01/11/2019 **AML Quiz on AE, CNN, and GAN** **Marks:30**

Write True/False clearly and provide precise justification wherever applicable. Any instance of cheating is considered academic dishonesty.

Questions 1 to 6 are of 2 marks each.

1. “Convolutional Neural Networks can perform all basic geometric transformations (translation, rotation, and scaling)”. Is this statement True or False? Justify.

Ans: False, Data Preprocessing steps (viz translation rotation, scaling) is necessary before you give the data to neural network because the neural network cannot do it itself.

2. Increasing the size of the kernel would increase the performance of a CNN (True/False). Justify your answer

Ans: False, Increasing kernel size would not necessarily increase performance. This depends heavily on the dataset.

3. Which of the following statements is true when you use 1×1 convolutions in a CNN?
   a. It can help in dimensionality reduction
   b. It can be used for feature pooling
   c. It suffers less overfitting due to small kernel size
   d. All of the above

Ans (d) 1×1 convolutions are called bottleneck structure in CNN.

4. Suppose there is an issue while training a neural network. The training loss/validation loss remains constant. What could be the possible reason?
   a. Architecture is not defined correctly
   b. Data given to the model is noisy
   c. Initialize all the weights with constant
   d. All of the above
   e. Only a and b

Ans: (d), Justification for initializing all the weights with constant: If the neurons start with the same weights, then all the neurons will follow the same gradient, and will always end up doing the same thing as one another.

https://stats.stackexchange.com/questions/27112/danger-of-setting-all-initial-weights-to-zero-in-backpropagation

5. Which of the following is incorrect about AutoEncoders?
   a. There is a constraint on the symmetry of an autoencoder.
   b. The neural network’s target output is its input.
   c. It minimizes the same objective function as PCA
   d. The weights learn in the encoder part are not flipped version of the decoder.

Ans (a)

6. Which of the following costs is the non-saturating generator cost for GANs (G is the generator and D is the discriminator)?

$$\text{(i)}\quad J^{(G)} = \frac{1}{m}\sum_{i=1}^{m} \log\left(1 - D(G(z^{(i)}))\right)$$

$$\text{(ii)}\quad J^{(G)} = -\frac{1}{m}\sum_{i=1}^{m} \log\left(D(G(z^{(i)}))\right)$$

$$\text{(iii)}\quad J^{(G)} = \frac{1}{m}\sum_{i=1}^{m} \log\left(1 - G(D(z^{(i)}))\right)$$

$$\text{(iv)}\quad J^{(G)} = -\frac{1}{m}\sum_{i=1}^{m} \log\left(G(D(z^{(i)}))\right)$$

Ans (ii)

7. **(4 marks)** The input image has been converted into a matrix of size 28 X 28 and a kernel/filter of size 7 X 7 with a stride of 2. What will be the size of the convoluted matrix?

Ans: The size of the convoluted matrix is given by C=((I-F+2P)/S)+1, where C is the size of the Convoluted matrix, I is the size of the input matrix, F the size of the filter matrix and P the padding applied to the input matrix.

8. **(4 marks)** Explain DropOut and DropConnect. Why DropConnect is a generalization of DropOut?

Ans: Dropout: To apply DropOut, we randomly select a subset of the units and clamp their output to zero, regardless of the input; this effectively removes those units from the model. A different subset of units is randomly selected every time we present a training example.

Dropconnect: DropConnect works similarly, except that we disable individual weights (i.e., set them to zero), instead of nodes, so a node can remain partially active.

DropConnect is a generalization of DropOut because it produces even more possible models, since there are almost always more connections than units. However, you can get similar outcomes on an individual trial.

9. **(4 marks)** Compare GAN and VAE. Why images generated by GAN are sharper than images generated by VAE?

Ans: Variational autoencoders are generative algorithm that add an additional constraint to encoding the input data, namely that the hidden representations are normalized. Variational autoencoders are capable of both compressing data like an autoencoder and synthesizing data like a GAN. However, while GANs generate data in fine, granular detail, images generated by VAEs tend to be more blurred.

Why GAN better: GANs training paradigm (see VAEGAN paper) is adversarial, so that we aren't using maximum likelihood learning (maximizing some sort of log likelihood) but learning through an critic, a discriminator which nudges the generator to produce more realistic samples. That the GAN produces 'crisper' images is a result of not using an artificial, user defined loss function (usually L1 or L2) but a loss that is learnt from the data with the discriminator.

10. **(6 marks)** Generative models create a model θ that maximizes the Maximum Likelihood Estimation(MLE). i.e. finding the best model parameters that fit the training data the most. Suppose you are using KL-divergence for MLE. This is same as minimizing the KL-divergence KL(p,q) which measures how the probability distribution q (estimated distribution) diverges from the expected probability distribution p (the real-life distribution).

$$D_{KL}(p||q) = \int_x p(x) \log \frac{p(x)}{q(x)} dx$$

KL-divergence is not symmetrical i.e.

$$D_{KL}(p||q) \neq D_{KL}(q||p)$$

What are the implications of using KL-divergence and how can we compensate for this?

Ans: The KL-divergence DL(p, q) penalizes the generator if it misses some modes of images: the penalty is high where p(x) > 0 but q(x) → 0. Nevertheless, it is acceptable that some images do not look real. The penalty is low when p(x) → 0 but q(x)>0. (Poorer quality but more diverse samples)

On the other hand, the reverse KL-divergence DL(q, p) penalizes the generator if the images do not look real: high penalty if p(x)→ 0 but q(x) > 0. But it explores less variety: low penalty if q(x) → 0 but p(x) > 0. (Better quality but less diverse samples)

https://medium.com/@jonathan_hui/gan-why-it-is-so-hard-to-train-generative-advisory-networks-819a86b3750b