NATIONAL INSTITUTE OF TECHNOLOGY, DURGAPUR

DEPARTMENT OF CHEMICAL ENGINEERING

- **Project Title- To maximize the average reaction rate of a photocatalytic reactor by using Machine Learning Techniques in MATLAB (GP and GA)**

SUBMITTED BY:

NAME- Saurabh Raj

ROLL NO. - 21CH8066

SEMESTER- 6th Semester (3rd year)

SESSION- 2023-24

SUBMITTED TO:

Prof. Sandip Kumar Lahiri

Associate Professor

Department of Chemical Eng

NIT DURGAPUR

ACKNOWLEDGEMENT

I wish to express my profound gratitude to Professor Sandip Kumar Lahiri from the Chemical Engineering Department at NIT DURGAPUR for granting me the invaluable opportunity to pursue this project under his guidance. His expertise, mentorship, and steadfast support have been pivotal in the successful completion of this endeavor.

"Enhancing the average reaction rate of a photocatalytic reactor through Machine Learning Techniques in MATLAB (GP and GA)."

Lastly, I am indebted for their unwavering encouragement and support throughout this journey. Their unwavering belief in me has served as a constant source of motivation.

To each of you who have played a role in this academic endeavor, I extend my heartfelt gratitude. Thank you for being an integral part of this journey.

Saurabh Raj

B-TECH- 6th semester
NATIONAL INSTITUTE OF TECHNOLOGY, DURGAPUR

CONTENTS

1. **ABSTRACT…(4)**

2. **PROBLEM STATEMENT…(5-6)**

3. **SOLUTION…(6-12)**

4. **RESULT AND DISCUSSIONS…(12-13)**

5. **DISCUSSIONS OF THE RESULTS…(13-14)**

6. **APPENDIX-1…(14-15)**

7. **APPENDIX-2…(16-18)**

ABSTRACT

Title: Maximizing Average Reaction Rate in a Photocatalytic Reactor: Optimization of Input Variables using Genetic Algorithms and Genetic Programming in MATLAB.

Abstract:

Photocatalytic reactors are vital in environmental cleanup and industry, using catalysts to hasten reactions under light. Optimizing parameters like catalyst load, lamp power, pH, and TOC concentration is crucial for achieving maximum reaction rates. This project aims to boost these rates in a photocatalytic reactor using genetic algorithms and programming in MATLAB. Through systematic experiments and data analysis, it seeks the best parameter combination for peak reaction rates. This optimization promises to enhance reactor efficiency, impacting environmental and industrial processes positively. The study's outcomes hold potential for tackling environmental issues and refining industrial operations, with profound societal and economic implications.

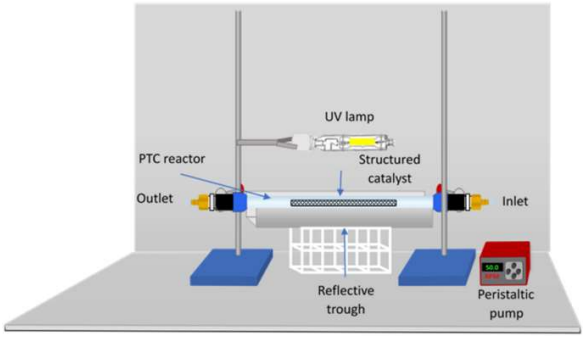
PROBLEM STATEMENT

PROBLEM:

We are provided with the data of operating conditions under which a photocatalytic reactor is operated at a plant. We can observe from the data (given in Appendix-I) that there are variations in the output of the average reaction rate at different input parameters [i.e., catalyst loading , lamp power, pH and total organic carbon (TOC)]. We can see that at some input concentrations of reactants, the yield is good but at some other concentrations, yield is too low.

EXPLANATION:

The function of a photocatalytic reactor is to facilitate chemical reactions, typically involving the degradation of organic pollutants or the conversion of substances, by utilizing photocatalysts activated under light irradiation. In essence, the reactor harnesses the energy of light to accelerate the desired chemical processes, such as the breakdown of pollutants into harmless byproducts or the synthesis of valuable compounds.



Here we have the operating data for the photocatalytic reactor . Using this data, we will make a model using Genetic Programming in MATLAB for the x1, x2, x3, x4 and y dataset and then use Genetic Algorithm method to maximize y, which is the average reaction rate for the photocatalytic reactor.

OBJECTIVE:

The main objective of this project is to maximize the value of average reaction rate of photocatalytic reactor by training a Machine Learning model using the given datasets of x1, x2, x3, x4 and y using MATLAB by applying Genetic Programming and Genetic Algorithm methodologies. We need to find the optimum value of each input parameters for which the combined output average reaction rate is maximum.

SOLUTION

Variables in Data given	Type of data	Range	Unit
X_1	Catalyst Loading	1.0-4.5	kg
X_2	Lamp Power	200-400	watt
X_3	pH	8-12	-
X_4	Total organic carbon (TOC)	40-120	Kg/l
y	Average Reaction Rate	0.021889392-0.085326893	Mol / l sec

Steps for modelling:-

Step 1: Data Cleaning:-

First, we need to remove the outlying data from the given data (in Appendix-I below). Outlying data are generated when there is some operational error or error in recording data from the system. These may result to bad model training or error in final results. First, we need to find our lower and upper bounds for each data set of x_1, x_2, x_3, x4 and y in excel.

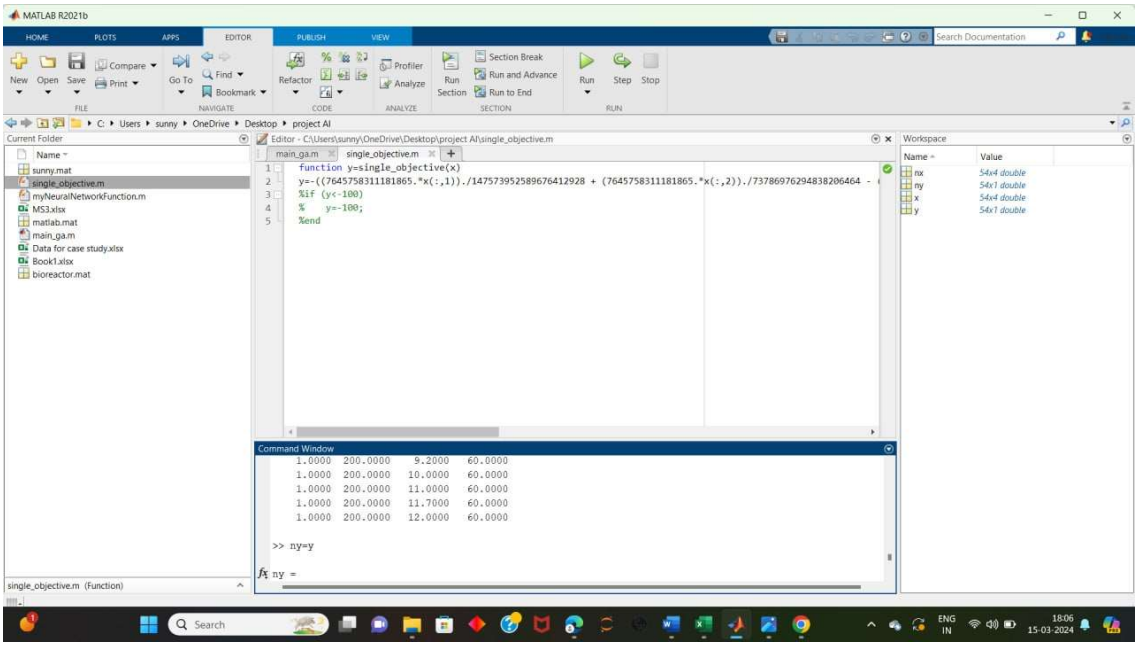
LB = Mean-3*Standard deviation

UB = Mean+3*Standard deviation

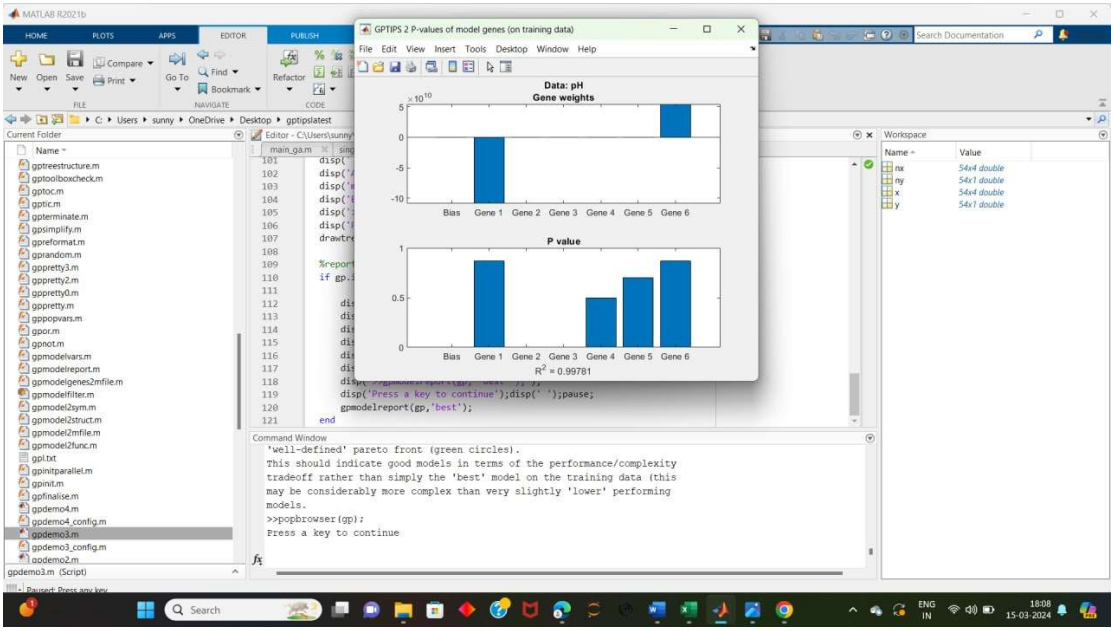
Then we need to delete those data which are less than their respective lower bound (LB) and higher than their respective higher bound (UB). We can do this by using ‘Conditional formatting’ option in Excel.

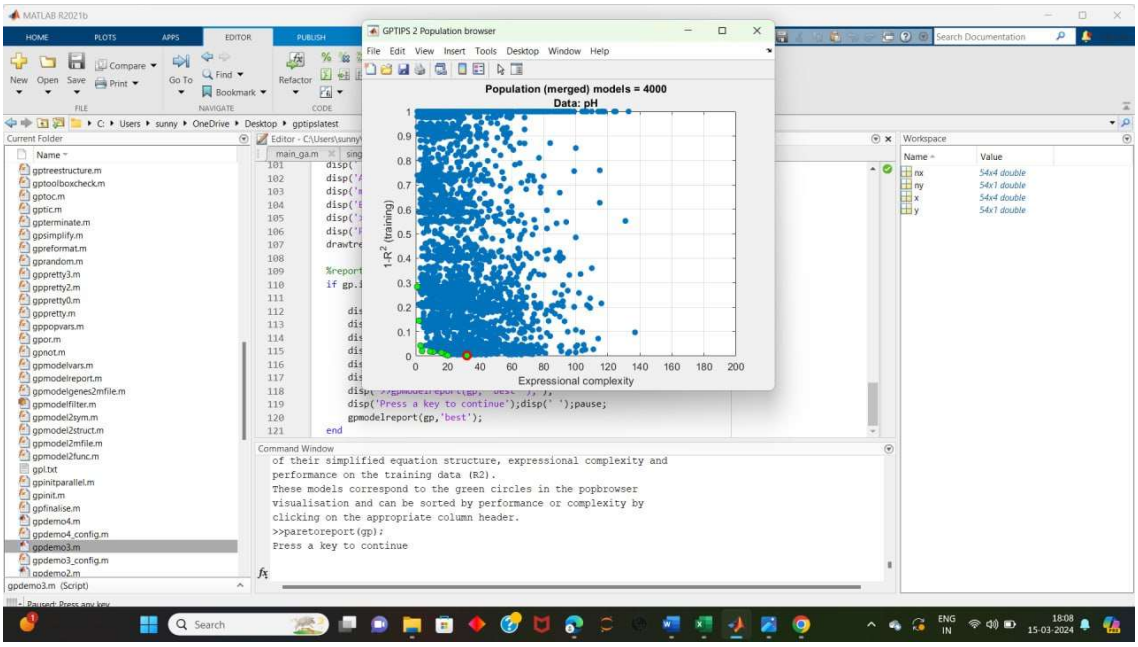
Step 2 : Model building by using MATLAB by Genetic Algorithms :-

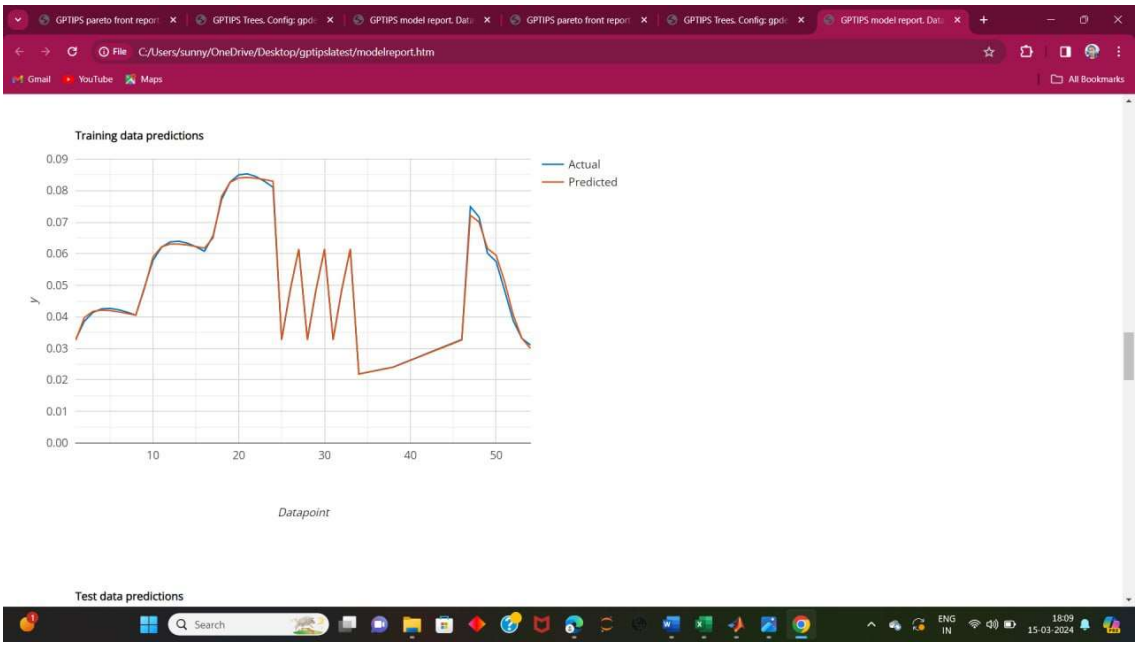
After cleaning the data, it’s time for GA model building. First, we select all x_1, x_2, x_3, x_4 values together and import it to MATLAB as x and all y values imported as y. The variables are created in the workspace. Now nx=x and ny=y is also created in the workspace and the workspace is saved with the file name as mydata.mat

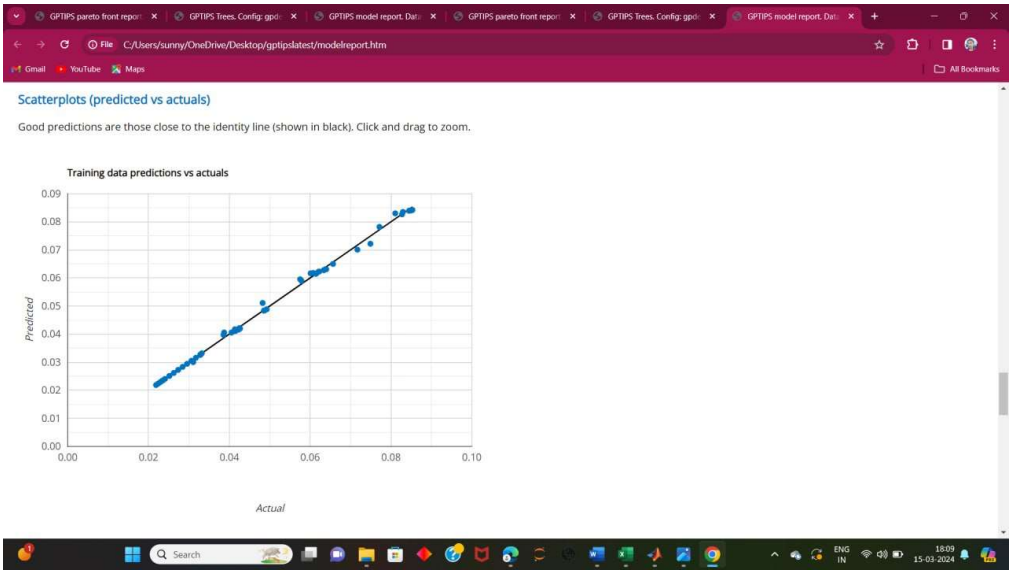


Now the gpdemo3.m and gpdemo3_config.m is opened in the editor window of the MATLAB and the gpdemo3.m has to be run. Various plots are opening in the MATLAB windows shows different performance chart which are attached below.

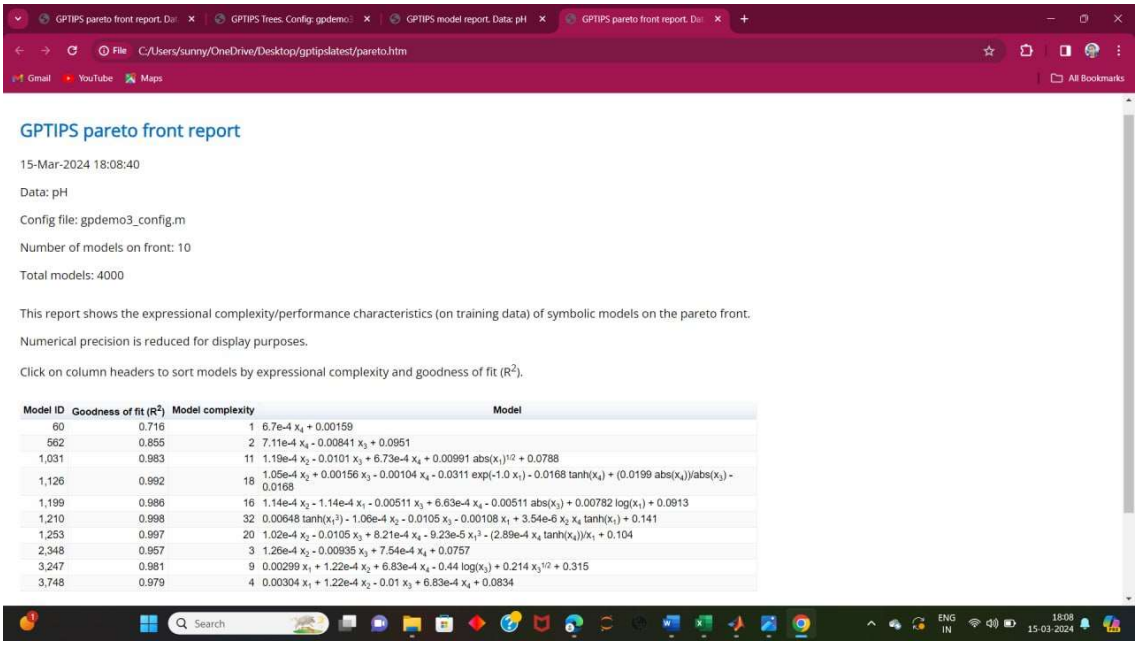

GPTIPS 2 P-values of model genes (on training data)
Data: pH
Gene weights
Bias Gene 1 Gene 2 Gene 3 Gene 4 Gene 5 Gene 6
P value
Bias Gene 1 Gene 2 Gene 3 Gene 4 Gene 5 Gene 6
R2 = 0.99781
'well-defined' pareto front (green circles).
This should indicate good models in terms of the performance/complexity
tradeoff rather than simply the 'best' model on the training data (this
may be considerably more complex than very slightly 'lower' performing
models.
>>popbrowser(gp);
Press a key to continue


GPTIPS 2 Population browser
Population (merged) models = 4000
Data: pH
1-R2 (training)
Expressional complexity
of their simplified equation structure, expressional complexity and
performance on the training data (R2).
These models correspond to the green circles in the popbrowser
visualisation and can be sorted by performance or complexity by
clicking on the appropriate column header.
>>paretoreport(gp);
Press a key to continue


Training data predictions
Actual
Predicted
y
Datapoint
Test data predictions


Scatterplots (predicted vs actuals)
Good predictions are those close to the identity line (shown in black). Click and drag to zoom.
Training data predictions vs actuals
Predicted
Actual

After complete execution of the GP program , a pareto front report is opening in the browser shows all the model its complexity and R^2 which was trained in the execution.



GPTIPS pareto front report

15-Mar-2024 18:08:40

Data: pH

Config file: gpdemo3_config.m

Number of models on front: 10

Total models: 4000

This report shows the expressional complexity/performance characteristics (on training data) of symbolic models on the pareto front.

Numerical precision is reduced for display purposes.

Click on column headers to sort models by expressional complexity and goodness of fit (R^2).

Model ID	Goodness of fit (R^2)	Model complexity	Model
60	0.716	1	6.7e-4 x_4 + 0.00159
562	0.855	2	7.11e-4 x_4 - 0.00841 x_3 + 0.0951
1,031	0.983	11	1.19e-4 x_2 - 0.0101 x_3 + 6.73e-4 x_4 + 0.00991 abs(x_1)^1/2 + 0.0788
1,126	0.992	18	1.05e-4 x_2 + 0.00156 x_3 - 0.00104 x_4 - 0.0311 exp(-1.0 x_1) - 0.0168 tanh(x_4) + (0.0199 abs(x_4))/abs(x_3) - 0.0168
1,199	0.986	16	1.14e-4 x_2 - 1.14e-4 x_1 - 0.00511 x_3 + 6.63e-4 x_4 - 0.00511 abs(x_3) + 0.00782 log(x_1) + 0.0913
1,210	0.998	32	0.00648 tanh(x_1^3) - 1.06e-4 x_2 - 0.0105 x_3 - 0.00108 x_1 + 3.54e-6 x_2 x_4 tanh(x_1) + 0.141
1,253	0.997	20	1.02e-4 x_2 - 0.0105 x_3 + 8.21e-4 x_4 - 9.23e-5 x_1^3 - (2.89e-4 x_4 tanh(x_4))/x_1 + 0.104
2,348	0.957	3	1.26e-4 x_2 - 0.00935 x_3 + 7.54e-4 x_4 + 0.0757
3,247	0.981	9	0.00299 x_1 + 1.22e-4 x_2 + 6.83e-4 x_4 - 0.44 log(x_3) + 0.214 $x_3^{1/2}$ + 0.315
3,748	0.979	4	0.00304 x_1 + 1.22e-4 x_2 - 0.01 x_3 + 6.83e-4 x_4 + 0.0834

The best model is selected whose complexity and R^2 is balanced in such a way to get the minimum deviation from the actual prediction. In this case, we selected the model ID 1253 whose R^2 is 0.997 and complexity is 20.

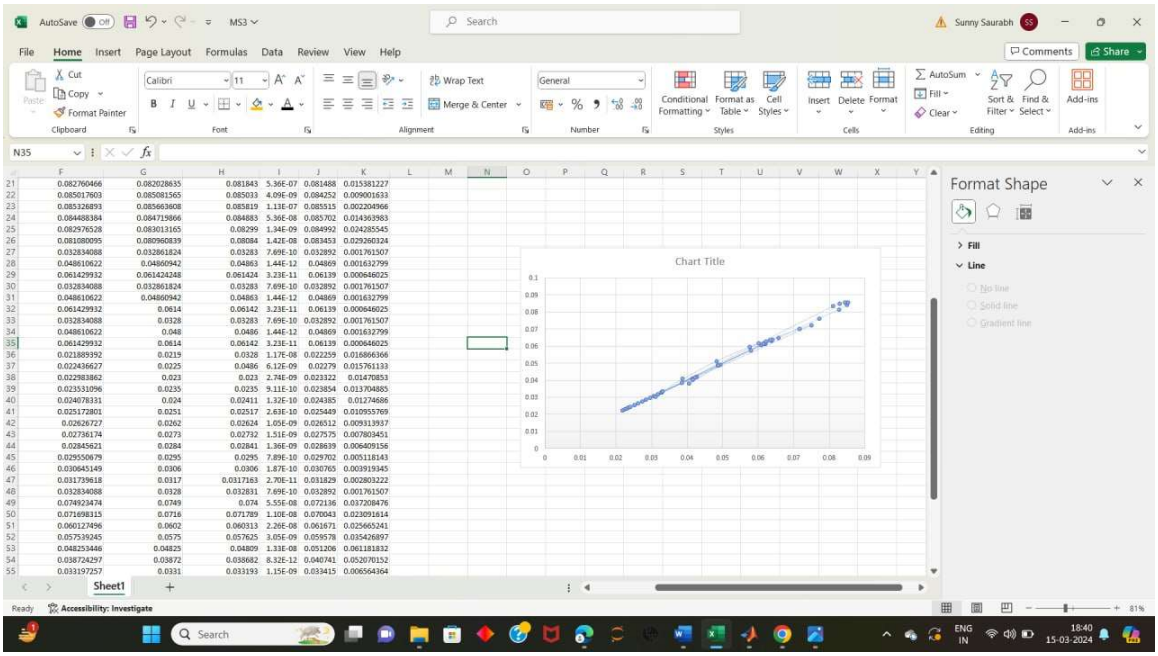
We can save this model by using gpmodel2mfile by writing this gpmodel2mfile(gp,1253,Saurabh) in command window and save it as Saurabh.m

Step-3 :- Trained model validation :-

Now we can find the predicted y from romi.m file. We can then copy this y predicted value from MATLAB to Excel and find the error between actual y value and value of y predicted by ANN and then its error%, RMSE and R^2 values. We can confirm the accuracy of our trained model using these data.

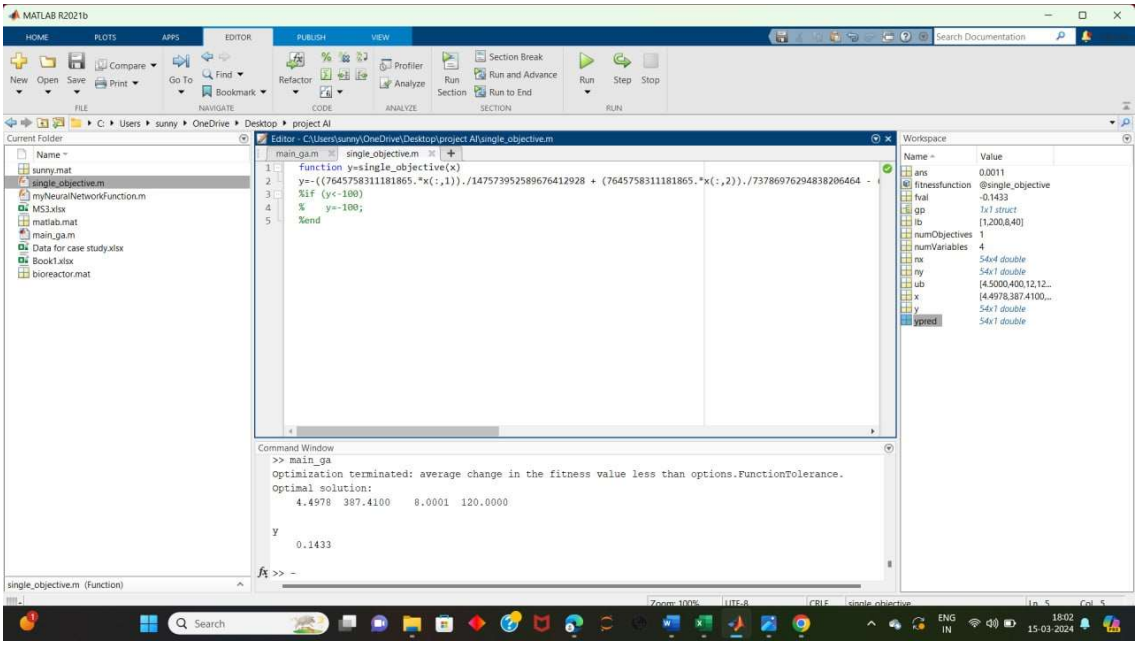
To find the predicted y, we input a command “ypred=Saurabh(x)” in the command window and hit enter. After that we saw that in workspace we get the ypred matrix.

Now we take take this predicted y i.e, ypred to the Excel and calculate the performance parameters i.e, error between actual y value and value of y predicted by ANN and then its error%, RMSE and R^2 values.



Step 4- Optimization:-

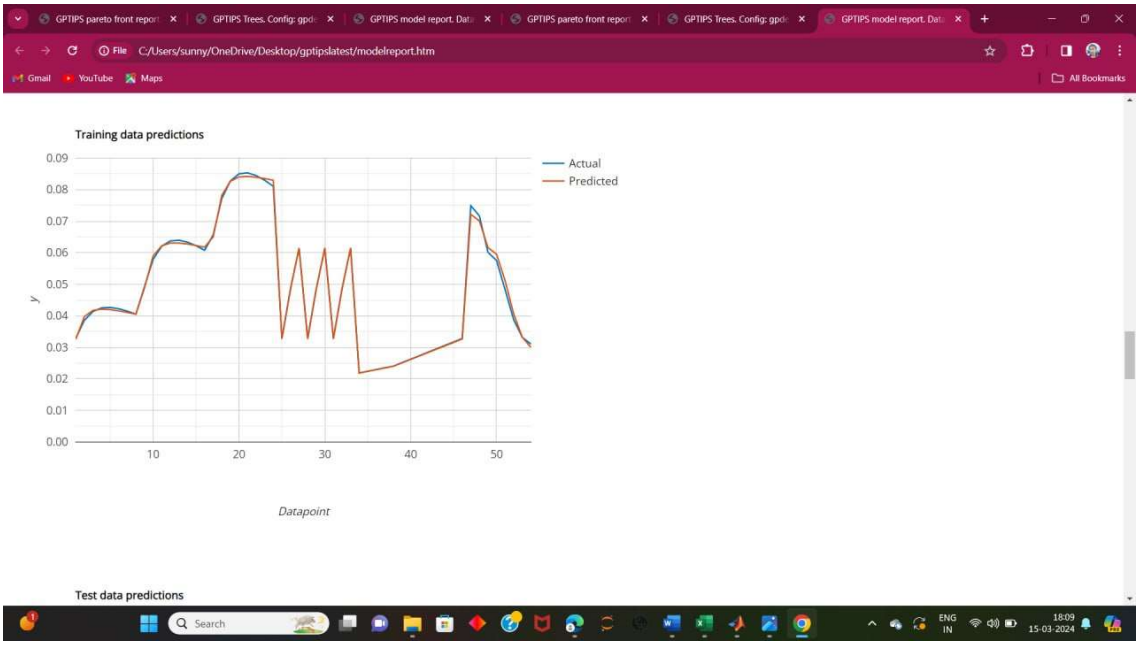
Now that we have obtained the model and its performance parameters, we have to find the values of x_1, x_2, x_3 and x_4 for which the value of y is maximum. This can be done by Genetic Algorithm in MATLAB. We need to input number of objectives and variables and ranges of all x data and run GA code for the trained model and GA will provide us the optimum data of x and y.



Output of the result we got is shown below in the command window:

RESULTS AND DISCUSSIONS

- Graph between Actual y and predicted y:



- Model performance parameters

Average error%	1.23
Mean square error (MSE)	$8.17X\ 10^{-7}$
Root mean square error (RMSE)	$9\ X\ 10^{-4}$
Coefficient of determination (R^2)	0.997

- Optimum values of x_1, x_2 and x_3 for which the value of y is maximum :-

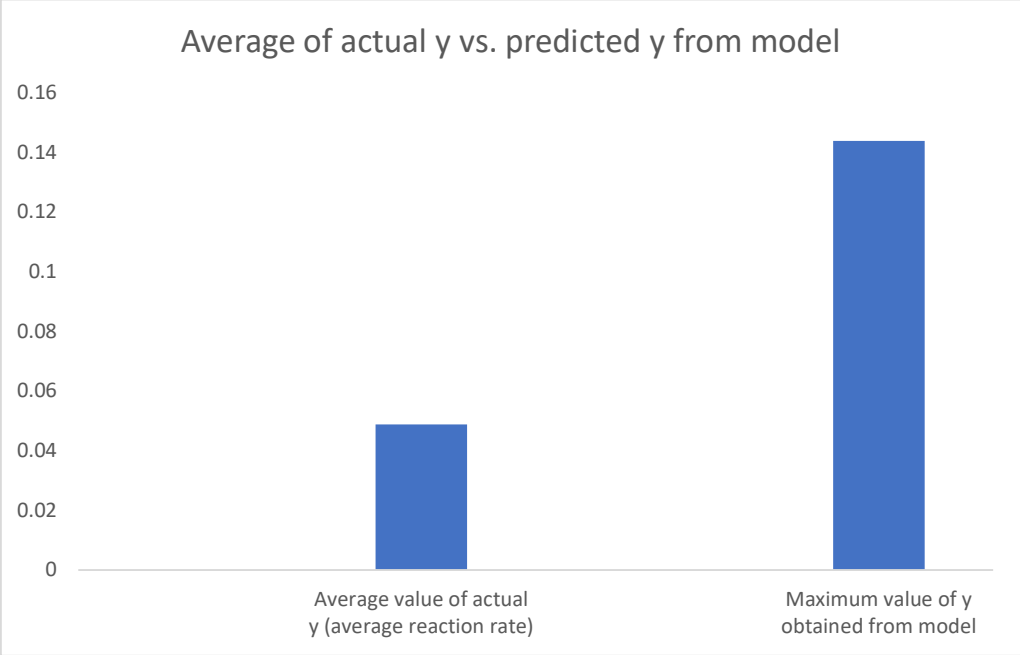
Optimum value of x_1 (catalyst loading , kg)	4.4978 g/L
Optimum value of x_2 (lamp power , watt)	387.41 watt
Optimum value of x_3 (pH)	.0001
Optimum value of x_4 (TOC)	120.00 kg/l
Optimum value of y (average reaction rate)	0.1433 mol/l sec

Model Obtained:- y= 1.02e(-4) x2 -0.0105 x3 + 8.21(e-4) x4 9.23e-5 (x1)^3 - (2.89e-4 x4 tanh(x4))/x1 +0.104

DISCUSSIONS OF THE RESULT

From the above result we found that the average value of actual y, maximum value of y obtained from the model, increase in value of y obtained and %increase in value of y after modelling and the bar chart between average y and maximum y from the model as shown below:

Average value of actual y (average reaction rate)	0.0489
Optimum value of y obtained from model	0.1455
Increase in value of y obtained	0.0966
%Increase in value of y after modelling	197.54%



APPENDIX – 1

Experimental Data of Photocatalytic Reactor:

Catalyst loading	Lamp Power	pH	TOC conc.	Average reaction rate
1	200	11.75	60	0.032834088
1.5	200	11.75	60	0.038579728
2	200	11.75	60	0.041380233
2.5	200	11.75	60	0.042508801
3	200	11.75	60	0.042663447
3.5	200	11.75	60	0.042244192
4	200	11.75	60	0.041488264
4.5	200	11.75	60	0.040540047
1	200	11.75	90	0.049251132
1.5	200	11.75	90	0.057869592
2	200	11.75	90	0.06207035
2.5	200	11.75	90	0.063763202
3	200	11.75	90	0.06399517
3.5	200	11.75	90	0.063366288
4	200	11.75	90	0.062232396
4.5	200	11.75	90	0.060810071
1	200	11.75	120	0.065668176
1.5	200	11.75	120	0.077159456

2	200	11.75	120	0.082760466
2.5	200	11.75	120	0.085017603
3	200	11.75	120	0.085326893
3.5	200	11.75	120	0.084488384
4	200	11.75	120	0.082976528
4.5	200	11.75	120	0.081080095
1	200	11.75	60	0.032834088
1.5	300	11.75	60	0.048610622
2	400	11.75	60	0.061429932
1	200	11.75	60	0.032834088
1.5	300	11.75	60	0.048610622
2	400	11.75	60	0.061429932
1	200	11.75	60	0.032834088
1.5	300	11.75	60	0.048610622
2	400	11.75	60	0.061429932
1	200	11.75	40	0.021889392
1	200	11.75	41	0.022436627
1	200	11.75	42	0.022983862
1	200	11.75	43	0.023531096
1	200	11.75	44	0.024078331
1	200	11.75	46	0.025172801
1	200	11.75	48	0.02626727
1	200	11.75	50	0.02736174
1	200	11.75	52	0.02845621
1	200	11.75	54	0.029550679
1	200	11.75	56	0.030645149
1	200	11.75	58	0.031739618
1	200	11.75	60	0.032834088
1	200	8	60	0.074923474
1	200	8.2	60	0.071698315
1	200	9	60	0.060127496
1	200	9.2	60	0.057539245
1	200	10	60	0.048253446
1	200	11	60	0.038724297
1	200	11.7	60	0.033197257
1	200	12	60	0.031076977

APPENDIX- 2

GP Function Code

```
% GPDEMO3 GPTIPS 2 demo of multigene symbolic regression on non-linear simulated
pH data.
%
%   Demonstrates multigene symbolic regression and some post run analysis
%   functions such as SUMMARY and RUNTREE and the use of the Symbolic Math
%   Toolbox to simplify expressions and create HTML reports using
%   PARETOREPORT, GPMODELREPORT and DRAWTREES to visualise the models.
%
%   (c) Dominic Searson 2009-2015
%
%   GPTIPS 2
%
%   See also GPDEMO3_CONFIG, GPDEMO1, GPDEMO2, GPDEMO4, PARETOREPORT,
%   GPMODELREPORT, DRAWTREES, SUMMARY, RUNTREE, GPPRETTY, POPBROWSER

clc;
disp('GPTIPS 2 Demo 3: multigene pareto symbolic regression on pH data');
disp('-----------------------------------------------------------------');
disp('In this example, the training data is 700 steady state data points');
disp('from a simulation of a pH neutralisation process.');
disp(' ' );
disp('Here we use use pareto tournaments to bias the model discovery process');
disp('towards low complexity models.');
disp(' ');
disp('GPTIPS is run 3 times for a maximum of 10 seconds per run or until a');
disp('RMSE of 0.2 is reached. The runs are merged into a single population');
disp('at the end.');
disp(' ');
disp('The output y has an unknown non-linear dependence on the 4 inputs x1,');
disp('x2, x3 and x4.');
disp(' ');
disp('300 data points are available as a test set to validate the evolved
model(s).');
disp('');
disp('The configuration file is gpdemo3_config.m and the raw data is in
ph2data.mat');
disp(' ');
disp('Here, 6 genes are used (plus a bias term) so the form of the model will
be');
disp('ypred = c0 + c1*tree1 + ... + c6*tree6');
disp('where ypred = predicted output, c0 = bias and c1,...,c6 are the gene
weights.')
disp(' ');
disp('Genes are limited to a depth of 4.');
disp(' ');
disp('The function nodes used are:  TIMES MINUS PLUS TANH MULT3 ADD3');
disp(' ');
disp('First, run GPTIPS using the configuration in gpdemo3_config.m :');
disp('>>gp=rungp(@gpdemo3_config);');
disp('Press a key to continue');
disp(' ');
```

```
pause;

%run GPTIPS using the configuration in gpdemo3_config.m
gp = rungp(@gpdemo3_config);

%run the best individual of the run on the fitness function
disp(' ');
disp('Evaluate the ''best'' individual of the run using:');
disp('>>runtree(gp,''best'');');

disp('Press a key to continue');disp(' ');pause;
runtree(gp,'best');

%run the best individual of the run on the fitness function
disp(' ');
disp('Next, display the population in terms of performance and complexity.');
disp('Because pareto tournaments have been enabled you should notice a');
disp('''well-defined'' pareto front (green circles).');
disp('This should indicate good models in terms of the performance/complexity');
disp('tradeoff rather than simply the ''best'' model on the training data (this');
disp('may be considerably more complex than very slightly ''lower'' performing');
disp('models.');
disp('>>popbrowser(gp);');

disp('Press a key to continue');disp(' ');pause;
popbrowser(gp);

%If Symbolic Math toolbox is present
if gp.info.toolbox.symbolic

    %pareto report
    disp(' ');
    disp('The PARETOREPORT function generates a standalone interactive HTML
report');
    disp('listing the multigene regression models on the Pareto front in terms');
    disp('of their simplified equation structure, expressional complexity and');
    disp('performance on the training data (R2).');
    disp('These models correspond to the green circles in the popbrowser');
    disp('visualisation and can be sorted by performance or complexity by');
    disp('clicking on the appropriate column header.');
    disp('>>paretoreport(gp);');
    disp('Press a key to continue');disp(' ');pause;
    paretoreport(gp);

    %gppretty
    disp(' ');
    disp('It is possible to display any multigene model at the command line.');
    disp('E.g. to use the GPPRETTY command on the ''best'' model on the training
data: ');
    disp('>>gppretty(gp,''best'')');
    disp('Press a key to continue');
    disp(' ');
    pause;
    gppretty(gp,'best');
end
disp(' ');
disp('Additionally, the DRAWTREES function can be used to draw the genes in any');
disp('model to a browser window.');
disp('E.g. to draw the genes in the ''best'' model on the training data use');
```

```
disp('>>drawtrees(gp,''best'');');
disp('Press a key to continue');disp(' ');pause;
drawtrees(gp,'best');

%reports
if gp.info.toolbox.symbolic

    disp(' ');
    disp('Finally, for multigene models the GPMODELREPORT function can be ');
    disp('used to generate a comprehensive model performance report for ');
    disp('reference purposes. This is created in a browser window.');
    disp('E.g. to generate a performance report for the ''best'' model');
    disp('on the training data use');
    disp('>>gpmodelreport(gp,''best'');');
    disp('Press a key to continue');disp(' ');pause;
    gpmodelreport(gp,'best');
end
```