Deep Learning and Computer Vision

Unsupervised Learning by Generative Adversarial Network

1. **What is the difference between supervised learning & unsupervised learning in image classification task?**

Image Classification	
Supervised Learning	*Unsupervised Learning*
Learn a mapping function such that the images and their corresponding labels can be correlated together.	Learn a mapping function such that the images and their corresponding labels can be clustered together in one or more categories.
Achieved by training a model to recognize the labels.	Achieved by training a model to learn the underlying features of an image.
Implemented with the cross-entropy loss function.	Implemented with the least-squares loss function.
Architectural examples include Support Vector Machines (SVMs) and Naïve Bayes classifiers.	Architectural examples include k-means clustering and Gaussian models.

2. **What is the difference between an auto-encoder and a generative adversarial network considering (1) model structure; (2) optimized objective function; (3) training procedure on different components.**

	Auto-encoder	**GAN**
Model Structure	Consists of a fully connected feedforward neural network created by combining two modules: - *Encoder:* Takes an input image and transforms it to a latent vector that maps to a decoder. - *Decoder*: Maps the latent vector to an output image with the same dimensions as the original image.	Consists of two conflicting modules that produce fake images and then compare with the seen regularities from the test data: - *Generator*: A neural network designed to produce new/unseen images. - *Discriminator*: A neural network designed to classify the generated images as real or fake.
Optimized objective function	*Mean Squared Error* (MSE): $J(x, z) = \|\|x - z\|\|^2$ where, the function "… measures how close the reconstructed input z is to the original input x"[1].	*Minimax Loss*[2]: $E_x[\log D(x)] + E_z[\log 1 - D(G(x))]$ Derived using the cross-entropy between the input (real) and generated (real or fake) distributions: a. E_x: Expected value for all real instances. b. E_z: Expected values for all the fake instances c. $D(x)$: Probability distribution highlighting the instances of real data x. d. $D(G(x))$: Probability distribution to highlight if a fake instance *is* real.

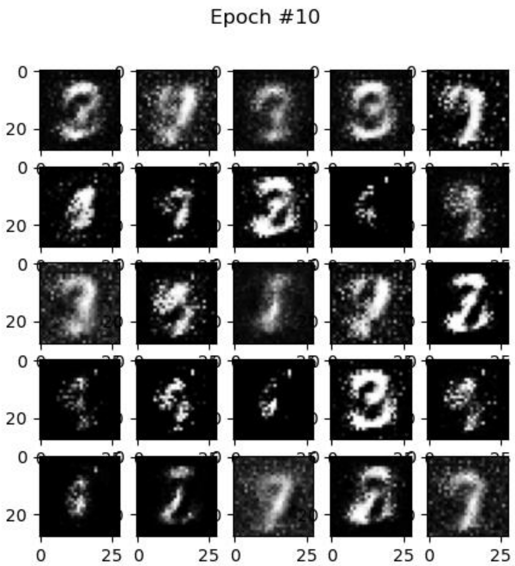
[1] Yale Data Science
[2] Google Developers Machine Learning

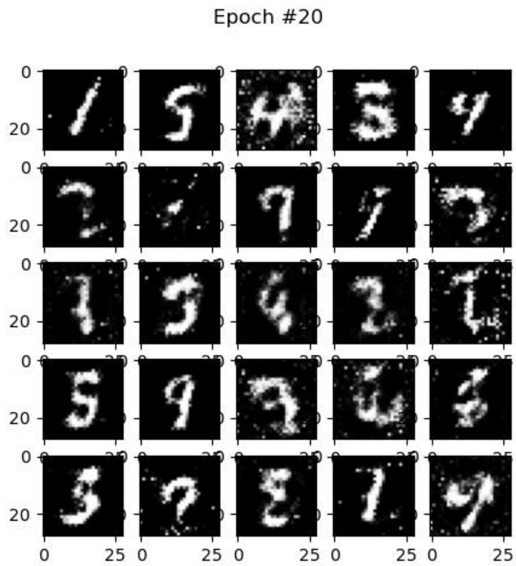
Training procedure on different components	1. Stochastic initialization of the weights. 2. Periodic updating of the weights performed via an iterative process. 3. Backpropagation used to minimize the total loss.	1. Generator produces a dataset comprised of artificial images by using a sample within the Gaussian distribution of the input (ground-truth) images. 2. Discriminator is trained to classify between ground-truth and artificial images with labelling. 3. Halt the weights of the discriminator model before retraining the generator model. 4. Train the generator model with the discriminator associated weights to produce images that can be classified by the model.

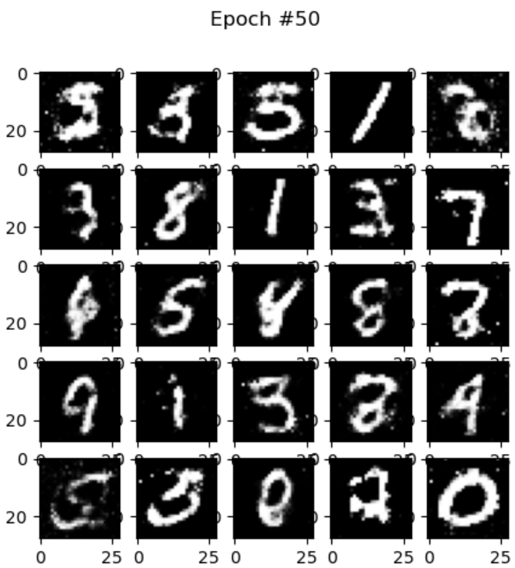
3. **How is the distribution $p_g(x)$ learned by the generator compared to the real data distribution $p_{data}(x)$ when the discriminator cannot tell the difference between these two distributions?**

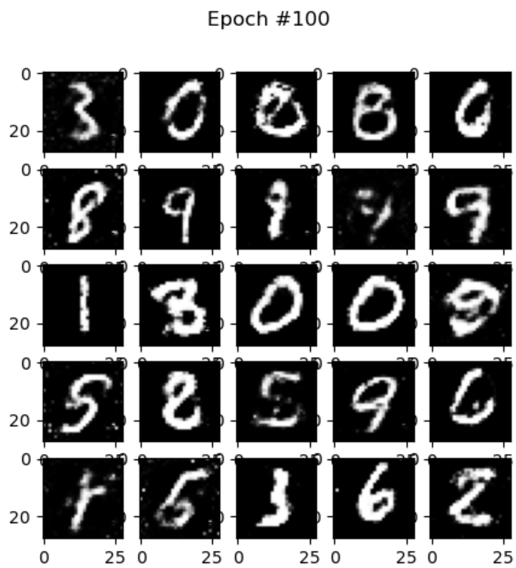
If the two distributions are coincident, such as the global optimality of $p_g(x) = p_{data}(x)$, they are perfectly superimposed. This translates into successful termination of the learning process when the GAN model is fully trained.

4. **Show the generated images at the 10th epoch, the 20th epoch, the 50th epoch, the 100th epoch by using the architecture required in Guideline.**









5. **Plot the loss curve during training.**