Review Questions 5

Group 1
Chuan Su
Diego Alonso Guillen Rosaperez

December 14, 2019

1. A generative model aims at learning as similar as possible to the true data distribution of the training set, since it is not always possible to learn the exact distribution of our data either implicitly or explicitly. Also, it try to generate new data points with some variations. An example is the Variational Autoencoders which aims at maximizing the lower bound of the data log-likelihood.

2. A limitation of GANs is that it still needs a wealth of training data to get started creating images. For instance, without enough pictures of human faces, it won't be able to come up with new faces. This means that areas where data is non-present won't be able to use GAN. Thus, it cannot invent totally new things, only expect them to combine what they already know in new ways.
   Also, there is a lack of control in the generations. For instance, if the discriminator is too weak, it will accept anything the generator produces, even if it's a dog with two heads or four eyes. On the other hand, if the discriminator is much stronger than the generator, it will constantly reject the results, resulting in an endless loop of disappointing data. Thus, Engineers must constantly optimize the generator and discriminator networks sequentially to avoid these effects

3. Minibatch standard deviation layers are added near the end of the discriminator. For each position in the inputs, it computes the standard deviation across all channels and all instances in the batch. These standard deviations are then averaged across all points to get a single value. Finally, an extra feature map is added to each instance in the batch and filled with the computed value. With minibatch standard deviation layers the discriminator will have easy access to this statistic, making it less likely to be fooled by a generator that produces too little diversity. This will encourage the generator to produce more diverse outputs, reducing the risk of mode collapse.

4. Nash equilibrium is a game theory that proposed solution of a non-cooperative game involving two or more players in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only their own strategy. GAN can only reach a single Nash equilibrium and In that case, the generator produces perfectly realistic images, and the discriminator is forced to guess (50% real, 50% fake).

5. (a) In supervised and unsupervised learning, the goal is generally to find patterns in the data. In Reinforcement Learning, the goal is to find a good policy.

   (b) Unlike in supervised learning, the agent is not explicitly given the "right" answer. It must learn by trial and error.

   (c) Unlike in unsupervised learning, there is a form of supervision, through rewards. We do not tell the agent how to perform the task, but we do tell it when it is making progress or when it is failing.

   (d) A Reinforcement Learning agent needs to find the right balance between exploring the environment, looking for new ways of getting rewards, and exploiting sources of rewards that it already knows. In contrast, supervised and unsupervised learning systems generally don't need to worry about explora- tion; they just feed on the training data they are given.

   (e) In supervised and unsupervised learning, training instances are typically independent (in fact, they are generally shuffled). In Reinforcement Learning, consecutive observations are generally not independent. An agent may remain in the same region of the environment for a while before it moves on, so consecu- tive observations will be very correlated. In some cases a replay memory is used to ensure that the training algorithm gets fairly independent observa- tions.

6. The credit assignment problem is the fact that when a Reinforcement Learning agent receives a reward, it has no direct way of knowing which of its previous actions contributed to this reward. It typically occurs when there is a large delay between an action and the resulting rewards (e.g., during a game of Atari's Pong, there may be a few dozen time steps between the moment the agent hits the ball and the moment it wins the point).

7. (a) With UCB1 for the next turn (15th):

$$Machine1 = \frac{4}{7} + \sqrt{\frac{ln(15)}{7}} \approx 1.19 \tag{1}$$

$$Machine2 = \frac{3}{5} + \sqrt{\frac{ln(15)}{5}} \approx 1.34 \tag{2}$$

$$Machine3 = \frac{0}{2} + \sqrt{\frac{ln(15)}{2}} \approx 1.16 \tag{3}$$

   Therefore, slot machine 2 should be chosen.

   (b) With ϵ-Greedy:

$$Machine1 = \frac{4}{7} \approx 0.57 \tag{4}$$

$$Machine2 = \frac{3}{5} \approx 0.6 \tag{5}$$

$$Machine3 = \frac{0}{2} \approx 0 \tag{6}$$

We generate a random number between 0 and 1. If the value is greater than ϵ, slot machine 2 would be chosen; otherwise, another random machine would be selected.

8. In tasks with small, finite state sets and action space, we should use tabular methods. However if there are far more states and action spaces are complex and large that exceeds memory support, we should use function approximator.