School of Computing and Information Systems (CIS)
The University of Melbourne
COMP90073
Security Analytics
Tutorial exercises: Week 8

1. State some relations between autoencoders and PCA.

   Solution: They are both feature representation learning methods. PCA is only linear transformation to the subspace while autoencoder is nonlinear transformation to the hidden units. If the autoencoder's activation functions are linear, it is very similar to PCA method.

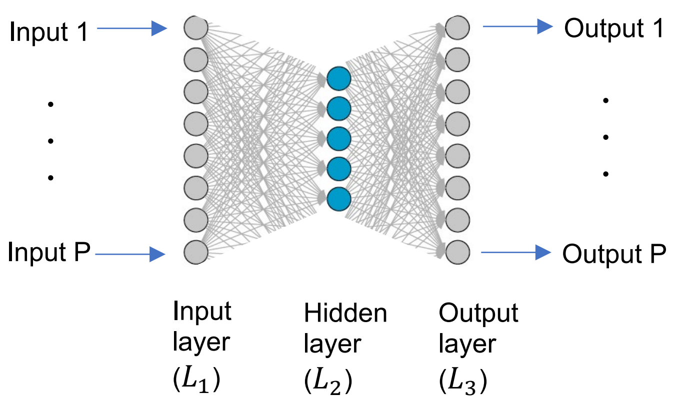
2. What is the complexity of the back-propagation algorithm for an autoencoder with *L* layers and *K* nodes per layer?

   Solution: $O(K^2L)$
   The dominant term is a multiplication of a vector with a $K \times K$ matrix and this has to be done in each of the *L* layers.

3. Assume that you initialize all weights in a neural net to the same value and you do the same for the bias terms. Is this a good idea? Justify your answer.

   Solution: This is a bad idea, since in this case every node on a particular level will learn the same feature.

4. An autoencoder is a neural network designed to learn feature representations in an unsupervised manner. Unlike a standard multi-layer network, an autoencoder has the same number of nodes in its output layer as its input layer. An autoencoder is trained to reconstruct its own input x, i.e. to minimize the reconstruction error. An autoencoder is shown below.



   Suppose the input is a set of P-dimensional unlabelled data $\{x^{(i)}\}_{i=1}^{N}$. Consider an autoencoder with H hidden units in the second layer L_2. We will use the following notation for this autoencoder:

- W^e denotes the $P \times H$ weight matrix between L_1 and L_2
- W^d denotes the $H \times P$ weight matrix between L_2 and L_3
- σ denotes the activation function for L_2 and L_3
- $s_j^{(i)} = \sum_{k=1}^{P} W_{kj}^{e} x_k^{(i)}$
- $h_j^{(i)} = \sigma(\sum_{k=1}^{P} W_{kj}^{e} x_k^{(i)})$
- $t_j^{(i)} = \sum_{k=1}^{H} W_{kj}^{d} h_k^{(i)}$
- $\hat{x}_j^{(i)} = \sigma\left(\sum_{k=1}^{H} W_{kj}^{d} h_k^{(i)}\right)$
- $J(W^e, W^d)^{(i)} = \left\|x^{(i)} - \hat{x}^{(i)}\right\|_2^2 = \sum_{j=1}^{P}\left(x_j^{(i)} - \hat{x}_j^{(i)}\right)^2$ is the reconstruction error for example $x^{(i)}$
- $J(W^e, W^d) = \sum_{j=1}^{N} J(W^e, W^d)^{(i)}$ is the total reconstruction error
- (We add element 1 to the input layer and hidden layer so that no bias term has to be considered)

Fill in the following derivative equations for W^e and W^d. Use the notation defined above; there should be no new notation needed.

$$\frac{\partial J^{(i)}}{\partial W_{kl}^{d}} = \sum_{j=1}^{P}\left(\boxed{} \cdot \frac{\partial \hat{x}_j^{(i)}}{\partial W_{kl}^{d}}\right)$$

$$\frac{\partial \hat{x}_j^{(i)}}{\partial W_{kl}^{d}} = \sigma'\left(\sum_{k=1}^{H} W_{kj}^{e} x_k^{(i)}\right) \cdot \boxed{}$$

$$\frac{\partial J^{(i)}}{\partial W_{kl}^{e}} = \frac{\partial J^{(i)}}{\partial s_j^{(i)}} \cdot \boxed{}$$

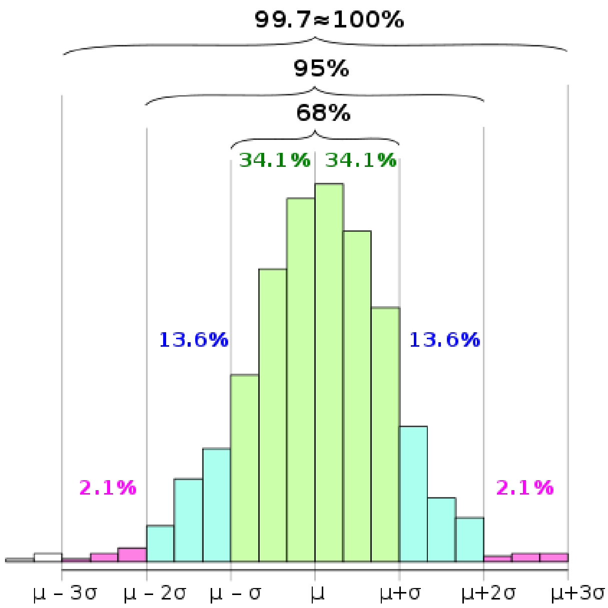
$$\frac{\partial J^{(i)}}{\partial s_j^{(i)}} = \sum_{k=1}^{H}\left(\frac{\partial J^{(i)}}{\partial t_k^{(i)}} \cdot \boxed{} \cdot \sigma'(s_j^{(i)})\right)$$

Solution:

- $2(\hat{x}_j^{(i)} - x_j^{(i)})$
- $h_k^{(i)}$
- $\frac{\partial s_j^{(i)}}{\partial W_{kl}^{e}} = x_k^{(i)}$
- W_{jk}^{d}

5. 3σ rule is a common technique used for anomaly detection. Describe what is the intuition of this rule for anomaly detection? How our result will be effected if we use other values of σ (e.g., 2σ, or 4σ)?

   Solution:
   A clear description can be find in
   https://en.wikipedia.org/wiki/68%E2%80%9395%E2%80%9399.7_rule



6. In the VAE, how sampling of the latent code is different during training and generation (generating a new sample)?

   Solution: During training, we are drawing samples from the posterior distribution, because we are trying to reconstruct a specific datapoint. While, during generation, we want to generate samples from the prior distribution of latent codes.

   During training, we are drawing $h \sim P(h|x)$, and then decoding with $\hat{x} = g(h)$
   During generation, we are drawing $h \sim P(h)$, and then decoding $x = g(h)$.