dl-pract4

April 28, 2024

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

[4]:
```
train_df = pd.read_csv(r'/content/Google_Stock_Price_Train.csv')
```

[5]:
```
train_df
```

[5]:

	Date	Open	High	Low	Close	Volume
0	1/3/2012	325.25	332.83	324.97	663.59	7,380,500
1	1/4/2012	331.27	333.87	329.08	666.45	5,749,400
2	1/5/2012	329.83	330.75	326.89	657.21	6,590,300
3	1/6/2012	328.34	328.77	323.68	648.24	5,405,900
4	1/9/2012	322.04	322.29	309.46	620.76	11,688,800
…	…	…	…	…	…	…
1253	12/23/2016	790.90	792.74	787.28	789.91	623,400
1254	12/27/2016	790.68	797.86	787.66	791.55	789,100
1255	12/28/2016	793.70	794.23	783.20	785.05	1,153,800
1256	12/29/2016	783.33	785.93	778.92	782.79	744,300
1257	12/30/2016	782.75	782.78	770.41	771.82	1,770,000

[1258 rows x 6 columns]

[7]:
```
test_df = pd.read_csv(r'/content/Google_Stock_Price_Train.csv')
```

[8]:
```
test_df
```

[8]:

	Date	Open	High	Low	Close	Volume
0	1/3/2012	325.25	332.83	324.97	663.59	7,380,500
1	1/4/2012	331.27	333.87	329.08	666.45	5,749,400
2	1/5/2012	329.83	330.75	326.89	657.21	6,590,300
3	1/6/2012	328.34	328.77	323.68	648.24	5,405,900
4	1/9/2012	322.04	322.29	309.46	620.76	11,688,800
…	…	…	…	…	…	…
1253	12/23/2016	790.90	792.74	787.28	789.91	623,400
1254	12/27/2016	790.68	797.86	787.66	791.55	789,100
1255	12/28/2016	793.70	794.23	783.20	785.05	1,153,800
1256	12/29/2016	783.33	785.93	778.92	782.79	744,300

```
1257  12/30/2016  782.75  782.78  770.41  771.82   1,770,000

[1258 rows x 6 columns]
```

[9]:
```
test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1258 entries, 0 to 1257
Data columns (total 6 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Date    1258 non-null   object
 1   Open    1258 non-null   float64
 2   High    1258 non-null   float64
 3   Low     1258 non-null   float64
 4   Close   1258 non-null   object
 5   Volume  1258 non-null   object
dtypes: float64(3), object(3)
memory usage: 59.1+ KB
```

[12]:
```
from sklearn.preprocessing import MinMaxScaler
```

[16]:
```
train_df['Close'] = train_df['Close'].astype(str).str.replace(',', '').
 ↪astype(float)
test_df['Close'] = test_df['Close'].astype(str).str.replace(',', '').
 ↪astype(float)
```

[17]:
```
train_scaler = MinMaxScaler()
train_df['Normalized Close'] = train_scaler.fit_transform(train_df['Close'].
 ↪values.reshape(-1, 1))
test_scaler = MinMaxScaler()
```

[18]:
```
test_df['Normalized Close'] = test_scaler.fit_transform(test_df['Close'].values.
 ↪reshape(-1, 1))
```

[19]:
```
x_train = train_df['Normalized Close'].values[:-1].reshape(-1, 1, 1)
y_train = train_df['Normalized Close'].values[1:].reshape(-1, 1, 1)
x_test = test_df['Normalized Close'].values[:-1].reshape(-1, 1, 1)
y_test = test_df['Normalized Close'].values[1:].reshape(-1, 1, 1)
```

[20]:
```
print("x_train shape: ",x_train.shape)
print("y_train shape: ",y_train.shape)
print("x_test shape: ",x_test.shape)
print("y_test shape: ",y_test.shape)
```

```
x_train shape:  (1257, 1, 1)
y_train shape:  (1257, 1, 1)
```

```
x_test shape:  (1257, 1, 1)
y_test shape:  (1257, 1, 1)
```

[21]:
```
train_df
```

[21]:

	Date	Open	High	Low	Close	Volume	Normalized Close
0	1/3/2012	325.25	332.83	324.97	663.59	7,380,500	0.237573
1	1/4/2012	331.27	333.87	329.08	666.45	5,749,400	0.241514
2	1/5/2012	329.83	330.75	326.89	657.21	6,590,300	0.228781
3	1/6/2012	328.34	328.77	323.68	648.24	5,405,900	0.216419
4	1/9/2012	322.04	322.29	309.46	620.76	11,688,800	0.178548
…	…	…	…	…	…	…	…
1253	12/23/2016	790.90	792.74	787.28	789.91	623,400	0.411656
1254	12/27/2016	790.68	797.86	787.66	791.55	789,100	0.413916
1255	12/28/2016	793.70	794.23	783.20	785.05	1,153,800	0.404958
1256	12/29/2016	783.33	785.93	778.92	782.79	744,300	0.401844
1257	12/30/2016	782.75	782.78	770.41	771.82	1,770,000	0.386726

[1258 rows x 7 columns]

[22]:
```
test_df
```

[22]:

	Date	Open	High	Low	Close	Volume	Normalized Close
0	1/3/2012	325.25	332.83	324.97	663.59	7,380,500	0.237573
1	1/4/2012	331.27	333.87	329.08	666.45	5,749,400	0.241514
2	1/5/2012	329.83	330.75	326.89	657.21	6,590,300	0.228781
3	1/6/2012	328.34	328.77	323.68	648.24	5,405,900	0.216419
4	1/9/2012	322.04	322.29	309.46	620.76	11,688,800	0.178548
…	…	…	…	…	…	…	…
1253	12/23/2016	790.90	792.74	787.28	789.91	623,400	0.411656
1254	12/27/2016	790.68	797.86	787.66	791.55	789,100	0.413916
1255	12/28/2016	793.70	794.23	783.20	785.05	1,153,800	0.404958
1256	12/29/2016	783.33	785.93	778.92	782.79	744,300	0.401844
1257	12/30/2016	782.75	782.78	770.41	771.82	1,770,000	0.386726

[1258 rows x 7 columns]

[23]:
```
test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1258 entries, 0 to 1257
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Date              1258 non-null   object
 1   Open              1258 non-null   float64
 2   High              1258 non-null   float64
 3   Low               1258 non-null   float64
```

```
 4   Close             1258 non-null   float64
 5   Volume            1258 non-null   object
 6   Normalized Close  1258 non-null   float64
dtypes: float64(5), object(2)
memory usage: 68.9+ KB
```

[25]:
```
from keras.models import Sequential
from keras.layers import LSTM, Dense
```

[26]:
```
model = Sequential()
model.add(LSTM(4, input_shape=(1, 1)))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam')
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 4)                 96

 dense (Dense)               (None, 1)                 5

=================================================================
Total params: 101 (404.00 Byte)
Trainable params: 101 (404.00 Byte)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

[27]:
```
model.fit(x_train, y_train, epochs=100, batch_size=1, verbose=1)
```

```
Epoch 1/100
1257/1257 [==============================] - 5s 2ms/step - loss: 0.0250
Epoch 2/100
1257/1257 [==============================] - 3s 2ms/step - loss: 0.0017
Epoch 3/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5574e-04
Epoch 4/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7431e-04
Epoch 5/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7530e-04
Epoch 6/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7398e-04
Epoch 7/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5993e-04
Epoch 8/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7669e-04
Epoch 9/100
```

```
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6593e-04
Epoch 10/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6596e-04
Epoch 11/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7368e-04
Epoch 12/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.7436e-04
Epoch 13/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6967e-04
Epoch 14/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6237e-04
Epoch 15/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6393e-04
Epoch 16/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.7087e-04
Epoch 17/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7413e-04
Epoch 18/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6629e-04
Epoch 19/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6784e-04
Epoch 20/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.7091e-04
Epoch 21/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6281e-04
Epoch 22/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6903e-04
Epoch 23/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6263e-04
Epoch 24/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5757e-04
Epoch 25/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.6562e-04
Epoch 26/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6686e-04
Epoch 27/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5424e-04
Epoch 28/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6621e-04
Epoch 29/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6654e-04
Epoch 30/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6614e-04
Epoch 31/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6061e-04
Epoch 32/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5649e-04
Epoch 33/100
```

```
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6629e-04
Epoch 34/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5487e-04
Epoch 35/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6101e-04
Epoch 36/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5972e-04
Epoch 37/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.5733e-04
Epoch 38/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5549e-04
Epoch 39/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6984e-04
Epoch 40/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5748e-04
Epoch 41/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6502e-04
Epoch 42/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6460e-04
Epoch 43/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5276e-04
Epoch 44/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6617e-04
Epoch 45/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6877e-04
Epoch 46/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6096e-04
Epoch 47/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7063e-04
Epoch 48/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5719e-04
Epoch 49/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5703e-04
Epoch 50/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.7153e-04
Epoch 51/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6043e-04
Epoch 52/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6008e-04
Epoch 53/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5131e-04
Epoch 54/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5312e-04
Epoch 55/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5790e-04
Epoch 56/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4596e-04
Epoch 57/100
```

```
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4908e-04
Epoch 58/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5993e-04
Epoch 59/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5561e-04
Epoch 60/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6295e-04
Epoch 61/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6251e-04
Epoch 62/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5303e-04
Epoch 63/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.6223e-04
Epoch 64/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5812e-04
Epoch 65/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4801e-04
Epoch 66/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6322e-04
Epoch 67/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5792e-04
Epoch 68/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5602e-04
Epoch 69/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5855e-04
Epoch 70/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5920e-04
Epoch 71/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4988e-04
Epoch 72/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6343e-04
Epoch 73/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5231e-04
Epoch 74/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5950e-04
Epoch 75/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6691e-04
Epoch 76/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.5052e-04
Epoch 77/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6203e-04
Epoch 78/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5067e-04
Epoch 79/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5240e-04
Epoch 80/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5981e-04
Epoch 81/100
```

```
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5710e-04
Epoch 82/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5437e-04
Epoch 83/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6623e-04
Epoch 84/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4824e-04
Epoch 85/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.5950e-04
Epoch 86/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5955e-04
Epoch 87/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4386e-04
Epoch 88/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5977e-04
Epoch 89/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.7178e-04
Epoch 90/100
1257/1257 [==============================] - 3s 3ms/step - loss: 7.5677e-04
Epoch 91/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5005e-04
Epoch 92/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5277e-04
Epoch 93/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5692e-04
Epoch 94/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.6555e-04
Epoch 95/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5678e-04
Epoch 96/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.5798e-04
Epoch 97/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6566e-04
Epoch 98/100
1257/1257 [==============================] - 4s 3ms/step - loss: 7.5569e-04
Epoch 99/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.4972e-04
Epoch 100/100
1257/1257 [==============================] - 3s 2ms/step - loss: 7.6312e-04
```

[27]:
```
<keras.src.callbacks.History at 0x7f1183d03e80>
```

[28]:
```
test_loss = model.evaluate(x_test, y_test)
print('Testing loss: ', test_loss)
```

```
40/40 [==============================] - 1s 2ms/step - loss: 7.0773e-04
Testing loss:  0.0007077282061800361
```

[29]:
```
y_pred = model.predict(x_test)
```

```
40/40 [==============================] - 1s 2ms/step
```

[30]:
```
y_test_actual = test_scaler.inverse_transform(y_test.reshape(-1, 1))
y_pred_actual = test_scaler.inverse_transform(y_pred.reshape(-1, 1))
```

[31]:
```
i=1
```

[32]:
```
print("Actual value: {:.2f}".format(y_test_actual[i][0]))
print("Predicted value: {:.2f}".format(y_pred_actual[i][0]))
```

```
Actual value: 657.21
Predicted value: 664.78
```