# Deep Learning

# Assignment 3

# Regression Problem

## Abstract

To perform regression on the dataset UCI communities and crime against various optimizers and loss functions and to evaluate their scores. Finally the better model is taken and regularization is performed if needed to increase the efficiency more.

## Problem Statement

The communities' dataset has 1994 instances of socio economic data and 128 attributes with some missing data. The data is loaded into the model, the missing values are filled with NaN integers, split into training, validation and testing sets. Finally various loss functions and optimizers are used to determine the best model for the regression problem. For each f the model various hyper parameters are changed in accordance to the optimizer. Finally some regularization techniques are performed to get the most optimized model.

## Proposed Solution

Since the aim is to evaluate against all optimizers and losses, multiple models are created to determine the most suitable optimizer and loss.

**Loss functions used:**

The loss functions used for the regression models are:

Mean Squared Error – It is the mean square of the error. It includes the sum of all the data points from which the correct values are subtracted and the final value is squared.

Mean Absolute Error – It is the mean difference between the predicted value and the absolute value.

Huber Loss – This loss aims at reducing the outliers. It determines the difference between the absolute value and the predicted value. If this value is below a given threshold it squares them else it multiplies them linearly to enhance the loss so that it can be reduced during the gradient descent.

Logcosh – This loss determines the logcosh of the difference between the predicted value and the correct value.

**Optimizers used:**

Rmsprop – It is a gradient based optimization technique to optimize the neural network model. It decreased the gradient that increases to stop from exploding and increases the gradient that decreases to prevent from vanishing.

Sgd – At each iteration, this optimizer performs a parameter update. It finds the local minima and updates it to the parameters frequently to get the better results. One demerit of this optimizer is the frequent updates in the variance can cause a fluctuation in the loss function.

Adam – It is an alternative to sgd optimizer. It combines the properties of rmsprop and adagrad. It keeps the exponentially decaying average of the past gradients. One advantage of this optimizer is it converges faster and easily.

Adagrad – This optimizer is better than optimizers like rmsprop as it includes another hyper parameter in addition to the existing ones called the learning rate. Learning rate can be defined as the update or modification given to a parameter once there is a change in the gradients. This parameter makes a less update for a frequent parameter and a big update for a less frequent parameter.

**Regularization**

Regularization includes certain methods followed to prevent the data from overfitting and under fitting once the correct loss functions and optimizers are determined.

The techniques used are:

Dropout – After each iteration the weight of the input or the parameter is decayed or decreased to prevent overfitting.

Batch Normalization – In this case the output of each layer of the neural network is normalized in batches to give a scalable input to the following layers.

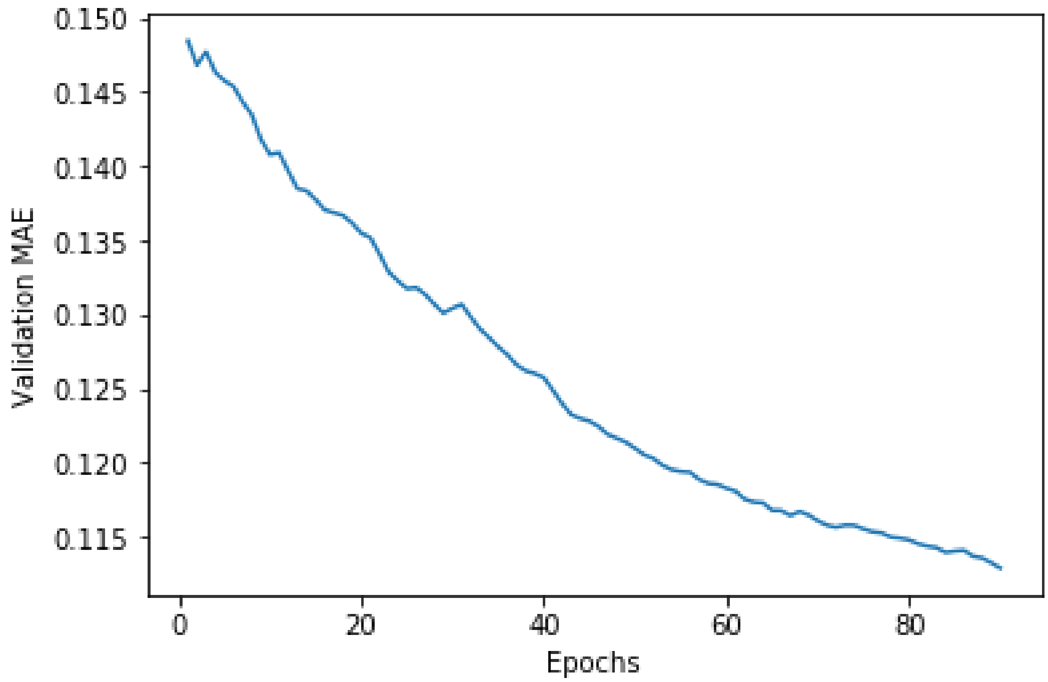
Ensemble classifier – Finally multiple models are trained on parts of the data and all the models are combined to obtain an averaged accuracy of all the models. By this method many incompletely trained models are combined to form a completely trained model.

## Evaluation metrics

### Rmsprop and mse

```
Epoch 95/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.0014 -
mae: 0.0268 - val_loss: 0.0289 - val_mae: 0.1222
Epoch 96/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0013 -
mae: 0.0260 - val_loss: 0.0287 - val_mae: 0.1168
Epoch 97/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.0012 -
mae: 0.0251 - val_loss: 0.0271 - val_mae: 0.1127
Epoch 98/100
1163/1163 [==============================] - 0s 33us/step - loss: 0.0014 -
mae: 0.0260 - val_loss: 0.0287 - val_mae: 0.1177
Epoch 99/100
1163/1163 [==============================] - 0s 38us/step - loss: 0.0013 -
mae: 0.0252 - val_loss: 0.0286 - val_mae: 0.1132
Epoch 100/100
```

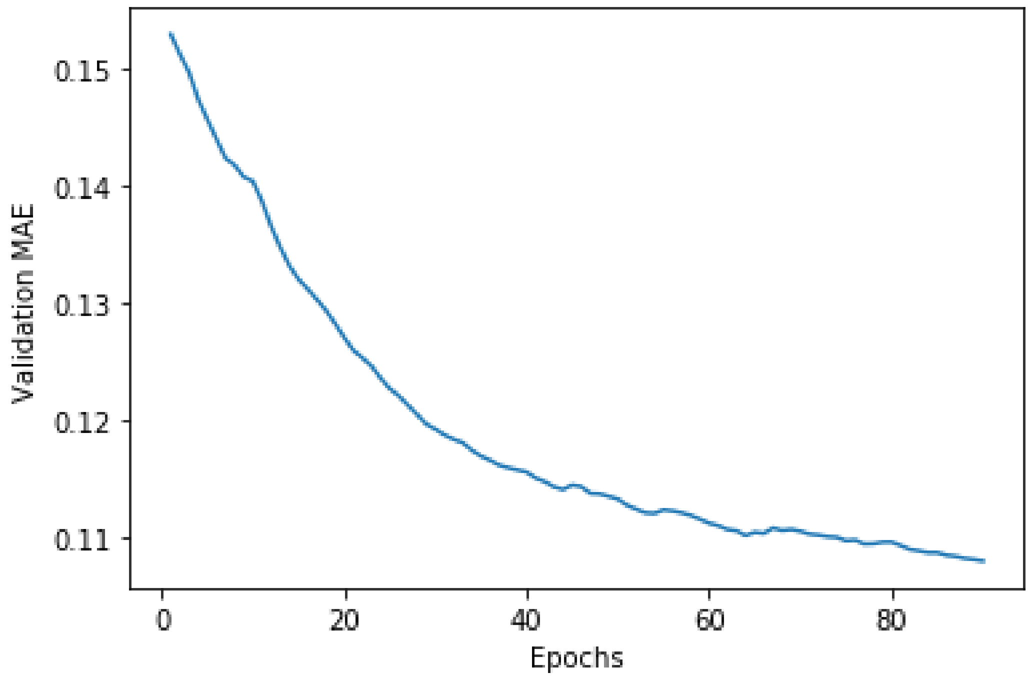
```
1163/1163 [==============================] - 0s 34us/step - loss: 0.0013 -
mae: 0.0251 - val_loss: 0.0269 - val_mae: 0.1146
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.10316283732652664
```

**Rms and mae**

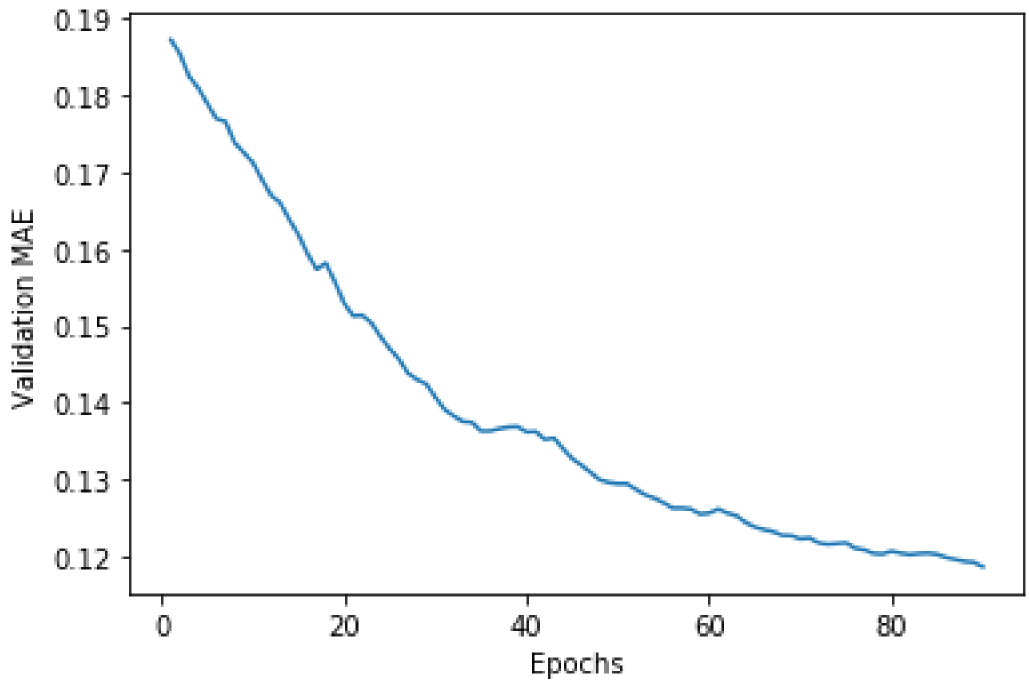
```
Epoch 95/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0296 -
mae: 0.0296 - val_loss: 0.1104 - val_mae: 0.1104
Epoch 96/100
1163/1163 [==============================] - 0s 34us/step - loss: 0.0300 -
mae: 0.0300 - val_loss: 0.1053 - val_mae: 0.1053
Epoch 97/100
1163/1163 [==============================] - 0s 33us/step - loss: 0.0284 -
mae: 0.0284 - val_loss: 0.1043 - val_mae: 0.1043
Epoch 98/100
1163/1163 [==============================] - 0s 32us/step - loss: 0.0290 -
mae: 0.0290 - val_loss: 0.1101 - val_mae: 0.1101
Epoch 99/100
1163/1163 [==============================] - 0s 34us/step - loss: 0.0282 -
mae: 0.0282 - val_loss: 0.1051 - val_mae: 0.1051
Epoch 100/100
1163/1163 [==============================] - 0s 33us/step - loss: 0.0276 -
mae: 0.0276 - val_loss: 0.1059 - val_mae: 0.1059
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.10624178498983383
```

**Rms and cross entropy**

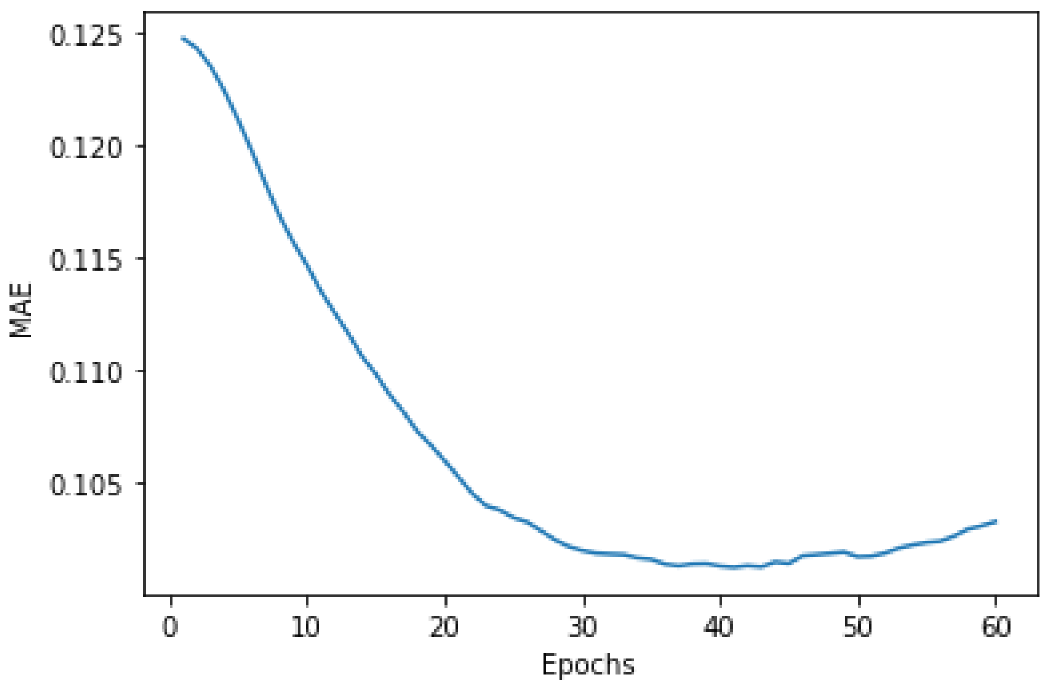
```
Epoch 95/100
1163/1163 [==============================] - 0s 37us/step - loss: 0.4333 -
mae: 0.0577 - val_loss: 0.4661 - val_mae: 0.1055
Epoch 96/100
1163/1163 [==============================] - 0s 38us/step - loss: 0.4301 -
mae: 0.0480 - val_loss: 0.4583 - val_mae: 0.0950
Epoch 97/100
1163/1163 [==============================] - 0s 42us/step - loss: 0.4391 -
mae: 0.0469 - val_loss: 0.4777 - val_mae: 0.1034
Epoch 98/100
1163/1163 [==============================] - 0s 37us/step - loss: 0.4376 -
mae: 0.0506 - val_loss: 0.4712 - val_mae: 0.0975
Epoch 99/100
1163/1163 [==============================] - 0s 39us/step - loss: 0.4291 -
mae: 0.0472 - val_loss: 0.5374 - val_mae: 0.1112
Epoch 100/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.4315 -
mae: 0.0469 - val_loss: 0.4467 - val_mae: 0.0978
```




```
Error rate
599/599 [==============================] - 0s 15us/step
599/599 [==============================] - 0s 37us/step
0.1124979704618454
```

**Rmsprop and huber**

```
Epoch 96/100
1163/1163 [==============================] - 0s 70us/step - loss: 0.0164 -
mae: 0.1303 - val_loss: 0.0103 - val_mae: 0.0961
Epoch 97/100
1163/1163 [==============================] - 0s 63us/step - loss: 0.0153 -
mae: 0.1272 - val_loss: 0.0124 - val_mae: 0.1031
Epoch 98/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0146 -
mae: 0.1234 - val_loss: 0.0143 - val_mae: 0.1059
Epoch 99/100
1163/1163 [==============================] - 0s 63us/step - loss: 0.0148 -
mae: 0.1253 - val_loss: 0.0124 - val_mae: 0.1002
Epoch 100/100
1163/1163 [==============================] - 0s 59us/step - loss: 0.0160 -
mae: 0.1290 - val_loss: 0.0102 - val_mae: 0.0985
```

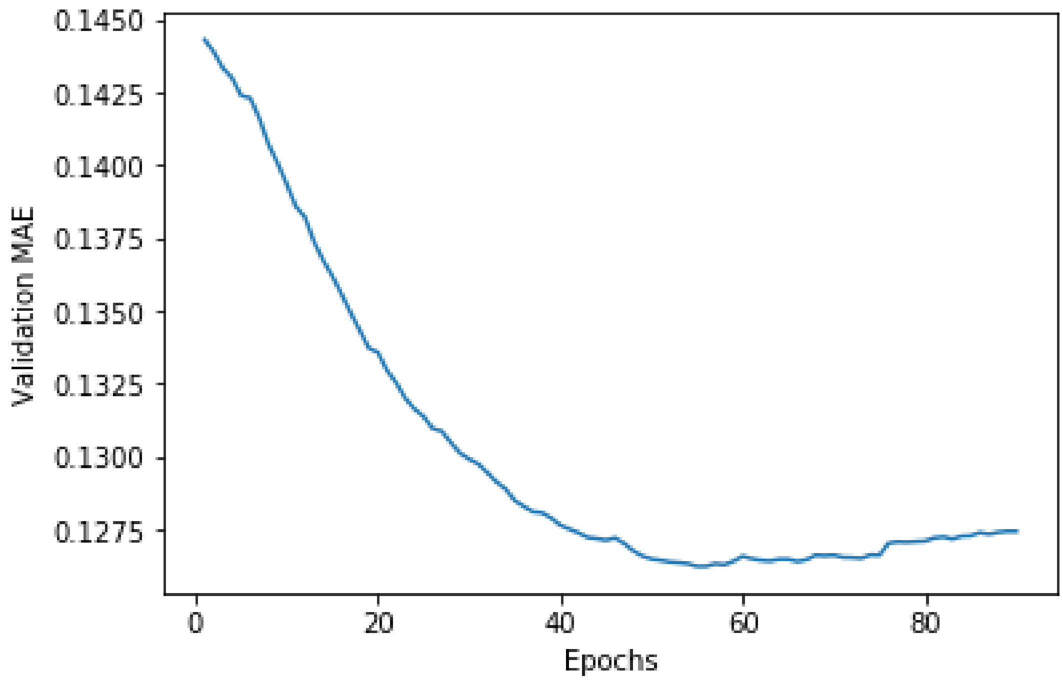



Error rate

```
599/599 [==============================] - 0s 15us/step
599/599 [==============================] - 0s 70us/step
0.1087832823395729
```

**Sgd and mse**

```
Epoch 95/100
1163/1163 [==============================] - 0s 30us/step - loss: 0.0065 -
mae: 0.0594 - val_loss: 0.0323 - val_mae: 0.1330
Epoch 96/100
1163/1163 [==============================] - 0s 32us/step - loss: 0.0064 -
mae: 0.0586 - val_loss: 0.0327 - val_mae: 0.1341
Epoch 97/100
1163/1163 [==============================] - ETA: 0s - loss: 0.0040 - mae:
0.053 - 0s 29us/step - loss: 0.0063 - mae: 0.0580 - val_loss: 0.0321 - val
_mae: 0.1323
Epoch 98/100
1163/1163 [==============================] - 0s 29us/step - loss: 0.0063 -
mae: 0.0577 - val_loss: 0.0337 - val_mae: 0.1346
Epoch 99/100
1163/1163 [==============================] - 0s 32us/step - loss: 0.0063 -
mae: 0.0579 - val_loss: 0.0343 - val_mae: 0.1358
Epoch 100/100
1163/1163 [==============================] - 0s 30us/step - loss: 0.0062 -
mae: 0.0574 - val_loss: 0.0317 - val_mae: 0.1348
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.1428678184747696
```

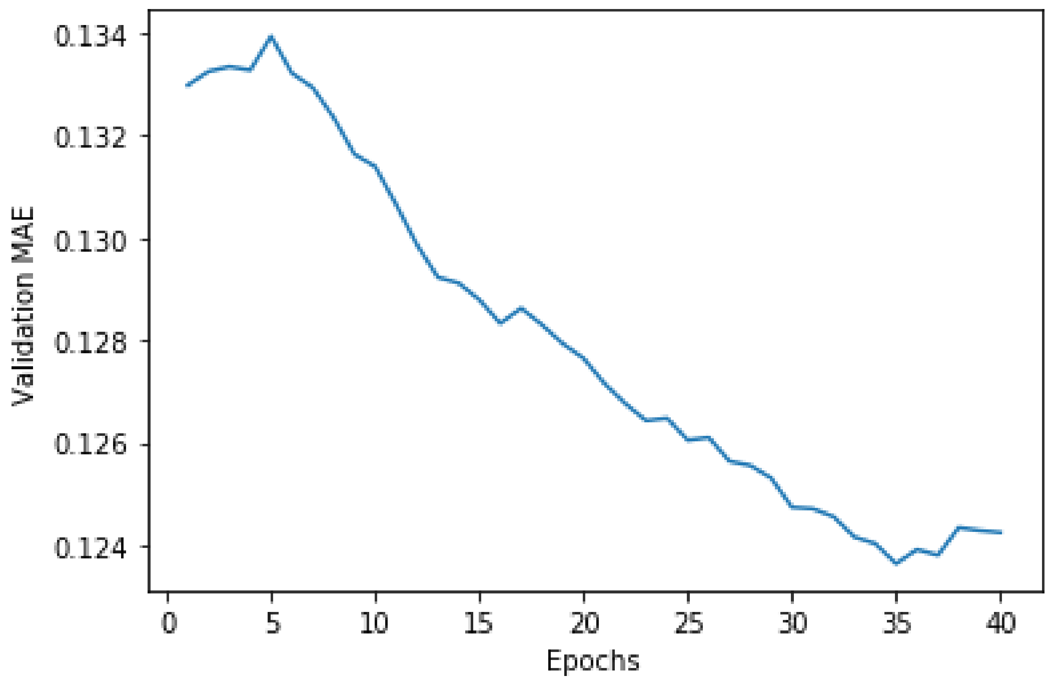
**Sgd and mae**

Hyper parameter updates:

Decrease in fold K= 5

Decrease in Epoch = 50

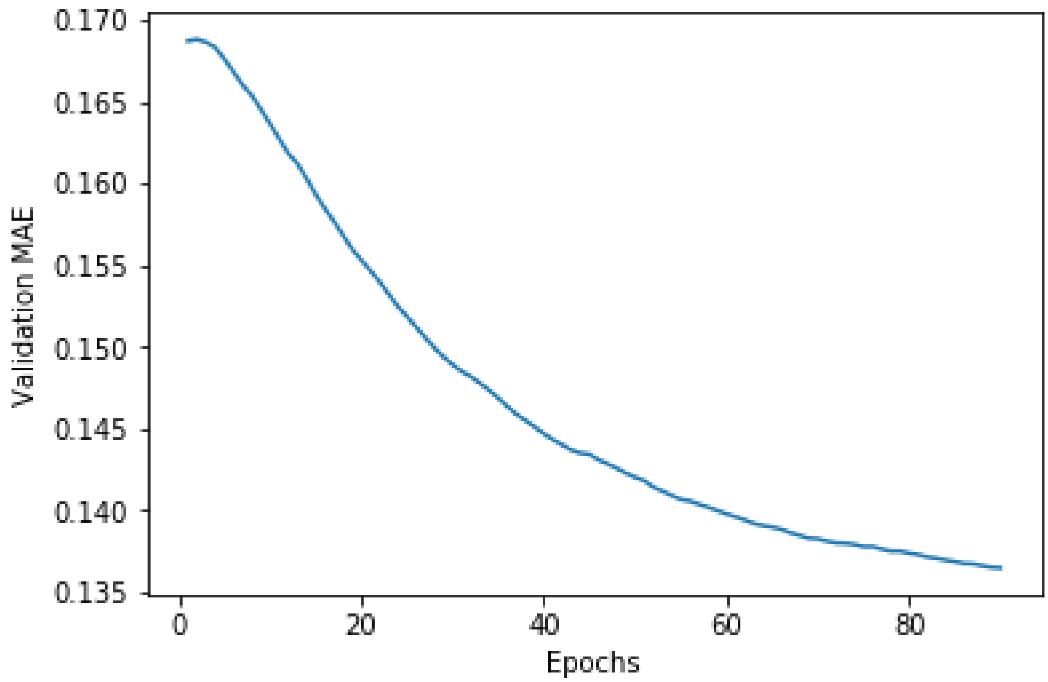
```
Epoch 45/50
1116/1116 [==============================] - 0s 33us/step - loss: 0.0623 - mae: 0.0623 - val_loss: 0.1295 - val_mae: 0.1295
Epoch 46/50
1116/1116 [==============================] - 0s 34us/step - loss: 0.0583 - mae: 0.0583 - val_loss: 0.1389 - val_mae: 0.1389
Epoch 47/50
1116/1116 [==============================] - 0s 37us/step - loss: 0.0588 - mae: 0.0588 - val_loss: 0.1378 - val_mae: 0.1378
Epoch 48/50
1116/1116 [==============================] - 0s 38us/step - loss: 0.0613 - mae: 0.0613 - val_loss: 0.1431 - val_mae: 0.1431
Epoch 49/50
1116/1116 [==============================] - 0s 34us/step - loss: 0.0589 - mae: 0.0589 - val_loss: 0.1336 - val_mae: 0.1336
Epoch 50/50
1116/1116 [==============================] - 0s 33us/step - loss: 0.0574 - mae: 0.0574 - val_loss: 0.1391 - val_mae: 0.1391
```




```
Error rate
599/599 [==============================] - 0s 12us/step
599/599 [==============================] - 0s 27us/step
0.11749739944934845
```

**Sgd and logcosh**

```
Epoch 95/100
1163/1163 [==============================] - 0s 38us/step - loss: 0.0054 -
mae: 0.0773 - val_loss: 0.0138 - val_mae: 0.1215
Epoch 96/100
1163/1163 [==============================] - 0s 33us/step - loss: 0.0054 -
mae: 0.0769 - val_loss: 0.0131 - val_mae: 0.1209
Epoch 97/100
1163/1163 [==============================] - 0s 34us/step - loss: 0.0053 -
mae: 0.0770 - val_loss: 0.0133 - val_mae: 0.1201
Epoch 98/100
1163/1163 [==============================] - 0s 34us/step - loss: 0.0053 -
mae: 0.0767 - val_loss: 0.0132 - val_mae: 0.1210
Epoch 99/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0053 -
mae: 0.0767 - val_loss: 0.0131 - val_mae: 0.1199
Epoch 100/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.0052 -
mae: 0.0763 - val_loss: 0.0131 - val_mae: 0.1199
```



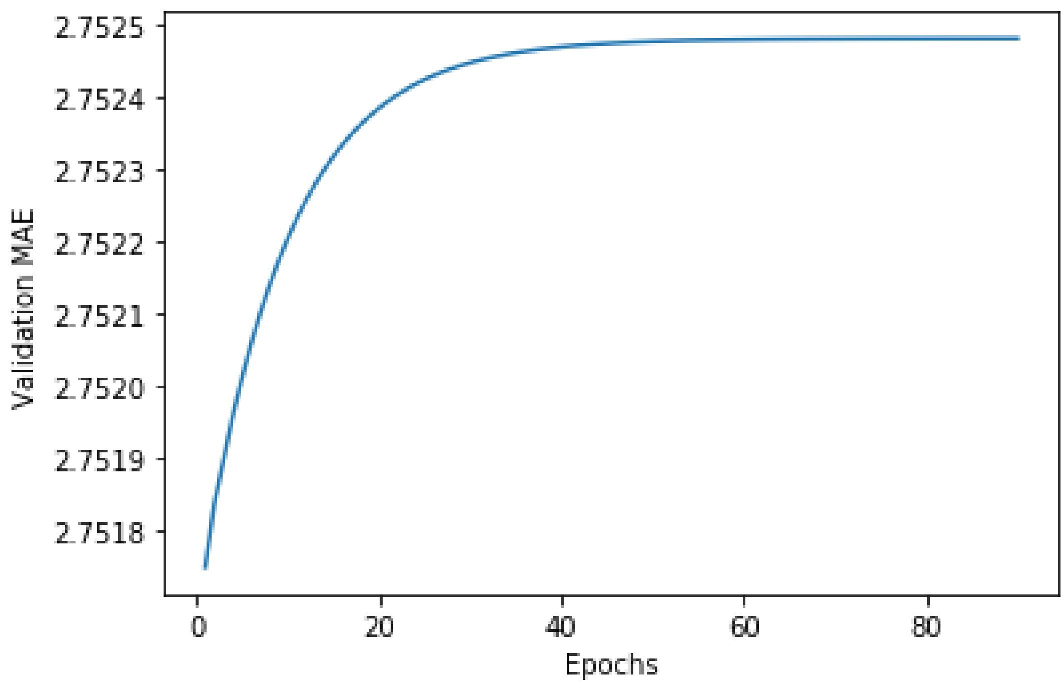

```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.11898112297058105
```

**Sgd and cross entropy**

**Hyper parameter change:**

Decrease in fold K = 2

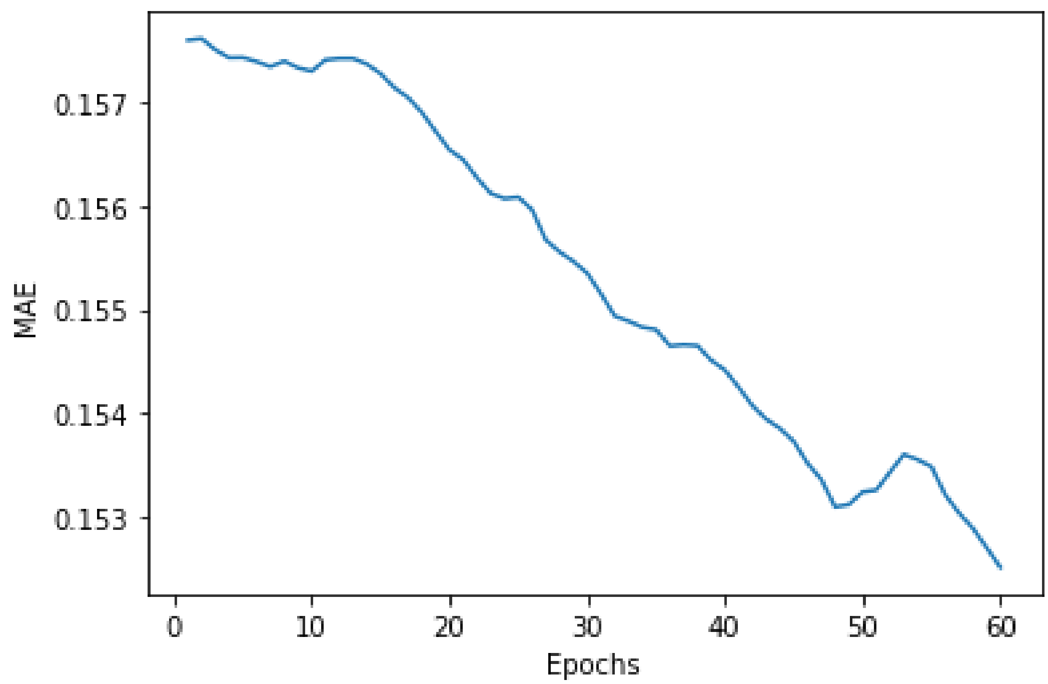
```
Epoch 95/100
698/698 [==============================] - 0s 53us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
Epoch 96/100
698/698 [==============================] - 0s 53us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
Epoch 97/100
698/698 [==============================] - 0s 50us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
Epoch 98/100
698/698 [==============================] - 0s 60us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
Epoch 99/100
698/698 [==============================] - 0s 53us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
Epoch 100/100
698/698 [==============================] - 0s 60us/step - loss: 3.6937 - m
ae: 2.1296 - val_loss: 3.7524 - val_mae: 2.1582
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 32us/step
2.1302602291107178
```

**Sgd and huber**

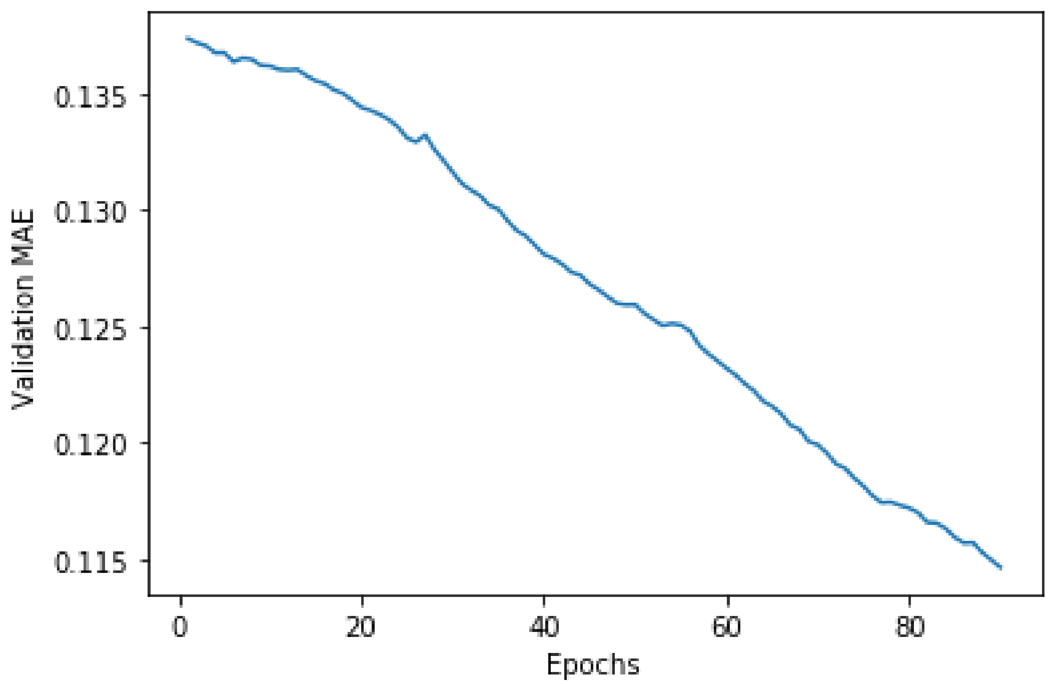
```
Epoch 95/100
1163/1163 [==============================] - 0s 56us/step - loss: 0.0250 -
mae: 0.1689 - val_loss: 0.0246 - val_mae: 0.1564
Epoch 96/100
1163/1163 [==============================] - 0s 61us/step - loss: 0.0238 -
mae: 0.1666 - val_loss: 0.0244 - val_mae: 0.1567
Epoch 97/100
1163/1163 [==============================] - 0s 56us/step - loss: 0.0243 -
mae: 0.1674 - val_loss: 0.0244 - val_mae: 0.1549
Epoch 98/100
1163/1163 [==============================] - 0s 55us/step - loss: 0.0246 -
mae: 0.1687 - val_loss: 0.0245 - val_mae: 0.1602
Epoch 99/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0250 -
mae: 0.1714 - val_loss: 0.0248 - val_mae: 0.1624
Epoch 100/100
1163/1163 [==============================] - 0s 56us/step - loss: 0.0252 -
mae: 0.1715 - val_loss: 0.0250 - val_mae: 0.1617
```




```
Error rate
599/599 [==============================] - 0s 20us/step
599/599 [==============================] - 0s 72us/step
0.15686504542827606
```

**Adam and mse**

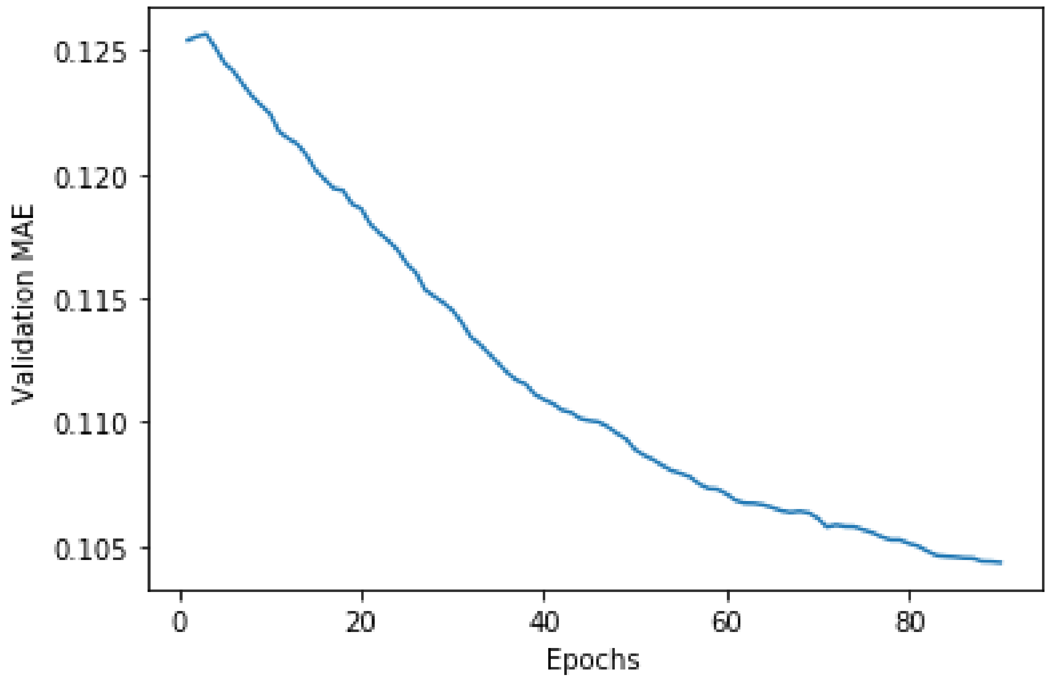
```
Epoch 95/100
1163/1163 [==============================] - 0s 38us/step - loss: 0.0011 -
mae: 0.0246 - val_loss: 0.0282 - val_mae: 0.1228
Epoch 96/100
1163/1163 [==============================] - 0s 37us/step - loss: 0.0013 -
mae: 0.0252 - val_loss: 0.0281 - val_mae: 0.1210
Epoch 97/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0013 -
mae: 0.0274 - val_loss: 0.0269 - val_mae: 0.1198
Epoch 98/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.0012 -
mae: 0.0248 - val_loss: 0.0265 - val_mae: 0.1173
Epoch 99/100
1163/1163 [==============================] - 0s 34us/step - loss: 9.7445e-
04 - mae: 0.0231 - val_loss: 0.0273 - val_mae: 0.1203
Epoch 100/100
1163/1163 [==============================] - 0s 37us/step - loss: 0.0013 -
mae: 0.0256 - val_loss: 0.0269 - val_mae: 0.1200
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.11480706185102463
```

**Adam and mae**

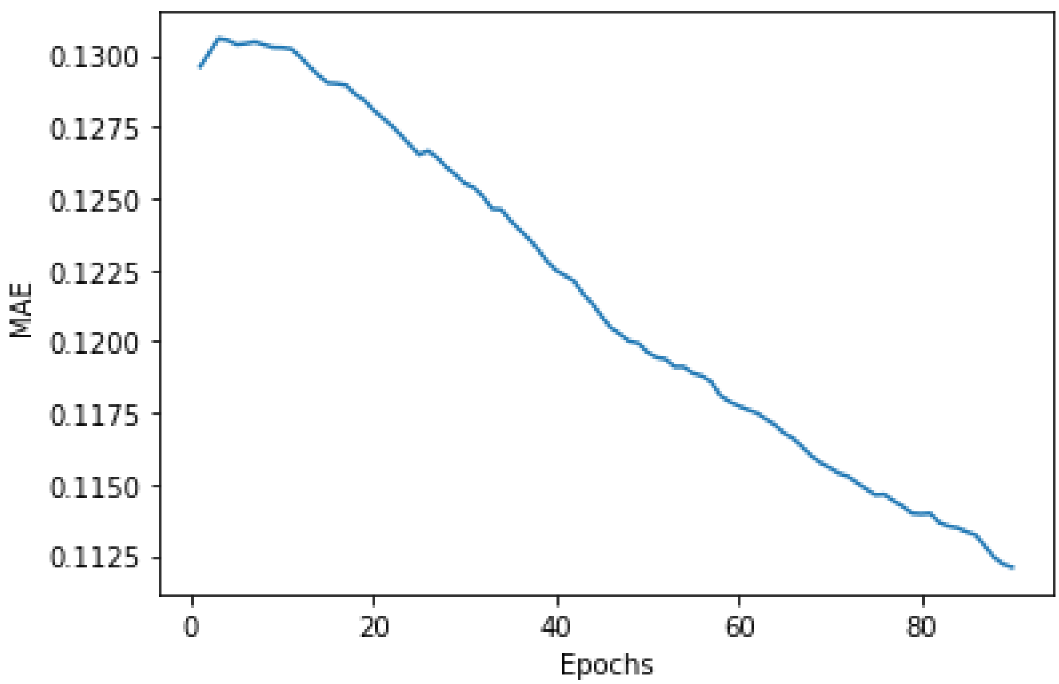
```
Epoch 95/100
1163/1163 [==============================] - 0s 41us/step - loss: 0.0294 -
mae: 0.0294 - val_loss: 0.0935 - val_mae: 0.0935
Epoch 96/100
1163/1163 [==============================] - 0s 39us/step - loss: 0.0302 -
mae: 0.0302 - val_loss: 0.0919 - val_mae: 0.0919
Epoch 97/100
1163/1163 [==============================] - 0s 36us/step - loss: 0.0301 -
mae: 0.0301 - val_loss: 0.0915 - val_mae: 0.0915
Epoch 98/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0303 -
mae: 0.0303 - val_loss: 0.0910 - val_mae: 0.0910
Epoch 99/100
1163/1163 [==============================] - 0s 35us/step - loss: 0.0283 -
mae: 0.0283 - val_loss: 0.0930 - val_mae: 0.0930
Epoch 100/100
1163/1163 [==============================] - ETA: 0s - loss: 0.0294 - mae:
0.029 - 0s 39us/step - loss: 0.0290 - mae: 0.0290 - val_loss: 0.0931 - val
_mae: 0.0931
```




```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 27us/step
0.10775543004274368
```

**Adam and logcosh**

```
Epoch 95/100
1163/1163 [==============================] - 0s 56us/step - loss: 0.0010 -
mae: 0.0348 - val_loss: 0.0117 - val_mae: 0.1072
Epoch 96/100
1163/1163 [==============================] - 0s 41us/step - loss: 7.9008e-
04 - mae: 0.0297 - val_loss: 0.0121 - val_mae: 0.1060
Epoch 97/100
1163/1163 [==============================] - 0s 47us/step - loss: 6.0772e-
04 - mae: 0.0257 - val_loss: 0.0115 - val_mae: 0.1053
Epoch 98/100
1163/1163 [==============================] - 0s 43us/step - loss: 6.3703e-
04 - mae: 0.0263 - val_loss: 0.0123 - val_mae: 0.1079
Epoch 99/100
1163/1163 [==============================] - 0s 45us/step - loss: 5.6209e-
04 - mae: 0.0240 - val_loss: 0.0119 - val_mae: 0.1048
Epoch 100/100
1163/1163 [==============================] - 0s 42us/step - loss: 4.4869e-
04 - mae: 0.0216 - val_loss: 0.0120 - val_mae: 0.1064
```




```
Error rate
599/599 [==============================] - 0s 15us/step
599/599 [==============================] - 0s 33us/step
0.11622636020183563
```

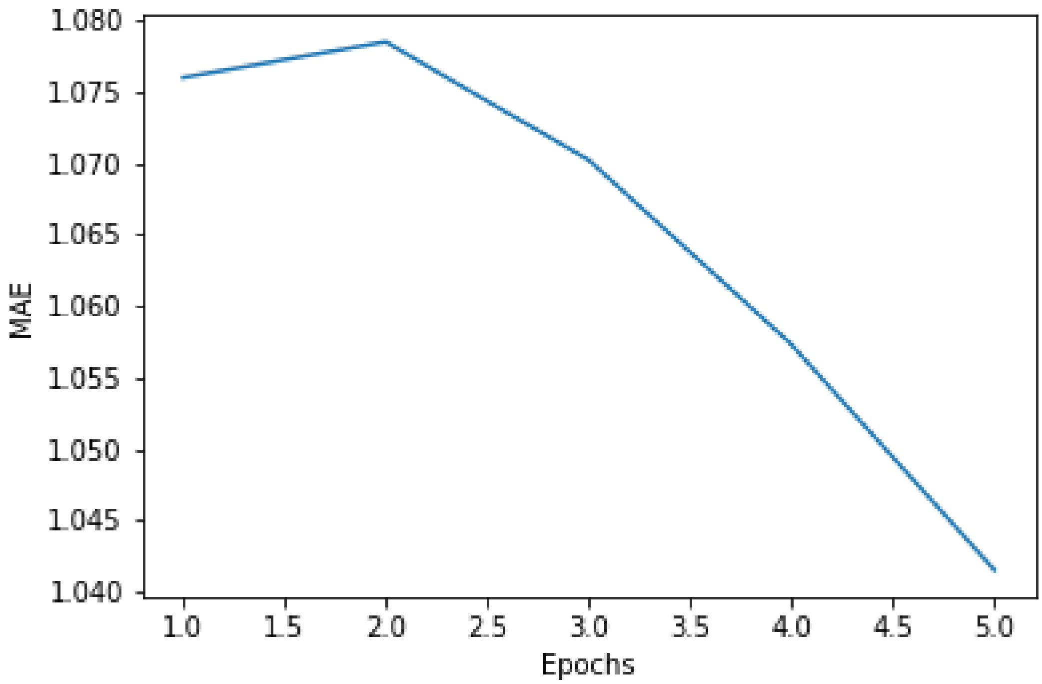
**Adam and cross entropy**

**Hyper parameter change:**

Decrease in fold K = 5

Decrease in Epoch = 50

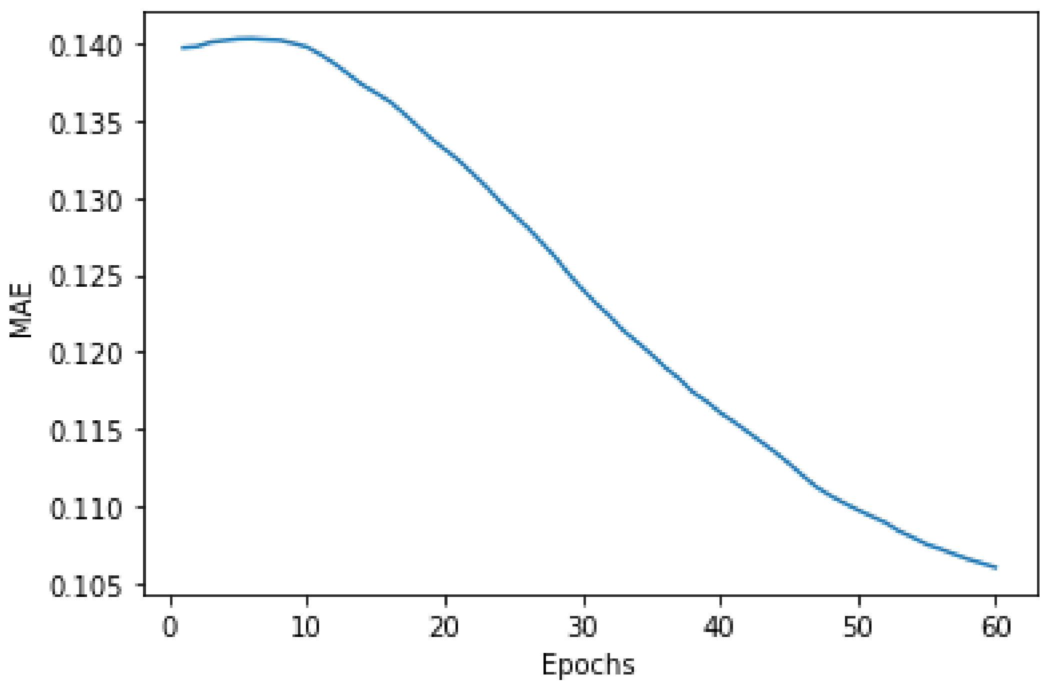
```
Epoch 46/50
1116/1116 [==============================] - 0s 45us/step - loss: 3.7253 -
mae: 2.0813 - val_loss: 3.4432 - val_mae: 2.0152
Epoch 47/50
1116/1116 [==============================] - 0s 43us/step - loss: 3.7253 -
mae: 2.0813 - val_loss: 3.4432 - val_mae: 2.0152
Epoch 48/50
1116/1116 [==============================] - 0s 41us/step - loss: 3.7253 -
mae: 2.0813 - val_loss: 3.4432 - val_mae: 2.0152
Epoch 49/50
1116/1116 [==============================] - 0s 39us/step - loss: 3.7253 -
mae: 2.0813 - val_loss: 3.4432 - val_mae: 2.0152
Epoch 50/50
1116/1116 [==============================] - 0s 44us/step - loss: 3.7253 -
mae: 2.0813 - val_loss: 3.4432 - val_mae: 2.0152
```




```
Error rate
599/599 [==============================] - 0s 17us/step
599/599 [==============================] - 0s 32us/step
1.9877111911773682
```

**Adam and huber**

```
Epoch 96/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0173 -
mae: 0.1326 - val_loss: 0.0097 - val_mae: 0.0956
Epoch 97/100
1163/1163 [==============================] - 0s 60us/step - loss: 0.0189 -
mae: 0.1345 - val_loss: 0.0090 - val_mae: 0.0952
Epoch 98/100
1163/1163 [==============================] - 0s 61us/step - loss: 0.0190 -
mae: 0.1409 - val_loss: 0.0092 - val_mae: 0.0944
Epoch 99/100
1163/1163 [==============================] - 0s 59us/step - loss: 0.0183 -
mae: 0.1359 - val_loss: 0.0094 - val_mae: 0.0958
Epoch 100/100
1163/1163 [==============================] - 0s 60us/step - loss: 0.0172 -
mae: 0.1328 - val_loss: 0.0094 - val_mae: 0.0963
```




```
Error rate
599/599 [==============================] - 0s 17us/step
599/599 [==============================] - 0s 73us/step
0.11896853893995285
```

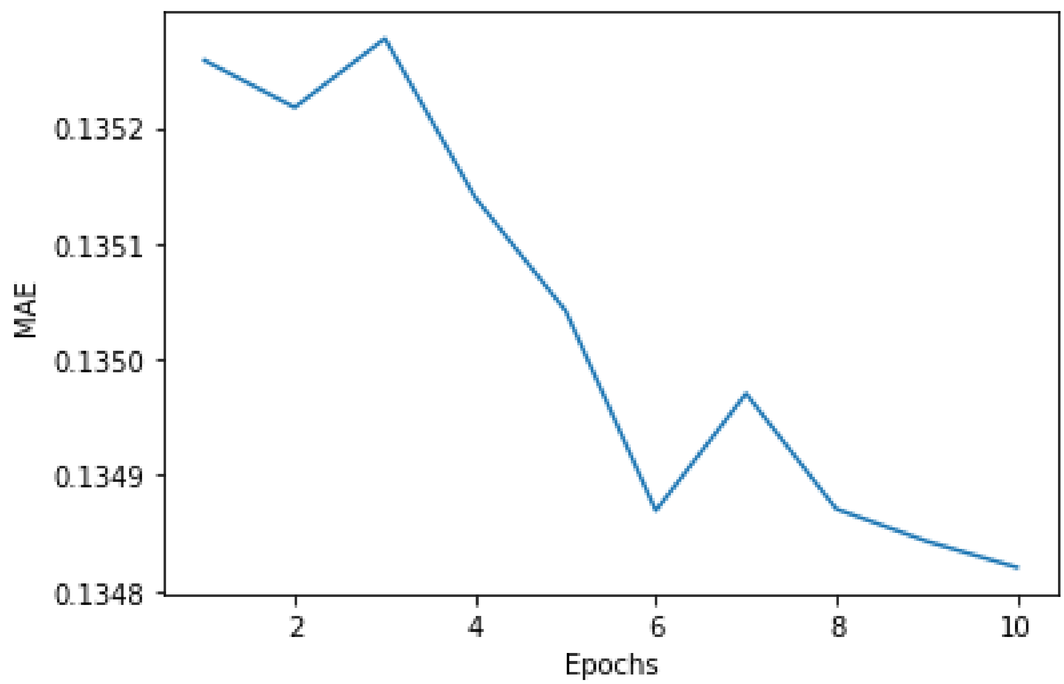
**Adagrad and mse**

**Hyper parameter change:**

Decrease in Epoch = 20

Decrease in fold K = 5

```
Epoch 16/20
1116/1116 [==============================] - 0s 38us/step - loss: 0.0057 -
mae: 0.0558 - val_loss: 0.0289 - val_mae: 0.1229
Epoch 17/20
1116/1116 [==============================] - 0s 37us/step - loss: 0.0054 -
mae: 0.0541 - val_loss: 0.0316 - val_mae: 0.1278
Epoch 18/20
1116/1116 [==============================] - 0s 38us/step - loss: 0.0050 -
mae: 0.0526 - val_loss: 0.0287 - val_mae: 0.1228
Epoch 19/20
1116/1116 [==============================] - 0s 36us/step - loss: 0.0046 -
mae: 0.0501 - val_loss: 0.0306 - val_mae: 0.1263
Epoch 20/20
1116/1116 [==============================] - 0s 35us/step - loss: 0.0044 -
mae: 0.0489 - val_loss: 0.0324 - val_mae: 0.1280
```



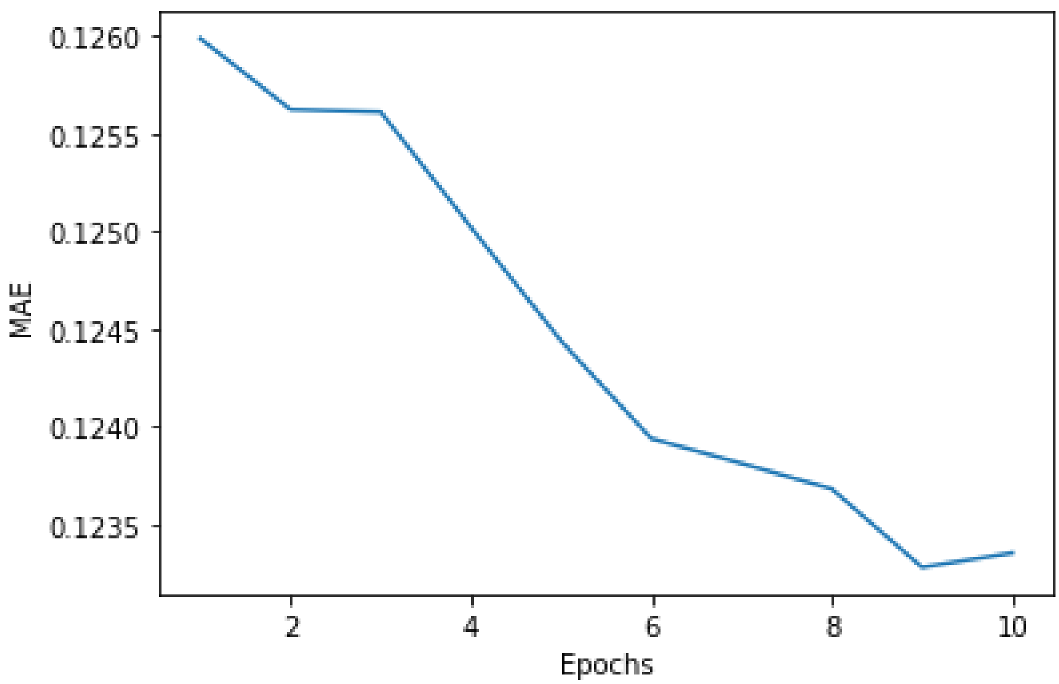

```
Error rate
599/599 [==============================] - 0s 13us/step
599/599 [==============================] - 0s 25us/step
0.13322484493255615
```

**Adagrad and mae**

**Hyper parameter change:**

Decrease in Epoch = 20

```
Epoch 16/20
1163/1163 [==============================] - 0s 33us/step - loss: 0.0481 - mae
: 0.0481 - val_loss: 0.1053 - val_mae: 0.1053
Epoch 17/20
1163/1163 [==============================] - 0s 34us/step - loss: 0.0457 - mae
: 0.0457 - val_loss: 0.1234 - val_mae: 0.1234
Epoch 18/20
1163/1163 [==============================] - 0s 33us/step - loss: 0.0449 - mae
: 0.0449 - val_loss: 0.1062 - val_mae: 0.1062
Epoch 19/20
1163/1163 [==============================] - 0s 36us/step - loss: 0.0429 - mae
: 0.0429 - val_loss: 0.1052 - val_mae: 0.1052
Epoch 20/20
1163/1163 [==============================] - 0s 35us/step - loss: 0.0427 - mae
: 0.0427 - val_loss: 0.1197 - val_mae: 0.1197
```



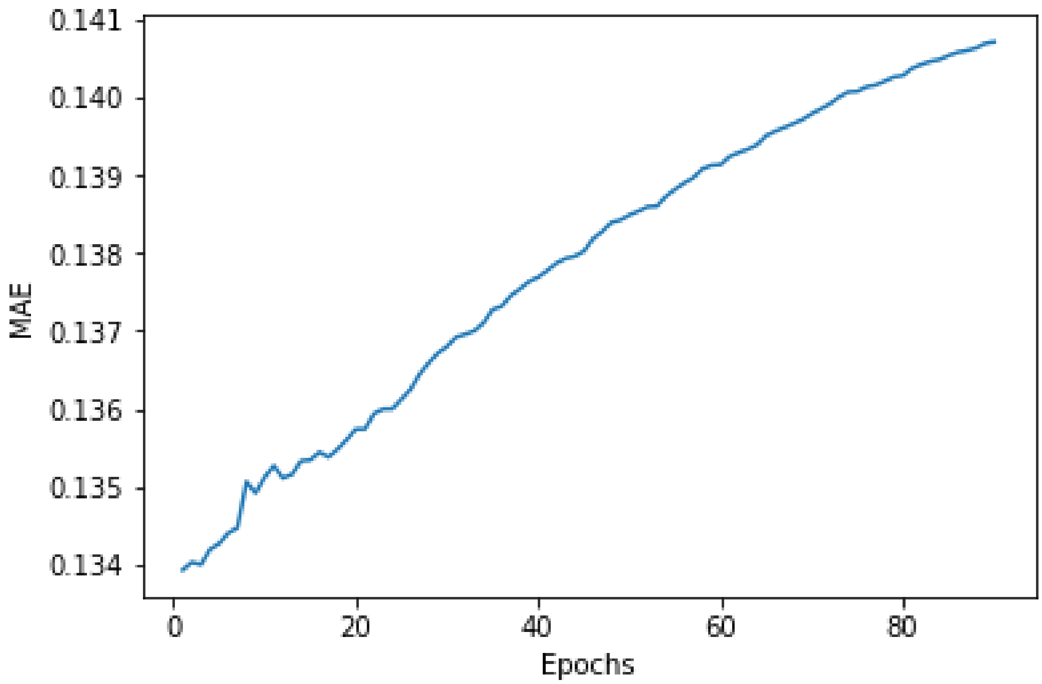

```
Error rate
599/599 [==============================] - 0s 15us/step
599/599 [==============================] - 0s 25us/step
0.12620513141155243
```

**Adagrad and logcosh**

**Hyper parameter change:**

Decrease in fold K = 4

```
Epoch 95/100
1047/1047 [==============================] - 0s 41us/step - loss: 3.7956e-
05 - mae: 0.0059 - val_loss: 0.0177 - val_mae: 0.1422
Epoch 96/100
1047/1047 [==============================] - 0s 38us/step - loss: 3.6947e-
05 - mae: 0.0058 - val_loss: 0.0176 - val_mae: 0.1415
Epoch 97/100
1047/1047 [==============================] - 0s 40us/step - loss: 3.4765e-
05 - mae: 0.0055 - val_loss: 0.0177 - val_mae: 0.1418
Epoch 98/100
1047/1047 [==============================] - 0s 39us/step - loss: 3.2847e-
05 - mae: 0.0054 - val_loss: 0.0176 - val_mae: 0.1416
Epoch 99/100
1047/1047 [==============================] - 0s 40us/step - loss: 3.2239e-
05 - mae: 0.0054 - val_loss: 0.0177 - val_mae: 0.1420
Epoch 100/100
1047/1047 [==============================] - 0s 44us/step - loss: 3.0464e-
05 - mae: 0.0052 - val_loss: 0.0176 - val_mae: 0.1416
```



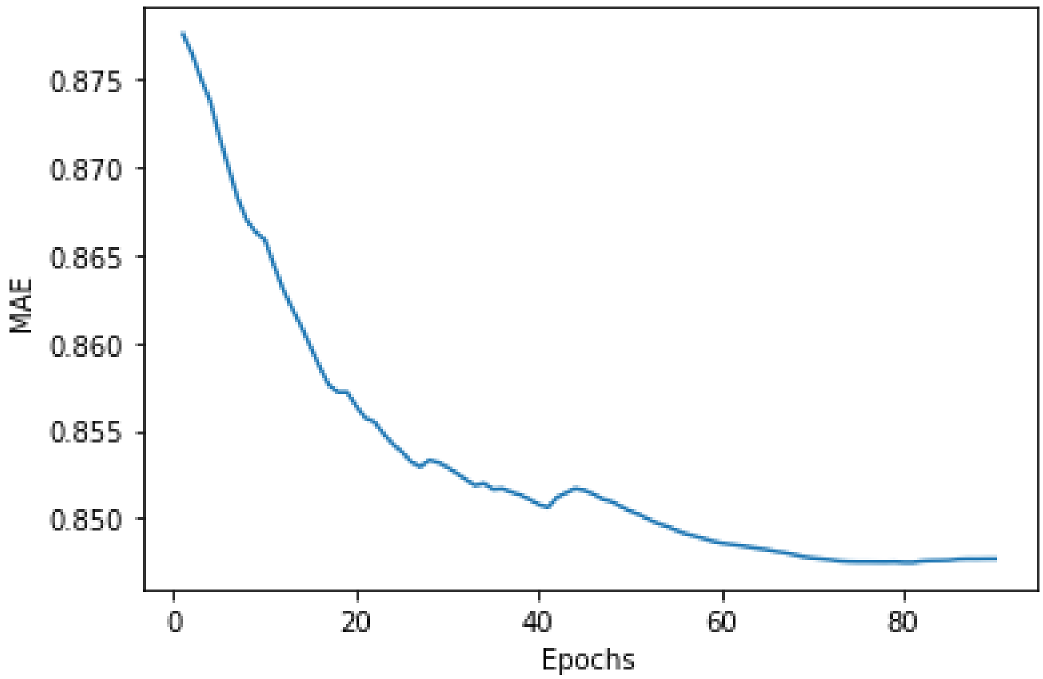

```
Error rate
599/599 [==============================] - 0s 15us/step
599/599 [==============================] - 0s 27us/step
0.1352742612361908
```

**Adagrad and cross entropy**

**Hyper parameter change:**

Increase of layers to 4

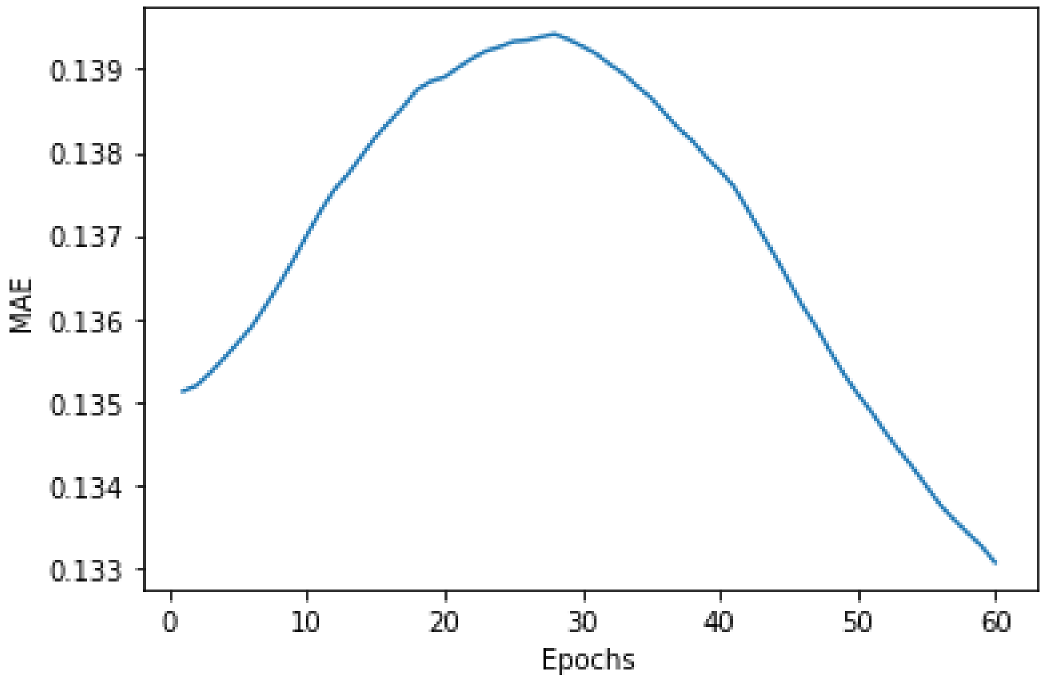
```
Epoch 95/100
1163/1163 [==============================] - 0s 38us/step - loss: 0.4228 -
mae: 0.0471 - val_loss: 0.5103 - val_mae: 0.1152
Epoch 96/100
1163/1163 [==============================] - 0s 39us/step - loss: 0.4226 -
mae: 0.0468 - val_loss: 0.5094 - val_mae: 0.1146
Epoch 97/100
1163/1163 [==============================] - 0s 39us/step - loss: 0.4224 -
mae: 0.0463 - val_loss: 0.5101 - val_mae: 0.1148
Epoch 98/100
1163/1163 [==============================] - 0s 41us/step - loss: 0.4222 -
mae: 0.0461 - val_loss: 0.5097 - val_mae: 0.1145
Epoch 99/100
1163/1163 [==============================] - 0s 45us/step - loss: 0.4220 -
mae: 0.0458 - val_loss: 0.5050 - val_mae: 0.1155
Epoch 100/100
1163/1163 [==============================] - 0s 37us/step - loss: 0.4218 -
mae: 0.0454 - val_loss: 0.5104 - val_mae: 0.1146
```




```
Error rate
599/599 [==============================] - 0s 17us/step
599/599 [==============================] - 0s 33us/step
0.10803474485874176
```

**Adagrad and huber**

```
Epoch 96/100
1163/1163 [==============================] - 0s 61us/step - loss: 0.0213 -
mae: 0.1529 - val_loss: 0.0162 - val_mae: 0.1290
Epoch 97/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0210 -
mae: 0.1516 - val_loss: 0.0162 - val_mae: 0.1295
Epoch 98/100
1163/1163 [==============================] - 0s 62us/step - loss: 0.0208 -
mae: 0.1518 - val_loss: 0.0165 - val_mae: 0.1304
Epoch 99/100
1163/1163 [==============================] - 0s 60us/step - loss: 0.0208 -
mae: 0.1496 - val_loss: 0.0163 - val_mae: 0.1293
Epoch 100/100
1163/1163 [==============================] - 0s 60us/step - loss: 0.0211 -
mae: 0.1530 - val_loss: 0.0162 - val_mae: 0.1297
```




```
Error rate
599/599 [==============================] - 0s 18us/step
599/599 [==============================] - 0s 75us/step
0.12807059288024902
```

**Applying regularization:**

The best combination of optimizer and loss found were rmsprop and mse with an average error of 0.09 (approximately).

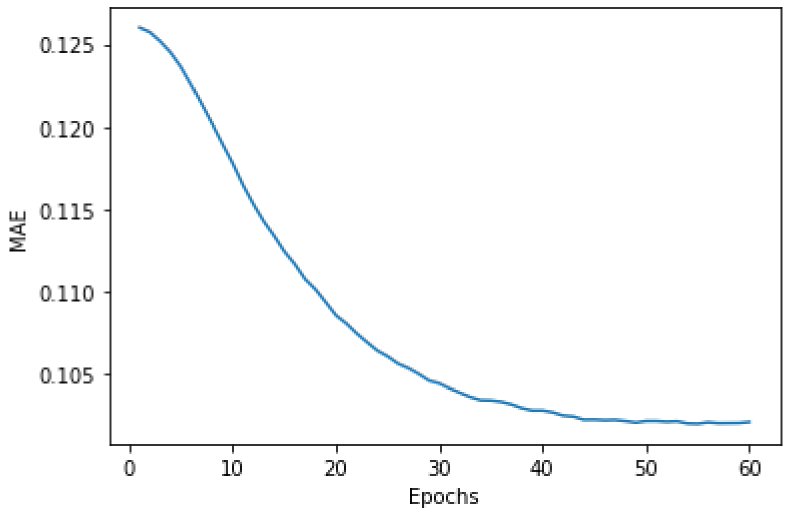
Dropout – A dropout of value 0.8 was added to the layers.

Batch Normalization – The outputs of the hidden layers were normalized before it reached the next layer.

Thus the final output is obtained as:

```
Epoch 96/100
1163/1163 [==============================] - 0s 61us/step - loss: 0.0268 -
mae: 0.1201 - val_loss: 0.0386 - val_mae: 0.1185
Epoch 97/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0270 -
mae: 0.1198 - val_loss: 0.0366 - val_mae: 0.1169
Epoch 98/100
1163/1163 [==============================] - 0s 58us/step - loss: 0.0282 -
mae: 0.1215 - val_loss: 0.0349 - val_mae: 0.1126
Epoch 99/100
1163/1163 [==============================] - 0s 62us/step - loss: 0.0261 -
mae: 0.1184 - val_loss: 0.0401 - val_mae: 0.1193
Epoch 100/100
```

```
1163/1163 [==============================] - 0s 60us/step - loss: 0.0290 - mae: 0.1251 - val_loss: 0.0355 - val_mae: 0.1134
```




```
Error rate
599/599 [==============================] - 0s 18us/step
599/599 [==============================] - 0s 78us/step
0.10245712846517563
```

The model now is identified to converge faster and to give little better result than the previous one.

Ensemble Classifier:

For better optimization an ensemble classifier can be developed by taking the models:

1. Adam and MAE
2. Adagrad and Cross entropy
3. Rmsprop and MSE

Averaging the error rate of all the three models can give an output as follows:

(0.10316283732652664 + 0.10775543004274368 + 0.10803474485874176) / 3

= 0.31895301222801208 / 3

= 0.106 (approx.)

It is found that the ensemble value is not as good as the regularized value. Thus an ensemble classifier is not required.

## Implementation details

Implemented using anaconda and jupyter.

The issues included

1. Large datasets took a longer time.

2. Time consumption due to evaluation against all optimizers and loss.
3. The code was ran on a CPU that consumed more time.
4. Not every model was saved because the models were misinterpreted by the jupyter kernel. Thus only the best model code is provided.
5. Since it was difficult to save multiple models ensemble classifiers were not implemented.

## References:


1. My own assignment 1 code
2. Code given by the professor