



COMSATS University Islamabad, Wah Campus

Machine Learning

[Weather/Temperature Prediction] Project Report

By

AHSAN AMAN

SP18-BCS-036-6A

Contents

Project Name:	3
Details about Dataset:	3
Target/Location Dataset:	3
Was dataset available online (in one collectable form)?.....	3
Tools used to Process Data	4
Macro Recorder	4
Source of data	4
Google Form.....	4
Implementation in MATLAB	5
Initial setup	5
Filter data	5
Displaying Plots	6
Regression	9
Code:	10
Testing or Demonstration:	11
First try:	11
Second try	12
Data Matrix/Comparison	13

Project Name:

Weather Prediction

Details about Dataset:

Target/Location Dataset:

Islamabad, Pakistan

Was dataset available online (in one collectable form)?

No, I wasn't able to find a collection or dataset relating to Pakistan weather I had to design and code a special program to that would scrap the data relating to Pakistani weather and collect and much data as possible.

<http://www.pmdnmcc.net/RealTime/Data/CT.txt>

CONSOLIDATED TEMPERATURE DATED 25-12-2020

=====			
SR.NO.	STATION.	MIN.TEMP	MAX.TEMP
01.	ISLAMABAD	02.0	16.0
02.	QUETTA	02.4	13.5
03.	SIBBI	05.0	25.2
04.	JACOBABAD	07.5	21.5
05.	ROHRI	09.0	23.0
06.	NAWABSHAH	08.0	25.5
07.	HYDERABAD	14.0	25.0
08.	KARACHI	11.5	27.0

The above image shows the data the PMD has published and this data itself has no additional parameters and the file itself is Plain Text and not CSV (Comma Separated Values). This led to the conclusion that the data published was not useful.

Tools used to Process Data

Macro Recorder

At first, I used **Macro recorder** to automate the process of collecting dataset from the online discrete sources available and coding as well as the screen capture was done all in a custom and unique way.

I did make a short video clip of the process which would demonstrate how the macro was working.

YouTube: <https://youtu.be/-duyzc3GyAM>

Source of data

I used the website www.worldweatheronline.com to capture data with the method mentioned above.

Since I was able to collect valuable data in a short period of time I decided to open-source the data so that other students or researchers on the internet can contribute and find it helpful as well.

Link to my profile:

<https://data.world/ahsanaman/> (Check the resources)

<https://data.world/ahsanaman/ahsanamanpakistan-climate-data-and-daily-weather-reports>

(this is direct link)

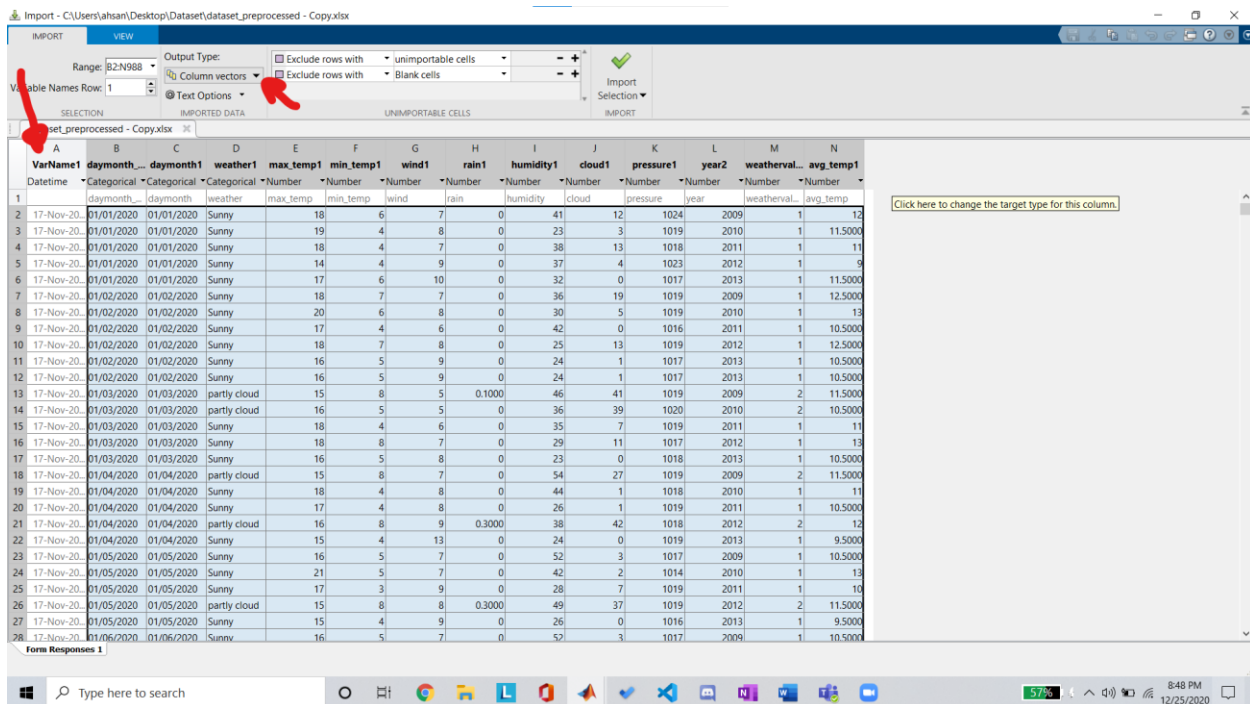
Google Form

When the data was captured the macro, script was recorded in a way that, it was able to copy each year's data in a variable and post it in Google Form that I designed specifically for this project.

Using Google Form gave me the advantage that it automatically aligns the data and keeps a clean record for the data entered, also it would export it in different file types like **CSV**, **PDF** ... etc.

Initial setup

At first, we need to import the data into the MATLAB work environment.



We need to keep in mind that for training purposes we do need to keep the number of indexes or categories same for each and every variable. Therefore, we need to set a condition for excluding rows with un-importable cells.

Filter data

After we import, we first need to run

```
%% Filter Data
daymonth=datetime(daymonth,'InputFormat','MM/dd/yyyy');
```

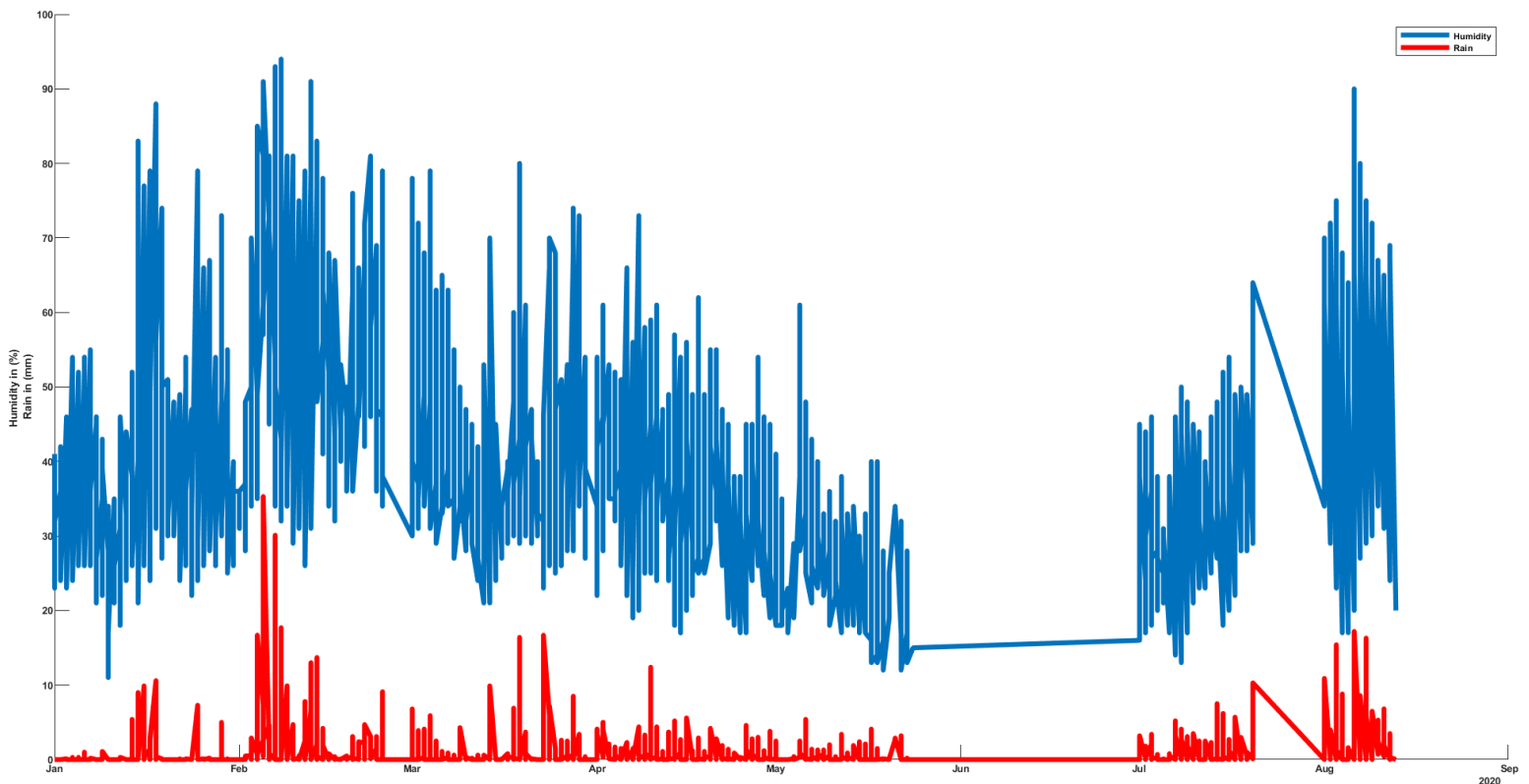
This will convert the string datetime to actual datetime format

Displaying Plots

Once we convert the class of the variable we can then use:

```
%% This is for Plotting the data for Humidity & Rain
hold on
plot(daymonth,humidity,'LineWidth',3)
plot(daymonth,rain,"r",'LineWidth',3)
set(gca,'FontSize',6,'FontWeight','bold')
hold off
```

to display this plot that show relation between **Humidity** and **Rain**.

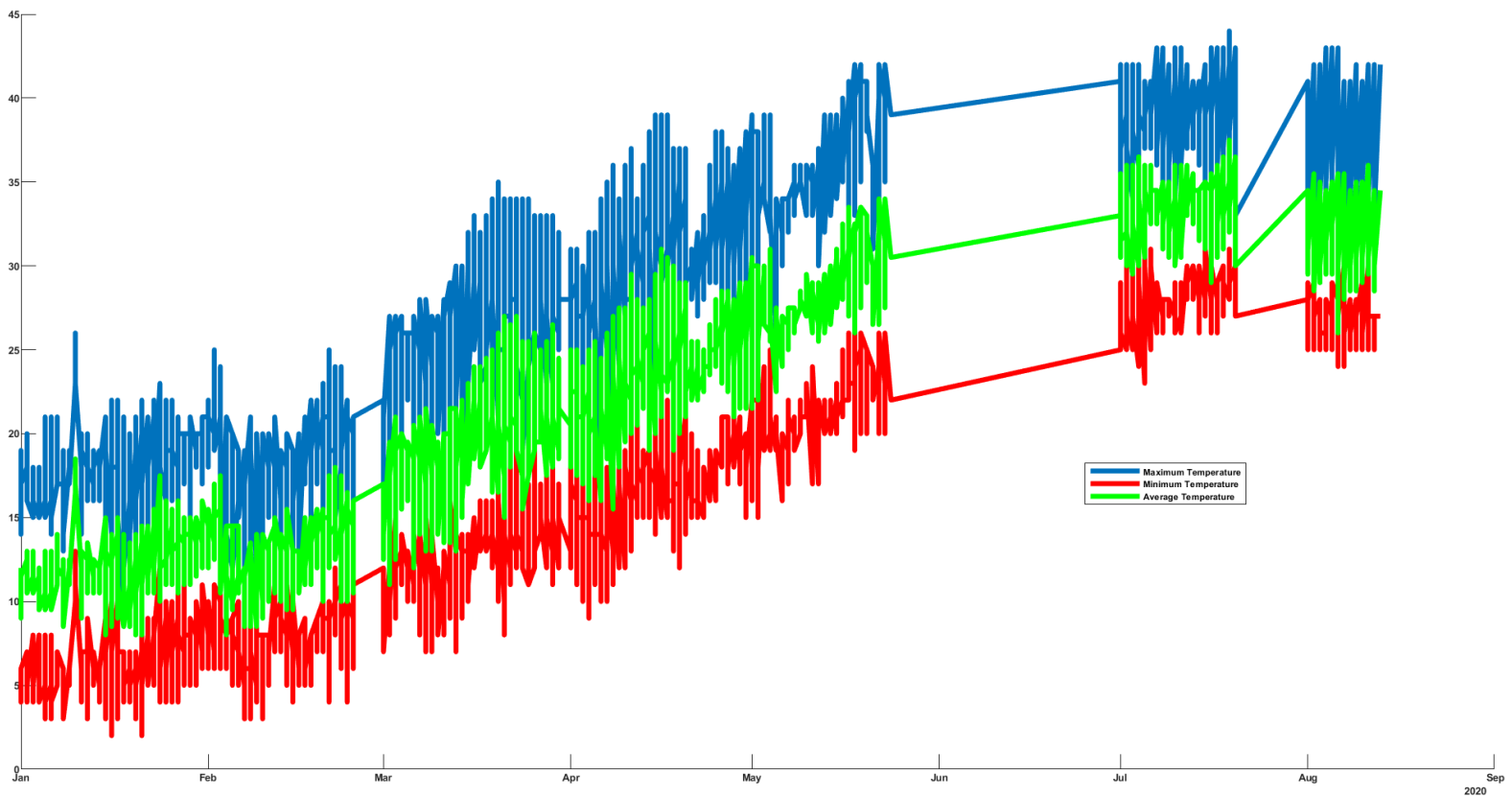


Keep in note that we have intentionally **excluded** the month June, October, November, December for testing purposes.

Now run this code:

```
%% This is for Plotting the data for Max and Min  
temperature  
hold on  
plot(daymonth,max_temp,'LineWidth',3)  
plot(daymonth,min_temp,"r",'LineWidth',3)  
plot(daymonth,avg_temp,"-g",'LineWidth',3)  
set(gca,'FontSize',6,'FontWeight','bold')  
hold off
```

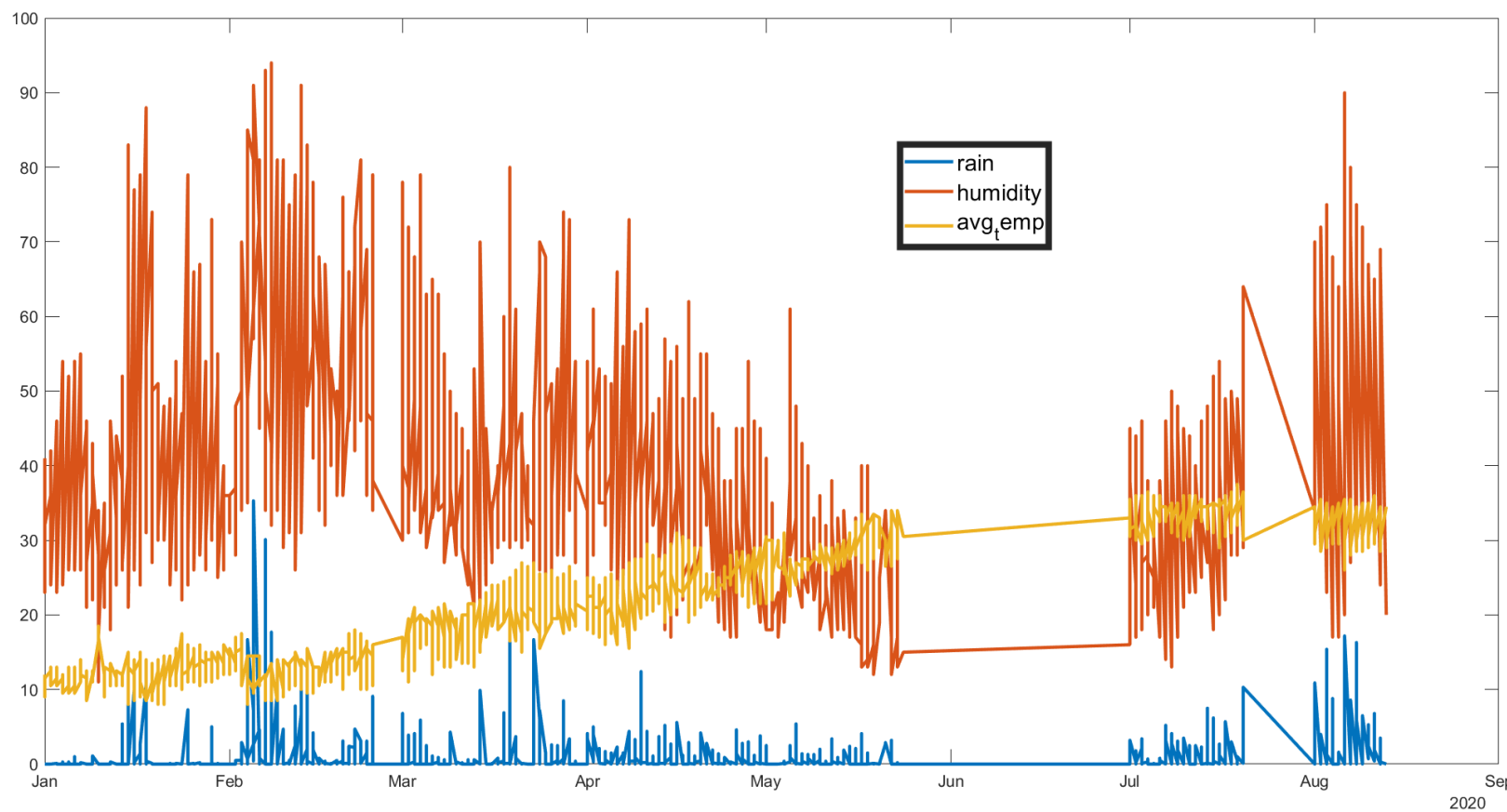
to display the plot that shows the relation between **MAX**, **MIN** & **AVERAGE** Temperature:



```

%% This Plotting represents the combine effect of climate
change
plot(daymonth,rain,'DisplayName','rain','LineWidth',2);
hold on;
plot(daymonth,humidity,'DisplayName','humidity','LineWidth'
,2);
plot(daymonth,avg_temp,'DisplayName','avg_temp','LineWidth'
,2);
hold off;

```



Regression

Since the workspace is all setup and running now, we need to open the Curve Fitting App that is all built-in in MATLAB and add the following variables

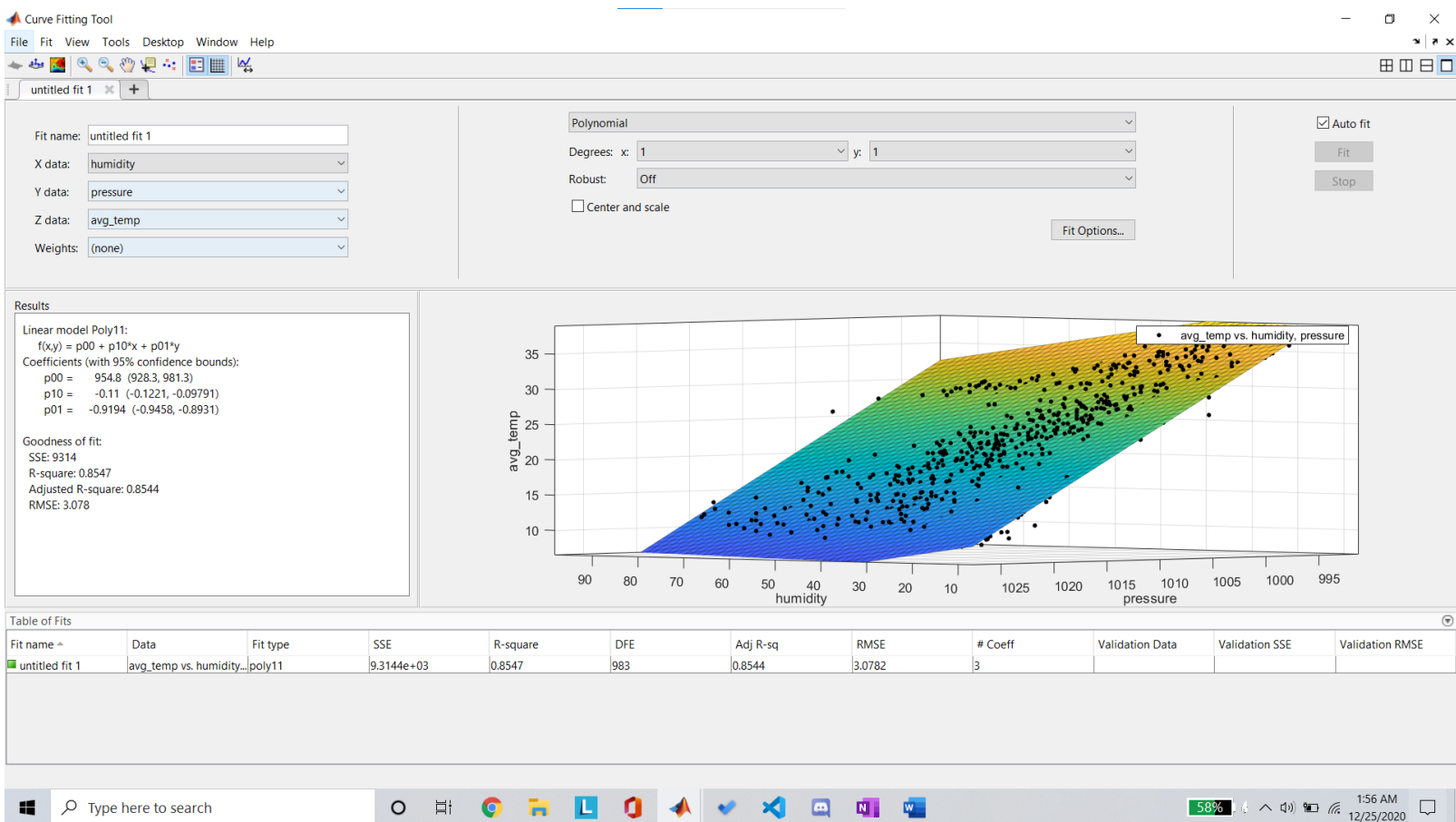
Fit name:

X data:

Y data:

Z data:

Weights:



Code:

```
function [fitresult, gof] = createFit(humidity, pressure,
avg_temp)
%CREATEFIT(HUMIDITY,PRESSURE,AVG_TEMP)
% Create a fit.
%
% Data for 'untitled fit 1' fit:
%     X Input : humidity
%     Y Input : pressure
%     Z Output: avg_temp
% Output:
%     fitresult : a fit object representing the fit.
%     gof : structure with goodness-of fit info.
%
% See also FIT, CFIT, SFIT.

% Auto-generated by MATLAB on 25-Dec-2020 02:00:07

%% Fit: 'untitled fit 1'.
[xData, yData, zData] = prepareSurfaceData( humidity,
pressure, avg_temp );

% Set up fittype and options.
ft = fittype( 'poly11' );

% Fit model to data.
[fitresult, gof] = fit( [xData, yData], zData, ft );

% Plot fit with data.
figure( 'Name', 'untitled fit 1' );
h = plot( fitresult, [xData, yData], zData );
legend( h, 'untitled fit 1', 'avg_temp vs. humidity,
pressure', 'Location', 'NorthEast', 'Interpreter', 'none'
);
% Label axes
xlabel( 'humidity', 'Interpreter', 'none' );
ylabel( 'pressure', 'Interpreter', 'none' );
zlabel( 'avg_temp', 'Interpreter', 'none' );
grid on
view( -169.7, 23.2 );
```

Testing or Demonstration:

As we mentioned earlier in the document, we excluded some of the Months intentionally for testing purpose now we are going to test our model for the test data.

To load our model we need to select the target path on the MATLAB and run the following code on Live script

```
model= createFit_Code(humidity,pressure,avg_temp)
```

our model is now created

First try:

The screenshot displays a MATLAB R2020a live script environment. The background shows a web browser with the 'Historical Weather on 23rd December over the years' for Islamabad. A calculator window is open, showing calculations for average temperature. The MATLAB editor contains a script 'createFit_Code.m' with fit type 'poly11'. The Command Window shows the model being called with inputs (41, 1014) and returning a predicted temperature of 17.9969. Red arrows and boxes highlight the data points and calculations used for the test.

Year	Weather	Max	Min	Wind	Rain	Humidity
2009	Sunny	19 °C	5 °C	7 km/h SSE	0.0 mm	37%
2010	Sunny	19 °C	6 °C	8 km/h E	0.0 mm	20%
2011	Partly cloudy	17 °C	5 °C	10 km/h S	0.0 mm	14%
2012	Partly cloudy	20 °C	8 °C	6 km/h SE	0.0 mm	26%
2013	Sunny	16 °C	5 °C	9 km/h ESE	0.0 mm	25%
2014	Cloudy	19 °C	10 °C	11 km/h NE	0.0 mm	30%
2015	Sunny	15 °C	6 °C	9 km/h SE	0.1 mm	40%
2016	Sunny	24 °C	10 °C	6 km/h S	0.0 mm	13%
2017	Partly cloudy	20 °C	11 °C	6 km/h SE	0.0 mm	25%
2018	Partly cloudy	17 °C	9 °C	8 km/h ESE	0.0 mm	24%
2019	Overcast	17 °C	10 °C	6 km/h SW	0.0 mm	36%
2020	Partly cloudy	20 °C	12 °C	6 km/h SSE	0.0 mm	41%

Yearly Averages

Calculator: $32 \div 2 = 16$

The average temperature according to the website is 16

Calculator: $20 + 12 = 32$

Editor - C:\Users\ahsan\Desktop\Dataset\Presentation Files\cre...
createFit_Code.m
% Fit: 'untitled fit 1'.
[xData, yData, zData] = prepareSurfaceD
% Set up fitttype and options.
ft = fitttype('poly11');
% Fit model to data.
[fitresult, gof] = fit([xData, yData], zData, ft);

Command Window
>> model(41,1014)
ans =
17.9969

This is the Temperature predicted by our model which is almost near to the actual value

Second try

climate-da | Channel analytics - YouTube Stud | Laptop & Computer Prices in Pak | Calculator

worldweatheronline.com/islamabad-weather-history/islamabad/pk.aspx

World Weather Online

Home API HWD World Sports Videos Holiday Blog Maps

Enter Post/Zip Code, City/Town, IATA, IP

18:00 Partly cloudy 41 °C 42 °C 14 km/h from WNW 17 km/h 0.0 mm 11% 0% 999 mb Excellent

21:00 Partly cloudy 38 °C 39 °C 12 km/h from WNW 18 km/h 0.0 mm 14% 2% 1001 mb Excellent

Islamabad

Historical Weather on 01st June over the years

Year	Weather	Max	Min	Wind	Rain	Humidity	Cloud	Pressure
2009	Sunny	38 °C	24 °C	10 km/h SSE	0.0 mm	18%	9%	1002 mb
2010	Partly cloudy	42 °C	24 °C	10 km/h SE	0.0 mm	13%	8%	1003 mb
2011	Moderate or heavy rain shower	32 °C	24 °C	11 km/h SE	5.8 mm	40%	14%	1001 mb
2012	Sunny	40 °C	25 °C	13 km/h SSE	0.5 mm	21%	5%	1000 mb
2013	Sunny	40 °C	26 °C	9 km/h SE	0.0 mm	13%	0%	1001 mb
2014	Sunny	34 °C	21 °C	13 km/h E	3.5 mm	37%	4%	1007 mb
2015	Sunny	36 °C	23 °C	9 km/h SSE	0.1 mm	22%	3%	1002 mb
2016	Patchy rain possible	38 °C	25 °C	11 km/h SE	1.0 mm	22%	44%	1002 mb
2017	Sunny	37 °C	24 °C	11 km/h S	0.1 mm	24%	1%	1004 mb

Calculator

Standard

62 ÷ 2 =

This is the actual average temperature on 1st June

31

MC MR M+ M- MS M*

% CE C <X>

1/x x^2 √x ÷

7 8 9 ×

4 5 6 −

1 2 3 +

0 . =

Fit

Details

Select a file to view details

Package App

Curve Fitting Optimization PID Tuner APPS

Users > ahsan > Desktop > Dataset > Presentation File

Editor - C:\Users\ahsan\Desktop\Dataset\Presentation File

Project.m createFit_Code.m +

```
%% Fit: 'untitled fit 1'.
[xData, yData, zData] = prepareSu

% Set up fittype and options.
ft = fittype( 'poly11' );

% Fit model to data.
[fitresult, gof] = fit( [xData, v

>> model(18,1002)

ans =

31.5607

fx >>
```

This is the predicted average temperature by our model

57%

Data Matrix/Comparison

#	Actual Average temperature	Predicted Average Temperature
1	16	17
2	31	31