Developing a Neural Network Regression Model

AIM

To develop a neural network regression model for the given dataset.

THEORY

Explain the problem statement

Neural Network Model

Include the neural network model diagram.

DESIGN STEPS

STEP 1:

Loading the dataset

STEP 2:

Split the dataset into training and testing

STEP 3:

Create MinMaxScalar objects ,fit the model and transform the data.

STEP 4:

Build the Neural Network Model and compile the model.

STEP 5:

Train the model with the training data.

STEP 6:

Plot the performance plot

STEP 7:

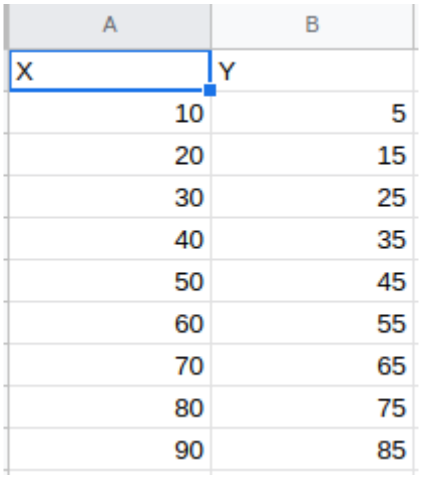
Evaluate the model with the testing data.

PROGRAM

~~~
```
from google.colab import auth
import gspread
from google.auth import default
import pandas as pd
auth.authenticate_user()
creds, _ = default()
gc = gspread.authorize(creds)
worksheet = gc.open('Ex01').sheet1
rows = worksheet.get_all_values()
df = pd.DataFrame(rows[1:], columns=rows[0])
df.head()
df.dtypes
df=df.astype({'A':'int'})
df=df.astype({'B':'float'})
df.dtypes
from sklearn.model_selection import train_test_split
X=df[['A']].values
Y=df[['B']].values
x_train,x_test,y_train,y_test=train_test_split(X,Y,test_size=0.33,random_state=20
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
scaler.fit(x_train)
x_train_scaled=scaler.transform(x_train)
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
ai_brain = Sequential([
    Dense(2,activation='relu'),
    Dense(1,activation='relu')
])
ai_brain.compile(optimizer='rmsprop',loss='mse')
ai_brain.fit(x=x_train_scaled,y=y_train,epochs=20000)
loss_df=pd.DataFrame(ai_brain.history.history)
~~~

~~~
loss_df.plot()
x_test_scaled=scaler.transform(x_test)
ai_brain.evaluate(x_test_scaled,y_test)
input=[[100]]
input_scaled=scaler.transform(input)
ai_brain.predict(input_scaled)
```
~~~

Dataset Information

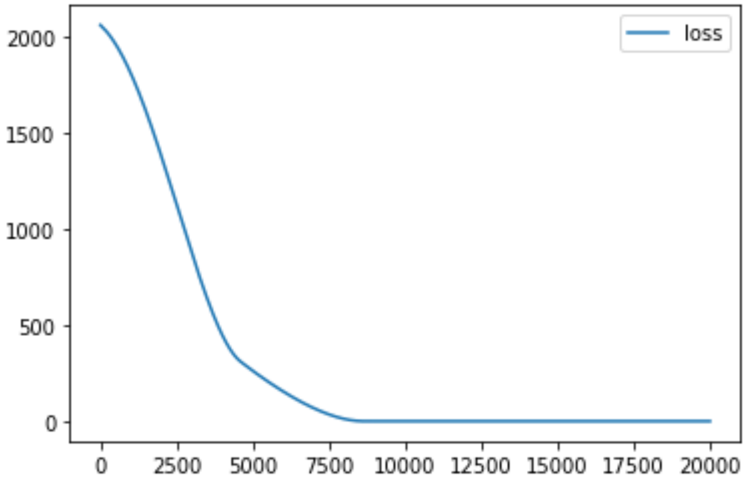


A	B
X	Y
10	5
20	15
30	25
40	35
50	45
60	55
70	65
80	75
90	85

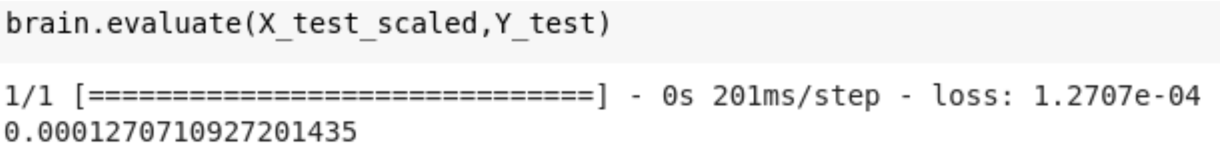
OUTPUT

Training Loss Vs Iteration Plot

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6320d813d0>
```



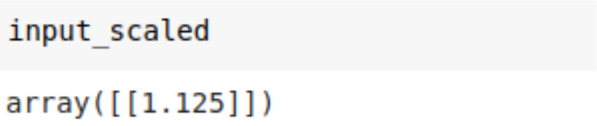
Test Data Root Mean Squared Error



```
brain.evaluate(X_test_scaled,Y_test)
```

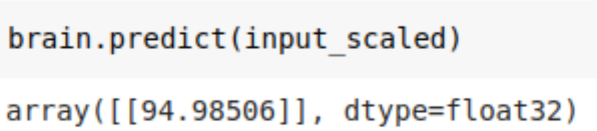
```
1/1 [==============================] - 0s 201ms/step - loss: 1.2707e-04
0.00012707109272014350
```

New Sample Data Prediction



```
input_scaled
```

```
array([[1.125]])
```



```
brain.predict(input_scaled)
```

```
array([[94.98506]], dtype=float32)
```

RESULT

A Basic neural network regression model for the given dataset is developed successfully.