CIE6032 and MDS6232: Homework #2

Due on Wednesday, December 5th, 2018, 5:30pm (DYB 224)

Problem 1

[**20 points**]

A recurrent neural network is shown in Figure 1,

$$\mathbf{h}_t = \tanh(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h)$$

$$\mathbf{z}_t = \mathrm{softmax}(\mathbf{W}_{hz}\mathbf{h}_t + \mathbf{b}_z)$$

The total loss for a given input/target sequence pair $(\mathbf{x}, \mathbf{y})$, measured in cross entropy

$$L(\mathbf{x}, \mathbf{y}) = \sum_t L_t = \sum_t -\log z_{t,y_t}$$

In the lecture, we provide the general idea of how to calculate the gradients $\frac{\partial L}{\partial \mathbf{W}_{hz}}$ and $\frac{\partial L}{\partial \mathbf{W}_{hh}}$. Please provide the details of the algorithms and equations, considering the mapping and cost functions provided above.

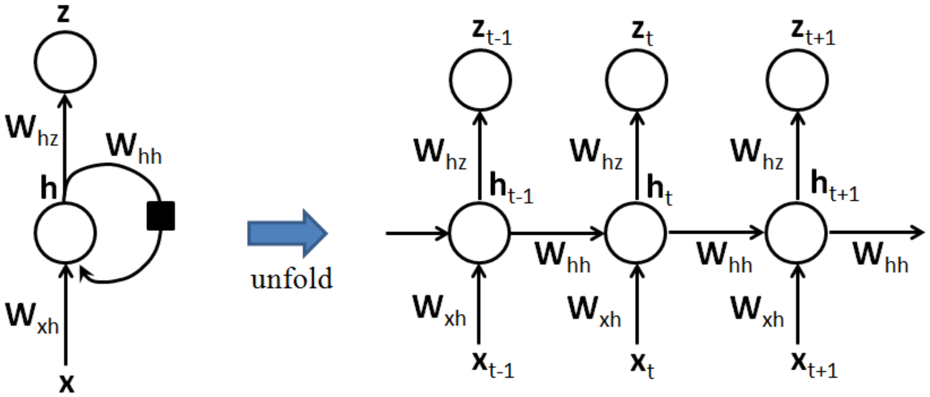


Figure 1:

Problem 2

[**20 points**]

A RNN model is shown in Figure 2. $\mathbf{x}_1, \ldots, \mathbf{x}_t$ are input variables. A hidden variable $\mathbf{h}_t = F_\theta(\mathbf{h}_{t-1}, \mathbf{x}_t)$ contains information about the whole past sequence. It defines a function which maps the whole past sequence $(\mathbf{x}_t, \ldots, \mathbf{x}_1)$ to the current state $\mathbf{h}_t = G_t(\mathbf{x}_t, \ldots, \mathbf{x}_1)$.

- Give the explicit expression of G_t in terms of F_θ, $\mathbf{x}_1, \ldots, \mathbf{x}_t$. [**10 points**]

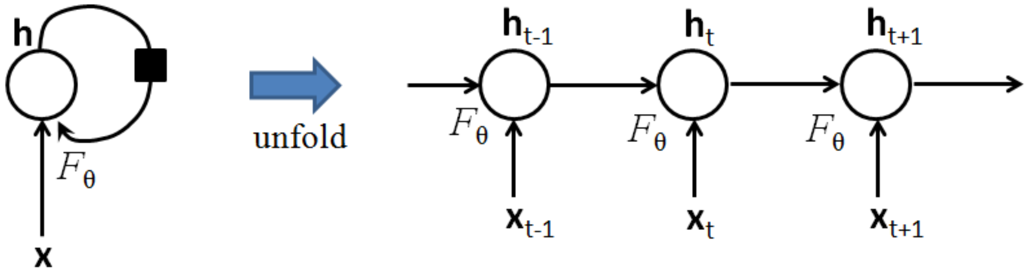


Figure 2:

- There are two major difficulties when directly modeling $\mathbf{h}_t = G_t(\mathbf{x}_t, \ldots, \mathbf{x}_1)$: (1) G_t is different for different t (i.e. sequence length). It limits the generalization power of the model and there may not be enough training samples for each sequence length. (2) The model complexity of G_t may increase exponentially with t. Provide two reasons to explain how RNN solves the two challenges. [**10 points**]

Problem 3

[**20 points**]

Consider a neural network with one input layer (which has d nodes), L hidden layers (each of which has n hidden nodes), and one output layer (which has c nodes). The neurons between two layers are fully connected.

- What is the space complexity of the neural network, i.e. the number of parameters to store? [**5 points**]
- Stochastic gradient descent is used for Backpropagation (BP), i.e. only one training sample is used to compute the gradients to update the weights in each training iteration. What is the time complexity for each iteration? Only the number of multiplication operations is considered in this problem. [**5 points**]
- Instead of using BP, the gradients of weights can also be computed in the following way:

$$\frac{\partial u_N}{\partial w_{ji}} = \frac{\partial u_N}{\partial u_i} \frac{\partial u_i}{\partial net_i} \frac{\partial net_i}{\partial w_{ji}},$$

  where u_N is last node to compute the cost function J, w_{ji} is a weight on the connection between node i and node j, net_i is the net activation of node i, and u_i is the output of node i. $\frac{\partial J}{\partial u_i}$ is computed by enumerating all the possible paths connecting nodes i and J, and applying the chain rule to these paths separately.

$$\frac{\partial u_N}{\partial u_i} = \sum_{\text{paths } u_{k_1} \ldots u_{k_n} : k_1 = i, k_n = N} \prod_{j=2}^{n} \frac{\partial u_{k_j}}{\partial u_{k_{j-1}}}.$$

  What is the time complexity of this approach of updating the weight at each training iteration? [**10 points**]

Problem 4

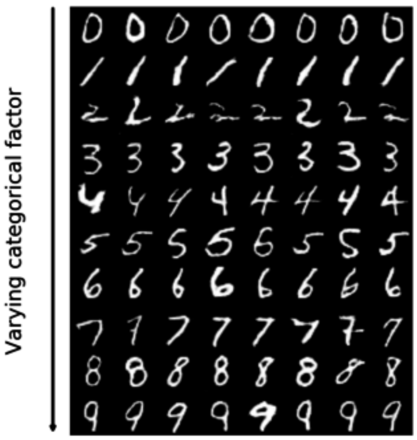
[**20 points**]

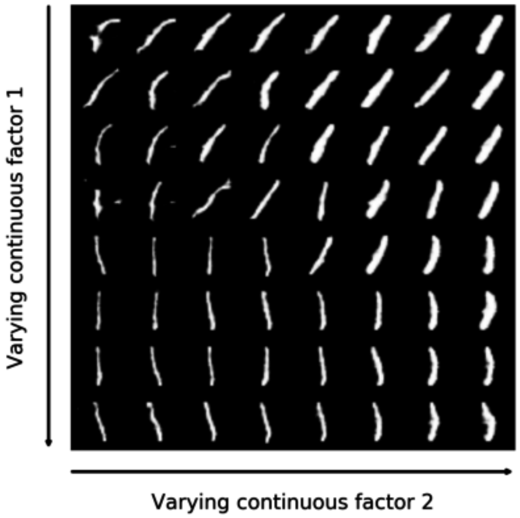
Please prove that in autoencoder, if there is one linear hidden layer and the mean squared error criterion is used to train the network, the k hidden unites learn to project the input in the span of the first k principal components of data obtained by PCA.

Problem 5

[**20 points**]

Train an InfoGan network based on the tutorial (better using MXnet) we introduced in the lecture using MNIST dataset. The expection is that the generated output should be varied through the categirical factor and continuous factor 1 and 2 as follows.


Varying categorical factor


Varying continuous factor 1
Varying continuous factor 2