Applied Statistic – Final Project Part-1

Firstly, with samples in excel :

ASTGEL… fx =(E46-(((((273+E37)*E21*E46)/(E33-2))^(1/2))*0,0238))*(((273+E37)*E21*0,0239)/((E33-2)*E30))

	B	C	D	E	F	G	H	I	J	K	L	M
16	Lab Temperature / *Lab Sıcaklığı* [°C]			21								
17							Person in Charge					
18	Sample *Numune*		Sample No / *Numune No*	1	2	3						
19		L_1	Thickness of Sample-1 / *Numunenin Kalınlığı-1, [mm]*	[illegible]	50,41	52,42						
20		L_2	Thickness of Sample-2 / *Numunenin Kalınlığı-2, [mm]*	51,71	50,59	52,72						
21			Average of Thickness	51,41	50,50	52,57						
22	Conditioning *Hazırlama*		Start Time of Sample in Vacuum (50 mmHg, 3h) / *Numunenin vakuma konulduğu saat (50 mmHg), [dd:mm:yyyy, hh:mm]*									
23			Start Time of Sample Immersed in Solution (1h) / *Numunenin çözeltiye konulduğu saat, [dd:mm:yyyy, hh:mm]*									
24			Time of Sample out of Bucket (After 18±2 h) / *Numunenin desikatörden çıkarıldığı saat, [dd:mm:yyyy, hh:mm]*									
25	Start *Başlangıç*	T_1	NaOH Temperature at Start / *Başlangıçtaki NaOH Sıcaklığı, [°C]*	23,5	23,5	23,5			51,4	50,5	52,6	0,0
26			Initial Voltage / *İlk Voltaj, [V]*	30	30	30						
27			Initial Current / *İlk Akım, [mA]*	43,50	42,60	40,10						
28		U	Starting Voltage / *Başlangıçtaki Voltaj, [V]*	30	30	30						
29			Starting Current / *Başlangıçtaki Akım, [mA]*	43,50	42,60	40,10						
30			Test Duration / *Test Süresi, [h]*	24	24	24						
31		t_1	Starting Time When New Voltage Is Applied / *Yeni Voltaj Verildiği Andaki Başlangıç Saati, [dd:mm:yyyy, hh:mm]*	8.08.2017	8.08.2017	8.08.2017						
32			Specimen Age, [days]	28	28	28						
33	End *Bitiş*		Ending Voltage / *Bitiş Voltajı, [V]*	30	30	30						
34			Ending Current / *Bitiş Akımı, [mA]*	48,80	45,70	40,80						
35		t_2	Ending Time / *Bitiş Zamanı, [dd:mm:yyyy, hh:mm]*	9.08.2017	9.08.2017	9.08.2017						
36		T_2	Ending Temperature of NaOH / *NaOH Son Sıcaklığı, [°C]*	24,3	24,3	23,8						
37			Average of NaOH Temperature	23,9	23,9	23,65						
38	Measuring *Ölçüm*		Chloride Ingress-1 / *Klor Derinliği-1, [mm]*	13,41	11,02	13,19						
39			Chloride Ingress-2 / *Klor Derinliği-2, [mm]*	12,21	9,72	11,35						
40			Chloride Ingress-3 / *Klor Derinliği-3, [mm]*	13,06	10,85	12,71						
41			Chloride Ingress-4 / *Klor Derinliği-4, [mm]*	13,23	11,71	13,85						
42			Chloride Ingress-5 / *Klor Derinliği-5, [mm]*	14,86	10,82	10,23						
43			Chloride Ingress-6 / *Klor Derinliği-6, [mm]*	23,55	15,20	17,14						
44			Chloride Ingress-7 / *Klor Derinliği-7, [mm]*	16,74	10,91	9,85						
45			Max. Depth of Chloride Ingress / *Max. Klor Derinliği, [mm]*	24,19	16,05	17,45						
46			Average Chloride Ingress, [mm] Xd	15,29	11,46	12,62						
47	Sample Dimensions *Numune Ölçüleri*		Mass in Air / *Havadaki Ağırlık, [g]*	970	954	973						
48			Mass in Water / *Sudaki Ağırlık, [g]*	569	559	567						
49			Saturated Unit Weight / Doygun Birim Ağırlık [g/cm^3]	2419	2415	2397						
50			Diameter / *Çap, [mm]*	100	100	100						
51			Dnssm (non-steady-state migration coefficient, m2/s;)	2)*E30))	5,1175413	5,89139413						

Independent variables in the formula forming the migration coefficient were found.

R-DATA PREPARATION :

(When running rds → The path specified in the file must be changed)

```
## Excel Operations
```{r pressure, echo=FALSE}
setwd("//Users//macboookair//Downloads//CM-RawData")
excel_data <- read_excel("YCMR-XXX 14.08.2017 G3HL Deneme 2.xlsx",sheet="Record_CMC")
```

Then, using R programming language :

→ Received data on Record_CMC sheet.

→ Found rows with independent variables.(Renamed column names etc.)

→ Respectively, merged data from 34 excel locations.

→34*3 = 102 data sample merging. (34 excel and 3 sample on each excel.)
```

```r
excel_data <- read_excel("YCMR-XXX 14.08.2017 G3HL Deneme 2.xlsx",sheet="Record_CMC")
## Kullandığımız datayı almak icin olcum yapilan yerler alinir
excel_data_1 <- excel_data %>% slice(17:51)
## Olcum sonucu ilgili kolonlar secilerek yeni df ye aktarilir
dfNew <- excel_data_1[,c(4:6)]

dfNew_1_factor <- dfNew[3,]
dfNew_2_factor <- dfNew[12,]
dfNew_3_factor <- dfNew[15,]
dfNew_4_factor <- dfNew[19,]
dfNew_5_factor <- dfNew[28,]
dfNew_ME_factor <- dfNew[33,]
```

(Same process for every excel)

After data merging:

	y	x1	x2	x3	x4	x5
1	7.1229118394783226	51.41	24	30	23.9	15.294285714285712
2	5.1175412878446522	50.5	24	30	23.9	11.461428571428572
3	5.8913941287808891	52.57	24	30	23.65	23.65
4	3.5003630201665183	51.384999999999998	24	40	24.3	10.272857142857143
5	3.746300075618004	51.81	24	40	24.3	10.860000000000001
6	3.2008295168693022	52.125	24	40	24.3	9.3514285714285705
7	3.6343439856487789	51.31	24	50	24.9	12.991428571428571
8	2.673882714516469	51.17	24	50	24.9	9.7814285714285738
9	3.2910779969053054	52.144999999999996	24	50	24.75	11.680000000000001
10	3.778446584754291	51.034999999999997	24	50	24.950000000000003	13.535714285714283
11	2.4894839696572251	51.015000000000001	24	50	24.8	9.1814285714285724
12	2.1740865782447463	52.515000000000001	24	50	24.75	7.9099999999999993
13	2.8704098347666958	51.010000000000005	24	40	24.65	8.6100000000000012
14	3.3024425591158799	51.16	24	40	24.65	9.7642857142857142
15	3.6871085985161107	52.215000000000003	24	40	24.65	10.62142857142857
16	3.3174682847411674	50.53	24	40	24.6	9.9085714285714293
17	3.3159906045894947	51.489999999999995	24	40	24.65	9.7485714285714273
18	3.2434503791638019	51.36	24	40	24.55	9.5757142857142856
19	4.479989880319124	51.94	24	50	23.9	15.692857142857141
20	6.1380250675144845	51.480000000000004	24	50	24.35	21.288571428571426
21	5.1707634657610999	52.064999999999998	24	50	24.2	17.915714285714287
22	5.4048585381434098	51.445	24	40	23.7	15.382857142857144
23	4.7929995677438066	50.89	24	40	23.75	13.87857142857143
24	5.7755259296049903	50.989999999999995	24	40	23.8	16.491428571428571
25	4.7686340414800199	51.204999999999998	24	40	23.7	13.741428571428571
26	2.6261397217291531	51.314999999999998	24	40	23.7	7.9228571428571444
27	3.5942550332639183	50.989999999999995	24	40	23.65	10.614285714285714
28	7.1094067071323703	51.284999999999997	24	40	23.4	19.974285714285713
29	5.7538448994087865	51.884999999999998	24	40	23.4	16.204285714285714
30	3.3860543682374131	51.78	24	40	23.4	9.9257142857142835
31	2.6984340507556857	51.474999999999994	24	50	23.35	9.8585714285714285
32	4.3328221529192419	51.725000000000001	24	50	23.450000000000003	15.282857142857141
33	2.588847341172011	51.43	24	50	23.55	9.4885714285714293
34	4.0785771013760339	50.93	24	40	23.1	11.959999999999999
35	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

(34*3 = 102 data)

Y -> Migration Coefficient(dnssm)

X1 -> Average Of Thickness

X2 -> Test Duration

X3 -> Ending Voltage

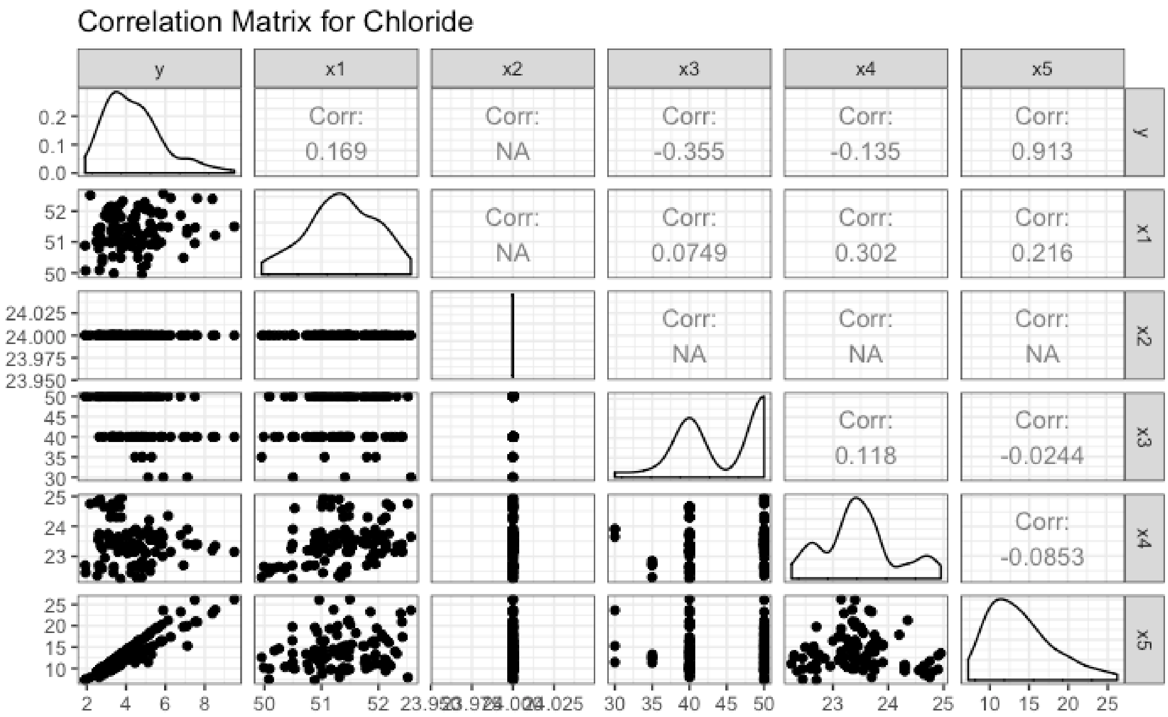
X4 ->Average of NaOH temperature

X5 -> Average Chloride Ingress

Data Analysis Step :

Correlation matrix was found to determine how much each independent variable affects the y value.
cor(dataFrameForMergingData) :

```
the standard deviation is zero            y          x1 x2          x3          x4          x5
y   1.0000000 0.16936953 NA -0.35462282 -0.13470247  0.91264675
x1  0.1693695 1.00000000 NA  0.07486525  0.30178589  0.21593835
x2         NA         NA  1          NA          NA          NA
x3 -0.3546228 0.07486525 NA  1.00000000  0.11836142 -0.02438902
x4 -0.1347025 0.30178589 NA  0.11836142  1.00000000 -0.08526458
x5  0.9126467 0.21593835 NA -0.02438902 -0.08526458  1.00000000
```



a correlation test was performed for each factor :

res_x4 <- cor.test(migration_data$y, migration_data$x4,
            method = "pearson")

```
        Pearson's product-moment correlation

data:  migration_data$y and migration_data$x4
t = -1.3594, df = 100, p-value = 0.1771
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.32077428  0.06138036
sample estimates:
       cor
-0.1347025


        Pearson's product-moment correlation

data:  migration_data$y and migration_data$x1
t = 1.7185, df = 100, p-value = 0.0888
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.02596046  0.35224223
sample estimates:
      cor
0.1693695
```

the standard deviation is zero

```
        Pearson's product-moment correlation

data:  migration_data$y and migration_data$x2
t = NA, df = 100, p-value = NA
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 NA NA
sample estimates:
cor
 NA

        Pearson's product-moment correlation

data:  migration_data$y and migration_data$x3
t = -3.7927, df = 100, p-value = 0.0002552
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.5136722 -0.1720107
sample estimates:
       cor
-0.3546228


        Pearson's product-moment correlation

data:  migration_data$y and migration_data$x5
t = 22.328, df = 100, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8731433 0.9402412
sample estimates:
      cor
0.9126467
```

After the correlation testing :

P-value was compared with the significant value(0.05) :

```
res_x4 <- cor.test(migration_data$y, migration_data$x4,
                   method = "pearson")
res_x4#p-value = 0.1771 p_Value larger than 0.05
#Correlation coefficient -> -0.1347025

res_x1 <- cor.test(migration_data$y, migration_data$x1,
                   method = "pearson")
res_x1 #p-value = p-value = 0.0888 p_Value larger than 0.05.
#Correlation coefficient -> 0.1693695

res_x2 <- cor.test(migration_data$y, migration_data$x2,
                   method = "pearson")
res_x2#p-value = NA

res_x3 <- cor.test(migration_data$y, migration_data$x3,
                   method = "pearson")
res_x3#p-value = 0.0002552 p_Value less than 0.05
# We can conclude that y and x3 are significantly correlated with a correlation coefficient of -0.3546228  and p-value of 0.0002552 .

res_x5 <- cor.test(migration_data$y, migration_data$x5,
                   method = "pearson")
res_x5#p-value = p-value < 2.2e-16 p_Value larger than 0.05
# We can conclude that y and x3 are significantly correlated with a correlation coefficient of 0.9126467 and p-value of 2.2e-16 .
```

Found :
X3 and x5 → significantly correlated

Then :
Multiple Regression applied
First , applied for all factors(5 factors):

```
Call:
lm(formula = y ~ x1 + x2 + x3 + x4 + x5, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.86135 -0.09795 -0.01359  0.06551  1.06431 

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.142910   3.090331   1.341    0.183    
x1           0.013639   0.063968   0.213    0.832    
x2                 NA         NA      NA       NA    
x3          -0.085218   0.006254 -13.626   <2e-16 ***
x4          -0.046446   0.058588  -0.793    0.430    
x5           0.319148   0.008843  36.089   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3643 on 97 degrees of freedom
Multiple R-squared:  0.9438,	Adjusted R-squared:  0.9415 
F-statistic: 407.4 on 4 and 97 DF,  p-value: < 2.2e-16
```

(Coefficients find by --coefficients(fit) # model coefficients)
When the equation is created :
Y <- 4.142910 + 0.013639 * 51.41 + -0.085218*30 + -0.046446*23.9 + 0.319148*15.29
(values taken by "YCMR-XXX 14.08.2017 G3HL Deneme 2.xlsx")

Result : [1] 6.057265

Then, I removed some variables from the equation based on the p values resulting from the correlation test :

First – I removed Second variable :

```
Call:
lm(formula = y ~ x1 + x3 + x4 + x5, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.86135 -0.09795 -0.01359  0.06551  1.06431 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.142910   3.090331   1.341    0.183    
x1           0.013639   0.063968   0.213    0.832    
x3          -0.085218   0.006254 -13.626   <2e-16 ***
x4          -0.046446   0.058588  -0.793    0.430    
x5           0.319148   0.008843  36.089   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3643 on 97 degrees of freedom
Multiple R-squared:  0.9438,	Adjusted R-squared:  0.9415 
F-statistic: 407.4 on 4 and 97 DF,  p-value: < 2.2e-16
```

But I realized that the same with the first equation. Because SD is 0.
Result : [1] 6.057265(Result is the same with first.)

Then I removed 2 and 4 factor.

```
Call:
lm(formula = y ~ x1 + x3 + x5, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.86786 -0.09820 -0.01228  0.07224  1.03648 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.899548   3.069211   1.271    0.207    
x1          -0.002733   0.060428  -0.045    0.964    
x3          -0.085691   0.006214 -13.791   <2e-16 ***
x5           0.320258   0.008715  36.747   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3636 on 98 degrees of freedom
Multiple R-squared:  0.9435,	Adjusted R-squared:  0.9417 
F-statistic: 545.1 on 3 and 98 DF,  p-value: < 2.2e-16
```

Result : 3.899548 + (-0.002733 * 51.41) + (-0.085691*30) + (0.320258*15.29) = 6.085059

Then I removed 1,2 and 4 factor.

```
Call:
lm(formula = y ~ x3 + x5, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.87068 -0.09774 -0.01032  0.07140  1.03610 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.761437   0.305642   12.31   <2e-16 ***
x3          -0.085714   0.006161  -13.91   <2e-16 ***
x5           0.320172   0.008462   37.84   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3617 on 99 degrees of freedom
Multiple R-squared:  0.9435,	Adjusted R-squared:  0.9423 
F-statistic: 825.9 on 2 and 99 DF,  p-value: < 2.2e-16
```

When I looked at the improvement in the results, I found that the best improvement was made by removing the factors 1,2 and 4.

Last Equation :

`y_last <-  3.761437  + (-0.085714*30) + (0.320172*15.29)  =` **6.085447**

y = 3.761437 + (-0.085714)*x3 + (0.320172)*x5

Comparing factors

- Results (Equation results compared -- closest to real value is sussessful)
- Adjusted R-Squared(used to compare the accuracy of the models – The largest value is true)
- Residual standard error (Small is success)

	Results	Adjusted-R-Squared	RSE
First Version – With Five Factors	6.057265	0.9415	0.3643
Second Factor Removed	6.057265	0.9415	0.3643
Second and fourth Factor Removed	6.085059	0.9417	0.3636
First,Second and Fourth Factor Removed	6.085447	0.9423	0.3617