1. What are the main tasks that autoencoders are used for?

ANS:

An autoencoder is an unsupervised learning technique for neural networks that learns efficient data representations (encoding) by training the network to ignore signal “noise.” Autoencoders can be used for image denoising, image compression, and, in some cases, even generation of image data.

1. Suppose you want to train a classifier, and you have plenty of unlabeled training data but only a few thousand labeled instances. How can autoencoders help? How would you proceed?

ANS:

Autoencoders are trained the same way as ANNs via backpropagation.

An autoencoder neural network is an unsupervised learning algorithm that applies backpropagation, setting the target values to be equal to the inputs. I.e., it uses y(i)=x(i) .

When we are using AutoEncoders for dimensionality reduction we'll be extracting the bottleneck layer and use it to reduce the dimensions. This process can be viewed as feature extraction. The type of AutoEncoder that we're using is Deep AutoEncoder, where the encoder and the decoder are symmetrical.

1. If an autoencoder perfectly reconstructs the inputs, is it necessarily a good autoencoder? How can you evaluate the performance of an autoencoder?

ANS:

Another way I evaluate the performance of an autoencoder is by simply visually comparing the input and output images taken from the test set. This is by no means very scientific, but it gives a good idea whether an autoencoder is able to reconstruct the input images.

Autoencoders work in two different parts. Encoder (first part) is used to learn the important and representative features of the given image and represent them into latent space. Decoder (second part) is used to reconstruct the image from the latent space by removing the noise and unimportant features from the image.

When the auto-encoders are trained with the encoded size which is large enough, the accuracy for both cases above is about 92%. Compared to the CNN which takes original data as input and achieves the accuracy of 99%, the accuracy loss due to auto-encoders is not too much.

1. What are undercomplete and overcomplete autoencoders? What is the main risk of an excessively undercomplete autoencoder? What about the main risk of an overcomplete autoencoder?

ANS:

The main risk of an excessively undercomplete autoencoder is that it may fail to reconstruct the inputs. The main risk of an overcomplete autoencoder is that it may just copy the inputs to the outputs, without learning any useful feature.

1. How do you tie weights in a stacked autoencoder? What is the point of doing so?

ANS:

Implementation of Tying Weights: To implement tying weights, we need to create a custom layer to tie weights between the layer using keras. This custom layer acts as a regular dense layer, but it uses the transposed weights of the encoder's dense layer, however having its own bias vector

The encoder is used to generate a reduced feature representation from an initial input x by a hidden layer h. The decoder is used to reconstruct the initial input from the encoder's output by minimizing the loss function. The autoencoder converts high-dimensional data to low-dimensional data.

1. What is a generative model? Can you name a type of generative autoencoder?

ANS:

A generative model includes the distribution of the data itself, and tells you how likely a given example is. For example, models that predict the next word in a sequence are typically generative models (usually much simpler than GANs) because they can assign a probability to a sequence of words.

Autoencoders are also generative models which can randomly generate new data that is similar to the input data (training data).

1. What is a GAN? Can you name a few tasks where GANs can shine?

ANS:

Generate Photographs of Human Faces. Generate Realistic Photographs. Generate Cartoon Characters. Image-to-Image Translation.

Generative adversarial networks (GANs) are an exciting recent innovation in machine learning. GANs are generative models: they create new data instances that resemble your training data. For example, GANs can create images that look like photographs of human faces, even though the faces don't belong to any real person.

1. What are the main difficulties when training GANs?

ANS:

Nevertheless, GANs are difficult to train, and training faces two major problems, namely mode collapse, and non-convergence. One feasible method to make GAN solve these two challenges is to redesign the network architecture to get a more powerful model.