Assignment 1

Problem Statement :

Linear regression by using Deep Neural network: Implement Boston housing price prediction problem by Linear regression using Deep Neural network. Use Boston House price prediction dataset.

```
import tensorflow as tf
from tensorflow.keras.datasets import boston_housing
from sklearn import preprocessing
import statistics

(train_x,train_y),(test_x,test_y)=boston_housing.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/boston_housing.npz
57026/57026 [==============================] - 0s 9us/step
```

```
print("Train shape",train_x.shape)
print("Test shape",test_x.shape)
print("Actual train output",train_x.shape)
print("Actual test output",test_y.shape)
```

```
Train shape (404, 13)
Test shape (102, 13)
Actual train output (404, 13)
Actual test output (102,)
```

```
print(train_x[0])
train_y[0]
```

```
[  1.23247   0.        8.14      0.        0.538     6.142    91.7
   3.9769    4.      307.       21.      396.9      18.72   ]

15.2
```

```
train_x = preprocessing.normalize(train_x)
test_x = preprocessing.normalize(test_x)

train_x[0]
```

```
array([0.0024119 , 0.        , 0.01592969, 0.        , 0.00105285,
       0.01201967, 0.17945359, 0.00778265, 0.00782786, 0.6007879 ,
       0.04109624, 0.77671895, 0.03663436])
```

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import *
```

```
def HousePricePredictionModel():
    model = Sequential()
    model.add(Dense(128,activation = 'relu' ,
input_shape=(train_x[0].shape)))
    model.add(Dense(64,activation = 'relu'))
    model.add(Dense(32,activation = 'relu'))
    model.add(Dense(1))
    model.compile(optimizer = 'rmsprop',loss = 'mse', metrics =
['mae'])
    return model

import numpy as np
k = 4
num_val__samples = len(train_x)
num_epochs = 100
all_scores = []

model = HousePricePredictionModel()
history = model.fit(x = train_x , y = train_y , epochs = num_epochs ,
batch_size = 1 , verbose = 1 , validation_data = (test_x,test_y))
```

```
Epoch 1/100
WARNING:tensorflow:From C:\Users\Dell\AppData\Local\Programs\Python\
Python310\lib\site-packages\keras\src\utils\tf_utils.py:492: The name
tf.ragged.RaggedTensorValue is deprecated. Please use
tf.compat.v1.ragged.RaggedTensorValue instead.

WARNING:tensorflow:From C:\Users\Dell\AppData\Local\Programs\Python\
Python310\lib\site-packages\keras\src\engine\base_layer_utils.py:384:
The name tf.executing_eagerly_outside_functions is deprecated. Please
use tf.compat.v1.executing_eagerly_outside_functions instead.

404/404 [==============================] - 3s 5ms/step - loss:
128.0591 - mae: 7.8230 - val_loss: 68.9412 - val_mae: 6.3187
Epoch 2/100
404/404 [==============================] - 1s 3ms/step - loss: 67.6059
- mae: 5.8474 - val_loss: 60.7842 - val_mae: 5.6364
Epoch 3/100
404/404 [==============================] - 1s 3ms/step - loss: 63.1525
- mae: 5.5251 - val_loss: 56.2599 - val_mae: 5.6630
Epoch 4/100
404/404 [==============================] - 1s 3ms/step - loss: 60.9098
- mae: 5.4064 - val_loss: 57.8930 - val_mae: 5.8615
Epoch 5/100
404/404 [==============================] - 1s 3ms/step - loss: 58.9843
- mae: 5.3897 - val_loss: 63.9455 - val_mae: 5.6201
Epoch 6/100
404/404 [==============================] - 1s 3ms/step - loss: 55.0451
- mae: 5.1313 - val_loss: 54.2596 - val_mae: 5.2990
Epoch 7/100
```

```
404/404 [==============================] - 1s 3ms/step - loss: 54.2607
- mae: 5.0763 - val_loss: 61.4212 - val_mae: 5.4867
Epoch 8/100
404/404 [==============================] - 1s 3ms/step - loss: 53.8978
- mae: 4.9398 - val_loss: 50.7711 - val_mae: 5.3108
Epoch 9/100
404/404 [==============================] - 1s 3ms/step - loss: 52.0195
- mae: 4.9370 - val_loss: 48.9594 - val_mae: 4.9985
Epoch 10/100
404/404 [==============================] - 1s 3ms/step - loss: 50.9098
- mae: 4.7568 - val_loss: 48.4572 - val_mae: 5.0687
Epoch 11/100
404/404 [==============================] - 1s 3ms/step - loss: 49.2079
- mae: 4.7740 - val_loss: 52.3460 - val_mae: 5.0752
Epoch 12/100
404/404 [==============================] - 1s 4ms/step - loss: 46.8090
- mae: 4.6680 - val_loss: 43.8154 - val_mae: 4.7239
Epoch 13/100
404/404 [==============================] - 1s 3ms/step - loss: 45.9078
- mae: 4.5930 - val_loss: 42.5392 - val_mae: 4.5271
Epoch 14/100
404/404 [==============================] - 1s 3ms/step - loss: 43.4102
- mae: 4.4959 - val_loss: 59.2844 - val_mae: 5.4363
Epoch 15/100
404/404 [==============================] - 2s 4ms/step - loss: 42.7661
- mae: 4.5453 - val_loss: 39.0927 - val_mae: 4.3184
Epoch 16/100
404/404 [==============================] - 2s 4ms/step - loss: 40.8858
- mae: 4.4501 - val_loss: 39.6468 - val_mae: 4.3140
Epoch 17/100
404/404 [==============================] - 2s 4ms/step - loss: 40.1377
- mae: 4.3915 - val_loss: 36.1830 - val_mae: 4.3275
Epoch 18/100
404/404 [==============================] - 1s 3ms/step - loss: 39.0187
- mae: 4.3241 - val_loss: 34.2512 - val_mae: 4.0915
Epoch 19/100
404/404 [==============================] - 1s 3ms/step - loss: 36.1298
- mae: 4.1357 - val_loss: 46.1434 - val_mae: 4.6565
Epoch 20/100
404/404 [==============================] - 1s 3ms/step - loss: 35.6581
- mae: 4.1835 - val_loss: 32.9330 - val_mae: 4.1309
Epoch 21/100
404/404 [==============================] - 1s 3ms/step - loss: 35.1733
- mae: 4.0955 - val_loss: 32.8552 - val_mae: 4.2409
Epoch 22/100
404/404 [==============================] - 1s 3ms/step - loss: 34.5290
- mae: 4.0744 - val_loss: 30.1412 - val_mae: 3.8964
Epoch 23/100
404/404 [==============================] - 2s 4ms/step - loss: 33.9822
```

```
- mae: 4.0255 - val_loss: 36.2702 - val_mae: 4.1493
Epoch 24/100
404/404 [==============================] - 2s 4ms/step - loss: 32.9094
- mae: 4.0761 - val_loss: 29.1224 - val_mae: 3.8251
Epoch 25/100
404/404 [==============================] - 1s 3ms/step - loss: 30.9283
- mae: 3.9159 - val_loss: 48.7245 - val_mae: 4.8987
Epoch 26/100
404/404 [==============================] - 1s 3ms/step - loss: 30.7553
- mae: 3.8292 - val_loss: 29.8893 - val_mae: 4.0170
Epoch 27/100
404/404 [==============================] - 1s 3ms/step - loss: 30.8381
- mae: 3.9341 - val_loss: 35.7099 - val_mae: 4.0735
Epoch 28/100
404/404 [==============================] - 1s 3ms/step - loss: 29.8851
- mae: 3.7983 - val_loss: 31.9667 - val_mae: 3.8874
Epoch 29/100
404/404 [==============================] - 1s 3ms/step - loss: 29.7331
- mae: 3.9637 - val_loss: 30.6505 - val_mae: 3.7979
Epoch 30/100
404/404 [==============================] - 1s 3ms/step - loss: 27.7593
- mae: 3.8146 - val_loss: 28.2646 - val_mae: 3.7082
Epoch 31/100
404/404 [==============================] - 1s 3ms/step - loss: 27.2651
- mae: 3.7619 - val_loss: 28.4177 - val_mae: 4.1479
Epoch 32/100
404/404 [==============================] - 1s 3ms/step - loss: 27.1605
- mae: 3.6074 - val_loss: 25.4591 - val_mae: 3.6637
Epoch 33/100
404/404 [==============================] - 1s 3ms/step - loss: 26.7894
- mae: 3.6441 - val_loss: 35.8763 - val_mae: 4.1622
Epoch 34/100
404/404 [==============================] - 2s 4ms/step - loss: 26.4238
- mae: 3.6945 - val_loss: 25.6150 - val_mae: 3.5820
Epoch 35/100
404/404 [==============================] - 1s 3ms/step - loss: 24.9481
- mae: 3.5590 - val_loss: 35.5714 - val_mae: 4.1870
Epoch 36/100
404/404 [==============================] - 1s 4ms/step - loss: 24.6436
- mae: 3.5287 - val_loss: 24.4494 - val_mae: 3.5551
Epoch 37/100
404/404 [==============================] - 1s 4ms/step - loss: 24.2750
- mae: 3.5034 - val_loss: 25.8649 - val_mae: 3.6792
Epoch 38/100
404/404 [==============================] - 1s 4ms/step - loss: 24.3524
- mae: 3.5263 - val_loss: 26.6710 - val_mae: 3.6159
Epoch 39/100
404/404 [==============================] - 2s 4ms/step - loss: 23.5969
- mae: 3.4885 - val_loss: 36.5503 - val_mae: 4.2475
```

```
Epoch 40/100
404/404 [==============================] - 1s 3ms/step - loss: 22.4592
- mae: 3.4193 - val_loss: 28.6207 - val_mae: 4.2058
Epoch 41/100
404/404 [==============================] - 2s 4ms/step - loss: 22.7920
- mae: 3.3916 - val_loss: 24.3072 - val_mae: 3.5271
Epoch 42/100
404/404 [==============================] - 1s 3ms/step - loss: 21.9568
- mae: 3.3257 - val_loss: 25.3539 - val_mae: 3.6888
Epoch 43/100
404/404 [==============================] - 1s 3ms/step - loss: 21.6191
- mae: 3.3876 - val_loss: 25.5542 - val_mae: 3.5290
Epoch 44/100
404/404 [==============================] - 1s 3ms/step - loss: 20.5246
- mae: 3.2976 - val_loss: 30.2846 - val_mae: 3.8610
Epoch 45/100
404/404 [==============================] - 1s 3ms/step - loss: 21.7870
- mae: 3.4451 - val_loss: 25.9018 - val_mae: 3.5374
Epoch 46/100
404/404 [==============================] - 1s 3ms/step - loss: 20.6074
- mae: 3.3189 - val_loss: 25.8129 - val_mae: 3.7734
Epoch 47/100
404/404 [==============================] - 1s 3ms/step - loss: 20.1333
- mae: 3.1198 - val_loss: 26.9875 - val_mae: 3.6932
Epoch 48/100
404/404 [==============================] - 1s 3ms/step - loss: 20.6364
- mae: 3.3685 - val_loss: 31.1392 - val_mae: 3.9131
Epoch 49/100
404/404 [==============================] - 1s 3ms/step - loss: 19.4160
- mae: 3.2307 - val_loss: 28.7727 - val_mae: 4.0031
Epoch 50/100
404/404 [==============================] - 1s 3ms/step - loss: 21.0954
- mae: 3.2292 - val_loss: 26.3803 - val_mae: 3.5611
Epoch 51/100
404/404 [==============================] - 1s 3ms/step - loss: 19.5987
- mae: 3.1358 - val_loss: 28.8821 - val_mae: 3.7873
Epoch 52/100
404/404 [==============================] - 1s 3ms/step - loss: 20.0229
- mae: 3.1643 - val_loss: 33.9240 - val_mae: 4.1435
Epoch 53/100
404/404 [==============================] - 1s 3ms/step - loss: 19.0345
- mae: 3.1406 - val_loss: 26.4216 - val_mae: 3.6331
Epoch 54/100
404/404 [==============================] - 1s 3ms/step - loss: 18.7779
- mae: 3.1805 - val_loss: 28.4030 - val_mae: 3.9584
Epoch 55/100
404/404 [==============================] - 1s 3ms/step - loss: 20.1457
- mae: 3.1775 - val_loss: 27.2741 - val_mae: 3.8400
Epoch 56/100
```

```
404/404 [==============================] - 1s 3ms/step - loss: 18.6990
- mae: 3.0941 - val_loss: 26.6840 - val_mae: 3.5288
Epoch 57/100
404/404 [==============================] - 1s 3ms/step - loss: 19.4063
- mae: 3.1427 - val_loss: 38.9547 - val_mae: 4.4885
Epoch 58/100
404/404 [==============================] - 1s 3ms/step - loss: 18.2911
- mae: 3.1791 - val_loss: 33.5011 - val_mae: 4.0668
Epoch 59/100
404/404 [==============================] - 1s 3ms/step - loss: 18.7334
- mae: 3.0623 - val_loss: 26.5348 - val_mae: 3.5416
Epoch 60/100
404/404 [==============================] - 1s 3ms/step - loss: 19.6298
- mae: 3.1293 - val_loss: 34.6186 - val_mae: 4.1526
Epoch 61/100
404/404 [==============================] - 1s 3ms/step - loss: 19.5372
- mae: 3.1269 - val_loss: 27.8989 - val_mae: 3.6327
Epoch 62/100
404/404 [==============================] - 1s 3ms/step - loss: 18.5952
- mae: 3.0574 - val_loss: 28.7944 - val_mae: 3.6100
Epoch 63/100
404/404 [==============================] - 1s 4ms/step - loss: 18.6549
- mae: 3.1885 - val_loss: 30.5573 - val_mae: 3.7372
Epoch 64/100
404/404 [==============================] - 2s 4ms/step - loss: 18.5624
- mae: 3.1386 - val_loss: 34.5406 - val_mae: 4.2747
Epoch 65/100
404/404 [==============================] - 1s 4ms/step - loss: 16.6315
- mae: 3.0581 - val_loss: 31.2109 - val_mae: 3.8233
Epoch 66/100
404/404 [==============================] - 1s 3ms/step - loss: 18.6400
- mae: 3.0429 - val_loss: 26.7627 - val_mae: 3.8113
Epoch 67/100
404/404 [==============================] - 1s 4ms/step - loss: 17.5068
- mae: 3.1240 - val_loss: 31.4267 - val_mae: 3.7336
Epoch 68/100
404/404 [==============================] - 2s 4ms/step - loss: 17.5432
- mae: 2.9807 - val_loss: 28.8556 - val_mae: 3.6217
Epoch 69/100
404/404 [==============================] - 2s 4ms/step - loss: 18.0975
- mae: 3.1664 - val_loss: 31.1792 - val_mae: 3.8508
Epoch 70/100
404/404 [==============================] - 1s 4ms/step - loss: 18.2561
- mae: 3.0782 - val_loss: 29.9481 - val_mae: 3.8888
Epoch 71/100
404/404 [==============================] - 1s 3ms/step - loss: 16.8610
- mae: 3.0550 - val_loss: 32.5425 - val_mae: 3.8812
Epoch 72/100
404/404 [==============================] - 1s 4ms/step - loss: 18.5199
```

```
- mae: 3.0208 - val_loss: 32.9388 - val_mae: 3.8158
Epoch 73/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0933
- mae: 2.9963 - val_loss: 30.9806 - val_mae: 3.7196
Epoch 74/100
404/404 [==============================] - 1s 3ms/step - loss: 16.9173
- mae: 2.9822 - val_loss: 32.0714 - val_mae: 3.8845
Epoch 75/100
404/404 [==============================] - 1s 3ms/step - loss: 18.4523
- mae: 3.0460 - val_loss: 29.4708 - val_mae: 3.7798
Epoch 76/100
404/404 [==============================] - 1s 3ms/step - loss: 17.3997
- mae: 2.8880 - val_loss: 31.2977 - val_mae: 3.8956
Epoch 77/100
404/404 [==============================] - 1s 3ms/step - loss: 16.3358
- mae: 2.9086 - val_loss: 35.8937 - val_mae: 4.6390
Epoch 78/100
404/404 [==============================] - 1s 3ms/step - loss: 17.2411
- mae: 2.9369 - val_loss: 36.4089 - val_mae: 4.3208
Epoch 79/100
404/404 [==============================] - 1s 3ms/step - loss: 17.6014
- mae: 3.0158 - val_loss: 32.0566 - val_mae: 3.8074
Epoch 80/100
404/404 [==============================] - 1s 3ms/step - loss: 17.5533
- mae: 2.9510 - val_loss: 28.3061 - val_mae: 3.7662
Epoch 81/100
404/404 [==============================] - 1s 3ms/step - loss: 17.3332
- mae: 2.9636 - val_loss: 37.3824 - val_mae: 4.1582
Epoch 82/100
404/404 [==============================] - 1s 3ms/step - loss: 18.0341
- mae: 2.9911 - val_loss: 34.4895 - val_mae: 3.9884
Epoch 83/100
404/404 [==============================] - 1s 3ms/step - loss: 18.0652
- mae: 3.0692 - val_loss: 28.7954 - val_mae: 3.6450
Epoch 84/100
404/404 [==============================] - 1s 3ms/step - loss: 17.1450
- mae: 2.9391 - val_loss: 29.6099 - val_mae: 3.7594
Epoch 85/100
404/404 [==============================] - 1s 3ms/step - loss: 17.3397
- mae: 3.0009 - val_loss: 52.9308 - val_mae: 5.4933
Epoch 86/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0946
- mae: 2.9237 - val_loss: 36.6663 - val_mae: 4.4242
Epoch 87/100
404/404 [==============================] - 1s 3ms/step - loss: 17.4586
- mae: 2.8675 - val_loss: 29.6442 - val_mae: 3.5867
Epoch 88/100
404/404 [==============================] - 1s 3ms/step - loss: 17.2447
- mae: 3.0148 - val_loss: 30.8007 - val_mae: 3.7223
```

```
Epoch 89/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0878
- mae: 2.9349 - val_loss: 30.4669 - val_mae: 3.6414
Epoch 90/100
404/404 [==============================] - 1s 3ms/step - loss: 17.6201
- mae: 2.9095 - val_loss: 29.1059 - val_mae: 3.6205
Epoch 91/100
404/404 [==============================] - 1s 3ms/step - loss: 17.4104
- mae: 2.9661 - val_loss: 29.8378 - val_mae: 3.9131
Epoch 92/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0254
- mae: 2.9467 - val_loss: 31.0635 - val_mae: 3.7694
Epoch 93/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0923
- mae: 2.9177 - val_loss: 26.6914 - val_mae: 3.5139
Epoch 94/100
404/404 [==============================] - 1s 3ms/step - loss: 16.9053
- mae: 2.9064 - val_loss: 26.1249 - val_mae: 3.4292
Epoch 95/100
404/404 [==============================] - 1s 3ms/step - loss: 17.1182
- mae: 2.9948 - val_loss: 31.4276 - val_mae: 3.8287
Epoch 96/100
404/404 [==============================] - 1s 3ms/step - loss: 17.1544
- mae: 2.9307 - val_loss: 27.6774 - val_mae: 3.5788
Epoch 97/100
404/404 [==============================] - 1s 3ms/step - loss: 17.6296
- mae: 2.9563 - val_loss: 26.9573 - val_mae: 3.4660
Epoch 98/100
404/404 [==============================] - 1s 3ms/step - loss: 16.3157
- mae: 2.8757 - val_loss: 35.3300 - val_mae: 4.0154
Epoch 99/100
404/404 [==============================] - 1s 3ms/step - loss: 17.0984
- mae: 2.9171 - val_loss: 30.8429 - val_mae: 4.0270
Epoch 100/100
404/404 [==============================] - 1s 3ms/step - loss: 17.1548
- mae: 2.9354 - val_loss: 26.4961 - val_mae: 3.5876

y_pred = model.predict(test_x)

from IPython.display import display
import pandas as pd
import numpy as np

res = np.array(y_pred)
result = res.ravel()

dict = {'y_predicted' : result,
        'y_actual' : test_y}

df = pd.DataFrame(dict)
```

```
# displaying the DataFrame
display(df)
```

```
4/4 [==============================] - 0s 3ms/step
```

	y_predicted	y_actual
0	9.804584	7.2
1	18.540216	18.8
2	23.167807	19.0
3	35.055805	27.0
4	25.245564	22.2
..	...	...
97	45.174702	21.9
98	24.418425	24.1
99	53.045532	50.0
100	27.694336	26.7
101	18.541021	25.0

```
[102 rows x 2 columns]
```