helicopter project

2025-03-13

```
# Load the data
data <- read.csv('~/Desktop/MAS31004 project excel.csv')
```

```
#Print the column names
names(data)
```

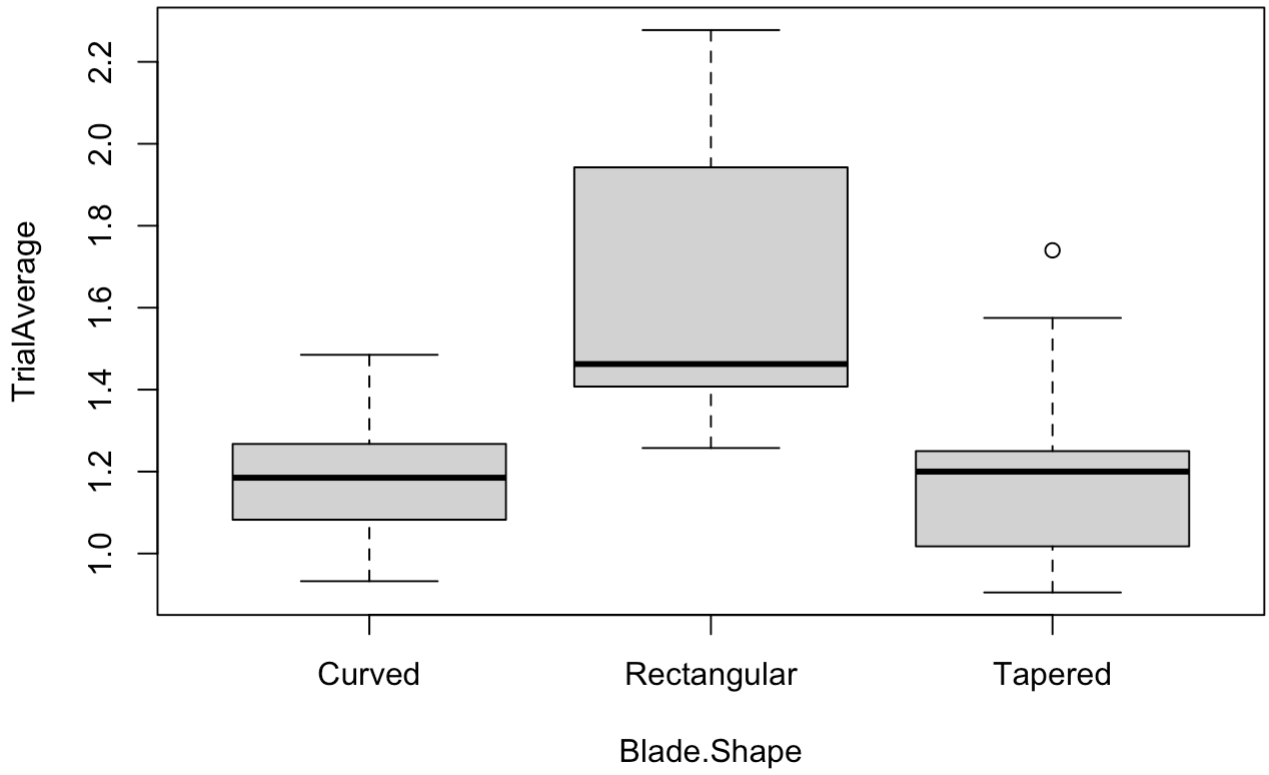
```
## [1] "Run.Number"        "Blade.Shape"       "Blade.Length..cm."
## [4] "Added.Weight..g."  "Trial.1..s."       "Trial.2..s."
## [7] "Trial.3..s."       "Trial.4..s."       "random.order"
```

```
# Calculate Trial Average
data$TrialAverage <- rowMeans(data[, c( "Trial.1..s.",  "Trial.2..s.",  "Trial.3..
s.",  "Trial.4..s.")])
```

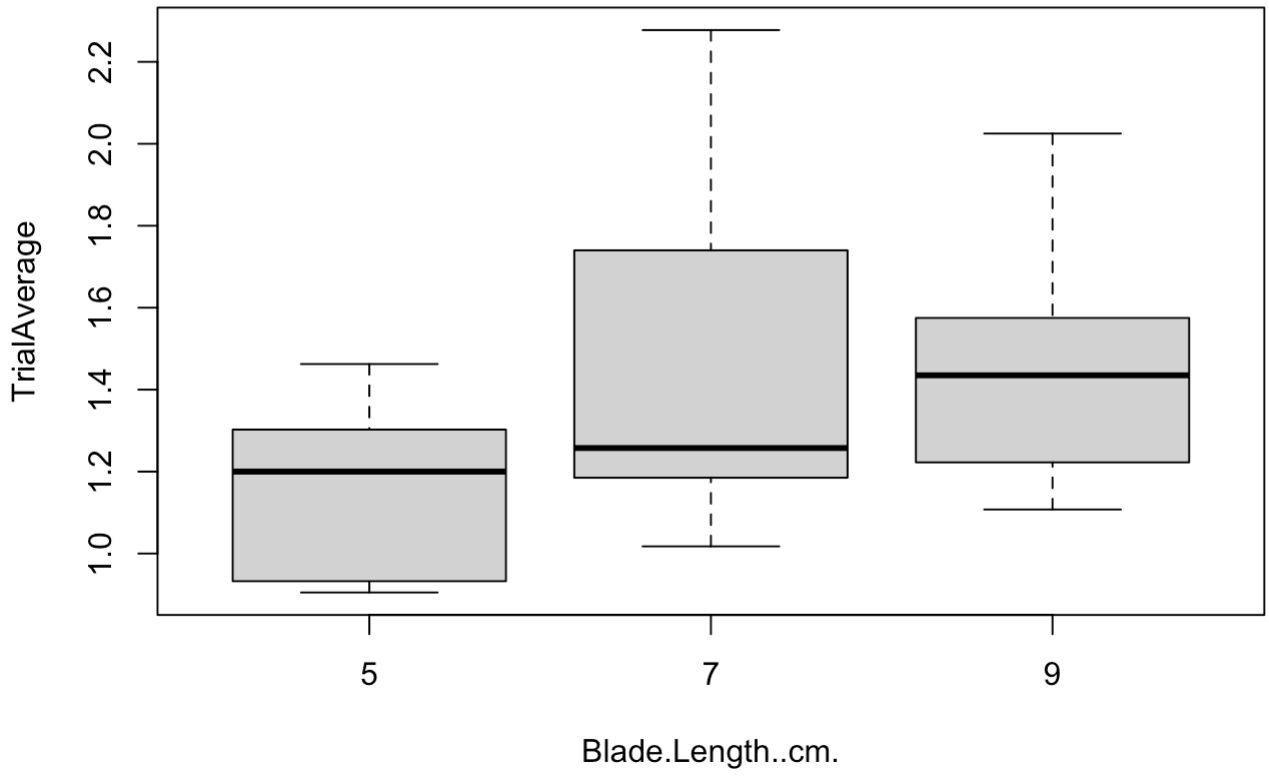
```
# Descriptive Statistics
summary(data)
```

```
##    Run.Number   Blade.Shape        Blade.Length..cm. Added.Weight..g.
##  Min.   : 1.0   Length:27          Min.   :5         Min.   :0.0
##  1st Qu.: 7.5   Class :character   1st Qu.:5         1st Qu.:0.0
##  Median :14.0   Mode  :character   Median :7         Median :0.5
##  Mean   :14.0                      Mean   :7         Mean   :0.5
##  3rd Qu.:20.5                      3rd Qu.:9         3rd Qu.:1.0
##  Max.   :27.0                      Max.   :9         Max.   :1.0
##   Trial.1..s.     Trial.2..s.     Trial.3..s.     Trial.4..s.
##  Min.   :0.830   Min.   :0.850   Min.   :0.790   Min.   :0.910
##  1st Qu.:1.090   1st Qu.:1.095   1st Qu.:1.155   1st Qu.:1.130
##  Median :1.240   Median :1.180   Median :1.290   Median :1.250
##  Mean   :1.327   Mean   :1.284   Mean   :1.434   Mean   :1.366
##  3rd Qu.:1.500   3rd Qu.:1.430   3rd Qu.:1.645   3rd Qu.:1.560
##  Max.   :2.670   Max.   :2.040   Max.   :2.340   Max.   :2.400
##   random.order       TrialAverage
##  Min.   :0.008864   Min.   :0.905
##  1st Qu.:0.160003   1st Qu.:1.119
##  Median :0.299235   Median :1.260
##  Mean   :0.367376   Mean   :1.353
##  3rd Qu.:0.533908   3rd Qu.:1.474
##  Max.   :0.996367   Max.   :2.277
```

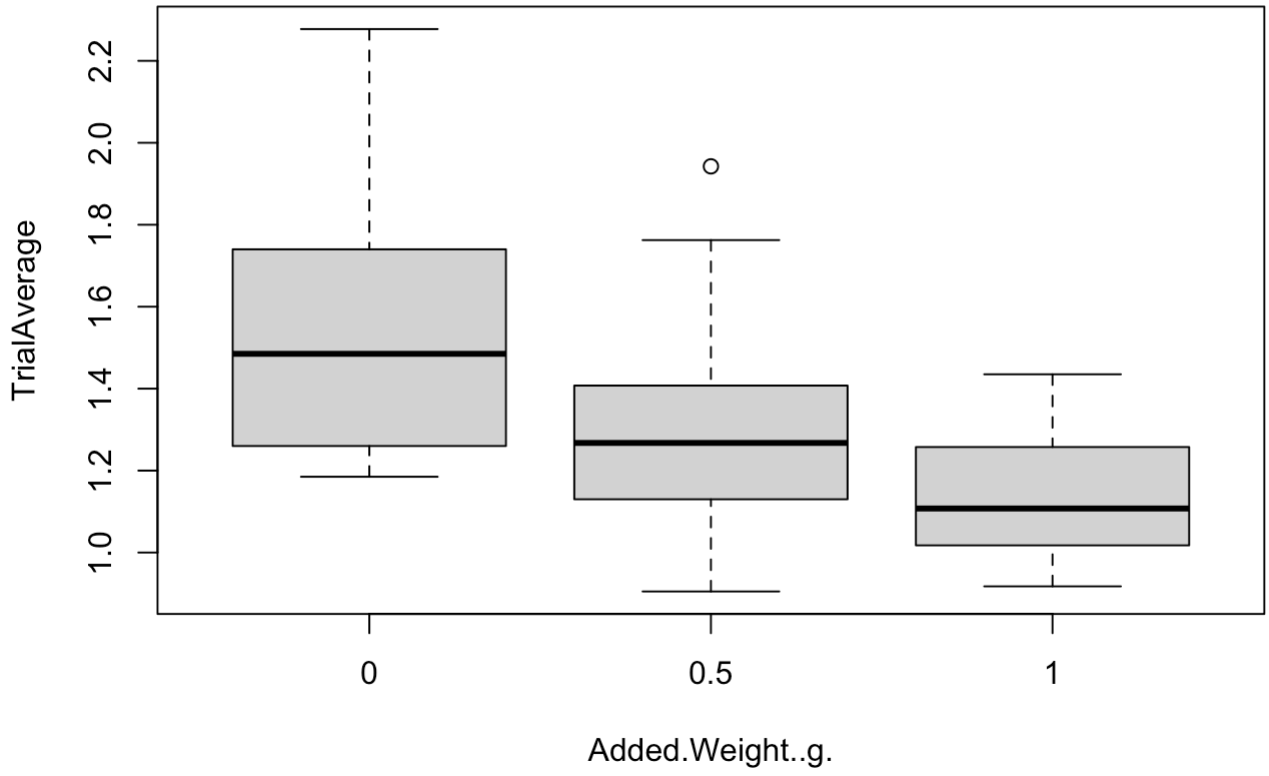
```
boxplot(TrialAverage ~ `Blade.Shape`  , data=data)
```



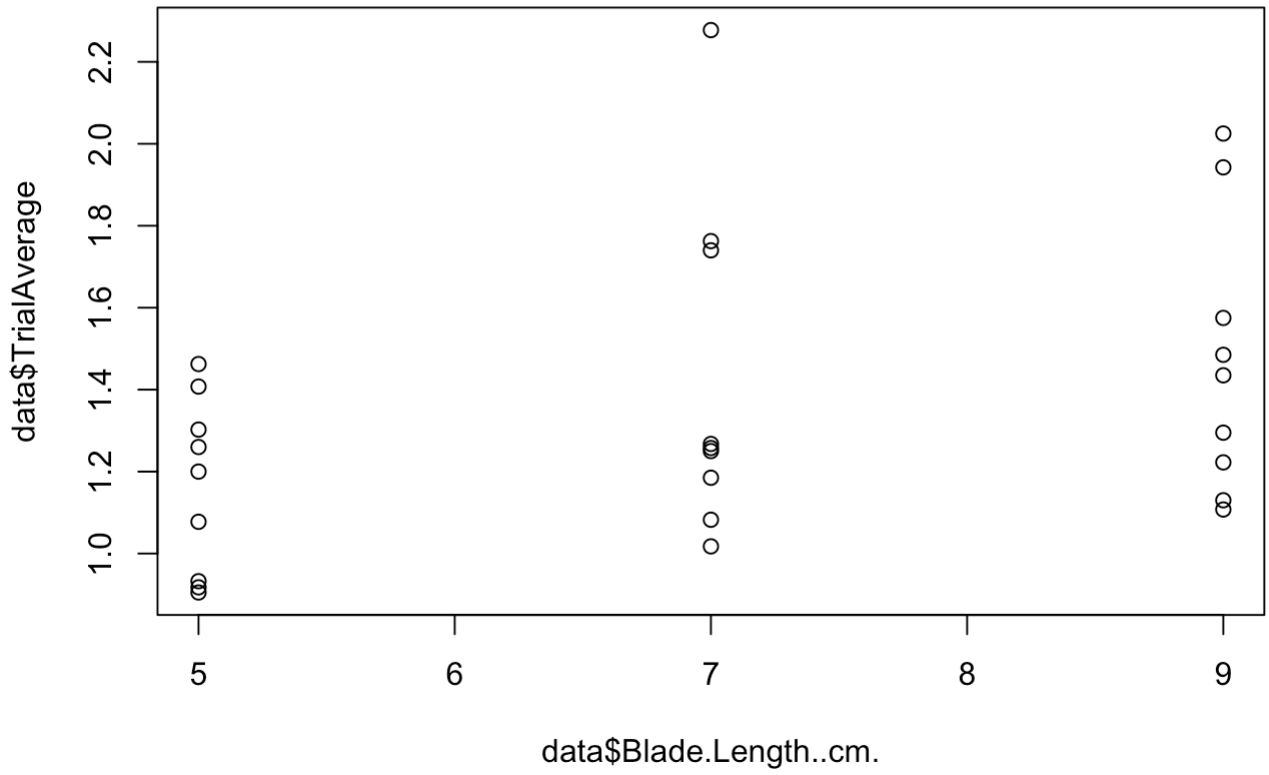

```
boxplot(TrialAverage ~ `Blade.Length..cm.` , data=data)
```



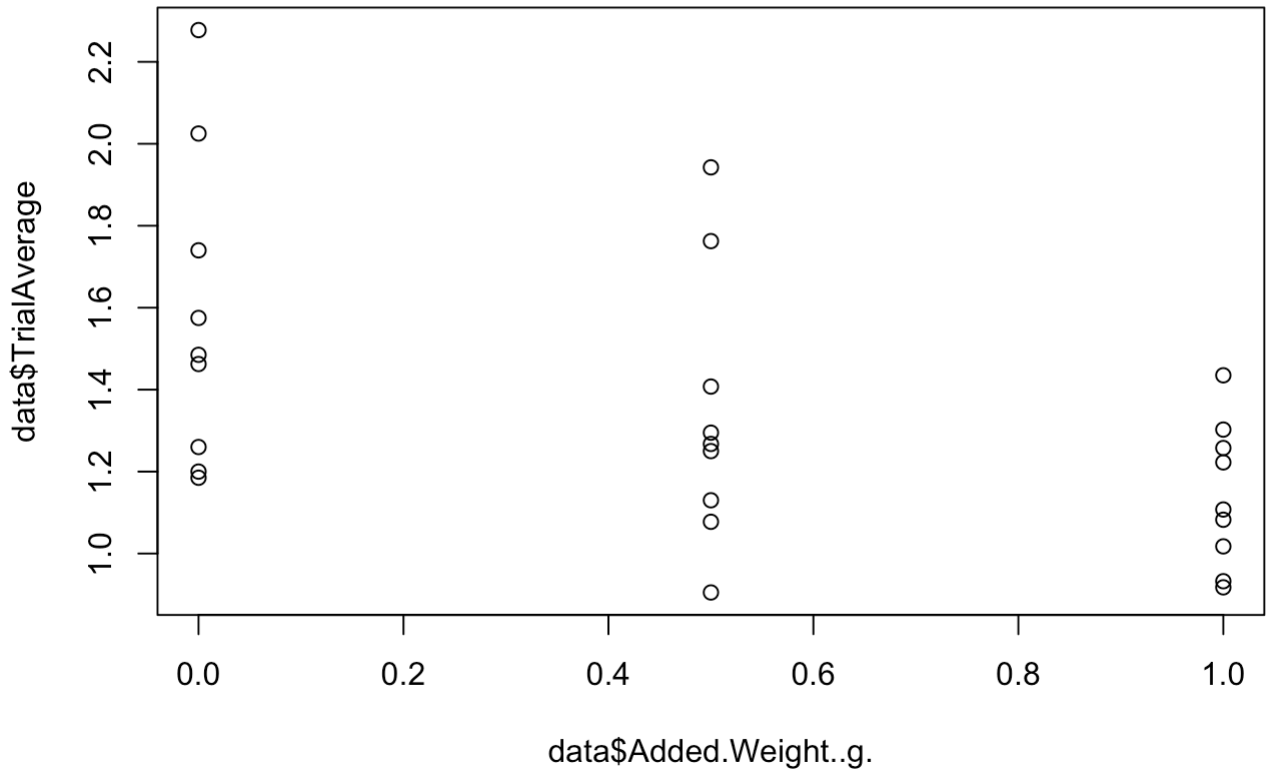
```
boxplot(TrialAverage ~ `Added.Weight..g.` , data=data)
```




```
plot(data$`Blade.Length..cm.` , data$TrialAverage)
```




```
plot(data$`Added.Weight..g.` , data$TrialAverage)
```



```
# ANOVA

anova_model <- aov(TrialAverage ~ `Blade.Shape` + `Blade.Length..cm.` + `Added.Weigh
t..g.`, data=data)
summary(anova_model)
```

```
##                   Df Sum Sq Mean Sq F value   Pr(>F)
## Blade.Shape        2 1.2174  0.6087   21.95 5.75e-06 ***
## Blade.Length..cm.  1 0.4209  0.4209   15.18 0.000778 ***
## Added.Weight..g.   1 0.8602  0.8602   31.02 1.34e-05 ***
## Residuals         22 0.6102  0.0277
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Correlation between blade length
cor_blade_length <- cor.test(data$Blade.Length..cm., data$TrialAverage)
print(cor_blade_length)
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$Blade.Length..cm. and data$TrialAverage
## t = 1.9786, df = 25, p-value = 0.05898
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.01401408  0.65621506
## sample estimates:
##       cor
## 0.3679595
```

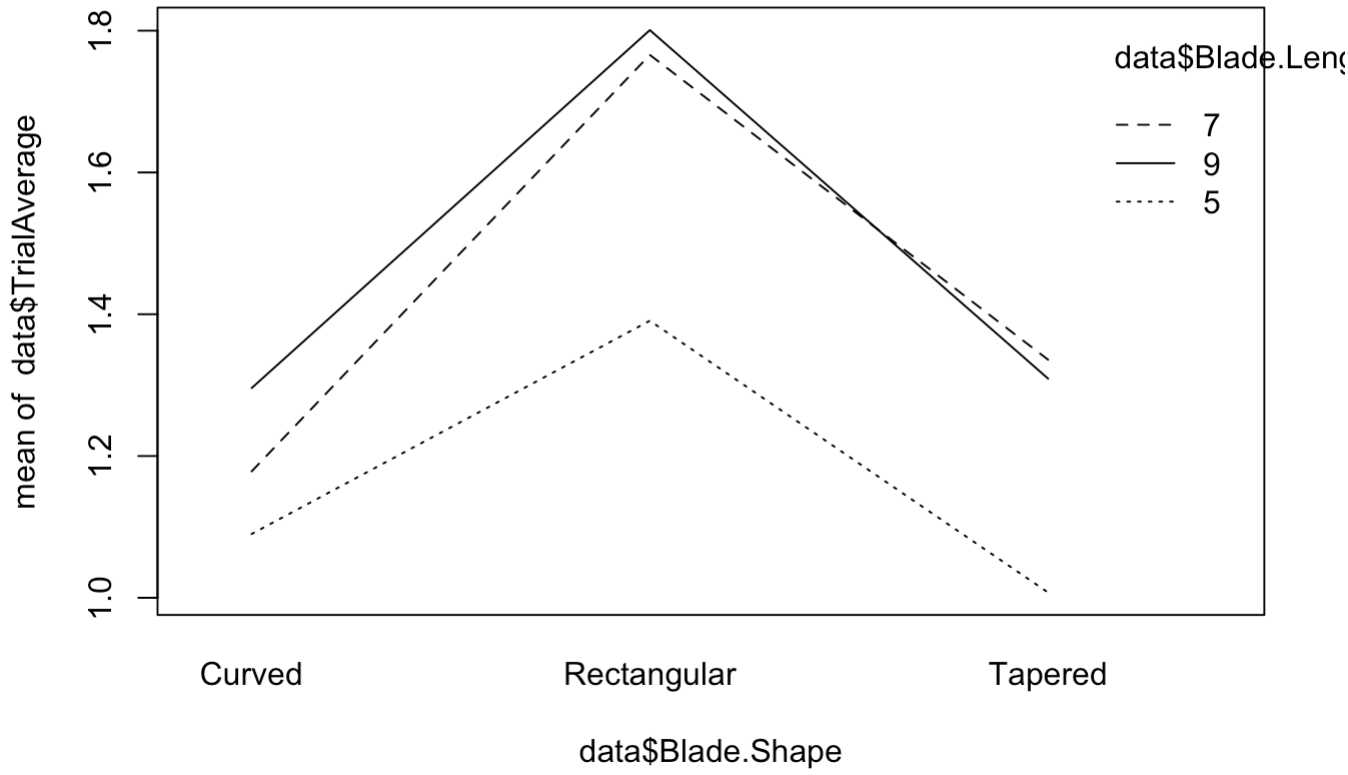
```
# Correlation between added weight
cor_added_weight <- cor.test(data$Added.Weight..g., data$TrialAverage)
print(cor_added_weight)
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$Added.Weight..g. and data$TrialAverage
## t = -3.0927, df = 25, p-value = 0.004828
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7551053 -0.1825081
## sample estimates:
##        cor
## -0.5260383
```

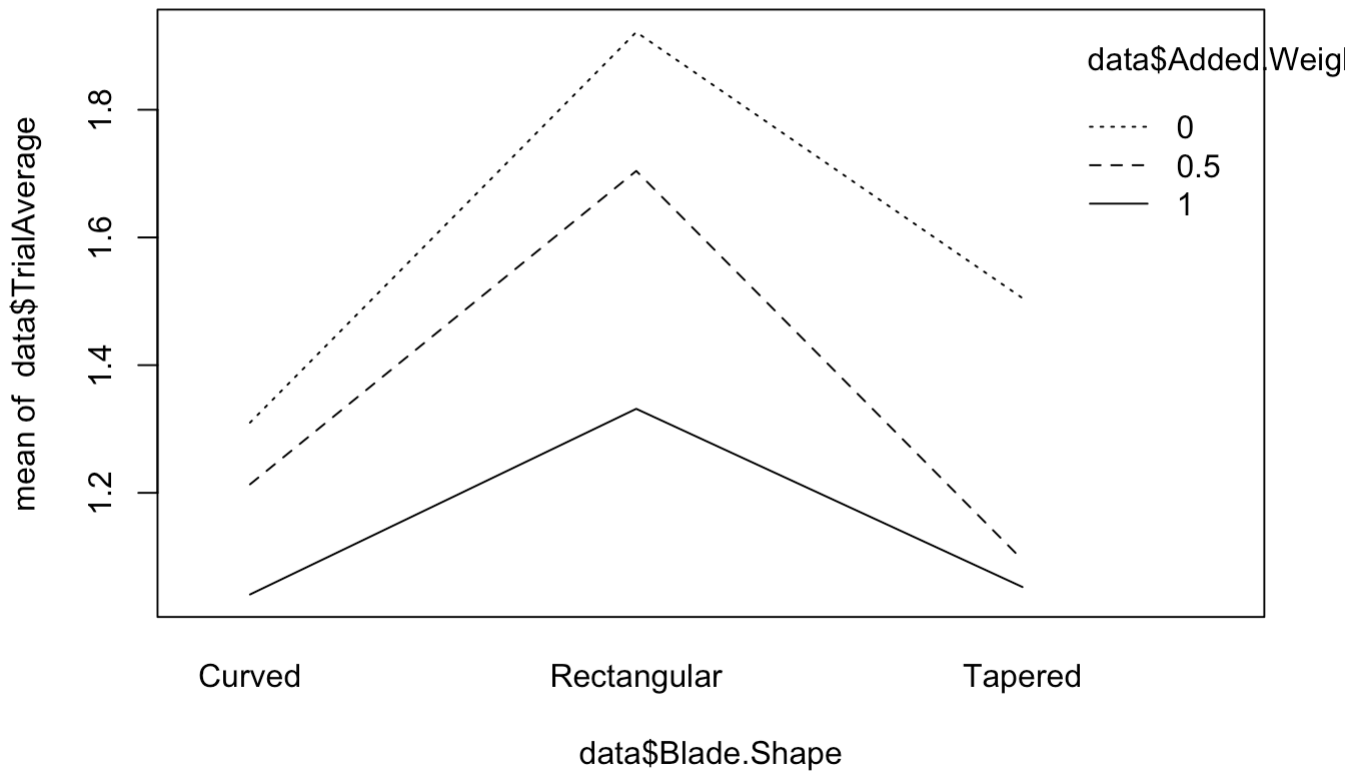
```
# Regression Analysis

#regression_model <- aov(TrialAverage ~ `Blade.Length..cm.` + `Added.Weight..g.`, dat
a = data)
#summary(regression_model)

regression_model_lm <- lm(TrialAverage ~ `Blade.Length..cm.` + `Added.Weight..g.`, da
ta = data)
summary(regression_model_lm)
```

```
##
## Call:
## lm(formula = TrialAverage ~ Blade.Length..cm. + Added.Weight..g.,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38630 -0.16894 -0.06449  0.15836  0.70620
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.03609    0.24264   4.270 0.000266 ***
## Blade.Length..cm.  0.07646    0.03252   2.351 0.027271 *
## Added.Weight..g.  -0.43722    0.13008  -3.361 0.002594 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.276 on 24 degrees of freedom
## Multiple R-squared:  0.4121, Adjusted R-squared:  0.3631
## F-statistic: 8.412 on 2 and 24 DF,  p-value: 0.001704
```

```
# Interaction Effects

interaction.plot(data$`Blade.Shape`, data$`Blade.Length..cm.`, data$TrialAverage)
```




```
interaction.plot(data$`Blade.Shape`, data$`Added.Weight..g.`, data$TrialAverage)
```



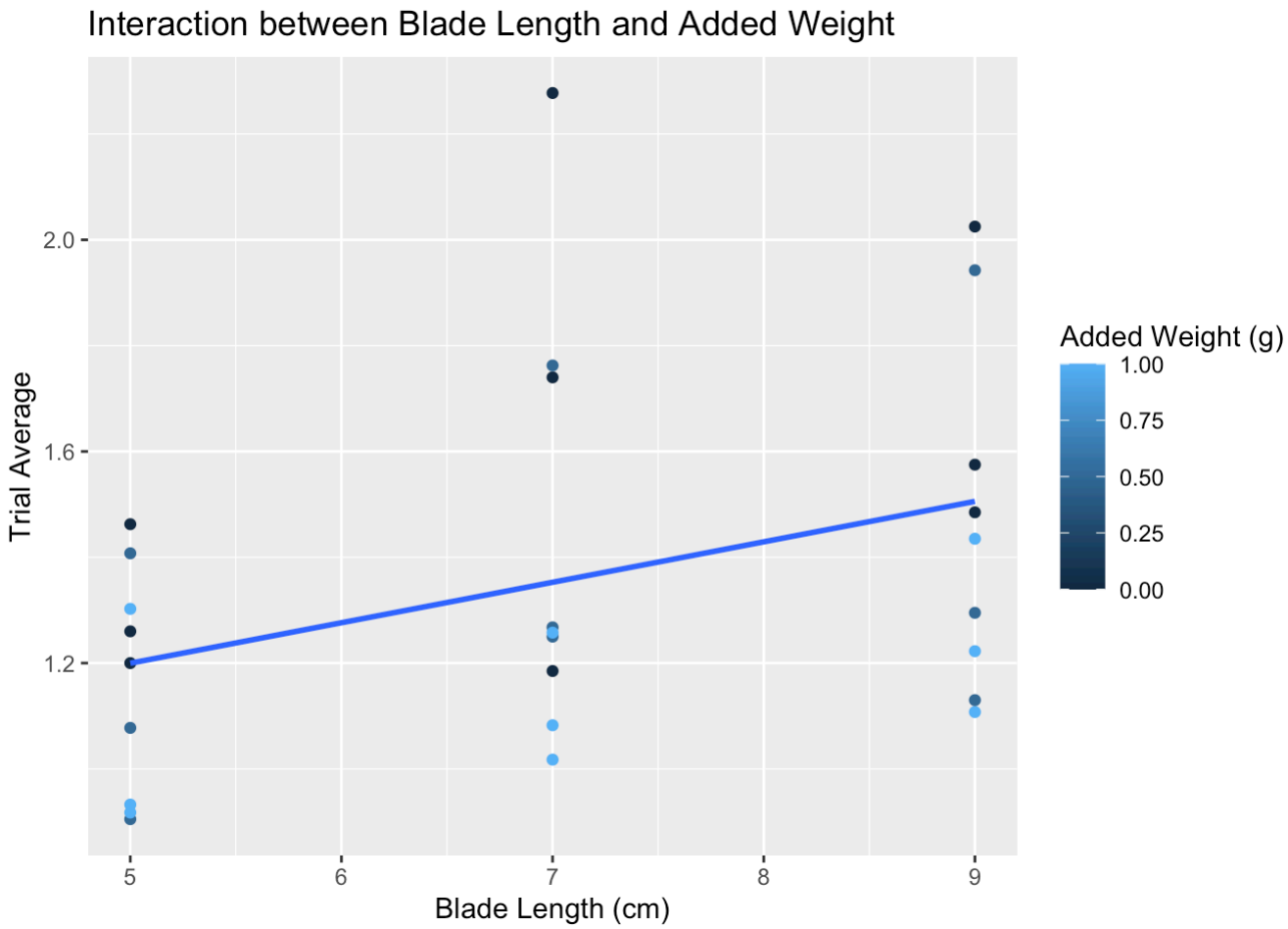
```
# Interaction effects

interaction_model <- lm(TrialAverage ~ Blade.Length..cm. * Added.Weight..g., data = d
ata)
summary(interaction_model)
```

```
##
## Call:
## lm(formula = TrialAverage ~ Blade.Length..cm. * Added.Weight..g.,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38630 -0.15310 -0.09449  0.18127  0.70620
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       0.87567    0.37495   2.335   0.0286 *
## Blade.Length..cm.                 0.09938    0.05216   1.905   0.0693 .
## Added.Weight..g.                 -0.11639    0.58086  -0.200   0.8430
## Blade.Length..cm.:Added.Weight..g. -0.04583    0.08081  -0.567   0.5761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2799 on 23 degrees of freedom
## Multiple R-squared:  0.4202, Adjusted R-squared:  0.3446
## F-statistic: 5.557 on 3 and 23 DF,  p-value: 0.005107
```

```
library(ggplot2)
ggplot(data, aes(x = Blade.Length..cm., y = TrialAverage, color = Added.Weight..g.))+
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title="Interaction between Blade Length and Added Weight",
       x = "Blade Length (cm)",
       y = "Trial Average",
       color = "Added Weight (g)")
```

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

```
## Warning: The following aesthetics were dropped during statistical transformation:
## colour.
## ℹ This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## ℹ Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```




```
turkey_result <- TukeyHSD(anova_model)
```

```
## Warning in replications(paste("~", xx), data = mf): non-factors ignored:
## Blade.Length..cm.
```

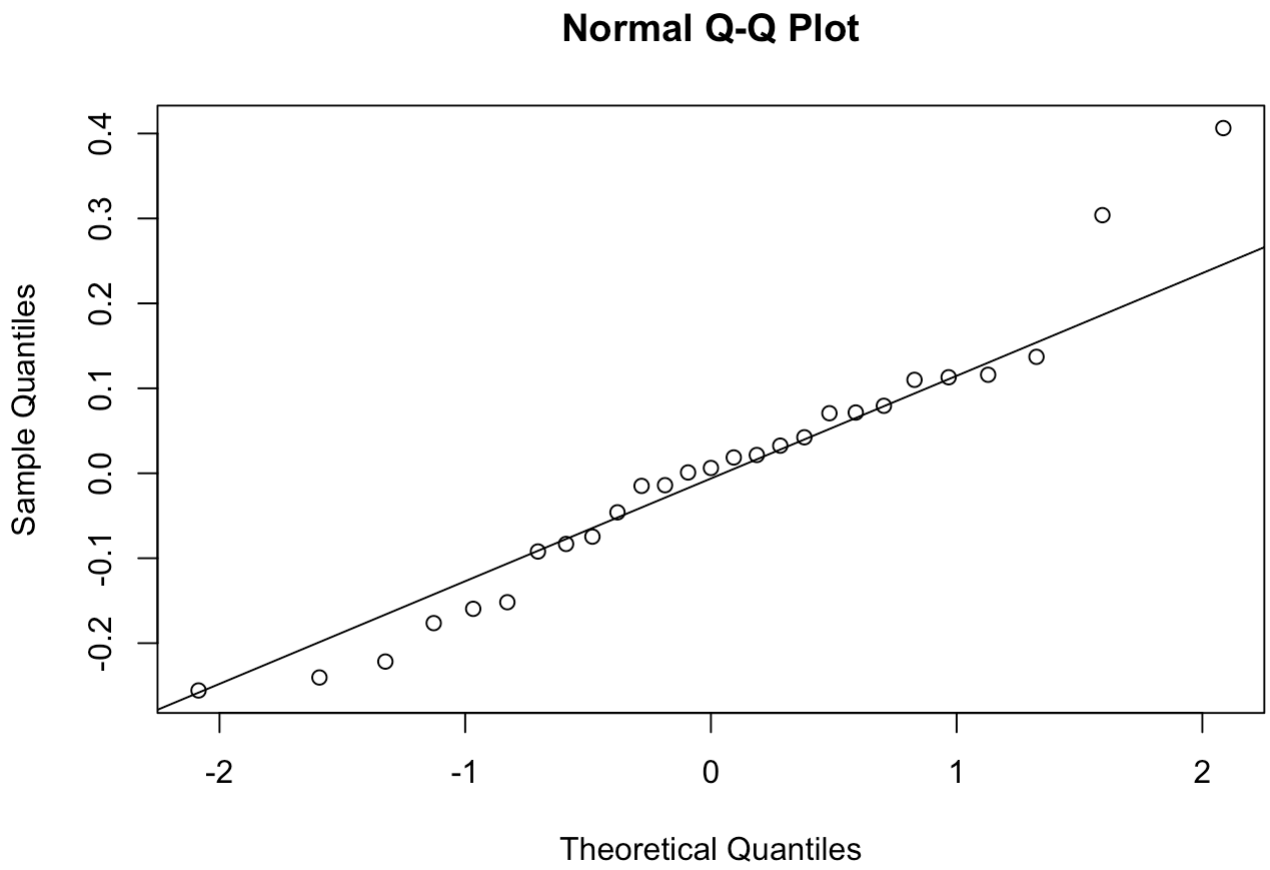
```
## Warning in replications(paste("~", xx), data = mf): non-factors ignored:
## Added.Weight..g.
```

```
## Warning in TukeyHSD.aov(anova_model): 'which' specified some non-factors which
## will be dropped
```

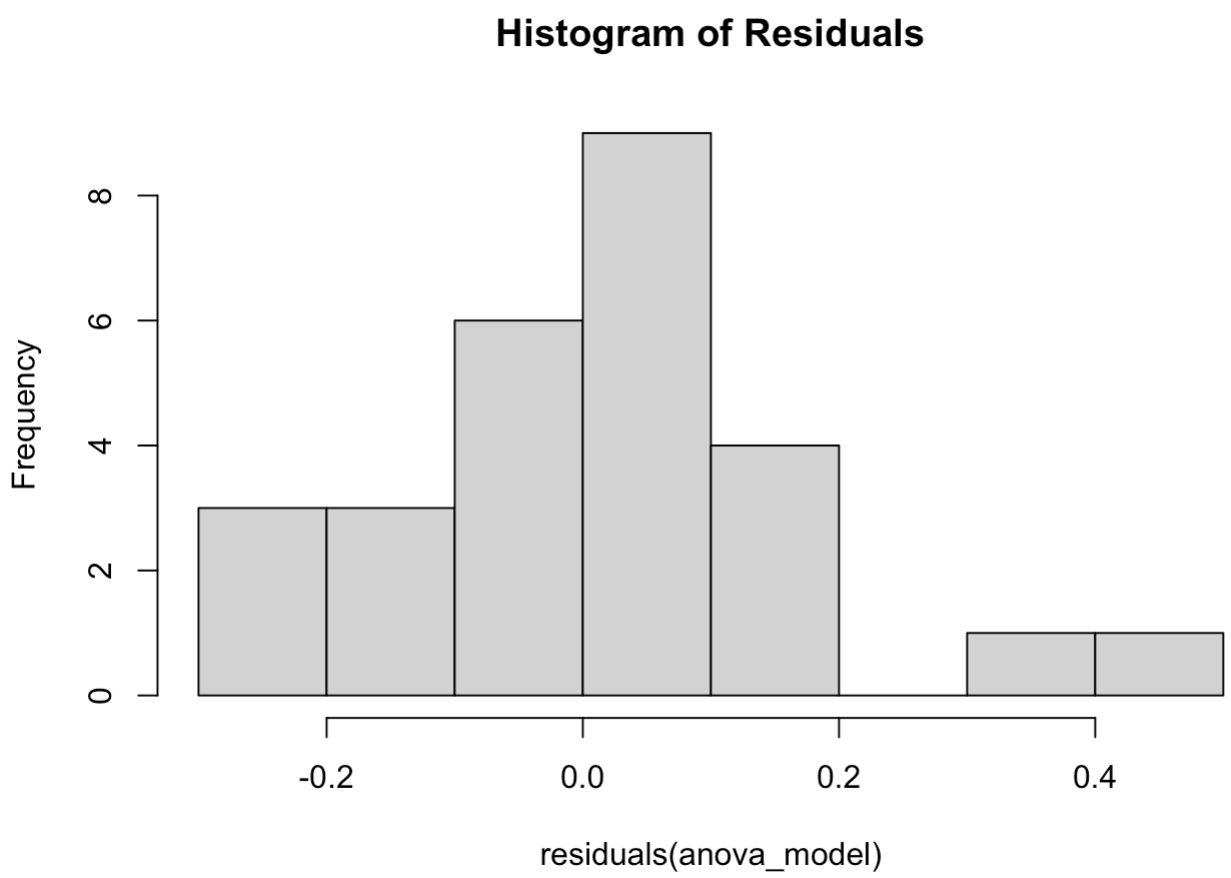
```
print(turkey_result)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = TrialAverage ~ Blade.Shape + Blade.Length..cm. + Added.Weight..
g., data = data)
##
## $Blade.Shape
##                              diff        lwr        upr     p adj
## Rectangular-Curved    0.46444444  0.2672278  0.6616611 0.0000171
## Tapered-Curved        0.02944444 -0.1677722  0.2266611 0.9256539
## Tapered-Rectangular -0.43500000 -0.6322166 -0.2377834 0.0000414
```

```
# ANOVA check

qqnorm(residuals(anova_model))
qqline(residuals(anova_model))
```



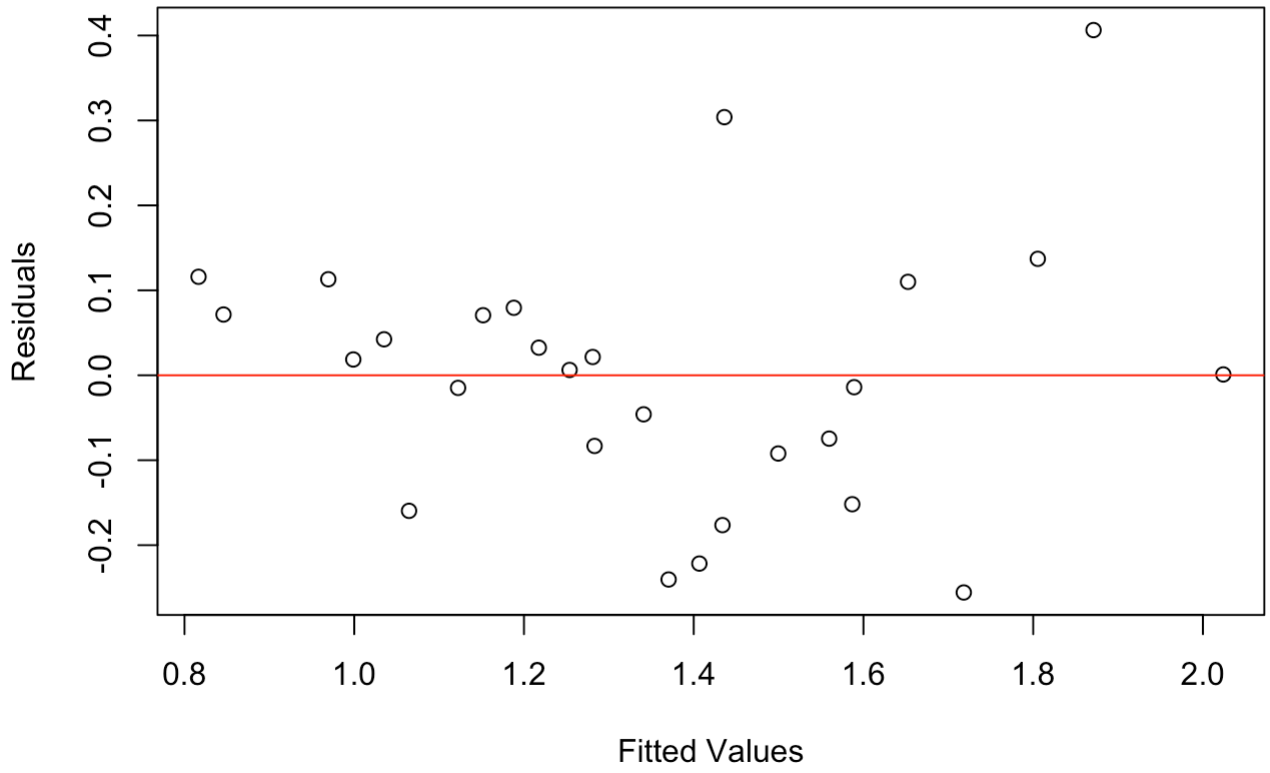

```
hist(residuals(anova_model), main = "Histogram of Residuals")
```



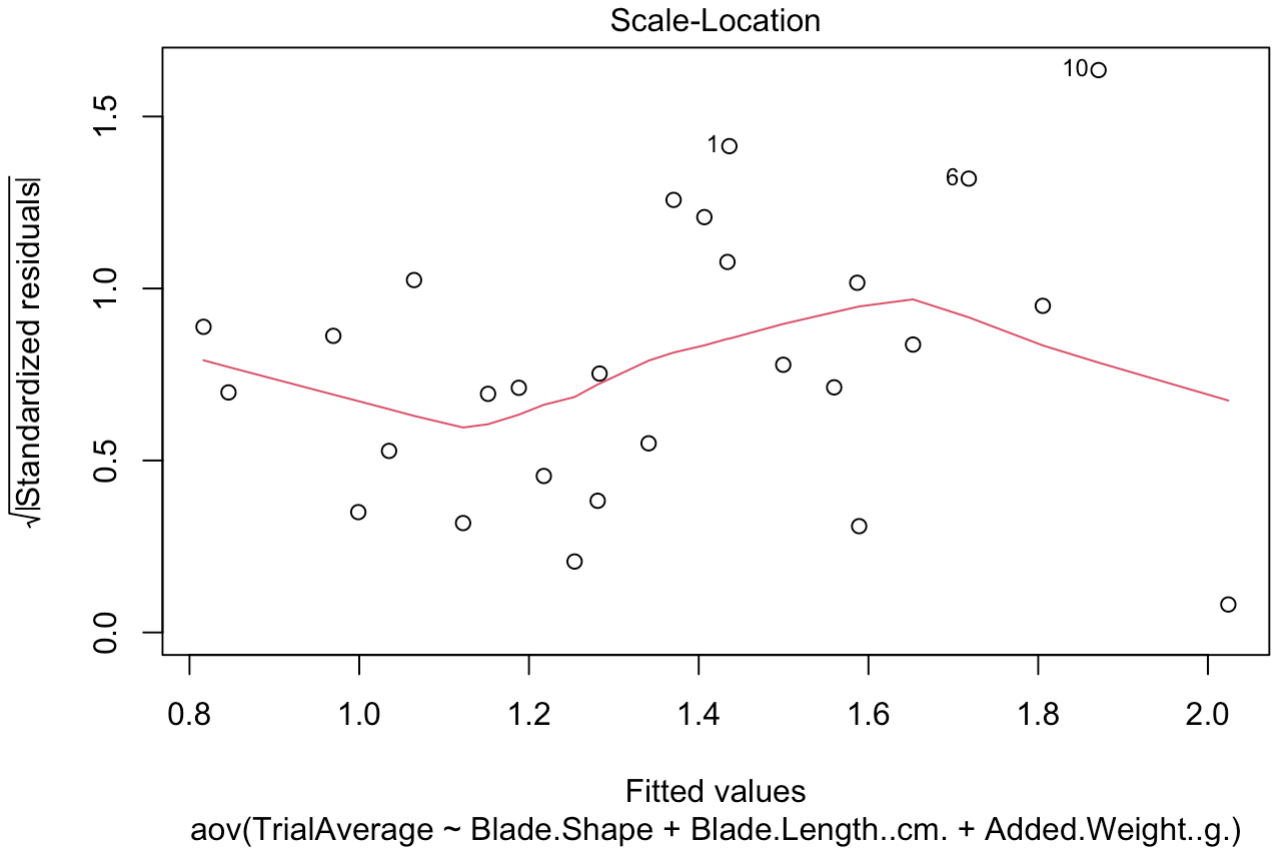

```
shapiro.test(residuals(anova_model))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(anova_model)
## W = 0.95498, p-value = 0.2823
```

```
# Residuals vs. Fitted values plot
plot(fitted(anova_model), residuals(anova_model), xlab = "Fitted Values", ylab = "Res
iduals")
abline(h=0, col="red")
```




```
# Scale-Location plot
plot(anova_model, 3)
```



```
#Levene's test
library(car)
```

```
## Loading required package: carData
```

```
leveneTest(TrialAverage ~ Blade.Shape, data = data)
```

```
## Warning in leveneTest.default(y = y, group = group, ...): group coerced to
## factor.
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2  1.5559 0.2316
##       24
```

```
# Bartlett's test
bartlett.test(TrialAverage ~ Blade.Shape, data = data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  TrialAverage by Blade.Shape
## Bartlett's K-squared = 4.5344, df = 2, p-value = 0.1036
```