Three Meter Autoencoder

I. Network structure

The network structure is as follows:

Decoder: Fully connected layer 3 with Tanh activation, **Fully connected layer 4** with Sigmoid activation.
Encoder: Fully Connected layer 1 with Tanh activation, **Fully connected layer 2**

The encoded data has dimension of 16, which is less than half of the raw data dimension 33. The loss function is Mean Absolute Error (MAE) because this network is doing auto encoding tasks. Only fully connected layer is used in this problem.

II. Representative weights

FC1:

tensor([0.0028, -0.0428, 0.2108, -0.0604, 0.1593, 0.1530, -0.0063, 0.1389,
-0.1000, 0.0116], device='cuda:0', grad_fn=<SliceBackward>)

FC2:

tensor([0.1406, -0.0719, -0.0077, 0.0415, -0.0475, 0.0632, -0.0394, -0.1200,
-0.0470, -0.0944], device='cuda:0', grad_fn=<SliceBackward>)

FC3:

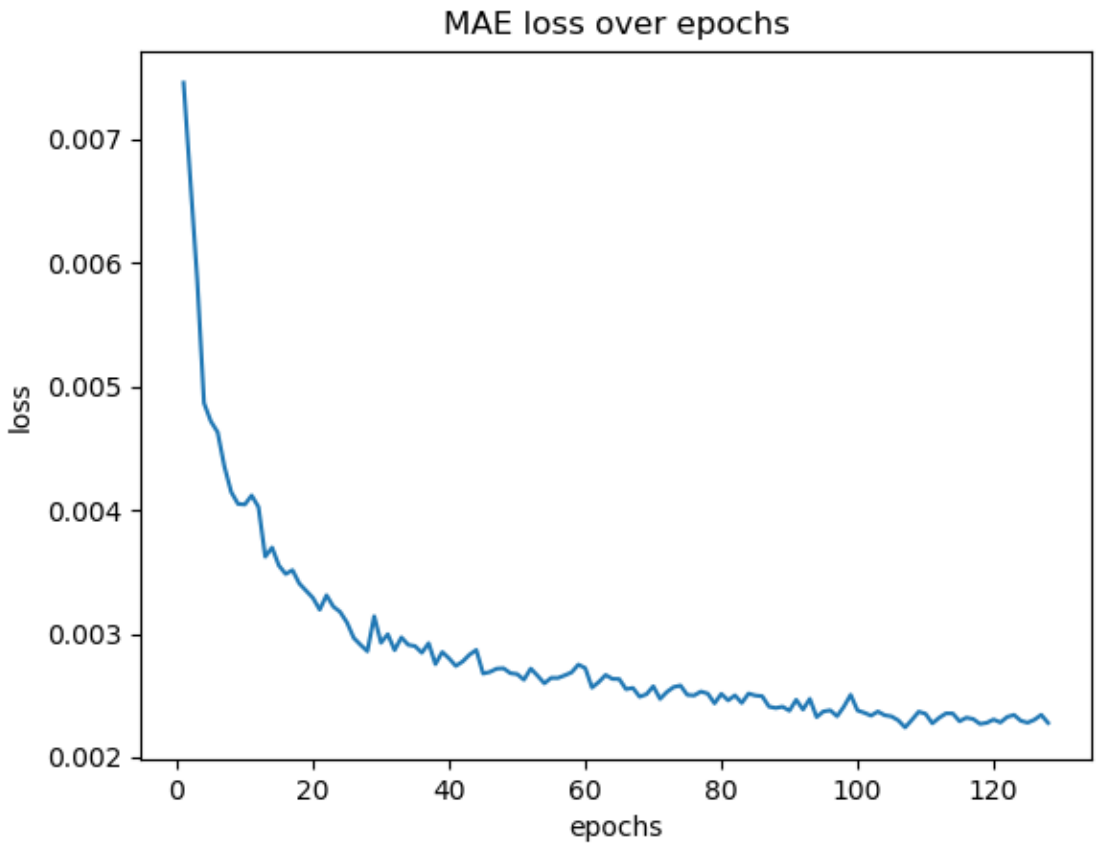
tensor([-0.0430, -0.1439, -0.3121, -0.1949, 0.1079, -0.0793, 0.1844, -0.0070,
-0.2553, 0.0746], device='cuda:0', grad_fn=<SliceBackward>)

FC4:

tensor([-0.0015, 0.0595, 0.0050, 0.0038, 0.0622, 0.0840, 0.0709, -0.0463,
0.0511, -0.0509], device='cuda:0', grad_fn=<SliceBackward>)

As we can see from the above prints, all of the weights are in the range [-1,1], which is a reasonable range. The weights are not tuned to be all zeros or identical, so they are likely to yield a good output.

III. MAE loss vs epoch



The MAE loss is decreasing with the number of epochs, which is expected. The final MAE is below 0.25%.