

Assignment IV– Individual Take Home Assignment

1. ANN regression model

In [30]:

```
#import libraries

import numpy as np
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error

from tensorflow.python.keras.models import Sequential
from tensorflow.python.keras.layers import Dense
from tensorflow.python.keras.wrappers.scikit_learn import KerasRegressor

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

import time
```

In [31]:

```
#Import Variables & standardize
dataset=np.loadtxt("data/school_grades.csv", delimiter=",", skiprows=1)
x=dataset[:,0:4]
y=dataset[:,4]
y=np.reshape(y, (-1,1))
scaler_x = MinMaxScaler()
scaler_y = MinMaxScaler()
print(scaler_x.fit(x))
xscale=scaler_x.transform(x)
print(scaler_y.fit(y))
yscale=scaler_y.transform(y)
```

```
MinMaxScaler(copy=True, feature_range=(0, 1))
MinMaxScaler(copy=True, feature_range=(0, 1))
```

In [32]:

```
#Split data into train & test
X_train, X_test, y_train, y_test = train_test_split(xscale, yscale)
```

In [33]:

```
#defining a 3 layer deep NN [12,8,1]
model = Sequential()
model.add(Dense(12, input_dim=4, kernel_initializer='normal', activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='linear'))
model.summary()
```

Model: "sequential_2"

Layer (type)	Output Shape	Param #
dense_6 (Dense)	(None, 12)	60
dense_7 (Dense)	(None, 8)	104

```
dense_8 (Dense)                (None, 1)                9
=====
Total params: 173
Trainable params: 173
Non-trainable params: 0
=====
```

In [34]:

```
#The mean_squared_error (mse) and mean_absolute_error (mae) are our loss functions - i.e. an estimate of how accurate the neural network is in predicting the test data.
model.compile(loss='mse', optimizer='adam', metrics=['mse', 'mae'])
```

In [35]:

```
#fitting model
start_time = time.time()
history = model.fit(X_train, y_train, epochs=150, batch_size=50, verbose=1, validation_split=0.2)
end_time = time.time()
ANN_time = end_time - start_time
```

Train on 236 samples, validate on 60 samples

```
Epoch 1/150
236/236 [=====] - 0s 852us/sample - loss: 0.3438 - mse: 0.3438 - mae: 0.5399 - val_loss: 0.3271 - val_mse: 0.3271 - val_mae: 0.5129
Epoch 2/150
236/236 [=====] - 0s 83us/sample - loss: 0.3161 - mse: 0.3161 - mae: 0.5153 - val_loss: 0.3055 - val_mse: 0.3055 - val_mae: 0.4949
Epoch 3/150
236/236 [=====] - 0s 47us/sample - loss: 0.3003 - mse: 0.3003 - mae: 0.5027 - val_loss: 0.2965 - val_mse: 0.2965 - val_mae: 0.4882
Epoch 4/150
236/236 [=====] - 0s 42us/sample - loss: 0.2932 - mse: 0.2932 - mae: 0.4968 - val_loss: 0.2892 - val_mse: 0.2892 - val_mae: 0.4823
Epoch 5/150
236/236 [=====] - 0s 42us/sample - loss: 0.2853 - mse: 0.2853 - mae: 0.4903 - val_loss: 0.2802 - val_mse: 0.2802 - val_mae: 0.4749
Epoch 6/150
236/236 [=====] - 0s 42us/sample - loss: 0.2748 - mse: 0.2748 - mae: 0.4814 - val_loss: 0.2688 - val_mse: 0.2688 - val_mae: 0.4653
Epoch 7/150
236/236 [=====] - 0s 55us/sample - loss: 0.2622 - mse: 0.2622 - mae: 0.4705 - val_loss: 0.2558 - val_mse: 0.2558 - val_mae: 0.4542
Epoch 8/150
236/236 [=====] - 0s 38us/sample - loss: 0.2488 - mse: 0.2488 - mae: 0.4586 - val_loss: 0.2428 - val_mse: 0.2428 - val_mae: 0.4427
Epoch 9/150
236/236 [=====] - 0s 38us/sample - loss: 0.2361 - mse: 0.2361 - mae: 0.4469 - val_loss: 0.2302 - val_mse: 0.2302 - val_mae: 0.4313
Epoch 10/150
236/236 [=====] - 0s 47us/sample - loss: 0.2230 - mse: 0.2230 - mae: 0.4345 - val_loss: 0.2169 - val_mse: 0.2169 - val_mae: 0.4189
Epoch 11/150
236/236 [=====] - 0s 42us/sample - loss: 0.2094 - mse: 0.2094 - mae: 0.4213 - val_loss: 0.2036 - val_mse: 0.2036 - val_mae: 0.4060
Epoch 12/150
236/236 [=====] - 0s 42us/sample - loss: 0.1958 - mse: 0.1958 - mae: 0.4077 - val_loss: 0.1901 - val_mse: 0.1901 - val_mae: 0.3923
Epoch 13/150
236/236 [=====] - 0s 38us/sample - loss: 0.1821 - mse: 0.1821 - mae: 0.3930 - val_loss: 0.1763 - val_mse: 0.1763 - val_mae: 0.3776
Epoch 14/150
236/236 [=====] - 0s 42us/sample - loss: 0.1680 - mse: 0.1680 - mae: 0.3772 - val_loss: 0.1624 - val_mse: 0.1624 - val_mae: 0.3620
Epoch 15/150
236/236 [=====] - 0s 38us/sample - loss: 0.1539 - mse: 0.1539 - mae: 0.3608 - val_loss: 0.1485 - val_mse: 0.1485 - val_mae: 0.3456
Epoch 16/150
236/236 [=====] - 0s 38us/sample - loss: 0.1401 - mse: 0.1401 - mae: 0.3436 - val_loss: 0.1348 - val_mse: 0.1348 - val_mae: 0.3283
Epoch 17/150
236/236 [=====] - 0s 42us/sample - loss: 0.1260 - mse: 0.1260 - mae: 0.3253 - val_loss: 0.1216 - val_mse: 0.1216 - val_mae: 0.3103
Epoch 18/150
```

236/236 [=====] - 0s 42us/sample - loss: 0.1127 - mse: 0.1127 - mae: 0.30
67 - val_loss: 0.1088 - val_mse: 0.1088 - val_mae: 0.2915
Epoch 19/150
236/236 [=====] - 0s 47us/sample - loss: 0.1001 - mse: 0.1001 - mae: 0.28
71 - val_loss: 0.0966 - val_mse: 0.0966 - val_mae: 0.2724
Epoch 20/150
236/236 [=====] - 0s 38us/sample - loss: 0.0877 - mse: 0.0877 - mae: 0.26
69 - val_loss: 0.0854 - val_mse: 0.0854 - val_mae: 0.2534
Epoch 21/150
236/236 [=====] - 0s 42us/sample - loss: 0.0769 - mse: 0.0769 - mae: 0.24
70 - val_loss: 0.0752 - val_mse: 0.0752 - val_mae: 0.2343
Epoch 22/150
236/236 [=====] - 0s 38us/sample - loss: 0.0669 - mse: 0.0669 - mae: 0.22
75 - val_loss: 0.0663 - val_mse: 0.0663 - val_mae: 0.2171
Epoch 23/150
236/236 [=====] - 0s 38us/sample - loss: 0.0579 - mse: 0.0579 - mae: 0.20
92 - val_loss: 0.0588 - val_mse: 0.0588 - val_mae: 0.2015
Epoch 24/150
236/236 [=====] - 0s 42us/sample - loss: 0.0506 - mse: 0.0506 - mae: 0.19
16 - val_loss: 0.0525 - val_mse: 0.0525 - val_mae: 0.1864
Epoch 25/150
236/236 [=====] - 0s 42us/sample - loss: 0.0440 - mse: 0.0440 - mae: 0.17
57 - val_loss: 0.0475 - val_mse: 0.0475 - val_mae: 0.1744
Epoch 26/150
236/236 [=====] - 0s 38us/sample - loss: 0.0391 - mse: 0.0391 - mae: 0.16
14 - val_loss: 0.0436 - val_mse: 0.0436 - val_mae: 0.1642
Epoch 27/150
236/236 [=====] - 0s 42us/sample - loss: 0.0354 - mse: 0.0354 - mae: 0.15
02 - val_loss: 0.0406 - val_mse: 0.0406 - val_mae: 0.1565
Epoch 28/150
236/236 [=====] - 0s 47us/sample - loss: 0.0326 - mse: 0.0326 - mae: 0.14
06 - val_loss: 0.0385 - val_mse: 0.0385 - val_mae: 0.1511
Epoch 29/150
236/236 [=====] - 0s 42us/sample - loss: 0.0304 - mse: 0.0304 - mae: 0.13
20 - val_loss: 0.0369 - val_mse: 0.0369 - val_mae: 0.1476
Epoch 30/150
236/236 [=====] - 0s 42us/sample - loss: 0.0288 - mse: 0.0288 - mae: 0.12
56 - val_loss: 0.0357 - val_mse: 0.0357 - val_mae: 0.1446
Epoch 31/150
236/236 [=====] - 0s 38us/sample - loss: 0.0276 - mse: 0.0276 - mae: 0.12
08 - val_loss: 0.0346 - val_mse: 0.0346 - val_mae: 0.1418
Epoch 32/150
236/236 [=====] - 0s 34us/sample - loss: 0.0266 - mse: 0.0266 - mae: 0.11
75 - val_loss: 0.0337 - val_mse: 0.0337 - val_mae: 0.1392
Epoch 33/150
236/236 [=====] - 0s 38us/sample - loss: 0.0258 - mse: 0.0258 - mae: 0.11
46 - val_loss: 0.0328 - val_mse: 0.0328 - val_mae: 0.1367
Epoch 34/150
236/236 [=====] - 0s 42us/sample - loss: 0.0250 - mse: 0.0250 - mae: 0.11
19 - val_loss: 0.0319 - val_mse: 0.0319 - val_mae: 0.1342
Epoch 35/150
236/236 [=====] - 0s 42us/sample - loss: 0.0243 - mse: 0.0243 - mae: 0.10
95 - val_loss: 0.0311 - val_mse: 0.0311 - val_mae: 0.1317
Epoch 36/150
236/236 [=====] - 0s 51us/sample - loss: 0.0236 - mse: 0.0236 - mae: 0.10
74 - val_loss: 0.0302 - val_mse: 0.0302 - val_mae: 0.1294
Epoch 37/150
236/236 [=====] - 0s 42us/sample - loss: 0.0228 - mse: 0.0228 - mae: 0.10
54 - val_loss: 0.0293 - val_mse: 0.0293 - val_mae: 0.1270
Epoch 38/150
236/236 [=====] - 0s 38us/sample - loss: 0.0222 - mse: 0.0222 - mae: 0.10
36 - val_loss: 0.0283 - val_mse: 0.0283 - val_mae: 0.1245
Epoch 39/150
236/236 [=====] - 0s 38us/sample - loss: 0.0215 - mse: 0.0215 - mae: 0.10
15 - val_loss: 0.0275 - val_mse: 0.0275 - val_mae: 0.1220
Epoch 40/150
236/236 [=====] - 0s 42us/sample - loss: 0.0208 - mse: 0.0208 - mae: 0.09
94 - val_loss: 0.0266 - val_mse: 0.0266 - val_mae: 0.1194
Epoch 41/150
236/236 [=====] - 0s 34us/sample - loss: 0.0200 - mse: 0.0200 - mae: 0.09
71 - val_loss: 0.0258 - val_mse: 0.0258 - val_mae: 0.1168
Epoch 42/150
236/236 [=====] - 0s 38us/sample - loss: 0.0193 - mse: 0.0193 - mae: 0.09
49 - val_loss: 0.0250 - val_mse: 0.0250 - val_mae: 0.1144
Epoch 43/150
236/236 [=====] - 0s 34us/sample - loss: 0.0186 - mse: 0.0186 - mae: 0.09
31 - val_loss: 0.0241 - val_mse: 0.0241 - val_mae: 0.1120

Epoch 44/150
236/236 [=====] - 0s 34us/sample - loss: 0.0180 - mse: 0.0180 - mae: 0.0917 - val_loss: 0.0231 - val_mse: 0.0231 - val_mae: 0.1097
Epoch 45/150
236/236 [=====] - 0s 38us/sample - loss: 0.0173 - mse: 0.0173 - mae: 0.0898 - val_loss: 0.0221 - val_mse: 0.0221 - val_mae: 0.1062
Epoch 46/150
236/236 [=====] - 0s 38us/sample - loss: 0.0164 - mse: 0.0164 - mae: 0.0871 - val_loss: 0.0208 - val_mse: 0.0208 - val_mae: 0.1021
Epoch 47/150
236/236 [=====] - 0s 34us/sample - loss: 0.0156 - mse: 0.0156 - mae: 0.0847 - val_loss: 0.0198 - val_mse: 0.0198 - val_mae: 0.0998
Epoch 48/150
236/236 [=====] - 0s 38us/sample - loss: 0.0149 - mse: 0.0149 - mae: 0.0834 - val_loss: 0.0191 - val_mse: 0.0191 - val_mae: 0.0976
Epoch 49/150
236/236 [=====] - 0s 34us/sample - loss: 0.0144 - mse: 0.0144 - mae: 0.0817 - val_loss: 0.0184 - val_mse: 0.0184 - val_mae: 0.0954
Epoch 50/150
236/236 [=====] - 0s 38us/sample - loss: 0.0139 - mse: 0.0139 - mae: 0.0796 - val_loss: 0.0178 - val_mse: 0.0178 - val_mae: 0.0927
Epoch 51/150
236/236 [=====] - 0s 38us/sample - loss: 0.0134 - mse: 0.0134 - mae: 0.0776 - val_loss: 0.0172 - val_mse: 0.0172 - val_mae: 0.0904
Epoch 52/150
236/236 [=====] - 0s 34us/sample - loss: 0.0130 - mse: 0.0130 - mae: 0.0759 - val_loss: 0.0167 - val_mse: 0.0167 - val_mae: 0.0885
Epoch 53/150
236/236 [=====] - 0s 38us/sample - loss: 0.0126 - mse: 0.0126 - mae: 0.0745 - val_loss: 0.0161 - val_mse: 0.0161 - val_mae: 0.0871
Epoch 54/150
236/236 [=====] - 0s 38us/sample - loss: 0.0122 - mse: 0.0122 - mae: 0.0735 - val_loss: 0.0156 - val_mse: 0.0156 - val_mae: 0.0861
Epoch 55/150
236/236 [=====] - 0s 34us/sample - loss: 0.0119 - mse: 0.0119 - mae: 0.0727 - val_loss: 0.0151 - val_mse: 0.0151 - val_mae: 0.0855
Epoch 56/150
236/236 [=====] - 0s 38us/sample - loss: 0.0115 - mse: 0.0115 - mae: 0.0718 - val_loss: 0.0147 - val_mse: 0.0147 - val_mae: 0.0840
Epoch 57/150
236/236 [=====] - 0s 38us/sample - loss: 0.0113 - mse: 0.0113 - mae: 0.0705 - val_loss: 0.0143 - val_mse: 0.0143 - val_mae: 0.0827
Epoch 58/150
236/236 [=====] - 0s 47us/sample - loss: 0.0110 - mse: 0.0110 - mae: 0.0691 - val_loss: 0.0140 - val_mse: 0.0140 - val_mae: 0.0812
Epoch 59/150
236/236 [=====] - 0s 47us/sample - loss: 0.0107 - mse: 0.0107 - mae: 0.0676 - val_loss: 0.0137 - val_mse: 0.0137 - val_mae: 0.0802
Epoch 60/150
236/236 [=====] - 0s 38us/sample - loss: 0.0105 - mse: 0.0105 - mae: 0.0670 - val_loss: 0.0133 - val_mse: 0.0133 - val_mae: 0.0797
Epoch 61/150
236/236 [=====] - 0s 38us/sample - loss: 0.0103 - mse: 0.0103 - mae: 0.0666 - val_loss: 0.0130 - val_mse: 0.0130 - val_mae: 0.0785
Epoch 62/150
236/236 [=====] - 0s 34us/sample - loss: 0.0101 - mse: 0.0101 - mae: 0.0654 - val_loss: 0.0127 - val_mse: 0.0127 - val_mae: 0.0771
Epoch 63/150
236/236 [=====] - 0s 38us/sample - loss: 0.0100 - mse: 0.0100 - mae: 0.0649 - val_loss: 0.0124 - val_mse: 0.0124 - val_mae: 0.0766
Epoch 64/150
236/236 [=====] - 0s 38us/sample - loss: 0.0098 - mse: 0.0098 - mae: 0.0641 - val_loss: 0.0122 - val_mse: 0.0122 - val_mae: 0.0754
Epoch 65/150
236/236 [=====] - 0s 38us/sample - loss: 0.0097 - mse: 0.0097 - mae: 0.0636 - val_loss: 0.0120 - val_mse: 0.0120 - val_mae: 0.0749
Epoch 66/150
236/236 [=====] - 0s 38us/sample - loss: 0.0097 - mse: 0.0097 - mae: 0.0632 - val_loss: 0.0119 - val_mse: 0.0119 - val_mae: 0.0743
Epoch 67/150
236/236 [=====] - 0s 38us/sample - loss: 0.0096 - mse: 0.0096 - mae: 0.0626 - val_loss: 0.0118 - val_mse: 0.0118 - val_mae: 0.0739
Epoch 68/150
236/236 [=====] - 0s 38us/sample - loss: 0.0095 - mse: 0.0095 - mae: 0.0624 - val_loss: 0.0116 - val_mse: 0.0116 - val_mae: 0.0741
Epoch 69/150
236/236 [=====] - 0s 34us/sample - loss: 0.0094 - mse: 0.0094 - mae: 0.0623

25 - val_loss: 0.0115 - val_mse: 0.0115 - val_mae: 0.0739
Epoch 70/150
236/236 [=====] - 0s 38us/sample - loss: 0.0094 - mse: 0.0094 - mae: 0.06
22 - val_loss: 0.0114 - val_mse: 0.0114 - val_mae: 0.0734
Epoch 71/150
236/236 [=====] - 0s 38us/sample - loss: 0.0093 - mse: 0.0093 - mae: 0.06
16 - val_loss: 0.0114 - val_mse: 0.0114 - val_mae: 0.0729
Epoch 72/150
236/236 [=====] - ETA: 0s - loss: 0.0095 - mse: 0.0095 - mae: 0.060 - 0s
38us/sample - loss: 0.0093 - mse: 0.0093 - mae: 0.0618 - val_loss: 0.0113 - val_mse: 0.0113 - val_mae: 0.0734
Epoch 73/150
236/236 [=====] - 0s 34us/sample - loss: 0.0092 - mse: 0.0092 - mae: 0.06
19 - val_loss: 0.0112 - val_mse: 0.0112 - val_mae: 0.0730
Epoch 74/150
236/236 [=====] - 0s 34us/sample - loss: 0.0092 - mse: 0.0092 - mae: 0.06
14 - val_loss: 0.0112 - val_mse: 0.0112 - val_mae: 0.0725
Epoch 75/150
236/236 [=====] - 0s 38us/sample - loss: 0.0092 - mse: 0.0092 - mae: 0.06
11 - val_loss: 0.0112 - val_mse: 0.0112 - val_mae: 0.0724
Epoch 76/150
236/236 [=====] - 0s 34us/sample - loss: 0.0091 - mse: 0.0091 - mae: 0.06
08 - val_loss: 0.0112 - val_mse: 0.0112 - val_mae: 0.0720
Epoch 77/150
236/236 [=====] - 0s 34us/sample - loss: 0.0091 - mse: 0.0091 - mae: 0.06
07 - val_loss: 0.0111 - val_mse: 0.0111 - val_mae: 0.0722
Epoch 78/150
236/236 [=====] - 0s 42us/sample - loss: 0.0091 - mse: 0.0091 - mae: 0.06
09 - val_loss: 0.0110 - val_mse: 0.0110 - val_mae: 0.0723
Epoch 79/150
236/236 [=====] - 0s 42us/sample - loss: 0.0090 - mse: 0.0090 - mae: 0.06
08 - val_loss: 0.0110 - val_mse: 0.0110 - val_mae: 0.0721
Epoch 80/150
236/236 [=====] - 0s 38us/sample - loss: 0.0090 - mse: 0.0090 - mae: 0.06
06 - val_loss: 0.0109 - val_mse: 0.0109 - val_mae: 0.0721
Epoch 81/150
236/236 [=====] - 0s 34us/sample - loss: 0.0090 - mse: 0.0090 - mae: 0.06
05 - val_loss: 0.0109 - val_mse: 0.0109 - val_mae: 0.0718
Epoch 82/150
236/236 [=====] - 0s 38us/sample - loss: 0.0090 - mse: 0.0090 - mae: 0.06
07 - val_loss: 0.0108 - val_mse: 0.0108 - val_mae: 0.0719
Epoch 83/150
236/236 [=====] - 0s 38us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.06
06 - val_loss: 0.0108 - val_mse: 0.0108 - val_mae: 0.0716
Epoch 84/150
236/236 [=====] - 0s 34us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.06
05 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0716
Epoch 85/150
236/236 [=====] - 0s 38us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.06
03 - val_loss: 0.0108 - val_mse: 0.0108 - val_mae: 0.0712
Epoch 86/150
236/236 [=====] - 0s 38us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.06
02 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0712
Epoch 87/150
236/236 [=====] - 0s 34us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.06
00 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0710
Epoch 88/150
236/236 [=====] - 0s 38us/sample - loss: 0.0089 - mse: 0.0089 - mae: 0.05
98 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0708
Epoch 89/150
236/236 [=====] - 0s 38us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.05
97 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0708
Epoch 90/150
236/236 [=====] - 0s 38us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.06
00 - val_loss: 0.0106 - val_mse: 0.0106 - val_mae: 0.0710
Epoch 91/150
236/236 [=====] - 0s 38us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.06
02 - val_loss: 0.0106 - val_mse: 0.0106 - val_mae: 0.0709
Epoch 92/150
236/236 [=====] - 0s 38us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.05
99 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0704
Epoch 93/150
236/236 [=====] - 0s 42us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.05
96 - val_loss: 0.0106 - val_mse: 0.0106 - val_mae: 0.0703
Epoch 94/150
236/236 [=====] - 0s 38us/sample - loss: 0.0088 - mse: 0.0088 - mae: 0.05
95 - val_loss: 0.0107 - val_mse: 0.0107 - val_mae: 0.0701

[illegible]

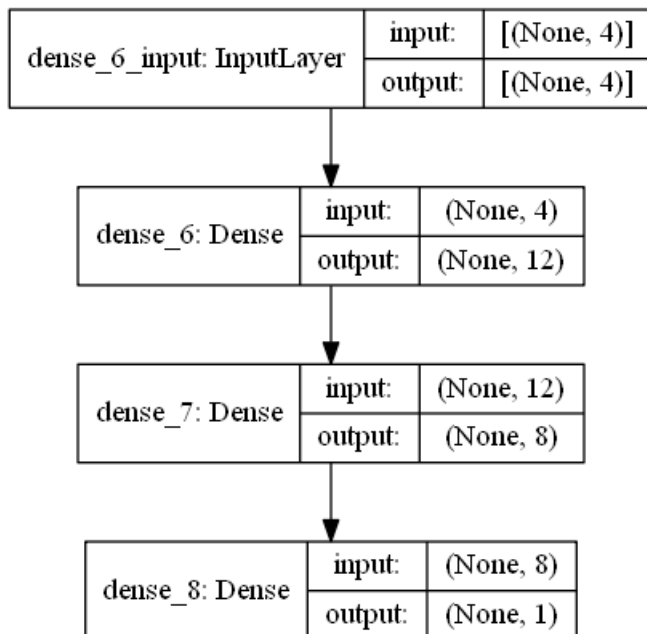
[illegible]

```
Epoch 146/150
236/236 [=====] - 0s 53us/sample - loss: 0.0085 - mse: 0.0085 - mae: 0.0571 - val_loss: 0.0105 - val_mse: 0.0105 - val_mae: 0.0675
Epoch 147/150
236/236 [=====] - 0s 30us/sample - loss: 0.0085 - mse: 0.0085 - mae: 0.0570 - val_loss: 0.0103 - val_mse: 0.0103 - val_mae: 0.0679
Epoch 148/150
236/236 [=====] - 0s 42us/sample - loss: 0.0085 - mse: 0.0085 - mae: 0.0585 - val_loss: 0.0102 - val_mse: 0.0102 - val_mae: 0.0684
Epoch 149/150
236/236 [=====] - 0s 13us/sample - loss: 0.0085 - mse: 0.0085 - mae: 0.0583 - val_loss: 0.0103 - val_mse: 0.0103 - val_mae: 0.0678
Epoch 150/150
236/236 [=====] - 0s 51us/sample - loss: 0.0084 - mse: 0.0084 - mae: 0.0572 - val_loss: 0.0104 - val_mse: 0.0104 - val_mae: 0.0676
```

In [36]:

```
from tensorflow.keras.utils import plot_model
#Vizualize Model
plot_model(model, to_file='ANN_model.png', show_shapes=True, show_layer_names=True, expand_nested=True)
```

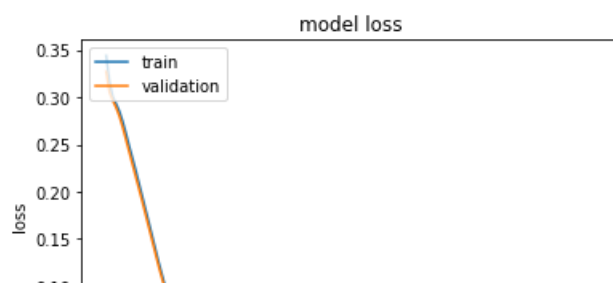
Out[36]:

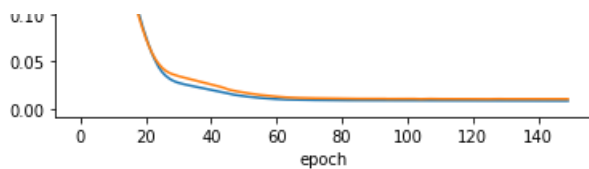


In [37]:

```
print(history.history.keys())
# plotting loss function with epoch
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'validation'], loc='upper left')
plt.show()
```

```
dict_keys(['loss', 'mse', 'mae', 'val_loss', 'val_mse', 'val_mae'])
```





In [38]:

```
y_pred = model.predict(X_test)

MSE_scaled = mean_squared_error(y_test, y_pred)

MSE = mean_squared_error(scaler_y.inverse_transform(y_test), scaler_y.inverse_transform(y_pred))

print("Scaled MSE:",MSE_scaled)
print("MSE:",MSE)
print("Runtime: %s seconds"%ANN_time)
```

Scaled MSE: 0.009582817080292176
MSE: 3.8331268356480934
Runtime: 2.195000648498535 seconds

2. Multiple Linear Regression Model

In [39]:

```
#import libraries

%matplotlib inline

from statsmodels.compat import lzip

import numpy as np
import pandas as pd
import statsmodels.formula.api as smf
import statsmodels.api as sm
import statsmodels.stats.api as sms
import matplotlib.pyplot as plt
```

In [40]:

```
#importing data
df = pd.read_csv('data/school_grades.csv')

#making copy to work with
df_copy = df.copy()

X = df_copy.drop(['G3'], axis=1)
y = df_copy["G3"]

#splitting training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3) # 70% training and 30% tes
t
```

In [41]:

```
# Fit regression model
start_time = time.time()
results = smf.ols('G3 ~ age + absences + G1 + G2', data=pd.concat([X_train,y_train], axis=1)).fit(
)
end_time = time.time()
MLR_time = end_time - start_time
# Inspect the results
print(results.summary())
```

OLS Regression Results

```
=====
Dep. Variable:          G3      R-squared:                0.783
Model:                  OLS      Adj. R-squared:           0.780
Method:                 Least Squares      Prob(F-statistic):    2.15e-1
```

```

Method:          Least Squares      F-statistic:          245.1
Date:            Fri, 28 Feb 2020    Prob (F-statistic):    9.94e-89
Time:            10:30:04           Log-Likelihood:        -600.71
No. Observations: 276              AIC:                   1211.
Df Residuals:    271              BIC:                   1230.
Df Model:        4
Covariance Type: nonrobust

```

```

=====
              coef      std err          t      P>|t|      [0.025      0.975]
-----
Intercept    0.7954      1.900        0.419    0.676    -2.944     4.535
age          -0.1952      0.109       -1.790    0.075    -0.410     0.019
absences      0.0516      0.016        3.162    0.002     0.019     0.084
G1            0.1962      0.080        2.445    0.015     0.038     0.354
G2            0.9641      0.073       13.160    0.000     0.820     1.108
=====
Omnibus:            132.410    Durbin-Watson:           1.965
Prob(Omnibus):      0.000    Jarque-Bera (JB):        509.188
Skew:               -2.095    Prob(JB):                2.70e-111
Kurtosis:           8.170    Cond. No.                 346.
=====

```

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [42]:

```

y_pred = results.predict(X_test)

MSE = mean_squared_error(y_test, y_pred)

print("MSE:",MSE)
print("Runtime: %s seconds"%MLR_time)

```

MSE: 1.4293875205069166

Runtime: 0.01698923110961914 seconds