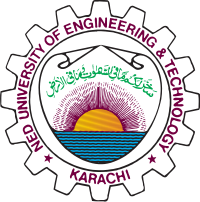
NED University of Engineering and Technology



Name: Hafiz Erass Mirza

Roll no: ME-17318

Section: ‘A’

Task: Open Ended Lab

Script file:

clear, clc, close

%% QUESTION 01

% EFI Engine given data

EFI\_rpm = [1522 1691 1803 1898 2000];

EFI\_bhp = [4.029 4.916 5.644 5.800 6.025];

% DIESEL Engine given data

DIESEL\_rpm = [1517 1707 1922 2009 2224];

DIESEL\_bhp = [1.023 1.207 1.437 1.809 1.744];

% PETROL Engine given data

PETROL\_rpm = [994 1097 1200 1309 1395];

PETROL\_bhp = [0.823 0.785 1.853 1.911 1.841];

**Part 1)**

% For curve fitting using polyfit function

p1 = polyfit(EFI\_rpm,EFI\_bhp,2);

p2 = polyfit(DIESEL\_rpm,DIESEL\_bhp,2);

p3 = polyfit(PETROL\_rpm,PETROL\_bhp,2);

% New bhp values after curve fitting

eb\_p = polyval(p1,EFI\_rpm); % EFI

db\_p = polyval(p2,DIESEL\_rpm); % DIESEL

pb\_p = polyval(p3,PETROL\_rpm); % PETROL

**Part 3)**

% Error estimation

errorEFI = (eb\_p - EFI\_bhp)\*100 % EFI

errorDIESEL = (db\_p - DIESEL\_bhp)\*100 % DIESEL

errorPETROL = (pb\_p - PETROL\_bhp)\*100 % PETROL

**Part 4)**

% Prediction

efi\_at\_1500 = polyval(p1,1500) % EFI

diesel\_at\_1500 = polyval(p2,1500) % DIESEL

petrol\_at\_1500 = polyval(p3,1500) % PETROL

**Part 2)**

% Graph

figure

subplot(3,1,1); hold on;

plot(EFI\_rpm,EFI\_bhp,'rx',EFI\_rpm,eb\_p);

plot(1500,efi\_at\_1500,'o','MarkerSize',6,'Linewidth',2); hold off;

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('EFI Engine'); grid;

legend('Given Data','Model','BHP at 1500');

subplot(3,1,2); hold on;

plot(DIESEL\_rpm,DIESEL\_bhp,'rx',DIESEL\_rpm,db\_p)

plot(1500,diesel\_at\_1500,'o','MarkerSize',6,'Linewidth',2); hold off;

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('DIESEL Engine'); grid;

legend('Given Data','Model','BHP at 1500');

subplot(3,1,3); hold on;

plot(PETROL\_rpm,PETROL\_bhp,'rx',PETROL\_rpm,pb\_p)

plot(1500,petrol\_at\_1500,'o','MarkerSize',6,'Linewidth',2); hold off;

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('PETROL Engine'); grid;

legend('Given Data','Model','BHP at 1500');

**Part 5)**

% Linear Relation

% For curve fitting using polyfit function

pl1 = polyfit(EFI\_rpm,EFI\_bhp,1);

pl2 = polyfit(DIESEL\_rpm,DIESEL\_bhp,1);

pl3 = polyfit(PETROL\_rpm,PETROL\_bhp,1);

% New bhp values after curve fitting

eb\_lp = polyval(pl1,EFI\_rpm); % EFI

db\_lp = polyval(pl2,DIESEL\_rpm); % DIESEL

pb\_lp = polyval(pl3,PETROL\_rpm); % PETROL

% Graph

subplot(3,2,2);

plot(EFI\_rpm,EFI\_bhp,'rx',EFI\_rpm,eb\_lp);

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('EFI Engine (Linear)'); grid;

legend('Given Data','Model');

subplot(3,2,4);

plot(DIESEL\_rpm,DIESEL\_bhp,'rx',DIESEL\_rpm,db\_lp)

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('DIESEL Engine (Linear)'); grid;

legend('Given Data','Model');

subplot(3,2,6);

plot(PETROL\_rpm,PETROL\_bhp,'rx',PETROL\_rpm,pb\_lp)

xlabel('Speed(rpm)'); ylabel('Power (bhp)');

title('PETROL Engine (Linear)'); grid;

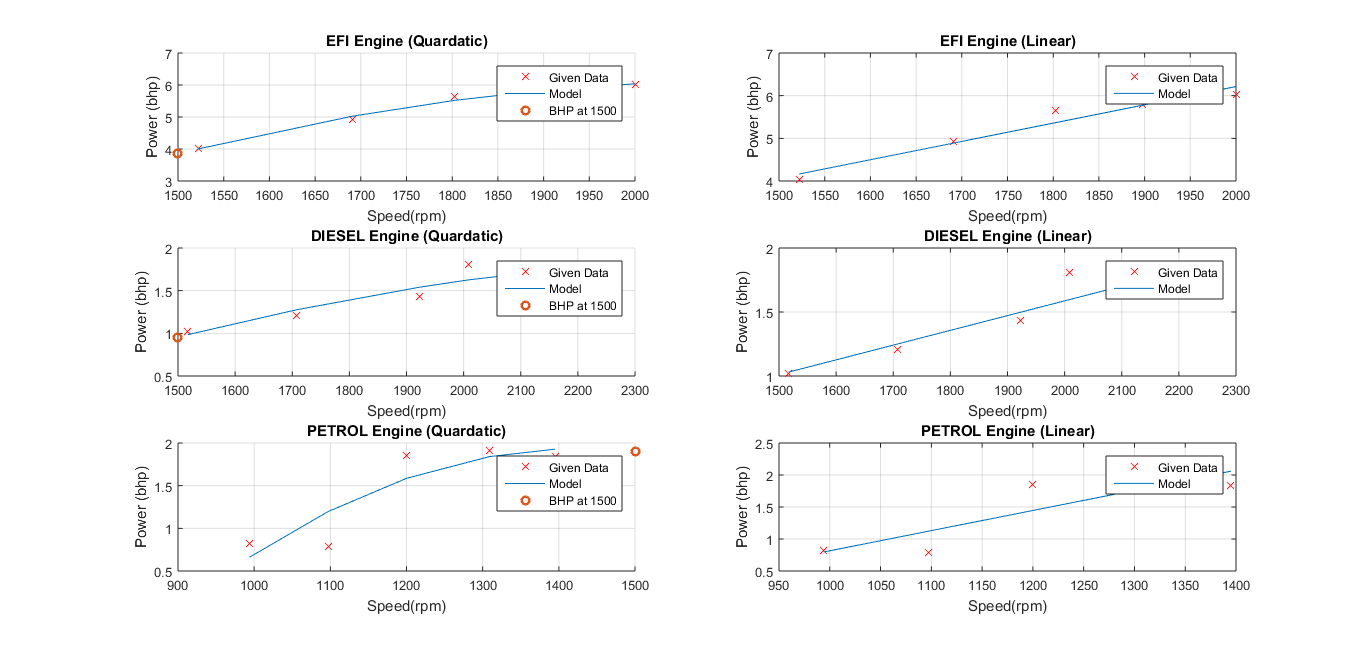
legend('Given Data','Model');

**Analysis:**

Rpm vs torque not follow the linear relation because linearity is directly proportional relation, and we observed that increase the value of rpm also increase the value of torque but once the value of torque reaches at peak value it decreases when increasing the rpm because higher the rpm results more friction and also not proper intake strokes as a result a parabola graph is formed when we plot those values, So we can say rpm vs torque graph is parabolic and also called as quadratic in nature.

Also we can clearly observe in the graph below that the linear graph of all three engines is not best fitted as compared to the quadratic graph of engines, so quadratic polynomial is preferable in curve fitting as compared to linear polynomial.

Graph:



Script file:

%% QUESTION 02

% log(IT) = - beta\*L + log(Io\*(1-R)^2)

% Intensity of incident beam (I\_B) = 5 W/m2

I\_B = 5;

% Length of a transparent solid

L = [0.5 1.2 1.7 2.2 4.5 6.0];

% Transmitted intensity

I\_T = [4.2 4.0 3.8 3.6 2.9 2.5];

% For curve fitting using polyfit function

p = polyfit(L,log(I\_T),1);

% Absorption coefficient

beta = - p(1);

fprintf('\tAbsoption coefficient = %f\n',beta);

% Fraction of light which is reflected at the interface

% p(2) = log(I\_B\*(1-R)^2)

R = 1 - sqrt(exp(p(2)) / I\_B);

fprintf('\tFraction of light = %f\n',R);

% Index of refraction for the transparent solid

% R = sqrt((n-1)/(n+1))

n = 2/(1-sqrt(R))-1;

fprintf('\tIndex of refraction = %f\n',n);

% Graph

Lp = linspace(0.5,6,100);

F = @ (x) I\_B\*(1-R)^2\*exp(-beta\*x);

I\_Tp = F(Lp);

figure

plot(L,I\_T,'ro',Lp,I\_Tp)

xlabel('L (cm)'); ylabel('I\_T (W/m^2)')

legend('Given Data','Model')

title('Intensity of light through transparent solid')

% Absorption coefficient & Index of refraction Values writen on graph

str = {'n = 1.6242','\beta = 0.0956'};

text(5.3,4.1,str)

Graph:

