Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] import numpy as np
    import pandas as pd
```

```
[ ] # basis of tem find pressure
```

```
df=pd.read_csv("/content/polynomial.csv")
df
```

	sno	Temperature	Pressure
0	1	0	0.0002
1	2	20	0.0012
2	3	40	0.0060
3	4	60	0.0300
4	5	80	0.0900
5	6	100	0.2700

```
[ ] df.columns
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] df.columns
```

```
Index(['sno', 'Temperature', 'Pressure'], dtype='object')
```

```
[ ] df.head()
```

	sno	Temperature	Pressure
0	1	0	0.0002
1	2	20	0.0012
2	3	40	0.0060
3	4	60	0.0300
4	5	80	0.0900

```
df.tail()
```

	sno	Temperature	Pressure
1	2	20	0.0012
2	3	40	0.0060

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] df.isna().sum()
```

```
sno            0
Temperature    0
Pressure       0
dtype: int64
```

```
[ ] df1=df.drop(['sno'],axis=1)
```

```
df1
```

	Temperature	Pressure
0	0	0.0002
1	20	0.0012
2	40	0.0060
3	60	0.0300
4	80	0.0900
5	100	0.2700

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] x=df1.iloc[:,:-1].values
    x
```

```
array([[  0],
       [ 20],
       [ 40],
       [ 60],
       [ 80],
       [100]])
```

```
[ ] y=df1.iloc[:,-1].values
    y
```

```
array([2.0e-04, 1.2e-03, 6.0e-03, 3.0e-02, 9.0e-02, 2.7e-01])
```

```
import matplotlib.pyplot as plt
plt.scatter(x,y)
```

```
<matplotlib.collections.PathCollection at 0x7f331113f5b0>
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search

ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

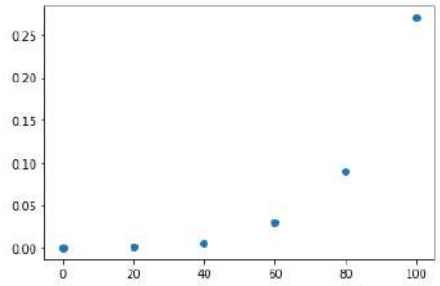
Comment Share

+ Code + Text

RAM Disk Editing

```
import matplotlib.pyplot as plt
plt.scatter(x,y)
```

```
<matplotlib.collections.PathCollection at 0x7f331113f5b0>
```




```
[ ] from sklearn.linear_model import LinearRegression
    model=LinearRegression()
    model.fit(x,y)
    y_pred=model.predict(x)
    y_pred
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search

ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

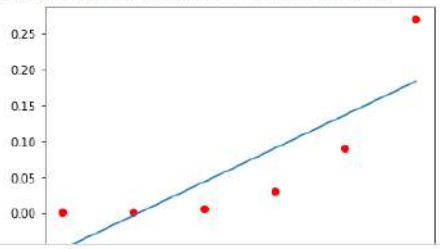
RAM Disk Editing

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x,y)
y_pred=model.predict(x)
y_pred
```

```
array([-0.05086667, -0.00402667,  0.04281333,  0.08965333,  0.13649333,
        0.18333333])
```

```
[ ] plt.scatter(x,y,color='r')
    plt.plot(x,y_pred)
```

```
[<matplotlib.lines.Line2D at 0x7f33110f9700>]
```



Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

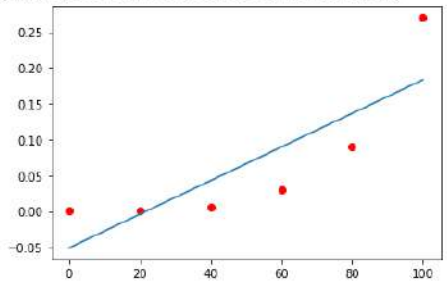
Comment Share

+ Code + Text

RAM Disk Editing

```
plt.scatter(x,y,color='r')
plt.plot(x,y_pred)
```

```
[<matplotlib.lines.Line2D at 0x7f33110f9700>]
```




```
[ ] print("slope is",model.coef_)
```

```
slope is [0.002342]
```

```
[ ] print("constant is",model.intercept_)
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] print("slope is",model.coef_)
```

```
slope is [0.002342]
```

```
[ ] print("constant is",model.intercept_)
```

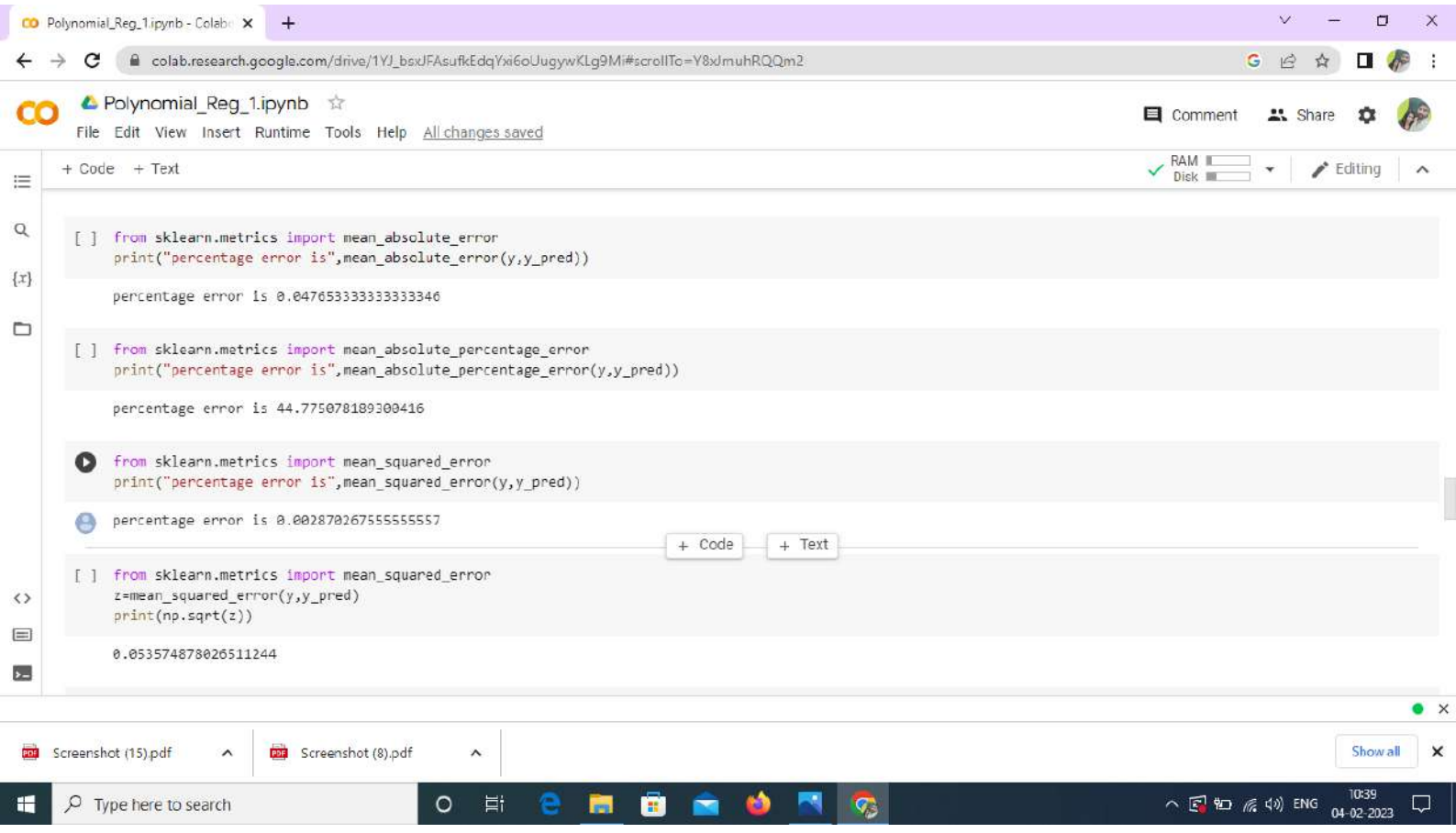
```
constant is -0.05086666666666667
```

```
df1=pd.DataFrame({"Actual value":y,"Predicted Value":y_pred})
df1
```

	Actual value	Predicted Value
0	0.0002	-0.050867
1	0.0012	-0.004027
2	0.0060	0.042813
3	0.0300	0.089653
4	0.0900	0.136493
5	0.2700	0.183333

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:39 04-02-2023


Polynomial_Reg_1.ipynb - Colab
colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2
Polynomial_Reg_1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
[ ] from sklearn.metrics import mean_absolute_error
print("percentage error is",mean_absolute_error(y,y_pred))
percentage error is 0.047653333333333346
[ ] from sklearn.metrics import mean_absolute_percentage_error
print("percentage error is",mean_absolute_percentage_error(y,y_pred))
percentage error is 44.775078189300416
from sklearn.metrics import mean_squared_error
print("percentage error is",mean_squared_error(y,y_pred))
percentage error is 0.002870267555555557
+ Code + Text
[ ] from sklearn.metrics import mean_squared_error
z=mean_squared_error(y,y_pred)
print(np.sqrt(z))
0.053574878026511244
Screenshot (15).pdf
Screenshot (8).pdf
Show all
Type here to search
ENG
10:39
04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] from sklearn.metrics import r2_score  # determints of coefficents
    print("R2_score is",r2_score(y,y_pred))
```

```
R2_score is 0.690349972603981
```

```
[ ] # Add features of x,
    from sklearn.preprocessing import PolynomialFeatures
    poly=PolynomialFeatures(degree=3)    # give degree to split x features
    x_poly=poly.fit_transform(x)
    x_poly
```

```
array([[1.00e+00, 0.00e+00, 0.00e+00, 0.00e+00],
       [1.00e+00, 2.00e+01, 4.00e+02, 8.00e+03],
       [1.00e+00, 4.00e+01, 1.60e+03, 6.40e+04],
       [1.00e+00, 6.00e+01, 3.60e+03, 2.16e+05],
       [1.00e+00, 8.00e+01, 6.40e+03, 5.12e+05],
       [1.00e+00, 1.00e+02, 1.00e+04, 1.00e+06]])
```

```
[ ] # to correct dimension fit x_poly and y first its in one dimension after split its convert to 2 dim so fit to correct dim
    poly.fit(x_poly,y)
    model1=LinearRegression()
    model1.fit(x_poly,y)
    y_poly=model1.predict(x_poly)    # predict value input in x_poly and y is otput value
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:40 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] # to correct dimension fit x_poly and y first its in one dimension after split its convert to 2 dim so fit to correct dim
    poly.fit(x_poly,y)
    model1=LinearRegression()
    model1.fit(x_poly,y)
    y_poly=model1.predict(x_poly)    # predict value input in x_poly and y is otput value
    y_poly
```

```
array([-0.00198889,  0.00724444,  0.00371111,  0.02248889,  0.09865556,
        0.26728889])
```

```
[ ] # plot ,
    plt.scatter(x,y,color='r')
    plt.plot(x,y_poly,color='g')
```

```
[<matplotlib.lines.Line2D at 0x7f330dfb0430>]
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:40 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

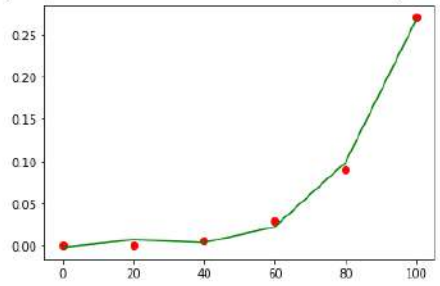
Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] # plot ,
    plt.scatter(x,y,color='r')
    plt.plot(x,y_poly,color='g')
```

```
[<matplotlib.lines.Line2D at 0x7f330dfb0430>]
```




```
[ ] df2=pd.DataFrame({"Actual value":y,"Predicted Value":y_pred,"polynomial_value":y_poly})
    df2
```

	Actual value	Predicted Value	polynomial_value

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:40 04-02-2023

Polynomial_Reg_1.ipynb - Colab

colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2

Polynomial_Reg_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

```
[ ] df2=pd.DataFrame({"Actual value":y,"Predicted Value":y_pred,"polynomial_value":y_poly})
    df2
```

	Actual value	Predicted Value	polynomial_value
0	0.0002	-0.050867	-0.001989
1	0.0012	-0.004027	0.007244
2	0.0060	0.042813	0.003711
3	0.0300	0.089653	0.022489
4	0.0900	0.136493	0.098656
5	0.2700	0.183333	0.267289

```
[ ] print("polynomial error percentage :",mean_absolute_percentage_error(y,y_poly))
```

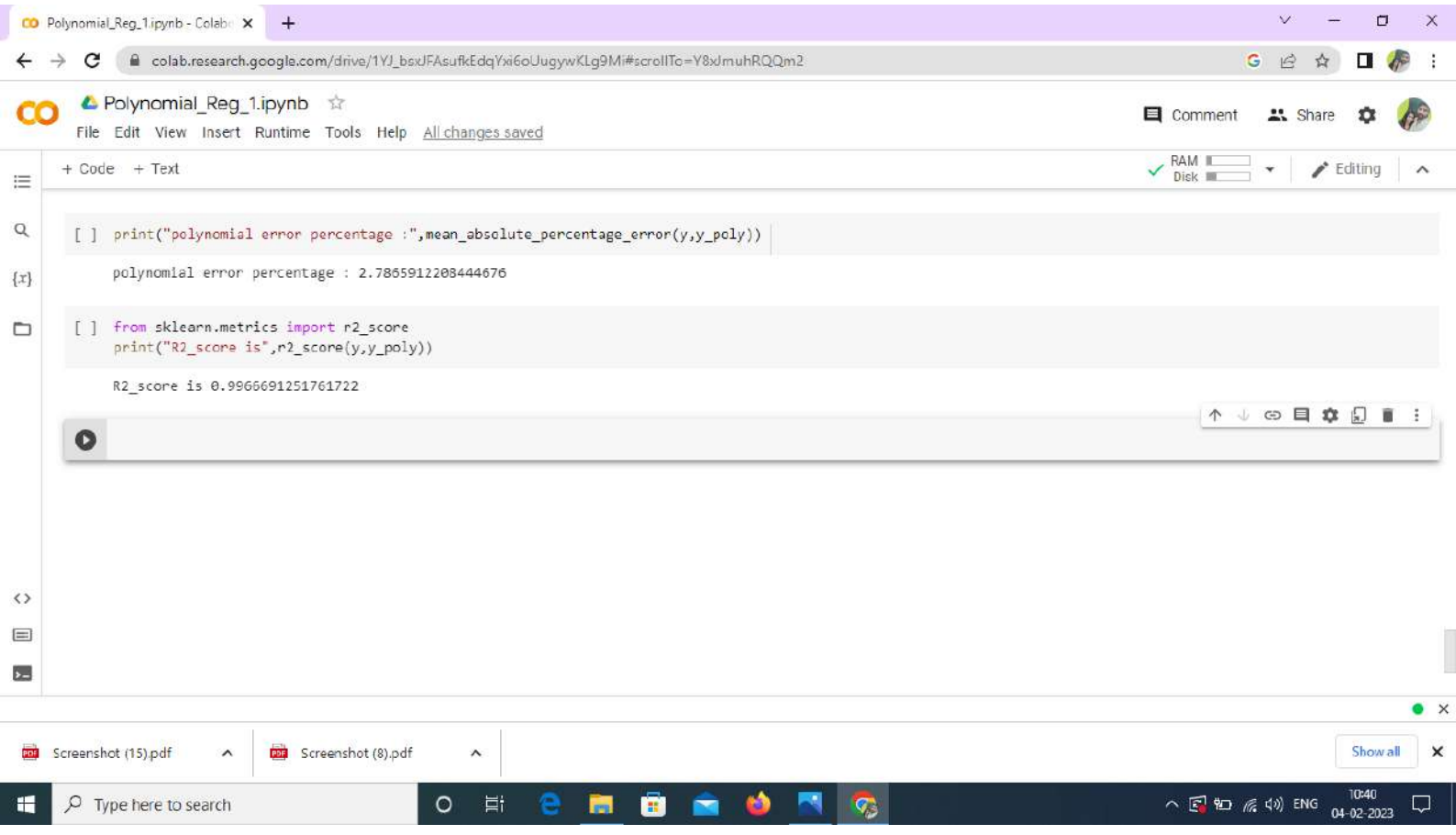
```
polynomial error percentage : 2.7865912208444676
```

```
[ ] from sklearn.metrics import r2_score
    print("R2_score is",r2_score(y,y_poly))
```

```
R2_score is 0.9966691251761722
```

Screenshot (15).pdf Screenshot (8).pdf Show all

Type here to search ENG 10:40 04-02-2023


Polynomial_Reg_1.ipynb - Colab
colab.research.google.com/drive/1YJ_bsxJFAsufkEdqYxi6oUugywKLg9Mi#scrollTo=Y8xJmuhRQQm2
Polynomial_Reg_1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
[ ] print("polynomial error percentage :",mean_absolute_percentage_error(y,y_poly))
polynomial error percentage : 2.7865912208444676
[ ] from sklearn.metrics import r2_score
print("R2_score is",r2_score(y,y_poly))
R2_score is 0.9966691251761722
Screenshot (15).pdf
Screenshot (8).pdf
Show all
Type here to search
ENG
10:40
04-02-2023