دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

هوش مصنوعی
بهار ۱۴۰۲
استاد: محمدحسین رهبان

آزمون پایان‌ترم

- زمان در نظر گرفته شده برای آزمون ۱۵۰ دقیقه است.
- لطفا پاسخ‌های خود را خوانا و خوش خط بنویسید.
- پاسخ هر سوال باید در یک برگه جداگانه نوشته شود. بالای هر برگه ی پاسخ نامه، نام و شماره دانشجویی خود را به صورت واضح بنویسید.

سوالات (۵+۱۰۰ نمره)

۱. (۲۰ نمره) به سوالات زیر به طور مختصر پاسخ دهید:

(آ) لایه pooling به‌دلیل نداشتن وزنی برای یادگیری تاثیری در backpropagation ندارد. این عبارت درست است یا غلط؟ باذکر دلیل مشخص کنید.

(ب) یک شبکه عصبی fully connected را درنظر بگیرید که تابع فعالسازی تمام لایه‌ها تابع tanh می‌باشد. برای مقداردهی اولیه وزن‌ها، همه وزن‌های شبکه را مقادیری بزرگ انتخاب می‌کنیم. این روش ایده خوبی برای کارکرد این شبکه عصبی نیست. این عبارت درست است یا غلط؟ باذکر دلیل مشخص کنید. راهنمایی:

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

(ج) اشکال تخمین زدن احتمال $P(A|B)$ به صورت $\frac{\text{تعداد رخداد A و B با یکدیگر}}{\text{تعداد رخداد B}}$ چیست؟ برای بهبود آن چه روشی را پیشنهاد می‌کنید؟

(د) توضیح دهید هرکدام از روش‌های زیر چه تأثیری در بیش‌برازش[۱] کردن مدل‌ها دارند.

- کم کردن تعداد برگ‌ها در درخت تصمیم
- محدود کردن حداکثر طول درخت تصمیم

درباره‌ی تأثیر آن‌ها در دقت مدل در داده‌های آموزش چه می‌توان گفت؟

(ه) عبارت زیر درست است یا غلط؟ با ذکر دلیل مشخص کنید. یک شبکه بیزی داریم که در آن X به شرط Z از Y مستقل است. ممکن است فرض استقلال این دو متغیر با شرطی کردن شواهد اضافه برای متغیرهای دیگر در شبکه، برقرار نباشد.

حل.

(آ) غلط است. زیرا همچنان نیاز به محاسبه مشتق نسبت به ورودی را داریم و همچنان در گرادیانی که در الگوریتم backpropagation ایجاد می‌شود تاثیرگذار است.

[۱] Overfit

(ب) درست است. زیرا مشتق تابع $tanh(x)$ برابر است با:

$$\frac{\mathrm{d}}{\mathrm{d}x}\tanh x = ۱ - \tanh x^{۲} = \frac{۴}{(e^{x}+e^{-x})^{۲}}$$

در این صورت وقتی که مقدار x بسیار بزرگ شود، باعث می‌شود که مقدار گرادیان تابع فعالساز نسبت به ورودی‌ها که مقدار بزرگی دارند، ناچیز بشود و به صفر میل کند و در این صورت پدیده gradient vanishing رخ بدهد.

(ج) ممکن است اصلا B رخ نداده باشد که باعث می‌شود این کسر تعریف نشود. یا اینکه A و B با یکدیگر رخ نداده باشند که باعث میشود مقدار صفر پیش‌بینی شود که می‌تواند در مواردی (مانند Naive Bayes Classifier) دردسرساز شود. برای بهبود آن می‌توان از Laplace Smoothing استفاده کرد.

(د) دو مورد اول باعث کاهش بیش‌برازش شدن مدل می‌شود و مورد آخر می‌تواند باعث افزایش بیش‌برازش مدل شود.
همچنین دو مورد اول با محدودتر کردن مدل، باعث کاهش دقت مدل روی داده‌های آموزش می‌شوند.

(ه) درست، چون مجموعه ای از متغیرها به نام W می توانند در شبکه موجود باشند که به دلیل وجود ساختار v شکل در میان آن ها گزاره برقرار نباشد.

۲. (۱۵ نمره) می‌خواهیم یک شبکه عصبی طراحی کنیم که عبارت زیر را پیاده سازی کند:

$$(x \vee \neg y) \oplus (\neg m \vee \neg n)$$

(علامت $\neg$ معادل not منطقی، علامت $\oplus$ معادل xor منطقی و علامت $\vee$ معادل or منطقی است.) در طراحی شبکه‌های عصبی تابع فعال‌سازی را به صورت زیر در نظر بگیرید:

$$f(x) = \begin{cases} ۱ & x > ۰ \\ ۰ & x \leq ۰ \end{cases}$$

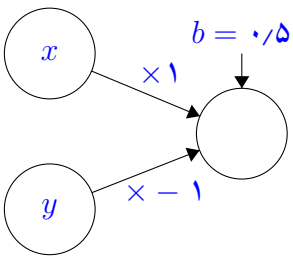
برای این منظور ابتدا برای عبارت‌های زیر شبکه عصبی طراحی کنید:

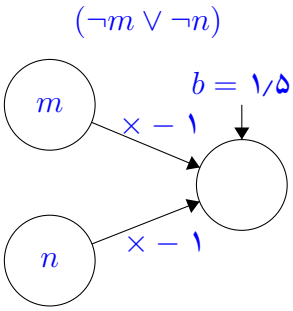
- $(x \vee \neg y)$
- $(\neg m \vee \neg n)$
- $A \oplus B$

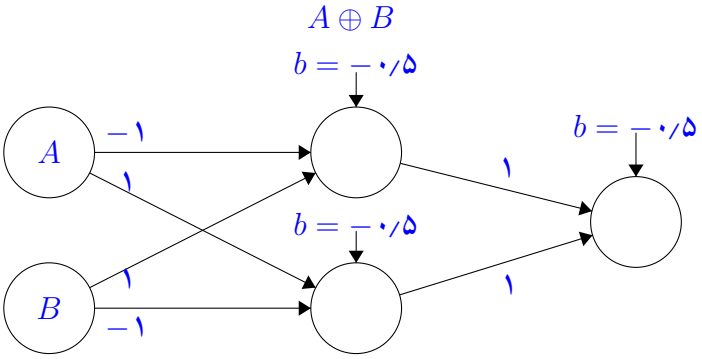
سپس با ترکیب شبکه‌های به دست آمده برای عبارت گفته‌شده شبکه عصبی طراحی کنید. (توجه کنید که وزن‌ها و بایاس‌ها را به صورت دقیق مشخص کنید.)

حل.

- $(x \vee \neg y)$

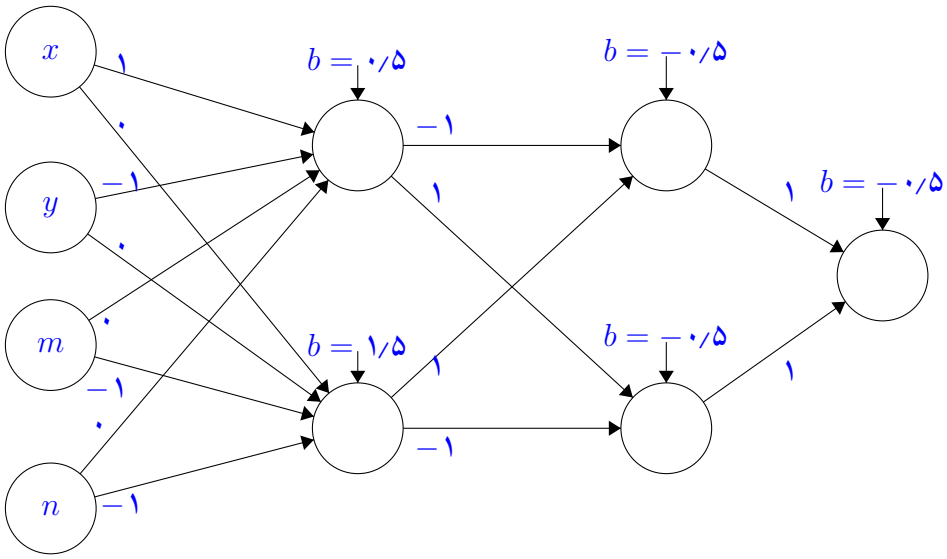


- 

- 

بنابراین شبکه عصبی مربوط به عبارت گفته شده، به صورت زیر خواهد بود:

$$(x \vee \neg y) \oplus (\neg m \vee \neg n)$$



۳. (۲۰ نمره) مسئله‌ی رگرسیون خطی $\hat{y} = w^T x$ را برای مجموعه داده‌ی $D = (x_i, y_i)_{i=۱}^n$ و با تابع زیان $J(w) = \sum_{i=۱}^n (y^{(i)} - \hat{y}^{(i)})^۲$ در نظر بگیرید.

(آ) ابتدا $argmin_w J(w)$ را به دست آورده و ساده کنید.

(ب) فرمول به دست آمده در قسمت قبل ممکن است چه مشکلاتی را به همراه داشته باشد؟ در این قسمت $argmin_w J(w) + \lambda ||w||^۲$ را بدست آورید و ساده کنید. سپس توضیح دهید این عبارت چگونه مشکل قسمت قبل را حل می‌کند.

حل. الف) برای یافتن مقدار w که تابع ضرر $J(w)$ را به حداقل می رساند، باید مشتق $J(w)$ را با توجه به w روی ۰ تنظیم کنیم.

$$\frac{\partial J}{\partial w_i} = \frac{\partial}{\partial w_i} \sum_{k=1}^{m} (y^{(k)} - \hat{y}^{(k)})^2$$

$$\implies \frac{\partial J}{\partial w_i} = \sum_{k=1}^{m} \frac{\partial}{\partial w_i} (y^{(k)} - \hat{y}^{(k)})^2$$

$$\implies \frac{\partial J}{\partial w_i} = \sum_{k=1}^{m} 2(y^{(k)} - \hat{y}^{(k)})(-x_i^{(k)}) = -2 \sum_{k=1}^{m} (y^{(k)} - \hat{y}^{(k)}) x^{(k)}$$

ماتریس X را طوری تعریف کنیم که برابر $[x_1, x_2, \cdots, x_m]$ باشد و هریک از x_i برابر ستون‌های ماتریس ورودی باشد. همچنین ماتریس Y را به عنوان $[y_1, y_2, \cdots, y_m]^T$ تعریف می‌کنیم که y_i خروجی ما است، سپس می‌توانیم مشتق خود را به صورت زیر حساب کنیم:

$$\frac{\partial J}{\partial w} = -2X(Y - X^T w)$$

برای اینکه مقدار w^* را بیابیم که J را کمینه می‌کند باید به صورت زیر عمل کنیم:

$$-2X(Y - X^T w^*) = 0 \implies \boxed{w^* = (XX^T)^{-1} XY}$$

ب) اگر معکوس XX^T وجود نداشته باشد، نمی‌توانیم w^* را با فرمول قبلی پیدا کنیم. با اضافه کردن $\lambda ||w||^2$ به تابع هزینه ما داریم:

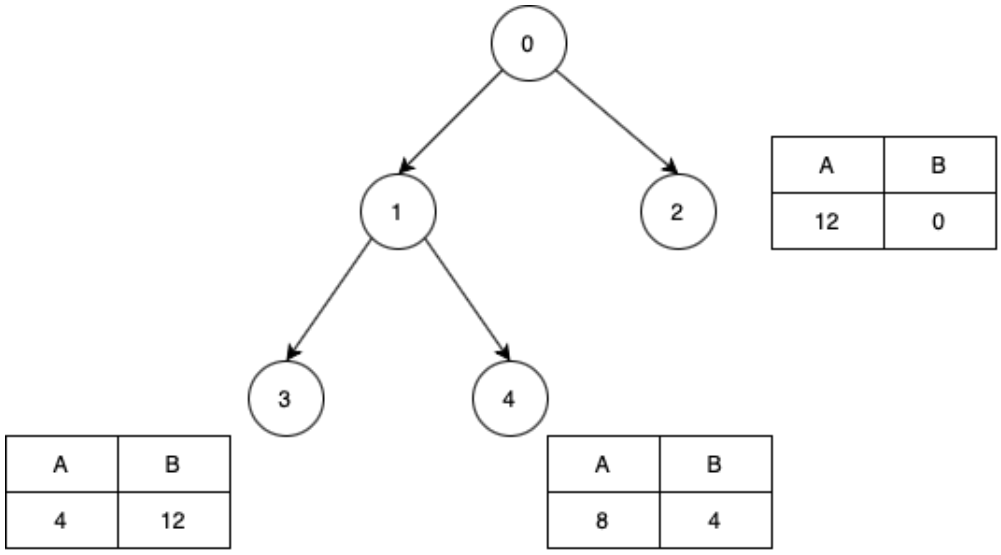
$$\frac{\partial J}{\partial w} = -2X(Y - X^T w) + 2\lambda w$$

$$\implies (XX^T + \lambda I) w^* = XY$$

$$\implies w^* = (XX^T + \lambda I)^{-1} XY$$

حال ما می توانیم لامبدا را طوری انتخاب کنیم که مقادیر ویژه $XX^T + \lambda I$ غیر صفر باشد، بنابراین این ماتریس non-singular خواهد بود و مشکل قسمت قبل حل می‌شود.

۴. (۱۰ نمره) درخت تصمیم زیر را در نظر بگیرید:



الف) آنتروپی و میزان اطلاعات به دست آمده بر اساس گره شماره صفر را به دست آورید.

ب) معیار دقت را برای classification انجام شده بر روی این درخت تصمیم به دست آورید.

پ) پیشنهاد شما برای افزایش این معیار چیست؟ راه حل پیشنهادی شما ممکن است چه مشکلاتی را برای این درخت تصمیم ایجاد کند و راهکار جایگزین شما برای این مورد چیست؟

حل.

الف) در کل ۴۰ سمپل داریم که تعداد ۲۴ سمپل از کلاس A هستند و باقی از کلاس .B پس آنتروپی گره صفر به صورت زیر محاسبه می‌شود:

$$H(۰) = -\frac{۲۴}{۴۰}\log_۲\frac{۲۴}{۴۰} - \frac{۱۶}{۴۰}\log_۲\frac{۱۶}{۴۰}$$

حال به محاسبه‌ی میزان اطلاعات به دست آمده می‌پردازیم:

$$H(۱) = -\frac{۱۲}{۲۸}\log_۲\frac{۱۲}{۲۸} - \frac{۱۶}{۲۸}\log_۲\frac{۱۶}{۲۸}$$

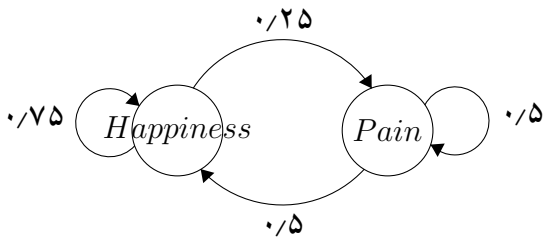
$$H(۲) = ۰$$

$$IG(۰) = H(۰) - (\frac{۲۸}{۴۰}H(۱) + \frac{۱۲}{۴۰}H(۲)) = H(۰) - ۰/۷H(۱)$$

ب) دقت دسته‌بندی انجام شده برابر است با $\frac{۳۲}{۴۰}$ یا همان ۸۰ درصد.

پ) با ادامه دادن گره‌های ۳ و۴ میتوانیم دقت یادگیری را افزایش دهیم اما ممکن است با ریسک overfit شدن مدل مواجه شویم که برای جلوگیری از آن، می‌توانیم درخت عمیق درست کرده و سپس آن را هرس کنیم. چرا که هرس کردن از overfit شدن جلوگیری می‌کند و باعث می‌شود دقت مدل روی داده‌ی تست افزایش یابد. همچنین دیگر پارامترهای یک درخت تصمیم را هم می‌توانیم کنترل کنیم مانند عمق درخت و تعداد فرزندان هر راس.

۵. (۲۰ نمره) markov chain زیر دو استیت گلی (شاد و غمگین) را نشان می‌دهد. در ابتدا در حالت شادی هستیم.



فرض کنید یک سری از استیت‌های گلی به صورت دنباله $x_1, x_2, x_3, ..., x_n$ از این markov chain نمونه‌برداری شده است. ما می‌توانیم گروه‌های این نمونه را تشکیل بدهیم. برای مثال:

$H, H, H, P, P, H, H, P, P, P, P, P$ شامل چهار گروه با اندازه‌های ۳، ۲، ۲، ۵ است.

G_n به صورت زیر تعریف می‌شود:

$$G_n = \frac{n}{\#groups}$$

برای مثال در سوال ما $G_n = \frac{۱۲}{۴} = ۳$ می‌شود.

گلی ادعا کرده است با پیدا کردن مقداری که G_n به آن همگرا می‌شود همیشه در استیت خوشحالی خواهد ماند. مقداری که G_n به آن همگرا می‌شود را پیدا کنید.

حل.

$$G_n = \frac{۱}{۲k} \sum (A_i + B_i)$$

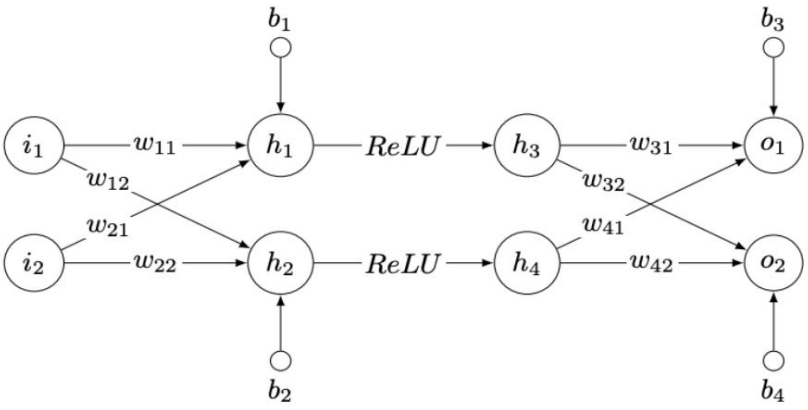
که در اینجا A_i سایز گروه‌های A و B_i سایز گروه‌های B است. $(k = \frac{n}{۲})$

دقت کنید که $A_i \sim Geometric(۰/۲۵)$ و $B_i \sim Geometric(۰/۵)$ هستند.

با میل کردن n به بینهایت این مقدار به تعریف Expected میل می کند و $G_n = \frac{E[Size_A]+E[Size_B]}{۲}$ همگرا می شود.

$$G_n = \frac{\frac{۱}{۰/۲۵} + \frac{۱}{۰/۵}}{۲} = ۳$$

۶. (۲۰ نمره) شبکه عصبی شکل ۱ را با تابع فعال‌سازی ReLU در نظر بگیرید. (i_1, i_2) ورودی هستند، دو لایه مخفی داریم و خروجی‌ها در انتها (o_1, o_2) هستند. برچسب داده‌ها با (t_1, t_2) ، وزن‌ها با w و بایاس با b نشان داده شده است.



شکل ۱ : شبکه عصبی

مقادیر متغیرها را هم می‌توانید در جدول شکل ۲ مشاهده کنید.

Variable	i_1	i_2	w_{11}	w_{12}	w_{21}	w_{22}	w_{31}	w_{32}	w_{41}	w_{42}	b_1	b_2	b_3	b_4	t_1	t_2
Value	2.0	-1.0	1.0	-0.5	0.5	-1.0	0.5	-1.0	-0.5	1.0	0.5	-0.5	-1.0	0.5	1.0	0.5

شکل ۲ : جدول مقادیر متغیرها

(آ) خروجی (o_1, o_2) را با توجه به مقادیر داده شده به دست بیاورید. تمامی محاسبات را بنویسید.

(ب) خطای MSE را حساب کنید.

(ج) فرض کنید تابع هزینه همان قسمت ب باشد. مقدار وزن w_{21} را با کمک gradient descent با نرخ یادگیری ۰/۱ آپدیت کنید. (تمامی محاسبات را بنویسید)

راهنمایی:

$$ReLU(x) = max(0, x)$$

حل.

Forward pass:

$$h_1 = i_1 \times w_{11} + i_2 \times w_{21} + b_1 = 2.0 \times 1.0 - 1.0 \times 0.5 + 0.5 = 2.0$$

$$h_2 = i_1 \times w_{12} + i_2 \times w_{22} + b_2 = 2.0 \times -0.5 + -1.0 \times -1.0 - 0.5 = -0.5$$

$$h_3 = max(0, h_1) = h_1 = 2$$

$$h_4 = max(0, h_2) = 0$$

$$o_1 = h_3 \times w_{31} + h_4 \times w_{41} + b_3 = 2 \times 0.5 + 0 \times -0.5 - 1.0 = 0$$

$$o_2 = h_3 \times w_{32} + h_4 \times w_{42} + b_4 = 2 \times -1.0 + 0 \times 1.0 + 0.5 = -1.5$$

شکل ۳: پاسخ الف

$$MSE = \frac{1}{2} \times (t_1 - o_1)^2 + \frac{1}{2} \times (t_2 - o_2)^2 = 0.5 \times 1.0 + 0.5 \times 4.0 = 2.5$$

شکل ۴: پاسخ ب

Backward pass (Applying chain rule):

$$\begin{aligned}\frac{\partial MSE}{\partial w_{21}} &= \frac{\partial \frac{1}{2}(t_1 - o_1)^2}{\partial o_1} \times \frac{\partial o_1}{\partial h_3} \times \frac{\partial h_3}{\partial h_1} \times \frac{\partial h_1}{\partial w_{21}} + \frac{\partial \frac{1}{2}(t_2 - o_2)^2}{\partial o_2} \times \frac{\partial o_2}{\partial h_3} \times \frac{\partial h_3}{\partial h_1} \times \frac{\partial h_1}{\partial w_{21}} \\ &= (o_1 - t_1) \times w_{31} \times 1.0 \times i_2 + (o_2 - t_2) \times w_{32} \times 1.0 \times i_2 \\ &= (0 - 1.0) \times 0.5 \times -1.0 + (-1.5 - 0.5) \times -1.0 \times -1.0 \\ &= 0.5 + -2.0 = -1.5\end{aligned}$$

Update using gradient descent:

$$w_{21}^{+} = w_{21} - lr * \frac{\partial MSE}{\partial w_{21}} = 0.5 - 0.1 * -1.5 = 0.65$$

شکل ۵: پاسخ پ