Statistical analysis of data from an hypocotyl measurement experiment in *A. thaliana* seedlings

CrisprCat

Contents

Introduction

This script describes the statistical analysis of an experiment in which hypocotyl length of 5 day old *A. thaliana* seedlings grown in darkness or in red light was determined. The hypocotyl length of 3 different mutants and wildtype seedlings were analysed. To identify statistical significant differences among the means of hypocotyl length of the mutants and treatments a Two-Way ANOVA is used.

The data used for this example is available under DOI: 10.1038/s41477-020-0725-0 and is the source data of Figure 1b.

Setup

Installing and loading required packages

```
# Store package names, required for the analysis in a vector
packages <- c("tidyverse", "car", "multcomp", "lsmeans", "multcompView")

# Install packages that are not yet installed
installed_packages <- packages %in% rownames(installed.packages())
if (any(installed_packages == FALSE)) {
```

```
  install.packages(packages[!installed_packages])
}

# Load packages
invisible(lapply(packages, library, character.only = TRUE))
```

Load the data

The data is usually stored in a .csv file with one column describing the "Gentoype", one column describing the "Treatment" and one describing the measured value "Hypocotyl_length".

```
# Read the .csv file
hypo <- read.csv(file = "hypocotyl_measurement.CSV", sep = ";", header = T)

# Change the class of Genotype and treatment from character to factor
hypo$Genotype <- as.factor(hypo$Genotype)
hypo$Treatment <- as.factor(hypo$Treatment)
```

Statistical analysis

Key assumptions for the Two-Way ANOVA are, that the data
* follows a normal distribution
* shows homogeneity of variances
* is balanced

As this biological data is independently and randomly sampled from a population I assume that these requirements are met. However, it is also possible to test for these assumptions.

Check if the data is balanced

```
# Generate a frequency table
table(hypo$Genotype, hypo$Treatment)
```

```
##
##             Dark Red
##   bbx20       20  23
##   bbx202122   21  22
##   bbx2122     22  23
##   WT          21  24
```

The data of the experiment has a balanced design (Roughly equal sample sizes in the different groups).

Test the assumption of equal Variances

With the Brown-Forsythe test the null hypothesis, that the variances of the analysed groups (hypocotyl length withn each genotype and treatment) are equal, is tested.

Brown-Forsythe test

The Brown-Forsythe test is more robust to not gaussian distributed data than the Levene's test.

```
# Test for homoscedasticity
leveneTest(hypo$Hypocotyl_length ~ hypo$Genotype * hypo$Treatment, center = median)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   7  1.4069 0.2055
##       168
```

As the p-value (Pr) is > 0.05, the null hypothesis cannot be rejected.

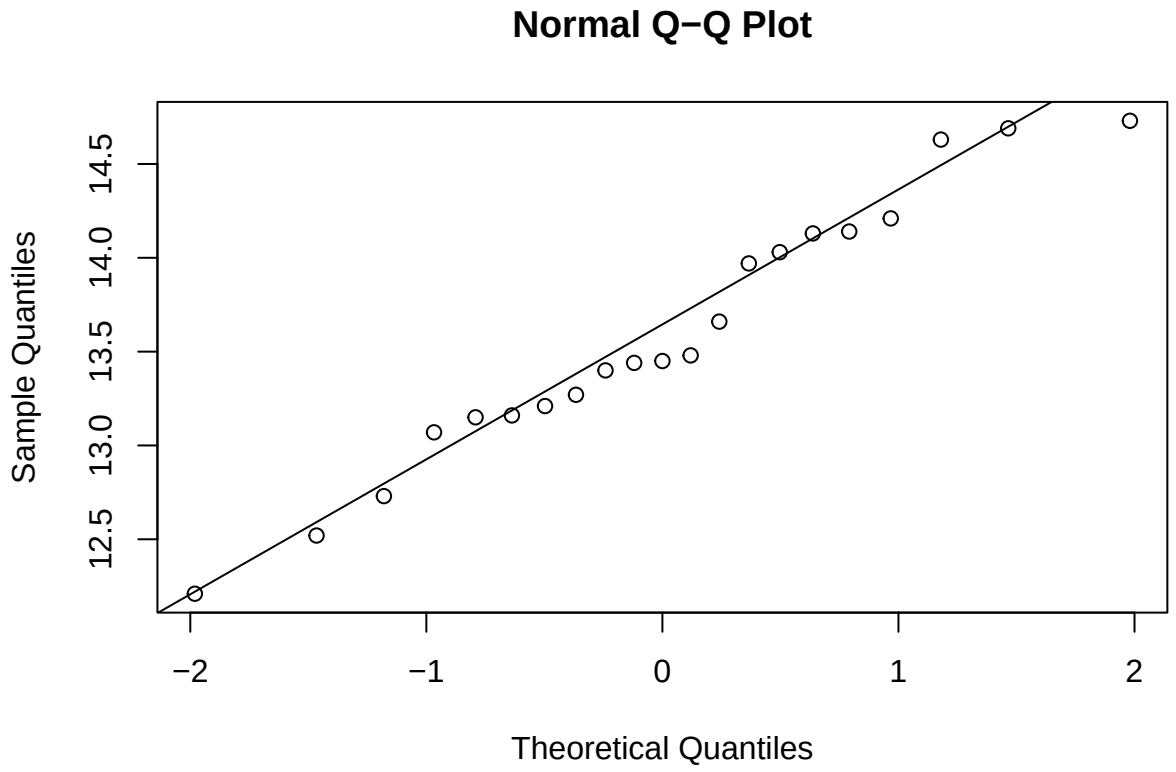
Test the assumption of normal distribution

The assumption of normal distribution of the data in each group = (genotype and treatment) divided by treatment can be graphically tested with a QQ-plot or with the Shapiro-Wilk's test.

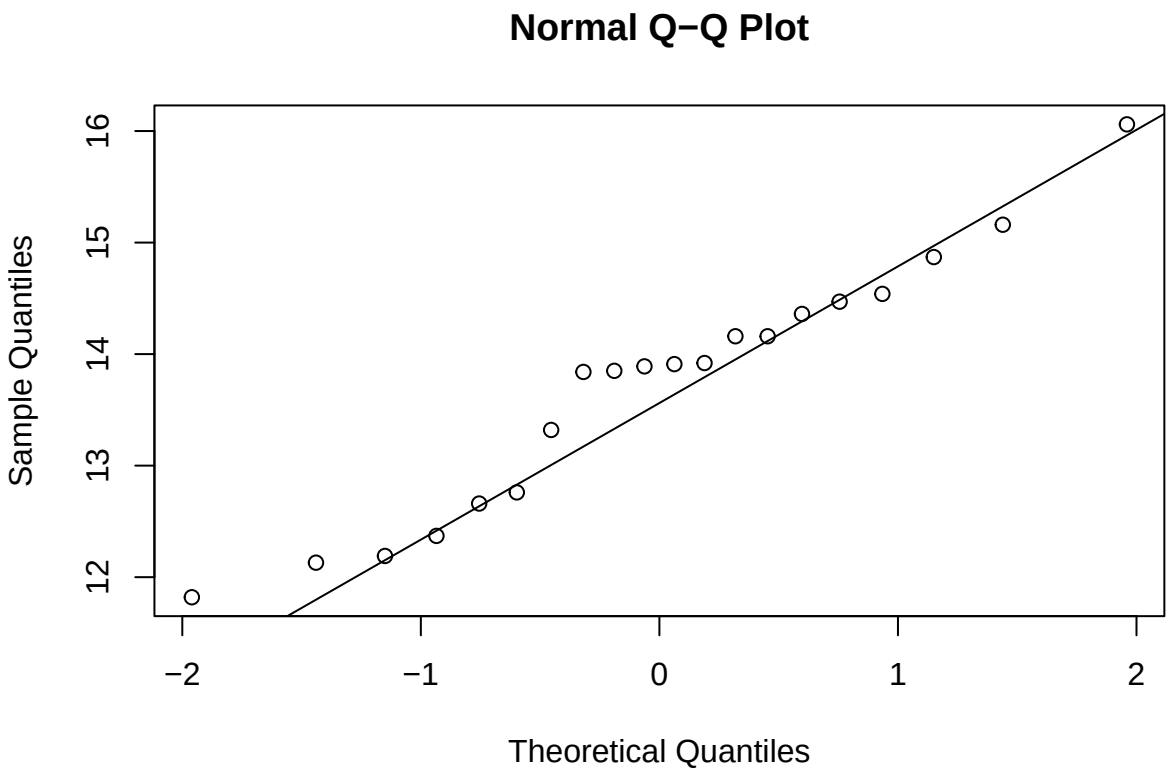
QQ-plot

When the points in the QQ-plot show linearity it suggests the data is normally distributed.

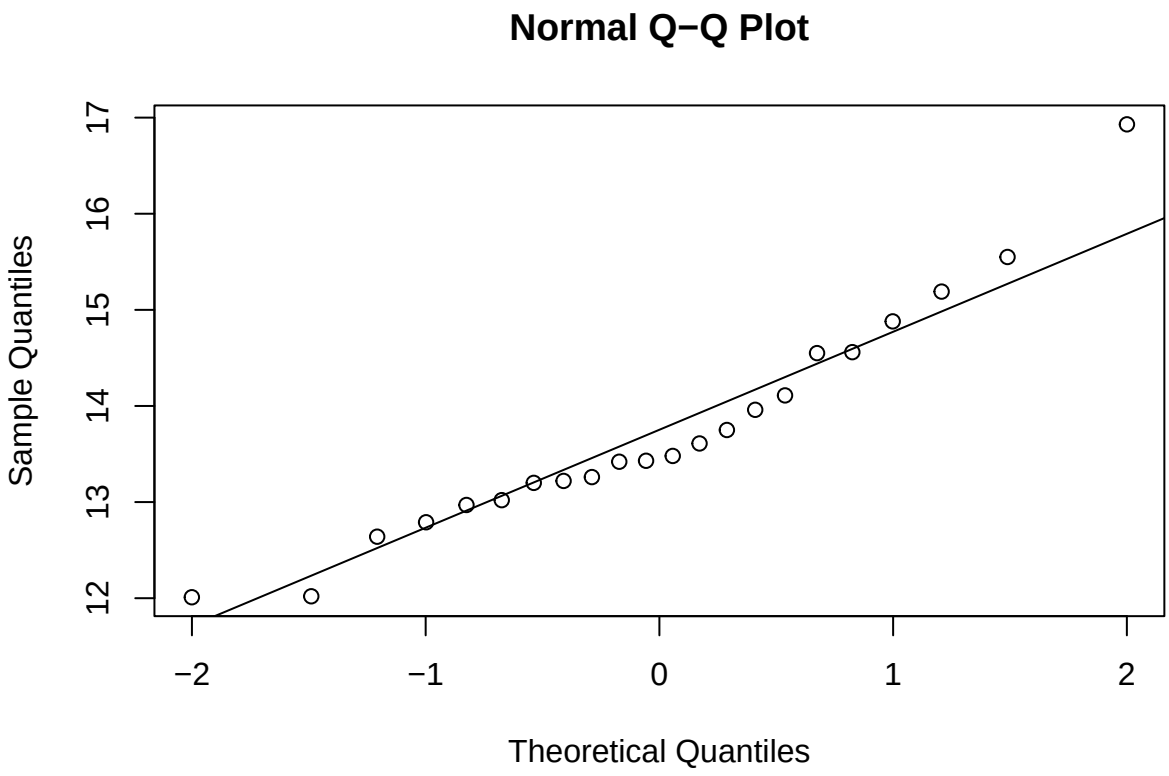
```
# Create a QQ-plot for WT hypocotyl length, seedlings grown in darkness
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Dark'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Dark'])
```



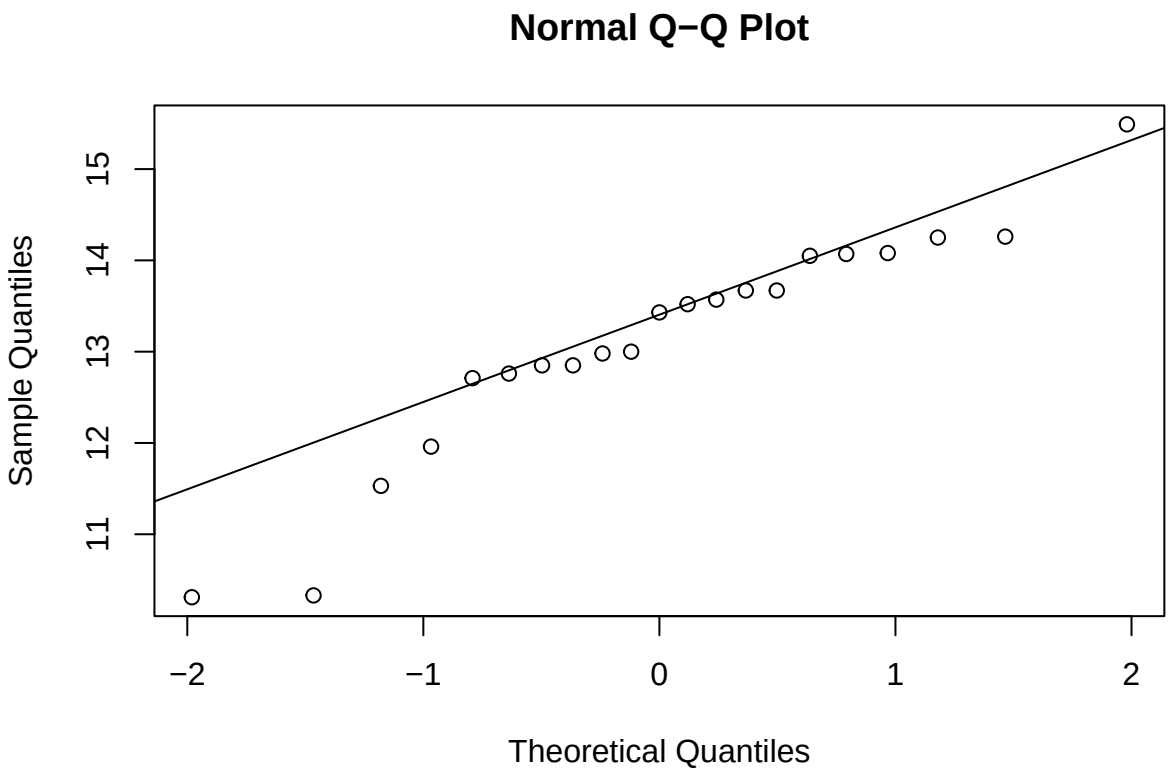

```
# Create a QQ-plot for *bbx20* hypocotyl length, seedlings grown in darkness
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Dark'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Dark'])
```



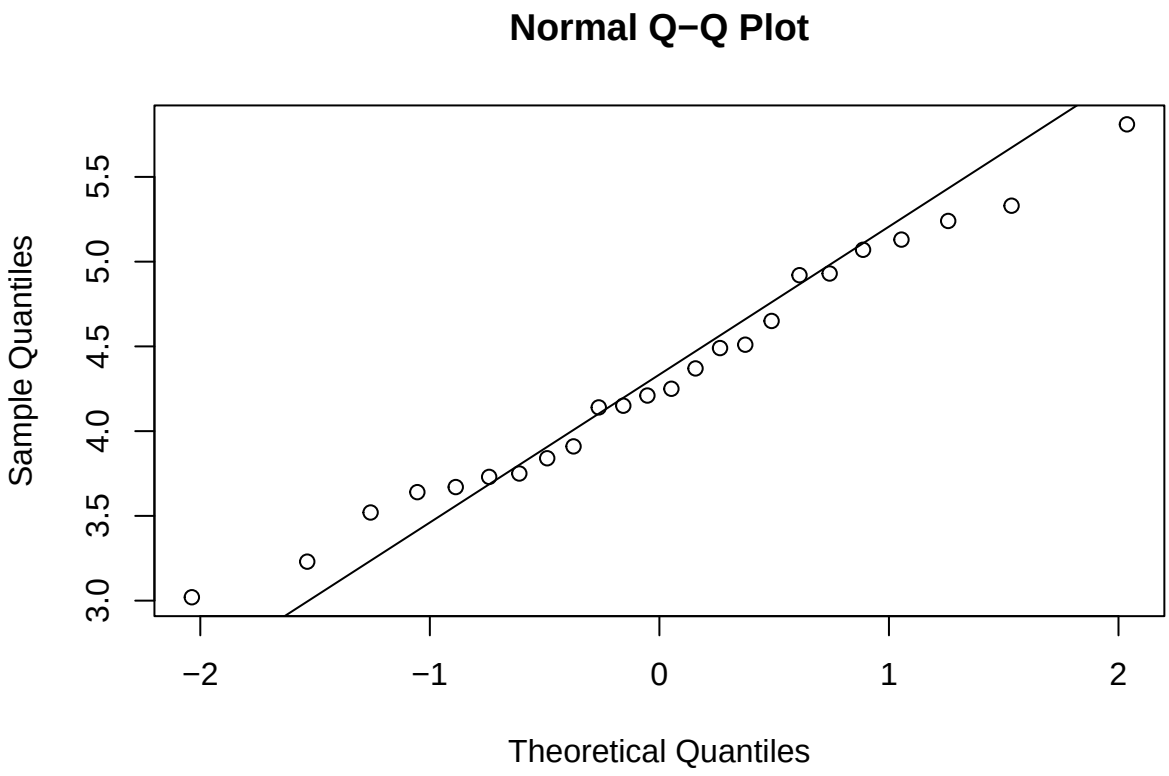

```
# Create a QQ-plot for *bbx2122* hypocotyl length, seedlings grown in darkness
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Dark'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Dark'])
```



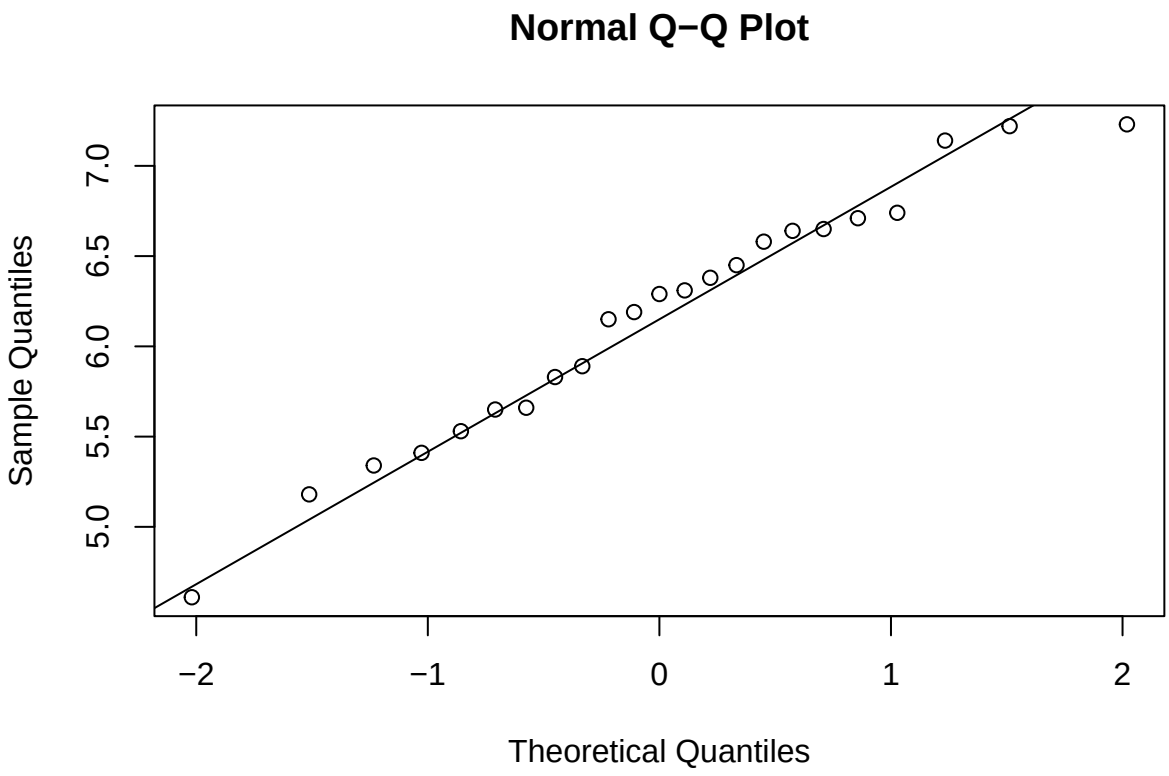

```
# Create a QQ-plot for *bbx202122* hypocotyl length, seedlings grown in darkness
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Dark'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Dark'])
```



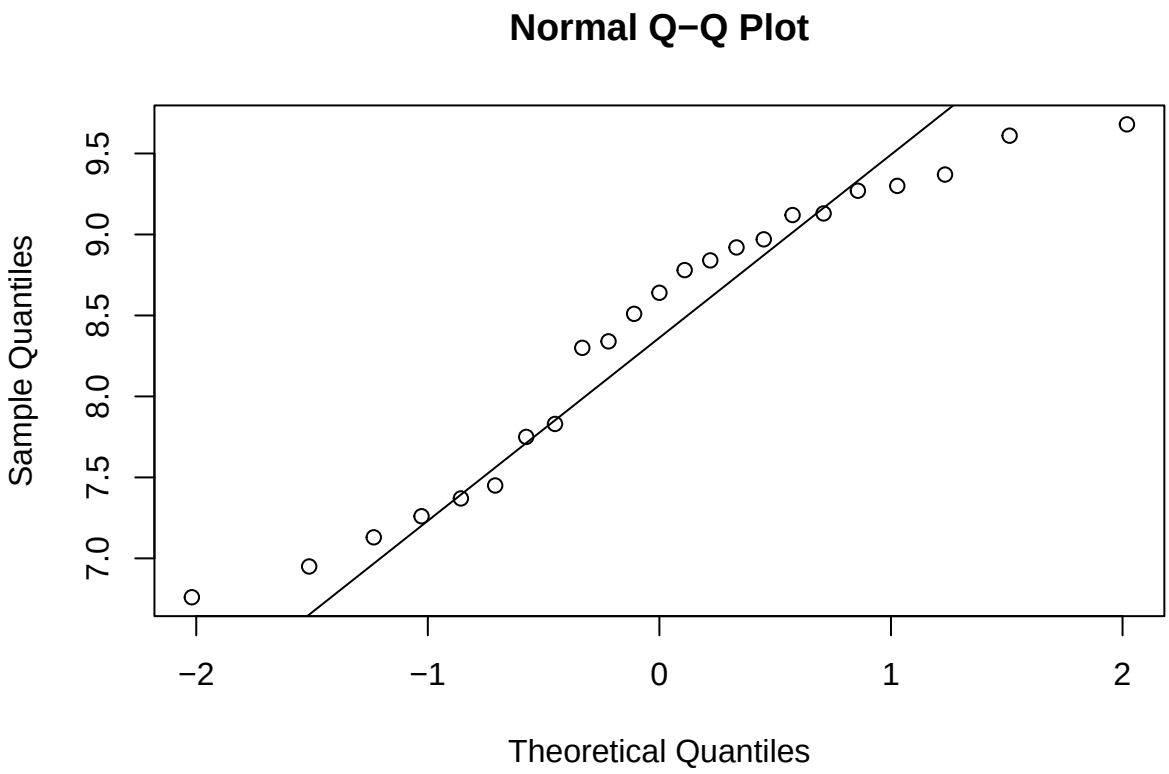

```
# Create a QQ-plot for WT hypocotyl length, seedlings grown in red light
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Red'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Red'])
```



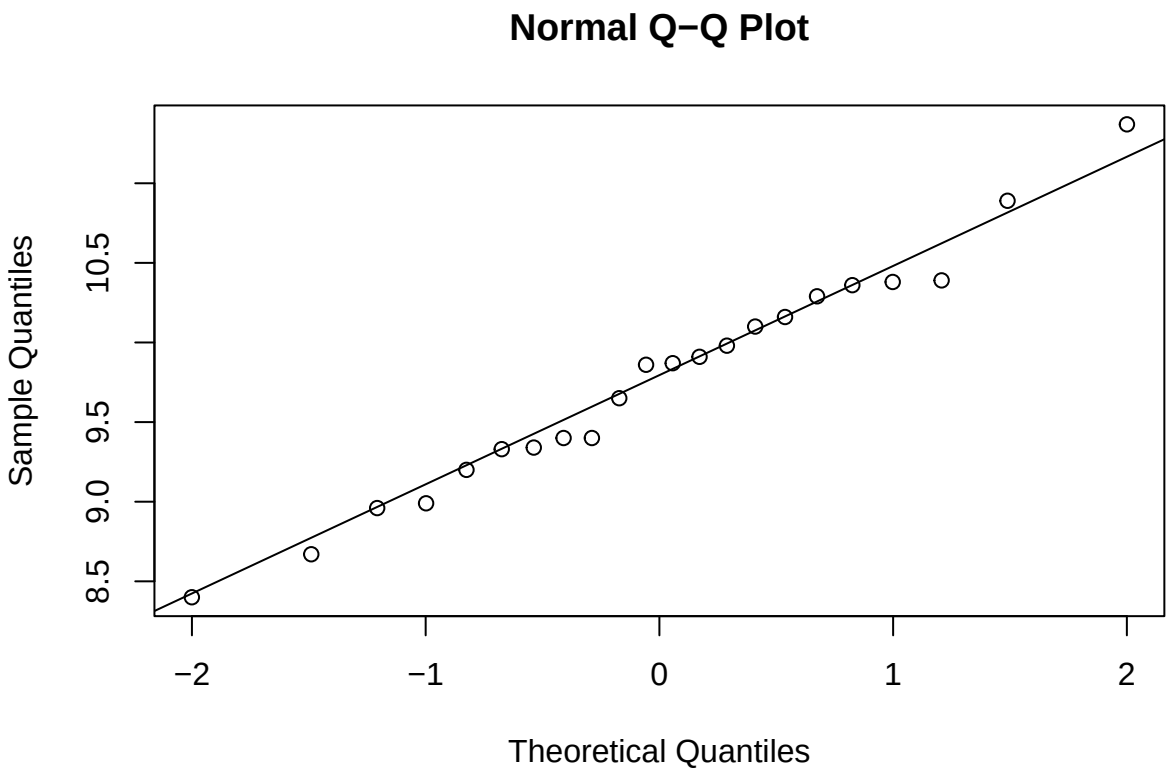

```
# Create a QQ-plot for *bbx20* hypocotyl length, seedlings grown in red light
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Red'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Red'])
```




```
# Create a QQ-plot for *bbx2122* hypocotyl length, seedlings grown in red light
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Red'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Red'])
```




```
# Create a QQ-plot for *bbx202122* hypocotyl length, seedlings grown in red light
qqnorm(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Red'])
qqline(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Red'])
```



Shapiro-Wilk's test

With the Shapiro-Wilk's test the null hypothesis, that the the samples in each analysed group (hypocotyl length in each genotype, divided by treatment) follow a normal distribution is tested.

```
# Test for normal distribution
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Dark'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "WT" & hypo$Treatment == "Dark"]
## W = 0.96649, p-value = 0.6548
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Dark'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx20" & hypo$Treatment == "Dark"]
## W = 0.95798, p-value = 0.5044
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Dark'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx2122" & hypo$Treatment == "Dark"]
```

```
## W = 0.93892, p-value = 0.1881
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Dark'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx202122" & hypo$Treatment == "Dark"]
## W = 0.92093, p-value = 0.09059
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'WT' & hypo$Treatment == 'Red'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "WT" & hypo$Treatment == "Red"]
## W = 0.98063, p-value = 0.9071
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx20' & hypo$Treatment == 'Red'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx20" & hypo$Treatment == "Red"]
## W = 0.97042, p-value = 0.699
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx2122' & hypo$Treatment == 'Red'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx2122" & hypo$Treatment == "Red"]
## W = 0.9292, p-value = 0.1051
```

```
shapiro.test(hypo$Hypocotyl_length[hypo$Genotype == 'bbx202122' & hypo$Treatment == 'Red'])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hypo$Hypocotyl_length[hypo$Genotype == "bbx202122" & hypo$Treatment == "Red"]
## W = 0.98483, p-value = 0.9738
```

As the p-value (Pr) is > 0.05, the null hypothesis cannot be rejected.

Two way ANOVA with interaction effect

With the Two-Way ANOVA the null hypothesis, that there are no statistical significant differences among the means of the hypocotyl length of the mutants divided by treatment, is tested.

```
# Fit an Analysis of Variance model
res.aov <- aov(data = hypo, Hypocotyl_length ~ Genotype * Treatment)

# Show the results of the fitted model
summary(res.aov)
```

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## Genotype             3  213.3    71.1   82.40 <2e-16 ***
## Treatment            1 1796.1  1796.1 2082.01 <2e-16 ***
## Genotype:Treatment   3  219.5    73.2   84.82 <2e-16 ***
```

```
## Residuals          168  144.9     0.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As the p-value (Pr) is > 0.05, the null hypothesis cannot be rejected.

Multiple comparisons

To determine if the means of specific groups are statistically significant different from each other the Tukey HSD post-hoc test is computed.

```
# Perform the post-hoc test
TukeyHSD(res.aov, conf.level = 0.95)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Hypocotyl_length ~ Genotype * Treatment, data = hypo)
##
## $Genotype
##                          diff        lwr        upr     p adj
## bbx202122-bbx20     1.7213953  1.2015980  2.2411927 0.0000000
## bbx2122-bbx20       1.3389096  0.8249202  1.8528989 0.0000000
## WT-bbx20           -1.0397571 -1.5537465 -0.5257678 0.0000027
## bbx2122-bbx202122 -0.3824858 -0.8964751  0.1315036 0.2190509
## WT-bbx202122       -2.7611525 -3.2751418 -2.2471631 0.0000000
## WT-bbx2122         -2.3786667 -2.8867817 -1.8705517 0.0000000
##
## $Treatment
##               diff       lwr       upr p adj
## Red-Dark -6.394089 -6.670808 -6.117371     0
##
## $`Genotype:Treatment`
##                                      diff         lwr        upr     p adj
## bbx202122:Dark-bbx20:Dark     -0.61057143  -1.5013414  0.2801986 0.4166722
## bbx2122:Dark-bbx20:Dark        0.03027273  -0.8505664  0.9111119 1.0000000
## WT:Dark-bbx20:Dark            -0.13723810  -1.0280081  0.7535319 0.9997578
## bbx20:Red-bbx20:Dark          -7.55765217  -8.4293252 -6.6859791 0.0000000
## bbx202122:Red-bbx20:Dark      -3.95381818  -4.8346573 -3.0729790 0.0000000
## bbx2122:Red-bbx20:Dark        -5.31852174  -6.1901948 -4.4468487 0.0000000
## WT:Red-bbx20:Dark             -9.40908333 -10.2722686 -8.5458981 0.0000000
## bbx2122:Dark-bbx202122:Dark    0.64084416  -0.2289401  1.5106284 0.3212022
## WT:Dark-bbx202122:Dark         0.47333333  -0.4065066  1.3531732 0.7182755
## bbx20:Red-bbx202122:Dark      -6.94708075  -7.8075812 -6.0865803 0.0000000
## bbx202122:Red-bbx202122:Dark -3.34324675  -4.2130310 -2.4734625 0.0000000
## bbx2122:Red-bbx202122:Dark    -4.70795031  -5.5684507 -3.8474499 0.0000000
## WT:Red-bbx202122:Dark         -8.79851190  -9.6504132 -7.9466106 0.0000000
## WT:Dark-bbx2122:Dark          -0.16751082  -1.0372951  0.7022734 0.9989479
## bbx20:Red-bbx2122:Dark        -7.58792490  -8.4381410 -6.7377088 0.0000000
## bbx202122:Red-bbx2122:Dark    -3.98409091  -4.8437019 -3.1244799 0.0000000
## bbx2122:Red-bbx2122:Dark      -5.34879447  -6.1990105 -4.4985784 0.0000000
## WT:Red-bbx2122:Dark           -9.43935606 -10.2808680 -8.5978442 0.0000000
## bbx20:Red-WT:Dark             -7.42041408  -8.2809145 -6.5599137 0.0000000
## bbx202122:Red-WT:Dark         -3.81658009  -4.6863644 -2.9467958 0.0000000
## bbx2122:Red-WT:Dark           -5.18128364  -6.0417841 -4.3207832 0.0000000
```

```
## WT:Red-WT:Dark               -9.27184524 -10.1237466 -8.4199439 0.0000000
## bbx202122:Red-bbx20:Red       3.60383399   2.7536179  4.4540501 0.0000000
## bbx2122:Red-bbx20:Red         2.23913043   1.3984143  3.0798466 0.0000000
## WT:Red-bbx20:Red             -1.85143116  -2.6833438 -1.0195186 0.0000000
## bbx2122:Red-bbx202122:Red    -1.36470356  -2.2149196 -0.5144875 0.0000537
## WT:Red-bbx202122:Red         -5.45526515  -6.2967771 -4.6137533 0.0000000
## WT:Red-bbx2122:Red           -4.09056159  -4.9224742 -3.2586490 0.0000000
```

When p adj value is < 0.05 the null hypothesis, that the means of the two compared groups are not statistically significant different is rejected.

To assign letters to each group, that indicate groups that are statistically significant different from each other the cld() function is used.

```
# Compute least-square means
lsm = lsmeans(res.aov, ~ Genotype + Treatment)

# Create a cld (compact letter display) object of the pairwise comparisons
CLD = cld(lsm, Letters = letters, adjust = "sidak", sort = TRUE, reversed = TRUE)
CLD$.group=gsub(" ", "", CLD$.group)

# Display the letters assigned to each group
print(CLD)
```

```
##  Genotype  Treatment lsmean    SE  df lower.CL upper.CL .group
##  bbx2122   Dark       13.75 0.198 168    13.21    14.30 a
##  bbx20     Dark       13.72 0.208 168    13.15    14.30 a
##  WT        Dark       13.58 0.203 168    13.03    14.14 a
##  bbx202122 Dark       13.11 0.203 168    12.55    13.67 a
##  bbx202122 Red         9.77 0.198 168     9.22    10.32 b
##  bbx2122   Red         8.40 0.194 168     7.87     8.94 c
##  bbx20     Red         6.16 0.194 168     5.63     6.70 d
##  WT        Red         4.31 0.190 168     3.79     4.84 e
##
## Confidence level used: 0.95
## Conf-level adjustment: sidak method for 8 estimates
## P value adjustment: sidak method for 28 tests
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping letter,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

Data visualization

Calculate summary statistics

These summary statistics are used to create a meaningful data visualization of the experiments result.

```
# Calculate summary statistics
hypo_summary = hypo %>%
  group_by(Genotype, Treatment) %>%
  summarise(mean_hypo = mean(Hypocotyl_length), # Calculate the mean of the hypocotyl
                                                # length for each genotype/treatment
            sd_hypo = sd(Hypocotyl_length), # Calculate the standard deviation of
                                            # hypocotyl length for each genotype/treatment
            n_hypo = n(), # Calculate the number of measurements of hypocotyl length
```

```
                        # for each genotype
            SE_hypo = sd_hypo / sqrt(n()), # Calculate the standard error of the mean of
                                           # hypocotyl length for each genotype/treatment
            max_hypo = max(Hypocotyl_length)) %>% # Define the maximum of hypocotyl
                                                  # length for each genotype/treatment
  arrange(desc(mean_hypo)) # sort the data by the mean in descending order
```

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

```
# Add a column with the letters indicating statistical significant different groups
hypo_summary$diff = print(CLD$.group)
```

```
## [1] "a" "a" "a" "a" "b" "c" "d" "e"
```

```
# Display the summary statistics
print(hypo_summary)
```

```
## # A tibble: 8 x 8
## # Groups:   Genotype [4]
##   Genotype  Treatment mean_hypo sd_hypo n_hypo SE_hypo max_hypo diff
##   <fct>     <fct>         <dbl>   <dbl>  <int>   <dbl>    <dbl> <chr>
## 1 bbx2122   Dark           13.8    1.17      22   0.249    16.9  a
## 2 bbx20     Dark           13.7    1.11      20   0.249    16.1  a
## 3 WT        Dark           13.6    0.696     21   0.152    14.7  a
## 4 bbx202122 Dark           13.1    1.26      21   0.276    15.5  a
## 5 bbx202122 Red             9.77   0.720     22   0.153    11.4  b
## 6 bbx2122   Red             8.40   0.907     23   0.189     9.68 c
## 7 bbx20     Red             6.16   0.687     23   0.143     7.23 d
## 8 WT        Red             4.31   0.715     24   0.146     5.81 e
```

Create a plot and save it as a pdf

```
# Save the graph as a pdf
pdf("hypocotyl_measurement.pdf", width = 4 , height = 6)
# Create a plot of the hypocotyl length vs. Genotype grouped by treatment
hypo_plot = ggplot(data = hypo,
                   aes(x = Genotype, y = Hypocotyl_length, fill = Treatment)) +
  # Create a box plot and customize its appearance
  stat_boxplot(geom = 'errorbar',
               width = 0.25,
               position = position_dodge(0.75)) +
  geom_boxplot(stat = "boxplot",
               aes(y = Hypocotyl_length)) +
  scale_x_discrete(limits=c("WT", "bbx20", "bbx2122", "bbx202122")) +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90,
                                   size=12,
                                   hjust = 1,
                                   vjust = 0.5,
                                   face = c("plain", "italic", "italic", "italic")),
        axis.title.y = element_text(size = 12)) +
  theme(axis.ticks = element_line()) +
  ylab("Hypocotyl length (mm)") +
  xlab(NULL) +
```

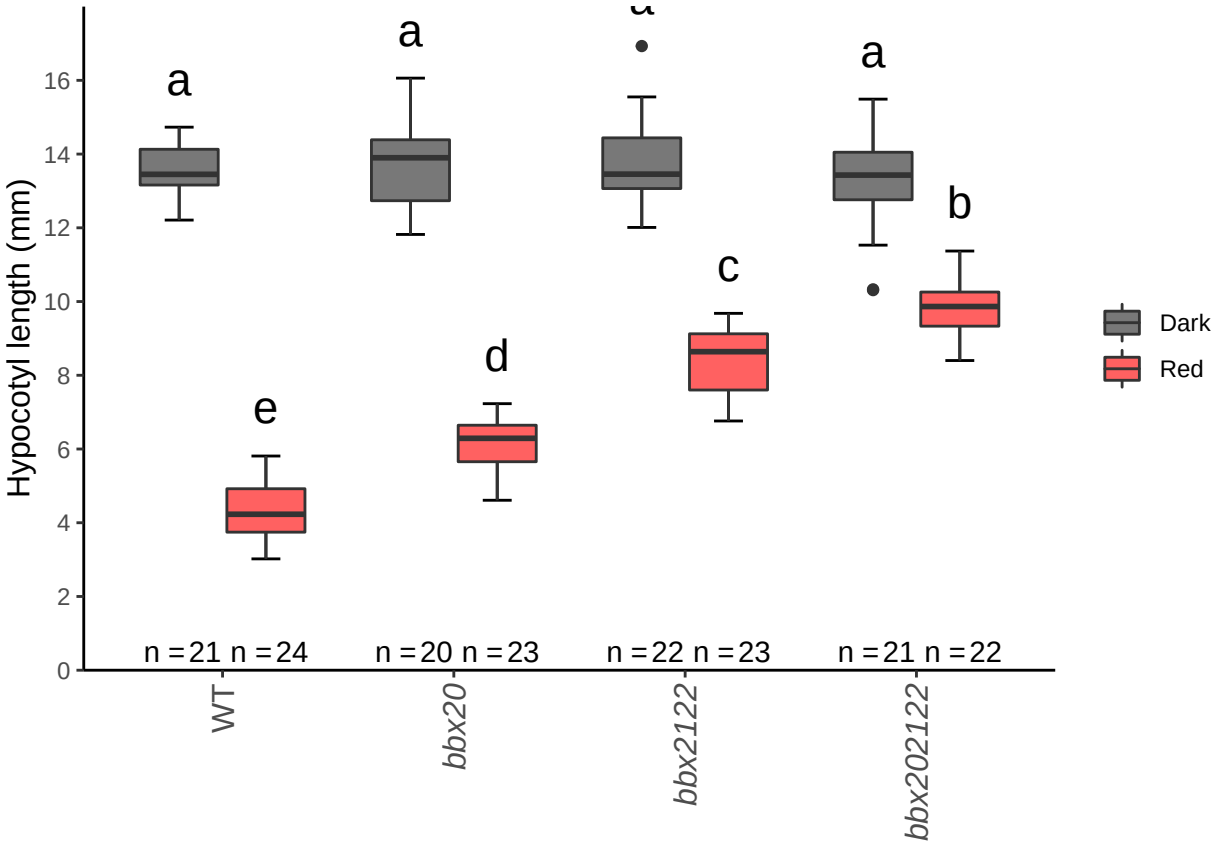
```
  scale_y_continuous(breaks = c(0,2,4,6,8,10,12,14,16),
                     expand = c(0,0),
                     limits = c(0,18)) +
  geom_text(data = hypo_summary,
            aes(y = max_hypo, label = diff),
            position = position_dodge(0.75),
            vjust = -1,
            size=6) +
  geom_text(data = hypo_summary,
            aes(y = 0.5, label = n_hypo),
            position = position_dodge(0.75),
            hjust = -0.3,
            size = 4) +
  geom_text(data = hypo_summary,
            aes(y = 0.5, label = 'n ='),
            position = position_dodge(0.75),
            hjust = 0.85,
            size = 4) +
  scale_fill_manual(values = c("#787878", "#FF6161")) +
  theme(legend.title = element_blank())
```

```
## Warning: Vectorized input to `element_text()` is not officially supported.
## Results may be unexpected or may change in future versions of ggplot2.
```

```
hypo_plot
dev.off()
```

```
## pdf
##   2
```

```
hypo_plot
```


Hypocotyl length (mm)
0
2
4
6
8
10
12
14
16
a
e
a
d
a
c
a
b
n = 21 n = 24
n = 20 n = 23
n = 22 n = 23
n = 21 n = 22
WT
bbx20
bbx2122
bbx202122
Dark
Red