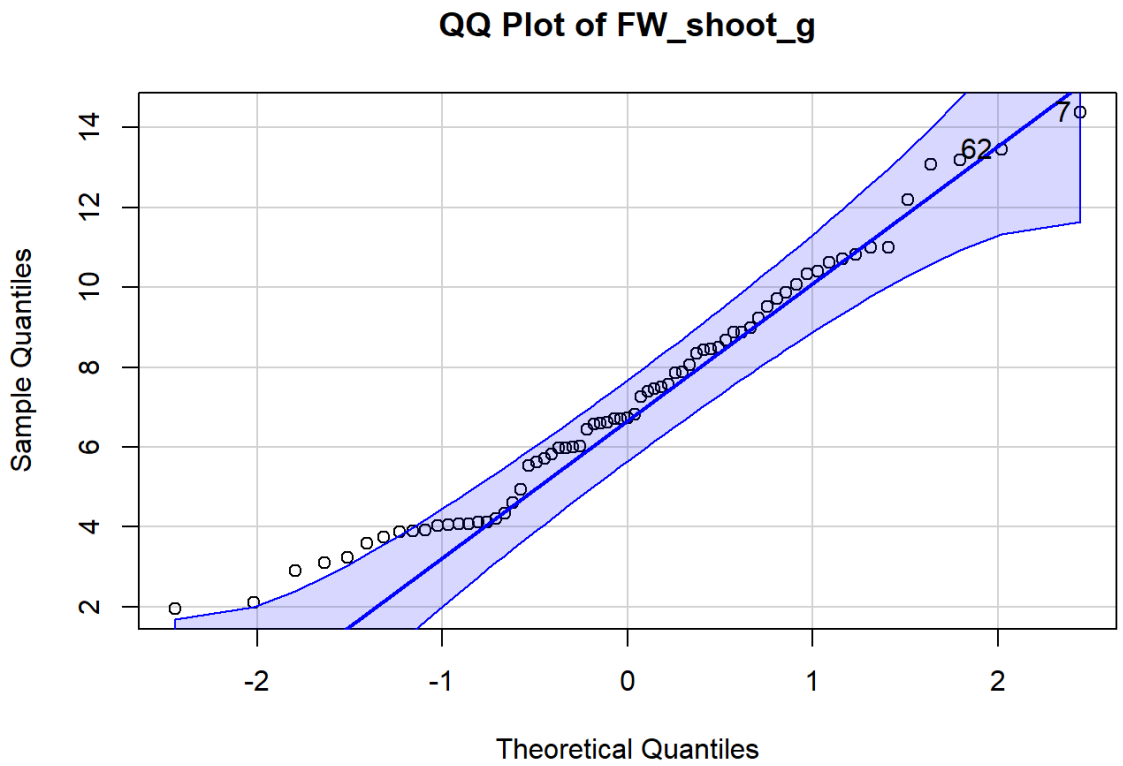
```
## Warning: Removed 1 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_density()`).
```

```
## Warning: Removed 101 rows containing missing values (`geom_function()`).
```



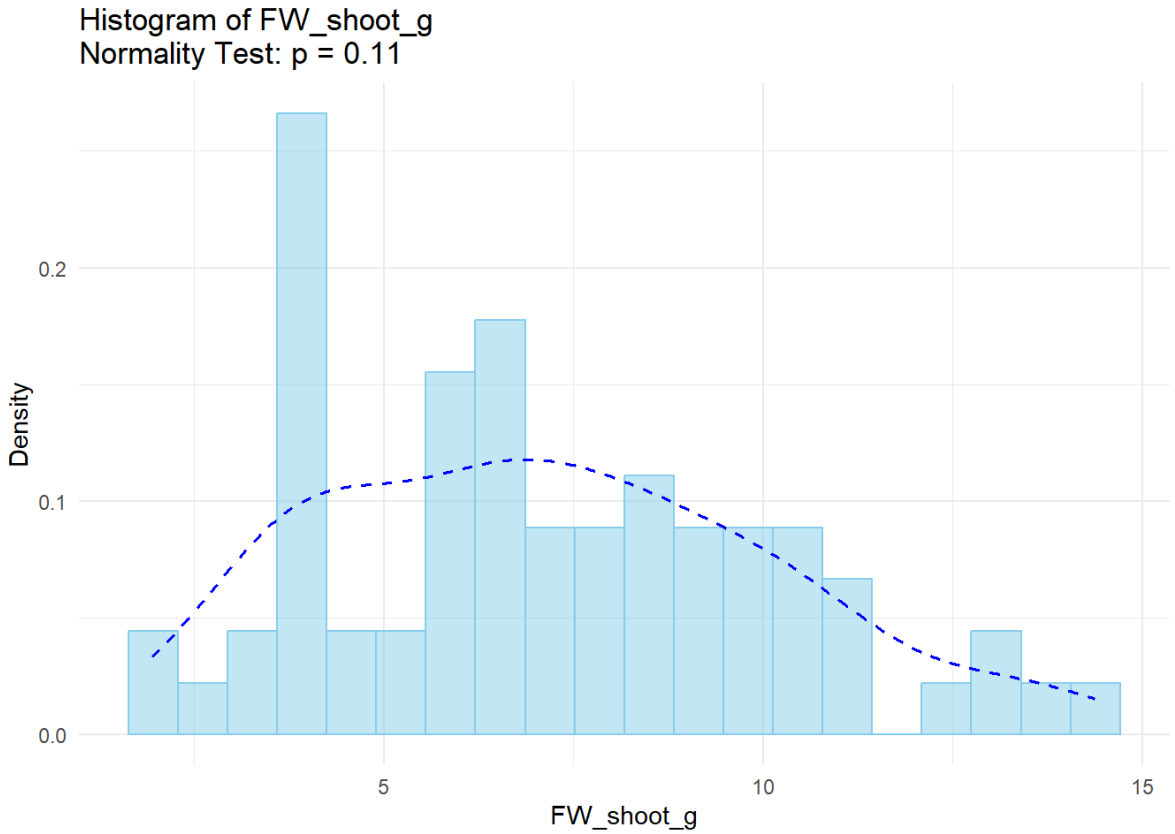

```
## [1]  7 42
```



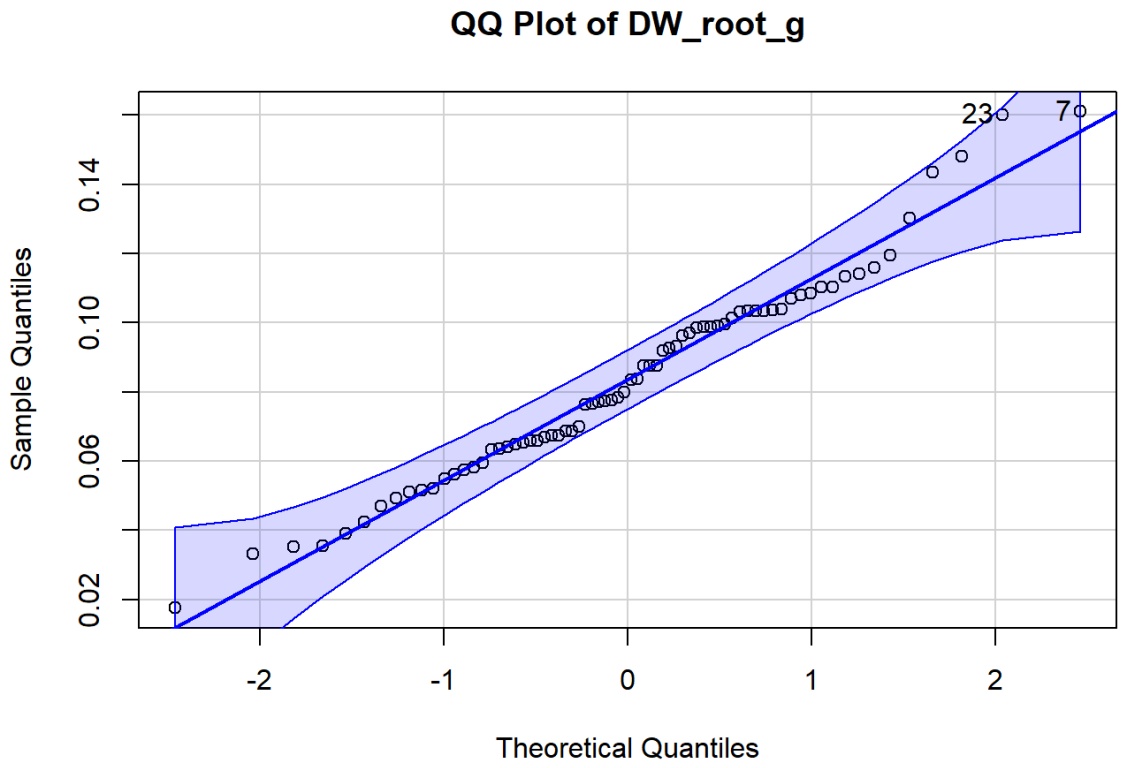

```
## Warning: Removed 3 rows containing non-finite values (`stat_bin()`).
```

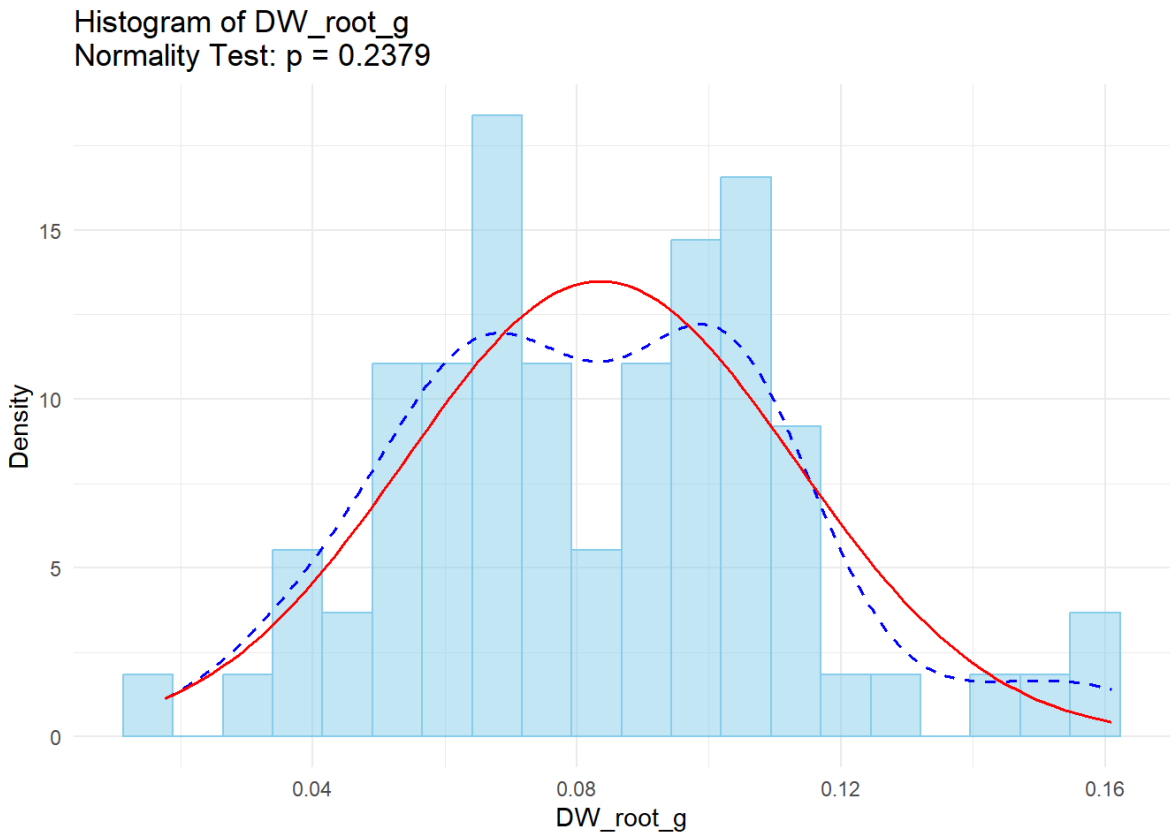
```
## Warning: Removed 3 rows containing non-finite values (`stat_density()`).
```

```
## Warning: Removed 101 rows containing missing values (`geom_function()`).
```

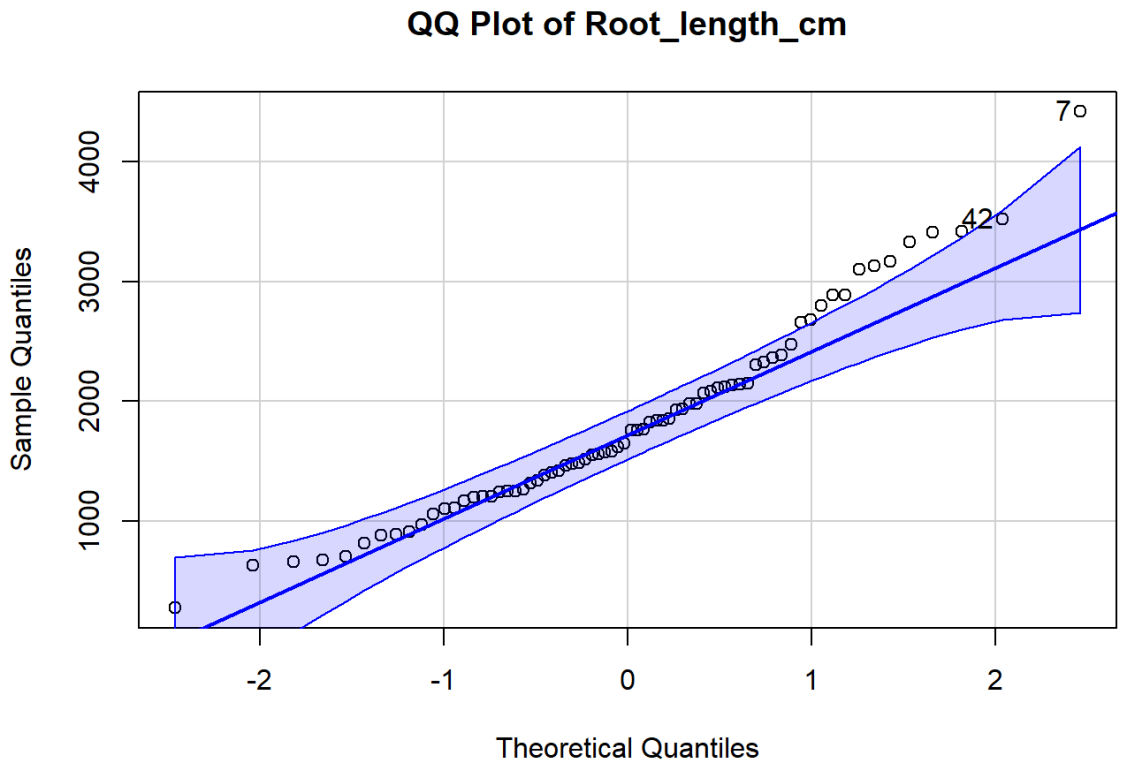

Histogram of FW_shoot_g
Normality Test: p = 0.11
Density
0.0
0.1
0.2
5
10
15
FW_shoot_g

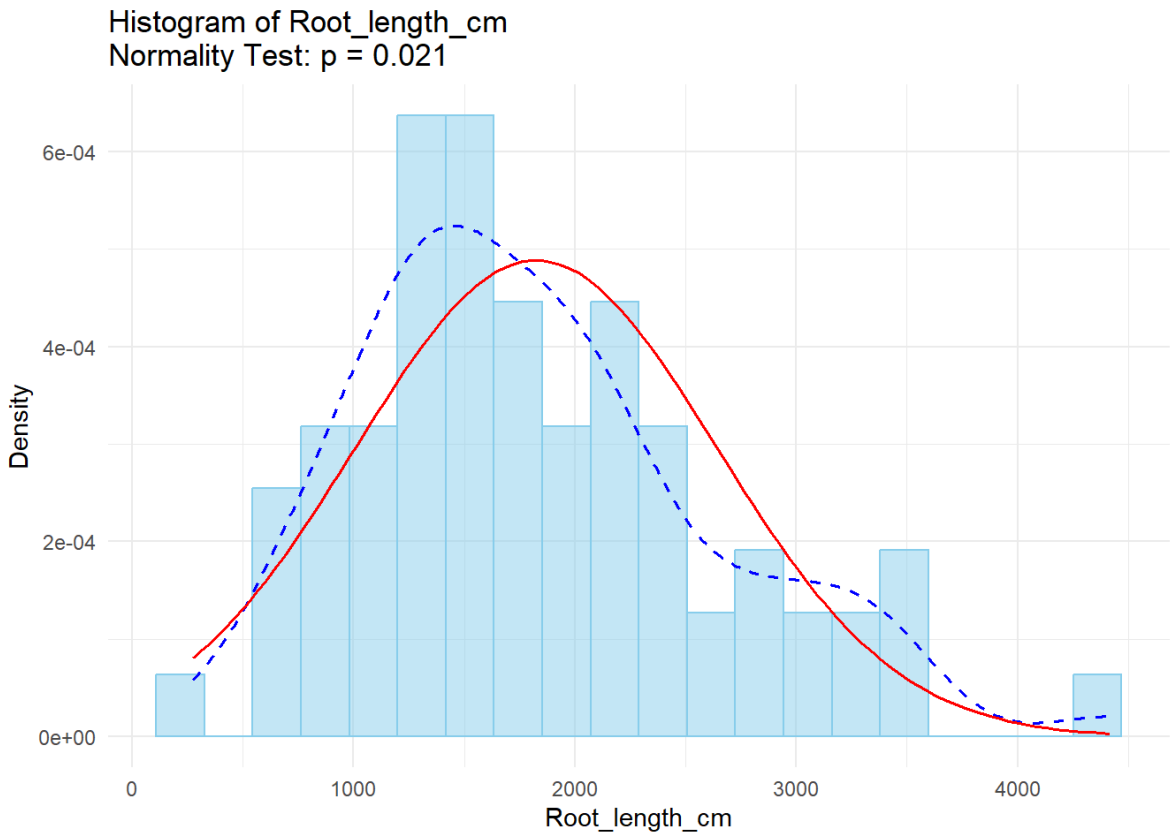
```
## [1]  7 62
```


QQ Plot of DW_root_g
Sample Quantiles
Theoretical Quantiles
0.02
0.06
0.10
0.14
-2
-1
0
1
2
23
7

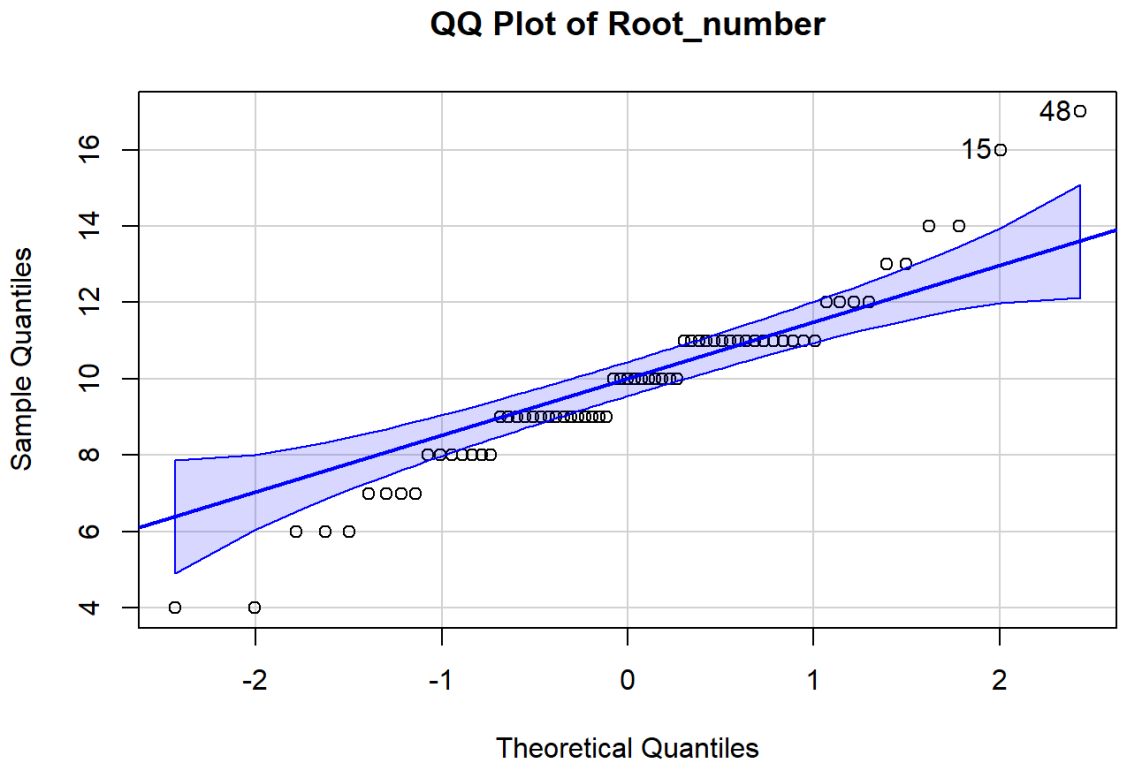

Histogram of DW_root_g
Normality Test: p = 0.2379
Density
DW_root_g
0
5
10
15
0.04
0.08
0.12
0.16

```
## [1]  7 23
```


QQ Plot of Root_length_cm
Sample Quantiles
Theoretical Quantiles
7
42
1000
2000
3000
4000
-2
-1
0
1
2


Histogram of Root_length_cm
Normality Test: p = 0.021
Density
Root_length_cm
0e+00
2e-04
4e-04
6e-04
0
1000
2000
3000
4000

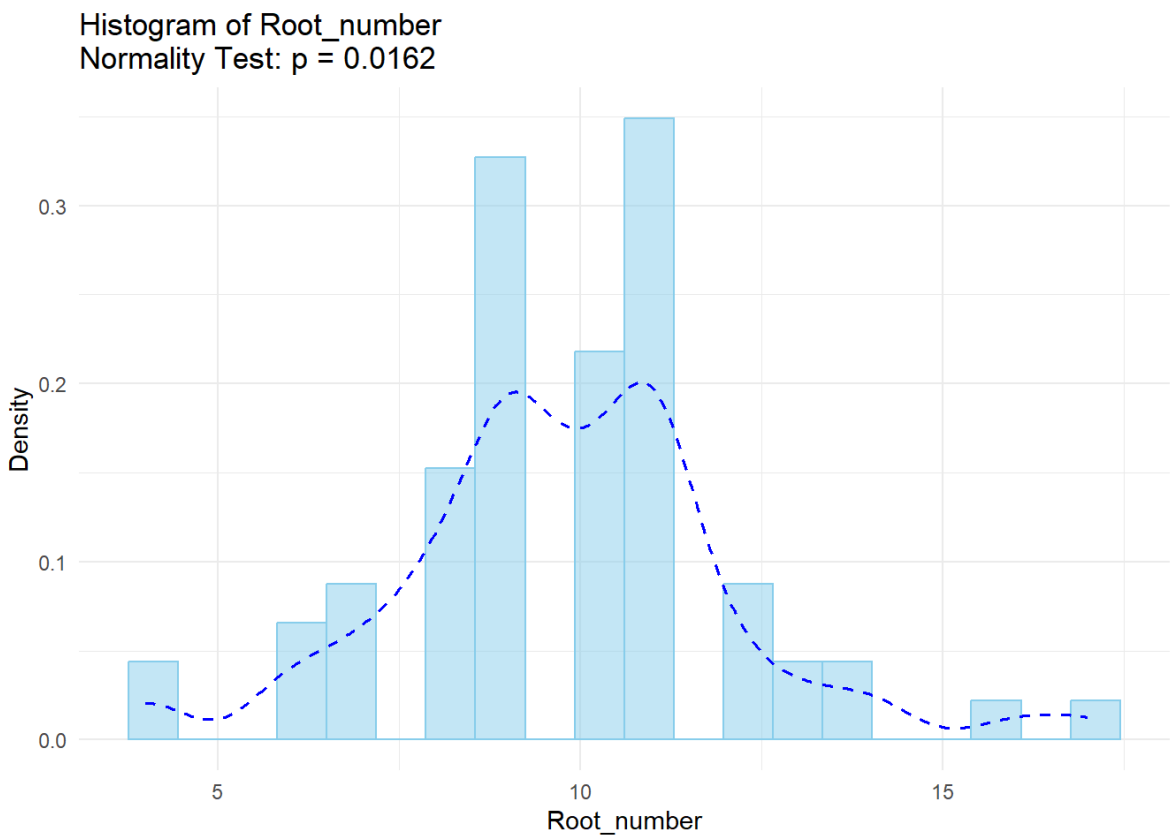
```
## [1]  7 42
```



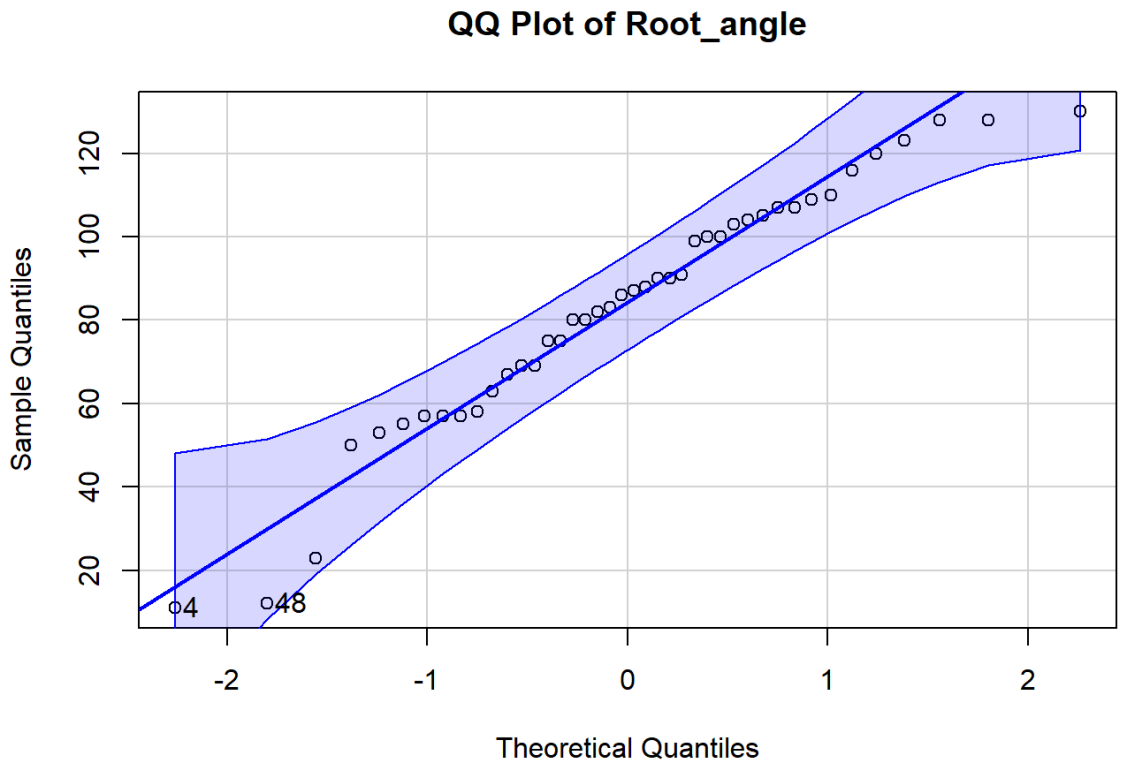

```
## Warning: Removed 5 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_density()`).
```

```
## Warning: Removed 101 rows containing missing values (`geom_function()`).
```



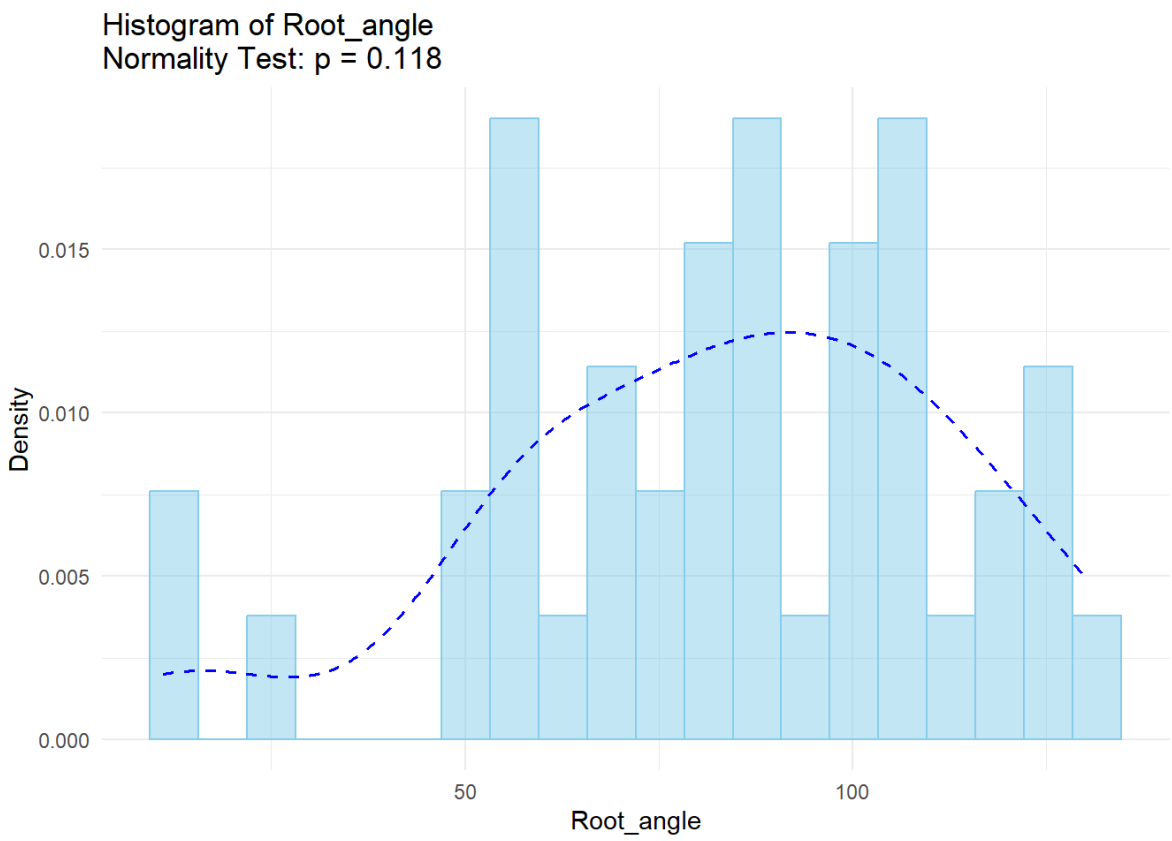

```
## [1] 48 15
```




```
## Warning: Removed 30 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 30 rows containing non-finite values (`stat_density()`).
```

```
## Warning: Removed 101 rows containing missing values (`geom_function()`).
```




```
## [1]  4 48
```

Remove the outliers, replacing them with NULL values and normality visual verification.

The function detect_replace_ouliers_by_genotype checks for outlying values, using the Tukey method.

Then run the function on all variables of the dataset.

```
endpoint_clean <- endpoint
# Run the function on the dataset for all the variables
endpoint_clean <- detect_replace_outliers_by_genotype(endpoint_clean)
```

Boxplots after outlier detection

```
create_boxplots(endpoint_clean, variables, "Genotype")
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_boxplot()`).
```

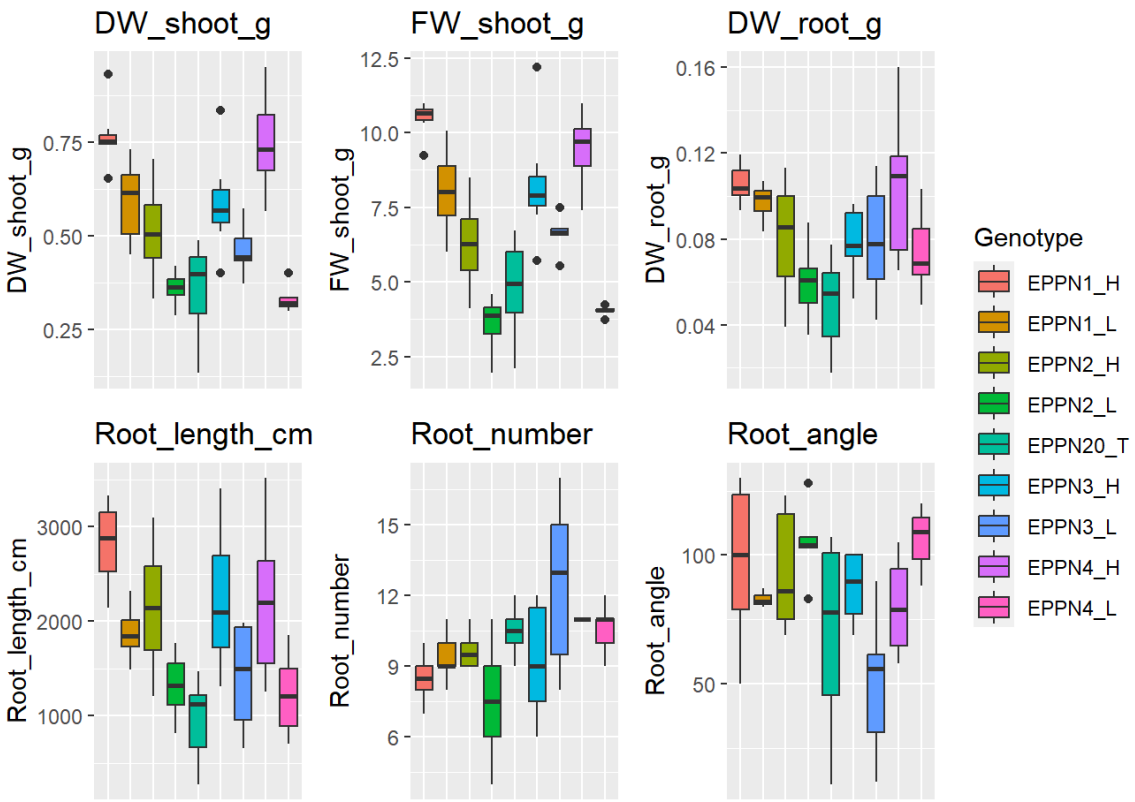
```
## Warning: Removed 11 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_boxplot()`).
```




```
create_boxplots(endpoint_clean, variables, "Plant_type")
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_boxplot()`).
```

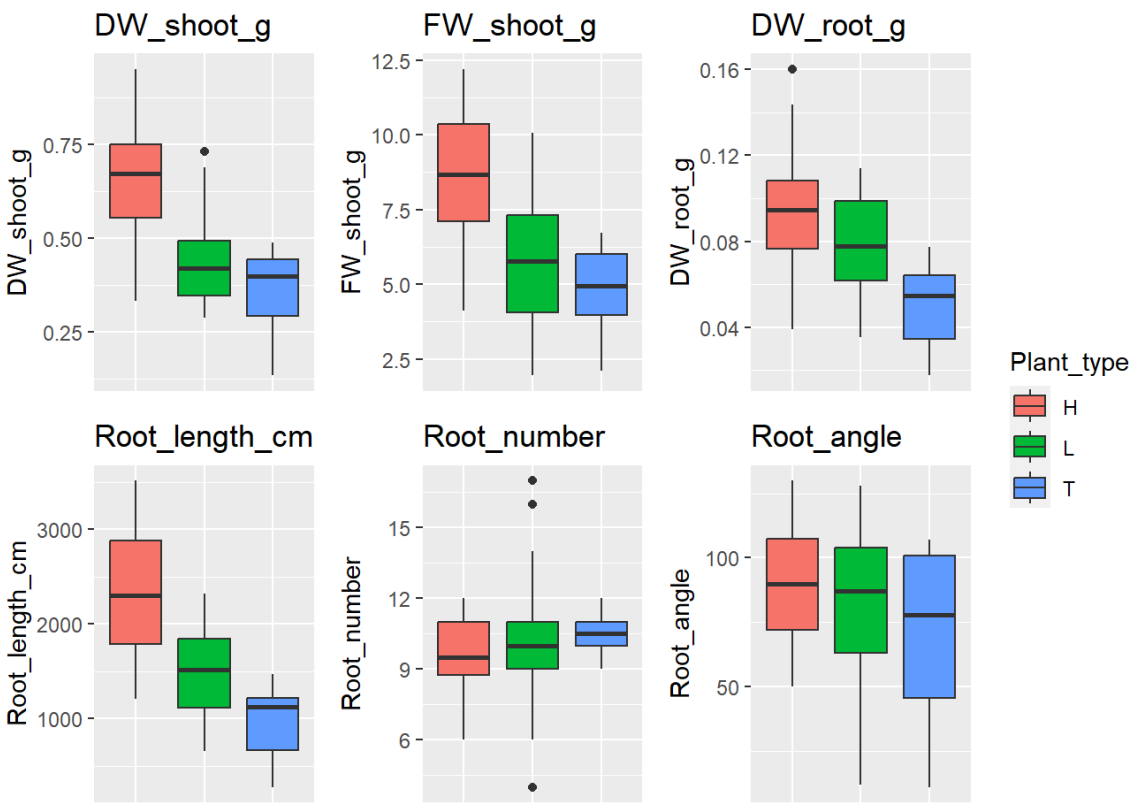
```
## Warning: Removed 11 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_boxplot()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_boxplot()`).
```



Violin and sina plots after outlier detection

```
create_violin_plots(endpoint_clean, variables, "Genotype")
```

```
## Warning: The `size` argument of `element_line()` is deprecated as of ggplot2 3.4.0.
## ℹ Please use the `linewidth` argument instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 11 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 11 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_ydensity()`).
```

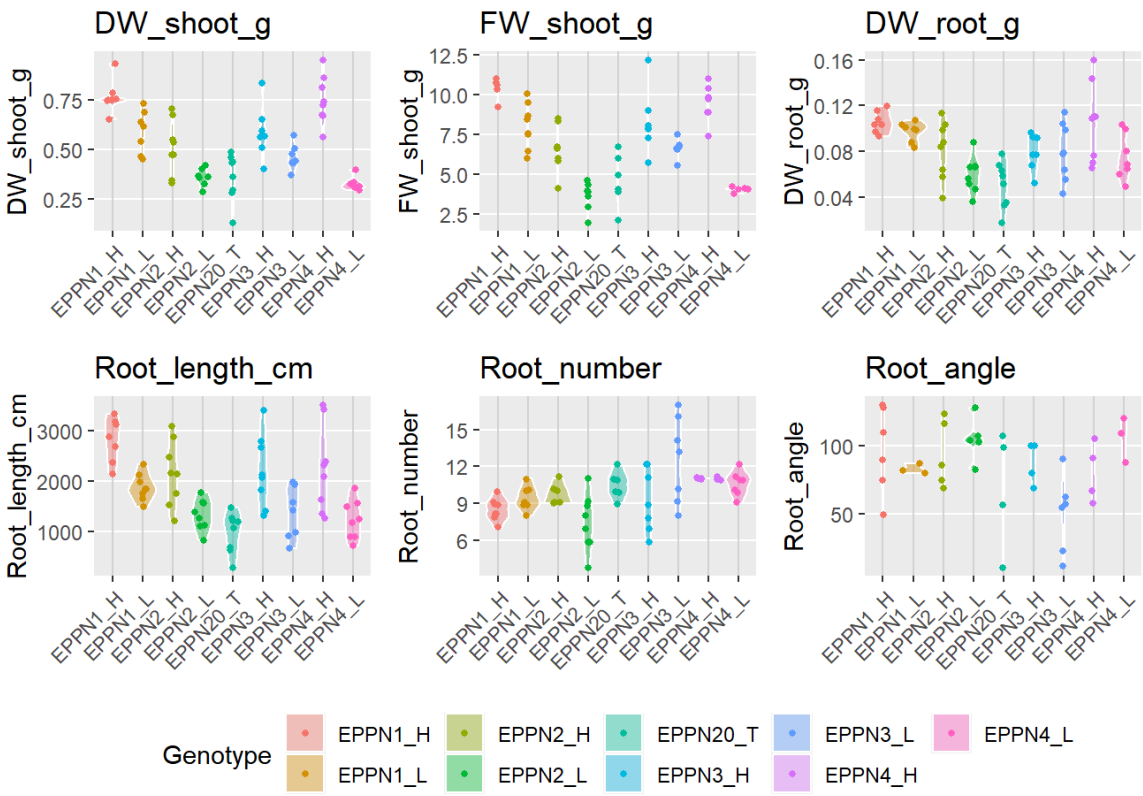
```
## Warning: Removed 1 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_sina()`).
```




```
create_violin_plots(endpoint_clean, variables, "Plant_type")
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 11 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 11 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 3 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_ydensity()`).
```

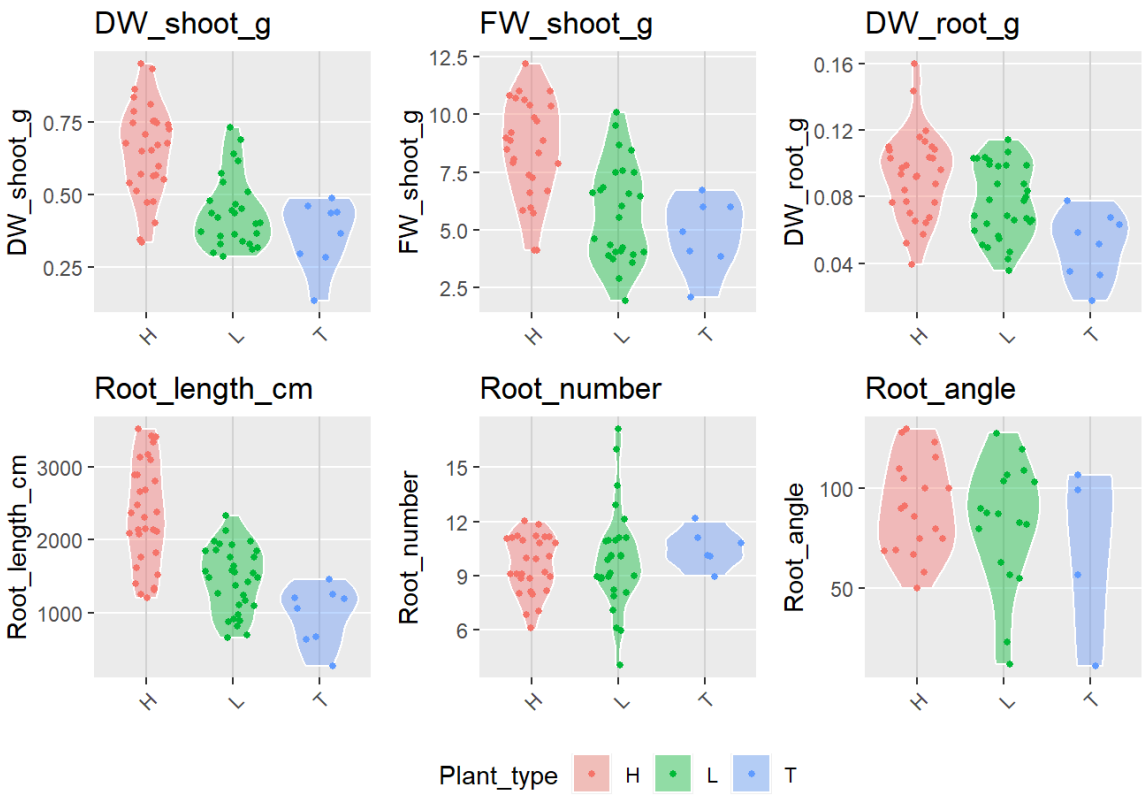
```
## Warning: Removed 1 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 9 rows containing non-finite values (`stat_sina()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 32 rows containing non-finite values (`stat_sina()`).
```



Exploration statistics for the variables after outlier detection

```
skim(endpoint_clean[variables])
```

Data summary

Name	endpoint_clean[variables]
Number of rows	72
Number of columns	6
_____	
Column type frequency:	
numeric	6
_____	

Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
DW_shoot_g	7	0.90	0.53	0.18	0.13	0.37	0.49	0.68	0.95	
FW_shoot_g	11	0.85	6.88	2.53	1.95	4.35	6.71	8.87	12.20	
DW_root_g	3	0.96	0.08	0.03	0.02	0.06	0.08	0.10	0.16	
Root_length_cm	1	0.99	1788.26	761.09	275.45	1247.28	1645.89	2144.60	3518.73	
Root_number	9	0.88	9.81	2.18	4.00	9.00	10.00	11.00	17.00	
Root_angle	32	0.56	84.67	29.53	11.00	68.50	87.50	105.50	130.00	

```
for (var in variables) {
  cat("\nSummary for:", var, "\n")
  endpoint_clean %>%
    group_by(Genotype) %>%
    summarize(mean    = mean(get(var), na.rm = TRUE),
              std.dev = sd(get(var), na.rm = TRUE),
              n_missing  = sum(is.na(get(var)))) %>%
    arrange(desc(mean)) %>%
    print(n = Inf)
}
```

```
##
## Summary for: DW_shoot_g
## # A tibble: 9 × 4
##   Genotype  mean std.dev n_missing
##   <fct>    <dbl>   <dbl>     <int>
## 1 EPPN1_H  0.768  0.0840         1
## 2 EPPN4_H  0.751  0.122          0
## 3 EPPN1_L  0.591  0.108          1
## 4 EPPN3_H  0.590  0.134          1
## 5 EPPN2_H  0.512  0.136          0
## 6 EPPN3_L  0.464  0.0642         1
## 7 EPPN20_T 0.362  0.119          0
## 8 EPPN2_L  0.360  0.0446         1
## 9 EPPN4_L  0.332  0.0361         2
##
## Summary for: FW_shoot_g
## # A tibble: 9 × 4
##   Genotype  mean std.dev n_missing
##   <fct>    <dbl>   <dbl>     <int>
## 1 EPPN1_H  10.4    0.639         2
## 2 EPPN4_H   9.44   1.18          1
## 3 EPPN3_H   8.28   2.00          1
## 4 EPPN1_L   8.03   1.42          0
## 5 EPPN3_L   6.62   0.631         2
## 6 EPPN2_H   6.27   1.65          0
## 7 EPPN20_T  4.82   1.59          1
## 8 EPPN4_L   4.03   0.176         3
## 9 EPPN2_L   3.60   0.909         1
##
## Summary for: DW_root_g
## # A tibble: 9 × 4
##   Genotype   mean std.dev n_missing
##   <fct>     <dbl>   <dbl>     <int>
## 1 EPPN1_H  0.106  0.00942         1
## 2 EPPN4_H  0.106  0.0342          0
## 3 EPPN1_L  0.0972 0.00860         1
## 4 EPPN2_H  0.0809 0.0254          0
## 5 EPPN3_L  0.0792 0.0251          0
## 6 EPPN3_H  0.0791 0.0159          1
## 7 EPPN4_L  0.0742 0.0188          0
## 8 EPPN2_L  0.0595 0.0158          0
## 9 EPPN20_T 0.0505 0.0202          0
##
## Summary for: Root_length_cm
## # A tibble: 9 × 4
##   Genotype  mean std.dev n_missing
##   <fct>    <dbl>   <dbl>     <int>
## 1 EPPN1_H  2816.    442.         1
## 2 EPPN4_H  2241.    866.         0
## 3 EPPN3_H  2200.    719.         0
## 4 EPPN2_H  2153.    654.         0
## 5 EPPN1_L  1875.    266.         0
## 6 EPPN3_L  1424.    522.         0
## 7 EPPN2_L  1321.    310.         0
## 8 EPPN4_L  1223.    393.         0
## 9 EPPN20_T  970.    400.         0
##
## Summary for: Root_number
## # A tibble: 9 × 4
##   Genotype  mean std.dev n_missing
##   <fct>    <dbl>   <dbl>     <int>
## 1 EPPN3_L  12.4    3.51          1
## 2 EPPN4_H  11      0             1
```

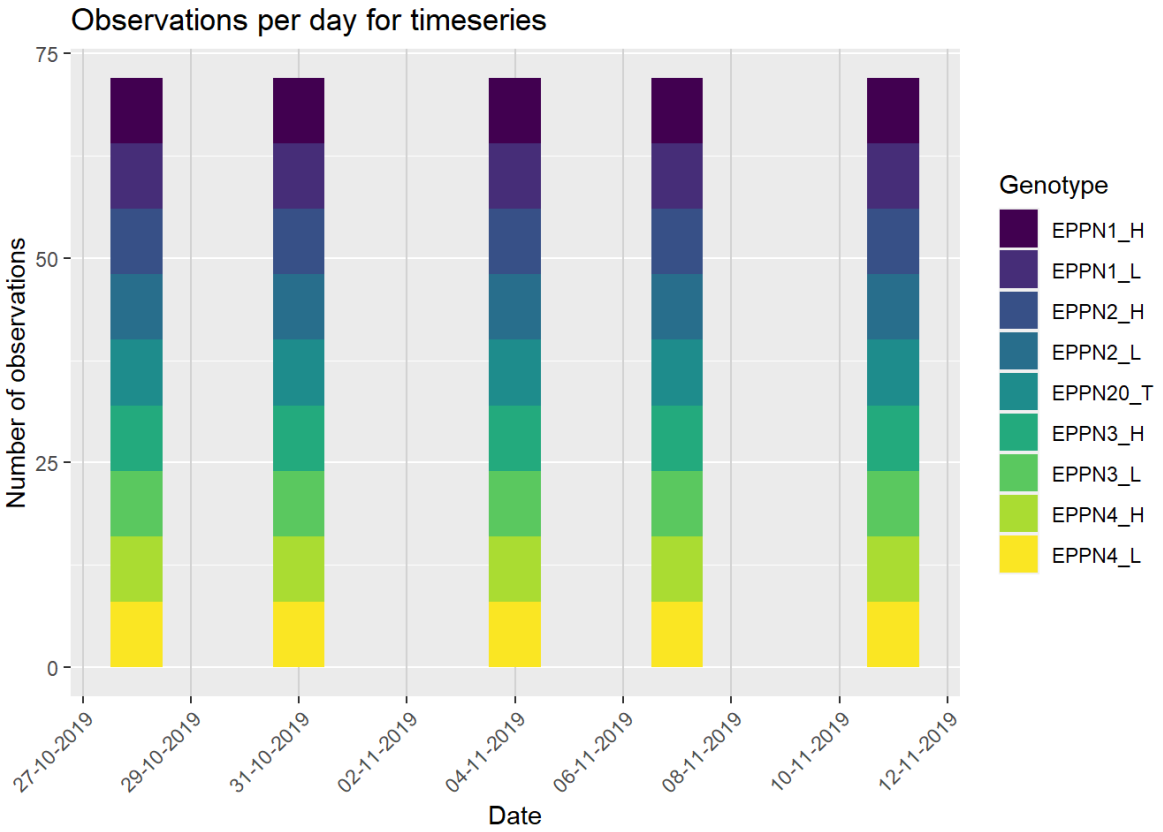
```
## 3 EPPN4_L  10.6    0.976         1
## 4 EPPN20_T 10.5    1.05          2
## 5 EPPN2_H   9.67   0.816         2
## 6 EPPN1_L   9.43   0.976         1
## 7 EPPN3_H   9.29   2.43          1
## 8 EPPN1_H   8.5    0.926         0
## 9 EPPN2_L   7.5    2.20          0
##
## Summary for: Root_angle
## # A tibble: 9 × 4
##   Genotype  mean std.dev n_missing
##   <fct>    <dbl>   <dbl>     <int>
## 1 EPPN4_L  106.    16.3          5
## 2 EPPN2_L  105     16.0          3
## 3 EPPN1_H   97.2   31.5          2
## 4 EPPN2_H   93.8   24.4          3
## 5 EPPN3_H   87.2   15.4          4
## 6 EPPN1_L   83      3.61         5
## 7 EPPN4_H   80.2   21.6          4
## 8 EPPN20_T  68.5   44.2          4
## 9 EPPN3_L   50     28.3          2
```

2. Exploration of the timeseries data

In this part, we look at the timeseries, S_timeseries and T_timeseries datasets, also using several functions, located in the functions.R script.

Number of data observations per day for the traits of the timeseries datasets

```
h1 <- ggplot(timeseries, aes(x = Date)) +
  geom_bar(aes(fill = Genotype), position = "stack", width = 0.96) +
  scale_fill_viridis_d(option = "D") +
  labs(x = "Date", y = "Number of observations", title = "Observations per day for timeseries") +
  scale_y_continuous(breaks = seq(from = 0, to = 325, by = 25)) +
  scale_x_date(date_breaks = "2 days", date_labels = "%d-%m-%Y") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        panel.grid.major.x = element_line(color = "lightgray", size = 0.5),
        panel.grid.minor.x = element_blank())

h1
```



A. Exploration of the timeseries dataframe

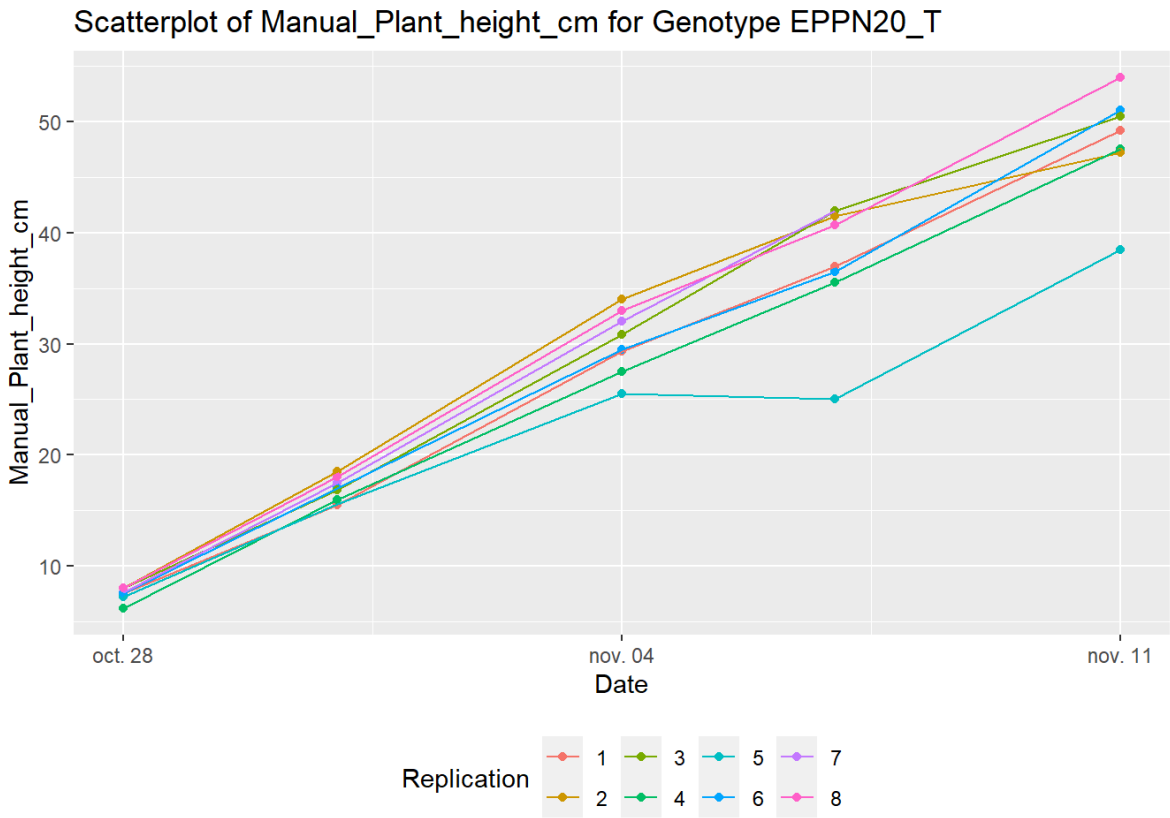
Scatter plots by Genotype

```
plot_scatter_by_genotype <- function(data, variables, genotype) {
  data_filtered <- data[data$Genotype == genotype, ]

  plots <- list()

  for (var in variables) {
    p <- ggplot(data_filtered, aes_string(x = "Date", y = var, group = "Replication", color = "factor
(Replication)")) +
      geom_point() +
      geom_line() +
      labs(title = paste("Scatterplot of", var, "for Genotype", genotype),
           x = "Date", y = var, color = "Replication") +
      theme(legend.position = "bottom")
    plots[[var]] <- p  # Ajouter le graphique à la liste
  }

  return(plots)  # Retourner la liste de graphiques
}
# Appeler la fonction pour chaque variable dans variables_t
for (var in variables_t) {
  plots <- plot_scatter_by_genotype(timeseries, var, "EPPN20_T")
  print(plots)
}
```

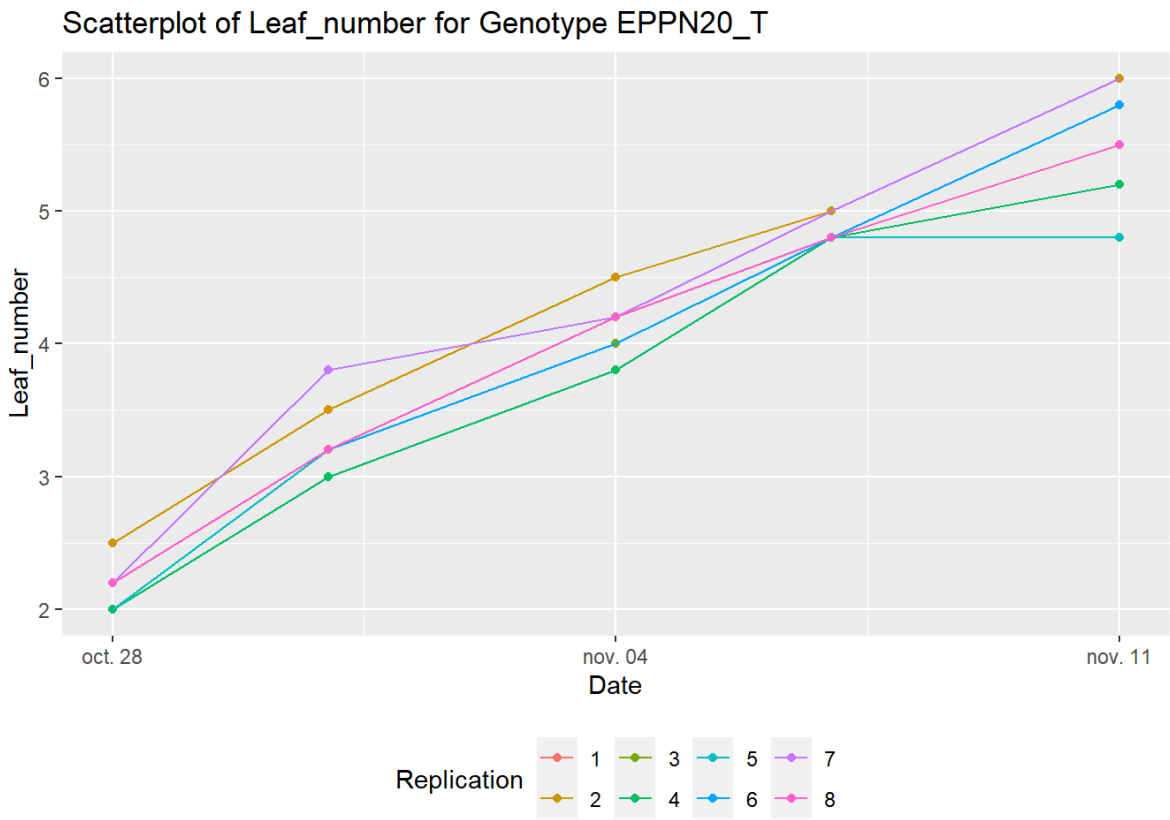
```
## $Manual_Plant_height_cm
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```




```
##
## $Leaf_number
```

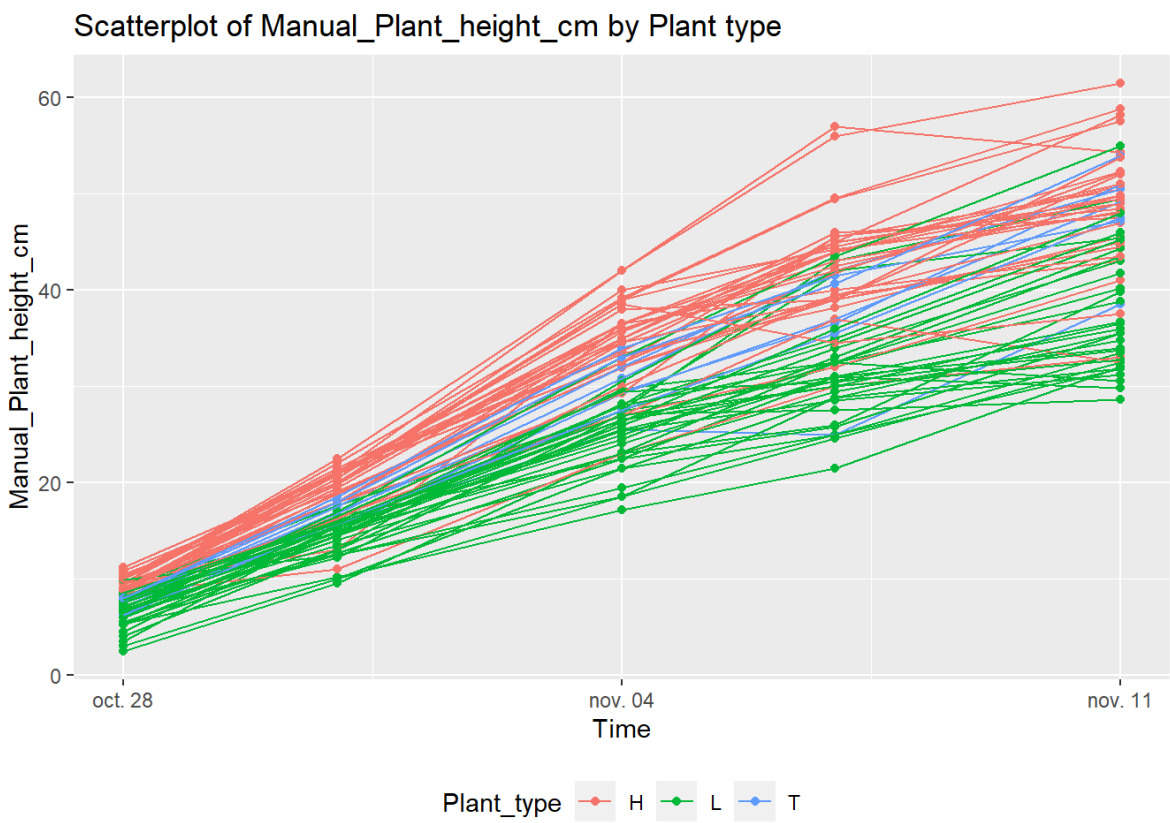


Scatterplots for all genotypes by Plant type (Hybride, Line, EPPN20_T) with smooth line.

```
plot_scatter_with_smooth <- function(data, variables) {
  for (var in variables) {
    p1 <- ggplot(data, aes_string(x = "Date", y = var, group = "Unit.ID", color = "factor(Plant_typ
e)")) +
      geom_point() +
      geom_line() +
      labs(title = paste("Scatterplot of", var, "by Plant type"),
           x = "Time", y = var, color = "Plant_type") +
      theme(legend.position = "bottom")

    p2 <- ggplot(data, aes_string(x = "Date", y = var, group = "Plant_type", color = "factor(Plant_ty
pe)")) +
      geom_smooth(method = "loess", se = FALSE) +
      labs(title = paste("Smooth line of", var, "by Plant type"),
           x = "Time", y = var, color = "Plant_type") +
      theme(legend.position = "bottom")

    print(p1)
    print(p2)
  }
}

plot_scatter_with_smooth(timeseries, variables_t)
```

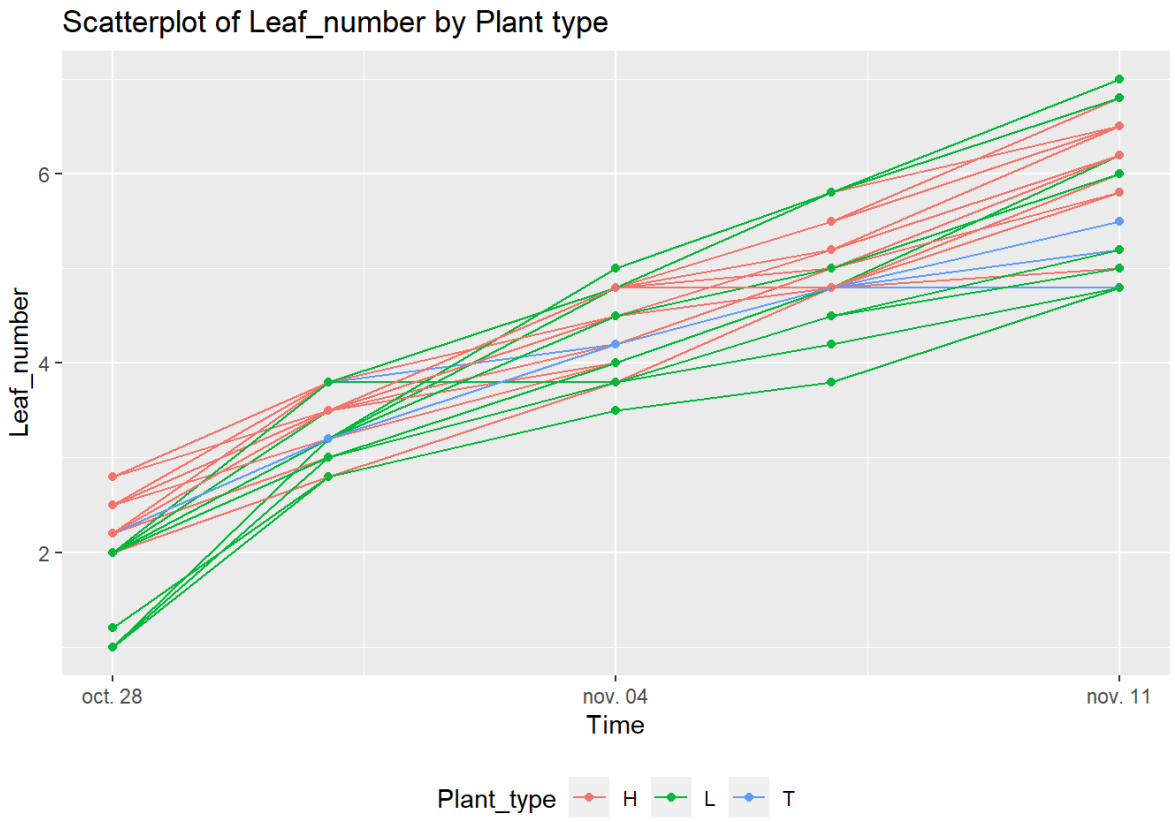
```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```

```
## Warning: Removed 2 rows containing missing values (`geom_line()`).
```

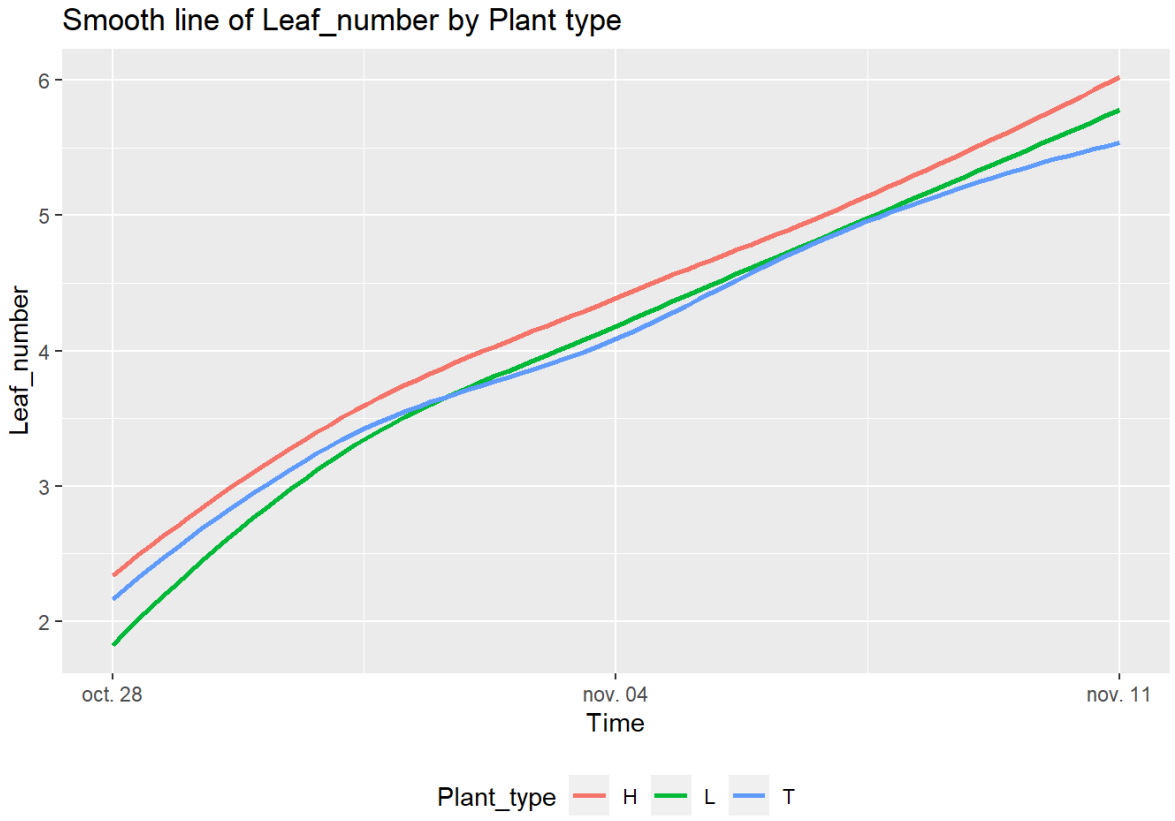



```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).
```




```
## `geom_smooth()` using formula = 'y ~ x'
```



Scatter plots for all genotypes by water treatment

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

B. Exploration of the S_timeseries dataframe

Scatter plots by Genotype

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

Scatterplots for all genotypes by Plant type (Hybride, Line, EPPN20_T) with smooth line.

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

Scatter plots for all genotypes by water treatment

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

C. Exploration of the T_timeseries dataframe

Scatter plots by Genotype

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

Scatterplots for all genotypes by Plant type (Hybride, Line, EPPN20_T) with smooth line.

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```

Scatter plots for all genotypes by water treatment

```
print(paste0("No data for", platform))
```

```
## [1] "No data forFZJ"
```