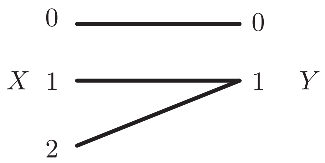
Introduction to Information and Coding Theory June 24, 2019
(Prof. Permuter Haim, Mr. Oron Sabag and Mr. Ben Marinberg)

Final Exam - Moed A
Total time for the exam: 3 hours!

Important: For **True / False** questions, copy the statement to your notebook and write clearly true or false. You should prove the statement if true, and provide counterexample otherwise.

1) **Optimal outputs distribution (33 Points)**: Consider the following deterministic channel:



a) (4 Points) Find the capacity .
b) (5 Points) **True/False** The optimal input distribution $P(x)$ unique (there is only one such distribution).
c) (6 Points) **True/False** The optimal output distribution $P(y)$ unique.

We will now show the phenomena above for general memoryless channels.
Define the joint distribution $P(\theta, x, y) = P(\theta)P(x|\theta)P(y|x)$.

d) (3 Points) Put sign between

$$I(X;Y) \ \ ?? \ \ I(X;Y|\theta).$$

e) (5 Points) Find a sufficient and necessary condition for equality above.
f) (10 Points) **True/False** For memoryless channels, the optimal outputs distribution is unique.

Solution Q1:

a) The capacity is $C = 1$. First $C \leq 1$ since $c \leq \log 2|\mathcal{Y}| = 1$. It can be achieved using $P_X(1) = P_X(2) = 0.5$ and $P_X(3) = 0$.
b) False. Any input distribution with $P(x = 0) = 0.5$ achieves the capacity.
c) True. The capacity is $\max_{P(x)} H(Y)$ with a unique maximum when $P(y) = 0.5$ for $y = 0, 1$
d) Consider,

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y|X,\theta) && (1)\\ &\geq H(Y|\theta) - H(Y|X,\theta) && (2)\\ &= I(X;Y|\theta), && (3)\end{aligned}$$

where the first inequality follows from the Markov chain and the inequality following from conditioning reduces entropy.
e) The inequality above holds with equality iff $I(Y;\theta) = 0$. That is, Y and θ are independent.
f) True. For a memoryless channel $P(y|x)$, consider two input distributions, $P_1(x)$ and $P_2(x)$, that achieve the capacity. As we saw in class, define $P(x|\theta = 1) = P_1(x)$ and $P(x|\theta = 2) = P_2(x)$.
Consider

$$\begin{aligned} C &\geq I(X;Y) && (4)\\ &\geq I(X;Y|\theta) && (5)\\ &= C, && (6)\end{aligned}$$

where the second inequality follows from the previous question, and the equality follows from the assumption that they input distributions achieve capacity. Thus, have that $I(X;Y) = I(X;Y|\theta)$. Lastly, a necessary condition is Y is independent of θ meaning that $P_1(y) = P_2(y)$ (the optimal output distribution is the same for $P_1(x)$ and $P_2(x)$).

2) **Divergence between Markov processes with the same transition probability, True or False (25 Points)**: Prove each relation or provide counter example.

a) (6 Points) Given are two input distributions P_X and Q_X that are transmitted on the same channel given by $P_{Y|X}$. The corresponding output distributions are denoted by P_Y and Q_Y.

$$D(P_X||Q_X) \geq D(P_Y||Q_Y). \qquad (7)$$

b) (12 Points) Define a Markov transition probability that, given x_1, generates $X_2, X_3, \ldots$ using $P_{X^+|X}$. For example, at time $t = 4$, the joint probability given x_1 is:

$$P(x_4, x_3, x_2|x_1) = P_{X^+|X}(x_4|x_3)P_{X^+|X}(x_3|x_2)P_{X^+|X}(x_2|x_1)$$

There are two distributions P_{X_1} and Q_{X_1} that serve as the initial distribution on X_1. Is the following true,

$$D(P_{X_i}||Q_{X_i}) \geq D(P_{X_j}||Q_{X_j}) \quad \text{for all } i \leq j. \tag{8}$$

c) (7 Points) Let $f(t)$ and $g(t)$ be convex functions over $t \in [0,1]$. Is the function $h(t) = \max(f(t), g(t))$ convex?

Solution Q2:

a) True. Consider

$$\begin{aligned} D(P_X||Q_X) &= D(P_{X,Y}||Q_{X,Y}) &(9)\\ &= D(P_Y||Q_Y) + D(P_{X|Y}||Q_{X|Y}|P_Y) &(10)\\ &\geq D(P_Y||Q_Y). &(11) \end{aligned}$$

where the first equality follows from $P_{Y|X} = Q_{Y|X}$, the second follows from the chain rule and the inequality follows from the non-negativity of (conditional) divergence.

b) True. The Markov transition can be thought as the channel from the previous question. Using the previous question, we use induction to show:

$$D(P_{X_i}||Q_{X_i}) \leq D(P_{X_{i+1}}||Q_{X_{i+1}}) \leq D(P_{X_{i+2}}||Q_{X_{i+2}}) \leq \cdots \leq D(P_{X_j}||Q_{X_j}).$$

c) True. Without loss of generality, assume that for $\lambda t_1 + \bar{\lambda} t_2$,

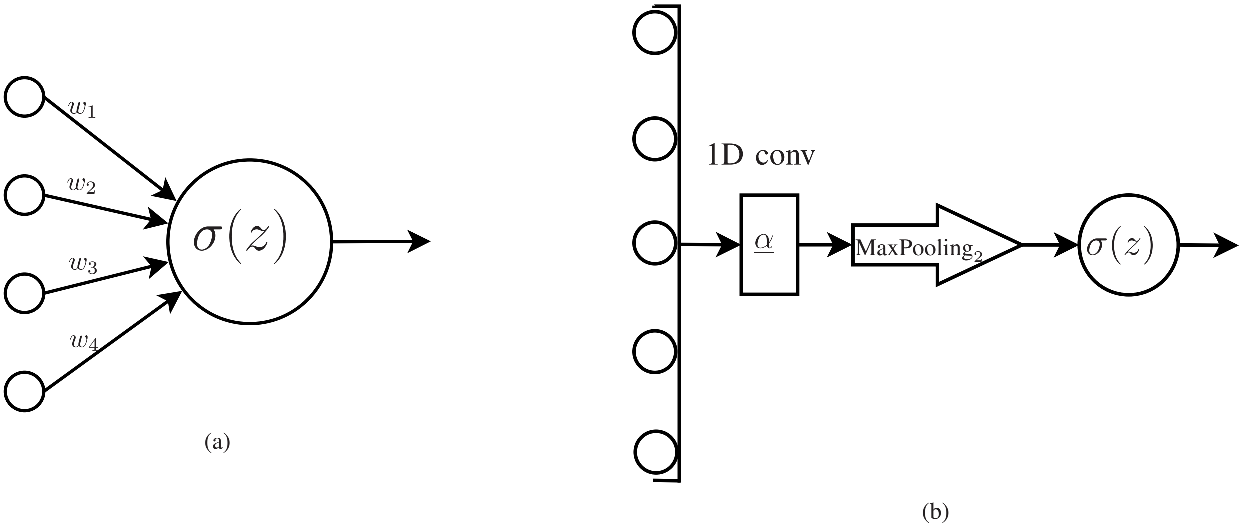
$$h(\lambda t_1 + \bar{\lambda} t_2) = f(\lambda t_1 + \bar{\lambda} t_2).$$

Consider the following derivation,

$$\begin{aligned} h(\lambda t_1 + \bar{\lambda} t_2) &= f(\lambda t_1 + \bar{\lambda} t_2) &(12)\\ &\leq \lambda f(t_1) + \bar{\lambda} f(t_2) &(13)\\ &\leq \lambda \max\{f(t_1), g(t_1)\} + \bar{\lambda} \max\{f(t_2), g(t_2)\} &(14)\\ &= \lambda h(t_1) + \bar{\lambda} h(t_2). &(15) \end{aligned}$$

where the first equality follows from the assumption, the second inequality follows from the convexity of $f(\cdot)$, the third is trivial and the last equality is by the definition of $h(\cdot)$.

3) **Derivatives of neural Network (32 Points)**:



(a)

(b)

Fig. 1

a) (12 Points) Consider the neural network in Fig. (1a) with no bias. Calculate the derivative of w_2 for both MSE and cross entropy cost functions, i.e. $\frac{\partial C_{MSE}}{\partial w_2}$ and $\frac{\partial C_{CE}}{\partial w_2}$ given $x = [0.5, 0.6, 0.1, 0.4]^T$,$y = 0$, $w = [0.25, 0.5, 0.8, 0.3]$ and σ is the sigmoid function.
Reminder: The MSE cost is $C_{MSE}(y,a) = \frac{1}{2}(y-a)^2$. Binary cross entropy cost is $C_{BCE}\ (y,a) = -y\log(a) - (1-y)\log(1-a)$.

Solution (a):
Note that a=$\sigma(z)$, and $\sigma'(z)$=$\sigma(z)$(1-$\sigma(z)$).

$$z = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 = 0.625 \tag{16}$$

$$\sigma(z) = \sigma(0.625) = 0.651 \tag{17}$$

Using the chain rule, we achieve:

$$\frac{\partial C_{MSE}}{\partial w_2} = \frac{\partial C_{MSE}}{\partial \sigma(z)} \frac{\partial \sigma(z)}{\partial z} \frac{\partial z}{\partial w_2} = (\sigma(z) - y)\sigma'(z)x_2 = (\sigma(z) - y)\sigma(z)(1 - \sigma(z))x_2 = 0.0887 \tag{18}$$

Now using the chain rule with respect to the Cross Entropy cost, we achieve:

$$\frac{\partial C_{CE}}{\partial w_2} = \frac{\partial C_{CE}}{\partial \sigma(z)} \frac{\partial \sigma(z)}{\partial z} \frac{\partial z}{\partial w_2} = \frac{\sigma(z) - y}{\sigma(z)(1 - \sigma(z)}\sigma'(z)x_2 = \frac{\sigma(z) - y}{\sigma(z)(1 - \sigma(z)}\sigma(z)(1 - \sigma(z))x_2 = (\sigma(z) - y)x_2 = 0.375 \tag{19}$$

b) (15 Points) Now we introduce you to two new layers of neural networks, 1D convolution and MaxPooling$_\beta$. 1D convolution pass the input layer in a FIR filter of size M, without zero padding. For input $x = [x(1), x(2), .., x(N)]^T$ and filter coefficients $\alpha = [\alpha(0), \alpha(1), .., \alpha(M-1)]$, the output y is defined as:

$$y(k) = \sum_{i=0}^{M-1} x(k+i) \cdot \alpha(i) \qquad k = 1, 2, .., N - M + 1. \tag{20}$$

MaxPooling$_\beta$ just pools the maximum value of β consecutive elements. For input $x = [x(1), x(2), .., x(N)]^T$ and some β, the output y is defined as:

$$y(k) = \max\{x(k), x(k+1), .., x(k+\beta-1)\} \qquad k = 1, 2, .., N - \beta + 1. \tag{21}$$

Consider the neural network in Fig. (1b) with MSE cost and no bias, calculate the derivative of w_2 , i.e. $\frac{\partial C_{MSE}}{\partial w_2}$, given $x = [0.5, 0.6, 1, 0.1, 0.4]^T$, $y = 0$, $\alpha = [0.25, 0.5, 0.25]$, $\beta = 2$ and $w = [0.25, 0.5]$.

Solution (b):
Step 1: Calculate feedforward. M=3, $\beta = 2$, N=5.

$$\begin{align}
y_{FIR}(1) &= x(1)\alpha(0) + x(2)\alpha(1) + x(3)\alpha(2) = 0.675 \tag{22}\\
y_{FIR}(2) &= x(2)\alpha(0) + x(3)\alpha(1) + x(4)\alpha(2) = 0.675 \tag{23}\\
y_{FIR}(3) &= x(3)\alpha(0) + x(4)\alpha(1) + x(5)\alpha(2) = 0.400 \tag{24}\\
& \tag{25}
\end{align}$$

Calculations result in: $y_{FIR} = [0.675, 0.675, 0.4]^T$
Now calculate the output of the Maxpooling:

$$\begin{align}
y_{M1} &= \max\{0.675, 0.675\} = 0.675 \tag{26}\\
y_{M2} &= \max\{0.675, 0.400\} = 0.675 \tag{27}\\
& \tag{28}
\end{align}$$

Calculations result in: $y_{Maxpooling} = [0.675, 0.675]^T$

$$\begin{align}
z &= w_1 y_{M1} + w_2 y_{M2} = 0.506 \tag{29}\\
\sigma(z) &= \sigma(0.506) = 0.623 \tag{30}\\
& \tag{31}
\end{align}$$

Step 2: Calculate the derivative using section (a):

$$\frac{\partial C_{MSE}}{\partial w_2} = \frac{\partial C_{MSE}}{\partial \sigma(z)} \frac{\partial \sigma(z)}{\partial z} \frac{\partial z}{\partial w_2} = (\sigma(z) - y)\sigma(z)(1 - \sigma(z) y_{M2} = 0.098 \tag{32}$$

c) (5 Points) A soft max with D input $z = [z(1), .., z(D)]$ and N outputs is

$$a(j) = \frac{\mathrm{e}^{z(j)}}{\sum_{n=1}^{D} \mathrm{e}^{z(n)}} \text{ for } j = 1, 2, ..., N. \tag{33}$$

Find $\frac{\partial a(i)}{\partial z(j)}$ and the sign of the derivative (positive or negative), i.e., sign $\left(\frac{\partial a(i)}{\partial z(j)}\right)$.

Solution (c):
We will split the answer into 2 cases:
i=j:

$$\frac{\partial a(j)}{\partial z(j)} = \frac{\mathrm{e}^{z(j)}(\sum_{n=1}^{D}\mathrm{e}^{z(n)}) - \mathrm{e}^{2z(j)}}{(\sum_{n=1}^{D}\mathrm{e}^{z(n)})^2} = \mathrm{e}^{z(j)}\frac{\sum_{n=1}^{D}\mathrm{e}^{z(n)} - \mathrm{e}^{z(j)}}{(\sum_{n=1}^{D}\mathrm{e}^{z(n)})^2} \tag{34}$$

$$\tag{35}$$

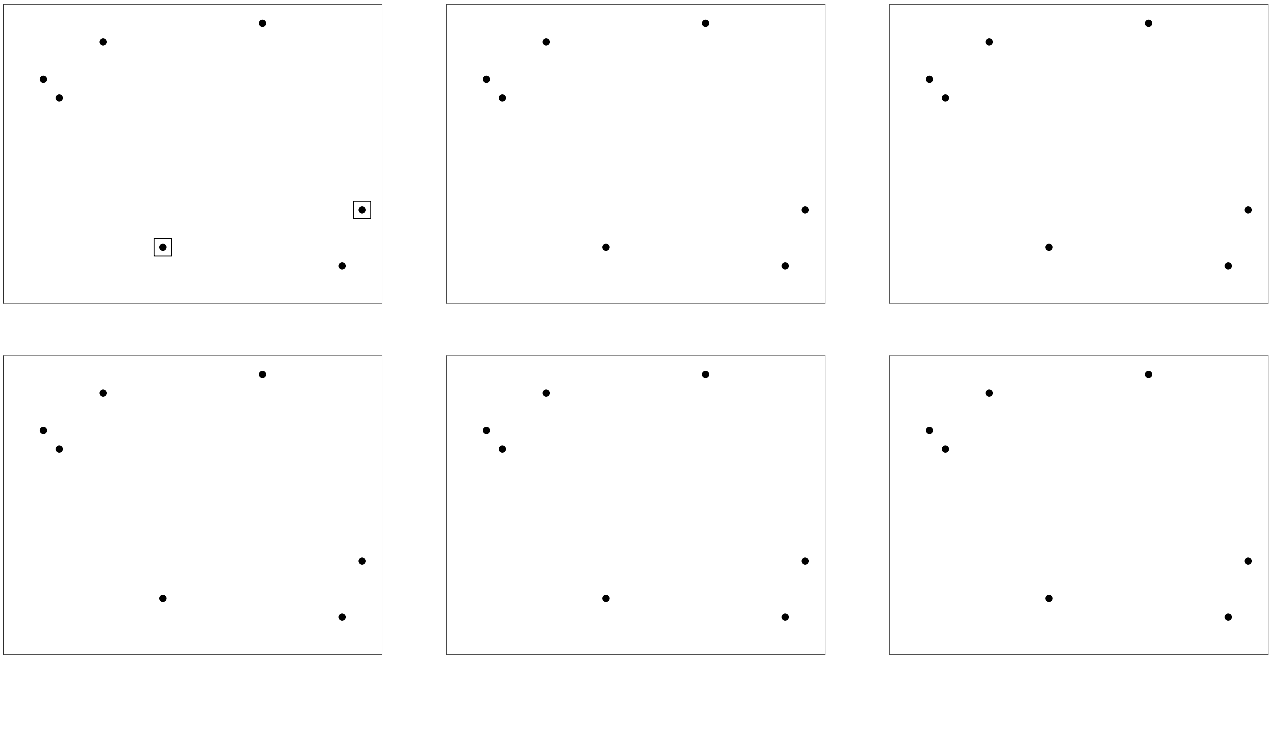
i≠j:

$$\frac{\partial a(i)}{\partial z(j)} = \frac{-\mathrm{e}^{z(i)}\mathrm{e}^{z(j)}}{(\sum_{n=1}^{D}\mathrm{e}^{z(n)})^2} \tag{36}$$

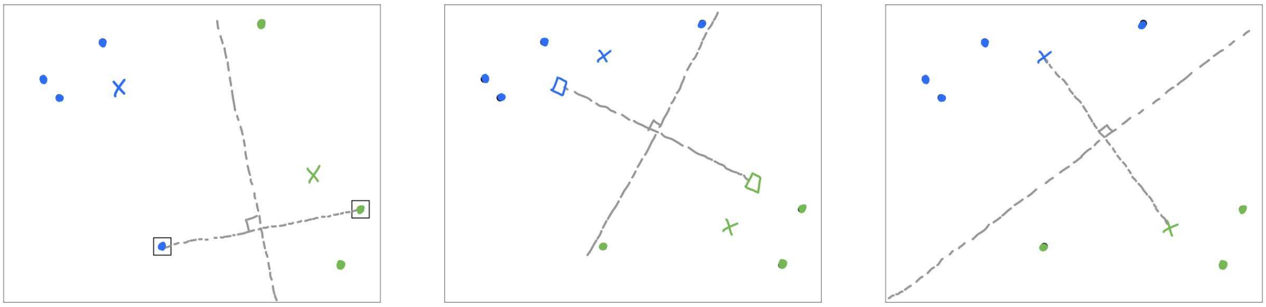
$$\tag{37}$$

Note that $\mathrm{e}^{z(j)} > 0$ and that $\sum_{n=1}^{D}\mathrm{e}^{z(n)} > \mathrm{e}^{z(j)} > 0$ resulting in a positive derivative for i=j and a negative derivative for i≠j. This makes sense since softmax represents probability. If z(j) increases then its probability will rise. Since probability adds up to 1, the other probabilities will decrease, resulting in a negative derivative.

4) **K-means (15 Points)** Perform K-means iterations by hand on the dataset given below. Circles are data points and the two initial cluster centers are squares. Draw the cluster centers and the decision boundaries that define each cluster. If no points belong to a particular cluster, assume its center does not change. Use as many of the pictures as you need for convergence.



Solution (K-means):
Using 3 iterations of the K-means algorithm studied in class we achieve:



Note that Squares resemble the initial centroid position, the colored dots (green/blue) resemble the assignment of a dot to a specific centroid, the X's resemble the estimation of the new centroid position according to the dots average location and the gray lines resemble the decision boundary derived using minimal euclidean distance. After 3 iterations the centroids will not change anymore, meaning the algorithm is done.

Good Luck!