

# Experimental Design - Homework 2

Jonathan Chandra / S5161533

## remark

The code used to output the image and tables are attached in the code listing section. The codes also attached as zip file in the same Brightspace submission box.

## Problem 1.a

Using the corrosion data from BH2 library, we compute the mean and the SD of the resistance for each heats by coating combination.

	heats	coating	mean_resistance	sd_resistance
1	T360	C1	50.00	24.04
2	T360	C2	40.50	45.96
3	T360	C3	64.50	26.16
4	T360	C4	71.50	24.75
5	T370	C1	102.50	53.03
6	T370	C2	116.50	36.06
7	T370	C3	104.00	24.04
8	T370	C4	118.00	45.25
9	T380	C1	131.50	33.23
10	T380	C2	113.50	19.09
11	T380	C3	118.50	40.31
12	T380	C4	182.50	41.72

**Table 1.** Results for 1.a

Based on Table 1 we can observe the following:

1. For the T360 heat level, Coating C4 has the highest mean resistance, while Coating C2 has the lowest.
2. For the T370 heat level, Coating C4 also shows the highest mean resistance, with Coating C1 having the lowest.
3. For the T380 heat level, Coating C4 again shows the highest mean resistance, but this time Coating C2 has the lowest.

The standard deviation also varies between each combination of heats and coating. This might affect the precision of the mean resistance values. For instance, the combination T360-C2 has a high standard deviation, which indicates a high variability in the resistance for this combination.

## Problem 1.b

Now, we analyse the data by two-factor analysis of variance and use F-testing to investigate for the presence of main and interaction effects. We use `aov(resistance ~ heats*coating)` to define fitting model.

```
> summary(fit)
      Df Sum Sq Mean Sq F value    Pr(>F)
heats        2  26519   13260  10.226 0.00256 ***
coating      3   4289    1430   1.103 0.38602
heats:coating 6   3270     545   0.420 0.85180
Residuals   12  15560    1297
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Figure 1.** AOV output of the corrosion data

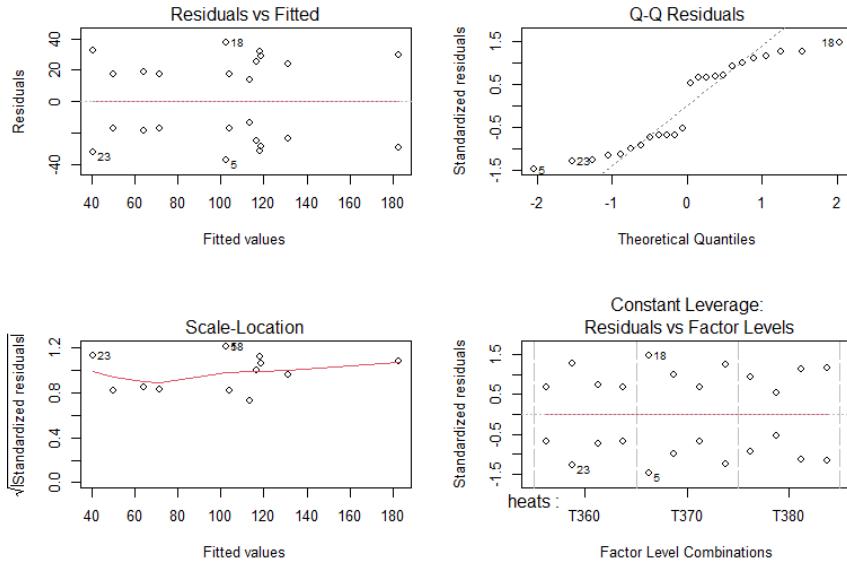
Based on Figure 1 we can observe the following:

1. For heats, the p-value is 0.00256 which is less than 0.05, implying that the effect of heats on resistance is statistically significant. The F-value is 10.226, indicating a relatively strong effect. Thus, the level of heat treatment significantly influences corrosion resistance.
2. For coating, the p-value is 0.38602, which is greater than 0.05. This suggests that the coating effect on resistance is not statistically significant at the 5% level. The type of coating does not significantly affect corrosion resistance, based on the model.
3. For the interaction between heat and coating, The p-value is 0.85180, also greater than 0.05, indicating that the interaction effect between heats and coating on resistance is not statistically significant at the 5% level. This suggests that the effect of heat treatment on corrosion resistance does not depend on the type of coating, and vice versa, according to the model.

In conclusion, based to this two-way ANOVA model, the level of heat treatment (heats) has a significant effect on corrosion resistance, while the type of coating (coating) and the interaction between heat treatment and coating (heats:coating) do not have significant effects.

### Problem 1.c

Now, we are going to investigate the diagnostics of the fit of the best model for the data. This is done by plotting the fitting model created in the previous problem.



**Figure 2.** Diagnostic plot of the fitting model

We can observe the following from Figure 2:

1. Through the Residuals vs Fitted plot, the residuals being randomly and evenly dispersed around the horizontal axis. This suggests that the model is correctly specified. There's no indication of non-linearity or omitted variables bias.
2. Through the Normal Q-Q Plot, the points mostly falling along the diagonal line which implies that the residuals are normally distributed
3. Through Scale-Location plot, the points are randomly scattered around the fitted line, it suggests that the residuals have constant variance (homoscedasticity).
4. Lastly, through the Residuals vs Leverage plot, the fact that there are no points far away from the bulk of the data suggests that there are no influential observations that could unduly affect the model estimated.

### Problem 1.d

Now, we use Tukey's HSD method to analyse the paired differences in the levels for heats.

```

> print(tukey)
Tukey multiple comparisons of means
 95% family-wise confidence level

Fit: aov.default(formula = resistance ~ heats * coating, data = corrosion.data)

$heats
      diff      lwr      upr     p adj
T370-T360 53.625  5.590333 101.65967 0.0288163
T380-T360 79.875 31.840333 127.90967 0.0021637
T380-T370 26.250 -21.784667 74.28467 0.3442858

```

**Figure 3.** Tukey HSD output of the fit

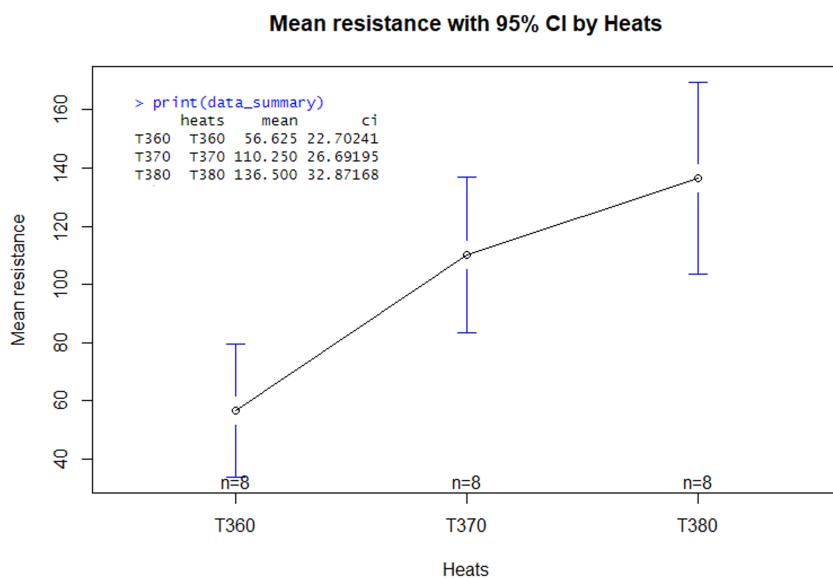
We can observe the Tukey's HSD test of the fit on comparing the means of "heats" level (T360, T370, T380) in Figure 3, and conclude the following:

1. T370-T360: The difference in means is 53.625. The 95% confidence interval of this difference is [5.590333, 101.65967]. Because this interval does not contain 0, it can be concluded that there is a statistically significant difference between the 'T370' and 'T360' levels of 'heats'. The p-value is 0.0288163, which is below the standard significance level of 0.05, confirming this significant difference.
2. T380-T360: The difference in means is 79.875. The 95% confidence interval of this difference is [31.840333, 127.90967]. Since this interval does not contain 0, there is a statistically significant difference between the 'T380' and 'T360' levels of 'heats'. The p-value is 0.0021637, which is also below 0.05, indicating a significant difference.
3. T380-T370: The difference in means is 26.25. The 95% confidence interval of this difference is [-21.784667, 74.28467]. This interval contains 0, suggesting that there is not a statistically significant difference between the 'T380' and 'T370' levels of 'heats'. The p-value is 0.3442858, which is above 0.05, supporting this lack of significant difference.

In conclusion, the mean resistance seems to be significantly different between T360 and T370, and between T360 and T380, but not between T370 and T380.

### Problem 1.e

Lastly, we construct a mean with confidence intervals plot for each of the levels of heats. This can be done directly using `plotmeans` function from `gplots` library.



**Figure 4.** Mean with confidence intervals plot

Figure 4 shows us the mean with confidence intervals plot of each of the levels of heats. We can conclude the following:

1. For the 'T360' level of 'heats', the 95% confidence interval for the mean resistance is approximately  $56.625 \pm 22.70241$ . This means that if we were to repeat the sampling process many times and calculate the confidence intervals, we would expect the true population mean resistance for 'T360' to fall within the range of approximately 33.9226 to 79.3274 in 95% of those intervals.
2. For the 'T370' level of 'heats', the 95% confidence interval for the mean resistance is approximately  $110.250 \pm 26.69195$ . This indicates that we would expect the true population mean resistance for 'T370' to lie within the range of approximately 83.1581 to 137.3419 in 95% of the intervals obtained from repeated sampling.
3. For the 'T380' level of 'heats', the 95% confidence interval for the mean resistance is approximately  $136.500 \pm 32.87168$ . This suggests that we would anticipate the true population mean resistance for 'T380' to be within the range of approximately 103.6283 to 169.3717 in 95% of the intervals derived from repeated sampling.

### Problem 2.a

We are going to test for the fixed effect of Source and Site from the Oxide data from the `nlme` library. The test will incorporate `anova` function

```
> print(anova(model_2a))
Analysis of Variance Table

Response: Thickness
          Df Sum Sq Mean Sq F value    Pr(>F)
Source     1 1830.1 1830.12 12.8221 0.000638 ***
Site       2   15.4    7.72  0.0541 0.947375
Residuals 68 9705.8 142.73
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Figure 5.** ANOVA test output

Figure 5 shows the ANOVA test output for the fixed effects of "Source" and "Site" on the "Thickness" response variable. From the output, we can draw the following conclusions:

1. Source: The p-value (0.000638) is less than the significance level of 0.05, indicating strong evidence against the null hypothesis. Therefore, there is a significant fixed effect of Source on the Thickness.
2. Site: The p-value (0.947375) is greater than the significance level of 0.05, indicating that there is not enough evidence to reject the null hypothesis. Therefore, there is no significant fixed effect of Site on the Thickness.

Overall, we can conclude that the "Source" variable has a significant effect on the "Thickness" response, while the "Site" does not have a significant effect.

### Problem 2.b

Now, we are going to estimate the model with the constant and only random lot effects and wafer. We are also going to compute the confidence intervals for the standard deviations of the random effects using `random ~ 1 | Lot/Wafer`.

```

> print(summary(model_2b))
Linear mixed-effects model fit by REML
  Data: dat
    AIC      BIC    LogLik
  462.0221 471.0728 -227.011

  Random effects:
  Formula: ~1 | Lot
            (Intercept)
  StdDev:   11.39768

  Formula: ~1 | wafer %in% Lot
            (Intercept) Residual
  StdDev:   5.988802 3.545341

  Fixed effects: Thickness ~ 1
                 value Std.Error DF t-value p-value
  (Intercept) 2000.153 4.231711 48 472.6581 0

  Standardized within-Group Residuals:
             Min       Q1       Med       Q3       Max
-1.8745889 -0.4990825  0.1046479  0.5510180  1.7921950

  Number of Observations: 72
  Number of Groups:
    Lot Wafer %in% Lot
      8          24

```

Fixed effects:			
	lower	est.	upper
(Intercept)	1991.644	2000.153	2008.661

Random Effects:			
Level:	Lot	wafer	
	lower	est.	upper
sd((Intercept))	6.390314	11.39768	20.32876
sd((Intercept))	4.064538	5.988802	8.824064

Within-group standard error:			
	lower	est.	upper
2.902638	3.545341	4.330352	

**Figure 6.** Estimated model summary and confidence interval output

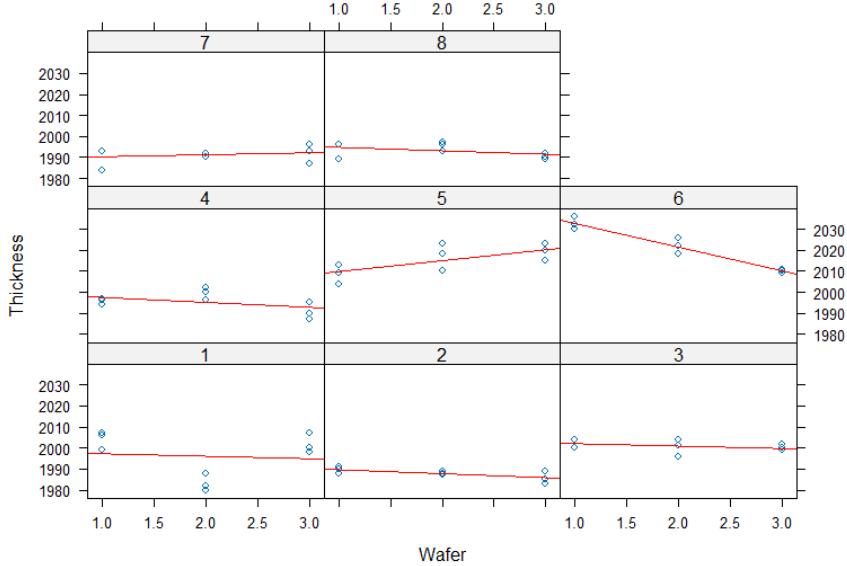
Figure 6 shows the estimated model summary and confident interval output of the model with the constant and only random lot effects and water within lot effects. We can observe the following:

1. Random lot effects: The estimated standard deviation (SD) of the random intercept for Lot is 11.39768. The 95% confidence interval for the SD of the random intercept for Lot is approximately 6.390314 to 20.32876. This means that we can be 95% confident that the true population SD falls within this range.
2. Random effects of Wafer within Lot: The estimated SD of the random intercept for Wafer within Lot is 5.988802. The 95% confidence interval for the SD of the random intercept for Wafer within Lot is approximately 4.064538 to 8.824064. This indicates that we can be 95% confident that the true population SD falls within this range.
3. The fixed intercept (constant term) is estimated to be 2000.153. The 95% confidence interval for the intercept (constant) ranges from 1991.644 to 2008.661. This means that we can be 95% confident that the true population value of the intercept falls within this range.

Overall, the estimated model suggests that there is significant variability in the intercepts across different Lots and within each Lot. The fixed intercept represents the average value of Thickness, while the random effects capture the additional variability attributed to Lot and Wafer within Lot

### Problem 2.c

Lastly, we are going to construct a plot of the Thickness measurements by lot per wafer including a line representing the predicted values from the model.



**Figure 7.** Plot of the Thickness measurements by lot per wafer

Based on the plot in Figure 7, we can conclude that the observed data points closely follow the line, which suggests that the estimated model's align well with the actual data. It can also be observed that Lot 6 has the largest wafer thickness (Wafer 1). Most of the wafer has a consistent thickness in every lot except for lot 5 and 6.

### Problem 3.a

We are going to construct a model of data with significant quadratic and interaction effect such that the yield is maximized. First we define two Factor levels (A and B in my case) with the value of  $\{-1, 0, 1\}$ . A quadratic model with interaction between two factor can be represented with:

$$\text{Yield} = \beta_0 + \beta_1 A + \beta_2 B + \beta_3 A^2 + \beta_4 B^2 + \beta_5 AB + \varepsilon$$

Where Yield is the dependent variable,  $\beta_0$  is the intercept,  $\beta_1$  and  $\beta_2$  is the coefficient of the linear terms of the factor.  $\beta_3$  and  $\beta_4$  are the coefficients for the quadratic terms,  $\beta_5$  is the coefficient for the interaction term, while  $\varepsilon$  is an error term. In this report, We assumed that  $\beta_0 = 1$ ,  $\beta_1 = 2$ ,  $\beta_2 = 3$ ,  $\beta_{3,4} = -1$ , and  $\beta_5 = -2$ .

```
> summary(model)

Call:
lm.default(formula = Yield ~ A + B + I(A^2) + I(B^2) + A:B, data = df)

Residuals:
    1     2     3     4     5     6     7     8     9 
-1.636e-16 1.449e-15 -1.285e-15 -6.516e-16 -9.407e-16 1.592e-15 8.151e-16 -5.084e-16 -3.068e-16 

Coefficients:
            Estimate Std. Error   t value Pr(>|t|)    
(Intercept) 1.000e+00 1.265e-15 7.905e+14 <2e-16 ***
A            2.000e+00 6.928e-16 2.887e+15 <2e-16 ***
B            3.000e+00 6.928e-16 4.330e+15 <2e-16 ***
I(A^2)      -1.000e+00 1.200e-15 -8.333e+14 <2e-16 ***
I(B^2)      -1.000e+00 1.200e-15 -8.333e+14 <2e-16 ***
A:B         -2.000e+00 8.486e-16 -2.357e+15 <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.697e-15 on 3 degrees of freedom
Multiple R-squared:     1, Adjusted R-squared:     1
F-statistic: 6.805e+30 on 5 and 3 DF,  p-value: < 2.2e-16

warning message:
In summary.lm(model) : essentially perfect fit: summary may be unreliable
```

**Figure 8.** Model summary of problem 3.a

Based on Figure 8, we conclude that the coefficients estimated by the model perfectly match the true coefficients used to generate the data. The estimated coefficients for the linear terms A and B are 2 and 3, respectively, which aligns with the model's design. The coefficients for the quadratic terms are -1 for both  $A^2$  and  $B^2$ , and the interaction term's coefficient is -2.

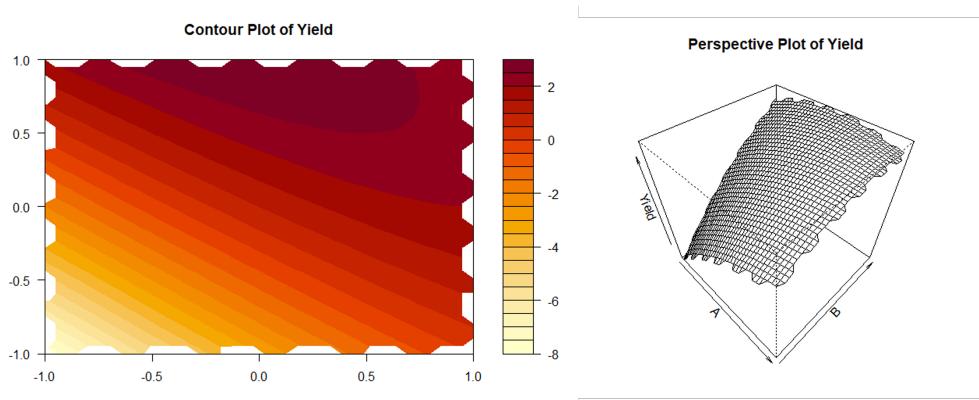
The p-values for all predictor variables are extremely small, indicating that they are all significantly associated with Yield. This is expected as we've used these predictors to generate Yield.

The model explains 100% of the variability in Yield (Multiple R-squared is 1), which is expected as we used the same model structure to both generate and fit the data.

The model shows an "essentially perfect fit". While this is an indication of overfitting or a model is just too complex, it's expected and acceptable here as we are perfectly modeling the data we just generated. One may incorporate the noise variable to "disrupt" the model.

### Problem 3.b

Now, we add a contour plot and a response perspective plot to illustrate our findings.



**Figure 9.** Contour and Perspective plot of the Yield

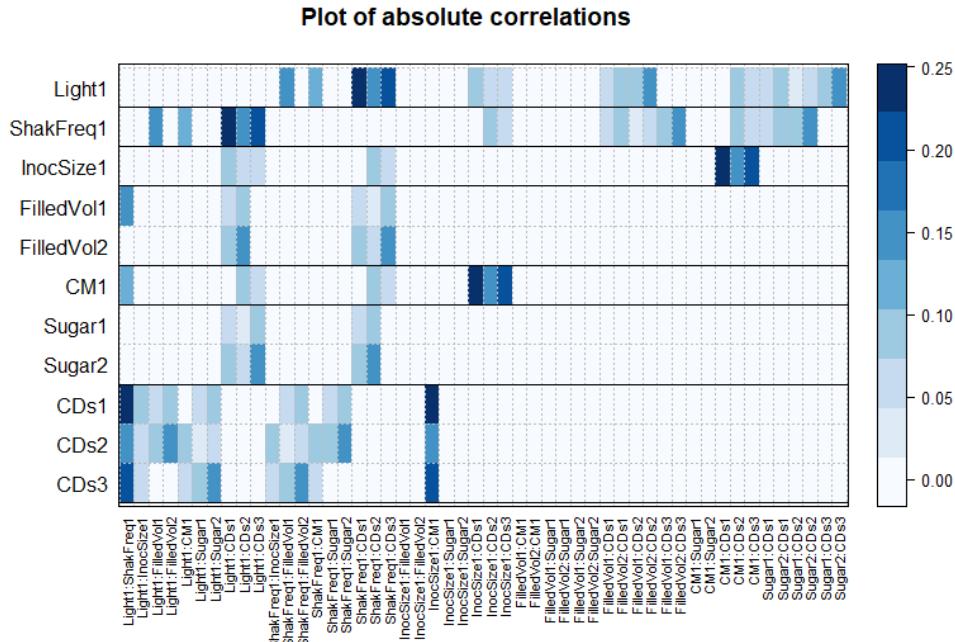
Based on Figure 9, both contour and perspective plot show the value of the yield based on a specific combination of A and B. We also find that the maximum Yield wasn't achieved in the highest value of A and B, this is due to the negative coefficients for the quadratic terms suggest that the yield first increases with each factor (A and B) and then starts decreasing after a certain point (which represents a maximum).

### Problem 4.a

We are going to use the VSGFS data from the library `DoE.base` with the data from a Taguchi design (Maybe it is Taguchi Design?). First, we should know that the generalized words of lengths 3 and 4 indicate the specific combinations of factors and levels in the design. Factors represent the independent variables being studied, and levels represent the different conditions or values of those factors. The numbers within the generalized words indicate the specific combinations of factor levels being tested in the experiment. Each digit in the generalized word represents a factor, and the value of the digit represents the level of that factor in the corresponding combination.

### Problem 4.b

Now, we are going to construct a correlation plot of the design on VSGFS dataset.



**Figure 10.** VSGFS Correlation plot

To identify the main effects confounded with the 2-way interaction effects, we'll perform a regression analysis using ANOVA.

```
> anova(lm(VSGFS, response = "Yield"))
Analysis of Variance Table

Response: Yield
          Df Sum Sq Mean Sq F value    Pr(>F)
Light      1  2133   2133.0 13.2347 0.0005729 ***
ShakFreq   1    139    139.3  0.8643 0.3562509
Inocsize   1   4638   4637.6 28.7750 1.374e-06 ***
Filledvol  2   4743   2371.5 14.7142 6.313e-06 ***
ShalCM     1    309    309.4  1.9196 0.1710281
Sugar      2  61077  30538.7 189.4819 < 2.2e-16 ***
Inocsize:CDs 3    512    170.7  1.0593 0.3731974
Residuals 60   9670   161.2
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Figure 11.** ANOVA of linear model of VSGFS dataset

Based on Figure 11, Light, InocSize, FilledVol, and Sugar are significantly associated with Yield. ShakFreq, CM, and CDs do not show significant associations with Yield. The model explains a significant portion of the variation in Yield (This was indicated by the small p-value in the Residuals row).

#### Problem 4.c

Now, we find a model for the most important main and two factor interactions for the outcome Yield. We can use `lm` function to create a linear model of VSGFS as before.

```

> summary(model_4b)
Number of observations used: 72
Formula:
yield ~ (light + shakfreq + inocsize + filledvol + CM + sugar +
cos)^2

Call:
lm.default(formula = fo, data = model.frame(fo, data = formula))

Residuals:
    Min      1Q  Median      3Q     Max 
-5.5048 -1.8139 -0.0794  1.5585  5.7869 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 104.0823   5.4600 19.063 3.43e-09 *** 
Light1       12.3339   3.6525  3.377 0.007040 **  
InocSize1    6.0439   4.5370  1.332 0.212378    
FilledvolFV0 14.7832   3.6196  4.084 0.002199 **  
FilledvolFV+ 22.6097   7.0454  3.209 0.009345 **  
CM1          -1.6973   4.3372 -0.391 0.703755    
SugarGluc   -42.9956   8.5752 -5.014 0.000526 *** 
SugarMannit  -89.6216   6.4039 -13.995 6.80e-08 *** 
CdsCD2      -12.9400   7.3076 -1.771 0.107022    
CdsCD3      -15.3933   6.8534 -2.246 0.048492 *  
CdsCD4      -33.2843   8.3924 -3.966 0.002660 **  
Light1:ShakFreq1 -0.3281   1.3729 -0.239 0.815937    
Light1:InocSize1  0.2439   1.3249  0.184 0.857611    
Light1:FilledvolFV0 7.2863   3.8725  1.882 0.089292 . 
Light1:FilledvolFV+ 6.3823   3.3895  1.883 0.089085 . 
Light1:CM1      2.3263   0.9944  2.339 0.041386 *  
Light1:sugarGluc -12.0072   3.0716 -3.909 0.002917 ** 
Light1:SugarMannit -8.6739   2.5410 -3.414 0.006618 ** 
Light1:CdsCD2   -2.9430   3.4610 -0.850 0.415030    
Light1:CdsCD3   -12.2578   3.8757 -3.163 0.010113 *  
Light1:CdsCD4   -3.0507   3.8876 -0.785 0.450796 . 
ShakFreq1:InocSize1  1.7188   1.5993  1.075 0.307225    
ShakFreq1:FilledvolFV0 -2.9600   3.5404 -0.836 0.422638 . 
ShakFreq1:FilledvolFV+ -0.9472   3.3974 -0.279 0.786070 . 
ShakFreq1:CM1      3.9550   2.0781  1.903 0.086167 . 
ShakFreq1:SugarGluc -11.3130   4.2227 -2.679 0.023128 * 
ShakFreq1:SugarMannit -11.2843   3.4403 -3.280 0.008288 ** 
ShakFreq1:CdsCD2   9.9045   4.1232  2.402 0.037179 * 
ShakFreq1:CdsCD3   8.0779   4.1910  1.927 0.082783 . 
ShakFreq1:CdsCD4   -3.6013   4.5167 -0.797 0.443780 . 
InocSize1:FilledvolFV0 0.7395   2.8786  0.257 0.802465    
InocSize1:FilledvolFV+ 3.9627   2.1021  1.885 0.088768 . 
InocSize1:CM1      -2.1678   1.0458 -2.073 0.064965 . 
InocSize1:sugarGluc 0.6338   2.9232  0.217 0.832718 . 
InocSize1:SugarMannit -1.8358   2.2830 -0.804 0.440018 . 
InocSize1:CdsCD2   -7.1062   3.1944 -2.225 0.050302 . 

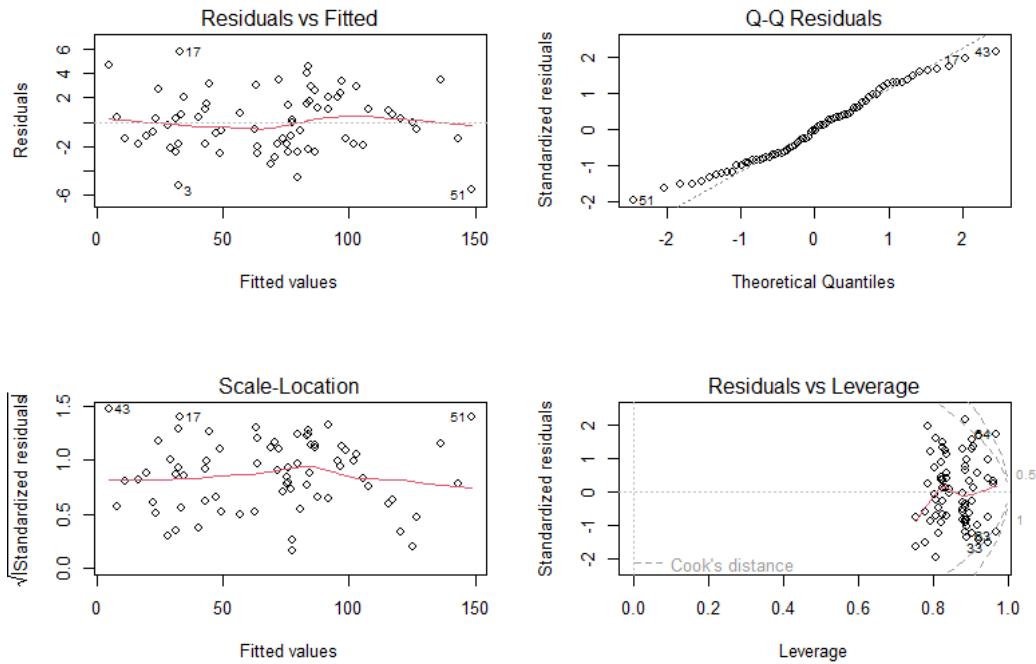
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.344 on 10 degrees of freedom
Multiple R-squared:  0.9952, Adjusted R-squared:  0.9657 
F-statistic: 33.73 on 61 and 10 DF,  p-value: 7.065e-07

```

**Figure 12.** linear model summary of VSGFS dataset

Based on Figure 12, The adjusted R-squared value of 0.9657 indicates that the model explains approximately 96.57% of the variation in the yield, after adjusting for the number of predictors and sample size. This suggests a strong relationship between the predictor variables and the yield. The F-statistic of 33.73 with a very low p-value (p-value < 0.001) suggests that the overall model is statistically significant. This implies that the predictor variables, as a whole, have a significant impact on predicting the yield. In conclusion, this linear regression model provides a good fit for predicting the yield based on the significant predictor variables, as indicated by the high adjusted R-squared value and significant F-statistic.



**Figure 13.** Linear model plot of VSGFS dataset

Through the observation of Figure 13, we can see that there are some presence of outliers as seen as in the residual plots that deviates from the normal distribution.

### Problem 5.a

We are going to compute a full 2 factorial model with a linear blocking effect and extract the model matrix X.

```
> full.factorial.model
Call:
lm.default(formula = y ~ ., data = BM93.e3.data)

Coefficients:
(Intercept) blk A B C D E F G H
20.52500 0.77500 -0.42500 -0.05000 3.18333 0.28333 -1.97500 -0.03333 0.31667 0.60000

> X
(Intercept) blk A B C D E F G H > print(dim(X))
[1] 20 10
1 1 -1 -1 -1 1 1 -1 1
2 1 -1 1 -1 -1 -1 1 1 1
3 1 -1 1 1 -1 -1 1 -1 1
4 1 -1 1 1 -1 -1 -1 1 1
5 1 -1 1 -1 1 1 -1 1 1
6 1 -1 1 -1 1 -1 -1 1 1
7 1 -1 1 1 1 -1 -1 1 1
8 1 -1 1 1 1 1 1 1 1
9 1 -1 1 1 1 -1 -1 1 1
10 1 -1 1 1 1 1 1 -1 1
11 1 -1 1 -1 1 1 -1 1 1
12 1 -1 -1 -1 1 -1 1 1 -1
13 1 -1 1 1 -1 -1 1 1 -1
14 1 -1 -1 1 -1 -1 1 1 -1
15 1 -1 1 -1 -1 1 1 -1 1
16 1 -1 -1 -1 -1 -1 1 -1 1
17 1 1 -1 1 1 1 -1 -1 1
18 1 1 -1 1 -1 -1 1 1 1
19 1 1 1 1 -1 1 1 -1 1
20 1 1 1 1 1 1 1 1 1

> identical_cols <- which(duplicated(xTX), arr.ind=TRUE)
> print(identical_cols)
dim1
> XTX
(Intercept) blk A B C D E F G H
(Intercept) 20 -12 0 4 0 0 0 0 0 4
blk -12 20 0 4 0 0 0 0 0 4
A 0 0 20 0 0 0 4 0 0 0
B 4 4 0 20 0 0 0 0 0 4
C 0 0 0 20 4 0 0 0 0 0
D 0 0 0 4 20 0 0 0 0 0
E 0 0 4 0 0 0 20 0 0 0
F 0 0 0 0 0 0 0 20 4 0
G 0 0 0 0 0 0 0 4 20 0
H 4 4 0 4 0 0 0 0 0 20
```

**Figure 14.** Key R Output of problem 5.a

Through the R output in Figure 14, we have acquire the model matrix X, the dimension of X that is [20 10]. It is also to be seen that there are no identical columns of  $X^T X$

### Problem 5.b

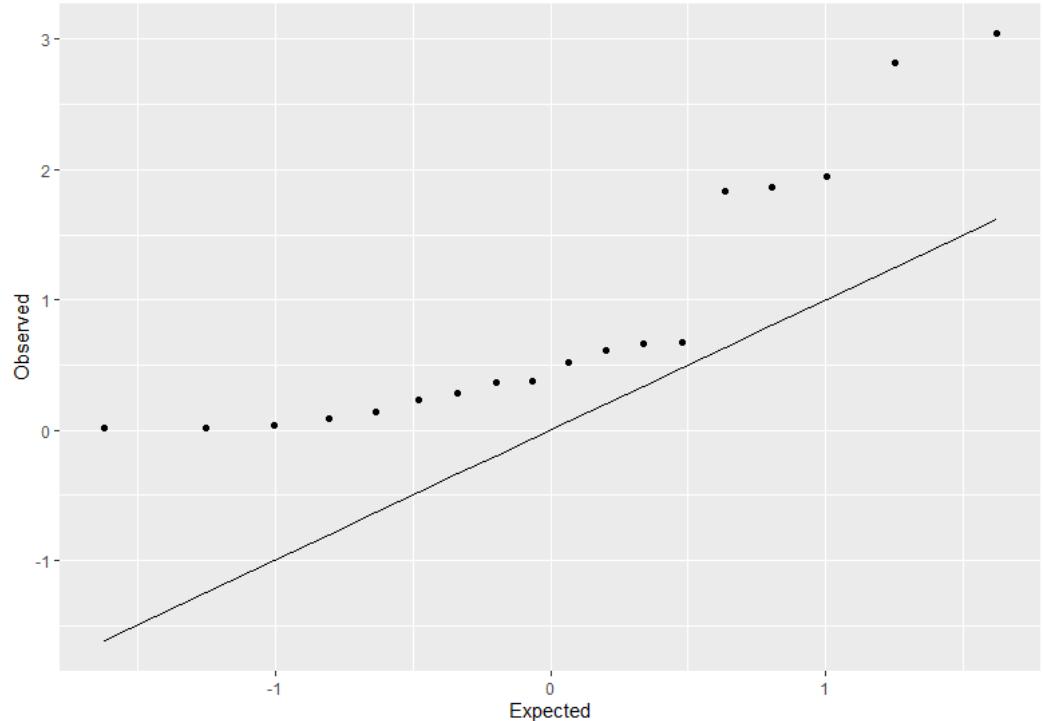
Now, we extract all non-diagonal pairs from the  $X^T X$  matrix with value 20. Note that they are equal (up to sign). Based on Figure 14, we can see that there are no non-diagonal pairs from  $X^T X$  with the value of 20. This is also done in the R where there are no output given of the column or row index of  $X^T X$  shown in Figure 15.

```
> mask <- rownames(XTX) != colnames(t(XTX))
> pairs <- which(mask & (XTX == 20 | XTX == -20), arr.ind=TRUE)
> print(pairs)
  row col
```

**Figure 15.** Key R Output of problem 5.b

### Problem 5.c

Now we compute a half normal plot with the effects and construct a model on the basis of this.



**Figure 16.** half normal plot

Based on Figure 16, it is to be seen that both main effects ('E', 'C', 'F', 'B', 'A', 'G', 'D') and interaction effects ('A:F', 'C:D', 'A:E', 'A:G', 'A:D', 'A:B', 'B:D') will be included into the reduced model. These points were chosen based on the large deviations from the expected line that should indicate significant effects.

```

> summary(reduced_model)

Call:
lm.default(formula = y ~ E + A:F + C:D + C + A:E + F + B + A:G +
D + A:D + A:B + A + G + B:D, data = BM93.e3.data)

Residuals:
    1      2      3      4      5      6      7      8 
1.48929 0.08571 1.36071 0.21429 -0.20357 1.20000 -0.33214 -0.51429 
    9     10     11     12     13     14     15     16 
-0.08571 -2.06786 -0.21429 -0.78214 -0.94286 -0.63214 -1.07143 -0.50357 
    17     18     19     20 
1.41429 0.25714 -0.25714 1.58571 

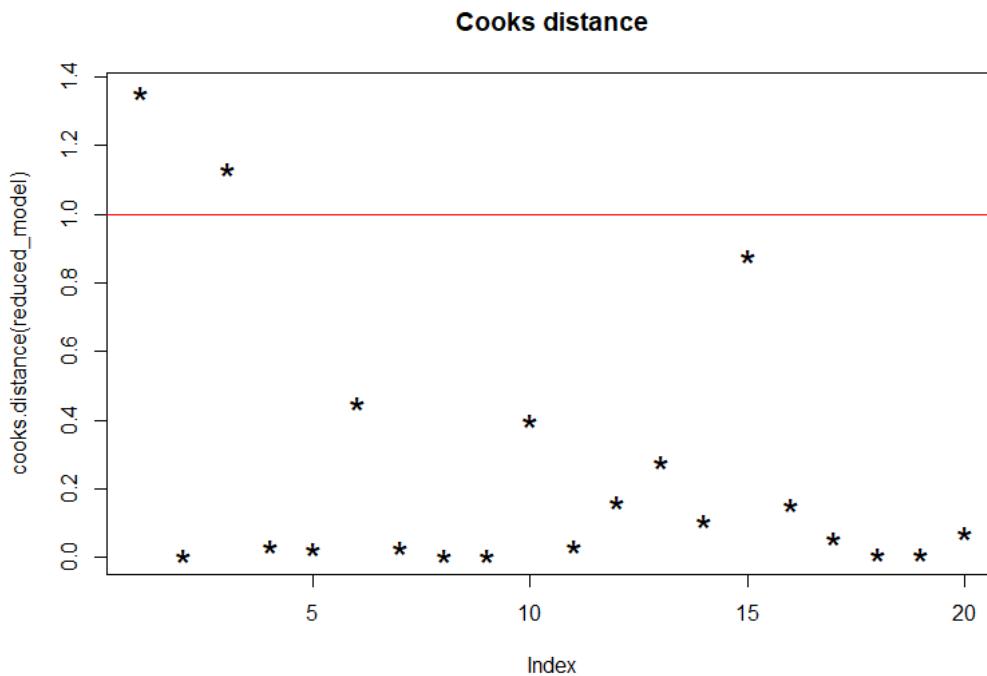
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 19.937500  0.462548 43.104 1.27e-07 ***
E           -1.921429  0.442281 -4.344  0.00740 **  
C            2.937500  0.462548  6.351  0.00143 **  
F           -0.007143  0.442281 -0.016  0.98774    
B            0.137500  0.462548  0.297  0.77822    
D            0.037500  0.462548  0.081  0.93853    
A           -0.371429  0.442281 -0.840  0.43932    
G            0.342857  0.442281  0.775  0.47327    
A:F          1.500000  0.955436  1.570  0.17722    
C:D          -1.462500  0.947942 -1.543  0.18352    
E:A          2.487500  0.462548  5.378  0.00300 **  
A:G          0.087500  0.462548  0.189  0.85740    
A:D          -0.157143  0.442281 -0.355  0.73686    
A:B          -0.321429  0.442281 -0.727  0.49996    
D:B          -0.112500  0.462548 -0.243  0.81750    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 1.911 on 5 degrees of freedom
Multiple R-squared:  0.9578,   Adjusted R-squared:  0.8395 
F-statistic: 8.098 on 14 and 5 DF,  p-value: 0.01523

```

**Figure 17.** Summary of the reduced model

Based on the summary on the reduced model on Figure 17 it can be concluded that The main factors E, C, and the interaction A:E were found to be statistically significant at the 1% level. This indicates these factors and interaction have a substantial impact on the response variable y. On another note, the Multiple R-squared value was 0.9578, suggesting that the model explains about 95.78% of the variability in the response variable, which shows a good fit of the model to the data.



**Figure 18.** Cook Distance of Reduced model

To indicate outliers in the reduced model, Cook's distance plot is used. Based on Figure 18, it can be seen that there are 2 outliers that is the first and third index.

### Problem 6.a

Constructing a full factorial design of  $2^4$  and computing its generalized words of lengths can be done using function of `DoE.base` library.

```
> summary(design_6a)
Call:
fac.design(nlevels = c(2, 2, 2, 2), factor.names = c("A", "B",
" C", "D"))

Experimental design of type full factorial
16 runs

Factor settings (scale ends):
 A B C D
1 1 1 1 1
2 2 2 2 2

The design itself:
 A B C D
1 2 2 2 2
2 2 2 1 1
3 2 1 2 2
4 2 1 2 1
5 2 2 2 1
6 2 1 1 2
7 1 2 2 2
8 2 2 1 2
9 1 2 1 1
10 1 2 2 1
11 1 1 2 2
12 2 1 1 1
13 1 1 2 1
14 1 1 1 2
15 1 2 1 2
16 1 1 1 1

class=design, type= full factorial
> print(gwl_6a)
0 1 2 3 4
1 0 0 0 0
```

**Figure 19.** Summary of full factorial design  $2^4$

Figure 19 shows the summary of the full factorial design  $2^4$  constructed with its generalized words of lengths defined as "gwl\_6a". It can be observed that the proportion of 0 length words is 1, which simply means that every run has at least one design point. While the proportion of 1,2,3,4-length are 0 which indicates that no pair of design points are "neighbors" (adjacent points that differ by only one factor level). This pattern suggests that the design constructed is likely an orthogonal array, which leads to precise and independent estimates of main effects and two-factor interactions.

### Problem 6.b

Constructing a full factorial design of  $2^4$  with 4 blocks and computing its generalized words of lengths should be done using function of `DOE.wrapper` library.

```
> summary(design_6b)
Call:
FrF2(nfactors = 4, nruns = 16, blocks = 4, alias.block.2fis = TRUE,
      randomize = FALSE)

Experimental design of type FrF2.blocked
16 runs
blocked design with 4 blocks of size 4

Factor settings (scale ends):
  A B C D
1 -1 -1 -1 -1
2 1 1 1 1

Design generating information:
$`Legend`
[1] A=A B=B C=C D=D
$`generators for design itself`
[1] full factorial
$`block generators`
ABC AD
7 9

no aliasing of main effects or 2fis among experimental factors

Aliased with block main effects:
[1] AD
```

The design itself:			
run.no	run.no.std.rp	Blocks	A B C D
1	1	2.1.1	1 -1 -1 -1 1
2	2	8.1.2	1 -1 1 1 1
3	3	11.1.3	1 1 -1 1 -1
4	4	13.1.4	1 1 1 -1 -1
run.no	run.no.std.rp	Blocks	A B C D
5	5	1.2.1	2 -1 -1 -1 -1
6	6	7.2.2	2 -1 1 1 -1
7	7	12.2.3	2 1 -1 1 1
8	8	14.2.4	2 1 1 -1 1
run.no	run.no.std.rp	Blocks	A B C D
9	9	4.3.1	3 -1 -1 1 1
10	10	6.3.2	3 -1 1 -1 1
11	11	9.3.3	3 1 -1 -1 -1
12	12	15.3.4	3 1 1 1 -1
run.no	run.no.std.rp	Blocks	A B C D
13	13	3.4.1	4 -1 -1 1 -1
14	14	5.4.2	4 -1 1 -1 -1
15	15	10.4.3	4 1 -1 -1 1
16	16	16.4.4	4 1 1 1 1

class=design, type= FrF2.blocked  
 NOTE: columns run.no and run.no.std.rp are annotation,  
 > print(gwl\_6b)

o 1 2 3 4	1 0 0 0 0
-----------	-----------

**Figure 20.** Summary of full factorial design  $2^4$  with 4 blocks

Figure 20 shows the summary of the full factorial design  $2^4$  with 4 blocks constructed with its generalized words of lengths defined as "gwl\_6b". It can be observed that the generalized words of lengths have the same results as problem 6.a, which also considered a good design for factorials, offering balance and allowing for clear interpretation of the results.

One thing to be noted the exsistance of the line "no aliasing of main effects or 2fis among experimental factors". This means that the factorial design is well chosen, such that the effects of the factors are independent of each other. This should be a desired property in experimental design.

### Problem 6.c

#### Three Choices of Generators for the $2^{7-2}_{IV}$ Design

Design A Generators: $F = ABC, G = BCD$ $I = ABCF = BCDG = ADFG$	Design B Generators: $F = ABC, G = ADE$ $I = ABCF = ADEG = BCDEFG$	Design C Generators: $F = ABCD, G = ABDE$ $I = ABCDF = ABDEG = CEFG$
Aliases (two-factor interactions) $AB = CF$ $AC = BF$ $AD = FG$ $AG = DF$ $BD = CG$ $BG = CD$ $AF = BC = DG$	Aliases (two-factor interactions) $AB = CF$ $AC = BF$ $AD = EG$ $AE = DG$ $AF = BC$ $AG = DE$	Aliases (two-factor interactions) $CE = FG$ $CF = EG$ $CG = EF$

**Figure 21.** Three choices of generators for the  $2^{7-2}_{IV}$  design

*Source: Design and Analysis of Experiments, Montgomery (2012)*

Based on Figure 21, we can compute the three  $2^{7-2}_{IV}$  design and compare the lengths of their word pattern which in turn we'll also verify the Montgomery's conclusions. We define the Generators for each design as an input for the `FrF2` function with 7 factors,  $2^{(7-2)}$  runs, and resolution == 4 for each design. Through `summary` function, we can conclude the aliases for each design and compare the lengths of their word pattern.

```

> summary(DesignA_6c)
Call:
FrF2(nfactors = 7, nruns = 2^(7 - 2), resolution = 4, generators = generatorsA_6c)

Experimental design of type FrF2.generators
32 runs

Factor settings (scale ends):
 A B C D E F G
1 -1 -1 -1 -1 -1 -1
2 1 1 1 1 1 1 1

Design generating information:
\$legend
[1] A=A B=B C=C D=D E=E F=F G=G

\$generators
[1] F=ABC G=BCD

Alias structure:
\$fi2
[1] AB=CF AC=BF AD=FG AF=BC=DF AG=DF BD=CG BG=CD

```

**Figure 22.**  $2_{IV}^{7-2}$  Design A summary

Figure 22 shows us the same aliases as Figure 21 as for the lengths of the word pattern, we can have  $I = ABCF = BCDG = ADFG$  by;

$$\begin{aligned}
F &= ABC, \\
FF &= ABCF, \\
I &= ABCF, \\
G &= BCD, \\
GG &= BCDG, \\
I &= BCDG,
\end{aligned}$$

Since we know,  $BC = AF$  from the aliases, then

$$\begin{aligned}
I &= (BC)DG, \\
I &= (AF)DG, \\
I &= ABCF = BCDG = ADFG
\end{aligned}$$

So  $I = ABCF = BCDG = ADFG$ . Where the length of three word is  $\{4,4,4\}$

```

> summary(DesignB_6c)
Call:
FrF2(nfactors = 7, nruns = 2^(7 - 2), resolution = 4, generators = generatorsB_6c)

Experimental design of type FrF2.generators
32 runs

Factor settings (scale ends):
 A B C D E F G
1 -1 -1 -1 -1 -1 -1
2 1 1 1 1 1 1 1

Design generating information:
\$legend
[1] A=A B=B C=C D=D E=E F=F G=G

\$generators
[1] F=ABC G=ADE

Alias structure:
\$fi2
[1] AB=CF AC=BF AD=EG AE=DG AF=BC AG=DE

```

**Figure 23.**  $2_{IV}^{7-2}$  Design B summary

Figure 23 shows us the same aliases as Figure 21 as for the lengths of the word pattern, we can have  $I$

$= ABCF = ADEG = BCDEF$  by

$$\begin{aligned} F &= ABC, \\ FF &= ABCF, \\ I &= ABCF, \\ G &= ADE, \\ GG &= ADEG, \\ I &= ADEG, \end{aligned}$$

Since we know,  $BC = AF$  from the aliases, then

$$\begin{aligned} I &= (BC)DG, \\ I &= (AF)DG, \\ I &= ABCF = BCDG = ADFG. \end{aligned}$$

Now, by analyzing the aliases  $AF=BC$  that includes factors B and C. The alias  $AG=DE$  that includes factors D and E. Considering the common factors B, C, D, and E from the aliases  $AF=BC$  and  $AG=DE$  along with factor G from the generator  $G=ADE$ , the factor combination  $I = BCDEF$  can be obtained. So  $I = ABCF = ADEG = BCDEF$ . Where the length of three word is  $\{4,4,6\}$

```
> summary(DesignC_6c)
Call:
FrF2(nfactors = 7, nruns = 2^(7 - 2), resolution = 4, generators = generatorsC_6c)

Experimental design of type FrF2.generators
32 runs

Factor settings (scale ends):
  A   B   C   D   E   F   G
1  -1  -1  -1  -1  -1  -1
2   1   1   1   1   1   1   1

Design generating information:
$Legend
[1] A=A B=B C=C D=D E=E F=F G=G

$generators
[1] F=ABCD G=ABDE

Alias structure:
$F12
[1] CE=FG CF=EG CG=EF
```

**Figure 24.**  $2^{7-2}_{IV}$  Design C summary

Figure 24 shows us the same aliases as Figure 21 as for the lengths of the word pattern, we can have  $I = ABCF = ADEG = BCDEF$  by

$$\begin{aligned} F &= ABCD, \\ FF &= ABCDF, \\ I &= ABCDF, \\ G &= ABDE, \\ GG &= ABDEG, \\ I &= ABDEG, \end{aligned}$$

Since we know,  $CF = EG$  from the aliases, then

$$\begin{aligned} F &= ABCD, \\ CF &= ABD(I), \\ I &= (ABD)EG, \\ I &= (CF)EG, \\ I &= ABCDF = ABDEG = CEFG. \end{aligned}$$

Now we have  $I = CEFG = ABDEG = ABCDF$ . Where the length of three word is  $\{4,4,5\}$ .

Montgomery states that the three word lengths in design A are  $\{4,4,4\}$ . For design B it is  $\{4,4,6\}$  and for design C it is  $\{4,4,5\}$ . Comparing all the numbers gives us the same result of the length that was computed with the one that Montgomery states.

### Problem 6.d

Constructing a  $2^{10-5}$  design (Design A) with the following generators:

```
generators = c("ABCD", "ABCE", "ABDE", "ACDE", "BCDE"),
```

and a  $2^{10-6}$  design (Design B) with the following generators:

```
generators = c("ABC", "BCD", "ACD", "ABD", "ABCD", "AB")
```

can be done using FrF2 function same as the previous problem. The Design results is given in Table.2 and Table.3. Now we call the summary function of the design results as shown in Figure 25 and Figure 26. Through the summary function we can conclude some information, that is:

```
> summary(DesignA)
Call:
FrF2(nfactors = 10, nruns = 2^(10 - 5), generators = generatorsA)

Experimental design of type FrF2.generators
32 runs

Factor settings (scale ends):
 A B C D E F G H J K
1 -1 -1 -1 -1 -1 -1 -1 -1 -1
2 1 1 1 1 1 1 1 1 1

Design generating information:
$legend
[1] A=A B=B C=C D=D E=E F=F G=G H=H J=J K=K

$generators
[1] F=ABCD G=ABCE H=ABDE J=ACDE K=BCDE

Alias structure:
$fr2
[1] AB=JK AC=HK AD=GK AE=FK AF=EK AG=DK AH=CK
[8] AJ=BK AK=BJ=CH=DG=EF BC=HJ BD=GJ BE=FJ BF=EJ BG=DJ
[15] BH=CJ CD=GH CE=FH CF=EH CG=DH DE=FG DF=EG
```

**Figure 25.** Summary of Design A

For Design A, we have:

1. A  $2^{10-5}$  design with 32 runs.
2. The alias structure shows how the main effects and two-factor interactions are confounded. For instance, AB=JK implies that the AB interaction is confounded with JK interaction. This is a consequence of using a fractional factorial design.

```
> summary(DesignB)
Call:
FrF2(nfactors = 10, nruns = 2^(10 - 6), generators = generatorsB)

Experimental design of type FrF2.generators
16 runs

Factor settings (scale ends):
 A B C D E F G H J K
1 -1 -1 -1 -1 -1 -1 -1 -1 -1
2 1 1 1 1 1 1 1 1 1

Design generating information:
$legend
[1] A=A B=B C=C D=D E=E F=F G=G H=H J=J K=K

$generators
[1] E=ABC F=BCD G=ACD H=ABD J=ABCD K=AB

Alias structure:
$fr2
[1] A=JK F=J B=AK=GJ C=EK=HJ D=AJ=HK E=CK=DJ F=AJ=GK G=BH=FK H=CJ=DK
[9] J=AF=BG=CH=DE K=AB=CE=DH=FG
$fr2
[1] AC=BE=DG=FH AD=BH=CG=EF AE=BC=DF=GH AG=BF=CD=EH=JK AH=BD=CF=EG
```

**Figure 26.** Summary of Design B

For Design B, we have:

1. A  $2^{10-6}$  design with 16 runs.

2. The alias structure shows similar confounding as in Design A. For instance,  $A=BK=FJ$  indicates that the effect of main effect A is confounded with BK and FJ interaction.

Choosing the design that is preferred to estimating main effects can be done by looking at the alias structures, we'll find the main effects are confounded with higher-order interactions in both designs.

In Design A, some main effects are aliased (i.e., confounded) with two-factor interactions. For instance,  $AG=DK$  implies that main effect A is confounded with the two-factor interaction DK. Design B in other hands, the main effects are confounded with two-factor and higher-order interactions. For instance,  $A=BK=FJ$  implies that the main effect A is confounded with the two-factor interactions BK and FJ.

From this perspective, Design A should be better choice because the main effects are only confounded with two-factor interactions, not with higher-order interactions. In Design B, the main effects are confounded with higher-order interactions as well, which makes it more difficult to accurately estimate the main effects. So in the end, Design A is likely to be chosen.

### Problem 6.e

Construting a 64 runs design with all  $(A+B+C)*(D+E+F+G+H)$  interactions estimable is done using FrF2 function same as the previous problem. The design result is given in Table. 4 in the Appendix section. Figure 27 shows the result of the 64 run design.

```
> summary(design_6e)
Call:
FrF2(64, nfactors = 8, factors = factors)

Experimental design of type FrF2
64 runs

Factor settings (scale ends):
  A B C D E F G H
1 -1 -1 -1 -1 -1 -1 -1
2  1  1  1  1  1  1  1

Design generating information:
$legend
[1] A=A B=B C=C D=D E=E F=F G=G H=H

$generators
[1] G=ABCD H=ABEF

Alias structure:
[[1]]
[1] no aliasing among main effects and 2fis
```

**Figure 27.** Summary of 64 run design

From Figure 27, we can see that the design is generated using two generators:  $G=ABCD$  and  $H=ABEF$ . This implies that the design is constructed in a way that allows for estimating all  $(A+B+C)*(D+E+F+G+H)$  interactions. It is also to be observed that there is no aliasing among the main effects and two-factor interactions (2fis). This means that each main effect and two-factor interaction can be estimated independently without any confounding.

The summary for the 64 run design confirms that the design satisfies the requirement. The design is specifically generated to include all the necessary interactions and avoids aliasing, ensuring that each interaction can be estimated uniquely.

## APPENDIX

**Table for Problem 6.c**  
**Table for Problem 6.d**

**Table 2.** Design A

	A	B	C	D	E	F	G	H	J	K		A	B	C	D	E	F	G	H	J	K
<b>1</b>	-1	1	-1	1	-1	1	-1	1	-1	1	<b>17</b>	-1	-1	1	1	1	1	1	1	-1	-1
<b>2</b>	-1	1	-1	-1	1	-1	1	1	-1	1	<b>18</b>	-1	1	-1	1	1	1	1	-1	1	-1
<b>3</b>	1	1	1	1	1	1	1	1	1	1	<b>19</b>	-1	-1	-1	-1	-1	1	1	1	1	1
<b>4</b>	-1	-1	-1	-1	1	1	-1	-1	-1	-1	<b>20</b>	-1	1	1	-1	-1	1	1	-1	-1	1
<b>5</b>	1	1	1	-1	-1	-1	-1	1	1	1	<b>21</b>	-1	-1	1	-1	1	-1	1	-1	1	1
<b>6</b>	1	1	1	-1	1	-1	1	-1	-1	-1	<b>22</b>	-1	1	1	1	1	-1	-1	-1	-1	1
<b>7</b>	1	-1	1	-1	-1	1	1	-1	1	-1	<b>23</b>	1	1	-1	-1	1	1	-1	-1	1	1
<b>8</b>	1	-1	-1	1	-1	1	-1	1	1	-1	<b>24</b>	-1	-1	-1	1	1	-1	-1	1	1	1
<b>9</b>	1	-1	-1	-1	1	-1	1	1	1	-1	<b>25</b>	1	-1	-1	1	1	1	1	-1	-1	1
<b>10</b>	-1	1	1	-1	1	1	-1	1	1	-1	<b>26</b>	-1	-1	-1	1	-1	-1	1	-1	-1	-1
<b>11</b>	-1	-1	1	-1	-1	-1	-1	1	-1	-1	<b>27</b>	-1	-1	1	1	-1	1	-1	-1	1	1
<b>12</b>	1	1	1	1	-1	1	-1	-1	-1	-1	<b>28</b>	1	-1	-1	-1	-1	-1	-1	-1	-1	1
<b>13</b>	1	1	-1	-1	-1	1	1	1	-1	-1	<b>29</b>	-1	1	1	1	-1	-1	1	1	1	-1
<b>14</b>	1	-1	1	-1	1	1	-1	1	-1	1	<b>30</b>	-1	1	-1	-1	-1	-1	-1	-1	1	-1
<b>15</b>	1	-1	1	1	-1	-1	1	1	-1	1	<b>31</b>	1	-1	1	1	1	-1	-1	-1	1	-1
<b>16</b>	1	1	-1	1	1	-1	-1	1	-1	-1	<b>32</b>	1	1	-1	1	-1	-1	1	-1	1	1

**Table 3.** Design B

	A	B	C	D	E	F	G	H	J	K
<b>1</b>	-1	1	1	-1	-1	-1	1	1	1	-1
<b>2</b>	-1	1	1	1	-1	1	-1	-1	-1	-1
<b>3</b>	-1	1	-1	1	1	-1	1	-1	1	-1
<b>4</b>	1	-1	-1	1	1	1	-1	-1	1	-1
<b>5</b>	-1	-1	-1	-1	-1	-1	-1	-1	1	1
<b>6</b>	-1	1	-1	-1	1	1	-1	1	-1	-1
<b>7</b>	-1	-1	1	-1	1	1	1	-1	-1	1
<b>8</b>	1	1	1	1	1	1	1	1	1	1
<b>9</b>	1	1	-1	-1	-1	1	1	-1	1	1
<b>10</b>	-1	-1	-1	1	-1	1	1	1	-1	1
<b>11</b>	1	-1	-1	-1	1	-1	1	1	-1	-1
<b>12</b>	-1	-1	1	1	1	-1	-1	1	1	1
<b>13</b>	1	1	1	-1	1	-1	-1	-1	-1	1
<b>14</b>	1	1	-1	1	-1	-1	-1	1	-1	1
<b>15</b>	1	-1	1	1	-1	-1	1	-1	-1	-1
<b>16</b>	1	-1	1	-1	-1	-1	1	-1	1	-1

**Table for Problem 6.e****Table 4.** 64 runs design

	A	B	C	D	E	F	G	H		A	B	C	D	E	F	G	H
<b>1</b>	-1	1	-1	-1	-1	-1	-1	-1	<b>33</b>	-1	-1	1	-1	1	1	-1	1
<b>2</b>	1	-1	1	1	-1	1	-1	1	<b>34</b>	-1	1	1	-1	-1	1	1	1
<b>3</b>	-1	1	1	-1	-1	-1	1	-1	<b>35</b>	1	-1	-1	1	1	-1	1	1
<b>4</b>	1	-1	1	-1	1	-1	1	1	<b>36</b>	1	1	-1	1	-1	1	-1	-1
<b>5</b>	-1	1	-1	-1	1	1	-1	-1	<b>37</b>	1	1	-1	-1	1	1	1	1
<b>6</b>	1	-1	-1	1	-1	-1	1	-1	<b>38</b>	-1	1	-1	1	1	1	1	-1
<b>7</b>	-1	1	1	-1	1	-1	1	1	<b>39</b>	1	1	-1	1	-1	-1	-1	1
<b>8</b>	-1	-1	-1	1	1	-1	-1	-1	<b>40</b>	1	-1	-1	-1	-1	-1	-1	-1
<b>9</b>	-1	1	1	1	-1	1	-1	1	<b>41</b>	-1	1	1	1	-1	-1	-1	-1
<b>10</b>	-1	-1	-1	-1	-1	-1	1	1	<b>42</b>	-1	1	1	-1	1	1	1	-1
<b>11</b>	1	-1	-1	-1	1	-1	-1	1	<b>43</b>	-1	1	1	1	-1	-1	1	1
<b>12</b>	-1	-1	-1	-1	1	-1	1	-1	<b>44</b>	1	-1	1	-1	-1	-1	1	-1
<b>13</b>	1	-1	1	1	1	1	-1	-1	<b>45</b>	1	1	1	1	-1	1	-1	-1
<b>14</b>	-1	-1	-1	-1	-1	1	1	-1	<b>46</b>	-1	-1	1	-1	1	-1	-1	-1
<b>15</b>	1	1	1	-1	-1	1	-1	-1	<b>47</b>	1	-1	-1	-1	1	1	-1	-1
<b>16</b>	1	1	1	-1	-1	-1	-1	1	<b>48</b>	-1	1	-1	1	1	-1	1	1
<b>17</b>	-1	-1	-1	1	-1	1	-1	-1	<b>49</b>	1	1	-1	1	1	-1	-1	-1
<b>18</b>	-1	-1	-1	1	1	1	-1	1	<b>50</b>	-1	-1	1	1	1	1	1	1
<b>19</b>	1	1	-1	-1	1	-1	1	-1	<b>51</b>	-1	1	1	1	1	-1	-1	-1
<b>20</b>	1	-1	1	-1	1	1	1	-1	<b>52</b>	1	-1	-1	-1	-1	1	-1	1
<b>21</b>	-1	-1	-1	-1	1	1	1	1	<b>53</b>	-1	-1	-1	1	-1	-1	-1	1
<b>22</b>	1	1	1	1	1	1	1	1	<b>54</b>	1	-1	-1	1	-1	1	1	1
<b>23</b>	-1	-1	1	1	-1	1	1	-1	<b>55</b>	1	-1	-1	1	1	1	1	-1
<b>24</b>	1	1	1	1	-1	1	1	-1	<b>56</b>	1	1	-1	-1	-1	1	1	-1
<b>25</b>	1	-1	1	1	1	-1	-1	1	<b>57</b>	1	-1	1	1	-1	-1	-1	-1
<b>26</b>	1	1	1	1	-1	-1	1	1	<b>58</b>	1	1	1	-1	1	1	-1	1
<b>27</b>	-1	-1	1	1	1	-1	1	-1	<b>59</b>	1	-1	1	-1	-1	1	1	1
<b>28</b>	1	1	-1	-1	-1	-1	1	1	<b>60</b>	1	1	1	-1	1	-1	-1	-1
<b>29</b>	-1	-1	1	-1	-1	1	-1	-1	<b>61</b>	1	1	-1	1	1	-1	1	1
<b>30</b>	-1	1	-1	1	-1	-1	1	-1	<b>62</b>	-1	1	-1	-1	1	-1	-1	1
<b>31</b>	-1	-1	1	1	-1	-1	1	1	<b>63</b>	-1	1	-1	-1	-1	1	-1	1
<b>32</b>	-1	1	-1	1	-1	1	1	1	<b>64</b>	-1	-1	1	-1	-1	-1	-1	1

## CODE LISTINGS

### Problem 1

```
rm()
library(xtable)
library(BHH2)
library(dplyr)
library(gplots)

#1a
data(corrosion.data)
mn_sd_table <- corrosion.data %>%
  group_by(heats, coating) %>%
  summarise(
    mean_resistance = mean(resistance, na.rm = TRUE),
    sd_resistance = sd(resistance, na.rm = TRUE)
  )
mn_sd_latex <- xtable(mn_sd_table, digits = 2, caption = "Results_for_1(a)")

#1b
corrosion.data$heats <- as.factor(corrosion.data$heats)
corrosion.data$coating <- as.factor(corrosion.data$coating)
fit <- aov(resistance ~ heats*coating, data = corrosion.data)
summary(fit)

#1c
fit <- aov(resistance ~ heats*coating, data = corrosion.data)
par(mfrow = c(2, 2))
  plot(fit, 1) # res vs fitted val
  plot(fit, 2) # normal Q-Q
  plot(fit, 3) # Scale-Location (sqrt|standardized residuals| vs fitted val)
  plot(fit, 5) # res vs leverage
par(mfrow = c(1, 1))

#1d
tukey <- TukeyHSD(fit, "heats")
print(tukey)

#1e
means <- tapply(corrosion.data$resistance, corrosion.data$heats, mean)
sds <- tapply(corrosion.data$resistance, corrosion.data$heats, sd)
n <- tapply(corrosion.data$resistance, corrosion.data$heats, length)
ci <- qt(0.975, df = n - 1) * sds / sqrt(n)
data_summary <- data.frame(heats = names(means), mean = means, ci = ci)
print(data_summary)
#1e can use this too, this is a direct method to plot means
plotmeans(resistance ~ heats, data = corrosion.data,
          xlab = "Heats", ylab = "Mean_resistance",
          main = "Mean_resistance_with_95%_CI_by_Heats")
```

### Problem 2

```
rm()
#2a
library(nlme)
dat <- as.data.frame(Oxide)
```

```

model_2a <- lm(Thickness ~ Source + Site, data = dat)
print(anova(model_2a))

#2b
library(lme4)
model_2b <- lme(Thickness ~ 1, random = ~1|Lot/Wafer, data = dat)
conf_int <- intervals(model_2b)
print(conf_int)
print(summary(model_2b))

#2c
library(lattice)
dat$Predicted <- predict(model_2b)

xyplot(Thickness ~ as.numeric(as.character(dat$Wafer)) | Lot, data = dat, type
       panel = function(x, y, ...) {
         panel.xyplot(x, y, ...)
         panel.abline(lm(y ~ x), col = "red")
       },
       xlab = "Wafer", ylab = "Thickness")

```

### Problem 3

```

rm()
set.seed(123)
#3a
library(dplyr)
df <- expand_grid(A = c(-1, 0, 1), B = c(-1, 0, 1))
df <- df %>%
  mutate(Yield = 1 + 2*A + 3*B - A^2 - B^2 - 2*A*B)
# I assume b0 = 1, b1 = 2, b2 = 3, b3 = b4 = -1, interaction coefficient = -2
print(df)

model <- lm(Yield ~ A + B + I(A^2) + I(B^2) + A:B, data = df)
summary(model)

#3b
library(lattice)
library(akima)
A_vals <- seq(-1, 1, length.out = 100)
B_vals <- seq(-1, 1, length.out = 100)
grid <- expand_grid(A = A_vals, B = B_vals)
grid$Yield <- predict(model, newdata = grid)

# smoother plot interp
interp <- akima::interp(x = grid$A, y = grid$B, z = grid$Yield)

# contour plot
contour_plot <- filled.contour(x = interp$x, y = interp$y, z = interp$z,
                                 xlab = "A", ylab = "B",
                                 plot.title = title(main = "Contour_Plot_of_Yield"))

# perspective plot
persp_plot <- persp(x = interp$x, y = interp$y, z = interp$z, theta = 45,
                      phi = 45, xlab = "A", ylab = "B", zlab = "Yield",
                      main = "Perspective_Plot_of_Yield")

```

```

# Combine the two plots in a single panel.
# Still wont work to show side by side
par(mfrow = c(1, 2))
print(persp.plot)
print(contour.plot)
par(mfrow = c(1, 1))

```

### Problem 4

```

rm()
#4b and c
library(DoE.base)

corrPlot(VSGFS)
summary(lm(VSGFS, response = "Yield"))
plot(VSGFS)
anova(lm(VSGFS, response = "Yield"))
library("car")
model_4b <- lm(VSGFS, response = "Yield")
print(model_4b)
par(mfrow = c(2, 2))
plot(model_4b, 1) # res vs fitted val
plot(model_4b, 2) # normal Q-Q
plot(model_4b, 3) # Scale-Location (sqrt|standardized residuals| vs fitted val)
plot(model_4b, 5) # res vs leverage
par(mfrow = c(1, 1))
summary(model_4b)

```

### Problem 5

```

rm()
#5a
library(BsMD)
data(BM93.e3.data, package="BsMD")

library(BsMD)

full.factorial.model <- lm(y ~ ., data = BM93.e3.data)
X <- model.matrix(full.factorial.model)
print(dim(X))
XTX <- t(X) %*% X

identical_cols <- which(duplicated(XTX), arr.ind=TRUE)
print(identical_cols)

#5b
mask <- rownames(XTX) != colnames(t(XTX))
pairs <- which(mask & (XTX == 20 | XTX == -20), arr.ind=TRUE)
print(pairs)

#5c
model <- lm(y ~ (A+B+C+D+E+F+G+H)^2, data = BM93.e3.data)
effects <- coef(model)[-1] # omit intercept

#increasing order -> absolute effect

```

```

ordered_effects <- sort(abs(effects))

#generate expected order statistics from a half-normal dist
n <- length(ordered_effects)
expected <- qnorm((1:n) / (n + 1))
#half normal plot with the effects
library(ggplot2)
ggplot() +
  geom_point(aes(expected, ordered_effects)) +
  geom_line(aes(expected, expected)) +
  labs(x = "Expected", y = "Observed")

# reduced model based on half normal plot
reduced_model <- lm(y ~ E + A:F + C:D + C + A:E + F + B + A:G + D + A:D + A:B
  + A + G + B:D, data = BM93.e3.data)
summary(reduced_model)

plot(reduced_model$fitted.values, resid(reduced_model))

library(car)
cutoff <- 4 / ((nrow(BM93.e3.data) - length(coef(reduced_model))) - 1)
plot(cooks.distance(reduced_model), pch = "*", cex = 2, main = "Cooks-distance")
abline(h = cutoff, col = "red")

```

## Problem 6

```

rm()
#6a
library(DoE.base)
design_6a <- fac.design(nlevels=c(2,2,2,2), factor.names=c("A","B","C","D"))
gwl_6a <- GWLP(design_6a)
summary(design_6a)
print(gwl_6a)

#6b
library(DoE.wrapper)
design_6b <- FrF2(nfactors=4, nruns=16, blocks=4, alias.block.2fis=TRUE,
  randomize=FALSE)
gwl_6b <- GWLP(design_6b)
summary(design_6b)
print(gwl_6b)

#6c
library(FrF2)
generatorsA_6c <- c("ABC", "BCD")
generatorsB_6c <- c("ABC", "ADE")
generatorsC_6c <- c("ABCD", "ABDE")
DesignA_6c <- FrF2(nfactors = 7, nruns = 2^(7-2), resolution = 4,
  generators = generatorsA_6c)
DesignB_6c <- FrF2(nfactors = 7, nruns = 2^(7-2), resolution = 4,
  generators = generatorsB_6c)
DesignC_6c <- FrF2(nfactors = 7, nruns = 2^(7-2), resolution = 4,
  generators = generatorsC_6c)

print(DesignA_6c)
print(DesignB_6c)

```

```

print(DesignC_6c)

summary(DesignA_6c)
summary(DesignB_6c)
summary(DesignC_6c)

#6d
library(FrF2)
generatorsA <- c("ABCD", "ABCE", "ABDE", "ACDE", "BCDE")
generatorsB <- c("ABC", "BCD", "ACD", "ABD", "ABCD", "AB")

DesignA <- FrF2(nfactors = 10, nruns = 2^(10-5), generators = generatorsA)
DesignB <- FrF2(nfactors = 10, nruns = 2^(10-6), generators = generatorsB)

print(DesignA)
print(DesignB)

summary(DesignA)
summary(DesignB)

#6e
library(FrF2)
factors <- c("A", "B", "C", "D", "E", "F", "G", "H")

design_6e <- FrF2(64, nfactors = 8, factors = factors)
print(design_6e)
summary(design_6e)

```