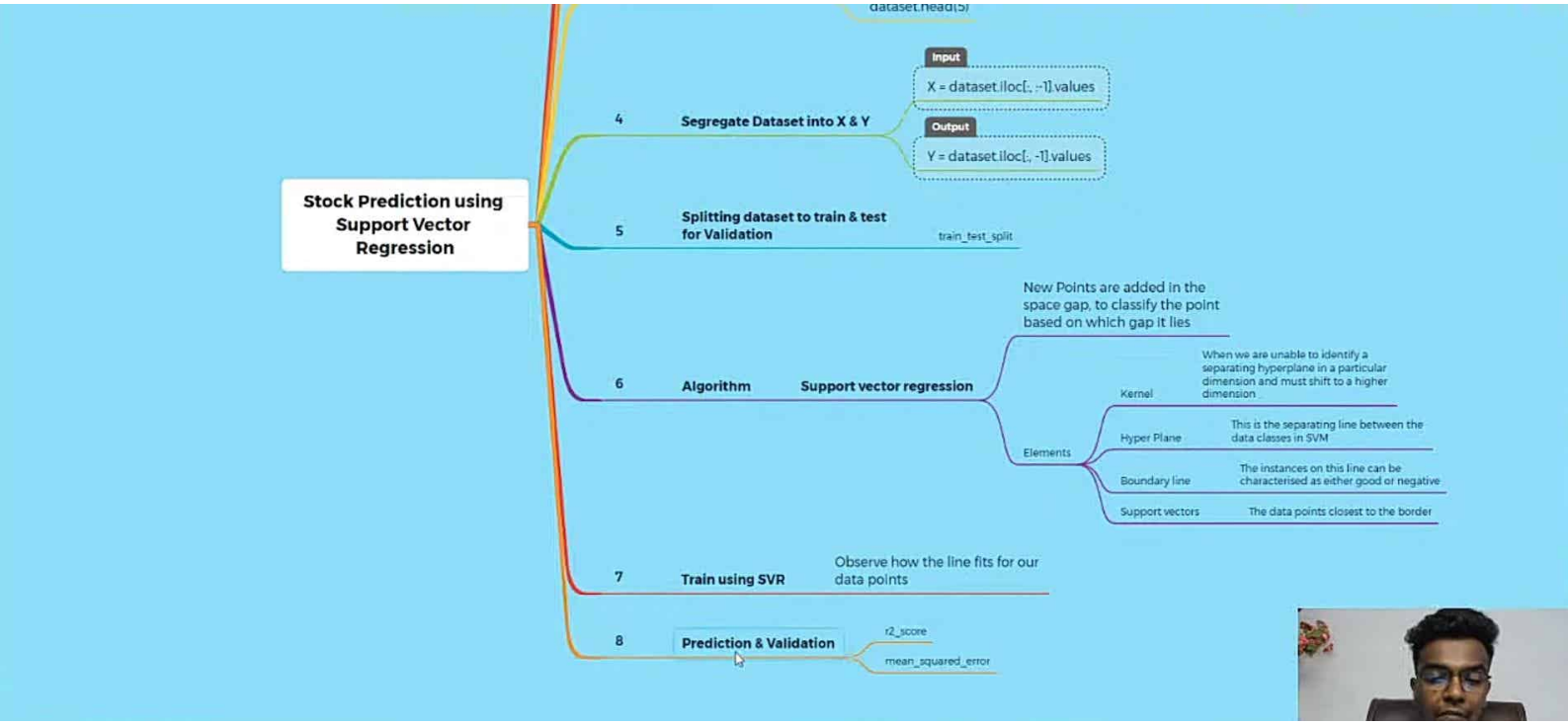
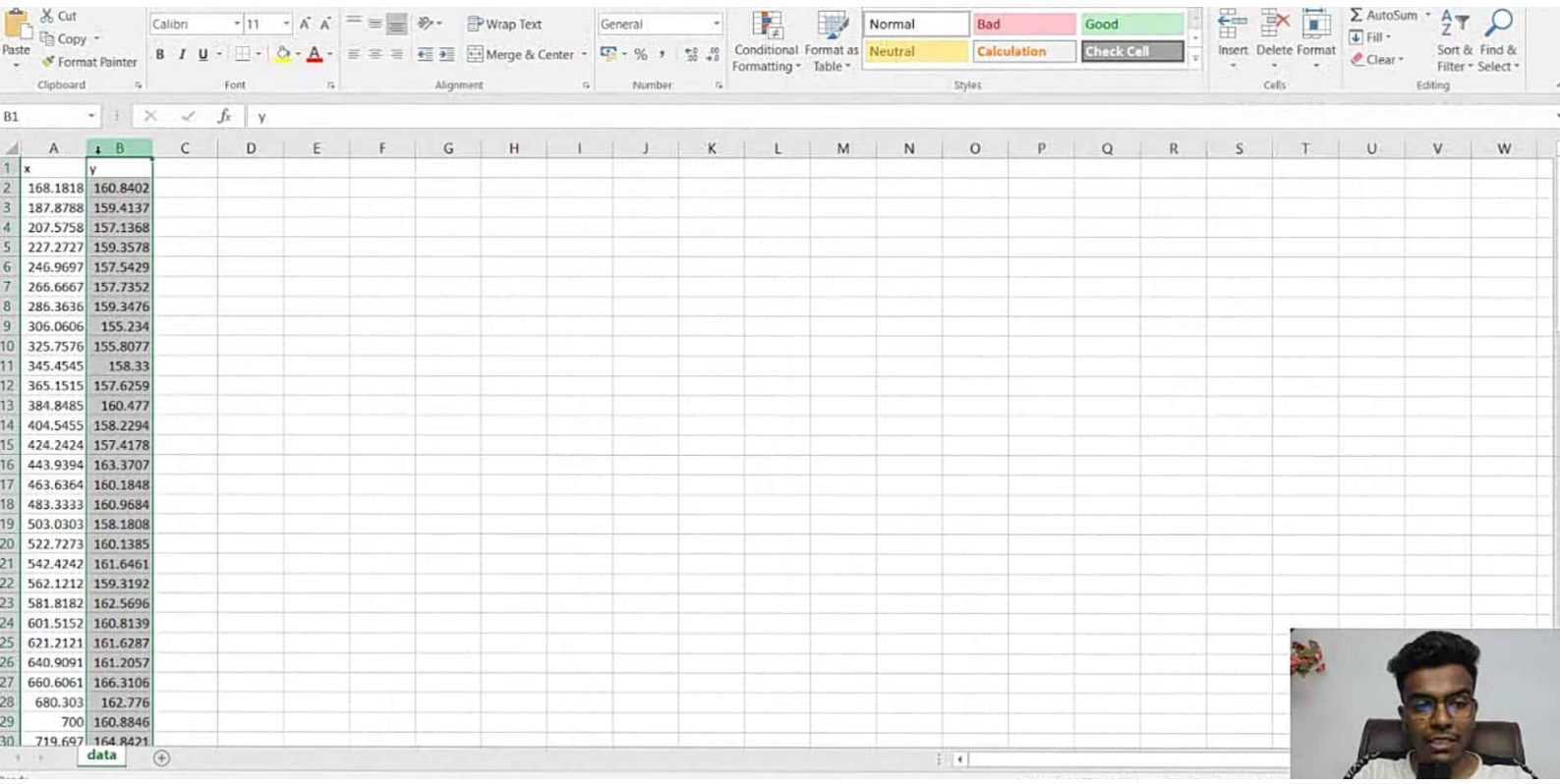

dataset.head(5)
Stock Prediction using Support Vector Regression
4
Segregate Dataset into X & Y
Input
X = dataset.iloc[:, :-1].values
Output
Y = dataset.iloc[:, -1].values
5
Splitting dataset to train & test for Validation
train_test_split
6
Algorithm
Support vector regression
New Points are added in the space gap, to classify the point based on which gap it lies
Elements
Kernel
When we are unable to identify a separating hyperplane in a particular dimension and must shift to a higher dimension
Hyper Plane
This is the separating line between the data classes in SVM
Boundary line
The instances on this line can be characterised as either good or negative
Support vectors
The data points closest to the border
7
Train using SVR
Observe how the line fits for our data points
8
Prediction & Validation
r2_score
mean_squared_error



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	x	y																					
2	168.1818	160.8402																					
3	187.8788	159.4137																					
4	207.5758	157.1368																					
5	227.2727	159.3578																					
6	246.9697	157.5429																					
7	266.6667	157.7352																					
8	286.3636	159.3476																					
9	306.0606	155.234																					
10	325.7576	155.8077																					
11	345.4545	158.33																					
12	365.1515	157.6259																					
13	384.8485	160.477																					
14	404.5455	158.2294																					
15	424.2424	157.4178																					
16	443.9394	163.3707																					
17	463.6364	160.1848																					
18	483.3333	160.9684																					
19	503.0303	158.1808																					
20	522.7273	160.1385																					
21	542.4242	161.6461																					
22	562.1212	159.3192																					
23	581.8182	162.5696																					
24	601.5152	160.8139																					
25	621.2121	161.6287																					
26	640.9091	161.2057																					
27	660.6061	166.3106																					
28	680.303	162.776																					
29	700	160.8846																					
30	719.697	164.8421																					

14_Stock prediction using SVM R

colab.research.google.com/drive/1NKGszyBHO2IT-w1iYX00xwAK-gesnhRu#scrollTo=1NTrKL3elgZ8

14_Stock prediction using SVM REGRESSION.ipynb

File Edit View Insert Runtime Tools Help Last edited on May 13

Comment Share

+ Code + Text

RAM Disk Editing

Day-14 | Stock Prediction using SUPPORT VECTOR REGRESSION

Importing Libraries

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

Load Dataset from Local directory

```
[ ] from google.colab import files
    uploaded = files.upload()
```

Load Dataset

```
[ ] dataset = pd.read_csv('data.csv')
```

Summarize Dataset

```
[ ] print(dataset.shape)
    print(dataset.head(5))
```

colab.research.google.com/drive/1NKGszyBHO2IT-w1iYX00xwAK-gesnhRu#scrollTo=63BR2xiKI7oZ

14_Stock prediction using SVM REGRESSION.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk Editing

```
          x           y
0  168.181818  160.840244
1  187.878788  159.413657
2  207.575758  157.136809
3  227.272727  159.357847
4  246.969697  157.542862
```

Segregate Dataset into Input X & Output Y

```
X = dataset.iloc[:, :-1].values
X
```

```
Y = dataset.iloc[:, -1].values
Y
```

Splitting Dataset for Testing our Model

0s completed at 7:13 PM

Type here to search

28°C Light rain

colab.research.google.com/drive/1NKGszyBHO2IT-w1iYX00xwAK-gesnhRu#scrollTo=Pc4kGEuDw36Q

14_Stock prediction using SVM REGRESSION.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

Splitting Dataset for Testing our Model

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size=0.20,random_state=0)
```

Training Dataset using Support Vector Regression

```
from sklearn.svm import SVR
model = SVR()
model.fit(x_train,y_train)
```

Prediction for all test data for validation

SSres is the sum of squares of the residual errors.

SStot is the total sum of the errors

0s completed at 7:14 PM

14_Stock prediction using SVM R | sklearn.svm import SVR - Google | sklearn.svm.SVR — scikit-learn

colab.research.google.com/drive/1NKGszyBHO2IT-w1iYX00xwAK-gesnhRu#scrollTo=4YOj1wVjerek

14_Stock prediction using SVM REGRESSION.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

SStot is the total sum of the errors.

$$R^2 = 1 - SS_{res} / SS_{tot}$$

```
ypred = model.predict(x_test)

from sklearn.metrics import r2_score,mean_squared_error
mse = mean_squared_error(y_test,ypred)
rmse=np.sqrt(mse)
print("Root Mean Square Error:",rmse)
r2score = r2_score(y_test,ypred)
print("R2Score",r2score*100)
```

```
Root Mean Square Error: 2.35947188444521
R2Score 86.64242653738361
```

0s completed at 7:24 PM