

Neural Networks & Deep Learning Assignment-4

Sai Vardhan Reddy Narra
700756163

Repository Link :

<https://github.com/saivardhan-dev/Assignment---4>

Video Link:

https://drive.google.com/file/d/1rQdsCqXeVF64DzesVKwY3MrO7tzXQIhO/view?usp=share_link

Code Screenshots:

Question - 1:

The first screenshot shows the initial code cell where the pandas library is imported and a CSV file is read into a DataFrame named 'df'. The second screenshot shows the output of 'df.describe()' and subsequent code cells for checking null values and calculating the mean of the 'Calories' column.

```
import pandas as pd
df = pd.read_csv('/content/data.csv')
```

	Duration	Pulse	Maxpulse	Calories
0	60	110	130	409.1
1	60	117	145	479.0
2	60	103	135	340.0
3	45	109	175	282.4
4	45	117	148	406.0
...
164	60	105	140	290.8
165	60	110	145	300.0
166	60	115	145	310.2
167	75	120	150	320.4
168	75	125	150	330.4

```
df.describe()
```

	Duration	Pulse	Maxpulse	Calories
count	169.000000	169.000000	169.000000	164.000000
mean	63.846154	107.461538	134.047337	375.790244
std	42.299949	14.510259	16.450434	266.379919
min	15.000000	80.000000	100.000000	50.300000
25%	45.000000	100.000000	124.000000	250.925000
50%	60.000000	105.000000	131.000000	318.600000
75%	60.000000	111.000000	141.000000	387.600000
max	300.000000	159.000000	184.000000	1860.400000

```
[9] df.isnull().sum()
```

```
Duration    0
Pulse       0
Maxpulse    0
Calories    5
dtype: int64
```

```
[10] calories_mean = df['Calories'].mean()
df['Calories'].fillna(calories_mean, inplace=True)
```

```
[11] df.isnull().sum()
```

```
Duration    0
Pulse       0
Maxpulse    0
Calories    0
dtype: int64
```

0s completed at 8:37 PM

Assignment - 4

File Edit View Insert Runtime Tools Help All changes saved

Comment Share S

+ Code + Text

RAM Disk

[12] df[['Calories', 'Pulse']].agg(['min', 'max', 'count', 'mean'])

	Calories	Pulse
min	50.300000	80.000000
max	1860.400000	159.000000
count	169.000000	169.000000
mean	375.790244	107.461538

df[(df['Calories'] >= 500) & (df['Calories'] <= 1000)]

	Duration	Pulse	Maxpulse	Calories
51	80	123	146	643.1
62	160	109	135	853.0
65	180	90	130	800.4
66	150	105	135	873.4
67	150	107	130	816.0
72	90	100	127	700.0
73	150	97	127	953.2
75	90	98	125	563.2
78	120	100	130	500.4
83	120	100	130	500.0
90	180	101	127	600.1
99	90	93	124	604.1
101	90	90	110	500.0

0s completed at 8:37 PM

Assignment - 4

File Edit View Insert Runtime Tools Help All changes saved

Comment Share S

+ Code + Text

RAM Disk

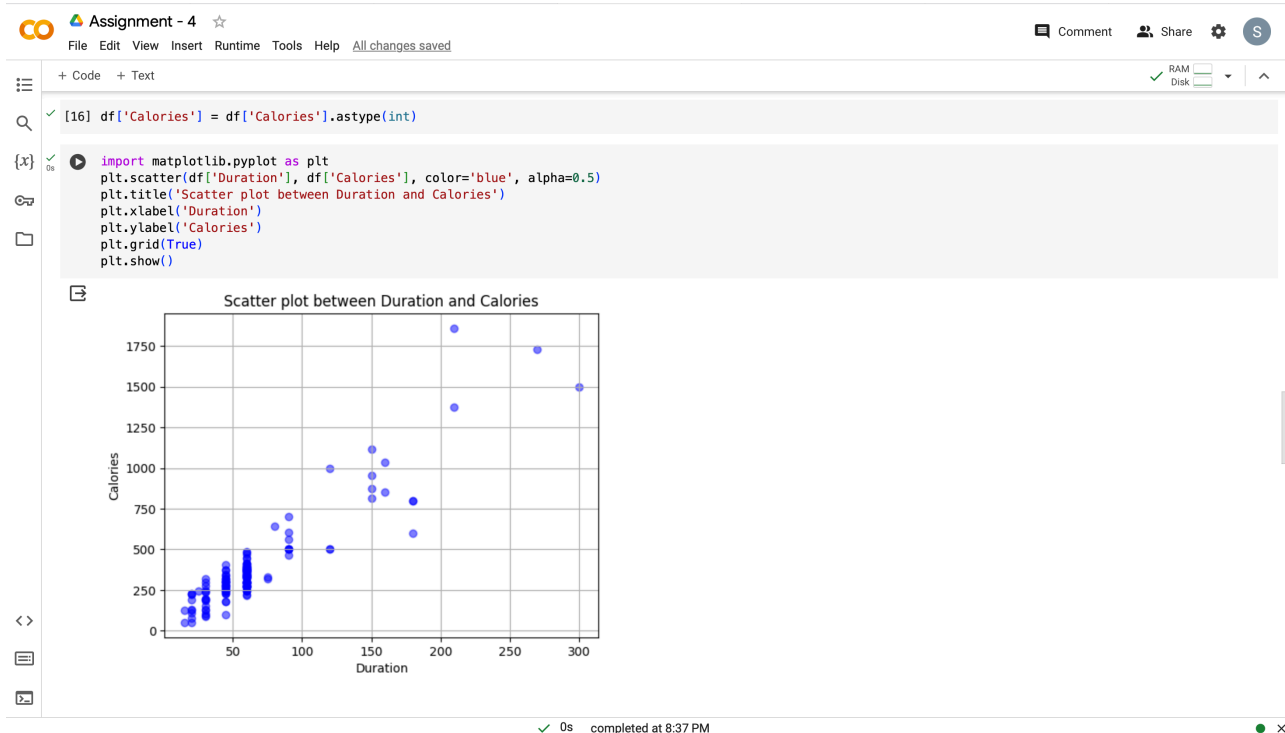
df[(df['Calories'] > 500) & (df['Pulse'] < 100)]

	Duration	Pulse	Maxpulse	Calories
65	180	90	130	800.4
70	150	97	129	1115.0
73	150	97	127	953.2
75	90	98	125	563.2
99	90	93	124	604.1
103	90	90	100	500.4
106	180	90	120	800.3
108	90	90	120	500.3

[15] df.drop(columns=['Maxpulse'])

	Duration	Pulse	Calories
0	60	110	409.1
1	60	117	479.0
2	60	103	340.0
3	45	109	282.4
4	45	117	406.0
...
164	60	105	290.8
165	60	110	300.0
166	60	115	310.2

0s completed at 8:37 PM



Question - 2:

Assignment - 4

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

2) Linear Regression

```
df2=pd.read_csv("/content/Salary_Data (2).csv")
print(df2)
```

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0
5	2.9	56642.0
6	3.0	60150.0
7	3.2	54445.0
8	3.2	64445.0
9	3.7	57189.0
10	3.9	63218.0
11	4.0	55794.0
12	4.0	56957.0
13	4.1	57081.0
14	4.5	61111.0
15	4.9	67938.0
16	5.1	66029.0
17	5.3	83088.0
18	5.9	81363.0
19	6.0	93940.0
20	6.8	91738.0
21	7.1	98273.0
22	7.9	101302.0
23	8.2	113812.0
24	8.7	109431.0
25	9.0	105582.0
26	9.5	116969.0
27	9.6	112635.0
28	10.3	122391.0
29	10.5	121872.0

0s completed at 8:37 PM

Assignment - 4

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM
Disk

0s completed at 8:37 PM

```
[28] from sklearn.model_selection import train_test_split

X = df2[['YearsExperience']]
y = df2['Salary']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=1/3, random_state=42)

[32] from sklearn.linear_model import LinearRegression

model = LinearRegression()
model.fit(X_train, y_train)
y_pred_train = model.predict(X_train)
y_pred_test = model.predict(X_test)

from sklearn.metrics import mean_squared_error

mse_train = mean_squared_error(y_train, y_pred_train)

mse_test = mean_squared_error(y_test, y_pred_test)

print("Mean Squared Error (Train):", mse_train)
print("Mean Squared Error (Test):", mse_test)

Mean Squared Error (Train): 29793161.082422983
Mean Squared Error (Test): 35301898.887134895

[30] plt.figure(figsize=(10, 6))
plt.scatter(X_train, y_train, color='blue', label='Training Data')
plt.scatter(X_test, y_test, color='red', label='Test Data')
plt.plot(X_train, y_pred_train, color='green', linewidth=2, label='Regression Line')

plt.title('Linear Regression - Salary Prediction')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.legend()
plt.show()
```

