

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
```

```
[2] from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
[6] df=pd.read_csv("/content/drive/MyDrive/abalone.csv")
```

```
[7] df
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.1500	15
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.0700	7
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.2100	9
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.1550	10
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.0550	7
...	...	...	...	...	...	...	...	...	...
1172	F	0.555	0.450	0.165	0.8070	0.2700	0.2200	0.2100	11

0s completed at 4:47 PM



Search



ENG  
IN



04:48 PM  
02-11-2022



+ Code + Text

RAM Disk Editing

[7] df

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.1500	15
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.0700	7
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.2100	9
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.1550	10
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.0550	7
...	...	...	...	...	...	...	...	...	...
4172	F	0.565	0.450	0.165	0.8870	0.3700	0.2390	0.2490	11
4173	M	0.590	0.440	0.135	0.9660	0.4390	0.2145	0.2605	10
4174	M	0.600	0.475	0.205	1.1760	0.5255	0.2875	0.3080	9
4175	F	0.625	0.485	0.150	1.0945	0.5310	0.2610	0.2960	10
4176	M	0.710	0.555	0.195	1.9485	0.9455	0.3765	0.4950	12

4177 rows x 9 columns

```
[8] df['age'] = df['Rings']+1.5  
df = df.drop('Rings', axis = 1)
```

```
[9] df.hist(figsize=(20,10), grid=False, layout=(2, 4), bins = 30)
```

0s completed at 4:47 PM



Search



ENG  
IN



04:48 PM  
02-11-2022



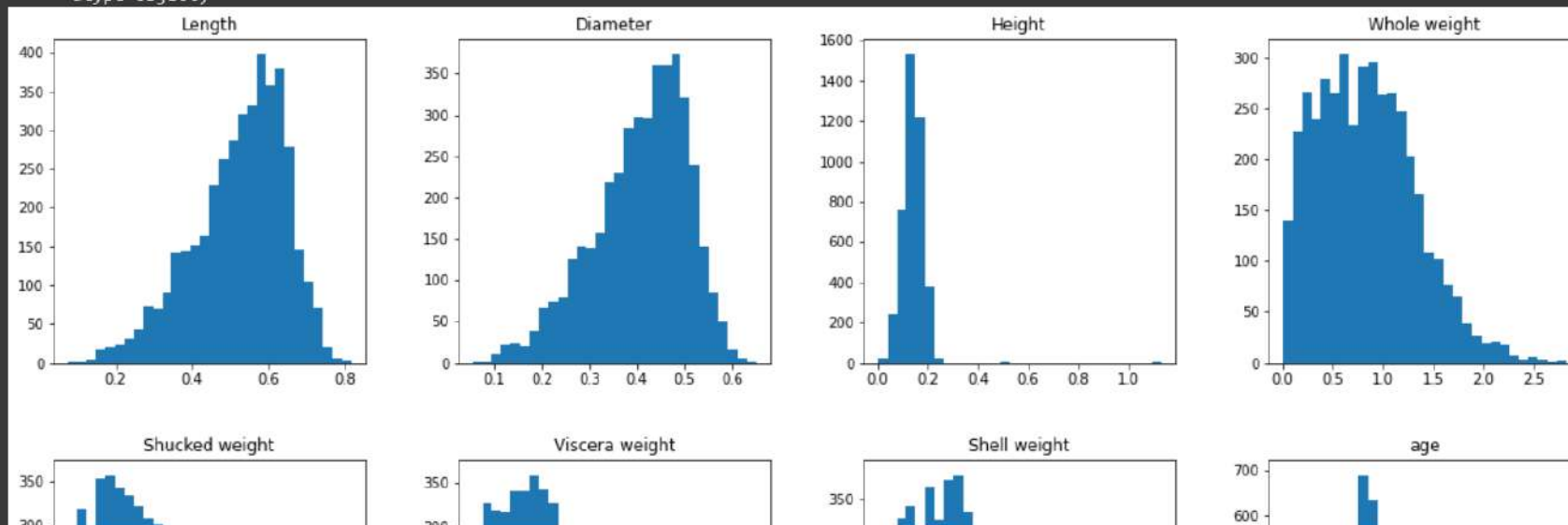
+ Code + Text

RAM | Disk

Editing

```
df.hist(figsize=(20,10), grid=False, layout=(2, 4), bins = 30)
```

```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fb713862c50>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb71272eb90>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb7126751d0>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb7126aa7d0>],  
[<matplotlib.axes._subplots.AxesSubplot object at 0x7fb712661dd0>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb712625410>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb7125dba90>,  
<matplotlib.axes._subplots.AxesSubplot object at 0x7fb712592fd0>]],  
dtype=object)
```



0s completed at 4:47 PM



Search

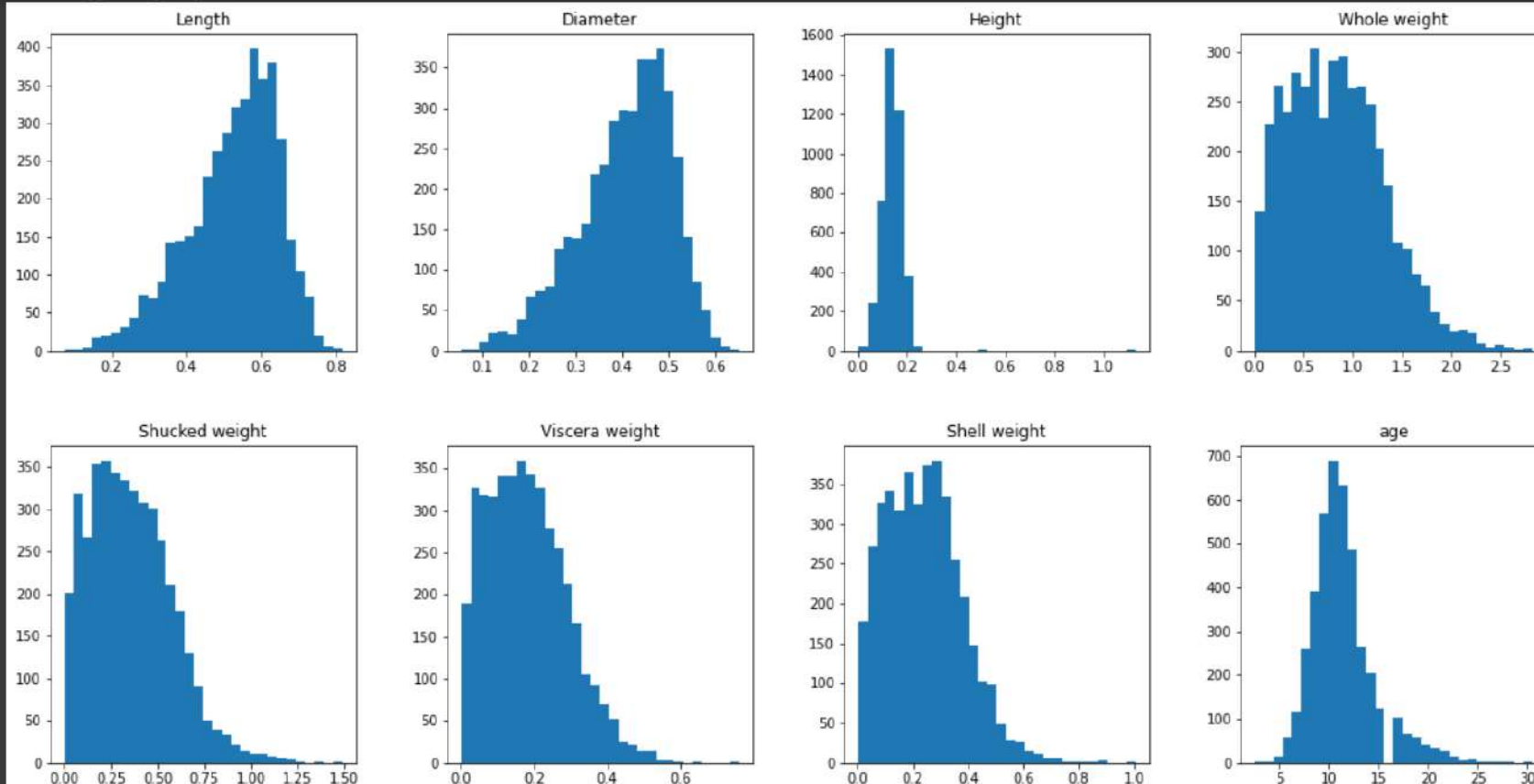


ENG  
IN



04:48 PM  
02-11-2022





0s completed at 4:47 PM



Search



ENG  
IN



04:48 PM  
02-11-2022





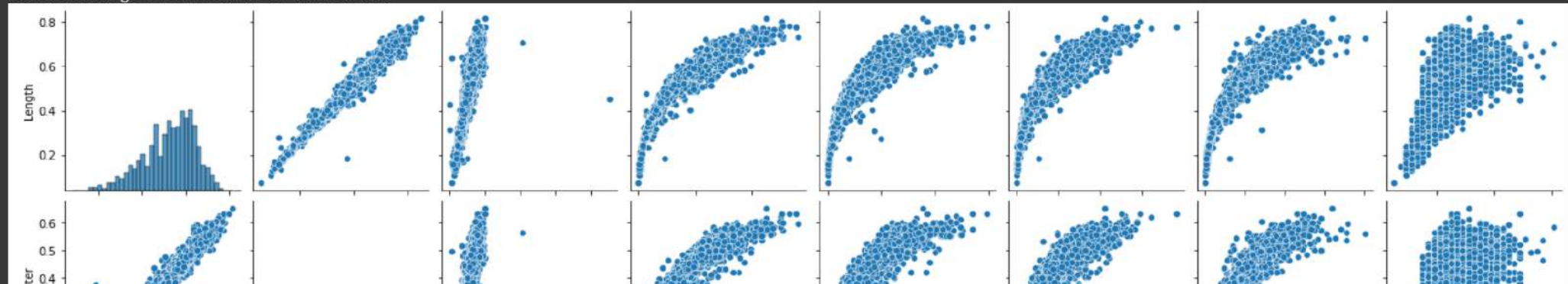
+ Code + Text

```
df.groupby('Sex')[['Length', 'Diameter', 'Height', 'Whole weight', 'Shucked weight', 'Viscera weight', 'Shell weight', 'age']].mean().sort_values('age')
```

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	age
Sex								
I	0.427746	0.326494	0.107996	0.431363	0.191035	0.092010	0.128182	9.390462
M	0.561391	0.439287	0.151381	0.991459	0.432946	0.215545	0.281969	12.205497
F	0.579093	0.454732	0.158011	1.046532	0.446188	0.230689	0.302010	12.629304

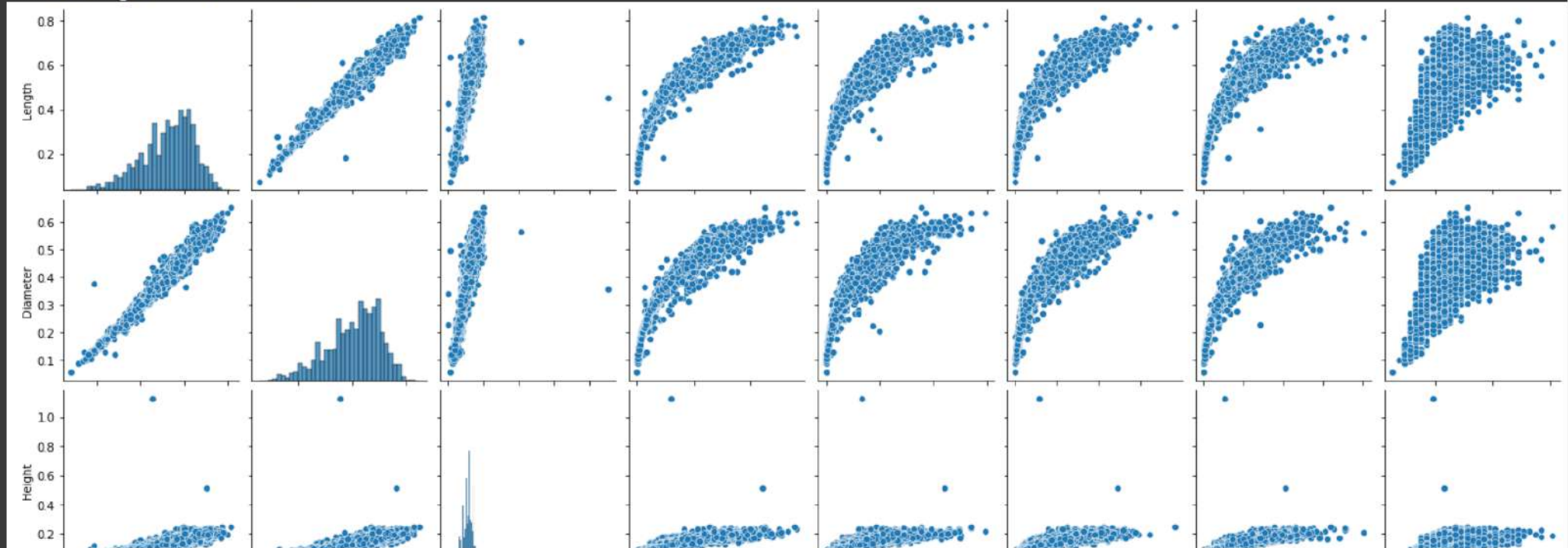
```
[11] numerical_features = df.select_dtypes(include = [np.number]).columns  
sns.pairplot(df[numerical_features])
```

<seaborn.axisgrid.PairGrid at 0x7fb7122b9c10>



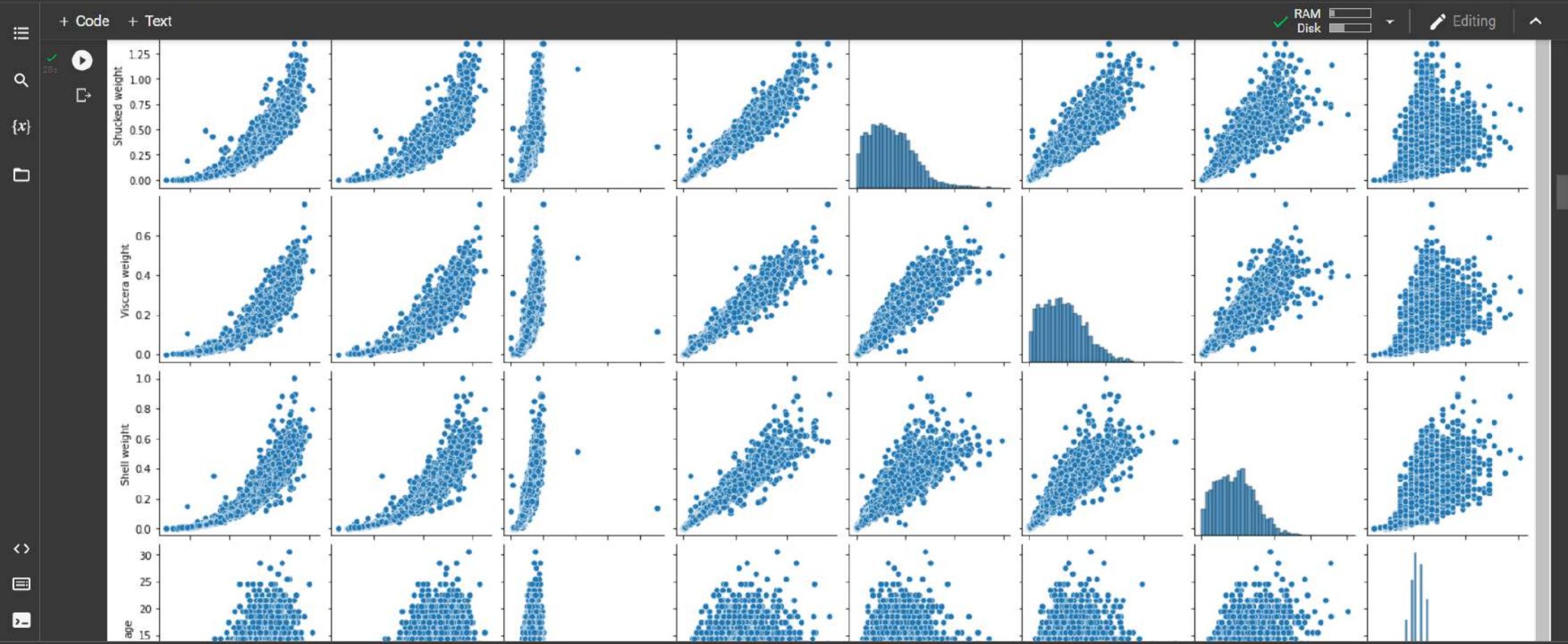
0s completed at 4:47 PM

```
[11] numerical_features = df.select_dtypes(include = [np.number]).columns
sns.pairplot(df[numerical_features])
<seaborn.axisgrid.PairGrid at 0x7fb7122b9c10>
```



0s completed at 4:47 PM





0s completed at 4:47 PM

+ Code + Text

df.describe()

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	age
count	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000
mean	0.523992	0.407881	0.139516	0.828742	0.359367	0.180594	0.238831	11.433684
std	0.120093	0.099240	0.041827	0.490389	0.221963	0.109614	0.139203	3.224169
min	0.075000	0.055000	0.000000	0.002000	0.001000	0.000500	0.001500	2.500000
25%	0.450000	0.350000	0.115000	0.441500	0.186000	0.093500	0.130000	9.500000
50%	0.545000	0.425000	0.140000	0.799500	0.336000	0.171000	0.234000	10.500000
75%	0.615000	0.480000	0.165000	1.153000	0.502000	0.253000	0.329000	12.500000
max	0.815000	0.650000	1.130000	2.825500	1.488000	0.760000	1.005000	30.500000

[13] df.isnull().sum()

```
Sex          0
Length       0
Diameter     0
Height       0
Whole weight 0
Shucked weight
Viscera weight
Shell weight 0
age          0
dtype: int64
```

0s completed at 4:47 PM



Search



ENG  
IN



04:49 PM  
02-11-2022

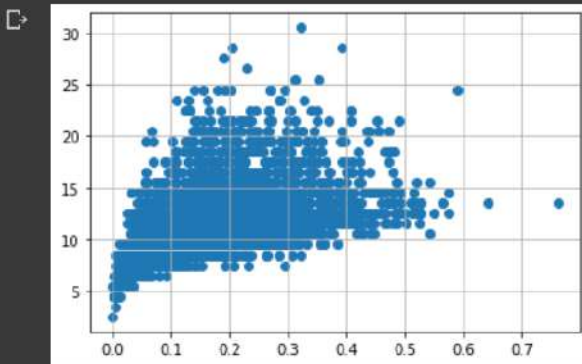




+ Code + Text

```
[14] df = pd.get_dummies(df)
     dummy_data = df.copy()
```

```
var = 'Viscera weight'
plt.scatter(x = df[var], y = df['age'],)
plt.grid(True)
```



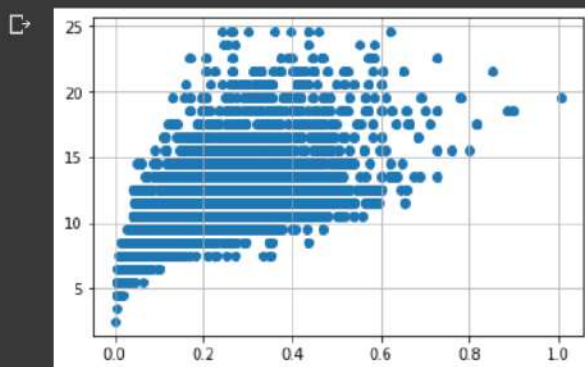
```
[16] df.drop(df[(df['Viscera weight'] > 0.5) & (df['age'] < 20)].index, inplace=True)
     df.drop(df[(df['Viscera weight'] < 0.5) & (df['age'] > 25)].index, inplace=True)
```

```
[17] var = 'Shell weight'
     plt.scatter(x = df[var], y = df['age'],)
     plt.grid(True)
     #Outliers removal
```

0s completed at 4:47 PM

+ Code + Text

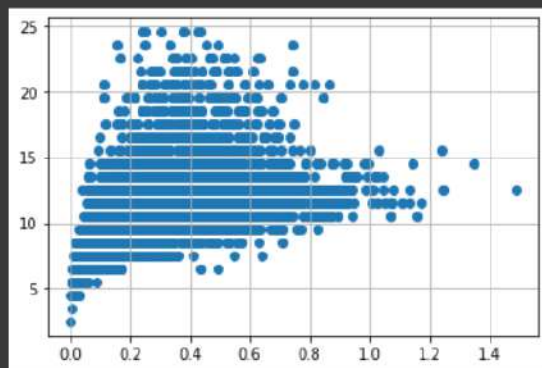
```
var = 'Shell weight'
plt.scatter(x = df[var], y = df['age'],)
plt.grid(True)
#Outliers removal
df.drop(df[(df['Shell weight'] > 0.6) & (df['age'] < 25)].index, inplace=True)
df.drop(df[(df['Shell weight'] < 0.8) & (df['age'] > 25)].index, inplace=True)
```



```
[18] var = 'Shucked weight'
plt.scatter(x = df[var], y = df['age'],)
plt.grid(True)

#Outlier removal
df.drop(df[(df['shucked weight'] >= 1) & (df['age'] < 20)].index, inplace=True)
df.drop(df[(df['Shucked weight'] < 1) & (df['age'] > 20)].index, inplace=True)
```

0s completed at 4:47 PM



```
[19] var = 'Whole weight'
plt.scatter(x = df[var], y = df['age'])
plt.grid(True)

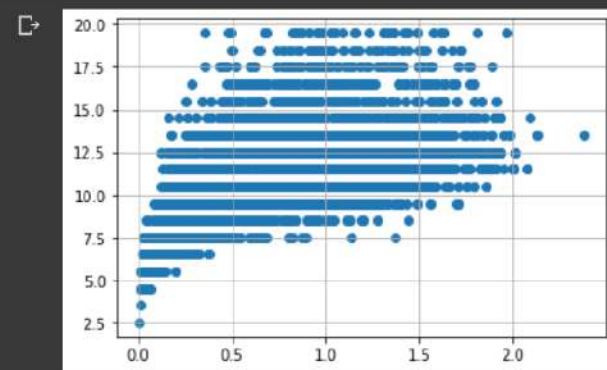
df.drop(df[(df['Whole weight'] >= 2.5) &
          (df['age'] < 25)].index, inplace = True)
df.drop(df[(df['Whole weight'] < 2.5) &
          (df['age'] > 25)].index, inplace = True)
```



0s completed at 4:47 PM



```
(df['age'] < 25).index, inplace = True)  
df.drop(df[(df['Whole weight'] < 2.5) & (  
df['age'] > 25)].index, inplace = True)
```

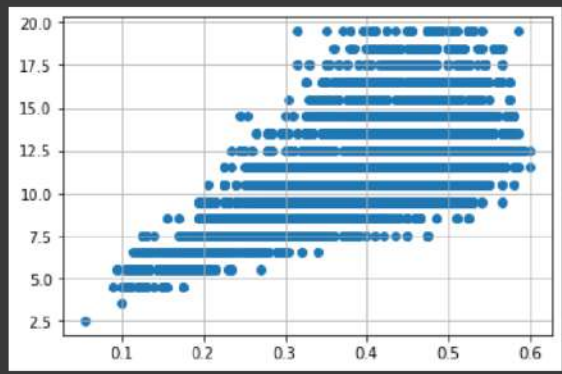


```
[20] var = 'Diameter'  
plt.scatter(x = df[var], y = df['age'])  
plt.grid(True)  
  
df.drop(df[(df['Diameter'] < 0.1) &  
(df['age'] < 5)].index, inplace = True)  
df.drop(df[(df['Diameter'] < 0.6) & (  
df['age'] > 25)].index, inplace = True)  
df.drop(df[(df['Diameter'] >= 0.6) & (  
df['age'] < 25)].index, inplace = True)
```



0s completed at 4:47 PM

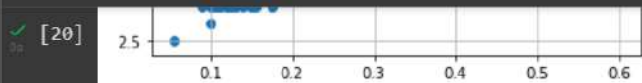
```
+ Code + Text
(df['age'] < 5).index, inplace = True
df.drop(df[(df['Diameter'] < 0.6) & (
df['age'] > 25)].index, inplace = True)
df.drop(df[(df['Diameter'] >= 0.6) & (
df['age'] < 25)].index, inplace = True)
```



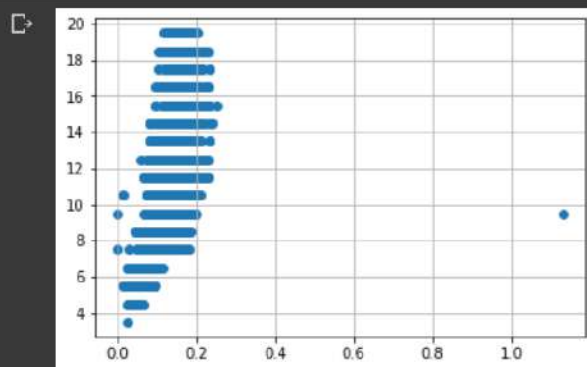
```
[21] var = 'Height'
plt.scatter(x = df[var], y = df['age'])
plt.grid(True)
df.drop(df[(df['Height'] > 0.4) &
(df['age'] < 15)].index, inplace = True)
df.drop(df[(df['Height'] < 0.4) & (
df['age'] > 25)].index, inplace = True)
```



+ Code + Text



```
var = 'Height'
plt.scatter(x = df[var], y = df['age'])
plt.grid(True)
df.drop(df[(df['Height'] > 0.4) &
          (df['age'] < 15)].index, inplace = True)
df.drop(df[(df['Height'] < 0.4) &
          (df['age'] > 25)].index, inplace = True)
```



```
[22] var = 'Length'
plt.scatter(x = df[var], y = df['age'])
plt.grid(True)
```

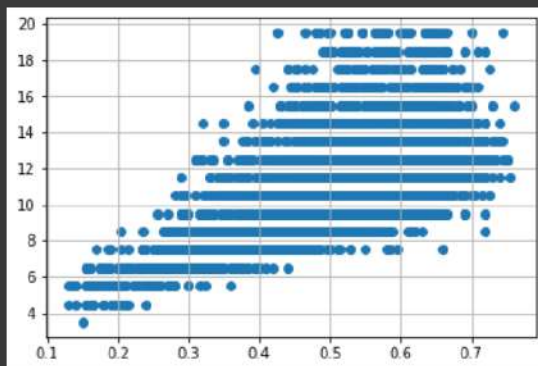
0s completed at 4:47 PM



+ Code + Text

```
[22] var = 'Length'
plt.scatter(x = df[var], y = df['age'])
plt.grid(True)

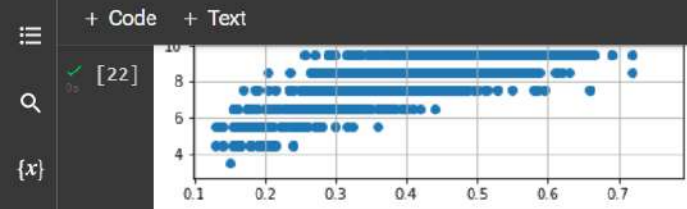
df.drop(df[(df['Length'] < 0.1) &
          (df['age'] < 5)].index, inplace = True)
df.drop(df[(df['Length'] < 0.8) & (
df['age'] > 25)].index, inplace = True)
df.drop(df[(df['Length'] > 0.8) & (
df['age'] < 25)].index, inplace = True)
```



```
[23] numerical_features = df.select_dtypes(include = [np.number]).columns
categorical_features = df.select_dtypes(include = [np.object]).columns
```

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:2: DeprecationWarning: `np.object` is a deprecated alias for the builtin `object`. To silence this warning, use `object` by deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>

0s completed at 4:47 PM



[23]

```
numerical_features = df.select_dtypes(include = [np.number]).columns
categorical_features = df.select_dtypes(include = [np.object]).columns
```

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:2: DeprecationWarning: `np.object` is a deprecated alias for the builtin `object`. To silence this warning, use `object` by  
Deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>

[24]

```
numerical_features
```

Index(['Length', 'Diameter', 'Height', 'Whole weight', 'Shucked weight',  
 'Viscera weight', 'Shell weight', 'age', 'Sex\_F', 'Sex\_I', 'Sex\_M'],  
 dtype='object')

[25]

```
categorical_features
```

Index([], dtype='object')

[ ]

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
print(df.Length.value_counts())
```

0s completed at 4:47 PM

```
Index(['Length', 'Diameter', 'Height', 'Whole weight', 'Shucked weight',  
      'Viscera weight', 'Shell weight', 'age', 'Sex_F', 'Sex_I', 'Sex_M'],  
      dtype='object')
```

```
Index([], dtype='object')
```

```
[ ] from sklearn.preprocessing import LabelEncoder  
le=LabelEncoder()  
print(df.Length.value_counts())
```

```
0.575    93  
0.625    91  
0.580    89  
0.550    89  
0.620    83  
..  
0.220     2  
0.150     1  
0.755     1  
0.135     1  
0.760     1  
Name: Length, Length: 126, dtype: int64
```



+ Code + Text

```
0.760 1
Name: Length, Length: 126, dtype: int64
```

```
[26] x=df.iloc[:,5]
x
```

	Length	Diameter	Height	Whole weight	Shucked weight
0	0.455	0.365	0.095	0.5140	0.2245
1	0.350	0.265	0.090	0.2255	0.0995
2	0.530	0.420	0.135	0.6770	0.2565
3	0.440	0.365	0.125	0.5160	0.2155
4	0.330	0.255	0.080	0.2050	0.0895
...	...	...	...	...	...
4172	0.565	0.450	0.165	0.8870	0.3700
4173	0.590	0.440	0.135	0.9660	0.4390
4174	0.600	0.475	0.205	1.1760	0.5255
4175	0.625	0.485	0.150	1.0945	0.5310
4176	0.710	0.555	0.195	1.9485	0.9455

3995 rows x 5 columns

```
[27] y=df.iloc[:,5:]
y
```

0s completed at 4:47 PM



Search



ENG  
IN



04:50 PM  
02-11-2022



+ Code + Text

[27] y=df.iloc[:,5:]  
y

	Viscera weight	Shell weight	age	Sex_F	Sex_I	Sex_M
0	0.1010	0.1500	16.5	0	0	1
1	0.0485	0.0700	8.5	0	0	1
2	0.1415	0.2100	10.5	1	0	0
3	0.1140	0.1550	11.5	0	0	1
4	0.0395	0.0550	8.5	0	1	0
...	...	...	...	...	...	...
4172	0.2390	0.2490	12.5	1	0	0
4173	0.2145	0.2605	11.5	0	0	1
4174	0.2875	0.3080	10.5	0	0	1
4175	0.2610	0.2960	11.5	1	0	0
4176	0.3765	0.4950	13.5	0	0	1

3995 rows x 6 columns

[28] from sklearn.model\_selection import train\_test\_split  
x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2)

[29] from sklearn.linear\_model import LinearRegression

0s completed at 4:47 PM



Search



ENG  
IN



04:50 PM  
02-11-2022



+ Code + Text

```
[28] from sklearn.model_selection import train_test_split  
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
```

```
[29] from sklearn.linear_model import LinearRegression  
mlr=LinearRegression()  
mlr.fit(x_train,y_train)  
  
LinearRegression()
```

```
[30] x_test[0:5]
```

	Length	Diameter	Height	Whole weight	Shucked weight
1809	0.640	0.475	0.140	1.0725	0.4895
2506	0.375	0.300	0.075	0.1440	0.0590
2369	0.560	0.440	0.170	0.9445	0.3545
3577	0.600	0.475	0.230	1.1570	0.5220
2869	0.370	0.280	0.085	0.1980	0.0805

```
[31] y_test[0:5]
```

	Viscera weight	Shell weight	age	Sex_F	Sex_I	Sex_M
1809	0.2295	0.310	9.5	1	0	0

0s completed at 4:47 PM



	Viscera weight	Shell weight	age	Sex_F	Sex_I	Sex_M
1809	0.2295	0.310	9.5	1	0	0
2506	0.0300	0.044	8.5	0	1	0
2369	0.2175	0.300	13.5	0	1	0
3577	0.2235	0.360	12.5	0	0	1
2869	0.0455	0.058	6.5	0	1	0

```
[32] from sklearn.preprocessing import StandardScaler
      ss=StandardScaler()
      x_train=ss.fit_transform(x_train)
```

Assignment 4 - 727619BIT026.ipynb

IBM-Project-2330-1658469808/

colab.research.google.com/drive/1tyF8xveNnBdyRUymPxRg-GxIGR3TVnd0#scrollTo=cElac1TtwVyt

Groww - Online De...YouTubeTamilBlasters | Lates...DriveOutlook

Assignment 4 - 727619BIT026.ipynb

File Edit View Insert Runtime Tools Help All changes saved

CommentShare

RAMDisk

Editing

+ Code + Text

mlrpred

array([[ 0.23724206, 0.28762719, 11.33779331, 0.32597418, 0.263786 ,  
 0.41023982],  
 [ 0.02800235, 0.05602483, 8.95920642, 0.09936378, 0.71899922,  
 0.18163701],  
 [ 0.21233191, 0.29619422, 13.23170802, 0.44369084, 0.14988261,  
 0.40642655],  
 [ 0.25581777, 0.34225978, 13.56715107, 0.52538667, -0.04653311,  
 0.52114644],  
 [ 0.04284146, 0.06607175, 8.9115833 , 0.09880874, 0.70608811,  
 0.19510315],  
 [ 0.12993526, 0.18904414, 11.3359012 , 0.29656264, 0.38645424,  
 0.31698312],  
 [ 0.24608307, 0.3097537 , 11.87958685, 0.38981072, 0.16508757,  
 0.44510171],  
 [ 0.33539203, 0.41945829, 13.66126541, 0.54167171, -0.10682847,  
 0.56515676],  
 [ 0.16836738, 0.22441874, 11.35510508, 0.30963623, 0.33582888,  
 0.35453489]])

from sklearn.metrics import r2\_score  
r2\_score(mlr.predict(x\_test),y\_test)

-3.2397564414374376

0s completed at 4:47 PM

ENG IN 04:50 PM 02-11-2022