

# Analysis of Textile Dye Extraction

Cyrus Davis, David Kaufman, Eugene Monforte, Max Wang

March 27, 2025

## Abstract

California produces about 1.6 million tons of orchard waste annually, with 40,000 acres of almond trees removed each year. Improper agricultural waste management contributes to air pollution, soil degradation and inefficient resource use. In response, the California almond community has committed to achieving zero waste by 2025, emphasizing the need for innovative solutions. One promising approach is the extraction of natural dyes from agricultural byproducts, such as almond shells and hulls, to replace synthetic dyes in the textile industry. This study aims to optimize dye extraction and application processes, contributing to the development of sustainable and eco-friendly textile dyeing alternatives.

This project explores the potential of repurposing almond waste from agriculture byproducts as a sustainable source of textile dyes. We will explore the extraction results of brown pigments from almond shells and hulls to optimize color shade, strength and permanence. A full factorial design was used to evaluate the effects of three independent variables: temperature, extraction time, and pH level, on the dye extraction process. The extracted dyes were analyzed based on their color strength (K/S value) to determine optimal conditions for achieving vibrant and lasting colors. This study provides a statistical analysis of the impact of extraction and dyeing factors on dye absorption and color intensity, contributing to the development of eco-friendly textile dyeing alternatives.

## Introduction

To investigate this potential, a full factorial experiment to optimize the extraction process of almond-based dyes was conducted. This extraction process was evaluated using three independent variables (factors). Factor A is temperature (25°C, 50°C, 80°C), factor B is time (20 min, 40 min, 60 min) and factor C is the pH Level (4, 7, 10). Once extracted, the almond-based dye was applied to textile fabric through an orthogonal experimental design to examine the effects of liquid ratio, pH and dyeing time on color absorption. The dependent variable in this study is the absorbance (ABS) value, which measures dye concentration in solution and the K/S value, which quantifies the color strength in fabrics.

Our objectives for this project are to understand the research and data, conduct a standard three-way factorial design and do non-parametric testing on the three-way design. To understand the research, the three factors: temperature, time and pH will be visualized through traditional boxplots with their respective means and standard deviations. A standard-three-way factorial design will be performed on all three factors where we investigate the two-way and three-way interactions between each factor. 3 two-way interaction plots ( $A \times B$ ,  $A \times C$ , and  $B \times C$ ) and 9 three-way interactions at their respective factors (i.e.:  $A \times B$  at each level of the third factor  $k$ ,  $A \times C$  at each level of the factor  $j$ , and  $B \times C$  at each level of the factor  $i$ , where  $i = \text{Factor A}$ ,  $j = B$ ,  $k = C$ ).

Parametric and nonparametric techniques will be used in order to study this model. Parametric techniques such as ANOVA and residual analysis will be conducted. Tukey's HSD post-hoc test

will be conducted to identify statistically significant effects of the three-way, two-way and simple effects on the model. Nonparametric techniques such as Aligned Rank Transform (ART) ANOVA testing will also be conducted on all simple, two-way, and three-way interactions. This version of ANOVA will align the data, rank it and apply factorial ANOVA techniques that occur when the model itself violates normality, homogeneity, or contains outliers.

### Methodology:

This report will be divided into three sections: (1) Understanding the Research Data, (2) Three-way Factorial Design, (3) Nonparametric Testing

#### **1) Understanding the Research Data**

In this section, we notice that there are three independent variables (factors): Temperature (A), Time (B), and pH (C) with a dependent (response) variable of ABS. Each of these factors have three different levels where temperature is at 25°C, 50°C, 80°C, time at 20 min, 40 min, and 60 min, and pH Level at 4, 7, and 10. Boxplots for all observations between all factors will be created along with plots for each factor will be created. Each plot will include a 5 number summary that contains a minimum, 1<sup>st</sup> Quartile, median, 3<sup>rd</sup> Quartile, max, mean, and standard deviation of the ABS value compared to each circumstance.

#### **2) Three-way Factorial Design**

For this section, a standard Three-way Factorial Design will be conducted between each of the variables. A full model would be created to perform this factorial design. The full model would be, for  $i = 25, 50, 80$ ;  $j = 20, 40, 60$ ;  $k = 4, 7, 10$ ;  $l = 1, 2, \dots, 81$ ; since there are 81 ABS results.

$$y_{ijkl} = \mu + \tau_i + \beta_j + \gamma_k + (\tau\beta)_{ij} + (\tau\gamma)_{ik} + (\beta\gamma)_{jk} + (\tau\beta\gamma)_{ijk} + \epsilon_{ijkl} ;$$

where:  $y_{ijkl}$  is the ABS values,  $\mu$  is the overall mean effect,  $(\tau)_i$  are the main effects of temperature on  $i$ ,  $(\beta)_j$  the main effects of time on  $j$ ,  $(\gamma)_k$  the main effects of pH on  $k$ ,  $(\tau\beta)_{ij}$  is the two-way interaction between temperature and time,  $(\tau\gamma)_{ik}$  is the two-way interaction between temperature and pH,  $(\beta\gamma)_{jk}$  is the two way interaction between time and pH,  $(\tau\beta\gamma)_{ijk}$  is the three-way interaction between temperature, time and pH, and  $\epsilon_{ijkl}$  is the random error term.

We assume this model follows all the regression conditions of normality, constant variance and randomness but we will conduct all residual diagnostics. Residual analysis testing will be done by the Shapiro-Wilk Normality Test, Breusch-Bagan Non-constant Variance Test, and Durbin Waston Randomness Test all at the level. A residual plot along with a histogram and boxplot will be used to display the data. Hypotheses testing will also be conducted for each factor along with their two-way and three-way interactions. Prior to hypothesis testing for the two-way and three-way interactions will include boxplots to depict any interactions with each factor. The following displays the format for each conducting test.

Shapiro-Wilk Normality Test:

$H_0$ : Data comes from a normal distribution

$H_A$ : *not*  $H_0$

W = \_\_\_\_\_ , p-value = \_\_\_\_\_

Breusch-Pagan Constant Variance Test:

$H_0$ : The error variance is constant

$H_A$ : *not*  $H_0$

BP = \_\_\_\_\_ , p-value = \_\_\_\_\_

Durbin-Watson Randomness Test:

$H_0$ : Data are random/uncorrelated

$H_A$ : *not*  $H_0$

DW = \_\_\_\_\_ , p-value = \_\_\_\_\_

Hypothesis Testing:

Testing Three-Way Interactions:

$H_0$ :  $(\tau\beta\gamma)_{ijk} = 0$  for all  $i, j, k$

$H_A$ : *not*  $H_0$

F = \_\_\_\_\_ ; df = \_\_\_\_\_ ; p-value = \_\_\_\_\_

Testing Two-Way Interactions:

**Temperature and Time (A x B):**

$H_0$ :  $(\tau\beta)_{ij} = 0$  for all  $i, j$

$H_A$ : *not*  $H_0$

F = \_\_\_\_\_ ; df = \_\_\_\_\_ ; p-value = \_\_\_\_\_

**Temperature and pH (A x C):**

$H_0$ :  $(\tau\gamma)_{ik} = 0$  for all  $i, k$

$H_A$ : *not*  $H_0$

F = \_\_\_\_\_ ; df = \_\_\_\_\_ ; p-value = \_\_\_\_\_

**Time and pH (B x C):**

$H_0$ :  $(\beta\gamma)_{jk} = 0$  for all  $j, k$

$H_A$ : *not*  $H_0$

F = \_\_\_\_\_ ; df = \_\_\_\_\_ ; p-value = \_\_\_\_\_

Testing only main effects:

**Temperature (A):**

$$H_0: (\tau)_i = 0 \text{ for all } i$$

$$H_A: \text{not } H_0$$

$$F = \text{____} ; df = \text{____} ; p\text{-value} = \text{____}$$

**Time (B):**

$$H_0: (\beta)_j = 0 \text{ for all } j$$

$$H_A: \text{not } H_0$$

$$F = \text{____} ; df = \text{____} ; p\text{-value} = \text{____}$$

**PH (C):**

$$H_0: (\gamma)_k = 0 \text{ for all } k$$

$$H_A: \text{not } H_0$$

$$F = \text{____} ; df = \text{____} ; p\text{-value} = \text{____}$$

For any significant factors, we will do the post-hoc test: Tukey's HSD test. This post hoc test is conducted to identify statistically significant effects of the three-way, two-way and simple interactions. This will be done when there are significant effects for each of these interactions. Tukey's HSD test would take the following form where a table will be generated depicting each of the main effects.

Tukey's HSD test:

$$H_0: \mu_i = \mu_j \text{ where } i \neq j$$

$$H_A: \mu_i \neq \mu_j$$

$$p\text{-value} = \text{____}$$

### 3) Nonparametric Testing

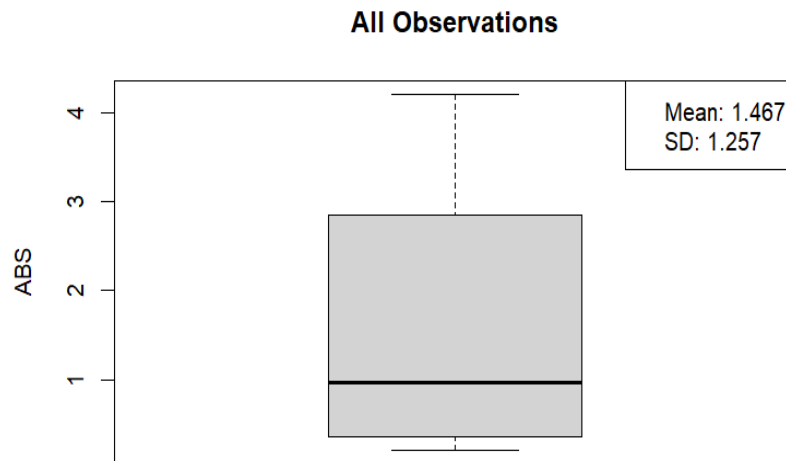
For this section, we will do nonparametric testing on the full model shown in the previous section. Nonparametric tests don't rely on distribution assumptions such as normality and it's useful to deal with skewed distributions, ordinal variables and outliers. For the sake of this report, we will conduct an ART ANOVA, a nonparametric alternative to traditional ANOVA where it's useful to deal with ordinal, non-normally distributed data that violates homoscedasticity (constant variance and homogeneity of variance). R functions such as "art()" and "art.con()" are used to investigate tests relationships.

Like in part 2, ART ANOVA will test all simple, two-way, and three-way interactions by hypothesis testing and using the art.con() function to test their significant effects using the Bonferroni method. This will reduce the risk of false positive Type I errors. The function will take the form: art.con(model, "factor(s)", adjust = "Bonferroni"), where factor(s) tests the significant effects.

## Results:

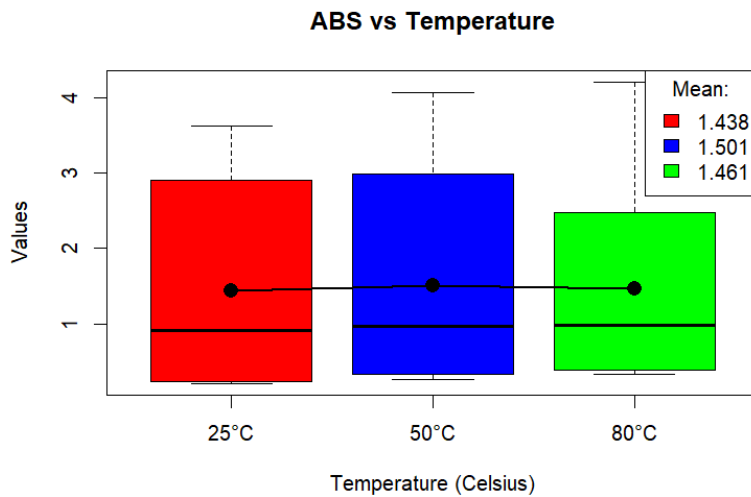
### 1) Understanding the Research Data

As mentioned, the response variable for this report is the absorption (ABS) of the textile dye where the three factors with three different levels: temperature (25, 50, 80), time (20, 40, 60) and pH (4, 7, 10). From here on factors A, B, and C will be known for temperature, time and pH, respectively. Below lists the following boxplots and 5 number summaries for all observations and for each individual factor. The distribution looks rightly skewed with no outliers at a mean of 1.467 and standard deviation of 1.257.



Five Number Summary for All Observations:

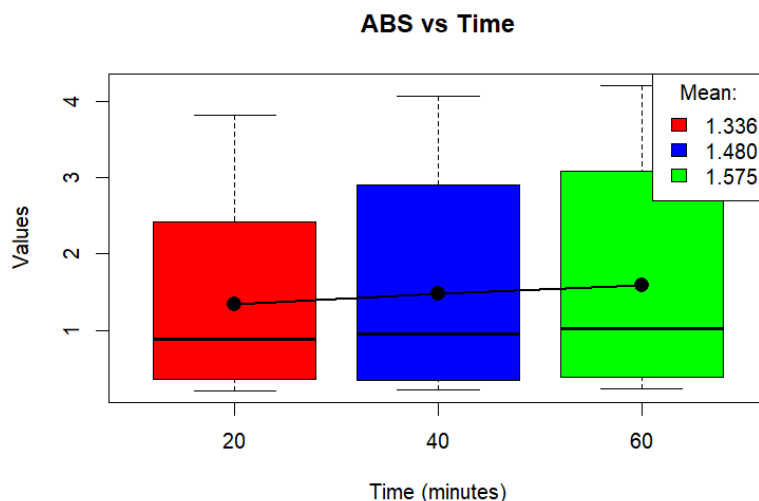
Min.	1 <sup>st</sup> Qu.	Median	3 <sup>rd</sup> Qu.	Max	Mean	Standard Deviation
0.213	0.353	0.960	2.852	4.196	1.467	1.257



Five Number Summary for Temperature:

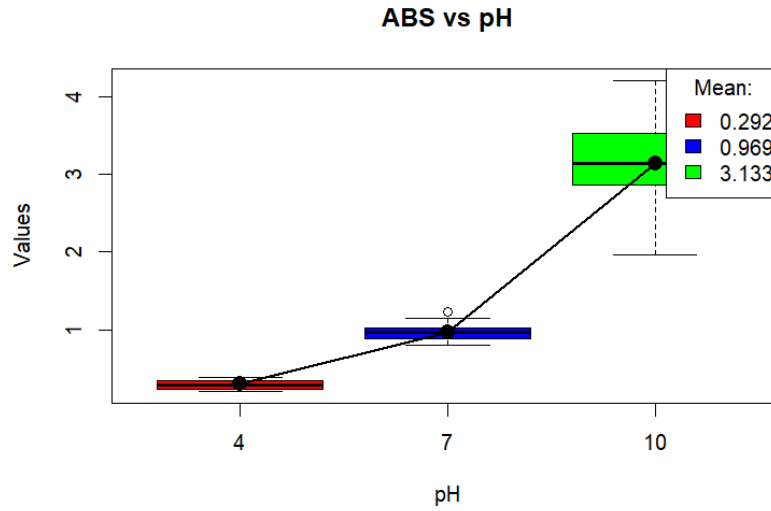
Factor/level	25°C	50°C	80°C
Min.	0.213	0.259	0.334
1 <sup>st</sup> Qu.	0.238	0.326	0.386
Median	0.913	0.960	0.983
3 <sup>rd</sup> Qu.	2.898	2.991	2.459
Max	3.616	4.071	4.196
Mean	1.438	1.501	1.461
Standard deviation	1.265	1.317	1.236

As for the distributions for each factor, there are relatively minor variations in the box plots for the temperature and time factors but there is an increased variation among the pH box plots. The temperature box plots, the concentration of the ABS values in temperature looks relatively stable whereas for time, the ABS values slowly increase from 20 to 40 and 60 minutes. There is a noticeable increase in ABS when comparing it with pH. When pH is 10, the ABS values of this factor are the highest of them.



Five Number Summary for Time:

factor/level	20	40	60
Min.	0.213	0.224	0.235
1 <sup>st</sup> Qu.	0.357	0.345	0.381
Median	0.880	0.948	1.018
3 <sup>rd</sup> Qu.	2.413	2.907	3.078
Max	3.810	4.071	4.196
Mean	1.336	1.480	1.575
Standard deviation	1.143	1.289	1.365



Five Number Summary for pH:

factor/level	4	7	10
Min.	0.213	0.798	1.197
1 <sup>st</sup> Qu.	0.238	0.886	2.860
Median	0.292	0.960	3.144
3 <sup>rd</sup> Qu.	0.351	1.018	3.530
Max	0.393	1.225	4.196
Mean	0.292	0.969	3.133
Standard deviation	0.062	0.103	0.537

## 2) Three-way Factorial Design

The full model for the data is:

$$\begin{aligned}
 \hat{y} = & 0.2145 + 0.06601(\tau)_{50} + 0.1501(\tau)_{80} + 0.0127(\beta)_{40} + 0.0272(\beta)_{60} + 0.6254(\tau)_7 + \\
 & 2.5815(\tau)_{10} + 0.0133(\tau\beta)_{50,40} - 0.0244(\tau\beta)_{80,40} - 0.000699(\tau\beta)_{50,60} - \\
 & 0.00791(\tau\beta)_{80,60} \\
 & + 0.007713(\tau\gamma)_{50,7} - 0.0445(\tau\gamma)_{80,7} - 0.0101(\tau\gamma)_{50,10} - 0.1274(\tau\gamma)_{80,10} + \\
 & 0.1164(\beta\gamma)_{40,7} + 0.2255(\beta\gamma)_{60,7} + 0.3377(\beta\gamma)_{40,10} + 0.5919(\beta\gamma)_{60,10} - \\
 & 0.0815(\tau\beta\gamma)_{50,40,7} \\
 & - 0.1045(\tau\beta\gamma)_{80,40,7} - 0.1426(\tau\beta\gamma)_{50,60,7} - 0.1543(\tau\beta\gamma)_{50,40,10} - 0.1883(\tau\beta\gamma)_{80,40,10} \\
 & - 0.0472(\tau\beta\gamma)_{50,60,10} - 0.0865(\tau\beta\gamma)_{80,60,10}
 \end{aligned}$$

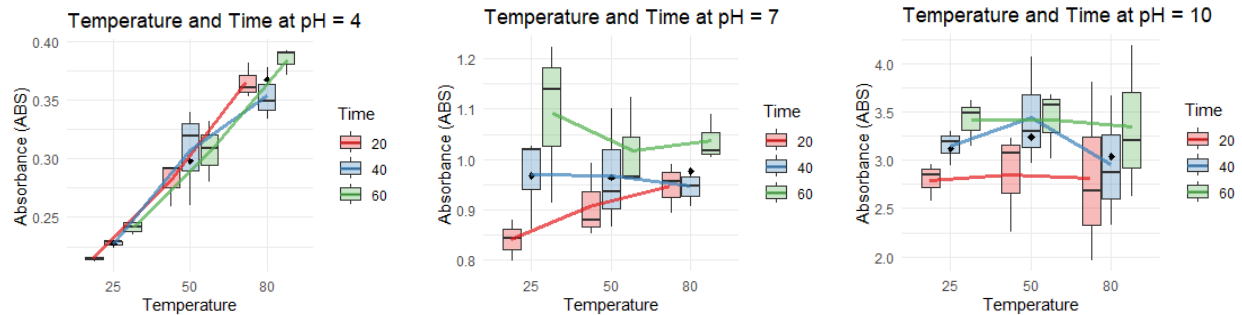
We assume the model is normal, but we must confirm it by testing the factor interactions and residual analysis.



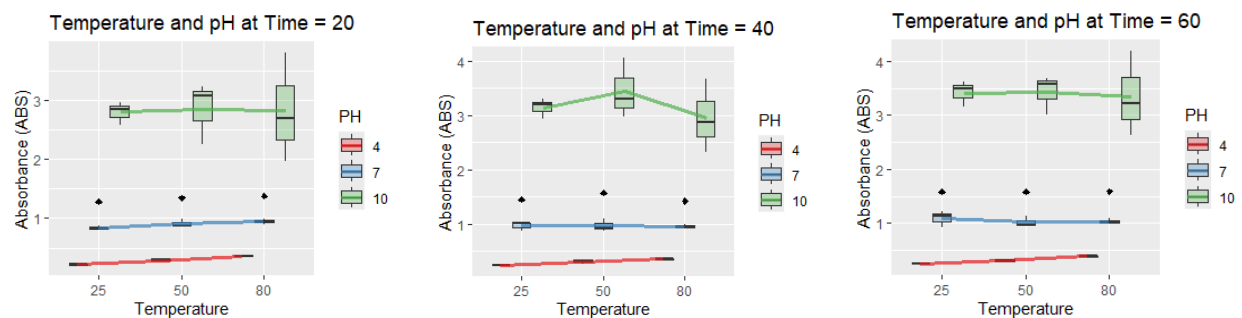
### Testing Three-Way Interactions:

Below depicts the three-way interaction plots for each factor A x B (temperature and time) at C (pH = 4,7,10), A x C (temperature and pH) at B (time = 20, 40, 60), and B x C (time and pH) at A (temperature = 25, 50, 80).

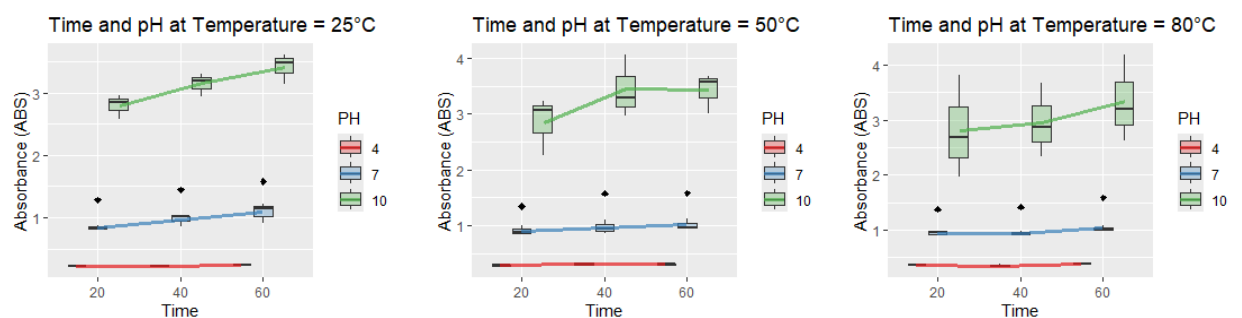
#### A x B at C (pH = 4,7,10)



#### A x C at B (Time = 20, 40, 60)



#### B X C at A (Temperature = 25, 50, 80)



	Temperature	Time	PH	Mean	SD
	<fct>	<fct>	<fct>	<dbl>	<dbl>
1	25	20	4	0.215	0.00187
2	25	20	7	0.840	0.0409
3	25	20	10	2.80	0.199
4	25	40	4	0.227	0.00281
5	25	40	7	0.969	0.0932
6	25	40	10	3.15	0.182
7	25	60	4	0.242	0.00695
8	25	60	7	1.09	0.162
9	25	60	10	3.42	0.244
10	50	20	4	0.281	0.0189
11	50	20	7	0.907	0.0747
12	50	20	10	2.85	0.527
13	50	40	4	0.307	0.0414
14	50	40	7	0.968	0.122
15	50	40	10	3.45	0.565
16	50	60	4	0.307	0.0261
17	50	60	7	1.02	0.0924
18	50	60	10	3.42	0.361
19	80	20	4	0.365	0.0148
20	80	20	7	0.946	0.0496
21	80	20	10	2.82	0.930
22	80	40	4	0.354	0.0224
23	80	40	7	0.946	0.0384
24	80	40	10	2.96	0.676
25	80	60	4	0.384	0.0119
26	80	60	7	1.04	0.0457
27	80	60	10	3.34	0.793

For the A x B three-way interaction (at each level of pH), the pH level does alter the interaction plot in a large manner. As the pH level increases, there is a trend. Some of the lines shift their direction from increasing to decreasing within each level, suggesting some effect on A x B.

For A x C, we notice each level of time doesn't change in shape. It can be noted that time doesn't have a huge effect on the factors and similarly for the three-way interaction of B x C on temperature. The three-way interaction on B x C at each level of temperature doesn't change when compared to the previous three-way interaction. However, there is a noticeable change at the pH level of 10. There seems to be a change in the ABS results as the means fluctuate at 50°C and decrease to 80°C.

#### Three-Way Interaction Hypothesis Testing:

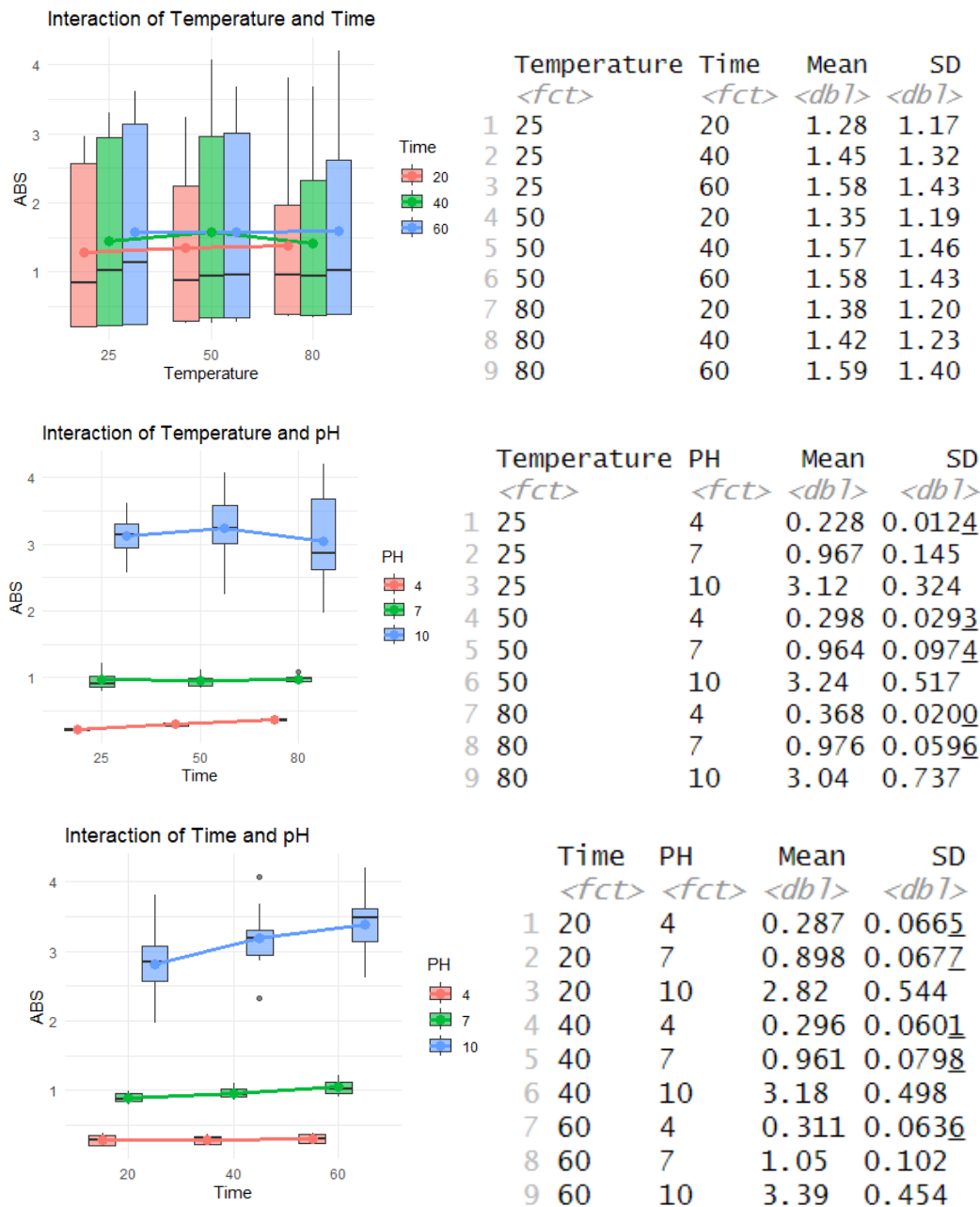
$$H_0: (\tau\beta\gamma)_{ijk} = 0 \text{ for all } i, j, k \quad H_A: \text{not } H_0$$

$$F = 0.1374; df = 8; p\text{-value} = 0.99713$$

For the three-way interaction for all factors, hypothesis testing reveals an F statistic of 0.1374 with a df (degrees of freedom) of 8 from the total 9 levels of each factor. Since the p-value is greater than the 0.05 level, we don't reject the null hypothesis and conclude that the three-way interaction between each factor is not significant. From here we move onto the 3 types of two-way interactions between each factor.

## Testing Two-Way Interactions:

Below depicts the two-way interactions between each factor. The three plots depict the two-way interactions: A x B (temperature and time), A x C (temperature and pH) and B x C (time and pH).



For the two-way interaction A x B (temperature and time), there was overlap between the two factors unlike the interactions between A x C (temperature and pH) and B x C (time and pH). This would indicate a small interaction between temperature and time due to the converging lines. For A x C and B x C, they are primarily the same. A x C depicts a parallel relationship

between each the two its very small. B x C there are no overlaps, but they are parallel pointing to the fact there may be an interaction. Further hypothesis testing for each factor is listed below.

**Temperature and Time Hypothesis Testing: (A x B)**

$$H_0: (\tau\beta)_{ij} = 0 \text{ for all } i, j \quad H_A: \text{not } H_0$$

$$F = 0.2546; df = 4; p\text{-value} = 0.90563$$

**Temperature and pH Hypothesis Testing: (A x C)**

$$H_0: (\tau\gamma)_{ik} = 0 \text{ for all } i, k \quad H_A: \text{not } H_0$$

$$F = 0.5105; df = 4; p\text{-value} = 0.72820$$

**Time and pH Hypothesis Testing: (B x C)**

$$H_0: (\beta\gamma)_{jk} = 0 \text{ for all } j, k \quad H_A: \text{not } H_0$$

$$F = 1.7945; df = 4; p\text{-value} = 0.14337$$

Based on the results for each of the respective interactions, they have the respective F statistics: 0.2546, 0.5105, 1.7945 all at a  $df = 4$  since the total number of levels for each factor is 6 which is subtracted by the 2 factors. The respective p-values: 0.90563, 0.72820, and 0.1437 are not greater than the 0.05 level so we don't reject any of the null hypothesis. We can conclude all the two-way interactions for A x B, A x C and B x C are not significant in any way based on the results. Since these are not significant, we further test the main effects of each factor individually.

Testing only main effects:

Hypothesis testing for each of the factors, each at their respective level have the following results.

**Temperature (A):**

$$H_0: (\tau)_i = 0 \text{ for all } i \quad H_A: \text{not } H_0$$

$$F = 0.2535; df = 2; p\text{-value} = 0.77701$$

**Time (B):**

$$H_0: (\beta)_j = 0 \text{ for all } j \quad H_A: \text{not } H_0$$

$$F = 3.9494; df = 2; p\text{-value} = 0.02507$$

### PH (C):

$$H_0: (\gamma)_k = 0 \text{ for all } k \quad H_A: \text{not } H_0$$

$$F = 555.0739; df = 2; p\text{-value} < 2e-6$$

For temperature, time, and pH, each has its respective F statistics at 0.2535, 3.9494, and 555.0739 all at the  $df = 2$ . It's worth noting that the p-value for each respective factor is: 0.77701, 0.02507 and  $<2e-6$ . From these results we do not reject the null hypothesis for temperature indicating that this factor is not significant but for time and pH we do. Both factors have a p-value less than the 0.05 level (0.02507 and  $<2e-6$ , respectively for time and pH) which indicates that time and pH have significant effects on the model itself.

### Tukey's HSD Testing:

Further testing of these main effects is done by Tukey's HSD test. We have the following hypotheses for both time and pH where "i" represents the first selected testing level which is compared to another level, "j", which is not the first selected level.

### Time (B):

$$H_0: \mu_i = \mu_j \text{ where } i \neq j \quad H_A: \text{not } H_0$$

where  $i = 20, 40, 60$  and  $j = \text{not } i$ ; (20-40, 20-60 and 40-60)

```
Time (only difference is with time at 60-20)
TukeyHSD(model, "Time")

## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = ABS ~ Temperature * Time * PH, data = df)
##
## $Time
##      diff      lwr      upr    p adj
## 40-20 0.1444016 -0.0699573 0.3587604 0.2446185
## 60-20 0.2489180  0.0345591 0.4632768 0.0191513
## 60-40 0.1045164 -0.1098425 0.3188753 0.4730414
```

According to Tukey's HSD test it, the interaction effects between time in minutes for 20-40, 20-60, and 40-60 reveal a large mean difference of 0.2489 between 20-60. The p-values for comparing 20 and 60 minutes (0.0191513) are less than the 0.05 level. This would indicate that the time spent at 20-60 of the textile dye has a huge effect on its ABS compared to the means of 20-40 & 40-60.

### PH (C):

$$H_0: \mu_i = \mu_k \text{ where } i \neq k \quad H_A: \text{not } H_0$$

where  $i = 4, 7, 10$  and  $k = \text{not } i$ ; (4-7, 4-10, 7-10)

```

PH (all significant)
TukeyHSD(model, "PH")

## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = ABS ~ Temperature * Time * PH, data = df)
##
## $PH
##      diff      lwr      upr p adj
## 7-4  0.6710505 0.4566916 0.8854093    0
## 10-4 2.8354063 2.6210474 3.0497651    0
## 10-7 2.1643558 1.9499969 2.3787146    0

```

For pH, Tukey's HSD reveals the interaction effects between each level of time for the pH levels between 4-7, 4-10, and 7-10. All the p-values for each comparison are less than the 0.05 level since each have a p-value = 0. This would further emphasize that pH has a significant effect on the ABS.

As for the full model of the data, since we assume the model is normal, we must confirm it by doing residual analysis.

#### Shapiro-Wilk Normality Test:

$H_0$ : Data comes from a normal distribution  $H_A$ : not  $H_0$   
W = 0.79947, p-value = 3.926e-09

#### Breusch-Pagan Constant Variance Test:

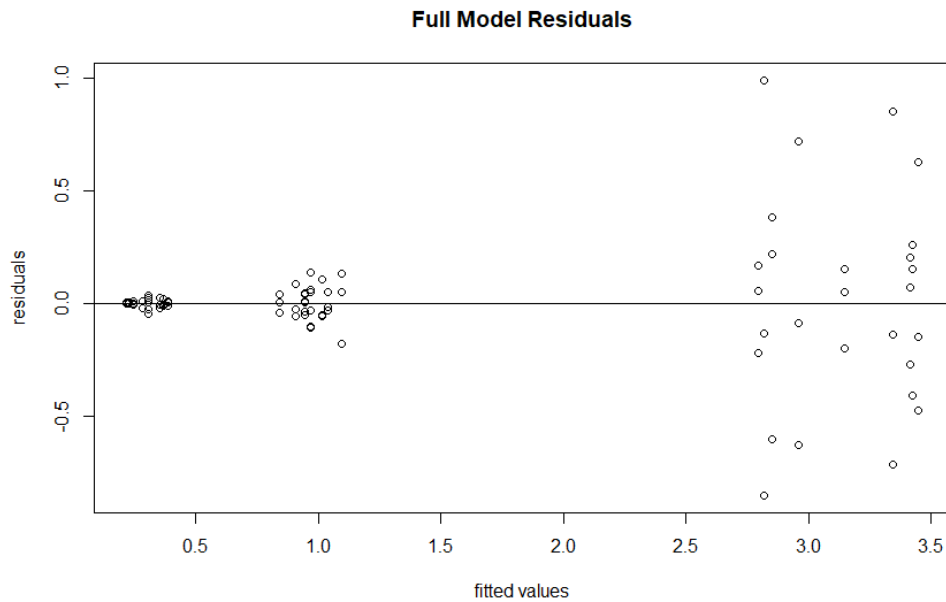
$H_0$ : The error variance is constant  $H_A$ : not  $H_0$   
BP = 49.889, df = 26, p-value = 0.003242

#### Durbin-Watson Randomness Test:

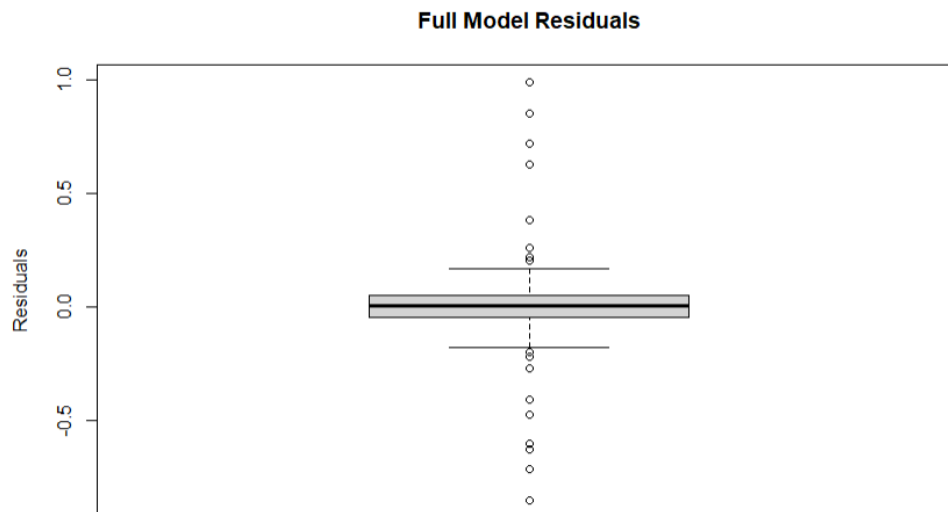
$H_0$ : Data are random/uncorrelated  $H_A$ : not  $H_0$   
DW = 2.4551, p-value = 0.1517

Based on these tests, the full model indicates that the data itself doesn't come from a normal distribution or have a constant variance as each has a p-value less than the 0.05 level. We reject the null hypothesis for both the Shapiro-Wilk and Breusch-Pagan tests, but the model appears to pass the Durbin-Watson test. It has a p-value of 0.1517 which is greater than the alpha level of 0.05 so the data itself is random. The full model has randomness to the data but is not normal or has constant variance.

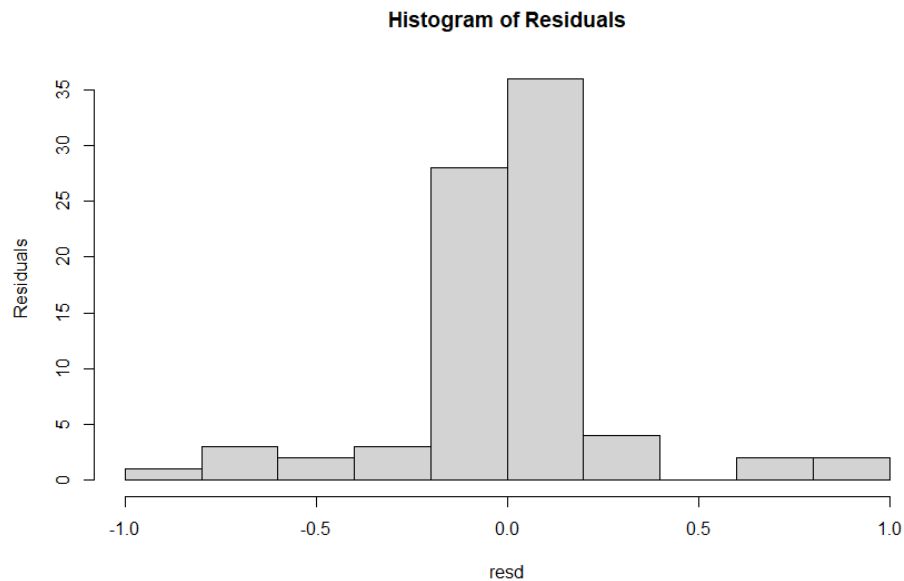
To further these findings, a residuals plot of the data was created along with a boxplot and histogram to depict the residuals. The residuals plot of the full model reveals there is a pattern going on. The residuals data itself reveals there is a pattern going on since there are three different clusters on the plot itself.



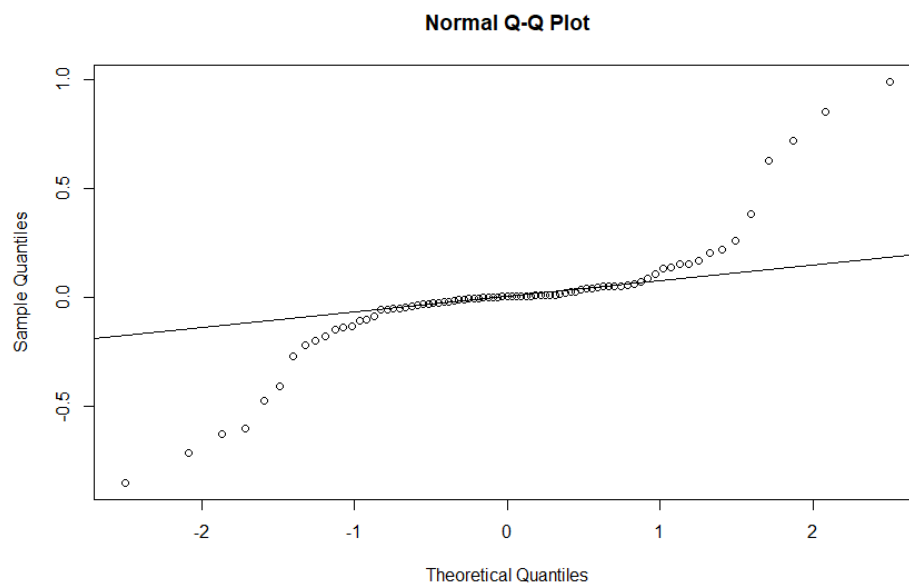
More analysis was conducted with a boxplot and histogram of the residuals. The boxplot reveals many residual outliers outside of the whiskers of the data. This would suggest that the model doesn't conform to a normal distribution or constant variance which is confirmed by the Shapiro-Wilk test and Bresch-Pagan test.



The histogram residual plot, the histogram is primary symmetric which suggests it may follow a normal distribution. However, it's worth noting that the larger bin from 0.0 to 0.25 of the residuals to be the peak. It can be suggested that the model itself is skewed to the left.



A QQ plot was generated for the residual analysis reveals a non-normal distribution. It's also worth noting that it has heavy tails. This would not contradict the previous claim that the residuals appear to be skewed. The model itself has heavy tails, but since it failed the Shapiro-Wilk test the model is not normal.





### 3) Nonparametric Testing

Due to the full model not being normal and violating the constant variance assumption, nonparametric testing via ART ANOVA was conducted. This is distribution free meaning it doesn't rely on standard models that accept the conditions. ART ANOVA ranks each of the factors to check each of the distribution of the factors and any interactions. Based on the following results, since each is approximately zero, the data itself would indicate the factors are independent. Further hypotheses testing reveals the following results.

Aligned Rank Transform of Factorial Model

Call:

```
art(formula = ABS ~ PH * Temperature * Time, data = emily)
```

Column sums of aligned responses (should all be ~0):

PH	Temperature	Time	PH:Temperature	PH:Time
0	0	0	0	0
Temperature:Time	PH:Temperature:Time			
0	0			

F values of ANOVAs on aligned responses not of interest (should all be ~0):

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	0	0	0	0

#### ART ANOVA Results:

##### Three-Way Interaction Hypothesis Testing:

$$H_0: (\tau\beta\gamma)_{ijk} = 0 \text{ for all } i, j, k \quad H_A: \text{not } H_0$$

$$F = 0.3752; df = 8; p\text{-value} = 0.929217$$

For the three-way interactions, the  $F = 0.3752$  with a p-value of 0.929217. Since the p-value is greater than the 0.05 level, we do not reject the null hypothesis and conclude the three-way interaction between each factor has no effect on the ABS.

##### Two-Way Interaction Hypothesis Testing:

##### **Temperature and Time (A x B):**

$$H_0: (\tau\beta)_{ij} = 0 \text{ for all } i, j \quad H_A: \text{not } H_0$$

$$F = 1.2940; df = 4; p\text{-value} = 0.283916$$

##### **Temperature and pH (A x C):**

$$H_0: (\tau\gamma)_{ik} = 0 \text{ for all } i, k \quad H_A: \text{not } H_0$$

$$F = 2.6467; df = 4; p\text{-value} = 0.043171$$

**Time and pH (B x C):**

$$H_0: (\beta\gamma)_{jk} = 0 \text{ for all } j, k \quad H_A: \text{not } H_0$$

$$F = 4.5500; df = 4; p\text{-value} = 0.003066$$

For the two-way interactions, we have the F statistics: 1.2940, 2.6467, and 4.5500, respectively for A x B, A x C and B x C all at df = 4. What's worth noting are the interaction effects for A x B and A x C. A x C has a p-value of 0.0043171 and B x C has a p-value of 0.003066. These p-values are less than the 0.05 level unlike A x B where it has a p-value of 0.283916. We reject the null hypotheses for A x C and B x C but not for A x B and conclude the interaction effects between temperature and pH along with Time and pH have a significant effect on ABS but not from temperature and time.

Main Effects Hypothesis Testing:**Temperature (A):**

$$H_0: (\tau)_i = 0 \text{ for all } i \quad H_A: \text{not } H_0$$

$$F = 2.1719; df = 2; p\text{-value} = 0.123815$$

**Time (B):**

$$H_0: (\beta)_j = 0 \text{ for all } j \quad H_A: \text{not } H_0$$

$$F = 13.4782; df = 2; p\text{-value} = 1.7858e-05$$

**pH (C):**

$$H_0: (\gamma)_k = 0 \text{ for all } k \quad H_A: \text{not } H_0$$

$$F = 221.3338; df = 2; p\text{-value} < 2.22e-16$$

For each of the main effects, each factor has F statistics: 2.1719, 13.4782, and 221.33378, respectively for A, B and C. It's also noticeable that the p-value for temperature and time are less than the 0.05 level whereas for time it's not. From this we reject the null hypotheses for both temperature and pH but not for time. We conclude that time and pH have a significant effect on ABS and temperature does not.

From here on we will conduct ART ANOVA to study the effects of time, pH, and the two-way interactions: temperature and pH (A x C) and time and pH (B x C).

### ART ANOVA Main Factors Testing:

Using the `art.con()` function, of the ART ANOVA results subjected to the Bonferroni correction which reduces the chance of type I errors in statistical testing, the following shows the results for the interaction effects between A x C and B x C and for the main factor B and C. Each category is like the Tukey's HSD test, but it's measured for two different categories. An example would be for B, the comparison is between (Time20 – Time40). This would be where it lists the first line as (10,25, - 10,50). This implies that two-way interaction of pH = 10 and temperature = 25 is being compared with pH = 10 and temperature = 50. If there are any yellow highlights, they would indicate a non-significant effect.

#### **B (Time):**

##	contrast	estimate	SE	df	t.ratio	p.value
##	Time20 - Time40	-16.6	6.14	54	-2.695	0.0281
##	Time20 - Time60	-31.9	6.14	54	-5.191	<.0001
##	Time40 - Time60	-15.3	6.14	54	-2.496	0.0470

By `art.con(ART_anova,"Time",adjust= "bonferroni")`, the above depicts the interaction effects between each of the levels of the time. Like Tukey's HSD test, time is being compared at 20-40, 20-60, and 40-60. The interaction effects between all these levels indicate a significant comparison as the p-values are: 0.0281, < 0.0001, and 0.0470. All are less than the 0.05 level which indicates for time comparisons between 20-40, 20-60, and 40-60 on the textile's ABS value is significant.

#### **C (pH):**

##	contrast	estimate	SE	df	t.ratio	p.value
##	PH4 - PH7	-27	2.56658	54	-10.520	<.0001
##	PH4 - PH10	-54	2.56658	54	-21.040	<.0001
##	PH7 - PH10	-27	2.56658	54	-10.520	<.0001

For pH, the Bonferroni method reveals that all the interaction effects between the pH levels between 4-7, 4-10, and 7-10 are significant. All the p-values for each comparison are less than the 0.05 level since each have a p-value < 0.0001. This would further emphasize that pH has a significant effect on the ABS.

### ART ANOVA Two-Way Interaction Testing:

An example for a comparison for the two-way interaction would be for A x C where it lists the first line as (10,25, - 10,50). This implies that two-way interaction of pH = 10 and temperature = 25 is being compared with pH = 10 and temperature = 50. The yellow highlights indicate the interaction effects that are not significant.

#### **Temperature and pH (A x C):**

##	contrast	estimate	SE	df	t.ratio	p.value
##	10,25 - 10,50	-1.556	3.72	54	-0.418	1.0000
##	10,25 - 10,80	1.222	3.72	54	0.329	1.0000
##	10,25 - 4,25	62.889	3.72	54	16.909	<.0001
##	10,25 - 4,50	53.889	3.72	54	14.489	<.0001
##	10,25 - 4,80	44.889	3.72	54	12.069	<.0001
##	10,25 - 7,25	26.444	3.72	54	7.110	<.0001
##	10,25 - 7,50	28.444	3.72	54	7.648	<.0001
##	10,25 - 7,80	25.778	3.72	54	6.931	<.0001
##	10,50 - 10,80	2.778	3.72	54	0.747	1.0000
##	10,50 - 4,25	64.444	3.72	54	17.327	<.0001
##	10,50 - 4,50	55.444	3.72	54	14.907	<.0001
##	10,50 - 4,80	46.444	3.72	54	12.487	<.0001
##	10,50 - 7,25	28.000	3.72	54	7.528	<.0001
##	10,50 - 7,50	30.000	3.72	54	8.066	<.0001
##	10,50 - 7,80	27.333	3.72	54	7.349	<.0001
##	10,80 - 4,25	61.667	3.72	54	16.580	<.0001
##	10,80 - 4,50	52.667	3.72	54	14.160	<.0001
##	10,80 - 4,80	43.667	3.72	54	11.741	<.0001
##	10,80 - 7,25	25.222	3.72	54	6.781	<.0001
##	10,80 - 7,50	27.222	3.72	54	7.319	<.0001
##	10,80 - 7,80	24.556	3.72	54	6.602	<.0001
##	4,25 - 4,50	-9.000	3.72	54	-2.420	0.0813
##	4,25 - 4,80	-18.000	3.72	54	-4.840	0.0004
##	4,25 - 7,25	-36.444	3.72	54	-9.799	<.0001
##	4,25 - 7,50	-34.444	3.72	54	-9.261	<.0001
##	4,25 - 7,80	-37.111	3.72	54	-9.978	<.0001
##	4,50 - 4,80	-9.000	3.72	54	-2.420	0.0813
##	4,50 - 7,25	-27.444	3.72	54	-7.379	<.0001
##	4,50 - 7,50	-25.444	3.72	54	-6.841	<.0001
##	4,50 - 7,80	-28.111	3.72	54	-7.558	<.0001
##	4,80 - 7,25	-18.444	3.72	54	-4.959	0.0003
##	4,80 - 7,50	-16.444	3.72	54	-4.421	0.0017
##	4,80 - 7,80	-19.111	3.72	54	-5.138	0.0001
##	7,25 - 7,50	2.000	3.72	54	0.538	1.0000
##	7,25 - 7,80	-0.667	3.72	54	-0.179	1.0000
##	7,50 - 7,80	-2.667	3.72	54	-0.717	1.0000

From interaction effects results between temperature and pH, there is a huge interaction effect when the levels of pH and temperature are at the highest level. We noticed earlier for the interaction effects between temperature and pH, pH of 10 has the largest ABS results compared to that of the pH at 4 and 7. The boxplots would relate to the p-values conducted by `art.con(ART_anova,"PH:Temperature",adjust= "bonferroni")` and it can be concluded that temperature has a significant effect when pH = 10. For other levels of pH at 4 and 7, there are some minor interaction effects, but it could be that as temperature increases, the textile dye reaches the optimal temperature needed for the textile.

### Time and pH (B x C):

##	contrast	estimate	SE	df	t.ratio	p-value
##	10,20 - 10,40	-5.00	3.71	54	-1.346	1.0000
##	10,20 - 10,60	-8.33	3.71	54	-2.243	1.0000
##	10,20 - 4,20	54.44	3.71	54	14.656	<.0001
##	10,20 - 4,40	49.89	3.71	54	13.429	<.0001
##	10,20 - 4,60	44.33	3.71	54	11.934	<.0001
##	10,20 - 7,20	28.67	3.71	54	7.717	<.0001
##	10,20 - 7,40	22.78	3.71	54	6.131	<.0001
##	10,20 - 7,60	16.22	3.71	54	4.367	0.0021
##	10,40 - 10,60	-3.33	3.71	54	-0.897	1.0000
##	10,40 - 4,20	59.44	3.71	54	16.002	<.0001
##	10,40 - 4,40	54.89	3.71	54	14.775	<.0001
##	10,40 - 4,60	49.33	3.71	54	13.280	<.0001
##	10,40 - 7,20	33.67	3.71	54	9.063	<.0001
##	10,40 - 7,40	27.78	3.71	54	7.477	<.0001
##	10,40 - 7,60	21.22	3.71	54	5.713	<.0001
##	10,60 - 4,20	62.78	3.71	54	16.899	<.0001
##	10,60 - 4,40	58.22	3.71	54	15.673	<.0001
##	10,60 - 4,60	52.67	3.71	54	14.177	<.0001
##	10,60 - 7,20	37.00	3.71	54	9.960	<.0001
##	10,60 - 7,40	31.11	3.71	54	8.375	<.0001
##	10,60 - 7,60	24.56	3.71	54	6.610	<.0001
##	4,20 - 4,40	-4.56	3.71	54	-1.226	1.0000
##	4,20 - 4,60	-10.11	3.71	54	-2.722	0.3140
##	4,20 - 7,20	-25.78	3.71	54	-6.939	<.0001
##	4,20 - 7,40	-31.67	3.71	54	-8.524	<.0001
##	4,20 - 7,60	-38.22	3.71	54	-10.289	<.0001
##	4,40 - 4,60	-5.56	3.71	54	-1.495	1.0000
##	4,40 - 7,20	-21.22	3.71	54	-5.713	<.0001
##	4,40 - 7,40	-27.11	3.71	54	-7.298	<.0001
##	4,40 - 7,60	-33.67	3.71	54	-9.063	<.0001
##	4,60 - 7,20	-15.67	3.71	54	-4.217	0.0034
##	4,60 - 7,40	-21.56	3.71	54	-5.802	<.0001
##	4,60 - 7,60	-28.11	3.71	54	-7.567	<.0001
##	7,20 - 7,40	-5.89	3.71	54	-1.585	1.0000
##	7,20 - 7,60	-12.44	3.71	54	-3.350	0.0533
##	7,40 - 7,60	-6.56	3.71	54	-1.765	1.0000

For the interaction effects between time and pH, pH still has a huge interaction effect when we compare it to time at pH = 10. Like the interaction between temperature and pH, there are high significant effects when pH = 10. There are huge interactions between pH = 10 as time is at 20, 40, or 60 minutes. However, when it's not at the maximum pH of 10, there aren't that many significant effects as some p-values aren't greater than the 0.05 level. But time and pH do have a significant effect as some points are mostly statistically significant. At pH = 4 and pH = 7, there are some significances concluded from these tests but it's generally the best at pH = 10.

## Conclusion:

The primary objectives of this analysis on the textile data were to provide a statistical analysis for a methodology that extracts liquids from discarded almond shells to create dyes to be used in the textile manufacturing process. For this analysis, there were 3 critical process-related independent variables: temperature, time and pH levels coupled with ABS, a variable which measures the dye concentration levels. An important preliminary objective was to create visualizations of the data through a series of box plots combined with summary statistical tables. Looking at all of the observations, we notice that the model itself is skewed to the right and further analysis of each factor: temperature, time and pH each have their own effect. For time and pH boxplots, there is an increase in variation whereas for temperature they are relatively stable. It's worth noting that pH has huge increase when comparing it solely with its ABS outputs.

Given the main objective of exploring interactions among the variables an ANOVA, analysis of variance model became our model of choice. Data analysis of the full model reveals that there isn't a significant two-way or three-way interaction between each of the factors through simple ANOVA. When testing for only the main effects of each factor, we do notice that time and pH have a significant effect on the model which led us to question their significant factors. Tukey's HSD test also reveals that time and pH are significant. The conducted Shapiro-Wilk, Breusch-Pagan and Durbin-Watson tests, reveal the full model has randomness to the data, but it doesn't come from a normal distribution or have a constant variance. There may be some variation between the models as it was later confirmed with the residuals. Residuals plot of the data reveals data clusters, confirmed by boxplot and histogram, that these clusters have many outliers that may have influenced the model itself.

Nonparametric testing continues our testing with ART ANOVA. Through ART ANOVA, this helps our understanding of the model since our data doesn't follow a normal distribution. The main effects of time and pH along with the two-way interactions: temperature and pH & time and pH have significant effects on the full model. For time, there are significant effects when time compared at for all levels of 20, 40, and 60 and similarly for pH at the 4, 7, and 10 pH levels. For the two-way interactions, these were later studied based on these initial understandings and is confirmed when either B x C or A x C at a pH of 10 have a huge effect on each respective interaction. Time plays a huge effect with its two-way interaction of pH unlike the interaction between temperature and pH. Although most of the A x C and B x C interactions were significant at most levels, for both interactions approximately 20% were not. Overall, the factors: time, pH, and the two-way interactions: temperature and pH & time and pH appear to be significant in this nonparametric model.

## Appendices:

### Part 1: Research Data

```
library(readr)
emily <- read_csv("STA classes/STA 5900/Group Project 2/Emily_FF_clean.csv")
view(emily)

#remove entry column
emily <- emily[,!names(emily) %in% "...1"]

# see!!!
view(emily)

# get the names
names(emily)

# convert names to factors
A <- as.factor(emily$Temperature)
B <- as.factor(emily$Time)
C <- as.factor(emily$PH)
ABS <- emily$ABS

A
B
C
ABS

# (1) all observations

# 5 number summary, mean and standard deviation
summary(ABS)

# Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
# 0.2126  0.3533  0.9603   1.4667  2.8519  4.1962

IQR(ABS) # 2.50
mean(ABS); sd(ABS) # 1.47, 1.26

#plot for all ABS vals
boxplot(ABS, ylab = "ABS", main = "All Observations")
legend("topright",c("Mean: 1.467", "SD: 1.257"))

#Temperature

# Temperature at 25°C
temp25_df = emily %>% filter(A == 25)
temp25_df

# mean and sd for temperature 25°C
m_temp25 <- mean(temp25_df$ABS, na.rm = TRUE)
sd_temp25 <- sd(temp25_df$ABS, na.rm = TRUE)

m_temp25; sd_temp25 # 1.44, 1.27
```

```

# Temperature at 50°C
temp50_df = emily %>% filter(A == 50)

# mean and sd for temperature 50°C
m_temp50 <- mean(temp50_df$ABS, na.rm = TRUE)
sd_temp50 <- sd(temp50_df$ABS, na.rm = TRUE)

m_temp50; sd_temp50 # 1.50, 1.32

# Temperature at 80°C
temp80_df = emily %>% filter(A == 80)

# mean and sd for temperature 80°C
m_temp80 <- mean(temp80_df$ABS, na.rm = TRUE)
sd_temp80 <- sd(temp80_df$ABS, na.rm = TRUE)

m_temp80; sd_temp80 # 1.46, 1.24

# combine for Temperature
ABS.Temp.25 <- temp25_df$ABS
ABS.Temp.50 <- temp50_df$ABS
ABS.Temp.80 <- temp80_df$ABS

# merged boxplot for temperature;
data1 <- data.frame(ABS.Temp.25, ABS.Temp.50, ABS.Temp.80)
names(data1) <- c("25°C", "50°C", "80°C") # Set proper column names

# 5 number summary
summary(data1)

# Compute means
means1 <- c(m_temp25, m_temp50, m_temp80)

# Boxplot for Temperature
boxplot(data1, main = "ABS vs Temperature", xlab = "Temperature (Celsius)",
        ylab = "Values", col= c("red", "blue", "green"), names = c("25°C", "50°C", "80°C"))

# Add points for the means
points(1:3, means1, col="black", pch=16, cex=1.5) # pch=16 for filled circles

# Add a line connecting the means
lines(1:3, means1, col="black", lwd=2, lty=1) # lty=2 makes it dashed

legend("topright", legend = c("1.438", "1.501", "1.461"),
      fill = c("red", "blue", "green"), xpd = TRUE,
      inset = c(0, 0), title = "Mean:")

# Time (minutes) at 20, 40 and 60

# Time (minutes) at 20
time20_df = emily %>% filter(Time == 20)

# mean and sd for Time = 20
m_time20 <- mean(time20_df$ABS, na.rm = TRUE)
sd_time20 <- sd(time20_df$ABS, na.rm = TRUE)

m_time20; sd_time20 # 1.34, 1.14

# Time (minutes) at 40
time40_df = emily %>% filter(Time == 40)

# mean and sd for Time = 40
m_time40 <- mean(time40_df$ABS, na.rm = TRUE)
sd_time40 <- sd(time40_df$ABS, na.rm = TRUE)

m_time40; sd_time40 # 1.48, 1.29

```



```

# Time (minutes) at 60
time60_df = emily %>% filter(Time == 60)

# mean and sd for Time = 60
m_time60 <- mean(time60_df$ABS, na.rm = TRUE)
sd_time60 <- sd(time60_df$ABS, na.rm = TRUE)

m_time60; sd_time60 # 1.58, 1.36

# combine for Time
ABS.Time.20 <- time20_df$ABS
ABS.Time.40 <- time40_df$ABS
ABS.Time.60 <- time60_df$ABS

# Boxplot for Time
data2 <- data.frame("20" = ABS.Time.20, "40" = ABS.Time.40, "60" = ABS.Time.60)
names(data2) <- c("20", "40", "60") # Set proper column names

# 5 number summary
summary(data2)

# Compute means
means2 <- c(m_time20, m_time40, m_time60)

# Boxplot for Time
boxplot(data2, main = "ABS vs Time", xlab = "Time (minutes)",
        ylab = "Values", col= c("red", "blue", "green"), names = c("20", "40", "60"))

# Add points for the means
points(1:3, means2, col="black", pch=16, cex=1.5) # pch=16 for filled circles

# Add a line connecting the means
lines(1:3, means2, col="black", lwd=2, lty=1) # lty=2 makes it dashed

legend("topright", legend = c("1.336", "1.480", "1.575"),
      fill = c("red", "blue", "green"), xpd = TRUE,
      inset = c(0, 0), title = "Mean:")

# pH at 4, 7 and 10

# pH at 4
ph4_df = emily %>% filter(PH == 4)

# mean and sd for ph = 4
m_ph4 <- mean(ph4_df$ABS, na.rm = TRUE)
sd_ph4 <- sd(ph4_df$ABS, na.rm = TRUE)

m_ph4; sd_ph4 # 0.298, 0.062

# pH at 7
ph7_df = emily %>% filter(PH == 7)

# mean and sd for ph = 7
m_ph7 <- mean(ph7_df$ABS, na.rm = TRUE)
sd_ph7 <- sd(ph7_df$ABS, na.rm = TRUE)

m_ph7; sd_ph7 # 0.979, 0.103

```

```

# pH at 10
ph10_df = emily %>% filter(PH == 10)

# mean and sd for ph = 10
m_ph10 <- mean(ph10_df$ABS, na.rm = TRUE)
sd_ph10 <- sd(ph10_df$ABS, na.rm = TRUE)

m_ph10; sd_ph10 # 3.133, 0.537

# merged boxplot
data3<- data.frame("4" = ABS.pH.4, "7" = ABS.pH.7, "10" = ABS.pH.10)
names(data3) <- c("4", "7", "10") # Set proper column names

# 5 number summary
summary(data3)

# Compute means
means3 <- c(m_ph4, m_ph7, m_ph10)

# Boxplot for Temperature
boxplot(data3, main = "ABS vs pH", xlab = "pH",
        ylab = "Values", col= c("red", "blue", "green"), names = c("4", "7", "10"))

# Add points for the means
points(1:3, means3, col="black", pch=16, cex=1.5) # pch=16 for filled circles

# Add a line connecting the means
lines(1:3, means3, col="black", lwd=2, lty=1) # lty=2 makes it dashed

legend("topright", legend = c("0.292", "0.969", "3.133"),
      fill = c("red", "blue", "green"), xpd = TRUE,
      inset = c(0, 0), title = "Mean:")

# Function to compute mean and sd
compute_stats <- function(df, group_vars) {
  df %>%
    group_by(across(all_of(group_vars))) %>%
    summarise(
      Mean = mean(ABS),
      SD = sd(ABS),
      .groups = 'drop'
    )
}

# Compute statistics for two-way interactions
stats_AB <- compute_stats(emily, c("Temperature", "Time"))
stats_AC <- compute_stats(emily, c("Temperature", "PH"))
stats_BC <- compute_stats(emily, c("Time", "PH"))

# Compute statistics for three-way interaction
stats_ABC <- compute_stats(emily, c("Temperature", "Time", "PH"))

# Print results
print("Mean and SD for Two-Way Interactions:")
print(stats_AB)
print(stats_AC)
print(stats_BC)

print("Mean and SD for Three-Way Interaction:")
print(stats_ABC)
print(stats_ABC, n = 27)

```

## Two-way interaction plots

```
# two way interactions

# Convert to factors for plotting
A2 <- as.factor(emily$Temperature) #Temperature
B2 <- as.factor(emily$Time)        # Time
C2 <- as.factor(emily$PH)          # pH

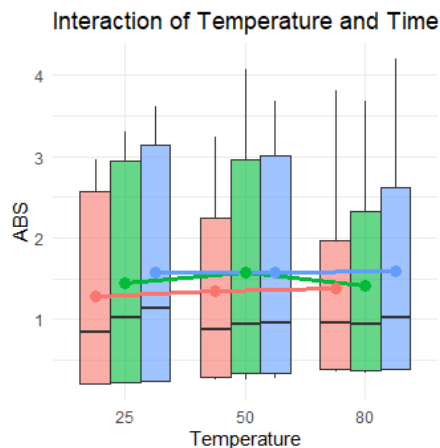
# Plot boxplot with Temperature and Time interactions

ggplot(emily, aes(x = A2, y = ABS, fill = as.factor(B2))) +
  geom_boxplot(position = position_dodge(width = 0.75), alpha = 0.6) + # Boxplots with fill and transparency
  stat_summary(aes(group = B2, color = as.factor(B2)), fun = mean, geom = "line",
    position = position_dodge(width = 0.75), size = 1.2) + # Mean interaction lines
  stat_summary(aes(group = B2, color = as.factor(B2)), fun = mean, geom = "point",
    position = position_dodge(width = 0.75), size = 3) + # Mean points
  labs(title = "Interaction of Temperature and Time", x = "Temperature", y = "ABS", fill = "Time", color = "Time") +
  theme_minimal()

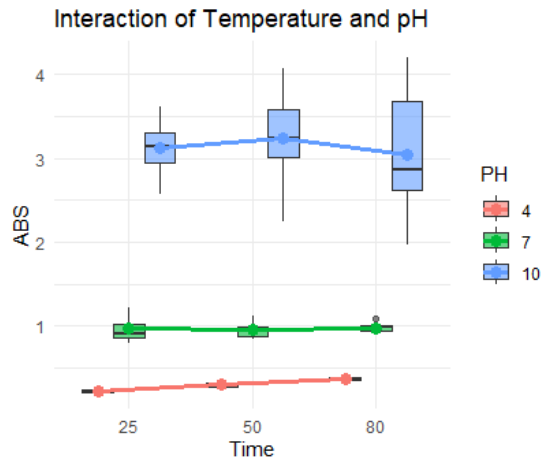
# Plot boxplot with Temperature and pH interactions

ggplot(emily, aes(x = A2, y = ABS, fill = as.factor(C2))) +
  geom_boxplot(position = position_dodge(width = 0.75), alpha = 0.6) + # Boxplots with fill and transparency
  stat_summary(aes(group = C2, color = as.factor(C2)), fun = mean, geom = "line",
    position = position_dodge(width = 0.75), size = 1.2) + # Mean interaction lines
  stat_summary(aes(group = C2, color = as.factor(C2)), fun = mean, geom = "point",
    position = position_dodge(width = 0.75), size = 3) + # Mean points
  labs(title = "Interaction of Temperature and pH", x = "Time", y = "ABS", fill = "PH", color = "PH") +
  theme_minimal()

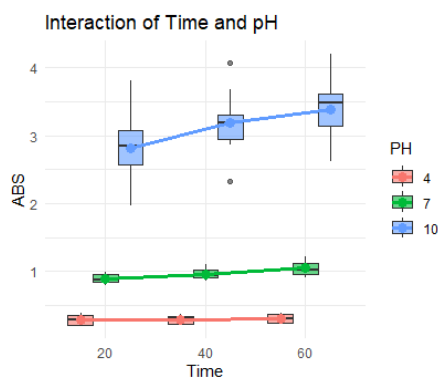
# Plot boxplot with interaction lines for Time and pH
ggplot(emily, aes(x = B2, y = ABS, fill = as.factor(C2))) +
  geom_boxplot(position = position_dodge(width = 0.75), alpha = 0.6) + # Boxplots with fill and transparency
  stat_summary(aes(group = C2, color = as.factor(C2)), fun = mean, geom = "line",
    position = position_dodge(width = 0.75), size = 1.2) + # Mean interaction lines
  stat_summary(aes(group = C2, color = as.factor(C2)), fun = mean, geom = "point",
    position = position_dodge(width = 0.75), size = 3) + # Mean points
  labs(title = "Interaction of Time and pH", x = "Time", y = "ABS", fill = "PH", color = "PH") +
  theme_minimal()
```



```
> print(stats_AB)
# A tibble: 9 × 4
  Temperature Time    Mean    SD
  <fct>        <fct> <dbl> <dbl>
1 25          20    1.28  1.17
2 25          40    1.45  1.32
3 25          60    1.58  1.43
4 50          20    1.35  1.19
5 50          40    1.57  1.46
6 50          60    1.58  1.43
7 80          20    1.38  1.20
8 80          40    1.42  1.23
9 80          60    1.59  1.40
```



```
> print(stats_AC)
# A tibble: 9 × 4
  Temperature PH      Mean      SD
  <fct>      <fct> <dbl> <dbl>
1 25         4      0.228 0.0124
2 25         7      0.967 0.145
3 25        10      3.12  0.324
4 50         4      0.298 0.0293
5 50         7      0.964 0.0974
6 50        10      3.24  0.517
7 80         4      0.368 0.0200
8 80         7      0.976 0.0596
9 80        10      3.04  0.737
```



```
> print(stats_BC)
# A tibble: 9 × 4
  Time PH      Mean      SD
  <fct> <fct> <dbl> <dbl>
1 20    4      0.287 0.0665
2 20    7      0.898 0.0677
3 20   10      2.82  0.544
4 40    4      0.296 0.0601
5 40    7      0.961 0.0798
6 40   10      3.18  0.498
7 60    4      0.311 0.0636
8 60    7      1.05  0.102
9 60   10      3.39  0.454
```

### Three-way interaction plots:

```
# three way interaction
```

```
# Interaction for AxB at different pH levels (4, 7, 10)
```

```
# pH 4 at (A x B)
ph4_df <- emily %>%
  filter(PH == 4) %>%
  mutate(Temperature = as.factor(Temperature),
         Time = as.factor(Time))

# Create the plot
ggplot(ph4_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, Time))) +
  geom_boxplot(aes(fill = Time), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = Time, color = Time),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") + # Works properly now
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and Time at pH = 4", x = 'Temperature',
       y = 'Absorbance (ABS)', color = 'Time', fill = "Time") +
  theme_minimal()
```

```

# pH 7 at (A x B)
ph7_df <- emily %>%
  filter(PH == 7) %>%
  mutate(Temperature = as.factor(Temperature),
         Time = as.factor(Time))

# Create the plot
ggplot(ph7_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, Time))) +
  geom_boxplot(aes(fill = Time), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = Time, color = Time),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") + # Works properly now
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and Time at pH = 7", x = 'Temperature',
       y = 'Absorbance (ABS)', color = 'Time', fill = "Time") +
  theme_minimal()

# pH 10 at (A x B)
ph10_df <- emily %>%
  filter(PH == 10) %>%
  mutate(Temperature = as.factor(Temperature),
         Time = as.factor(Time))

# Create the plot
ggplot(ph10_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, Time))) +
  geom_boxplot(aes(fill = Time), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = Time, color = Time),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") + # Works properly now
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and Time at pH = 10", x = 'Temperature',
       y = 'Absorbance (ABS)', color = 'Time', fill = "Time") +
  theme_minimal()

# Interaction for AxC at different Time levels (20, 40, 60)

# Time = 20 at (A x C)
time20_df <- emily %>% filter(Time == 20) %>%
  mutate(Temperature = as.factor(Temperature),
         PH = as.factor(PH))

ggplot(time20_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and pH at Time = 20", x = 'Temperature',
       y = 'Absorbance (ABS)', theme_minimal() )

# Time = 40 at (A x C)
time40_df <- emily %>% filter(Time == 40) %>%
  mutate(Temperature = as.factor(Temperature),
         PH = as.factor(PH))

ggplot(time40_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and pH at Time = 40", x = 'Temperature',
       y = 'Absorbance (ABS)', theme_minimal() )

```

```

# Time = 60 at (A x C)
time60_df <- emily %>% filter(Time == 60) %>%
  mutate(Temperature = as.factor(Temperature),
         PH = as.factor(PH))

ggplot(time60_df, aes(x = Temperature, y = ABS, group = interaction(Temperature, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Temperature),
              size = 1.2, fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Temperature and pH at Time = 60", x = 'Temperature',
       y = 'Absorbance (ABS)', theme_minimal() )

# Temperature = 50 at (B x C)
temperature50_df <- emily %>% filter(Temperature == 50) %>%
  mutate(Time = as.factor(Time),
         PH = as.factor(PH))

ggplot(temperature50_df, aes(x = Time, y = ABS, group = interaction(Time, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Time), size = 1.2,
              fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Time and pH at Temperature = 50°C", x = 'Time',
       y = 'Absorbance (ABS)', theme_minimal() )

# Temperature = 80 at (B x C)
temperature80_df <- emily %>% filter(Temperature == 80) %>%
  mutate(Time = as.factor(Time),
         PH = as.factor(PH))

ggplot(temperature80_df, aes(x = Time, y = ABS, group = interaction(Time, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Time), size = 1.2,
              fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Time and pH at Temperature = 80°C", x = 'Time',
       y = 'Absorbance (ABS)', theme_minimal() )

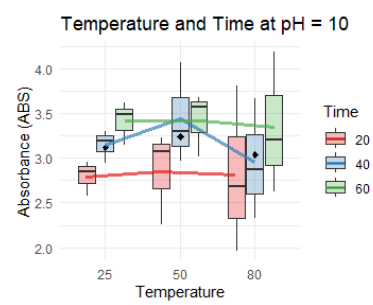
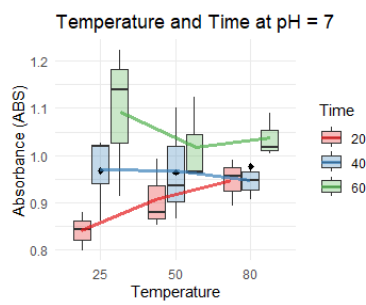
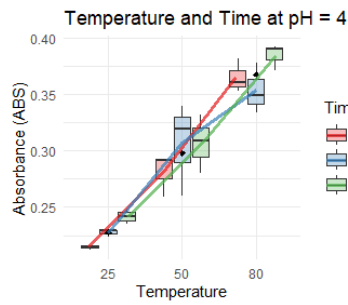
# Interaction for BxC at different Temperature levels (25, 50, 80)

# Temperature = 25 at (B x C)
temperature25_df <- emily %>% filter(Temperature == 25) %>%
  mutate(Time = as.factor(Time),
         PH = as.factor(PH))

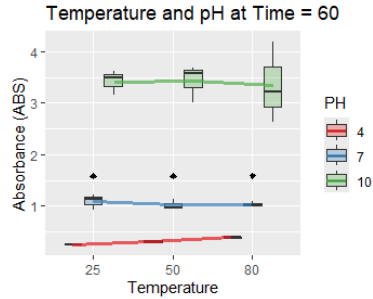
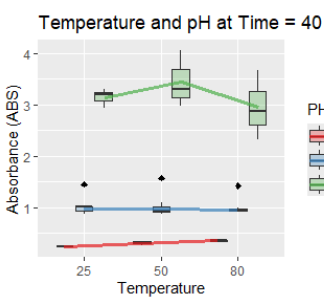
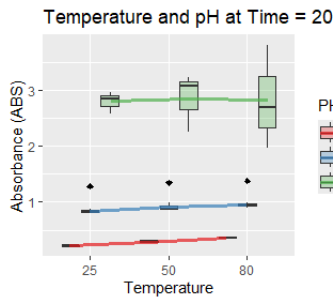
ggplot(temperature25_df, aes(x = Time, y = ABS, group = interaction(Time, PH))) +
  geom_boxplot(aes(fill = PH), alpha = 0.3) +
  stat_summary(fun = mean, geom = "point", aes(group = Time), size = 1.2,
              fill = "black", shape = 23) +
  stat_summary(fun = mean, geom = "line", aes(group = PH, color = PH),
              size = 1.2, alpha = 0.7, position = position_dodge(width = 0.8)) +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  labs(title = "Time and pH at Temperature = 25°C", x = 'Time',
       y = 'Absorbance (ABS)', theme_minimal() )

```

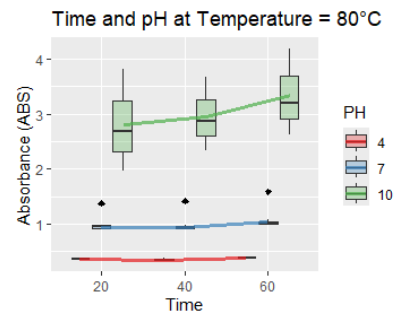
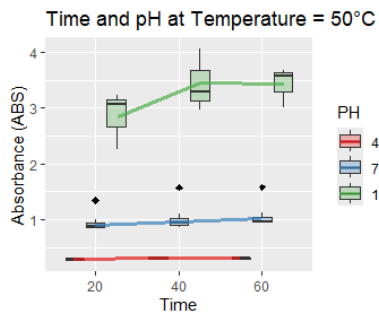
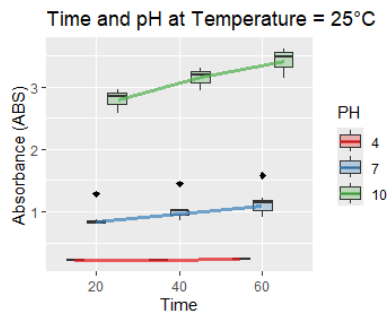
## A x B at C (pH = 4,7,10)



## A x C at B (Time = 20, 40, 60)



## B X C at A (Temperature = 25, 50, 80)



	Temperature	Time	PH	Mean	SD
	<fct>	<fct>	<fct>	<dbl>	<dbl>
1	25	20	4	0.215	0.00187
2	25	20	7	0.840	0.0409
3	25	20	10	2.80	0.199
4	25	40	4	0.227	0.00281
5	25	40	7	0.969	0.0932
6	25	40	10	3.15	0.182
7	25	60	4	0.242	0.00695
8	25	60	7	1.09	0.162
9	25	60	10	3.42	0.244
10	50	20	4	0.281	0.0189
11	50	20	7	0.907	0.0747
12	50	20	10	2.85	0.527
13	50	40	4	0.307	0.0414
14	50	40	7	0.968	0.122
15	50	40	10	3.45	0.565
16	50	60	4	0.307	0.0261
17	50	60	7	1.02	0.0924
18	50	60	10	3.42	0.361
19	80	20	4	0.365	0.0148
20	80	20	7	0.946	0.0496
21	80	20	10	2.82	0.930
22	80	40	4	0.354	0.0224
23	80	40	7	0.946	0.0384
24	80	40	10	2.96	0.676
25	80	60	4	0.384	0.0119
26	80	60	7	1.04	0.0457
27	80	60	10	3.34	0.793

## Part 2: Three-Way Anova

### CONDUCT THE THREE-WAY FACTORIAL DESIGN/ANOVA

```

emily <- read.csv("Emily_FF_clean.csv")
#to remove the variable "X" which just indexed our observations (not needed)
emily <- emily[, !names(emily) %in% "X"] # Remove a column
emily$PH = as.factor(emily$PH)
emily$Temperature = as.factor(emily$Temperature)
emily$Time = as.factor(emily$Time)
attach(emily)
three.way <- aov(ABS ~ Temperature*Time*PH)
anova(three.way)

## Analysis of Variance Table
##
## Response: ABS
##
##           Df Sum Sq Mean Sq F value Pr(>F)
## Temperature  2  0.054   0.027   0.2535 0.77701
## Time         2  0.844   0.422   3.9494 0.02507 *
## PH           2 118.568  59.284  555.0739 < 2e-16 ***
## Temperature:Time  4  0.109   0.027   0.2546 0.90563
## Temperature:PH    4  0.218   0.055   0.5105 0.72820
## Time:PH           4  0.767   0.192   1.7945 0.14337
## Temperature:Time:PH  8  0.117   0.015   0.1374 0.99713
## Residuals      54  5.767   0.107
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Full model
model <- lm(ABS ~ Temperature*Time*PH, data = emily)
model

```



```

Call:
lm(formula = ABS ~ Temperature * Time * PH, data = emily)

Coefficients:
(Intercept)          Temperature50
0.2145607          0.0660780
Temperature80          Time40
0.1506840          0.0127260
Time60          PH7
0.0271547          0.6253713
PH10          Temperature50:Time40
2.5814960          0.0132777
Temperature80:Time40          Temperature50:Time60
-0.0244123          -0.0006990
Temperature80:Time60          Temperature50:PH7
-0.0079133          0.0007713
Temperature80:PH7          Temperature50:PH10
-0.0445547          -0.0101180
Temperature80:PH10          Time40:PH7
-0.1273940          0.1163607
Time60:PH7          Time40:PH10
0.2255213          0.3377473
Time60:PH10          Temperature50:Time40:PH7
0.5919187          -0.0814690
Temperature80:Time40:PH7          Temperature50:Time60:PH7
-0.1047690          -0.1426277
Temperature80:Time60:PH7          Temperature50:Time40:PH10
-0.1543180          0.2306790
Temperature80:Time40:PH10          Temperature50:Time60:PH10
-0.1882810          -0.0471643
Temperature80:Time60:PH10
-0.0865033

# test the assumptions

#Residuals
resd = residuals(model)
resd

plot(fitted(model), resd, xlab = "fitted values",
     ylab = "residuals", main = "Full Model Residuals")
abline(h=0)

# Residuals boxplot
boxplot(resd, main = "Full Model Residuals", ylab = "Residuals")

# Residuals histogram
hist(resd, main = "Histogram of Residuals", ylab = "Residuals")

# normality (shapiro-wilk test) normal vs not normal
shapiro.test(resd)

# constant variance (Breusch-Pagan test) constant vs non-constant
bptest(model)

# randomness test (Durbin-Watson test) random vs not random
dwtest(model)

# non normality
qqPlot(resd)          #qqnorm(resd)
qqline(resd)

```

```

> shapiro.test(resd)

      shapiro-wilk normality test

data:  resd
W = 0.79947, p-value = 3.926e-09

> # constant variance (Breusch-Pagan test) constant vs non-constant
> bptest(model)

      studentized Breusch-Pagan test

data:  model
BP = 49.889, df = 26, p-value = 0.003242

> # randomness test (Durbin-Watson test) random vs not random
> dwtest(model)

      Durbin-Watson test

data:  model
DW = 2.4551, p-value = 0.1517
alternative hypothesis: true autocorrelation is greater than 0

```

```

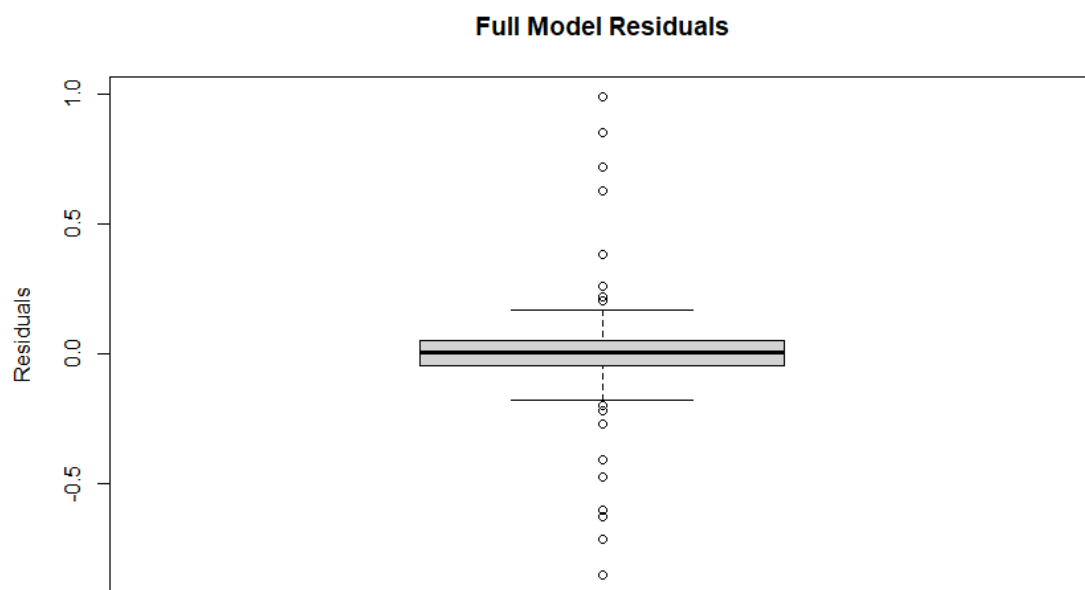
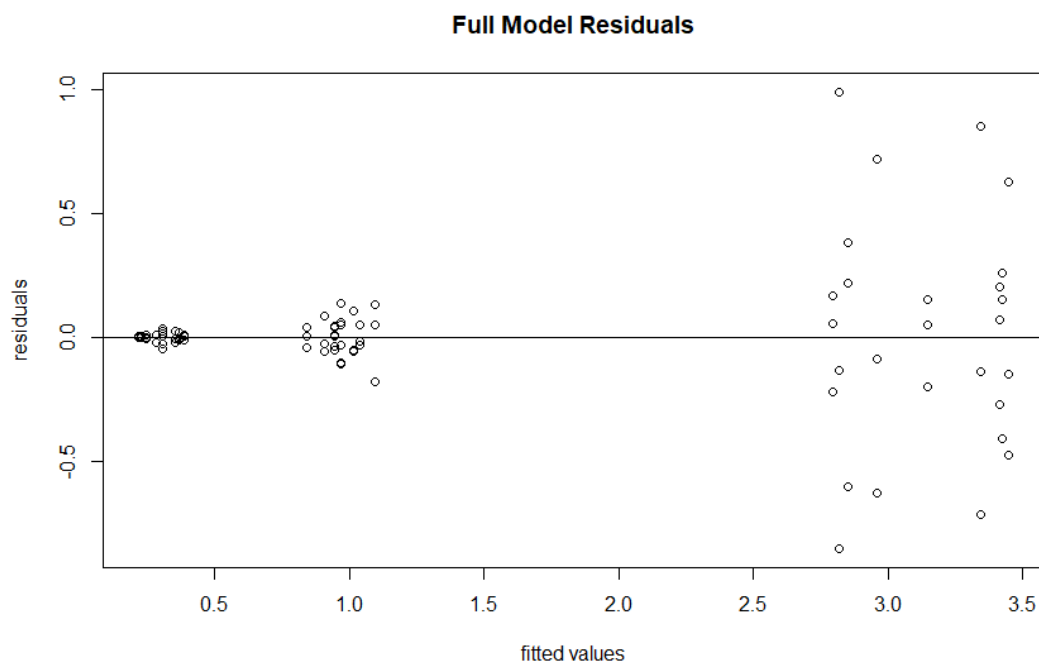
      2 way interaction
      PH * Time (none)
      PH * Temperature (none)
      time * temp (none)
      Time (only difference is with time at 60-20)
      TukeyHSD(model, "Time")

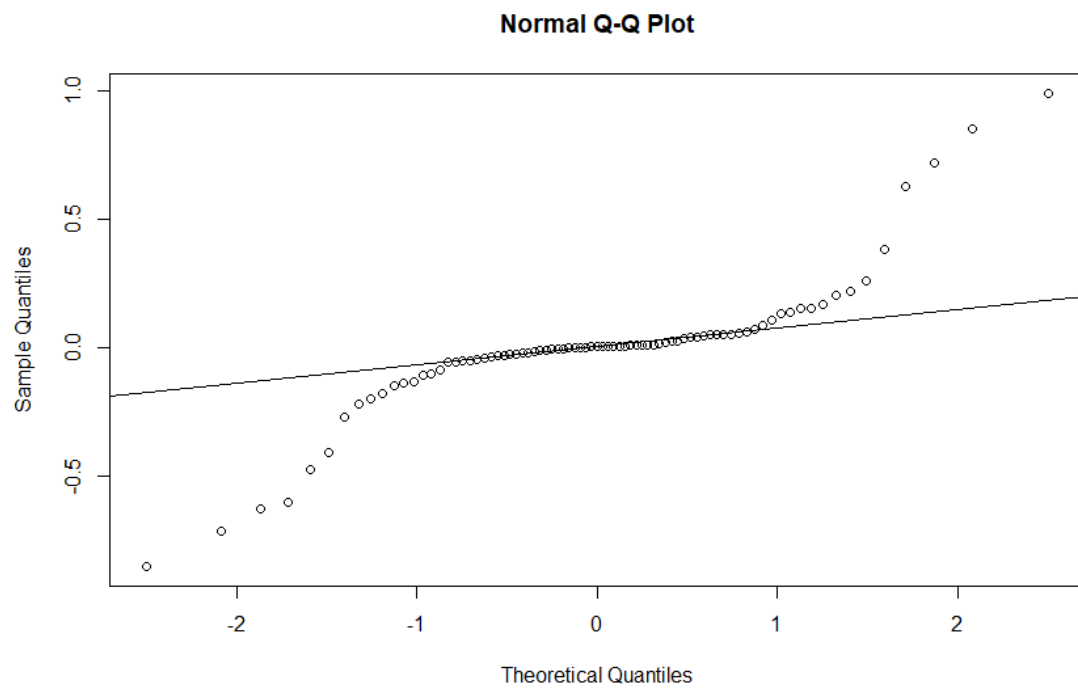
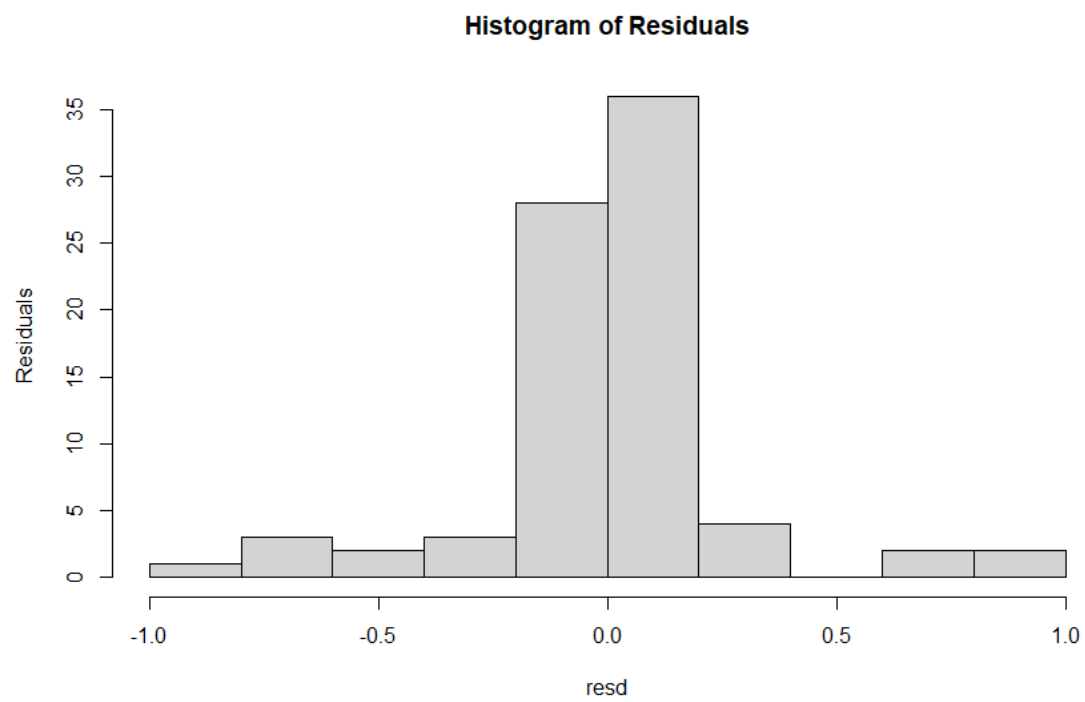
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = ABS ~ Temperature * Time * PH, data = df)
##
## $Time
##      diff      lwr      upr      p adj
## 40-20 0.1444016 -0.0699573 0.3587604 0.2446185
## 60-20 0.2489180  0.0345591 0.4632768 0.0191513
## 60-40 0.1045164 -0.1098425 0.3188753 0.4730414

      PH (all significant)
      TukeyHSD(model, "PH")

## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = ABS ~ Temperature * Time * PH, data = df)
##
## $PH
##      diff      lwr      upr p adj
## 7-4  0.6710505 0.4566916 0.8854093    0
## 10-4 2.8354063 2.6210474 3.0497651    0
## 10-7 2.1643558 1.9499969 2.3787146    0

```





## Part 3: Non-Parametric Testing

### Try non-parametric

```
library(ARTool)
```

```
## Warning: package 'ARTool' was built under R version 4.4.3
```

#### Fit Model

```
ART_anova <- art(ABS ~ PH*Temperature*Time, df)
ART_anova
```

Aligned Rank Transform of Factorial Model

Call:

```
art(formula = ABS ~ PH * Temperature * Time, data = emily)
```

Column sums of aligned responses (should all be ~0):

	PH	Temperature	Time	PH:Temperature	PH:Time
	0	0	0	0	0
Temperature:Time	0	0			

F values of ANOVAS on aligned responses not of interest (should all be ~0):

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	0	0	0	0

```
anova(ART_anova)
```

```
## Analysis of Variance of Aligned Rank Transformed Data
##
## Table Type: Anova Table (Type III tests)
## Model: No Repeated Measures (lm)
## Response: art(ABS)
##
##              Df Df.res F value    Pr(>F)
## 1 PH              2      54 221.3338 < 2.22e-16 ***
## 2 Temperature      2      54  2.1719  0.123815
## 3 Time              2      54 13.4782 1.7858e-05 ***
## 4 PH:Temperature   4      54  2.6467  0.043171 *
## 5 PH:Time          4      54  4.5500  0.003066 **
## 6 Temperature:Time 4      54  1.2940  0.283961
## 7 PH:Temperature:Time 8      54  0.3752  0.929217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
# we get 4 we want to look at
# PH
# Time
# PH:Temperature
# PH:Time
```

#### Looking at the relationship (Post-Hoc for ART ANOVA)

```
Alt_PH = art.con(ART_anova,"PH",adjust= "bonferroni")
```

#### PH

```
## NOTE: Results may be misleading due to involvement in interactions
Alt_PH
```

```
## contrast estimate SE df t.ratio p.value
## PH4 - PH7      -27 2.57 54 -10.520 <.0001
## PH4 - PH10     -54 2.57 54 -21.040 <.0001
## PH7 - PH10     -27 2.57 54 -10.520 <.0001
##
## Results are averaged over the levels of: Temperature, Time
## P value adjustment: bonferroni method for 3 tests
```

```

print("Statiscally significant")

## [1] "Statiscally significant"
SS = contrast_results[contrast_results["p.value"] < 0.05,]
print(SS)

## contrast estimate SE df t.ratio p.value
## PH4 - PH7 -27 2.56658 54 -10.520 <.0001
## PH4 - PH10 -54 2.56658 54 -21.040 <.0001
## PH7 - PH10 -27 2.56658 54 -10.520 <.0001
##
## Results are averaged over the levels of: Temperature, Time
## P value adjustment: bonferroni method for 3 tests
print("-----")

## [1] "-----"
print("NOT Statiscally significant")

## [1] "NOT Statiscally significant"
NSS = contrast_results[contrast_results["p.value"] > 0.05,]
print(NSS)

##
##
##
## Results are averaged over the levels of: Temperature, Time
## P value adjustment: bonferroni method for 3 tests

Alt_Time = art.con(ART_anova,"Time",adjust= "bonferroni")

Time

## NOTE: Results may be misleading due to involvement in interactions
Alt_Time

## contrast estimate SE df t.ratio p.value
## Time20 - Time40 -16.6 6.14 54 -2.695 0.0281
## Time20 - Time60 -31.9 6.14 54 -5.191 <.0001
## Time40 - Time60 -15.3 6.14 54 -2.496 0.0470
##
## Results are averaged over the levels of: PH, Temperature
## P value adjustment: bonferroni method for 3 tests
contrast_results = as.data.frame(Alt_Time)

```

[ See next page for significant values of time]

```

SS = contrast_results[contrast_results["p.value"] < 0.05,]
print(SS)

## contrast      estimate      SE df t.ratio p.value
## Time20 - Time40 -16.55556 6.14348 54 -2.695 0.0281
## Time20 - Time60 -31.88889 6.14348 54 -5.191 <.0001
## Time40 - Time60 -15.33333 6.14348 54 -2.496 0.0470
##
## Results are averaged over the levels of: PH, Temperature
## P value adjustment: bonferroni method for 3 tests
print("-----")

## [1] "-----"
print("NOT Statiscally significant")

## [1] "NOT Statiscally significant"
NSS = contrast_results[contrast_results["p.value"] > 0.05,]
print(NSS)

##
##
##
## Results are averaged over the levels of: PH, Temperature
## P value adjustment: bonferroni method for 3 tests

Alt_PH_Temp = art.con(ART_anova,"PH:Temperature",adjust= "bonferroni")

PH Temp
## NOTE: Results may be misleading due to involvement in interactions
Alt_PH_Temp

## contrast      estimate      SE df t.ratio p.value
## 10,25 - 10,50 -1.556 3.72 54 -0.418 1.0000
## 10,25 - 10,80 1.222 3.72 54 0.329 1.0000
## 10,25 - 4,25 62.889 3.72 54 16.909 <.0001
## 10,25 - 4,50 53.889 3.72 54 14.489 <.0001
## 10,25 - 4,80 44.889 3.72 54 12.069 <.0001
## 10,25 - 7,25 26.444 3.72 54 7.110 <.0001
## 10,25 - 7,50 28.444 3.72 54 7.648 <.0001
## 10,25 - 7,80 25.778 3.72 54 6.931 <.0001
## 10,50 - 10,80 2.778 3.72 54 0.747 1.0000
## 10,50 - 4,25 64.444 3.72 54 17.327 <.0001
## 10,50 - 4,50 55.444 3.72 54 14.907 <.0001
## 10,50 - 4,80 46.444 3.72 54 12.487 <.0001
## 10,50 - 7,25 28.000 3.72 54 7.528 <.0001
## 10,50 - 7,50 30.000 3.72 54 8.066 <.0001
## 10,50 - 7,80 27.333 3.72 54 7.349 <.0001
## 10,80 - 4,25 61.667 3.72 54 16.580 <.0001
## 10,80 - 4,50 52.667 3.72 54 14.160 <.0001
## 10,80 - 4,80 43.667 3.72 54 11.741 <.0001
## 10,80 - 7,25 25.222 3.72 54 6.781 <.0001

## 10,80 - 7,50 27.222 3.72 54 7.319 <.0001
## 10,80 - 7,80 24.556 3.72 54 6.602 <.0001
## 4,25 - 4,50 -9.000 3.72 54 -2.420 0.6813
## 4,25 - 4,80 -18.000 3.72 54 -4.840 0.0004
## 4,25 - 7,25 -36.444 3.72 54 -9.799 <.0001
## 4,25 - 7,50 -34.444 3.72 54 -9.261 <.0001
## 4,25 - 7,80 -37.111 3.72 54 -9.978 <.0001
## 4,50 - 4,80 -9.000 3.72 54 -2.420 0.6813
## 4,50 - 7,25 -27.444 3.72 54 -7.379 <.0001
## 4,50 - 7,50 -25.444 3.72 54 -6.841 <.0001
## 4,50 - 7,80 -28.111 3.72 54 -7.558 <.0001
## 4,80 - 7,25 -18.444 3.72 54 -4.959 0.0003
## 4,80 - 7,50 -16.444 3.72 54 -4.421 0.0017
## 4,80 - 7,80 -19.111 3.72 54 -5.138 0.0001
## 7,25 - 7,50 2.000 3.72 54 0.538 1.0000
## 7,25 - 7,80 -0.667 3.72 54 -0.179 1.0000
## 7,50 - 7,80 -2.667 3.72 54 -0.717 1.0000
##
## Results are averaged over the levels of: Time
## P value adjustment: bonferroni method for 36 tests

```



(Non)Significant values of Temperature-pH (A x C)

```
## [1] "-----"
print("Statiscally significant")

## [1] "Statiscally significant"
SS = contrast_results[contrast_results["p.value"] < 0.05,]
print(SS)
```

contrast	estimate	SE	df	t.ratio	p.value
10,25 - 4,25	62.88889	3.719319	54	16.909	<.0001
10,25 - 4,50	53.88889	3.719319	54	14.489	<.0001
10,25 - 4,80	44.88889	3.719319	54	12.069	<.0001
10,25 - 7,25	26.44444	3.719319	54	7.110	<.0001
10,25 - 7,50	28.44444	3.719319	54	7.648	<.0001
10,25 - 7,80	25.77778	3.719319	54	6.931	<.0001
10,50 - 4,25	64.44444	3.719319	54	17.327	<.0001
10,50 - 4,50	55.44444	3.719319	54	14.907	<.0001
10,50 - 4,80	46.44444	3.719319	54	12.487	<.0001
10,50 - 7,25	28.00000	3.719319	54	7.528	<.0001
10,50 - 7,50	30.00000	3.719319	54	8.066	<.0001
10,50 - 7,80	27.33333	3.719319	54	7.349	<.0001
10,80 - 4,25	61.66667	3.719319	54	16.580	<.0001
10,80 - 4,50	52.66667	3.719319	54	14.160	<.0001
10,80 - 4,80	43.66667	3.719319	54	11.741	<.0001
10,80 - 7,25	25.22222	3.719319	54	6.781	<.0001
10,80 - 7,50	27.22222	3.719319	54	7.319	<.0001
10,80 - 7,80	24.55556	3.719319	54	6.602	<.0001
4,25 - 4,80	-18.00000	3.719319	54	-4.840	0.0004
4,25 - 7,25	-36.44444	3.719319	54	-9.799	<.0001
4,25 - 7,50	-34.44444	3.719319	54	-9.261	<.0001
4,25 - 7,80	-37.11111	3.719319	54	-9.978	<.0001
4,50 - 7,25	-27.44444	3.719319	54	-7.379	<.0001
4,50 - 7,50	-25.44444	3.719319	54	-6.841	<.0001
4,50 - 7,80	-28.11111	3.719319	54	-7.558	<.0001
4,80 - 7,25	-18.44444	3.719319	54	-4.959	0.0003
4,80 - 7,50	-16.44444	3.719319	54	-4.421	0.0017
4,80 - 7,80	-19.11111	3.719319	54	-5.138	0.0001

```
##
## Results are averaged over the levels of: Time
## P value adjustment: bonferroni method for 36 tests

print("NOT Statiscally significant")

## [1] "NOT Statiscally significant"
NSS = contrast_results[contrast_results["p.value"] > 0.05,]
print(NSS)
```

contrast	estimate	SE	df	t.ratio	p.value
10,25 - 10,50	-1.555556	3.719319	54	-0.418	1.0000
10,25 - 10,80	1.222222	3.719319	54	0.329	1.0000
10,50 - 10,80	2.777778	3.719319	54	0.747	1.0000
4,25 - 4,50	-9.000000	3.719319	54	-2.420	0.6813
4,50 - 4,80	-9.000000	3.719319	54	-2.420	0.6813
7,25 - 7,50	2.000000	3.719319	54	0.538	1.0000
7,25 - 7,80	-0.666667	3.719319	54	-0.179	1.0000
7,50 - 7,80	-2.666667	3.719319	54	-0.717	1.0000

```
##
## Results are averaged over the levels of: Time
## P value adjustment: bonferroni method for 36 tests
```



## PH Time

## NOTE: Results may be misleading due to involvement in interactions

Alt\_PH\_Time

##	contrast	estimate	SE	df	t.ratio	p.value
##	10,20 - 10,40	-5.00	3.71	54	-1.346	1.0000
##	10,20 - 10,60	-8.33	3.71	54	-2.243	1.0000
##	10,20 - 4,20	54.44	3.71	54	14.656	<.0001
##	10,20 - 4,40	49.89	3.71	54	13.429	<.0001
##	10,20 - 4,60	44.33	3.71	54	11.934	<.0001
##	10,20 - 7,20	28.67	3.71	54	7.717	<.0001
##	10,20 - 7,40	22.78	3.71	54	6.131	<.0001
##	10,20 - 7,60	16.22	3.71	54	4.367	0.0021
##	10,40 - 10,60	-3.33	3.71	54	-0.897	1.0000
##	10,40 - 4,20	59.44	3.71	54	16.002	<.0001
##	10,40 - 4,40	54.89	3.71	54	14.775	<.0001
##	10,40 - 4,60	49.33	3.71	54	13.280	<.0001
##	10,40 - 7,20	33.67	3.71	54	9.063	<.0001

##	10,40 - 7,40	27.78	3.71	54	7.477	<.0001
##	10,40 - 7,60	21.22	3.71	54	5.713	<.0001
##	10,60 - 4,20	62.78	3.71	54	16.899	<.0001
##	10,60 - 4,40	58.22	3.71	54	15.673	<.0001
##	10,60 - 4,60	52.67	3.71	54	14.177	<.0001
##	10,60 - 7,20	37.00	3.71	54	9.960	<.0001
##	10,60 - 7,40	31.11	3.71	54	8.375	<.0001
##	10,60 - 7,60	24.56	3.71	54	6.610	<.0001
##	4,20 - 4,40	-4.56	3.71	54	-1.226	1.0000
##	4,20 - 4,60	-10.11	3.71	54	-2.722	0.0140
##	4,20 - 7,20	-25.78	3.71	54	-6.939	<.0001
##	4,20 - 7,40	-31.67	3.71	54	-8.524	<.0001
##	4,20 - 7,60	-38.22	3.71	54	-10.289	<.0001
##	4,40 - 4,60	-5.56	3.71	54	-1.495	1.0000
##	4,40 - 7,20	-21.22	3.71	54	-5.713	<.0001
##	4,40 - 7,40	-27.11	3.71	54	-7.298	<.0001
##	4,40 - 7,60	-33.67	3.71	54	-9.063	<.0001
##	4,60 - 7,20	-15.67	3.71	54	-4.217	0.0034
##	4,60 - 7,40	-21.56	3.71	54	-5.802	<.0001
##	4,60 - 7,60	-28.11	3.71	54	-7.567	<.0001
##	7,20 - 7,40	-5.89	3.71	54	-1.585	1.0000
##	7,20 - 7,60	-12.44	3.71	54	-3.350	0.0533
##	7,40 - 7,60	-6.56	3.71	54	-1.765	1.0000

## Results are averaged over the levels of: Temperature

## P value adjustment: bonferroni method for 36 tests

(non)Significant values of Time-pH (B x C)

```
## [1] "-----"
print("Statiscally significant")

## [1] "Statiscally significant"
SS = contrast_results[contrast_results["p.value"] < 0.05,]
print(SS)

## contrast      estimate      SE df t.ratio p.value
## 10,20 - 4,20  54.44444 3.714891 54   14.656 <.0001
## 10,20 - 4,40  49.88889 3.714891 54   13.429 <.0001
## 10,20 - 4,60  44.33333 3.714891 54   11.934 <.0001
## 10,20 - 7,20  28.66667 3.714891 54    7.717 <.0001
## 10,20 - 7,40  22.77778 3.714891 54    6.131 <.0001
## 10,20 - 7,60  16.22222 3.714891 54    4.367 0.0021
## 10,40 - 4,20  59.44444 3.714891 54   16.002 <.0001
## 10,40 - 4,40  54.88889 3.714891 54   14.775 <.0001
## 10,40 - 4,60  49.33333 3.714891 54   13.280 <.0001
## 10,40 - 7,20  33.66667 3.714891 54    9.063 <.0001
## 10,40 - 7,40  27.77778 3.714891 54    7.477 <.0001
## 10,40 - 7,60  21.22222 3.714891 54    5.713 <.0001
## 10,60 - 4,20  62.77778 3.714891 54   16.899 <.0001
## 10,60 - 4,40  58.22222 3.714891 54   15.673 <.0001
## 10,60 - 4,60  52.66667 3.714891 54   14.177 <.0001

## 10,60 - 7,20  37.00000 3.714891 54    9.960 <.0001
## 10,60 - 7,40  31.11111 3.714891 54    8.375 <.0001
## 10,60 - 7,60  24.55556 3.714891 54    6.610 <.0001
## 4,20 - 7,20  -25.77778 3.714891 54   -6.939 <.0001
## 4,20 - 7,40  -31.66667 3.714891 54   -8.524 <.0001
## 4,20 - 7,60  -38.22222 3.714891 54  -10.289 <.0001
## 4,40 - 7,20  -21.22222 3.714891 54   -5.713 <.0001
## 4,40 - 7,40  -27.11111 3.714891 54   -7.298 <.0001
## 4,40 - 7,60  -33.66667 3.714891 54   -9.063 <.0001
## 4,60 - 7,20  -15.66667 3.714891 54   -4.217 0.0034
## 4,60 - 7,40  -21.55556 3.714891 54   -5.802 <.0001
## 4,60 - 7,60  -28.11111 3.714891 54   -7.567 <.0001
##
## Results are averaged over the levels of: Temperature
## P value adjustment: bonferroni method for 36 tests
print("-----")

## [1] "-----"
print("NOT Statiscally significant")

## [1] "NOT Statiscally significant"
NSS = contrast_results[contrast_results["p.value"] > 0.05,]
print(NSS)

## contrast      estimate      SE df t.ratio p.value
## 10,20 - 10,40  -5.000000 3.714891 54   -1.346 1.0000
## 10,20 - 10,60  -8.333333 3.714891 54   -2.243 1.0000
## 10,40 - 10,60  -3.333333 3.714891 54   -0.897 1.0000
## 4,20 - 4,40   -4.555556 3.714891 54   -1.226 1.0000
## 4,20 - 4,60  -10.111111 3.714891 54   -2.722 0.3140
## 4,40 - 4,60   -5.555556 3.714891 54   -1.495 1.0000
## 7,20 - 7,40   -5.888889 3.714891 54   -1.585 1.0000
## 7,20 - 7,60  -12.444444 3.714891 54   -3.350 0.0533
## 7,40 - 7,60   -6.555556 3.714891 54   -1.765 1.0000
##
## Results are averaged over the levels of: Temperature
## P value adjustment: bonferroni method for 36 tests
```