This quiz covers units 7, 8, and 9. Each question is worth 25%. Please make sure to follow the hand-in instructions described in Canvas announcements and in the course website.

Question 1: Given two probability density functions p and q, the Jensen-Shannon (JS) Divergence is,

$$D_{JS}(p||q) = \frac{1}{2}\Big(D_{KL}\big(p \;||\; \frac{p+q}{2}\big) + D_{KL}\big(q \;||\; \frac{p+q}{2}\big)\Big),$$

where,

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

Consider the vanilla GAN (original GAN) formulation,

$$\min_G \max_D V(D, G),$$

with,

$$V(D, G) = \mathbb{E}_{X \sim p_{\text{data}}}[\log D(X)] + \mathbb{E}_{Z \sim p_z}[\log(1 - D(G(Z)))].$$

Here p_{data} is the density of the data and p_z is the density of the noise. Now given a generator $G(\cdot)$, p_g is the probability distribution of $G(Z)$ where $Z \sim p_z$.

(a) Show that for fixed generator $G(\cdot)$, the optimal discriminator is

$$D^*(x) = \frac{p_{\text{data}(x)}}{p_{\text{data}(x)} + p_g(x)}.$$

(b) Show now that,

$$V(D^*, G) = -2\log 2 + D_{JS}(p_{\text{data}} \;||\; p_g).$$

Solution:
(a)

$$\begin{aligned} V(D, G) &= \int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log(D(\boldsymbol{x}))dx + \int_z p_{\boldsymbol{z}}(\boldsymbol{z}) \log(1 - D(G(\boldsymbol{z})))dz \\ &= \int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log(D(\boldsymbol{x})) + p_g(\boldsymbol{x}) \log(1 - D(\boldsymbol{x}))dx. \end{aligned}$$

Now consider the function $f(y) = a\log(y) + b\log(1 - y)$. It achieves it's maximum at,

$$y = \frac{a}{a+b},$$

so for any fixed x. This can be shown by differentiation. Hence,

$$\begin{aligned} &\int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log(D(\boldsymbol{x})) + p_g(\boldsymbol{x}) \log(1 - D(\boldsymbol{x}))dx \\ &\leq \int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log\Big(\frac{p_{\text{data}(x)}}{p_{\text{data}(x)} + p_g(x)}\Big) + p_g(\boldsymbol{x}) \log\Big(1 - \frac{p_{\text{data}(x)}}{p_{\text{data}(x)} + p_g(x)}\Big)dx \\ &:= C(G) = V(D^*, G). \end{aligned}$$

And this yields $D^*(\cdot)$ as the optimal discriminator for fixed G.

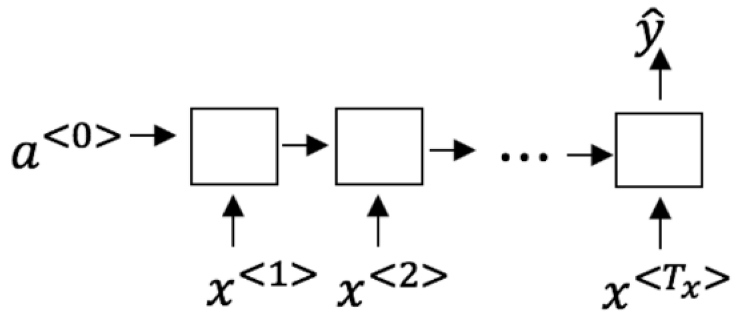
(b) Just notice that,

$$
\begin{aligned}
2D_{JS}(p_{\text{data}} \mid\mid \ p_g) &= D_{KL}\big(p_{\text{data}} \mid\mid \frac{p_{\text{data}} + p_g}{2}\big) + D_{KL}\big(p_g \mid\mid \frac{p_{\text{data}} + p_g}{2}\big) \\
&= \int_x p_{\text{data}}(x) \log 2 \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)}\, dx. + \int_x p_g(x) \log 2 \frac{p_g(x)}{p_{\text{data}}(x) + p_g(x)}\, dx.
\end{aligned}
$$

Or after rearranging (while observing the integral of a probability distribution over the whole domain is 1),

$$
\begin{aligned}
2D_{JS}(p_{\text{data}} \mid\mid \ p_g) - 2\log 2 &= \int_x p_{\text{data}}(x) \log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)}\, dx. + \int_x p_g(x) \log \frac{p_g(x)}{p_{\text{data}}(x) + p_g(x)}\, dx. \\
&= V(D^*, G).
\end{aligned}
$$

Question 2: Consider an RNN described by the following illustration:



(a) Is this a “one-to-one“, “one-to-many“, “many-to-one“, or “many-to-many” architecture?

(b) Can you suggest an application for this type of architecture? Describe the data x and y that you would use for such an application.

(c) Illustrate a one-to-many RNN architecture, using similar notation to the illustration above.

Solution:
(a) This is a “many-to-one” RNN.

(b) One possible application is sentiment analysis where x would be a sequence of (one-hot encoded) words and y would be the sentiment.

(c) Looks just like the above but there is an output y from each step.

Question 3: Let $\mathcal{M} = [2, 1000]$ and let $F : \mathcal{M} \to \mathcal{M}$ be is mapping defined by

$$F(x) = \frac{x}{2} + \frac{2}{x}, \quad x \in \mathcal{M}.$$

Further suppose that $\mathcal{D}$ denotes $\mathcal{L}_1$-norm, i.e., for any $x, y \in \mathcal{M}$,

$$\mathcal{D}(x, y) = |x - y|.$$

(a) Is F a contraction mapping on the metric space $(\mathcal{M}, \mathcal{D})$?

(b) If your answer to (a) is 'no', find two points $x, y \in \mathcal{M}$ that satisfy

$$\mathcal{D}(F(x), F(y)) \geq \mathcal{D}(x, y). \tag{1}$$

If your answer to (a) is 'yes', what is the fixed point of F in $\mathcal{M}$?

Hint: observe that, for all $x, y \in \mathcal{M}$,

$$0 \leq \frac{1}{2}\left(1 - \frac{4}{xy}\right) \leq \frac{1}{2}.$$

Solution: First observe that, for all $x, y \in \mathcal{M}$,

$$|F(x) - F(y)| = \left|\frac{x}{2} + \frac{2}{x} - \frac{y}{2} - \frac{2}{y}\right| = \left|\left(\frac{x}{2} - \frac{y}{2}\right) - \frac{4}{xy}\left(\frac{x}{2} - \frac{y}{2}\right)\right| = \frac{1}{2}\left(1 - \frac{4}{xy}\right)|x - y|.$$

Thus, $|F(x) - F(y)| \leq \frac{1}{2}|x - y|$.

Therefore, the answer to (a) is 'yes': F is a contraction mapping on the metric space $(\mathcal{M}, \mathcal{D})$. So, there is no pair (x, y) with $x, y \in \mathcal{M}$ that satisfies (1).

To find the fixed point, solve $F(x) = x$.: That is

$$\frac{x}{2} + \frac{2}{x} = x.$$

So, the solutions are $x = \pm 2$. Since $2 \in \mathcal{M}$, the unique fixed point of F in $\mathcal{M}$ is 2.

Question 4: Note that parts (a) and (b) of this question are independent.

(a) Consider the value iteration algorithm for a Markov decision process (S, A, T, R, γ) with discount factor $\gamma = 1/2$. Start the algorithm with zero initial values, i.e., $\mathbf{v}_0 = \mathbf{0}$. Further assume that the expected immediate rewards for each state-action pair is non-negative, i.e., for all $(s, a) \in S \times A$,

$$r(s,a) = \mathbb{E}_{s' \sim T(s,a,\cdot)}\big[R(s,a,s')\big] \geq 0.$$

Recall that $\mathbf{V}^*$ denotes the optimal value function and $\mathbf{v}_k$ denotes the value approximation of $\mathbf{V}^*$ at the k^{th} iteration. Let $\mathbf{u}_k = \mathbf{V}^* - \mathbf{v}_k$ for each $k = 1, 2, \ldots$.

Show that

$$\max_{s \in S} |\mathbf{u}_{100}(s)| \leq \frac{1}{2^{99}} \max_{(s,a) \in S \times A} r(s,a).$$

(b) The basic iterative algorithms for solving Markov decision processes cannot be used for reinforcement learning (RL) directly as is. Consider each of the items below and state if they are a cause for this or not.

(i) Computing the expected immediate rewards $r(s, a)$ exactly is not feasible when the transition probabilities $T(s, a, s')$'s are unknown.

(ii) In the RL framework, the agent cannot fully observe the state of the system.

(iii) We need to know the underlying model of the system to execute the algorithms.

(iv) When the dimension $|S| \times |A|$ of the state-action space is high, these algorithms can be become computationally demanding.

Solution:

(a) From Theorem 9.5 of lecture notes, for all $k = 1, 2, \ldots$, we have that

$$\|\mathbf{v}_k - \mathbf{V}^*\|_\infty \leq \frac{\gamma^k}{1-\gamma} \|\mathbf{v}_1 - \mathbf{v}_0\|_\infty,$$

where $\mathbf{v}_k$ is the approximation of value function in the k^{th} iteration. Therefore, by substituting $\mathbf{v}_0 = \mathbf{0}$, $\gamma = 1/2$ and $\mathbf{u}_k = \mathbf{v}_k - \mathbf{V}^*$, we get,

$$\|\mathbf{u}_k\|_\infty \leq \frac{1}{2^{k-1}} \|\mathbf{v}_1\|_\infty.$$

Using the update of the value function, for each state $s \in S$,

$$\mathbf{v}_1(s) = \max_{a \in A} r(s,a) \geq 0.$$

Thus,

$$\|\mathbf{v}_1\|_\infty = \max_{s \in S} \mathbf{v}_1(s) = \max_{(s,a) \in S \times A} r(s,a).$$

Also,

$$\|\mathbf{u}_k\|_\infty = \max_{s \in S} |\mathbf{u}_k(s)|.$$

We have the result by taking $k = 100$.

(b)

Items (i), (iii), and (iv) are causes for the basic iterative algorithms not directly working in the RL framework. (ii) is not a cause.

In the RL framework the transition probabilities of the model are unknown to the agent, hence (i) and (iii) are causes. Item (iv) is a cause because computing expected immediate rewards $\{r(s,a) : (s,a) \in S \times A\}$ and updating the values function for every pair (s,a) in every iteration can be computationally demanding when $|S| \times |A|$ is large.