STATS 101B Group Cereal Killer Project Report

By Changhao Chen, Chenkai Li, Benson Zu

1) Introduction.

To save the labor force, currently, garden irrigation is performed by garden sprinklers. When using a garden sprinkler, the quality characteristics that define a good sprinkler are low consumption of water and a wide spray range. The first goal of this study is to identify the relevant factors that drive the water consumption and spray range. The second goal is to find the factors' settings that minimize the water consumption and maximize the spray range.

2) Methodology & Procedure

2.1) Experimental Background

[Table 1. Variables used in the Experiment]

Predictors	Minimum	Maximum	Unit
alpha	0	90	°
beta	0	90	°
Aq	2.00E-06	4.00E-06	m2
d	0.1	0.2	m
mt	0.01	0.02	Nm
mf	0.01	0.02	Nm
pin	1	2	Bar
dzul	5	10	mm
Responsor	**Minimum**	**Maximum**	**Unit**
Consumption	0	Unlimited	1/min
Range	0	Unlimited	m

2.1.1) Experimental Design Choice

We mainly study main effects due to budget limits, we only study 2 levels of the 8 predictors, which are the upper and lower limits of their settings (Table 1.).

2.2.2) Experimental data

Since we are unable to build sprinklers ourselves and measure their water consumption (expressed in l/min) and range (expressed in m), we will hire a research agency that specializes in performing experiments by using the website http://twilights.be/sprinkler/.

The website returned the data that consists of our experimental design, as well as the measured water consumption and spray range for each test combination.

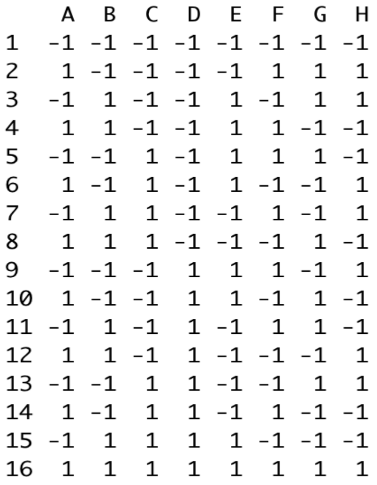
2.2) Experimental Design

Question 1. Propose a cost-efficient experimental design. Motivate your decision in statistical and practical terms.

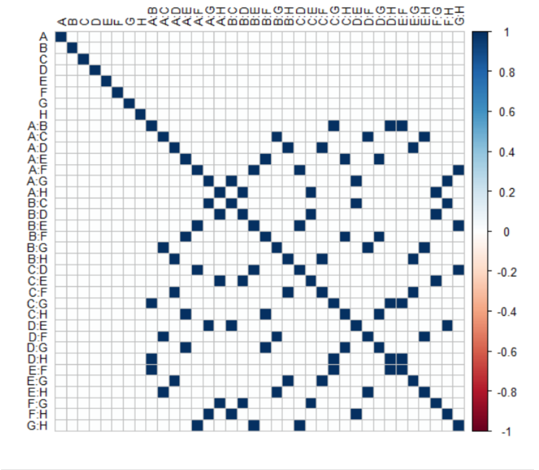
Since the nozzle will be completely closed with alpha = 90 and when beta = 90, the nozzle will be completely touched with the sprinkler, we set the experimental parameter of alpha and beta to [0,80]. Due to the limited budgets of 20 runs, we design to study the main effects only. And we aim to create a design that its main effects have no alias with any single or multiple interaction effects.

As a result, we propose a 2^{8-4} Fractional Factorial Designs and save 4 runs for the confirmation experiments. The 1/16 fraction of 8 factors in 16 runs, and each main effect are not only aliases with any other main effects or two-factor interaction effects. Assuming the effect hierarchy that interactions between more than 3 factors are zero, we will have the satisfied design showing in Table 2(left), which -1 and 1 represent the two levels choice for each parameter.

[Table 2. Our Experimental Design (left) and its Correlation Plots up to two factors interactions(right)]



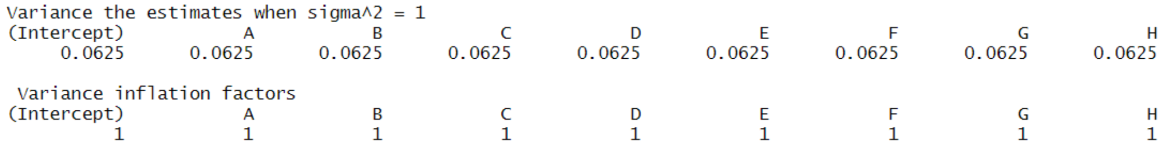
```
    A  B  C  D  E  F  G  H
1  -1 -1 -1 -1 -1 -1 -1 -1
2   1 -1 -1 -1 -1  1  1  1
3  -1  1 -1 -1  1 -1  1  1
4   1  1 -1 -1  1  1 -1 -1
5  -1 -1  1 -1  1  1  1 -1
6   1 -1  1 -1  1 -1 -1  1
7  -1  1  1 -1 -1  1 -1  1
8   1  1  1 -1 -1 -1  1 -1
9  -1 -1 -1  1  1  1 -1  1
10  1 -1 -1  1  1 -1  1 -1
11 -1  1 -1  1 -1  1  1 -1
12  1  1 -1  1 -1 -1 -1  1
13 -1 -1  1  1 -1 -1  1  1
14  1 -1  1  1 -1  1 -1 -1
15 -1  1  1  1  1 -1 -1 -1
16  1  1  1  1  1  1  1  1
```



Question 2. What is the performance of your design for studying the main effects of the factors only? Can your design estimate all two-factor interactions? Why or why not?

Based on our inspection of the design performance, its main effects are not aliased with any other main effect or two-factor interaction effects (Table 2.right). Moreover, its VIF values are all 1, which means that the predictors are not correlated with other variables (Table 3). But our design can't estimate all two-factor interactions since most of them are perfectly aliased with other two-factor interactions. Therefore, the estimation will not be accurate since the estimation results for some two-factor interactions are actually the aggregation of one or several interaction effects.

[Table 3 Check VIF for the main effect only]

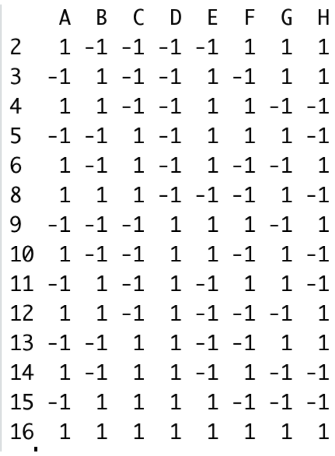


```
Variance the estimates when sigma^2 = 1
(Intercept)           A           B           C           D           E           F           G           H 
     0.0625      0.0625      0.0625      0.0625      0.0625      0.0625      0.0625      0.0625      0.0625 

 Variance inflation factors
(Intercept)           A           B           C           D           E           F           G           H 
          1           1           1           1           1           1           1           1           1 
```

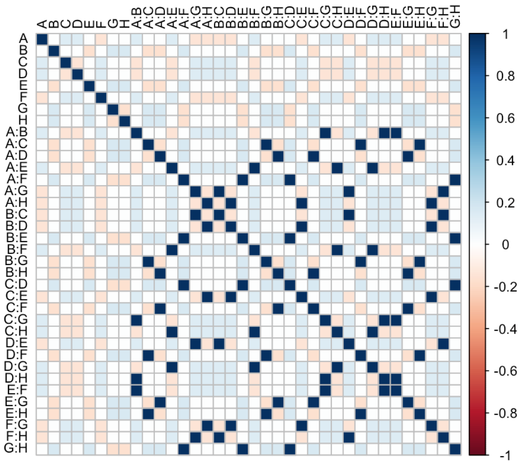
Question 3. The production engineers are concerned about having some failed tests in the experiment, given by sprinklers that cannot spray water. If you remove two randomly chosen test combinations, what is the performance of the resulting design?

If two chosen tests are randomly removed, the performance of the results design will be harmed. The main effects are partially aliased with both other main effects and two-factor interactions, which will potentially harm our modeling.

[Table 4. The Reduced Experimental Design with 14 cases (left) and its Correlation Plots up to two factors interactions(right)]

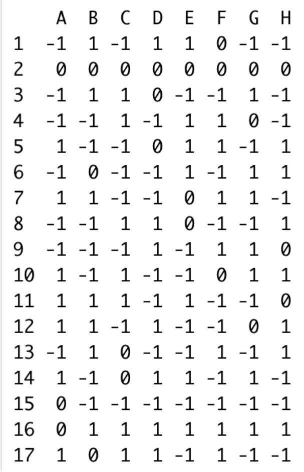


```
    A  B  C  D  E  F  G  H
2   1 -1 -1 -1 -1  1  1  1
3  -1  1 -1 -1  1 -1  1  1
4   1  1 -1 -1  1  1 -1 -1
5  -1 -1  1 -1  1  1  1 -1
6   1 -1  1 -1  1 -1 -1  1
8   1  1  1 -1 -1 -1  1 -1
9  -1 -1 -1  1  1  1 -1  1
10  1 -1 -1  1  1 -1  1 -1
11 -1  1 -1  1 -1  1  1 -1
12  1  1 -1  1 -1 -1 -1  1
13 -1 -1  1  1 -1 -1  1  1
14  1 -1  1  1 -1  1 -1 -1
15 -1  1  1  1  1 -1 -1 -1
16  1  1  1  1  1  1  1  1
```

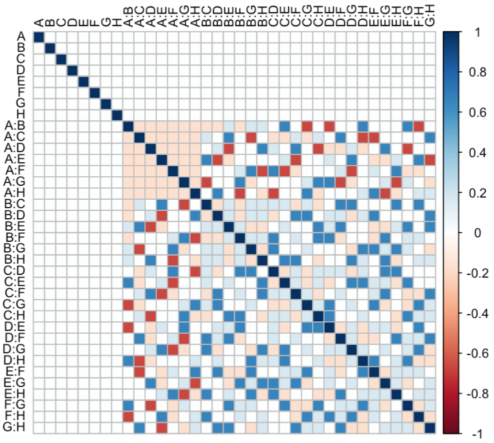


Question 4. The production engineers took an introductory course in experimental design. Using a commercial software, they came up with the experimental plan shown in Table 3. How does your full design compare with this one?

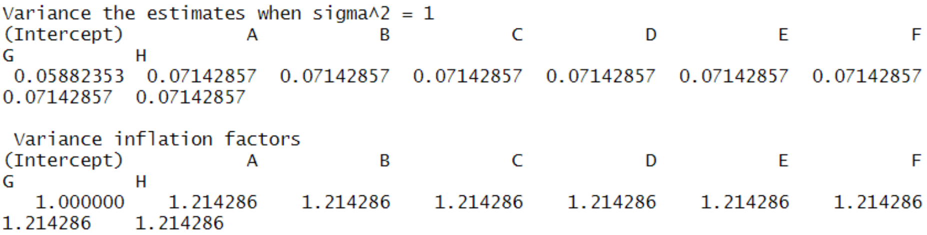
[Table 5: Alternative Experimental Design and its Correlation Plots]



```
    A  B  C  D  E  F  G  H
1  -1  1 -1  1  1  0 -1 -1
2   0  0  0  0  0  0  0  0
3  -1  1  1  0 -1 -1  1 -1
4  -1 -1  1 -1  1  1  0 -1
5   1 -1 -1  0  1  1 -1  1
6  -1  0 -1 -1  1 -1  1  1
7   1  1 -1 -1  0  1  1 -1
8  -1 -1  1  1  0 -1 -1  1
9  -1 -1 -1  1 -1  1  1  0
10  1 -1  1 -1 -1  0  1  1
11  1  1  1 -1  1 -1 -1  0
12  1  1 -1  1 -1 -1  0  1
13 -1  1  0 -1 -1  1 -1  1
14  1 -1  0  1  1 -1  1 -1
15  0 -1 -1 -1 -1 -1 -1 -1
16  0  1  1  1  1  1  1  1
17  1  0  1  1 -1  1 -1 -1
```



[Table 4: Alternative Experimental Design's VIF Table]



```
Variance the estimates when sigma^2 = 1
(Intercept)           A           B           C           D           E           F 
G           H 
 0.05882353  0.07142857  0.07142857  0.07142857  0.07142857  0.07142857  0.07142857 
0.07142857  0.07142857 

 Variance inflation factors
(Intercept)           A           B           C           D           E           F 
G           H 
   1.000000    1.214286    1.214286    1.214286    1.214286    1.214286    1.214286 
1.214286    1.214286 
```

The original experimental design is converted into a 3 level factorial design with -1, 0, 1 to represent the values in each column from smallest to the largest. Compared the design with ours, we found that:

Similarities:

1. The model also produces an estimation of the main effect with no allies of them. In other words, its main effect is also not aliased with any other effects.
2. Use 17 runs, which is also under 20 runs budgets.

Differences:

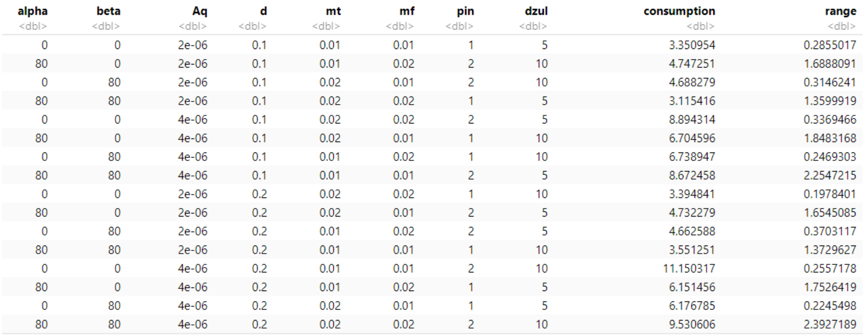
1. The VIF of the alternatives experimental design is 1.214 for all factors, which indicates that the factors are kind of correlated with each other, so their estimation effect will be inflated. Compared with our model of VIF = 1, our model is better.
2. The experimental design allows us to study 3 levels of each factor, which is better than our 2 level design model considering the width of choices.

2.3) Analysis of the Results

Question 5. Collect data using your recommended design in Question 1. Conduct a detailed data analysis.

We collect the data and collect the results through experimentation (Table 6)

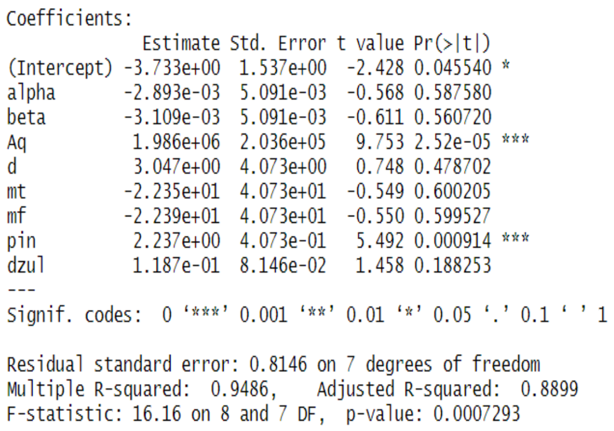
[Table 6: Experimental data]



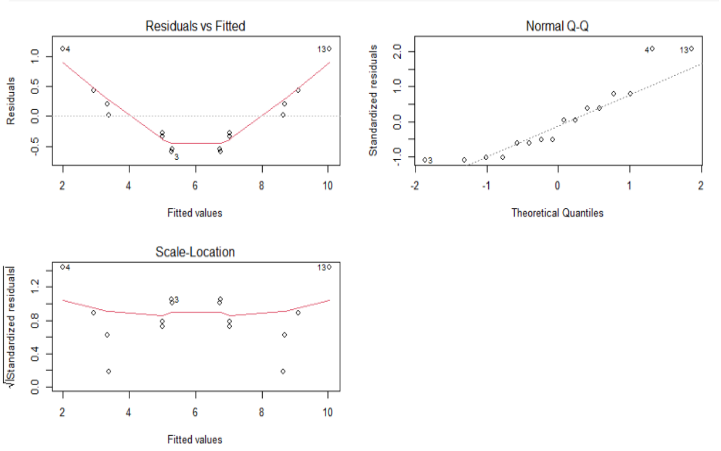
alpha <dbl>	beta <dbl>	Aq <dbl>	d <dbl>	mt <dbl>	mf <dbl>	pin <dbl>	dzul <dbl>	consumption <dbl>	range <dbl>
0	0	2e-06	0.1	0.01	0.01	1	5	3.350954	0.2855017
80	0	2e-06	0.1	0.01	0.02	2	10	4.747251	1.6888091
0	80	2e-06	0.1	0.02	0.01	2	10	4.688279	0.3146241
80	80	2e-06	0.1	0.02	0.02	1	5	3.115416	1.3599919
0	0	4e-06	0.1	0.02	0.02	2	5	8.894314	0.3369466
80	0	4e-06	0.1	0.02	0.01	1	10	6.704596	1.8483168
0	80	4e-06	0.1	0.01	0.02	1	10	6.738947	0.2469303
80	80	4e-06	0.1	0.01	0.01	2	5	8.672458	2.2547215
0	0	2e-06	0.2	0.02	0.02	1	10	3.394841	0.1978401
80	0	2e-06	0.2	0.02	0.01	2	5	4.732279	1.6545085
0	80	2e-06	0.2	0.01	0.02	2	5	4.662588	0.3703117
80	80	2e-06	0.2	0.01	0.01	1	10	3.551251	1.3729627
0	0	4e-06	0.2	0.01	0.01	2	10	11.150317	0.2557178
80	0	4e-06	0.2	0.01	0.02	1	5	6.151456	1.7526419
0	80	4e-06	0.2	0.02	0.01	1	5	6.176785	0.2245498
80	80	4e-06	0.2	0.02	0.02	2	10	9.530606	2.3927189

First, we build the raw models on consumption and range.

[Table 6: Raw Consumption model summary(left) and its residual Analysis(right)]



```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -3.733e+00  1.537e+00  -2.428 0.045540 *  
alpha       -2.893e-03  5.091e-03  -0.568 0.587580    
beta        -3.109e-03  5.091e-03  -0.611 0.560720    
Aq           1.986e+06  2.036e+05   9.753 2.52e-05 ***
d            3.047e+00  4.073e+00   0.748 0.478702    
mt          -2.235e+01  4.073e+01  -0.549 0.600205    
mf          -2.239e+01  4.073e+01  -0.550 0.599527    
pin          2.237e+00  4.073e-01   5.492 0.000914 ***
dzul         1.187e-01  8.146e-02   1.458 0.188253    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8146 on 7 degrees of freedom
Multiple R-squared:  0.9486,	Adjusted R-squared:  0.8899 
F-statistic: 16.16 on 8 and 7 DF,  p-value: 0.0007293
```



Raw Consumption model:

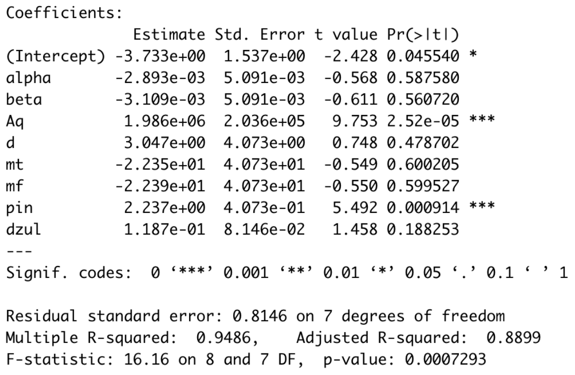
The Aq and pin significantly affect the consumption
R-squared is 0.9486, p-value is 0.0007293

Residuals:

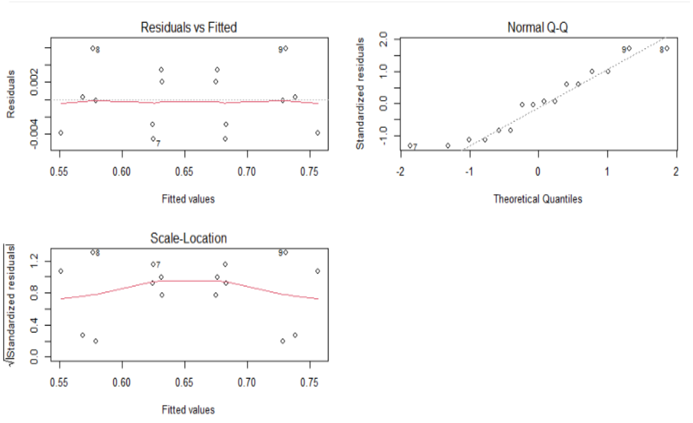
According to the residuals vs fitted plot, there is a non-linear trend and the constant-variance assumption is not satisfied.
According to the normal Q-Q plot, points are deviating at both the beginning and the end, which shows the normality assumption is not satisfied.
The scale location plot shows a horizontal line and the spread of points around the red line vary with the fitted value, which roughly satisfies the null hypothesis of homoscedasticity with a p-value close to 0
In general, it is not a satisfying model, so we do the transformation to refine the model.

Consumption Transformation

[Table 7: Transformed Consumption model summary(left) and its residual Analysis(right)]



```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -3.733e+00  1.537e+00  -2.428 0.045540 *  
alpha       -2.893e-03  5.091e-03  -0.568 0.587580    
beta        -3.109e-03  5.091e-03  -0.611 0.560720    
Aq           1.986e+06  2.036e+05   9.753 2.52e-05 ***
d            3.047e+00  4.073e+00   0.748 0.478702    
mt          -2.235e+01  4.073e+01  -0.549 0.600205    
mf          -2.239e+01  4.073e+01  -0.550 0.599527    
pin          2.237e+00  4.073e-01   5.492 0.000914 ***
dzul         1.187e-01  8.146e-02   1.458 0.188253    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8146 on 7 degrees of freedom
Multiple R-squared:  0.9486,	Adjusted R-squared:  0.8899 
F-statistic: 16.16 on 8 and 7 DF,  p-value: 0.0007293
```



Through the power transformation test, the consumption variable is recommended to take a power of (-0.25).

Transformation Consumption model:

The Aq and pin significantly affect the consumption value, the r-squared is 0.9486, the p-value is 0.0007293

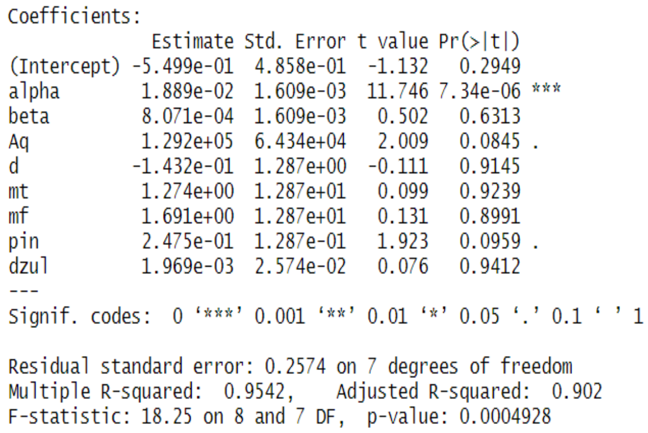
Residuals:

After the transformation, according to the residual vs fitted model, there is linearity that points are distributed evenly around the 0 line, so the constant variance and randomization assumption is satisfied.

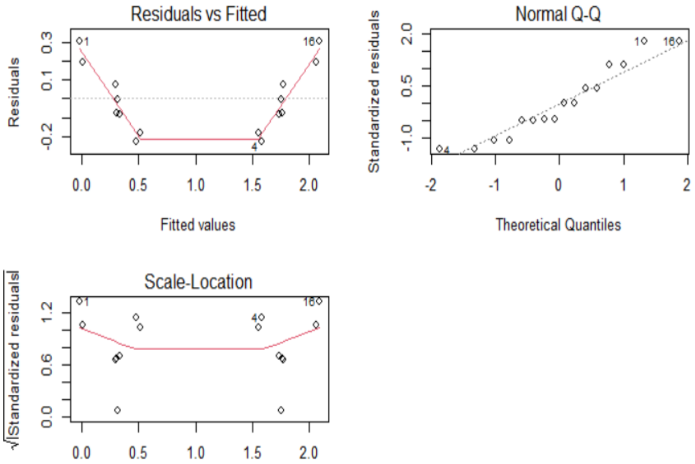
The normal q-q plot shows points are distributed linearly along the line, so the normality assumption is satisfied. The scale-location vs fitted value plot shows an approximately horizontal red line, but the spread of the points slightly varying with the fitted value, of all thing considered, we fail to reject the heteroskedasticity hypothesis

Range model

[Table 8: Raw Range model summary(left) and its residual Analysis(right)]=



```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.499e-01  4.858e-01  -1.132   0.2949
alpha        1.889e-02  1.609e-03  11.746 7.34e-06 ***
beta         8.071e-04  1.609e-03   0.502   0.6313
Aq           1.292e+05  6.434e+04   2.009   0.0845 .
d           -1.432e-01  1.287e+00  -0.111   0.9145
mt           1.274e+00  1.287e+01   0.099   0.9239
mf           1.691e+00  1.287e+01   0.131   0.8991
pin          2.475e-01  1.287e-01   1.923   0.0959 .
dzul         1.969e-03  2.574e-02   0.076   0.9412
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2574 on 7 degrees of freedom
Multiple R-squared:  0.9542,    Adjusted R-squared:  0.902
F-statistic: 18.25 on 8 and 7 DF,  p-value: 0.0004928
```



Raw Range Model:

The alpha significantly affects the range value, the r-squared is 0.9542, the p-value is 0.0004928

Residuals:

According to the residuals vs fitted plot, there is no linearity, despite that points are distributed evenly around the line, so the constant variance assumption is not satisfied.

The normal q-q plot shows some points deviating from the line, so the normality assumption is not satisfied.

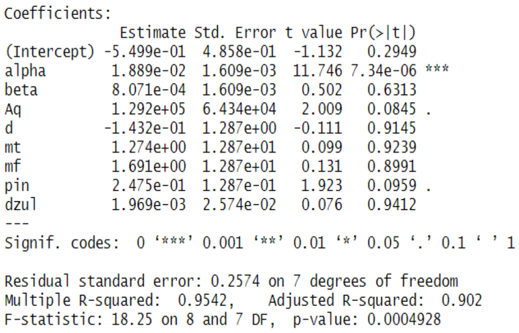
The scale-location vs fitted value plot shows a horizontal line, but the spread of points varies with the fitted values, so the residuals are heteroscedastic.

In general, the model is not satisfactory, so we do the transformation to refine the model.

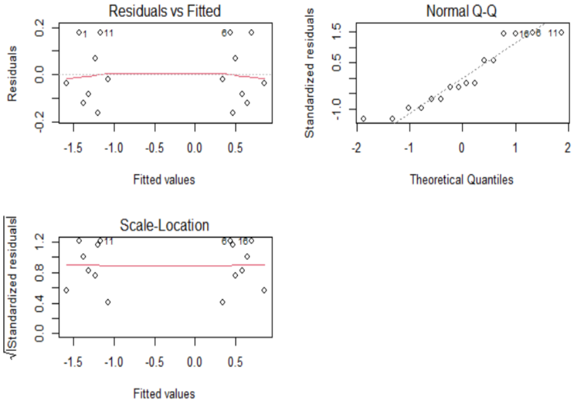
Range Transformation

Through the power transformation test, the range variable is recommended to take a log transformation.

[Table 9: Transformed Range model summary(left) and its residual Analysis(right)]



```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.499e-01  4.858e-01  -1.132   0.2949
alpha        1.889e-02  1.609e-03  11.746 7.34e-06 ***
beta         8.071e-04  1.609e-03   0.502   0.6313
Aq           1.292e+05  6.434e+04   2.009   0.0845 .
d           -1.432e-01  1.287e+00  -0.111   0.9145
mt           1.274e+00  1.287e+01   0.099   0.9239
mf           1.691e+00  1.287e+01   0.131   0.8991
pin          2.475e-01  1.287e-01   1.923   0.0959 .
dzul         1.969e-03  2.574e-02   0.076   0.9412
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2574 on 7 degrees of freedom
Multiple R-squared:  0.9542,    Adjusted R-squared:  0.902
F-statistic: 18.25 on 8 and 7 DF,  p-value: 0.0004928
```



Transformation Range model
The alpha significantly affects the range value. The r-squared is 0.9542
p-value is 0.0004928

Residuals:

According to the residual vs fitted value plot, there is a horizontal line and the points are evenly distributed, so the constant variance assumption is satisfied.

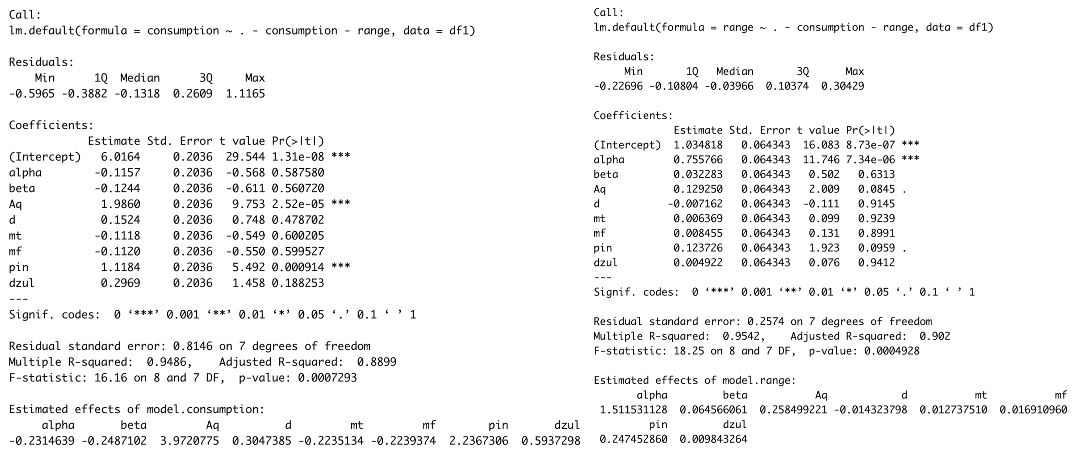
The normal q-q plot shows the points are distributed linearly along the line, so the normality assumption is satisfied.

In the scale-location vs fitted value plot, the red line is horizontal, but the spread of points varies with the fitted value, but we can conclude residuals are homoscedastic.

Question 6. What are the most influential factors?

In order to check the most influential factors in models, we have to check the p-values and the effects. The factors that have p-values smaller than alpha level, 0.05, are treated as significant factors, and we will check the effects of factors as assistance.

[Table 10: Full consumption model & effects(left) and full range model & effects(right)]



```
Call:
lm.default(formula = consumption ~ . - consumption - range, data = df1)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.5965 -0.3882 -0.1318  0.2609  1.1165 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   6.0164     0.2036  29.544 1.31e-08 ***
alpha        -0.1157     0.2036  -0.568 0.587580    
beta         -0.1244     0.2036  -0.611 0.560720    
Aq            1.9860     0.2036   9.753 2.52e-05 ***
d             0.1524     0.2036   0.748 0.478702    
mt           -0.1118     0.2036  -0.549 0.600205    
mf           -0.1120     0.2036  -0.550 0.599527    
pin           1.1184     0.2036   5.492 0.000914 ***
dzul          0.2969     0.2036   1.458 0.188253    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8146 on 7 degrees of freedom
Multiple R-squared:  0.9486,	Adjusted R-squared:  0.8899 
F-statistic: 16.16 on 8 and 7 DF,  p-value: 0.0007293

Estimated effects of model.consumption:
     alpha       beta         Aq          d         mt         mf        pin       dzul 
-0.2314639 -0.2487102  3.9720775  0.3047385 -0.2235134 -0.2239374  2.2367306  0.5937298 
```

```
Call:
lm.default(formula = range ~ . - consumption - range, data = df1)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.22696 -0.10804 -0.03966  0.10374  0.30429 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.034818   0.064343  16.083 8.73e-07 ***
alpha        0.755766   0.064343  11.746 7.34e-06 ***
beta         0.032283   0.064343   0.502   0.6313    
Aq           0.129250   0.064343   2.009   0.0845 .  
d           -0.007162   0.064343  -0.111   0.9145    
mt           0.006369   0.064343   0.099   0.9239    
mf           0.008455   0.064343   0.131   0.8991    
pin          0.123726   0.064343   1.923   0.0959 .  
dzul         0.004922   0.064343   0.076   0.9412    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2574 on 7 degrees of freedom
Multiple R-squared:  0.9542,	Adjusted R-squared:  0.902 
F-statistic: 18.25 on 8 and 7 DF,  p-value: 0.0004928

Estimated effects of model.range:
       alpha         beta           Aq            d           mt           mf 
 1.511531128  0.064566061  0.258499221 -0.014323798  0.012737510  0.016910960 
         pin         dzul 
 0.247452860  0.009843264 
```

For the consumption full model, Aq and pin are the most influential factors.
For the range full model, alpha is the most influential factor.

Question 7. Recommend the settings of the factors that optimize the water consumption and spray range simultaneously.

Due to model performance and linear regression assumptions, we operate the optimization process on the transformed reduced model. We get the following formula from powerTransform. These two formulas improve the R-squares of the models and have better satisfaction with the linear regression assumptions.

1. consumption^(-0.35) ~ Aq + pin
2. log(range) ~ alpha

Then, we use optim() to minimize water consumption and maximize range.

[Table 11: Optim() results of transformed consumption model(left) and transformed range model(right)]

```
$par
[1] -1 -1

$value
[1] 3.374649

$counts
function gradient
       3        3

$convergence
[1] 0

$message
[1] "CONVERGENCE: NORM OF PROJECTED GRADIENT <= PGTOL"
```

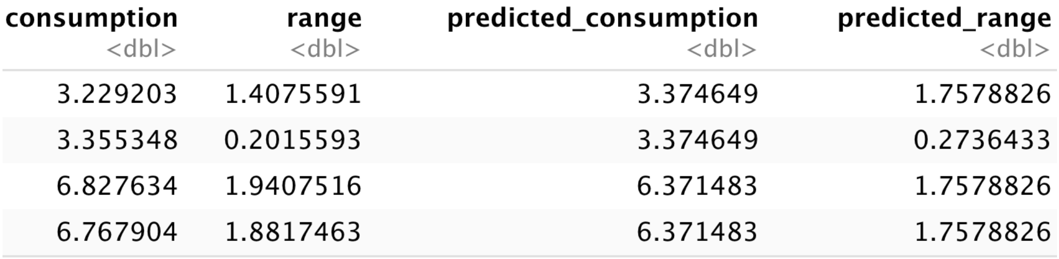
```
$par
[1] 1

$value
[1] -1.757883

$counts
function gradient
       3        3

$convergence
[1] 0

$message
[1] "CONVERGENCE: NORM OF PROJECTED GRADIENT <= PGTOL"
```

The optimum setting to minimize water consumption is Aq = 2e-06, pin = 1.
The optimum setting to maximize range is alpha = 80.
Since the influential factors for the two models are different, we can combine those factors.
Therefore, the recommended setting is Aq = 2e-06, pin = 1, alpha = 80.

Question 8. Conduct confirmation experiments using your recommended settings. Are your predictions accurate?

Since we have used 16 out of 20 runs, there are 4 runs left for us as confirmation runs. To secure we test both model's performance, the first 3 runs are generated randomly, but the influential factors are modified.
For the 1st run, we use the recommended setting: Aq = 2e-06, pin = 1, alpha = 80.
For the 2nd run, we use the optimum setting for consumption: Aq = 2e-06, pin = 1.
For the 3rd run, we use the optimum setting for range: alpha = 80.
For the 4th run, we randomly generate all the levels randomly.

[Table 12: Confirmation runs' results]



consumption	range	predicted_consumption	predicted_range
<dbl>	<dbl>	<dbl>	<dbl>
3.229203	1.4075591	3.374649	1.7578826
3.355348	0.2015593	3.374649	0.2736433
6.827634	1.9407516	6.371483	1.7578826
6.767904	1.8817463	6.371483	1.7578826

From the above results, we observe that the recommended setting has the smallest true water consumption and medium range. Compared with other runs, it balances the consumption and range. The performance of the models is considerably good. With only naked eyes, we may observe predicted values are close to the true values.

Running cor() between true values and predicted values, we get 0.9996039 for consumption and 0.9552685 for range.

3) *Conclusions and Recommendations*

Overall, the design is considered successful because its strengths including:

Strength:

- High R score of 0.98 for the refined consumption model and 0.95 for the refined range model.
- Reasonable Residuals check so that the constant variance and normality assumption is met
- High confirmation accuracy between predictions and the actual value with a correlation value of 0.9996039 for the consumption model and 0.9552685 for the range model.

But it's not perfect due to __ weakness:

Weakness:

- The assumption that greater than 3 factors interaction effects are zero.
- Only two levels for each factor are study

For future experimentation, the prediction model can be improved by having more budget runs to allow more levels of experimental design. In that case, more levels of the factor will be studied, and potential interaction between factors will also be evaluated.

4) Appendix.

```
# Q1:
Based on the the Appendix X from Textbook # E = BCD; F=ACD;
G = ABC; H = ABD
set.seed(1)
#Generate our design
my.design <- FrF2(16, 8, generators=
c("BCD","ACD","ABC","ABD"),randomize=F)
my.design.matrix <- desnum(my.design)
my.design.matrix
#Transfer the results into true values
copy_my.design.matrix = my.design.matrix
#OUTPUT WITH TRUE VALUE
my.design.matrix[,1:2][my.design.matrix[,1:2]==-1] <- 0
my.design.matrix[,1:2][my.design.matrix[,1:2]==1] <- 80
my.design.matrix[,3][my.design.matrix[,3]==-1] <- 0.000002
my.design.matrix[,3][my.design.matrix[,3]==1] <- 0.000004
my.design.matrix[,4][my.design.matrix[,4]==-1] <- 0.1
my.design.matrix[,4][my.design.matrix[,4]==1] <- 0.2
my.design.matrix[,5:6][my.design.matrix[,5:6]==-1] <- 0.01
my.design.matrix[,5:6][my.design.matrix[,5:6]==1] <- 0.02
my.design.matrix[,7][my.design.matrix[,7]== 1 ] <- 2
my.design.matrix[,7][my.design.matrix[,7]== -1 ] <- 1
my.design.matrix[,8][my.design.matrix[,8]==-1] <- 5
my.design.matrix[,8][my.design.matrix[,8]==1] <- 10
write.table(my.design.matrix, "my_design_matrix.txt",  sep="\t",
quote=FALSE, dec=".", row.names=FALSE,col.names=FALSE)


# Q2:
#Check the main effect alias
X.two.level <- model.matrix(~(A + B + C + D + E + F + G +
H)-1,data.frame(copy_my.design.matrix))
contrast.vectors.correlations.two <- cor(X.two.level)
corrplot(contrast.vectors.correlations.two,type="full",addgrid.col=
"gray",tl.col="black",tl.srt=90,method="color",tl.cex=0.8)
#Check the two interaction alias
X.two.level <- model.matrix(~(A + B + C + D + E + F + G +
H)^2-1,data.frame(copy_my.design.matrix))
contrast.vectors.correlations.two <- cor(X.two.level)
corrplot(contrast.vectors.correlations.two,type="full",addgrid.col=
"gray",tl.col="black",tl.srt=90,method="color",tl.cex=0.8)
#Check VIF for main effect only
X.opt.me <- model.matrix(~., data.frame(copy_my.design.matrix))
XtX <- t(X.opt.me)%*%X.opt.me
inv.XtX <- solve(XtX)
var.eff <- diag(inv.XtX)
cat("Variance the estimates when sigma^2 = 1 \n")
print(var.eff)
cat("\n Variance inflation factors \n")
print(nrow(my.design.matrix)*var.eff)

# Q3:
#Generate two random rows to be removed
remove = sample(1:16,2,replace = F)
cat("Those two rows will be removed: ", remove)
my.design.matrix.remove = copy_my.design.matrix[-remove,]
#Check its evaluation
X.two.level.remove <- model.matrix(~(A + B + C + D + E + F +
G + H)^2-1,data.frame(my.design.matrix.remove))
contrast.vectors.correlations.two.remove <-
cor(X.two.level.remove)
corrplot(contrast.vectors.correlations.two.remove,type="full",addgrid
.col="gray",tl.col="black",tl.srt=90,method="color",tl.cex=0.8)

# Q4:
# The data use the 3 level fractional factorial design
# Convert it into the coded experimental data
Alpha = c(-1,0,-1,-1,1,-1,1,-1,-1,1,1,1,-1,1,0,0,1) # 0, 45, 90
Beta = c(1,0,1,-1,-1,0,1,-1,-1,-1,1,1,1,-1,-1,1,0) # 0, 45, 90
A_q = c(-1,0,1,1,-1,-1,-1,1,-1,1,1,-1,0,0,-1,1,1) #
0.00002,0.00003,0.00004
d = c(1,0,0,-1,0,-1,-1,1,1,-1,-1,1,-1,1,-1,1,1) # 0.01,0.015,0.02
M_t = c(1,0,-1,1,1,1,0,0,-1,-1,1,-1,-1,1,-1,1,-1) # 0.01,0.015,0.02
M_f = c(0,0,-1,1,1,-1,1,-1,1,0,-1,-1,1,-1,-1,1,1) # 0.01,0.015,0.02
p_in = c(-1,0,1,0,-1,1,1,-1,1,1,-1,0,-1,1,-1,1,-1) # 1, 1.5, 2
d_zul = c(-1,0,-1,-1,1,1,-1,1,0,1,0,1,1,-1,-1,1,-1) # 5, 7.5,10
alternative.design.4 =
data.frame("A"=Alpha,"B"=Beta,"C"=A_q,"D"=d,"E"=M_t,"F"
=M_f,"G"=p_in,"H"=d_zul)
alternative.matrix <- model.matrix(~(A + B + C + D + E + F +
G + H)^2-1,data.frame(alternative.design.4))
alternative.correlations.three.remove <- cor(alternative.matrix)
corrplot(alternative.correlations.three.remove,type="full",addgrid
.col="gray",tl.col="black",tl.srt=90,method="color",tl.cex=0.8)
write.table(alternative.design.4, "alternative_design_4.txt",
sep="\t", quote=FALSE,
dec=".",row.names=FALSE,col.names=FALSE)
#Check the VIF
X.opt.me.4 <- model.matrix(~., alternative.design.4)
XtX.4 <- t(X.opt.me.4)%*%X.opt.me.4
inv.XtX.4 <- solve(XtX.4)
var.eff.4 <- diag(inv.XtX.4)
cat("Variance the estimates when sigma^2 = 1 \n")
print(var.eff.4)
cat("\n Variance inflation factors \n")
print(nrow(alternative.design.4)*var.eff.4)

# Q5:
library(readr)
# after the experimentation, we import the result data
result_my_design <- read_csv("result.txt")
result_my_design
# conduct the transformation test if the responses are needed to be
transfer
# will be used if the model have large residuals
library(car)
attach(result_my_design)
summary(powerTransform(cbind(range,consumption)))
```

```
result_my_design[range==0,]
#Consumption Model
model.consumption =
lm(consumption~.-consumption-range,data=result_my_design)
summary(model.consumption)
#Check the residuals
par(mfrow=c(2,2))
plot(model.consumption)
#Consumption Model after the transformation suggested by the
power transform
model.consumption_transform =
lm(consumption^(-0.25)~.-consumption-range,data=result_my_de
sign)
summary(model.consumption)
par(mfrow=c(2,2))
plot(model.consumption_transform)
#Range Model and inspection on its residuals
model.range =
lm(range~.-consumption-range,data=result_my_design)
summary(model.range)
par(mfrow=c(2,2))
plot(model.range)
#Range Model after the transformation and inspection on its
residuals
model.range_transform =
lm(log(range)~.-consumption-range,data=result_my_design)
summary(model.range)
par(mfrow=c(2,2))
plot(model.range_transform)
```

```
# Q6:
# model.consumption
df1 <- cbind(copy_my.design.matrix, result_my_design[9:10])
names(df1) <- names(result_my_design)
model.consumption.original <-
lm(consumption~.-consumption-range,data=df1)
summary(model.consumption.original)
#2*coefficient of the model are the main effects for them
consumption_effects <- 2*(coef(model.consumption.original)[-1])
cat("Estimated effects of model.consumption: \n")
print(consumption_effects)

# model.range
model.range.original <-
lm(range~.-consumption-range,data=df1)
summary(model.range.original)
#2*coefficient of the model are the main effects for them
range_effects <- 2*(coef(model.range.original)[-1])
cat("Estimated effects of model.range: \n")
print(range_effects)
```

```
# Q7:
# model consumption
model.consumption.refined <- lm(consumption ~ Aq + pin, data =
df1)
summary(model.consumption.refined)
#check the residuals of the refined model of consumption
par(mfrow=c(2,2))
plot(model.consumption.refined)

# reconduct th power transform
powerTransform(model.consumption.refined)
model.consumption.refined_transform <- lm(consumption^(-0.35)
~ Aq + pin, data = df1)
summary(model.consumption.refined_transform)
par(mfrow=c(2,2))
plot(model.consumption.refined_transform)

# model range
model.range.refined <- lm(range ~ alpha, data = df1)
summary(model.range.refined)
# check th residuals
par(mfrow=c(2,2))
plot(model.range.refined)

#power transfer is needed
powerTransform(model.range.refined)
model.range.refined_transform <- lm(log(range) ~ alpha, data =
df1)
summary(model.range.refined_transform)
par(mfrow=c(2,2))
plot(model.range.refined_transform)

# conduct the optium test to find the min value of the consumption
obj_func1 <- function(x){
  result <- 0.553145 - 0.065147*x[1] - 0.035019*x[2]
  return(result^(-1/0.35))}
optim(par = c(0, 0), fn = obj_func1, lower = -1, upper = 1,
method = "L-BFGS-B")

# conduct the optium test to find the max value of the range
obj_func2 <- function(x){
  results <- -0.36591 + 0.93002 * x
  return(-exp(results))}
optim(par = c(0), fn = obj_func2, lower = -1, upper = 1, method
= "L-BFGS-B")
```

```
# Q8:
# Recommended Setting
set.seed(1000)
sample1 <- sample(c(-1,1), size = 8, replace = T)
sample1[c(1,3,7)] <- c(1, -1, -1)
# Fix Aq, pin
sample2 <- sample(c(-1,1), size = 8, replace = T)
sample2[c(3,7)] <- c(-1, -1)
# Fix alpha
sample3 <- sample(c(-1,1), size = 8, replace = T)
sample3[1] <- 1
# Random
sample4 <- sample(c(-1,1), size = 8, replace = T)
# Confirmation Runs
df2 <- rbind(sample1, sample2, sample3, sample4)
temp <- df2
df2[,1:2][df2[,1:2]==-1] <- 0
df2[,1:2][df2[,1:2]==1] <- 80
df2[,3][df2[,3]==-1] <- 0.000002
```

```
df2[,3][df2[,3]==1] <- 0.000004
df2[,4][df2[,4]==-1] <- 0.1
df2[,4][df2[,4]==1] <- 0.2
df2[,5:6][df2[,5:6]==-1] <- 0.01
df2[,5:6][df2[,5:6]==1] <- 0.02
df2[,7][df2[,7]== 1 ] <- 2
df2[,7][df2[,7]== -1 ] <- 1
df2[,8][df2[,8]==-1] <- 5
df2[,8][df2[,8]==1] <- 10
# export the data
write.table(df2, "confirmation_runs.txt", sep="\t", quote=FALSE,
dec=".",row.names=FALSE,col.names=FALSE)
# import the confirmaiton test results
confirmation_result <- read_csv("confirmation_runs_result.txt")
confirmation_result
# for the same setting, predict the confirmation samples
consumption
predicted_consumption <- rep(NA, 4)
for(i in 1:4){
  predicted_consumption[i] <- obj_func1(temp[i ,c(3,7)])}
# for the same setting, predict the confirmation samples range
predicted_range <- rep(NA, 4)
for(i in 1:4){
  predicted_range[i] <- -obj_func2(temp[i ,1])}
# check the results
confirmation_result <- cbind(confirmation_result,
predicted_consumption, predicted_range)
print(confirmation_result)
#perform an confirmation test of the model by checking their
correlations
cor(confirmation_result[9],confirmation_result[11])
cor(confirmation_result[10],confirmation_result[12])
```

5) Statement of Contribution. Clearly state the contribution of each team member to this project and report.
Benson Zu: team leader, organize the project, complete questions 1-6, write reports.
Chenkai Li: team member, complete questions 6, 7, and 8, write reports.
Changhao Chen: write reports.

6) References.

Garden sprinkler simulation. Garden Sprinkler Simulator. (n.d.). http://twilights.be/sprinkler/.