University of Bath

Integrated Design Engineering

Design Optimisation Project

Optimisation Skills Report

Author
Ege Arabul

October 9, 2020

Contents

1 Assignment 1a - Analytical Optimisation

Analytical Optimisation refers to the calculation of a set of optimal solutions based for a given set of parameters. The initial task involves optimising the space within a box, which is to be filled with cylinders [Flynn, 2020]. Packing problems are a common case where the result can be calculated analytically. The process involved determining expressions for the dimensions of the box, minimising the number of parameters and achieving a density ratio (the ratio the box can be filled as a percentage) by finding the correct parameters.

Early literature review shows there are two potential packing options, which are as illustrated in Figure 1. As shown, the triangular packing option involves placing the cylinders (initially investigated as circles at the bottom of the box for simplicity) at corners [Nrich, 2019]. This leads the cylinder rows to be $r\sqrt{3}$ (r being the radius) distance apart, leading to a higher density ratio compared to rectangular configuration, where cylinders are placed side by side and 2r apart. Therefore, the triangular packing method was preferred.

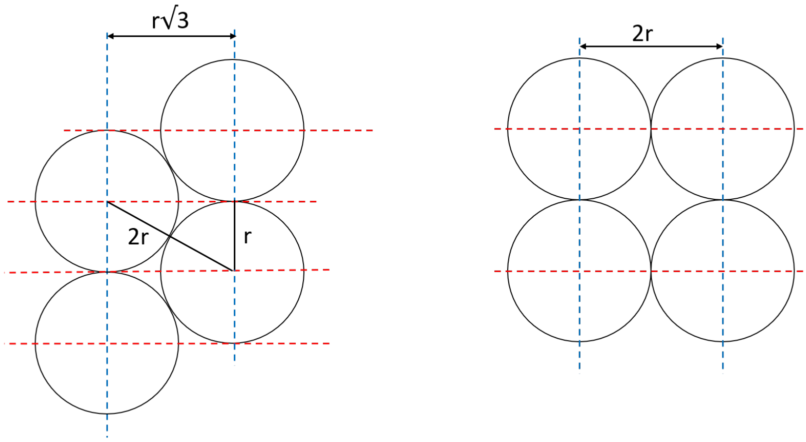

Figure 1: Comparison of two packing options,left to right,triangular and rectangular patterns respectively

Once the packing method was chosen, expressions for the unknown parameters were to be determined.For m circles in each column and n columns, two expressions for sides of the rectangle could be derived, which are $(2m+1)*r$ and $(2+(n-1)\sqrt{3})*r$. As the given box is cuboid shaped , these two equations were equated, leading to an expression for n in terms of m and r [Bower, 2020]. Furthermore, as the volume of a cylinder is fixed, it can be used to derive an expression for height (h) of the cylinder in terms of r. With these equation, an expression for the number of cylinders could be derived, with constraint being the size of the box.

With the expressions were determined, the process was iterated until a satisfactory density ratio was achieved. Literature review suggests that rectangular pattern can only have up to 70% density ratio whereas triangular pattern could allow for a density ratio up to 90% . One inspiration for the radius of the cylinders was the dimensions of a generic soda can which also has a volume of 500ml. By solving for m in the equation, a total of 605 cylinders were calculated to be possible, along with the key dimensions shown in table 1. While this configuration occupies occupies only the 78% of the volume of the box, it offers a higher performance then the rectangular pattern while still being conservative to avoid overloading the box. The complete calculations of the assignment can be found in Appendix A.

Optimal Radius of each can	33.1
Optimal Height of each can	145.3
Number of Cans that can be fitted within the Crate	605

Table 1: Summary of the results for the packing problem

2 Assignment 1b - Design of Experiments

Design of experiments is the study of planning, conducting, analysing and interpreting controlled tests to evaluate the factors that control the value of a parameter or group of parameters [Moore, 2016].

Figure 2 illustrates the experimental setup. It involves use of a fan to simulate the wind, pointed towards the model wind turbine, which generates lift as the air travels through the blades that are placed at certain angle, the pitch angle. The generated torque by the turbine is then transmitted through the gears to a motor, where the rotation produces torque. This torque can be characterised as voltages or amperes. Note that for measuring amperes, there needs to be a resistance in series connected to the turbine. The generated torque, which is to the proportional rotor RPM and voltage, is then measured using a voltmeter.

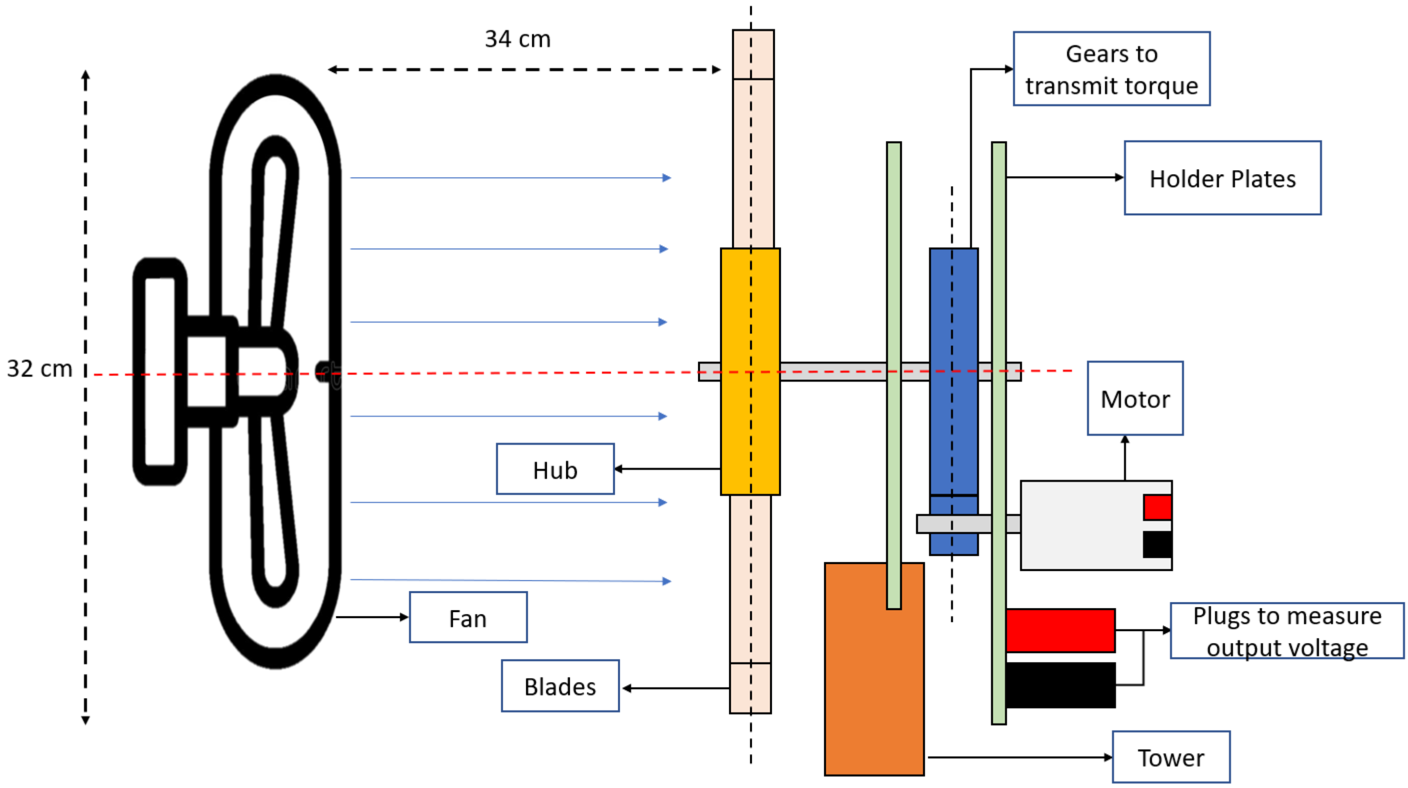

Figure 2: Diagram of the experimental setup

Within the context of the experiment, parameters affecting the performance of the aerofoil were investigated as a learning experience. These included the pitch angle, chord length and the camber of the blades. The pitch angle determines the angle between the chord of an aerofoil and the level surface, which can be ground [Smithsonian, 2016]. While ,conventionally, pitch angle is usually not constant throughout the blade in a wind turbine, the experimental setup allows a simplified model for investigating the parameter. The pitch angle had 3 levels. This is due to the reduction in lift generated at high extreme angles closer to 0 and 90 and such distribution can only be covered in 3 levels.Note that within the experiment, the pitch has been accepted to be positive counter clockwise.

Second factor, the chord length, involved varying length of the chord constantly throughout its' span. Literature reviews suggests the chord length (c) is used in significant as it appears in dimensionless groups such as the Reynolds number as well as effecting the total blade area. For observing its' effect, 2 levels were chosen as this would mean halving the area of the blade,expected to have significant effects.Another interesting investigation could have been to vary different distribution of the chord length along the length of the blade, which is a relevant concept in wind turbine design [Mao, 2016].

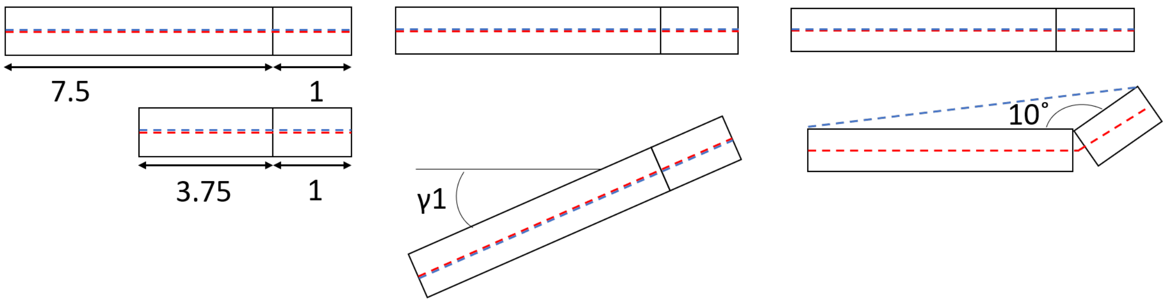

Figure 3: Side view of the of blade with varying parameters.From left to right, chord lenght, pitch angle(counterclockwise positive) and camber line. Blue line illustrates the chord line where as red line is the camber line. Angles are exaggerated for clarity purposes

Third factor, the camber of the blade was altered by adding a smaller section at a trailing edge of the blade using tape. Such alteration is aimed to change the camber line of the blades, which is as shown in figure 3, along with varying other parameters. This factor is inspired by the flaps in an aircraft, which are high lift devices,increasing the lift and drag but reducing the stall speed, by smoothing the boundary layer separation [Skybrary, 2017]. While not conventional in wind turbines, the factor still varies the camber of the blade, where many aerofoils asymmetric. Note that the size of the flap section was minimised to only investigate the effect of camber as a larger flap section would mean changing the chord length and the wing area as well. Two levels were selected for this factor as a negative flap angle would be fitting the linear relationship between the flap position, therefore making the measurement redundant. These parameters resulted in $2^2 * 3^1$ factorial design of experiment, with 12 observation for each experiment and 3 replicates.

Following the experimentation phase, the trials were investigated using ANOVA (Analysis of variance) method which is as summarised in table 2 with the Source of Variance table. As seen, all the factors and their interactions have provided a p-ratio which fits the 5th percentile of the f-ratio, which aligns with the initial research. While the experimental had many unquantifiable errors, such as vibrations induced at high speed causing misalignment, inaccurate weight distribution due to tape used to position the blades, high f ratio values and acceptable p-ratios illustrate a high explained variance in the output, which validates the experiment setup. The complete tables are as can be found in Appendix B. The logical following step would have been to fit model to the output data at different parameters and conduct an sensitivity analysis for investigating the effect of each parameter.

	Source of Variance Table							
	Source	Sum of Squares Calculation	Sum of Squares	Degrees of Freedom Calculation	Degrees of Freedoms	Mean Squares	F ratio	P Value
	[Chord Length] [A]	[A]-[T]	0.58	a-1	1	0.580136111	6.199323221	0.0201
	[Pitch Angle] [B]	[B]-[T]	4.78	b-1	2	2.387952778	25.51761703	< .00001
	[Camber] [C}	[C]-[T]	0.765625	c-1	1	0.765625	8.181453887	.001953
A x B	[Chord Length x Pitch Angle]	[AB]-[A]-[B]+[T]	0.76	(a-1)(b-1)	2	0.381202778	4.073525483	0.02998
A x C	[Chord Length x Camber]	[AC]-[A]-[C]+[T]	0.49	(a-1)(c-1)	1	0.492336111	5.261094126	0.03087
B x C	[Pitch Angle x Camber]	[BC]-[B]-[C]+[T]	101.06	(a-1)(b-1)(c-1)	2	50.53184167	539.9822791	< .00001
A x B x C	[Chord Length x Pitch Angle x Camber]	[ABC]-[AB]-[AC]-[BC]+[A]+[B]+[C]-[T]	314.5050111	(a-1)(b-1)(c-1)	2	157.2525056	1680.397222	< .00001
Within Y	Within Y	[Y]-[ABC]	2.245933333	abc(n-1)	24	0.093580556		
Total, T	Total, T	[Y]-[T]	9.798763889	abcn -1	35	0.279964683		

Table 2: Source of Variance table

3 Assignment 1c - Sensitivity Analysis

3.1 Part 1

Sensitivity analysis is to obtain the sensitivity of various aspects of model with respect to changes in its' parameters. As with experimental inputs, it can be concerned with large scale data such as sensory input. This task involves manipulation of data, identifying the key design parameters that will influence the output, fitting a model and determining the magnitude and direction of each parameter.

The wind turbine dataset provides large data with many parameters. After importing the data, some initial inputs were to be selected. While some inputs, such as wind speed and pitch angle, were selected based on previous literature review, many of the parameters were not as well-known. As for the first iteration, the parameters were selected based on intuition. For all calculations, the output parameter was chosen to be the Active Power,symbolised with P.

However,not all the information within the data set are relevant and therefore needs to be pre-processed before any sensitivity analysis to avoid fitting a wrong model. This "data cleaning process" (the cleaner script, can be found in Appendix C) included removing all the "Not a Number" (NaN) values, zero and minus inputs, missing data and any outliers using a MATLAB function which utilises Median Absolute Deviation method to detect outliers [MATLAB, 2017]. Another consideration was removing all the values where the grid voltage (Nu) would be zero as that would mean the wind turbine is simply not connected or turned on. Lastly, a live script ((the liveclean script, can be found in Appendix D)) from the Design Optimization toolbox was used to remove any remaining outliers which was validated visually by comparing scatter plots.

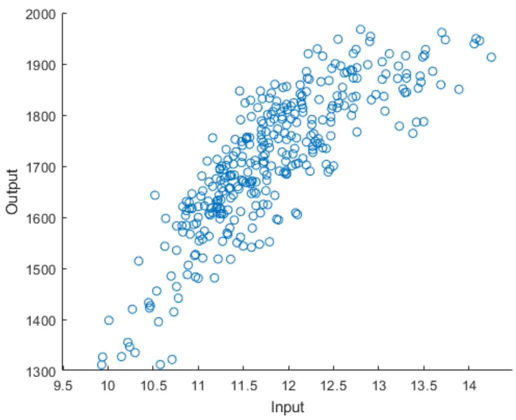

Figure 4: Scatter Plot of the Pitch Angle. Notice the positive coefficient between Inputs and outputs.

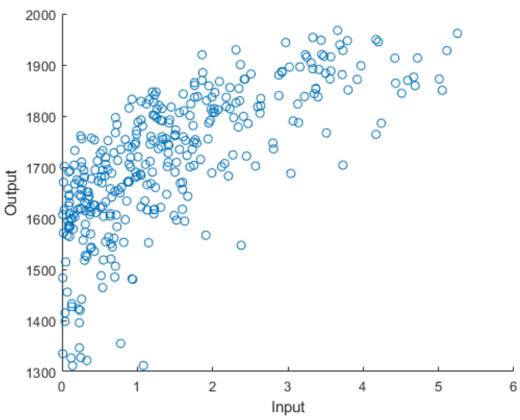

Figure 5: Scatter Plot of the Pitch Angle. Notice the positive coefficient between Inputs and outputs.

Following the pre-processing phase, an initial investigation was undertaken to determine effect of the parameters chosen to the output. This was achieved by creating scatter plot and calculating the Pearson Coefficient (pearsoncoef script, can be found in Appendix E)) . Initial estimates illustrated the Wind Speed and Pitch Angle, shown in figures 4 and 5, showed strong correlation while the effect of other input parameters seemed less. Therefore, less affecting parameters were changed with other available inputs until a reasonable correlation was achieved for each input, which are;

- Pitch Angle, symbolised as Ba, with a correlation factor of 0.7209
- Gearbox inlet temperature, symbolised as Ws, with a correlation factor 0.0617
- Wind Speed, symbolised as Ws, with a correlation factor 0.8322
- Outside temperature, symbolised as Ot , with a correlation factor -0.136
- Corrected Wind Angle, symbolised as Wa_c, with a correlation factor 0.084

With a better understanding of each parameter, a linear regression model was fitted to the standardised data, which provided magnitude and direction information regarding each parameter with a coefficient (scaledlinearregression script, available in Appendix F). Coefficients of each parameter, and a representation of the model, with a first order fit are as shown in table 3 and figure 6 respectively. As some potential outlier data was present within the model, the model had to be validated by calculating the R^2 value, which was determined to be **0.72**. By rearranging the R^2 equation, it can be seen that 50 % of all the standard deviation can be explained [peopleDuke, 2017].Furthermore, the Coefficients of the polynomial fit show correlation with Pearson Coefficients in terms of direction. For these reasons, the model was accepted to be a good fit.

Parameter No	Parameter Name	Coefficients of polynomial fit
X1	Pitch Angle, Ba	0.10569
X2	Gearbox Intlet Temperature, Git	0.023719
X3	Wind Speed, Ws	0.88253
X4	Outside Temperature, Ot	-0.20255
X5	Wind Angle Corrected, WA_c	0.036436

Table 3: Coeffients of the Polynomial fit for each selected parameter

As the linear Regression is concerned with small variances in inputs, a better sensitivity analysis would be the global, variance based analysis. The process involves scaling all the input and output parameters of the model, followed by generating a set of Sobol numbers, which are used to vary a parameters using large number of values between 0 and 1. Following the calculation of the first order and total order indices for each parameter (detailed in variancebased, can be viewed in Appendix G, along with the results) were calculated. While singular results were achieved, these did not align with the findings from the Pearson Coefficients and the linear fit. This could be due to the limitations of the implementation of the variance based analysis or intrinsic to the model, where the sample size is limited for this application or potential outliers that distort the data.

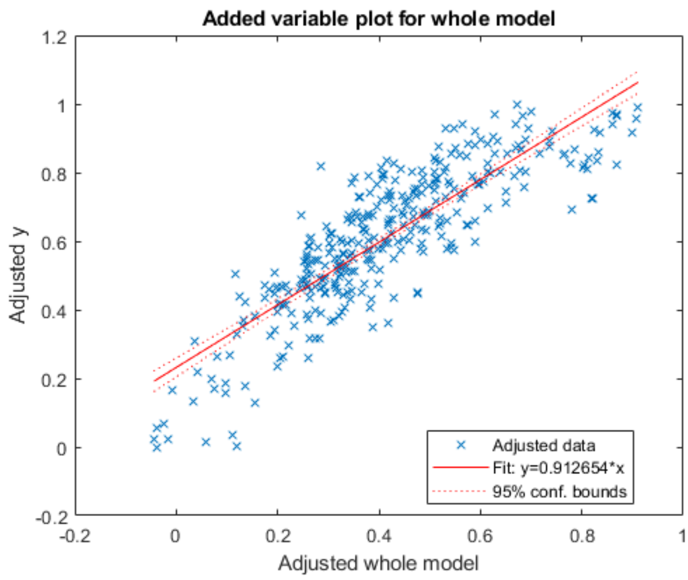

Figure 6: Linear fit of the model

3.2 Part 2

Second task involves utilising a "black box" approach to a given function and determining the interactions between its' parameters using the variance based sensitivity analysis. This was achieved by utilising Sobol numbers, which range between 0 and 1, as inputs, creating expressions for calculating and dividing it to total variance of the output for each parameter, which yields the following table 4. The script can be viewed in Appendix H.

	Employee ID = 1			
	Model 1		Model 2	
Parameter	Si	Sti	Si	Sti
A	0.2655	0.1014	0.2125	0.0105
B	0.133	0.0509	0.1062	0.0052
C	0.61	0.4027	0.075	0.0037
D	0.458	0.3021	0.0563	0.0027
E	0.3773	0.253	0.045	0.022

Table 4: Linear fit of the model

For Model 1, parameter C has the highest first order indices. This would mean that when varied in isolation, A would contribute the most significant variance in the output. This would be followed by parameters D,E,A,B respectively. As C,D,E all have high total order indices, this could mean they have second or higher level interactions among each other, is reflected in their total order indices. As for Model B,in terms of first order indices, A would lead to highest variance when varied in isolation. However, unlike the Model 1, the interactions between terms are minimal which suggest there are less higher order interactions between the parameters.

4 Assignment 1d - Linear Programming

Linear Programming is a method to optimise for the best solution to a mathematical problem that has linear constraints and a linear a objective function. This task involves optimising an investment based on constraints given which include the land mass and a limited investment. The process of linear programming included tabulating all the data, creating the formulae to calculate the profit for each location and resultant total profit. The limited land for locations A, B and C, and total investment being the constraints, Simplex Algorithm was implemented to achieve the optimum amount of turbines per location, which are as shown in table 5. Afterwards, the return on investment was calculated based on amount of turbines, their Expected Earning within 20 years, all their overhead costs and the initial investment divided among the locations, shown in table 6. The detailed spreadsheet can be found in Appendix I.

Employee ID=1	Turbine 1	Turbine 2	Turbine 3
Location A	1	11	0
Location B	1	66	0
Location C	0	0	3

Table 5: Linear fit of the model

Employee ID=1	Return on Investment
Location A	£39,458,666
Location B	£64,237,266
Location C	£88,662,166
Total	£192,358,097

Table 6: Linear fit of the model

5 References

References

K. M. Bower. https://asq.org/quality-resources/design-of-experiments, 2020.

J. M. Flynn. Assessment 1a. 2020.

Z. Mao. Influence analysis of blade chord length on the performance. https://aip.scitation.org/doi/full/10.1063/1.4943093, 2016. Last accessed 09 October 2020.

MATLAB. Outlier detection. https://www.mathworks.com/help/matlab/ref/isoutlier.html, 2017. Last accessed 09 October 2020.

K. Moore. Depth-first search (dfs). https://asq.org/quality-resources/design-of-experiments, 2016. Last accessed 09 October 2020.

Nrich. https://nrich.maths.org/604/solution, 2019.

peopleDuke. What's a good value for r-squared? https://people.duke.edu/ rnau/rsquared.htm, 2017. Last accessed 09 October 2020.

Skybrary. Flaps. https://www.skybrary.aero/index.php/Flaps, 2017. Last accessed 09 October 2020.

Smithsonian. Pitch angle and angle of attack. https://howthingsfly.si.edu/ask-an-explainer/when-are-angle-attack-and-pitch-angle-equal, 2016. Last accessed 09 October 2020.

6 Appendices

6.1 Appendix A

6.1.1 Analytical Optimisation Calculations

Volume of Box (mm^3)	384555119.4
Width of the Box (mm)	828.7
Depth of the Box (mm)	828.7
Height of the Box (mm)	560.0
Volume of Cylinder (mm^3) (check)	500000.0
Radius of Cylinder (mm) (r)	33.1
Height of Cylinder (mm)	145.3
No of circles in columns (m)	11.0
Number of columns(n)	14.3
Allowable Dimension accord dimensions (for m)	761.3
Allowable Dimensions accord dimensions (n)	827.5
Allowable Dimensions accord dimensions (p)	560.0
Amount of Cylinders in Stack (p)	3.9
Total Amount of Cylinders (m*n*p)	605.5
Total Volume of Cylinders (mm^3)	302751340.7
Density Ratio	0.8
Limit of Triangular Combination	0.9
Limit of Box	348752915.53
Optimal Radius of each can	**33.1**
Optimal Height of each can	**145.3**
Number of Cans that can be fitted within the Crate	**605**

Table 7: Analytical Calculations for Part1 a

6.2 Appendix B

6.2.1 Complete ANOVA Calculations

Trial 1	-1			0		
	25	35	45	25	35	45
-1	3.48	2.25	1.8	2.44	1.36	1.38
1	3.15	2.29	1.85	2.9	2.54	2.55

Trial 2	-1			0		
	25	35	45	25	35	45
-1	3.17	2.42	1.98	2.84	2.07	1.39
1	2.45	2.76	2.78	2.51	2.22	2.22

Trial 3	-1			0		
	25	35	45	25	35	45
-1	2.84	2.78	2.12	2.83	2.32	1.48
1	2.85	2.94	1.95	2.98	2.3	2.28

Anova Calculations	-1			0		
	25	35	45	25	35	45
	3.48	2.25	1.8	2.44	1.36	1.38
	3.17	2.42	1.98	2.84	2.07	1.39
-1	2.84	2.78	2.12	2.83	2.32	1.48
	3.15	2.29	1.85	2.9	2.54	2.55
	2.45	2.76	2.78	2.51	2.22	2.22
1	2.85	2.94	1.95	2.98	2.3	2.28

	-1							
	25		35		45			
	Mean	3.16	Mean	2.48	Mean	1.97		
	Sum	9.49	Sum	7.45	Sum	5.90	22.84	Sum of Sums
-1	Sum Squares	30.22	Sum Squares	18.65	Sum Squares	11.65		
	Mean	2.82	Mean	2.66	Mean	2.19		
	Sum	8.45	Sum	7.99	Sum	6.58	23.02	Sum of Sums
1	Sum Squares	24.05	Sum Squares	21.51	Sum Squares	14.95		
	Sum of Sums	17.94	Sum of Sums	15.44	Sum of Sums	12.48	45.86	Grand Sum

	0							
	25		35		45			
	Mean	2.70	Mean	1.92	Mean	1.42		
	Sum	8.11	Sum	5.75	Sum	4.25	18.11	Sum of Sums
-1	Sum Squares	22.03	Sum Squares	11.52	Sum Squares	6.03		
	Mean	2.80	Mean	2.35	Mean	2.35		
	Sum	8.39	Sum	7.06	Sum	7.05	22.50	Sum of Sums
1	Sum Squares	23.59	Sum Squares	16.67	Sum Squares	16.63		
	Sum of Sums	16.50	Sum of Sums	12.81	Sum of Sums	11.30	40.61	Grand Sum

Table 8: ANOVA Calculations - 1

Replicates	3
Levels in Chord Length	2
Levels in Pitch Angle	3
Levels in Camber	2

Main Effect Terms

[Chord Length]	-1	1	Row Totals	
-1	22.84	23.02	45.86	
0	18.11	22.50	40.61	
Column Totals	40.95	45.52	86.47	Grand Total
[A] [Chord Length] =	208.28			

[Pitch Angle]	25	35	45	Row Totals	
-1	17.94	15.44	12.48	45.86	
0	16.50	12.81	11.30	40.61	
Column Totals	34.44	28.25	23.78	86.47	Grand Total
[B] [Pitch Angle]=	212.47				

[Camber]	-1	1	Row Totals	
-1	22.84	23.02	45.86	
0	18.11	22.50	40.61	
Column Totals	40.95	45.52	86.47	Grand Total
[C] [Camber]=	208.2762722			

Two-Way Interaction Terms

[Chord Length x Pitch Angle]	25	35	45	Row Totals	
-1	17.60	13.20	10.15	40.95	
1	16.84	15.05	13.63	45.52	
Column Total	34.44	28.25	23.78	86.47	Grand Total
[AB] [Chord Length x Pitch Angle]=	213.8146				

[Chord Length x Camber]	
[AC] [Chord Length x Camber]=	209.5342

[Chord Length x Camber]	
[BC] [Camber x Pitch Angle]=	314.3014

Three-Way Interaction Terms

[Chord Length x Pitch Angle x Camber]		Pitch Angle		
Camber	Chord Length	25	35	45
-1	-1	9.49	7.45	5.90
	1	8.45	7.99	6.58
0	-1	8.11	5.75	4.25
	1	8.39	7.06	7.05
[ABC] [Chord Length x Pitch Angle x Camber]=	215.2489667			

Total Terms	
[T]	
[T]=	207.6961

Within Group Term	
[Y]	
[Y]=	217.4949

Source of Variance Table

	Source	Sum of Squares Calculation	Sum of Squares	Degrees of Freedom Calculation	Degrees of Freedoms	Mean Squares	F ratio	P Value
	[Chord Length] [A]	[A]-[T]	0.58	a-1	1	0.580136111	6.199323221	0.0201
	[Pitch Angle] [B]	[B]-[T]	4.78	b-1	2	2.387952778	25.51761703	< .00001
	[Camber] [C}	[C]-[T]	0.580136111	c-1	1	0.580136111	6.199323221	0.0201
A x B	[Chord Length x Pitch Angle]	[AB]-[A]-[B]+[T]	0.76	(a-1)(b-1)	2	0.381202778	4.073525483	0.02998
A x C	[Chord Length x Camber]	[AC]-[A]-[C]+[T]	0.68	(a-1)(c-1)	1	0.677825	7.243224791	0.01276
B x C	[Pitch Angle x Camber]	[BC]-[B]-[C]+[T]	101.25	(a-1)(b-1)(c-1)	2	50.62458611	540.9733444	< .00001
A x B x C	[Chord Length x Pitch Angle x Camber]	[ABC]-[AB]-[AC]-[BC]+[A]+[B]+[C]-[T]	314.3195222	(a-1)(b-1)(c-1)	2	157.1597611	1679.406156	< .00001
Within Y	Within Y	[Y]-[ABC]	2.245933333	abc(n-1)	24	0.093580556		
Total, T	Total, T	[Y]-[T]	9.798763889	abcn -1	35	0.279964683		

Table 9: ANOVA Calculations - 2

6.3 Appendix C

6.3.1 Cleaner Script

```
clear all
%loaddata
load lahautebornedata20172020.mat
%Assign Rawdata as inputs, output and reference(in case of Nu)
rawdata = lahautebornedata20172020;
inputs = [rawdata.Ba_avg, rawdata.Git_avg, rawdata.Ws_avg, rawdata.
    Ot_avg, rawdata.Wa_c_avg, rawdata.P_avg, rawdata.Nu_avg];
%inputs = [rawdata.Ba_avg, rawdata.DCs_avg, rawdata.Ws_avg, rawdata.
    Git_avg, rawdata.Dst_avg, rawdata.Rm_avg, rawdata.Nu_avg];
%remove nonvals
nanval = isnan(inputs);
nanvalreduced = any(nanval(),2);
inputs(nanvalreduced == 1, :) =[];

%remove zeros
inputs2 = inputs;
zeros = any(inputs2,3) <= 0;
zerosreduced = any(zeros(),2);
inputs2(zerosreduced == 1,:) = [];

%remove minuses
inputs3 = inputs2;
minuses = inputs3 <0;
minusesreduced = any(minuses(),2);
inputs3(minusesreduced == 1,:) = [];

%remove missings
inputs4 = inputs3;
missings = ismissing(inputs4);
missingsreduced = any(missings(),2);
inputs4(missingsreduced == 1, :) = [];


%remove outliers
inputs5 = inputs4;
outliers = isoutlier(inputs5);
outliersreduced = any(outliers(),2);
inputs5(outliersreduced == 1, :) = [];
finalinputs = inputs5;

%Now use live editor script for further clearing outliers
```

6.4 Appendix D

6.4.1 Liveclean Script

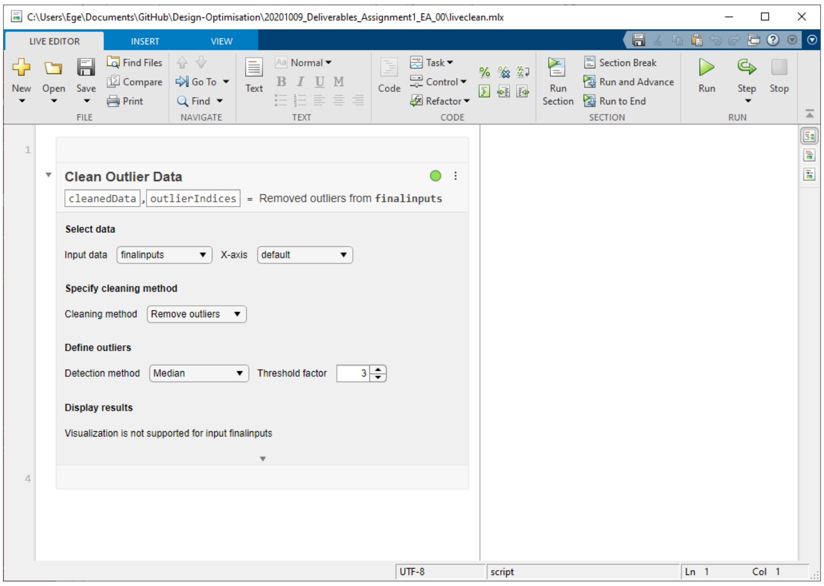

Figure 7: liveclean.mlx. function. To utilise the function, simply run the script with the shown parameter

6.5 Appendix E

6.5.1 Pearsoncoef Script

```
%Calculate Pearson Coefficiencts for cleaned data
pearsoncoeff1 = corrcoef(cleanedData(:,1),cleanedData(:,6))
pearsoncoeff2 = corrcoef(cleanedData(:,2),cleanedData(:,6))
pearsoncoeff3 = corrcoef(cleanedData(:,3),cleanedData(:,6))
pearsoncoeff4 = corrcoef(cleanedData(:,4),cleanedData(:,6))
pearsoncoeff5 = corrcoef(cleanedData(:,5),cleanedData(:,6))

%Create plots for each dataset
figure(1)
scatter(cleanedData(:,1),cleanedData(:,6))
figure(2)
scatter(cleanedData(:,2),cleanedData(:,6))
figure(3)
scatter(cleanedData(:,3),cleanedData(:,6))
figure(4)
scatter(cleanedData(:,4),cleanedData(:,6))
figure(5)
scatter(cleanedData(:,5),cleanedData(:,6))
```

6.6 Appendix F

6.6.1 Scaledregression Script

```

%Rearrange the inputs and outputs
cleanedinputs = [cleanedData(:,1),cleanedData(:,2),cleanedData(:,3),cleanedData(:,4),cleanedData(:,5)];
cleanedoutputs = cleanedData(:,6);

%standardisation of both inputs and outputs
for i = 1: size(cleanedinputs,2)
standardisedinputs(:,i) = (cleanedinputs(:,i)-mean(cleanedinputs(:,i)))/(std(cleanedinputs(:,i)));
end

for i = 1: size(cleanedoutputs,2)
standardisedoutputs(:,i) = (cleanedoutputs(:,i)-mean(cleanedoutputs(:,i)))/(std(cleanedoutputs(:,i)));
end

%scaling both inputs and outputs for the variance based analysis
scaledstandardisedinputs = rescale(standardisedinputs);
scaledstandardisedoutputs = rescale(standardisedoutputs);


%coefficient of determination

mdl = fitlm(scaledstandardisedinputs,scaledstandardisedoutputs)
beta = mdl.Coefficients(2:end,1)
beta = table2array(beta)

% A = rank(scaledstandardisedinputs);
% invinputs = pinv((scaledstandardisedinputs));
% beta = invinputs * scaledstandardisedoutputs;

plot(mdl)
```

6.7 Appendix G

6.7.1 Variancebased Script

```
%Size of the Input Matrix
sizek  = size(cleanedinputs);
N = sizek(1);
k = sizek(2);
% %Firstly create an object of Sobol pseudorandom numbers with twice
% as many columns as there are input factors to your model (i.e.
% 2*k)
Quas_Rand = sobolset(2*k);
% % Use the set to create matrices of pseudorandom numbers ignoring
% the first row (please do this). N is the number of required
% samples and k

A = Quas_Rand(2:N+1, 1:k);
B = Quas_Rand(2:N+1, k+1:end);

for i =1:size(A,2)
    At=A(:,1:i-1);
    Bt=B(:,i);
    Ct=A(:,i+1:end);

    A_B = [At, Bt, Ct];
    A_Bdata(i) = {A_B};
end

%                      Estimate        SE         tStat       pValue
%                     _________    ________    _______    __________
%
%    (Intercept)        0.23786    0.032778     7.2567    2.8102e-12
%    x1                 0.12228    0.066576     1.8367      0.067148
%    x2               0.0067747    0.042558    0.15919       0.87362
%    x3                   0.856    0.064613     13.248    1.8072e-32
%    x4                -0.19799    0.042988    -4.6056    5.8613e-06
%    x5                0.030022    0.037071    0.80985       0.41861

%Create Outputs for Generated Sobol Numbers

for j =2:N
    for i =1:k

Aeval(j,i) = (beta(1)*A(j,i)+ beta(2)*A(j,i) +beta(3)*A(j,i)+ beta(4)*A(j,i)+ beta(5)*A(j,i));
Beval(j,i) = (beta(1)*B(j,i)+ beta(2)*B(j,i) + beta(3)*B(j,i) + beta(4)*B(j,i)+ beta(5)*B(j,i));
A_Beval(j,i) = beta(1)*A_Bdata{i}(j,i)+ beta(2)*A_Bdata{i}(j,i) + beta(3)*A_Bdata{i}(j,i) + beta(4)*A_Bdata{i}(j,i)+ beta(5)*A_Bdata{i}(j,i);
```

```
%Simplified Fit
% Aeval(j,i) = 0.887599* A(j,i);
% Beval(j,i) = 0.887599* B(j,i);
% A_Beval(j,i) = 0.887599* A_B(j,i);

      end
end

%Calculate Numerator terms

for j =2:N
      for i =1:k
Vxi(j,i) = (1/N) * (Beval(j,i))*((A_Beval(j,i)) - (Aeval(j,i)));
      end
end

for j =2:N
      for i =1:k
Exi(j,i) = (1/(2*N)) * ((Aeval(j,i)) -(A_Beval(j,i)))^2;
      end
end

%Get varinace of total outputs
varoutputs = var(Vxi) + var(Exi);

%Get First Order Effects and Total Order Effects
Si = mean(Vxi,1) ./ varoutputs;
Sti = mean(Exi) ./ varoutputs;

%Calculate only one value for Si and Sti
Sifinal(:,1)= mean(Si(:,1));
Sifinal(:,2)= mean(Si(:,2));
Sifinal(:,3)= mean(Si(:,3));
Sifinal(:,4)= mean(Si(:,4));
Sifinal(:,5)= mean(Si(:,5))

Stifinal(:,1)= mean(Sti(:,1));
Stifinal(:,2)= mean(Sti(:,2));
Stifinal(:,3)= mean(Sti(:,3));
Stifinal(:,4)= mean(Sti(:,4));
Stifinal(:,5)= mean(Sti(:,5))
```

6.8 Appendix H

6.8.1 Pearsoncoef Script

```
clear all
%Size of the supposed matrix
sizek = [500 5];
N = sizek(1); % Sample Size
k = sizek(2); %Number of parameters

%calculating sobol numbers

Pseudo_Rand = sobolset(2*k);
A = Pseudo_Rand(2:N+1, 1:k);
B = Pseudo_Rand(2:N+1, k+1:end);
ABi = cell(1,k);

for k = 1:k
    ABi{k} = A;
    ABi{k}(:,k) = B(:,k);
end
% A =mat2cell(A,(size(A,1)),size(A,2))
% B =mat2cell(B,(size(B,1)),size(B,2))
% A_B = repmat(A, [k,1]);
% for i = 1:k
%     A_B((i-1)*N + 1: i*N, i) = B(:,i);
% end

%calculating outputs sobol sets
for k = 1:k
    for N =1:N

        Aout{k}(N) = TurbineModel_2020(A(N,:),'2',1);
        Bout{k}(N) = TurbineModel_2020(B(N,:),'2',1);
        A_Biout{k}(N) = TurbineModel_2020(ABi{k}(N,:),'2',1);

        %First order Indices

        Vxi{k}(N) = (1/N) * Bout{k}(N) * (A_Biout{k}(N) - Aout{k}(N
           ))^2;

        %Total Order Indices

        Exi{k}(N) = (1/2*N) * ( Aout{k}(N) - A_Biout{k}(N) )^2;




    end

        totalvar(k) = var(Vxi{k}) + var(Exi{k});
```

```
        Si{k} = Vxi{k}/ totalvar(k);
        Sifinal(k) = mean(cell2mat(Si));

        Sti{k} = Exi{k}/totalvar(k);
        Stifinal(k) = mean(cell2mat(Sti));
end

%Variance Set

disp(Sifinal);
disp(Stifinal);
```

6.9 Appendix I

6.9.1 Linear Programming Script

	Wind Turbine 1	Wind Turbine 2	Wind Turbine 3
Plot length per turbine (in a line), m	100	150	250
Life Expectancy	20	20	20
Annual Earnings for each turbine, location A	£138,300	£318,300	£320,400
Annual Earnings for each turbine, location B	£243,570	£421,360	£45,494
Annual Earnings for each turbine, location C	£248,875	£531,900	£1,488,000
Expected Earnings for each turbine, location A	£2,766,000	£6,366,000	£6,408,000
Expected Earnings for each turbine, location B	£4,871,400	£8,427,200	£909,880
Expected Earnings for each turbine, location C	£4,977,500	£10,638,000	£29,760,000
Purchase and installation cost of each turbine, location A	£75,030	£310,860	£3,379,200
Purchase and installation cost of each turbine, location B	£2,710,800	£3,603,800	£4,493,300
Purchase and installation cost of each turbine, location C	£3,766,600	£4,459,000	£6,619,800
Land cost for each turbine Location A	£191,400	£756,750	£1,193,750
Land cost for each turbine Location B	£310,200	£1,058,625	£1,800,000
Land cost for each turbine Location C	£486,900	£5,115,600	£3,063,200

Resources Used
1750
10000
750

	Plot Length
Location A	5000
Location B	10000
Location C	750
	Total Investment
	£100,000,000

Employee ID=1	Turbine 1	Turbine 2	Turbine 3				
Location A	1	11	0	Total Profit Location A	3639600	Cost of Investment Location A	£3,494,490.00
Location B	1	66	0	Total Profit Location B	49199430	Cost of Investment Location B	£42,352,600.00
Location C	0	0	3	Total Profit Location C	4464000	Cost of Investment Location C	£52,815,600.00
				Total Profit	£57,303,030.00	Total Investment	£98,662,690.00

Employee ID=1	Return on Investment
Location A	£39,458,666
Location B	£64,237,266
Location C	£88,662,166
Total	£192,358,097

Figure 8: Linear Programming Calculations for Assignment 1d