



**SKILL  
GEEK**  
Bringing out the you in you...

# TEMPERATURE PREDICTION AND AUTHENTICATING THE CLAIM OF VARIOUS SCIENTISTS

---

USING  
THE CLIMATE TEMPERATURE  
DATA

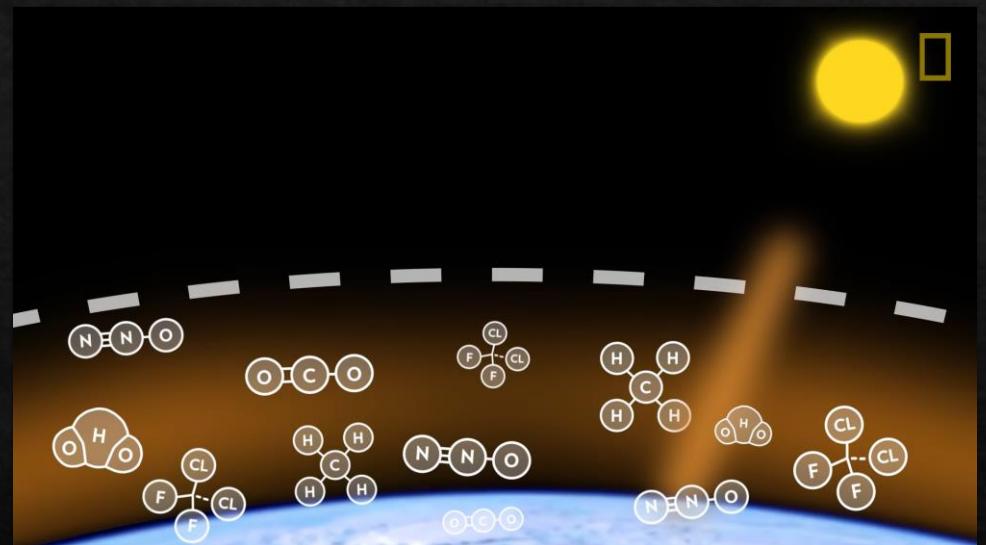


# Project Brief

- To understanding the “Climate Change phenomenon”.
- “Authenticating” various claim of Scientist using temperature data.
- Also “Predictions the temperature”.

# Introduction

- In this modern world increase in “Climate Temperature” is the main problem.
- Increase in temperature rapidly is because of “Greenhouse Gases”.
- Firstly, Greenhouse gases are Carbon Dioxide, Methane, Nitrous Oxide, Fluorinated Gases, ozone.
- If the Greenhouse gases are in limit there will be no problem in climate.



- But in 1859, according to “Irish physicist John Tyndall” after the industrial revolution more amount of Carbon dioxide is emitting by industries.
- That leads to “Global Warming”
- Seasonal Affective Disorder
- Glacier Melting increase water level and decrease the percentage of land



But, most of the “industrialist and some other scientists” says that the Co<sub>2</sub> released by industrials are in limit it not leads to danger to life on earth.

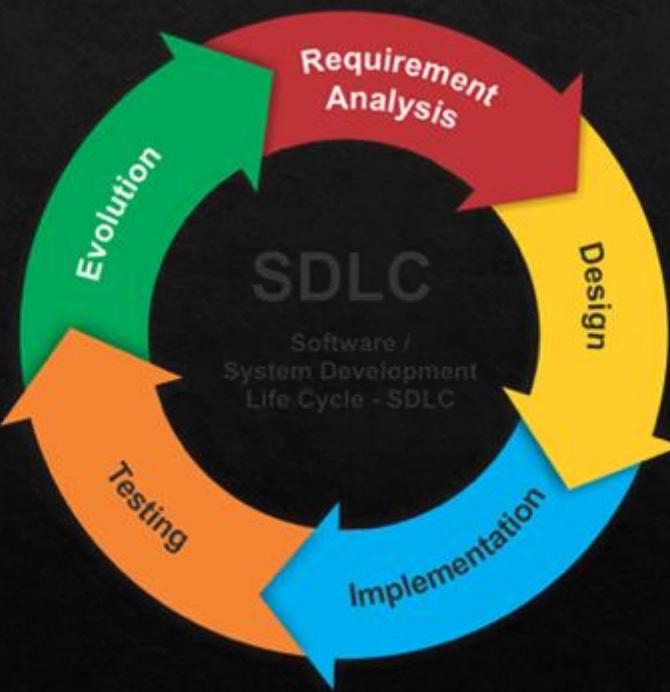


# Project Goal

- Study the Climate change Phenomena and dataset.
- Predict the temperature.
- “Authenticate” various claim of Scientist using temperature data.
- Means we want to prove one of the scientist theory is true.



# ARCHITECTURE PATTERN USED



## ARCHITECTURE PATTERN USED STEPS

- ❖ Stage 1: Planning and Requirement Analysis
- ❖ Stage 2: Defining Requirements
- ❖ Stage 3: Designing the Product Architecture
- ❖ Stage 4: Building or Developing the Product
- ❖ Stage 5: Testing the Product
- ❖ Stage 6: Deployment in the Market and Maintenance

# IMPLEMENTATION METHODOLOGY

**IDE:**

Python IDLE, Kaggle Notebook(python)

**Technologies:**

Python

**Dataset size:**

1 Crores

**Dataset:**

Climate Change: Earth Surface Temperature Data

**Algorithms:**

Linear Regression

Random Forest Regression

Decision Tree

Feedforward Neural Networking

**User Interface:**

Python Compiler

# PROGRAM IMPLEMENTATION

- Firstly, We want to read and check the dataset weather there is any missing values errors etc,
- Fill the missing with this three methods mean, medium, mode.
- Split the data training and testing.
- Know do the
  1. Linear Regression
  2. Random Forest Regression
  3. Feedfarward Neural Network
  4. Feedbackward Neural Network
- Visualize Outputs (Bar Graph)

# COMPARISON

DATASET	ALGORITHMS	BEFORE PREDICTION	AFTER PREDICTION
Avg Temp of City	Linear regression	26.544	19
Avg Temp of City	Random forests	26.544	26
Average Temp of City	Decision tree	26.544	26
Avg Temp of Country	Linear regression	26.864	19
Avg Temp of Country	Random forests	26.864	18
Avg Temp of Country	Decision tree	26.864	24
Avg Temp of Major City	Linear regression	26.612	19
Avg Temp of Major City	Random forests	26.612	26
Avg Temp of Major City	Decision tree	26.612	26
Avg Temp of State	Linear regression	18.402	10
Avg Temp of State	Random forests	18.402	9
Avg Temp of State	Decision tree	18.402	8

Dataset	Algorithms	Before Prediction	After Prediction
Avg Temp of Global	Linear regression	13.293	9
Avg Temp of Global	Random forests	13.293	15
Avg Temp of Global	Decision tree	13.293	25
Land min temp of global	Linear regression	-1.139	0
Land min temp of global	Random forests	-1.139	1
Land min temp of global	Decision tree	-1.139	-1
Land max temp of global	Linear regression	19.753	14
Land max temp of global	Random forests	19.753	19
Land max temp of global	Decision tree	19.753	19
Land and Ocean Avg temp of global	Linear regression	15.005	15
Land and Ocean Avg temp of global	Random forests	15.005	15
Land and Ocean Avg temp of global	Decision tree	15.005	15

# SCREEN SHOTS

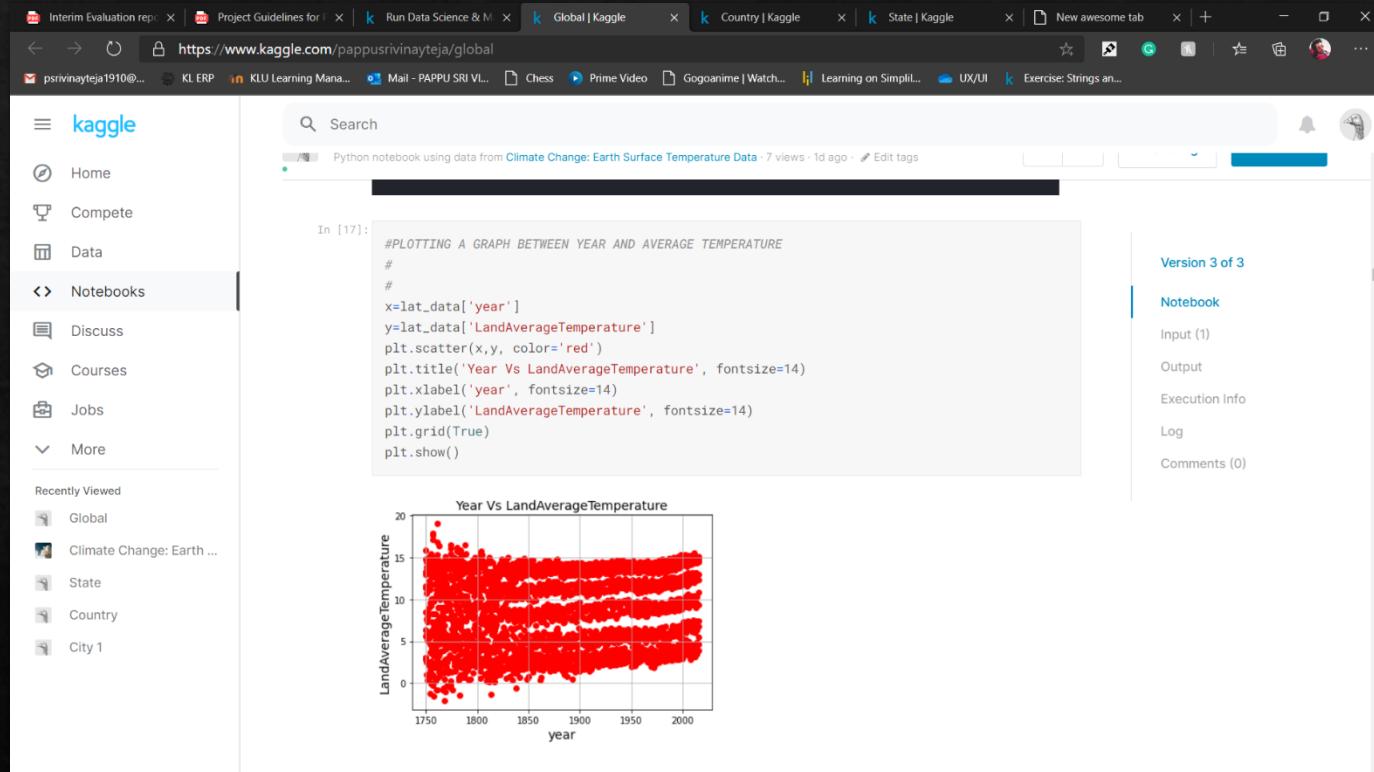


Fig 1.1: Graph plotted between Year and average Temperature

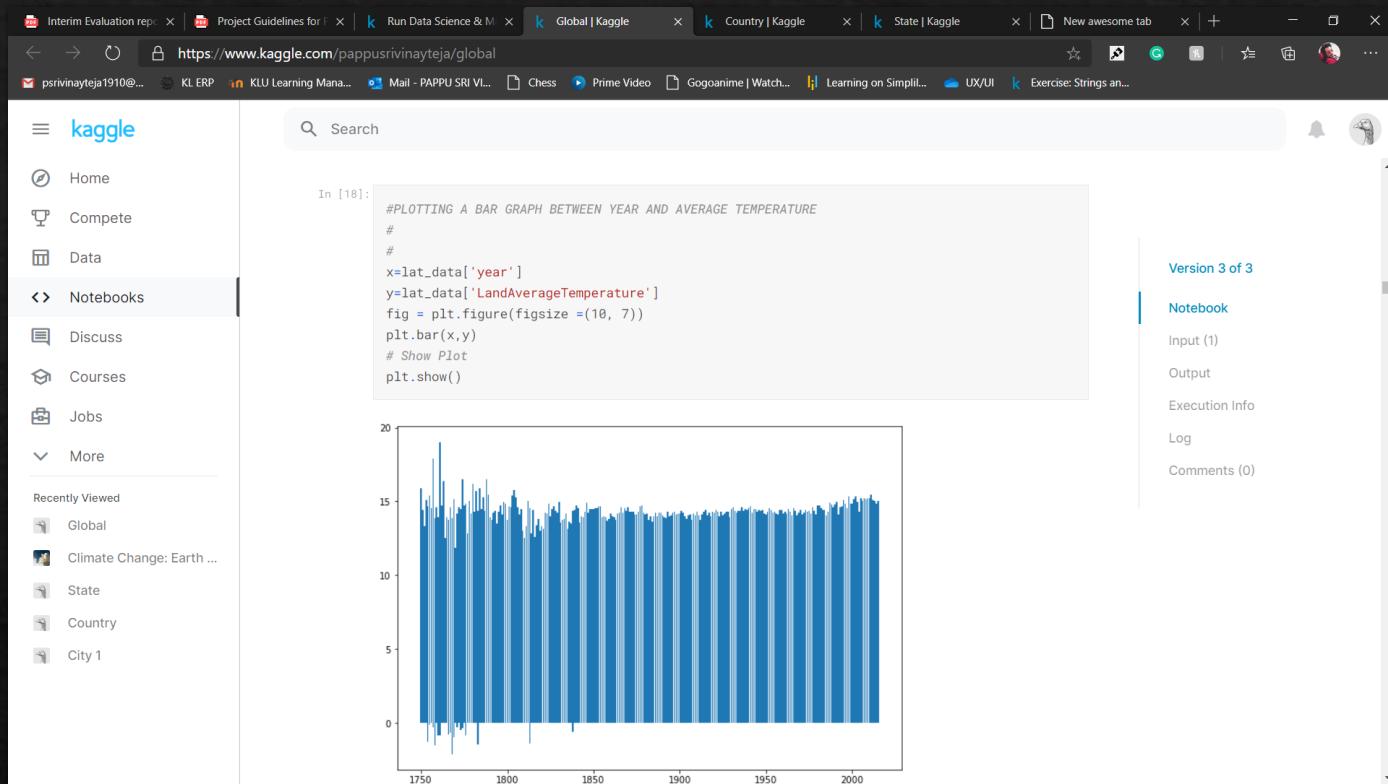


Fig 1.2: Bar Graph plotted between Year and average Temperature

The screenshot shows a Kaggle notebook titled "Global". The notebook contains the following Python code:

```
prediction = lr.predict(test_X)

np.mean((prediction-test_y)**2)

loat_lr_Final=pd.DataFrame({'actual':test_y,'prediction' : prediction,'diff':(test_y-prediction)})
print('Final Linear regression predicted values')
print(loat_lr_Final)
```

The output of the code is a table titled "Final Linear regression predicted values" with the following data:

	actual	prediction	diff
2414	14.278	15.153252	-0.875252
333	15.005	15.117516	-0.112516
1471	16.822	14.988467	1.841533
672	15.005	14.778470	0.226530
3113	17.260	15.495762	1.764238
...	...	...	...
1558	14.340	14.883744	-0.463744
2340	13.586	14.989541	-1.403541
459	15.005	14.871667	0.133333
889	15.005	14.866613	0.138387
2795	14.068	15.701600	-1.633600

[639 rows x 3 columns]

Fig 1.3: Using Linear Regression for prediction for Land and Ocean Avg Temperature

In [98]:

```
#PLOTTING BAR GRAPH OF LINEAR REGRESSION LAND OCEAN AVERAGE TEMPERATURE  
loat_lr_Final.plot.bar()
```

Out[98]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6175e02950>
```

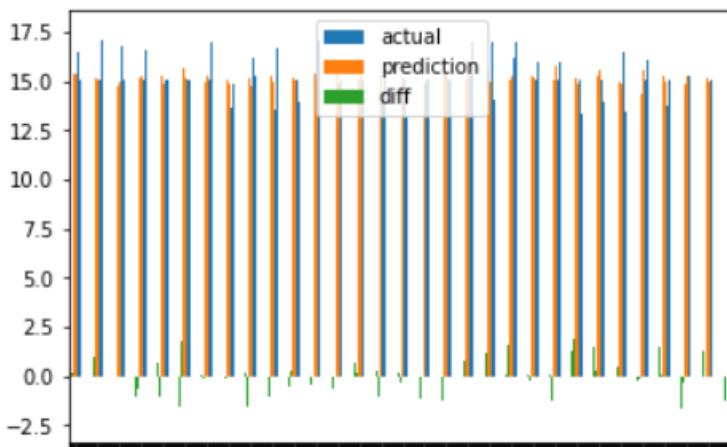


Fig 1.4: Bar graph of actual, prediction and diff of Land and Ocean Avg temperature (LR)

170030975 - Internal Evaluation | 170030975-Internal Report\_TEM | Global | Kaggle

https://www.kaggle.com/pappusrivinayteja/global

psravinayteja1910@... KL ERP KLU Learning Mana... Mail - PAPPU SRI VI... Chess Prime Video Gogoanime | Watch... Learning on Simpli... UX/UI Exercise: Strings an...

kaggle

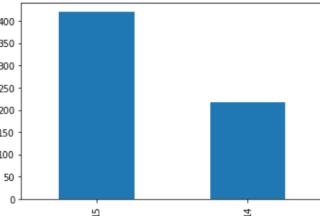
Home Compete Data Notebooks Discuss Courses Jobs More

Recently Viewed Climate Change: Earth ... State Global City Linear Regression City 1

In [100]: #FINDING THE MOST COMMONLY REPEATED VALUE OF LINEAR REGRESSION LAND OCEAN AVERAGE TEMPERATURE  
loat\_lr\_Final\_prediction\_value\_counts=loat\_lr\_Final\_int['prediction'].value\_counts()

In [101]: #PLOTTING BAR GRAPH OF LINEAR REGRESSION LAND AVERAGE TEMPERATURE  
loat\_lr\_Final\_prediction\_value\_counts.plot.bar()

Out[101]: <matplotlib.axes.\_subplots.AxesSubplot at 0x7f6176ae6710>



In [102]: #RANDOM FOREST REGRESSION

Version 3 of 3

Notebook

Input (1)  
Output  
Execution Info  
Log  
Comments (0)

Fig 1.5: Predicted Land and Ocean average temperature (LR)

The screenshot shows a Kaggle notebook interface. The left sidebar has a navigation menu with 'Notebooks' selected. The main area contains a code cell and its output.

```
#RANDOM FOREST REGRESSION

rf=RandomForestRegressor(max_depth=10, random_state=0,n_estimators=100)
rf.fit(train_X,train_y)

prediction = rf.predict(test_X)

np.mean((prediction-test_y)**2)

loat_rf_Final=pd.DataFrame({'actual':test_y,'prediction' : prediction,'diff':(test_y-prediction)})
print('Final Random Forest regression predicted value')
print(loat_rf_Final)
```

Final Random Forest regression predicted value

	actual	prediction	diff
2414	14.278	14.005724	0.272276
333	15.005	15.002309	0.002691
1471	16.822	16.046353	0.775647
672	15.005	15.002309	0.002691
3113	17.260	15.915160	1.344840
...	...	...	...
1558	14.340	15.114089	-0.774089
2340	13.586	14.604237	-1.018237
459	15.005	15.002309	0.002691
889	15.005	15.002309	0.002691
2795	14.068	15.732804	-1.664804

[639 rows x 3 columns]

Version 3 of 3

Notebook

Input (1)

Output

Execution Info

Log

Comments (0)

Fig 1.6: Using Random Forest for prediction for Land and Ocean Avg Temperature

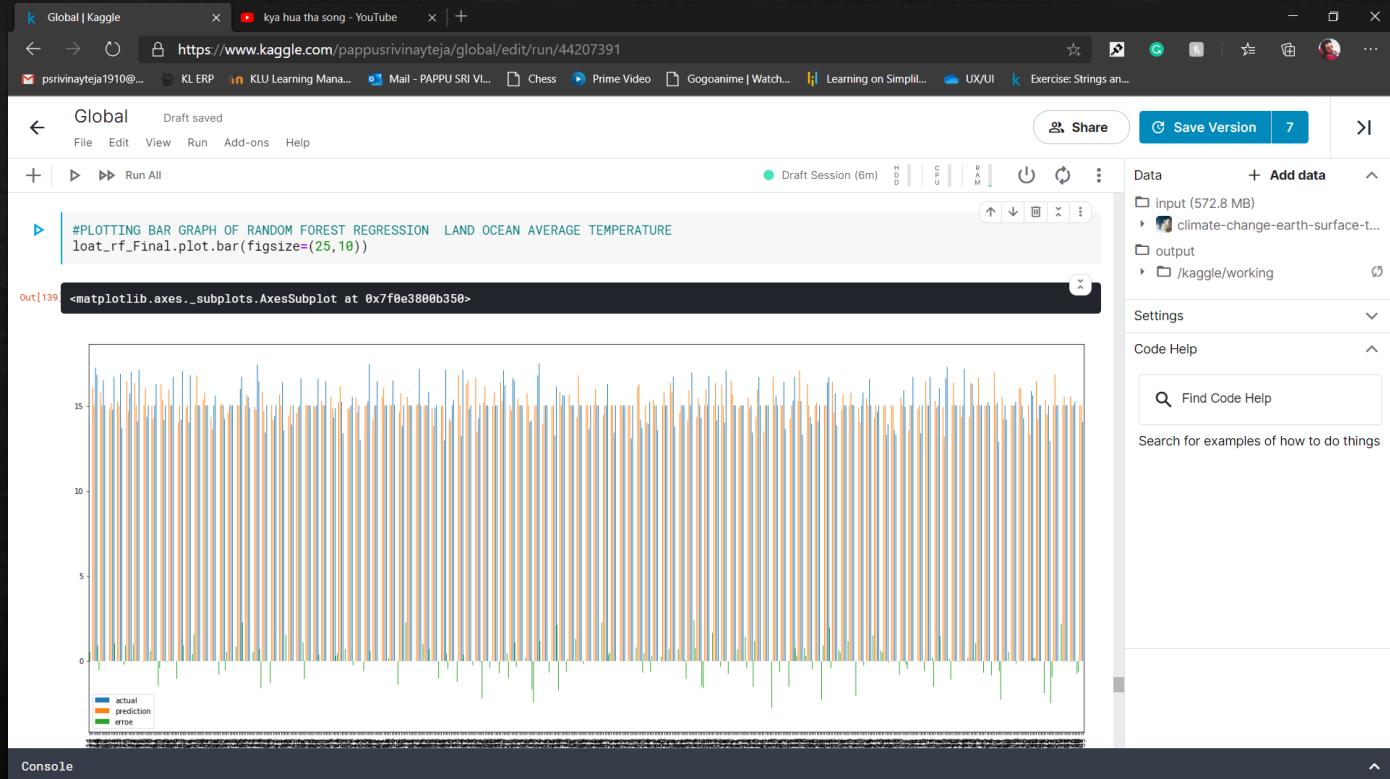


Fig 1.7: Bar graph of actual, prediction and diff of global temperature (RF)

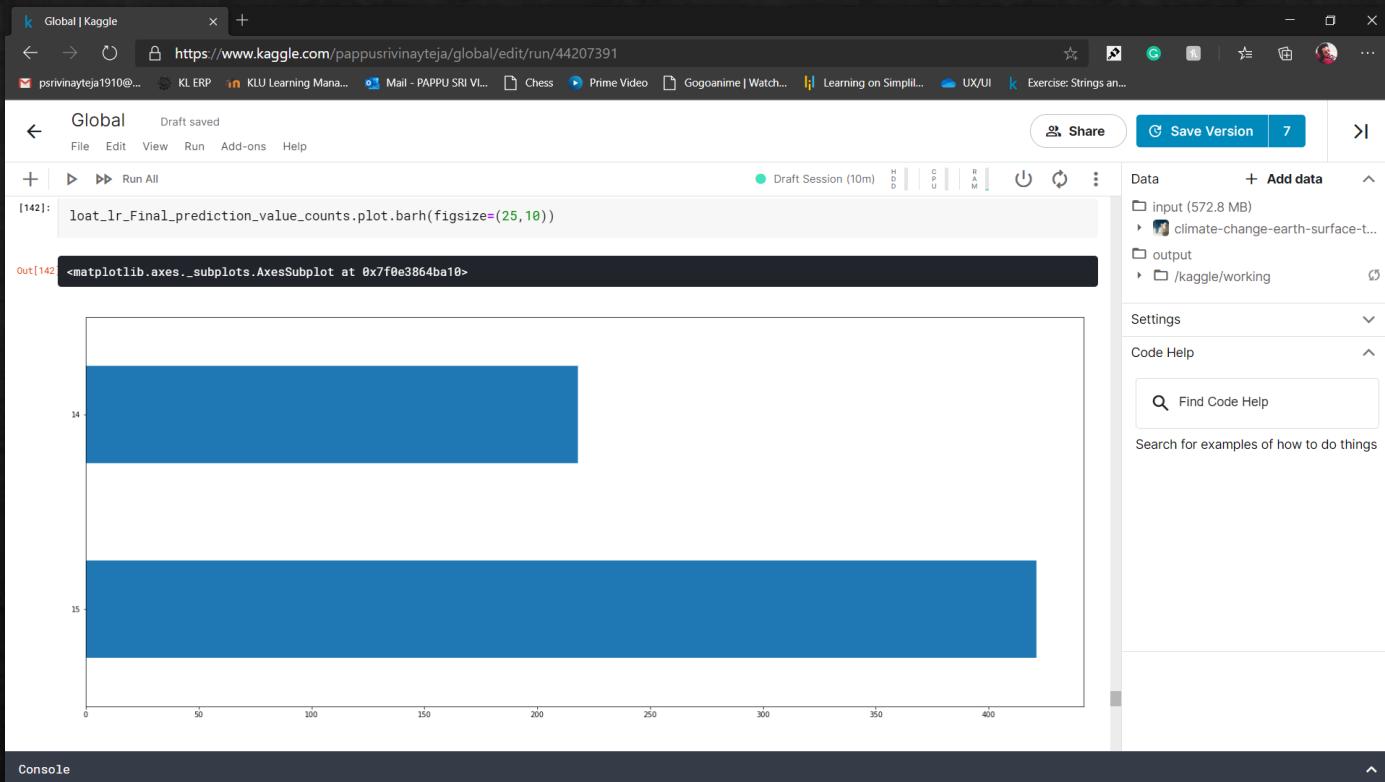


Fig 1.8: Predicted Land and Ocean average temperature (RF)

The screenshot shows a Kaggle Notebook interface titled "Global". The notebook has a "Draft saved" message at the top. The code cell [143] contains the following Python code:

```
#DECISION TREE
dt = DecisionTreeRegressor(max_leaf_nodes=100, random_state=1)

# fit your model
dt.fit(train_X, train_y)

loat_dt_prediction = dt.predict(test_X)

loat_dt_Final=pd.DataFrame({'actual':test_y,'prediction' : loat_dt_prediction,'diff':(test_y-loat_dt_prediction)})
print(loat_dt_Final)
```

The output of the code is a DataFrame with three columns: "actual", "prediction", and "diff". The first few rows of the output are:

	actual	prediction	diff
2414	14.278	13.884727	0.393273
333	15.005	15.005191	-0.000191
1471	16.822	15.888878	0.933122
672	15.005	15.005191	-0.000191
3113	17.260	15.798063	1.461938

[639 rows x 3 columns]

The right sidebar shows the "Data" section with "input (572.8 MB)" and "output" sections, and "Settings" and "Code Help" dropdowns.

Fig 1.9: Using Decision Tree for prediction for Land and Ocean Avg Temperature

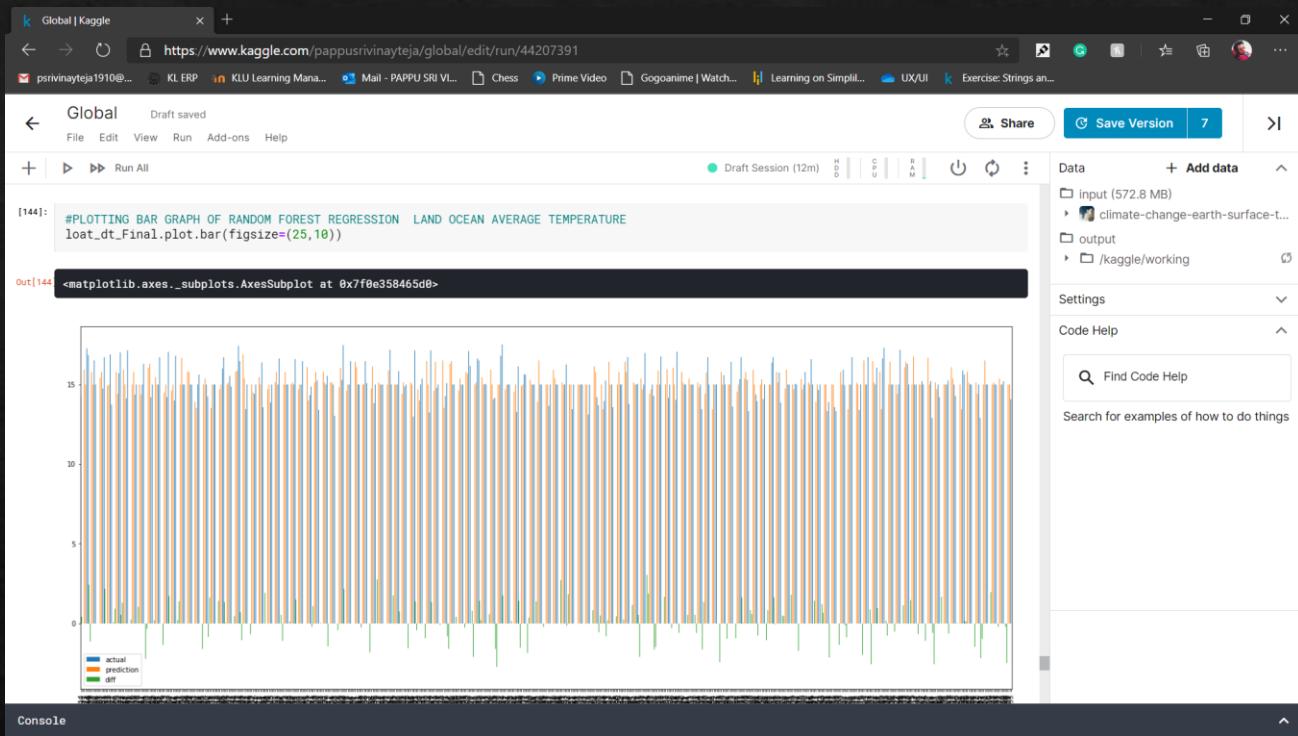


Fig 2.1: Bar graph of actual, prediction and diff of global temperature (DT)

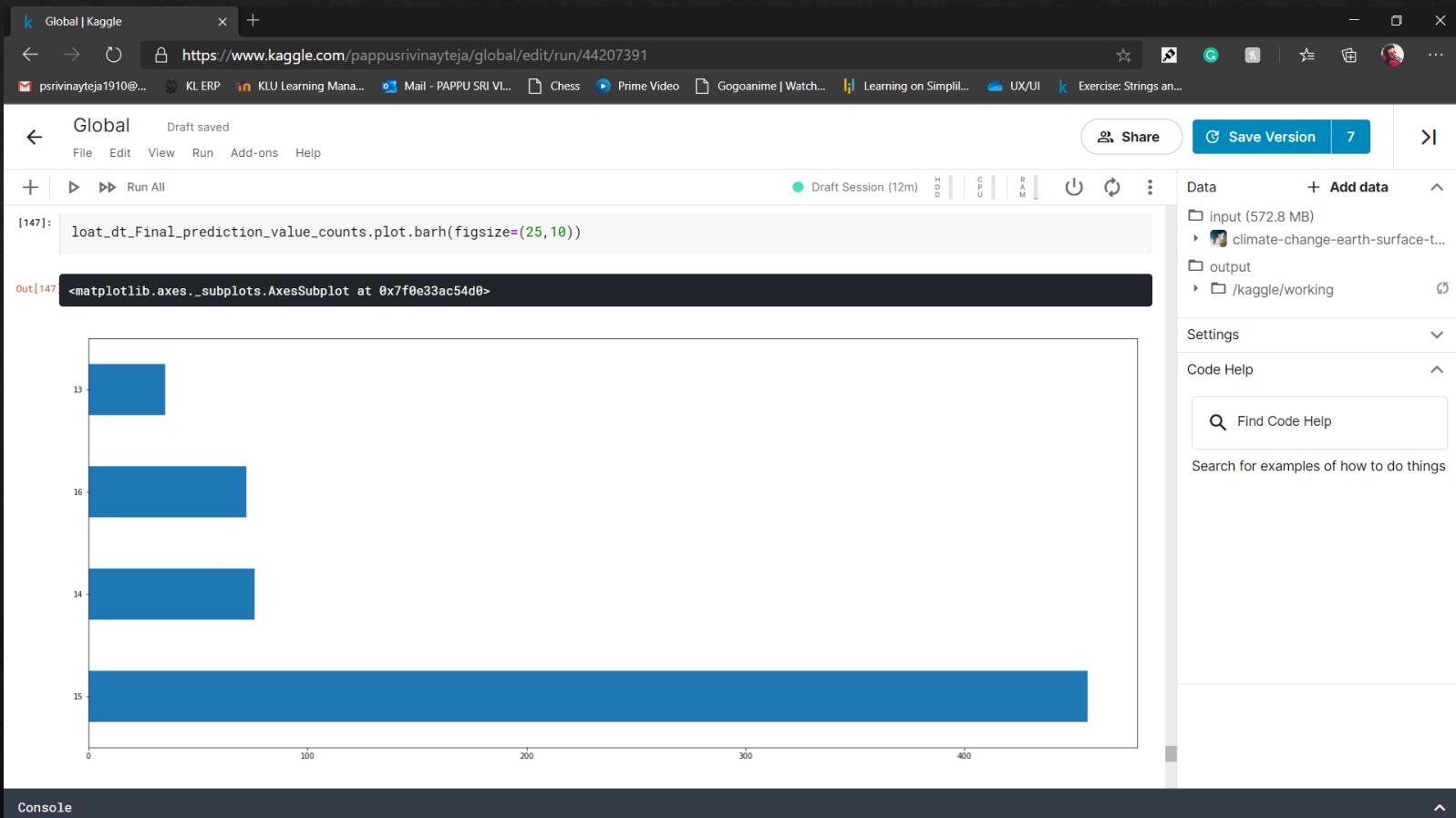


Fig 2.2: Predicted Land and Ocean average temperature (DT)

The screenshot shows a Kaggle Global session window. The top bar displays the title "Global | Kaggle" and the URL "https://www.kaggle.com/pappusrinayteja/global/edit/run/44207391". The main area contains Python code for initializing weights and calculating the output of a feed-forward neural network. The code uses mathematical operations like matrix multiplication and the sigmoid function to compute hidden layer outputs and the final output. The output section shows the results of these calculations.

```
#initializing the input weights
w1=0.2
w2=0.2
w3=0.2
w4=0.2
#initializing th hidden layer weights
w5=0.2
w6=0.2

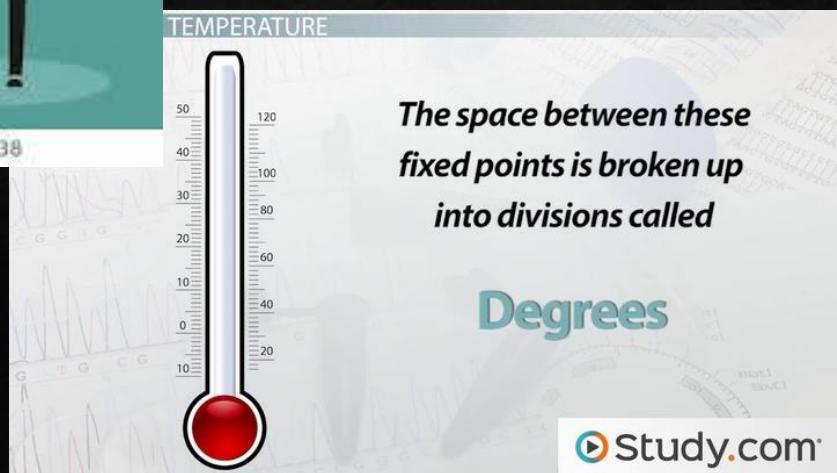
[152]: for i in range(0, len(x)):
    #calculating net input for hidden layer z1
    z1=x[i]*w1+x[i]*w3+b1
    #calculating net input for hidden layer z2
    z2=x[i]*w2+x[i]*w4+b2
    outz1=1/(1+math.exp(-z1))
    outz2=1/(1+math.exp(-z2))
    t_out=outz1*w5+outz2*w6+b3
    outy=1/(1+math.exp(-t_out))
    total_error=(1/2)*(Y[i]-outy)**2
print('the output of z1 in the hidden layer is',outz1)
print('the output of z2 in the hidden layer is',outz2)
print('net output of y is',outy)
print('the total error is',total_error)
print('net output of y is',outy)

the output of z1 in the hidden layer is 0.9975509425953525
the output of z2 in the hidden layer is 0.9975509425953525
net output of y is 0.6222290885668514
the total error is 79.66033358213157
net output of y is 0.6222290885668514
```

Fig 2.3: Using Feed Forward Neural Network for prediction for Land and Ocean Avg Temperature

# Use Case

- This algorithm is used in
  - Weather Forecasting
  - Study of Temperature and its effects
  - Temperature Prediction



# DRAW BACKS

- ❖ This project can't handle huge data at same time.
  - ❖ Required more time for predict.
- ❖ This project is only for study of prediction not for weather cast.

# CONCLUSION

- ❖ According to the records the global average temperature is  $13.8\text{ C}^0$  before 1700 and the global average temperature is  $15\text{ C}^0$  in 2016.
- ❖ In the pandemic situation 50% of car transport, 75% of air transport and 35% industrials works was stop for few months due to this lockdown 17% of  $\text{CO}_2$  decreased.
- ❖ When we get the predicted data the value are nearly  $15\text{ C}^0 - 16\text{ C}^0$ .
- ❖ I conclude that the temperature was increasing it leads to danger to life on earth.
- ❖ I also conclude that the Greenhouse effect and its effect theory given by “Irish physicist John Tyndall” is true.

# THANK YOU

BY

PAPPU SRI VINAY TEJA

(170030975)