HOMEWORK 2

Nikhil Malik
nmalik1@andrew.cmu.edu

Problem 1 (10 pts)

Loss Function

$$J = -log(y_m) \tag{1}$$

$$\frac{\partial J}{\partial \mathbf{y}} = [0, 0..., -1/y_m, ...0] \tag{2}$$

$$\tag{3}$$

Softmax Normalization

$$y_m = exp(x_m)/(\sum_k exp(x_k)) \tag{4}$$

$$\frac{\partial y_i}{\partial x_j} = y_i * [(i == j) - y_j] \tag{5}$$

$$\tag{6}$$

Softmax Weights: Where d_2 is the dimensionality of two post max pool feature maps

$$x_i = \beta_i + \sum_{l=1}^{2} \sum_{e=1}^{d_2} \sum_{f=1}^{d_2} w_{efi}^l * v_{ef}^l \tag{7}$$

$$\frac{\partial x_i}{\partial \beta_i} = 1 \tag{8}$$

$$\frac{\partial x_i}{\partial w_{efi}^l} = v_{ef}^l \tag{9}$$

$$\frac{\partial x_i}{\partial v_{ef}^l} = w_{efi}^l \tag{10}$$

$$\tag{11}$$

Maxpool Operation

$$v_{ef}^l = max((a_{e*p+c,f*p+d}^l)_{d=0}^{p-1})_{c=0}^{p-1} \tag{12}$$

$$\tag{13}$$

$$\frac{\partial v_{ef}^l}{\partial a_{gh}^l} = \begin{cases} 1, & a_{gh}^l \text{ was the max in max pool window .} \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

$$\tag{15}$$

Convolution Operation

$$a^l_{gh} = \theta_l + \sum_{u=0}^{k-1}\sum_{v=0}^{k-1} \theta^l_{uv} * input_{g*s+u,h*s+v} \tag{16}$$

$$\frac{\partial a^l_{gh}}{\partial \theta^l_{ab}} = \sum_{g,h} input_{g*s+a,h*s+b} \tag{17}$$

$$\tag{18}$$

Chain rule expressions, where all the individuals components have been calculated above

$$\frac{\partial J}{\partial \beta_i} = \frac{\partial J}{\partial \mathbf{y}} * \sum_j (\frac{\partial y_j}{\partial x_i} * \frac{\partial x_i}{\partial \beta_i}) \tag{19}$$

$$\frac{\partial J}{\partial w^l_{efi}} = \frac{\partial J}{\partial \mathbf{y}} * \sum_j (\frac{\partial y_j}{\partial x_i} * \frac{\partial x_i}{\partial w^l_{efi}}) \tag{20}$$

$$\frac{\partial J}{\partial \theta^l_{ab}} = \frac{\partial J}{\partial \mathbf{y}} * \sum_j (\frac{\partial y_j}{\partial x_i} * \frac{\partial x_i}{\partial v^l_{ef}} * \sum_{e,f} (\frac{\partial v^l_{ef}}{\partial a^l_{gh}} * \sum_{g,h} \frac{\partial a^l_{gh}}{\partial \theta^l_{ab}})) \tag{21}$$

Problem 2 (10 pts)

$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k|\text{pa}_k) \tag{22}$$

We need to prove

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = 1 \tag{23}$$

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = \sum_{x_1,x_2,..x_K} \prod_{k=1}^{K} p(x_k|\text{pa}_k) \tag{24}$$

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = \sum_{x_1,x_2,..x_K} p(x_K|\text{pa}_K) * \prod_{k=1}^{K-1} p(x_k|\text{pa}_k) \tag{25}$$

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = \sum_{x_1,x_2,..x_{K-1}} (\sum_{x_K} p(x_K|\text{pa}_K)) * \prod_{k=1}^{K-1} p(x_k|\text{pa}_k) \tag{26}$$

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = \sum_{x_1,x_2,..x_{K-1}} (1) * \prod_{k=1}^{K-1} p(x_k|\text{pa}_k) \tag{27}$$

Repeating the same steps as above K-1 times.

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = (1) * (1) * .. * (1) \tag{28}$$

$$\sum_{x_1,x_2,..x_K} p(\mathbf{x}) = 1 \tag{29}$$

This proves that the original representation was correctly normalized.

Problem 3 (10pts)

$$E(\mathbf{v}, \mathbf{h}; \theta) = -\sum_{i=1}^{D}\sum_{j=1}^{P} W_{ij} v_i h_j - \sum_{i=1}^{D} v_i b_i - \sum_{j=1}^{P} h_j a_j. \tag{30}$$

$$P_\theta(\mathbf{v}, \mathbf{h}) = \frac{1}{\mathcal{Z}} \exp(-E(\mathbf{v}, \mathbf{h}; \theta)) \tag{31}$$

$$\mathcal{Z} = \sum_{\mathbf{v},\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}; \theta)) \tag{32}$$

Since $h_2, h_3, ..., h_P$ are not part of markov blanket of h_1, we can write

$$P_\theta(h_1, h_2..., h_P|\mathbf{v}) = P_\theta(h_1|\mathbf{v}) * P_\theta(h_2..., h_P|\mathbf{v}) \tag{33}$$

Repeating the same step P times

$$P_\theta(h_1, h_2..., h_P|\mathbf{v}) = P_\theta(h_1|\mathbf{v}) * P_\theta(h_2|\mathbf{v}) * ... * P_\theta(h_P|\mathbf{v}) \tag{34}$$

$$P_\theta(h_1, h_2..., h_P|\mathbf{v}) = \prod_j P_\theta(h_j|\mathbf{v}) \tag{35}$$

$$P_\theta(h_j|\mathbf{v}) = \frac{1}{\mathcal{Z}_j} \exp(-E(\mathbf{v}, h_j; \theta)) \tag{36}$$

$$P_\theta(h_j|\mathbf{v}) = \frac{1}{\mathcal{Z}_j} \exp(\sum_{i=1}^{D} W_{ij} v_i h_j + \sum_{i=1}^{D} v_i b_i + h_j a_j) \tag{37}$$

$$P_\theta(h_j = 1|\mathbf{v}) = \frac{1}{\mathcal{Z}_j} \exp(\sum_{i=1}^{D} W_{ij} v_i + \sum_{i=1}^{D} v_i b_i + a_j) \tag{38}$$

$$P_\theta(h_j = 0|\mathbf{v}) = \frac{1}{\mathcal{Z}_j} \exp(\sum_{i=1}^{D} v_i b_i) \tag{39}$$

$$P_\theta(h_j = 1|\mathbf{v}) = P_\theta(h_j = 1|\mathbf{v})/(P_\theta(h_j = 1|\mathbf{v}) + P_\theta(h_j = 0|\mathbf{v})) \tag{40}$$

$$P_\theta(h_j = 1|\mathbf{v}) = \exp(\sum_{i=1}^{D} W_{ij} v_i + \sum_{i=1}^{D} v_i b_i + a_j)/(\exp(\sum_{i=1}^{D} W_{ij} v_i + \sum_{i=1}^{D} v_i b_i + a_j) + \exp(\sum_{i=1}^{D} v_i b_i)) \tag{41}$$

$$P_\theta(h_j = 1|\mathbf{v}) = \exp(\sum_{i=1}^{D} W_{ij} v_i + a_j)/(\exp(\sum_{i=1}^{D} W_{ij} v_i + a_j) + 1) \tag{42}$$

$$P_\theta(h_j = 1|\mathbf{v}) = 1/(1 + \exp(-\sum_{i=1}^{D} W_{ij} v_i - a_j)) \tag{43}$$

$$P_\theta(h_j = 1|\mathbf{v}) = Sigm(\sum_{i=1}^{D} W_{ij} v_i + a_j)) \tag{44}$$

$$\tag{45}$$

This proves the original expression

Problem 4 (10 pts)

$$E(\mathbf{x}, \mathbf{y}) = h \sum_i x_i - \beta \sum_{i,j} x_i x_j - \eta \sum_i x_i y_i, \tag{46}$$

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{\mathcal{Z}} \exp\left(-E(\mathbf{x}, \mathbf{y})\right) \tag{47}$$

- (5 pts)

$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) = h \sum_{i \neq k} x_i - \beta \sum_{i \neq k, j} x_i x_j - \eta \sum_{i \neq k} x_i y_i + h - \beta \sum_{j \in ne(x_k)} x_j - \eta y_k \tag{48}$$

$$E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) = h \sum_{i \neq k} x_i - \beta \sum_{i \neq k, j} x_i x_j - \eta \sum_{i \neq k} x_i y_i - h + \beta \sum_{j \in ne(x_k)} x_j + \eta y_k \tag{49}$$

$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) = 2h - 2\beta \sum_{j \in ne(x_k)} x_j - 2\eta y_k \tag{50}$$

As per this expression the difference in energy depends only on neighbourhood of particular element of x.

- (5 pts)

Substituting β and h in expression above

$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) = -2\eta y_k \tag{51}$$

Given $\eta >= 0$
If $y_k = 1$

$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) = -2\eta \tag{52}$$
$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) < 0 \tag{53}$$
$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) < E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) \tag{54}$$
$$\tag{55}$$

Lower value of energy corresponds to higher likelihood. Therefore $x_k = 1$

Similarly, if $y_k = -1$

$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) = 2\eta \tag{56}$$
$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) - E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) > 0 \tag{57}$$
$$E(x_1, ...x_k = 1, ..x_K, \mathbf{y}) > E(x_1, ...x_k = -1, ..x_K, \mathbf{y}) \tag{58}$$
$$\tag{59}$$

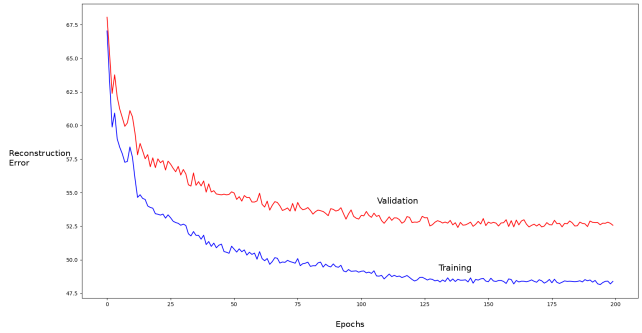
Hihger value of energy corresponds to lower likelihood. Therefore $x_k = -1$

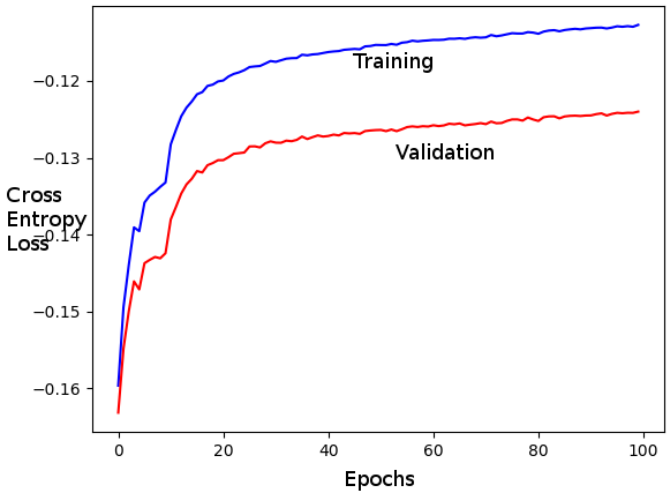
From the two results above the most probable configuration of latent variables corresponds to $y_k = x_k$

Problem 5 (60 pts)

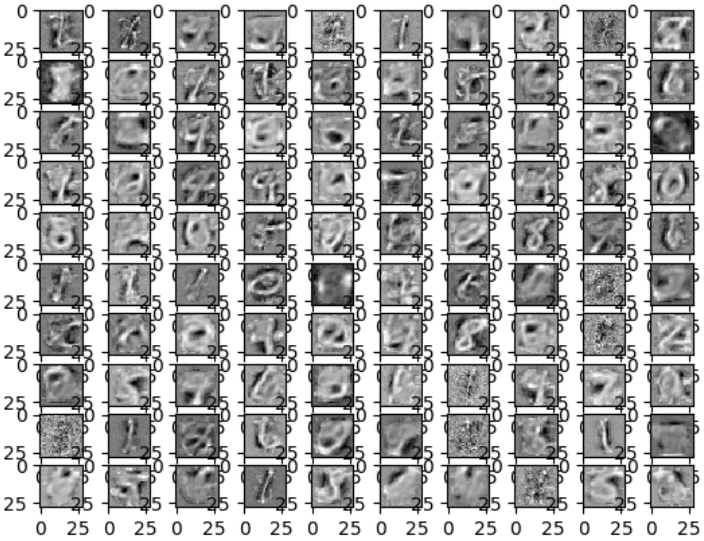
Contrastive Divergence (CD), Autoencoders

a) Basic generalization [20 points]



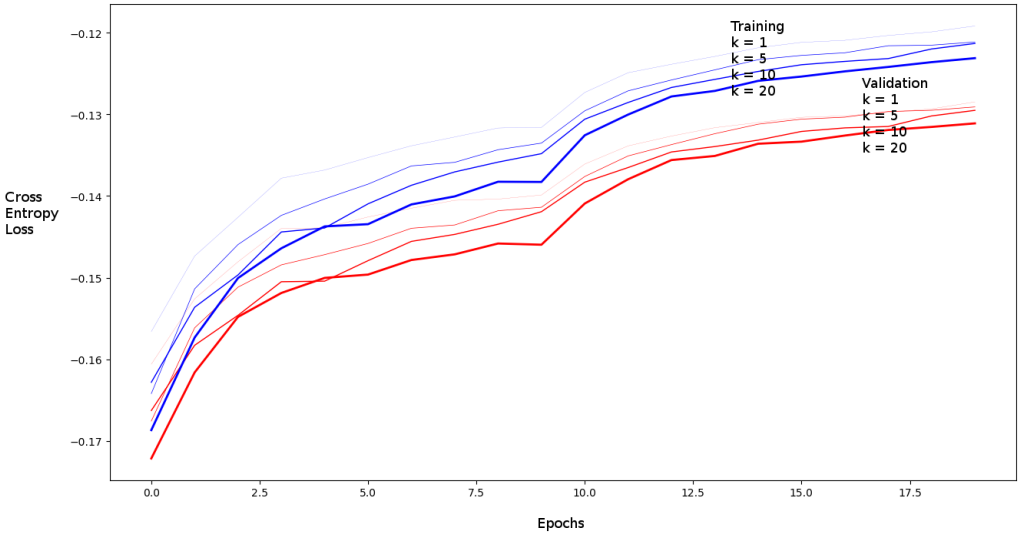


Training set error goes down faster compared to validation set as expected. After 125 epochs the validation error stops reducing representing the generalization error region.



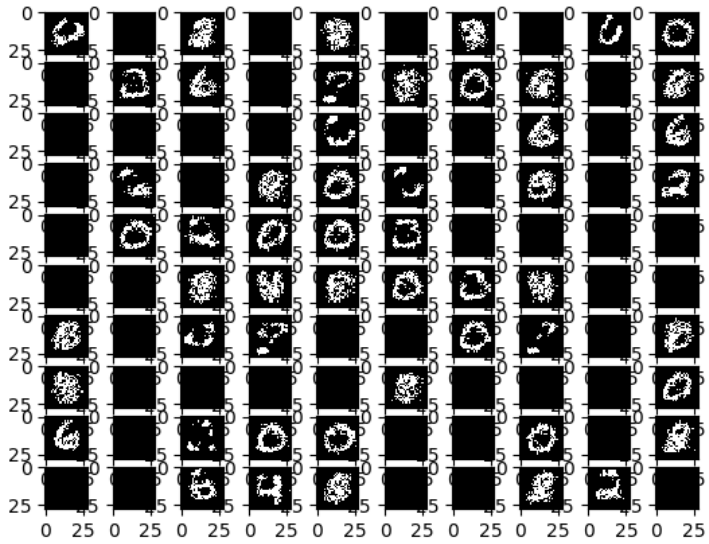
The visualized Weights W do represent structure similar to original digits or components such as curves that look similar to components of the original digits.

b) Number of CD steps [5 points]



In terms of convergence of training accuracy, higher value of k (indicated by thicker lines in the figure) converges slowly compared to smaller values (indicated by thicker lines in the figure) i.e. k=1 converges the fastest.
However in terms of generalization error, k = 5 does the best. k = 10 is comparable to k = 5. While k = 1 has the highest generalization error. k = 20 underfits, at least untill 50 epochs compared to smaller values of k. Discribe Observation

c) Sampling from the RBM model [5 points]

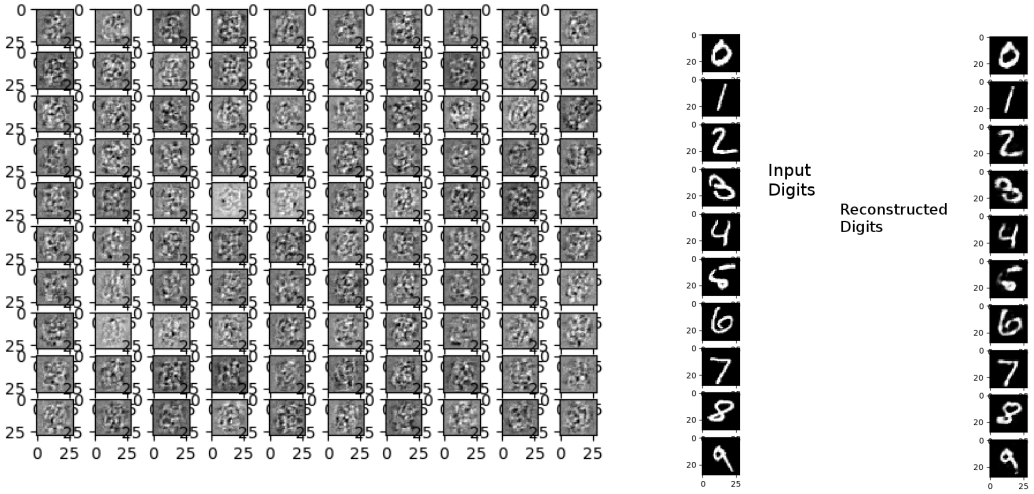


They do show a very rough structure of handwritten digits.

d) Unsupervised Learning as pretraining [5 points]

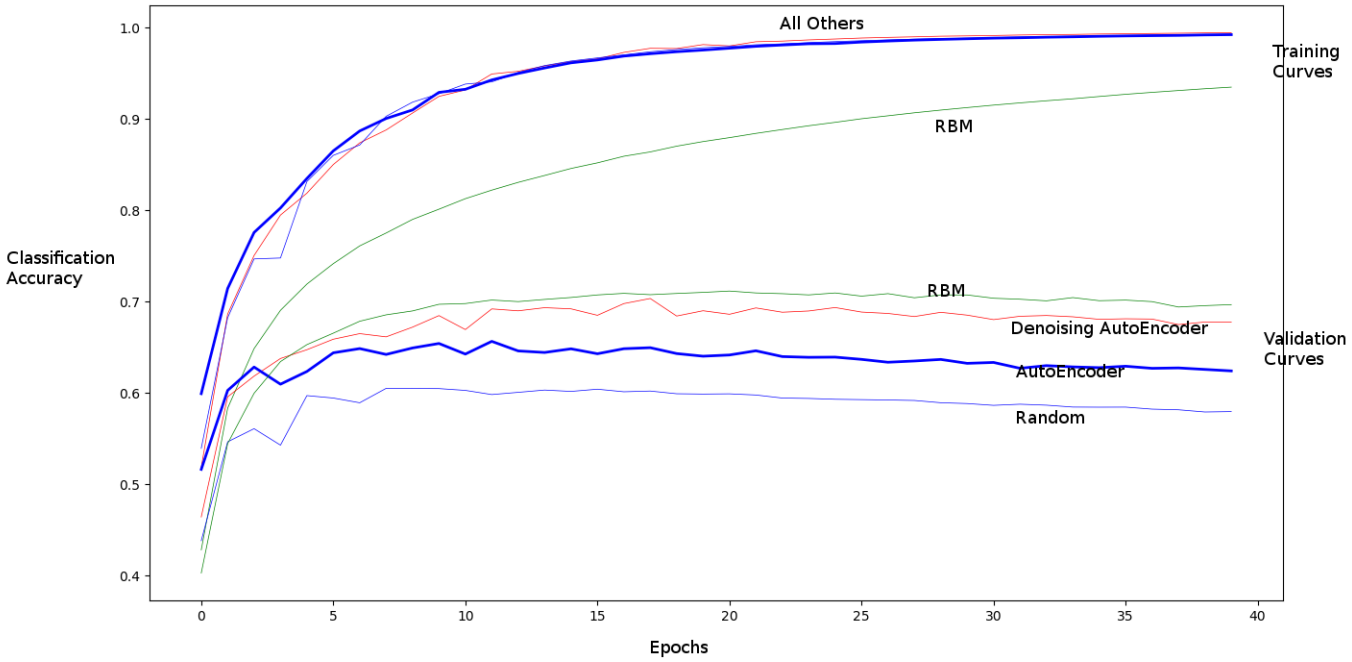
Yes, rbm initialization imporves the model generalizability. It acheives the best validation accuracy, and tends to not over fit unlike all other initialization methods. (see Figure A)

e) Autoencoder [5 points]

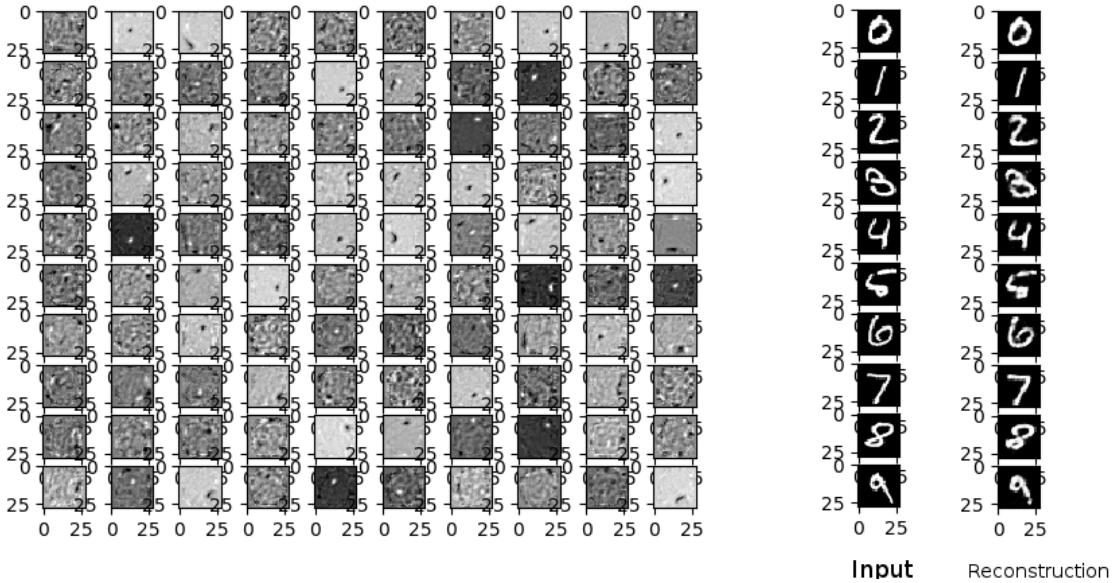


Unlike rbm, there is very limited structure to discern in the visualization of Auto encoder weights. Even so, the auto encoder is able to accomplish a fairly good reconstruction of the input digit.
The standard autoencoder does better than random initialization in terms of validation accuracy. However it still starts to overfit after 10 epochs or so. It doesn't offer the same generalizability as the rbm initialization. (see Figure A)

Figure 1: Figure A: Comparing Generalization for different initialization strategies

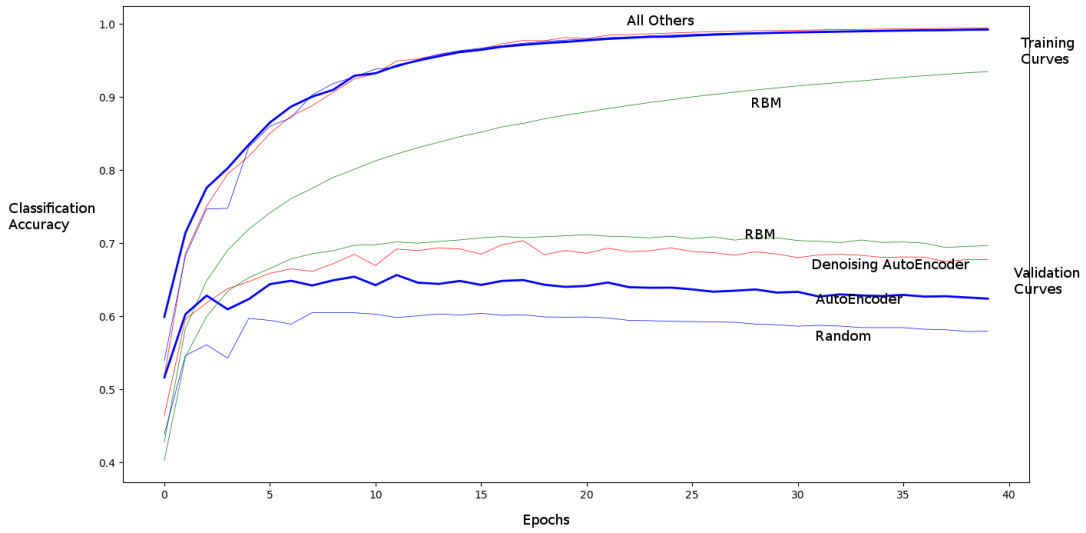


f) Denoising Autoencoder [10 points]



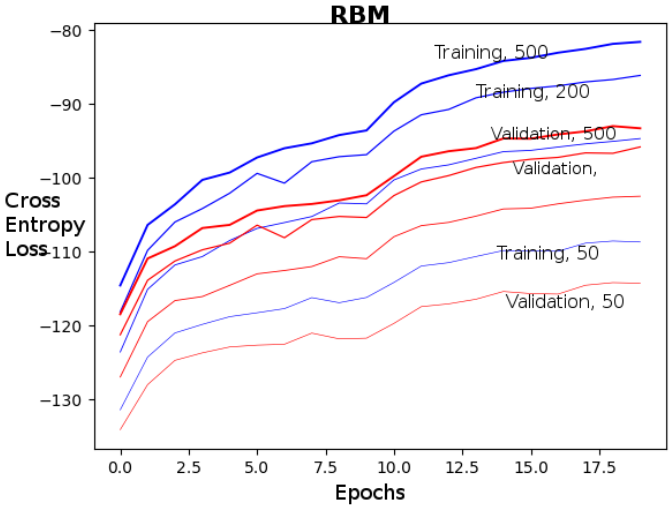
Compared to no drop out auto encoder, there is greater structure visible in the weight visualization. The reconstructed digits are also sharper compared to the standard autoencoder. The filters are still not as crisp as the rbm case.
The denoising autoencoder pre-training is an improvement over both random initialization and standard auto encoder.

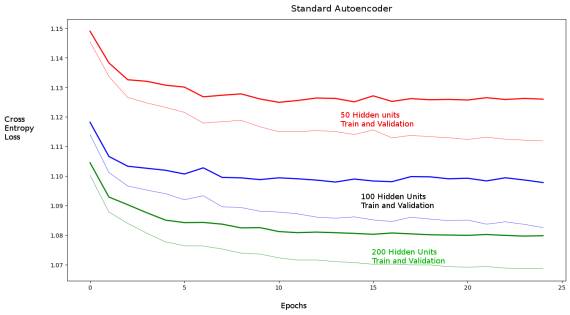
Figure 2: Figure A (repeated): Comparing Generalization for different initialization strategies

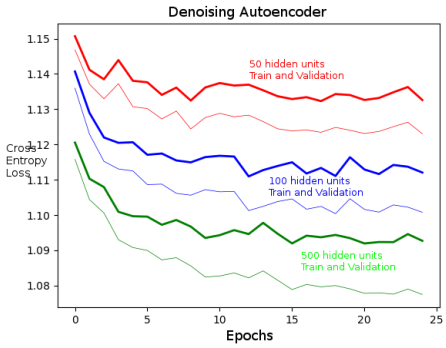


Compared to rbm initializations it performs as good or slightly worse on multiple attempts in terms of validation accuracy and generalizability. (see Figure A)

g) Number of hidden units [10 points]







(Note: Some of the loss plots are negative log likelihood.)
The larger the hidden units, the better training and validation accuracy across all 3 settings. It follows from intuition that larger hidden units allow preserving more information about the input making the reconstruction easier. Obviously, with larger number of hidden units the gradient descent takes longer to converge given the larger number of parameters to be learnt.