COMP9444 Neural Networks and Deep Learning

Quiz 8 Deep RL and Unsupervised Learning

This is an optional quiz to test your understanding of Deep RL and Unsupervised Learning.

---

1. Write out the steps in the REINFORCE algorithm, making sure to define any symbols you use.

   for each trial
   run trial and collect states s_t, acions a_t and reward r_{total}
   for t = 1 to length(trial)
   $\theta \leftarrow \theta + \eta(r_{\text{total}} - b)\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)$
   end
   end

   θ = parameters of policy, η = learning rate,
   r_{total} = total reward received during trial,
   b = baseline (constant), ∇_θ = gradient with respect to θ,
   $\pi_\theta(a \mid s)$ = probability of performing action a in state s.

2. In the context of Deep Q-Learning, explain the following:

   a. Experience Replay

      The agent(s) choose actions according to their current Q-function, using an ε-greedy strategy, and contribute to a central database of experiences in the form (s_t, a_t, r_t, s_{t+1}). Another thread samples experiences asynchronously from the experience database, and updates the Q-function by gradient descent, to minimize

      $[r_t + \gamma \max_b Q_w(s_{t+1},b) - Q_w(s_t,a_t)]^2$

   b. Double Q-Learning

      Two sets of Q values are maintained. The current Q-network w is used to select actions, and a slightly older Q-network $\bar{w}$ is used for the target value.

3. What is the Energy function for these architectures:

   a. Boltzmann Machine
   b. Restricted Boltzmann Machine

   Remember to define any variables you use.

   a. Boltzmann Machine

      $E(x) = -(\Sigma_{i<j}\, x_i\, w_{ij}\, x_j + \Sigma_i\, b_i\, x_i)$

      where x_i = activation of node i (0 or 1)

b. Restricted Boltzmann Machine

$E(v, h) = -(\Sigma_i\, b_i\, v_i + \Sigma_j\, c_j\, h_j + \Sigma_{i,\,j}\, v_i\, w_{ij}\, h_j)$

where v_i = visible unit activations, h_j = hidden unit activations

4. The Variational Auto-Encoder is trained to maximize

$\mathbf{E}_{z \sim q_\phi(z \mid x^{(i)})} [\log p_\theta(x^{(i)} \mid z)] \quad - \quad D_{KL}(q_\phi(z \mid x^{(i)}) \parallel p(z))$

Briefly state what each of these two terms aims to achieve.

The first term enforces that any sample z drawn from the conditional distribution $q_\phi(z \mid x^{(i)})$ should, when fed to the decoder, produce something approximationg $x^{(i)}$.
The second term encourages the distribution $q_\phi(z \mid x^{(i)})$ to approximate the Normal distribution $p(z)$ (by minimizing the KL-divergence between the two distributions)

5. Generative Adversarial Networks traditionally made use of a two-player zero-sum game between a Generator G_θ and a Discriminator D_ψ, to compute

$\min_\theta \max_\psi (V(G_\theta, D_\psi))$

a. Give the formula for $V(G_\theta, D_\psi)$.

$V(G_\theta, D_\psi) = \mathbf{E}_{x \sim p_{data}} [\log D_\psi(x)] + \mathbf{E}_{z \sim p_{model}} [\log(1 - D_\psi(G_\theta(z)))]$

b. Explain why it may be advantageous to change the GAN algorithm so that the game is no longer zero-sum, and write the formula that the Generator would try to maximize in that case.

The quality of the generated images tends to improve if the Generator instead tries to maximize

$\mathbf{E}_{z \sim p_{model}} [\log(D_\psi(G_\theta(z)))]$

This forces the Generator to put emphasis on improving the poor-quality images, rather than taking the images that are already good and making them slightly better.

6. In the context of GANs, briefly explain what is meant by *mode collapse*, and list three different methods for avoiding it.

Mode collapse is when the Generator produces only a small subset of the desired range of images, or converges to a single image (with minor variations). Methods for avoiding mode collapse include: Conditioning Augmentation, Minibatch Features and Unrolled GANs.
