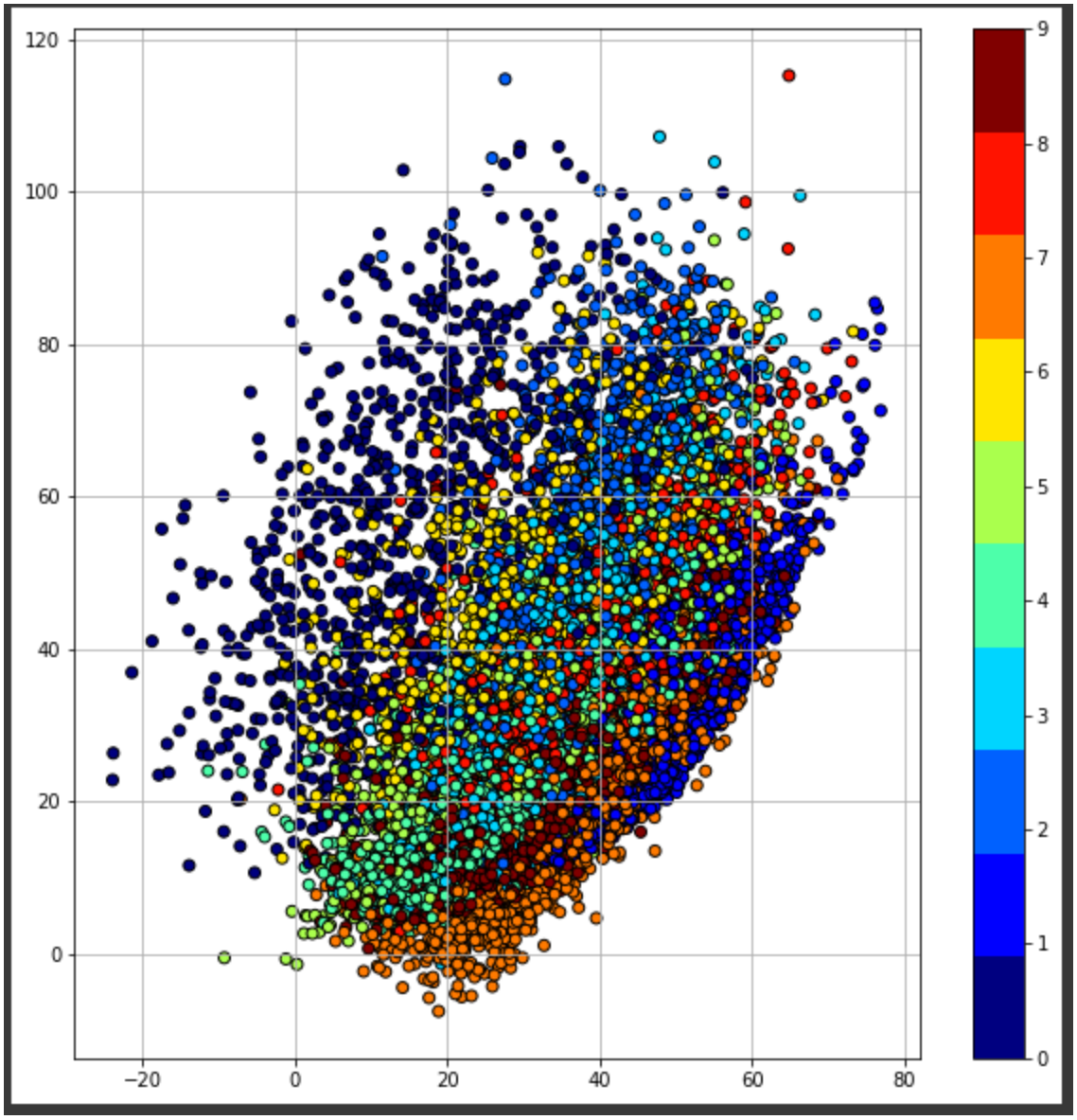
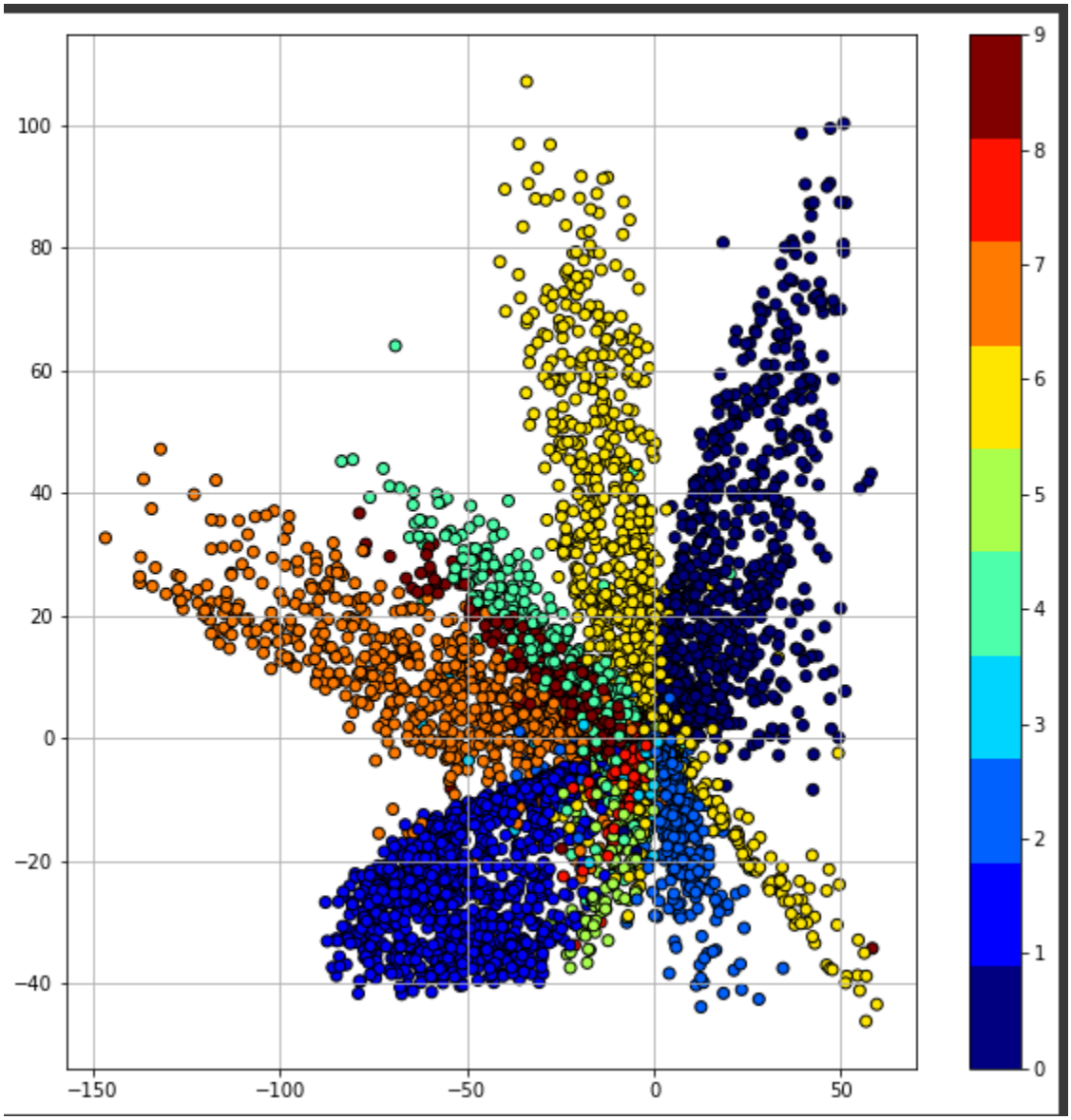
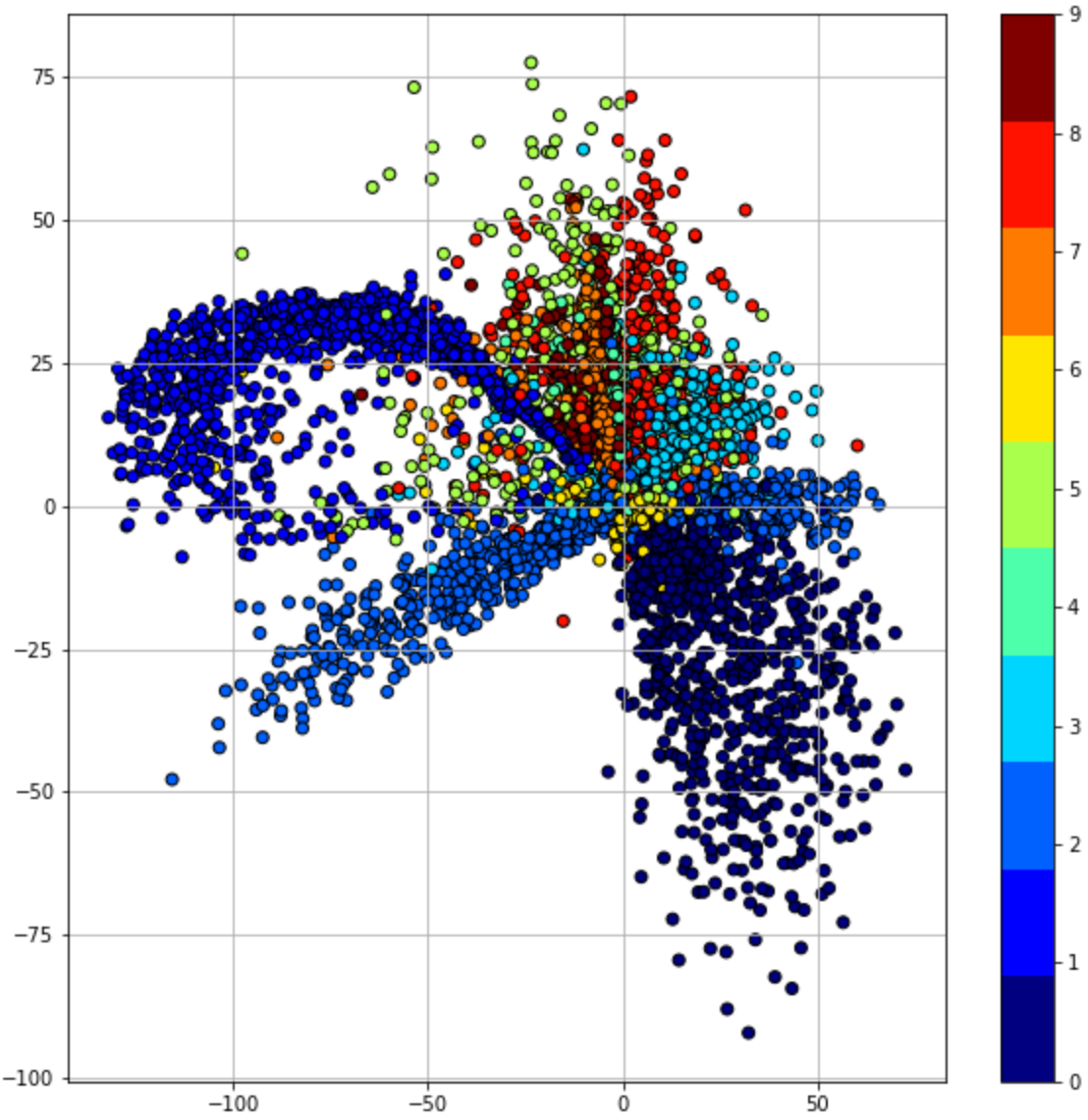


Lr = 1-e3
Epoch = 10
Val Loss = 0.74
(Single Linear Layer)

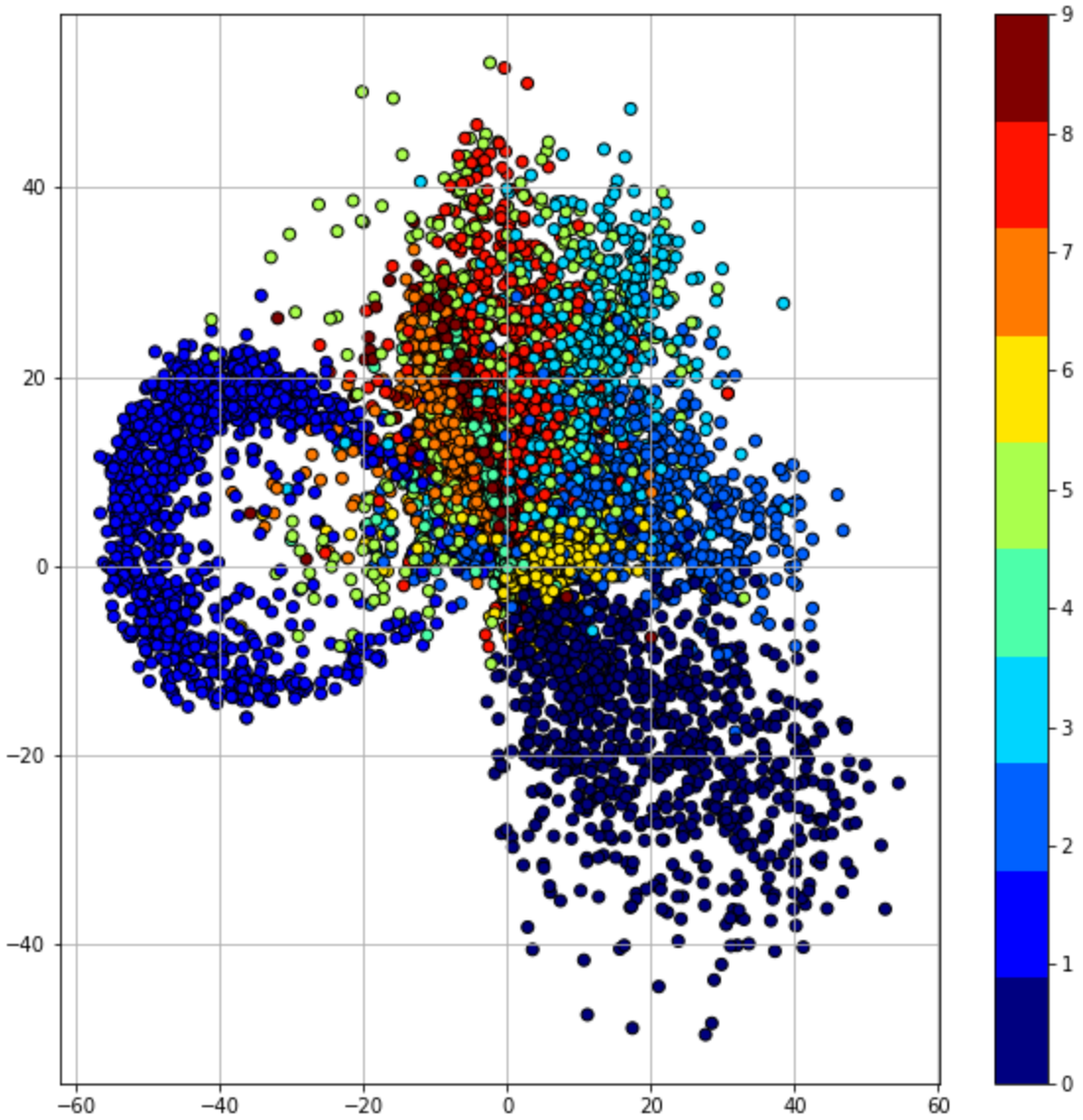
b)



Lr = 1e-3
Epochs = 10
Val loss = 0.6725



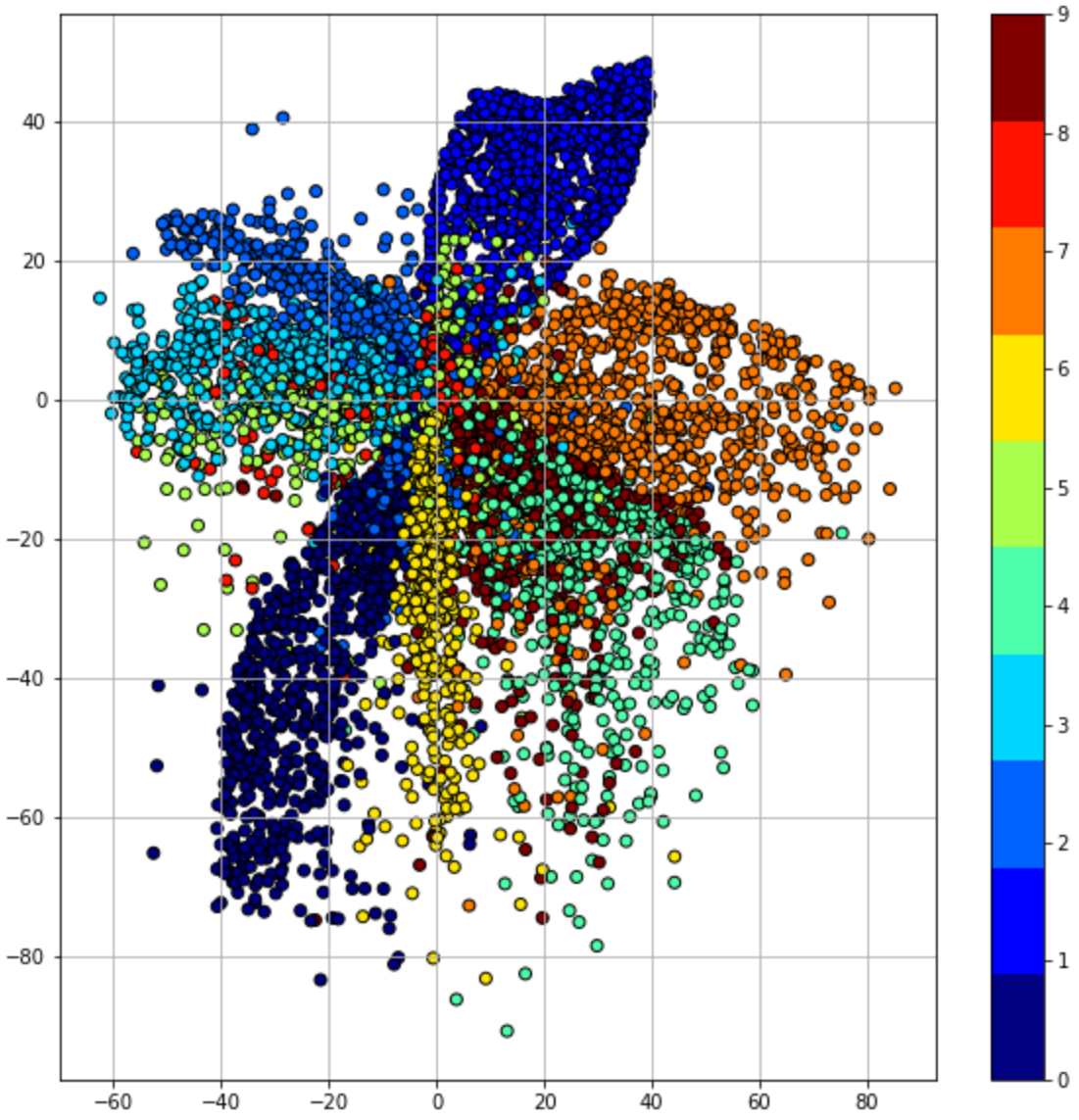
Lr = 1e-4
Epochs = 20
Val Loss : 0.6706



1e-4
5
0.6772
0.6819

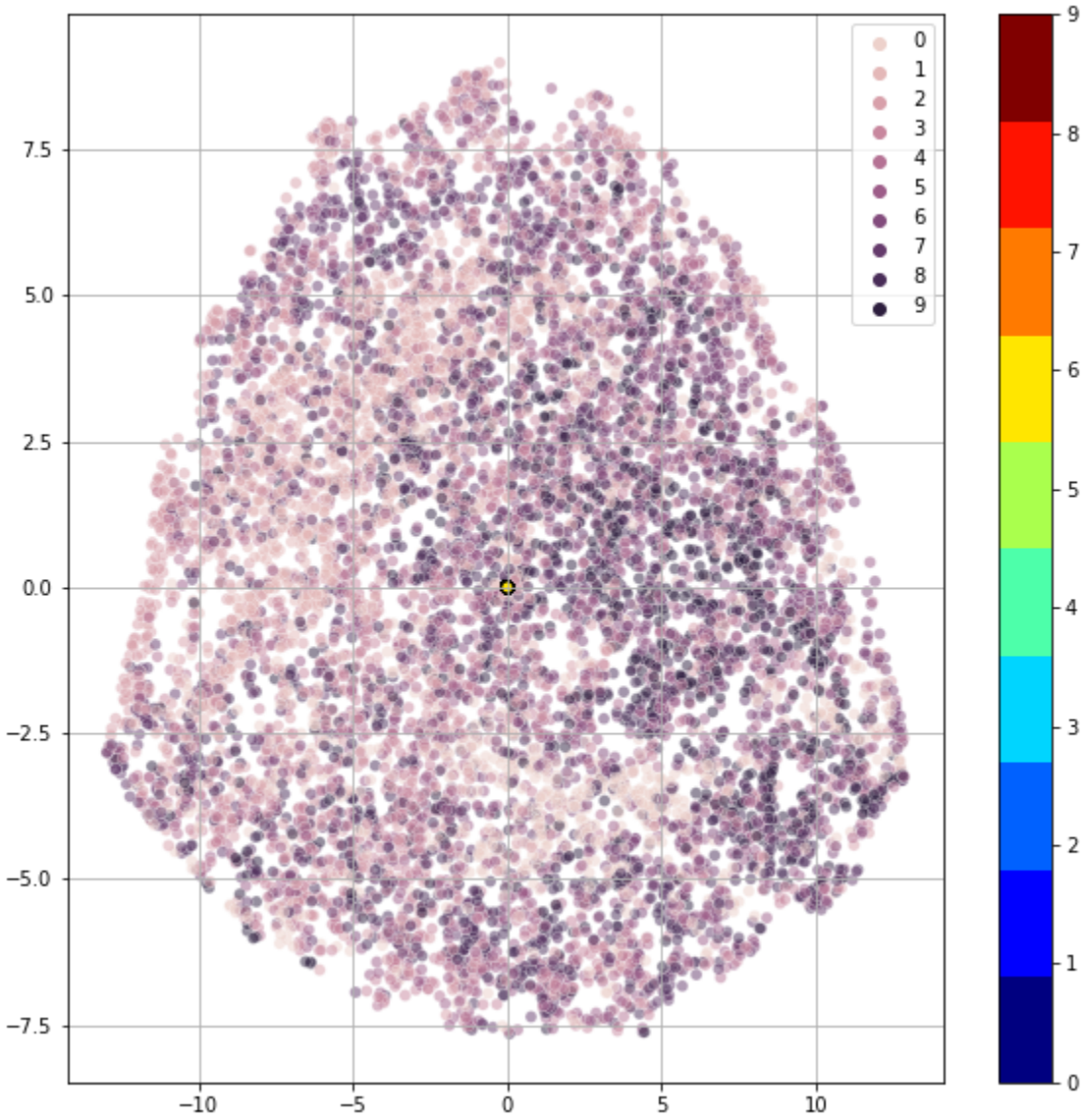
The plot differs because this architecture is more complicated than a single linear layer, It also has an activation function in the encoder as well and it also extrapolates the dense layer 1024 units before inserting into the bottleneck. The plot in a) is unidirectional and does not seem to curve around like the the points on b) do, it is able to capture dimensions moreso. We can also see a significant difference in the reconstructed images. However adding more layers means more computational complexity, the change in val loss is not significant and only lowers from 0.73 to 0.67.

c)



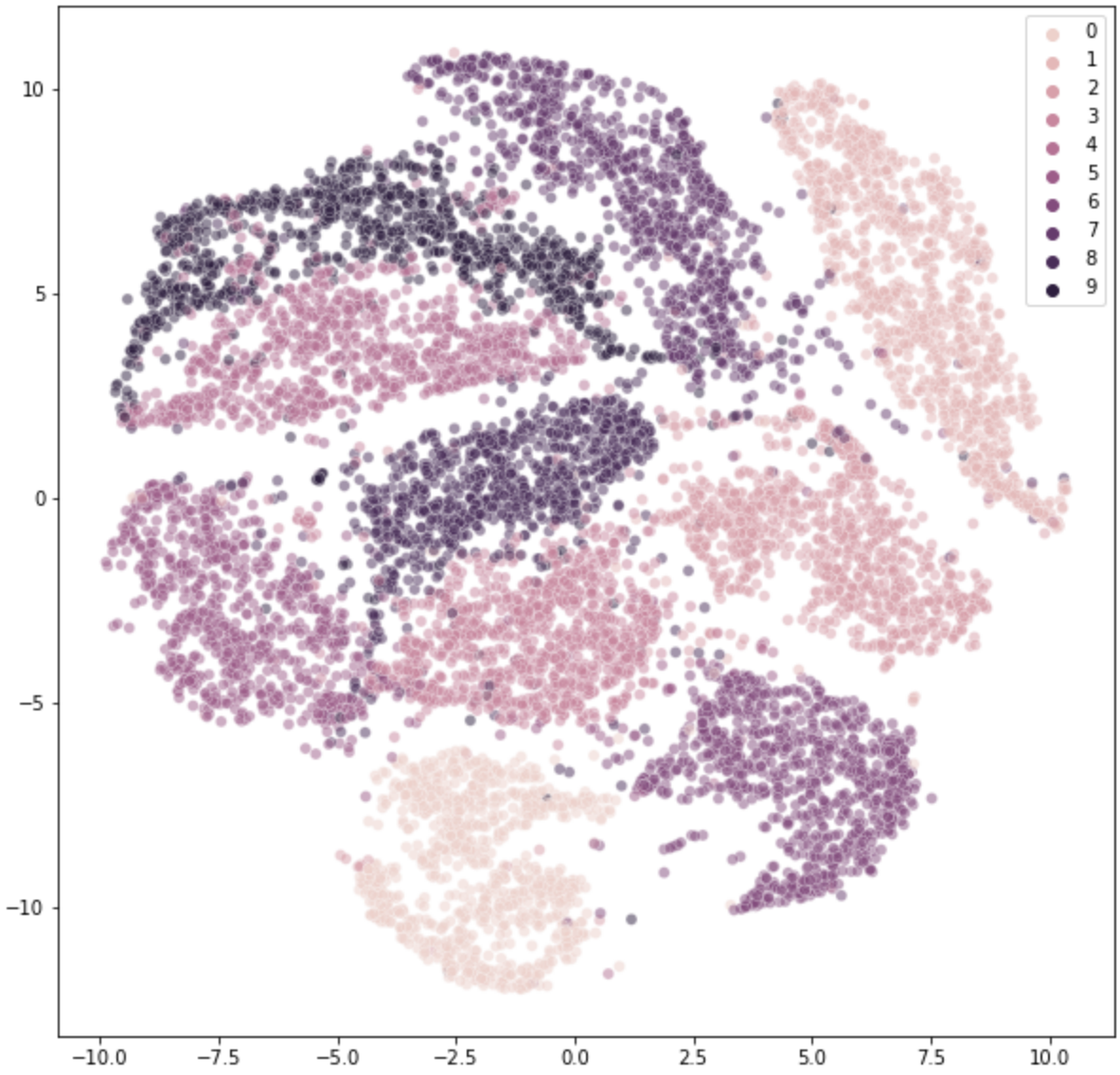
With sigmoid instead of ReLU
LR = 1e-3
Epoch = 10
Val Loss = 0.68

d) TSNE in 2 dims



Autoencoder is responsible for dimensionality reduction, Latent space allows the features from the data to be converted into recognizable form which is where the VAE distribution exists compared to vanilla. The data compressed exists in the latent space. KLD allows the sampling terms to stick to the mean since they were essentially inserted from the middle of the dataset. Tt seemed that smaller weight have more tightly grouped classes

Without kld (VANILLA)


0
1
2
3
4
5
6
7
8
9
10
5
0
−5
−10
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0