



یادگیری ماشین

پاییز ۱۴۰۳

استاد: علی شریفی زارچی

مسئول تمرین: علی شاه حیدر

دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

مهلت ارسال نهایی: ۹ آذر

تمرین سوم

مهلت ارسال امتیازی: ۲ آذر

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روزهای مشخص شده است.
- در طول ترم، برای هر تمرین می‌توانید تا ۵ روز تأخیر مجاز داشته باشید و در مجموع حداکثر ۱۵ روز تأخیر مجاز خواهید داشت. توجه داشته باشید که تأخیر در تمرین‌های عملی و تئوری به صورت جداگانه محاسبه می‌شود و مجموع تأخیر هر دو نباید بیشتر از ۱۵ روز شود. پس از اتمام زمان مجاز، دو روز اضافی برای آپلود غیرمجاز در نظر گرفته شده است که در این بازه به ازای هر ساعت تأخیر، ۲ درصد از نمره تمرین کسر خواهد شد.
- اگر بخش عملی یا تئوری تمرین را قبل از مهلت ارسال امتیازی آپلود کنید، ۲۰ درصد نمره اضافی به آن بخش تعلق خواهد گرفت و پس از آن، ویدئویی تحت عنوان راهنمایی برای حل تمرین منتشر خواهد شد.
- حتماً تمرین‌ها را بر اساس موارد ذکر شده در صورت سوالات حل کنید. در صورت وجود هرگونه ابهام، آن را در صفحه تمرین در سایت کوثر مطرح کنید و به پاسخ‌هایی که از سوی دستیار آموزشی مربوطه ارائه می‌شود، توجه کنید.
- در صورت هم‌فکری و یا استفاده از هر منابع خارج درسی، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال مورد نظر را ذکر کنید.
- فایل پاسخ‌های سوالات نظری را در قالب یک فایل pdf به فرمت `HW3_T_[STD_ID].pdf` آماده کنید و برای سوالات عملی، هریک را در یک فایل zip جداگانه قرار دهید و فایل zip اول را به فرمت `HW3_P1_[STD_ID].zip` و فایل zip دوم را به فرمت `HW3_P2_[STD_ID].zip` و فایل سوم را نیز به فرمت `HW3_P3_[STD_ID].zip` نامگذاری کرده و هرکدام را به صورت جداگانه آپلود کنید.
- گردآورندگان تمرین: کیارش جولایی، علیرضا صابونچی، امیرعلی لقمانی، آسمانه نافع، دنیا نوابی، فاطمه شیری

سوالات نظری (۱۰+۱۵ نمره)

۱. (۲۵ نمره) صحیح و غلط بودن هر مورد را مشخص کنید. در موارد (الف)، (و) و (ی) نیز با استدلال پاسخ دهید.

الف) می‌دانیم که 2^n تابع boolean متمایز بر روی n ورودی وجود دارند. در نتیجه برای ۲ ورودی، ۱۶ تابع boolean متمایز وجود دارد. چند تا از این ۱۶ تابع را می‌توان توسط یک perceptron نشان داد؟ چرا؟

ب) تابع فعال‌سازی tanh به دلیل صفرمحور بودن، برخلاف تابع sigmoid مشکل ناپدیدکردن گرادیان را ندارد.

ج) هر تابع منطقی را می‌توان با یک شبکه دولایه (یک لایه مخفی) و یک تابع فعال‌سازی ReLU نمایش داد.

د) افزایش عمق و عرض یک شبکه عصبی همیشه عملکرد آن را بهبود می‌بخشد.

ه) یک شبکه عصبی سطحی و بسیار پهن می‌تواند همیشه به اندازه یک شبکه عصبی عمیق هر تابعی را به صورت کارآمد تقریب بزند.

و) مشکل ناپدیدشدن گرادیان در آموزش شبکه‌های عصبی چیست و چگونه توابع فعال‌سازی مانند ReLU به کاهش آن کمک می‌کنند؟

ز) Stochastic Gradient Descent (SGD) با اندازه batch کوچک به دلیل فرکانس بالای بهروزرسانی تضمین می‌کند سریع‌تر از mini-batch gradient descent همگرا شود.

ی) با استفاده از Stochastic Gradient Descent با ممان اول، از نقطه اولیه $x_0 = -2/8$ ، با نرخ یادگیری $\gamma = 0/05$ ، ضریب ممان $\mu = 0/7$ و تعداد تکرار $i = 2$ کمینه کنید. مقادیر g_i, m_i, x_i و y_i و محاسبات هر مرحله باید ارائه شوند. (منظور از g_i گرادیان در مرحله i و مقدار ممان در مرحله i می‌باشد) نیازی به اصلاح بایاس نیست.

$$y = 0/3x^4 - 0/1x^3 - 2x^2 - 0/8x$$

۲. (۱۵ نمره) تابع $y = f(x_1, x_2, x_3)$ به صورت زیر تعریف شده است:

$$y = 6z_1 + 2z_2 + 4$$

$$z_1 = \exp\left(-\frac{\max\{|x_1 - 3|, |x_2 - 5|, |x_3 + 2|\}^2}{4}\right)$$

$$z_2 = \begin{cases} 0, & \text{if } \sqrt{(x_1 + 3)^2 + (x_3 - 4)^2} > 4 \\ 1, & \text{otherwise} \end{cases}$$

الف) شبکه عصبی تولیدکننده این تابع را با در نظر گرفتن مدل نورون عمومی رسم کنید. برای هر نورون، تابع ورودی شبکه، تابع فعال‌سازی و تابع خروجی را برحسب وزن‌های ورودی و مقادیر پارامتری مشخص کنید. وزن‌ها و مقادیر آستانه را تعیین کنید. (نکته: جواب این سؤال یکتا نیست)

ب) با یک شبکه عصبی متشکل از نورون‌های عمومی، تابع زیر را بسازید (فرض کنید f_{net} فقط می‌تواند جمع وزن‌دار ورودی‌ها باشد و f_{out} هم فقط می‌تواند تابع همانی باشد).

$$f(x_1, x_2, x_3) = \max\{x_3, x_1^{x_2}\}$$

۳. (۱۵ نمره) فرض کنید که یک شبکه عصبی دولایه داریم که به صورت زیر تعریف شده است:

$$z_1 = W_1 x^{(i)} + b_1$$

$$a_1 = \text{ReLU}(z_1)$$

$$z_2 = W_2 a_1 + b_2$$

$$\hat{y}^{(i)} = \sigma(z_2)$$

$$L^{(i)} = y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log (1 - \hat{y}^{(i)})$$

$$J = -\frac{1}{m} \sum_{i=1}^m L^{(i)}$$

deep
learning
standford

توجه کنید که $x^{(i)}$ یک نمونه ورودی منفرد است و شکل آن $1 \times D_x$ است. همچنین $y^{(i)}$ یک برچسب خروجی است و یک اسکالر محسوب می‌شود. در دیتاست ما m نمونه وجود دارد. ما از تعداد D_{a1} نود در لایه مخفی استفاده می‌کنیم، بنابراین شکل z_1 برابر $1 \times D_{a1}$ است.

۱. ابعاد W_1, b_1, W_2 و b_2 چیست؟ اگر بخواهیم شبکه را روی چندین نمونه vectorize کنیم، شکل وزن‌ها و بایاس‌ها چه تغییری خواهد کرد؟ اگر بخواهیم شبکه را روی چندین نمونه ورودی vectorize کنیم، شکل‌های X و Y چگونه خواهد بود؟ **خود جواب استندفورد**

۲. مقدار $\frac{\partial J}{\partial \hat{y}^{(i)}}$ چیست؟ این مقدار را $\delta^{(i)}$ نام‌گذاری کنید. با استفاده از این نتیجه، مقدار $\frac{\partial J}{\partial \hat{y}}$ چیست؟

۳. مقدار $\frac{\partial \hat{y}^{(i)}}{\partial z_2}$ چیست؟ این مقدار را $\delta_2^{(i)}$ نام‌گذاری کنید.

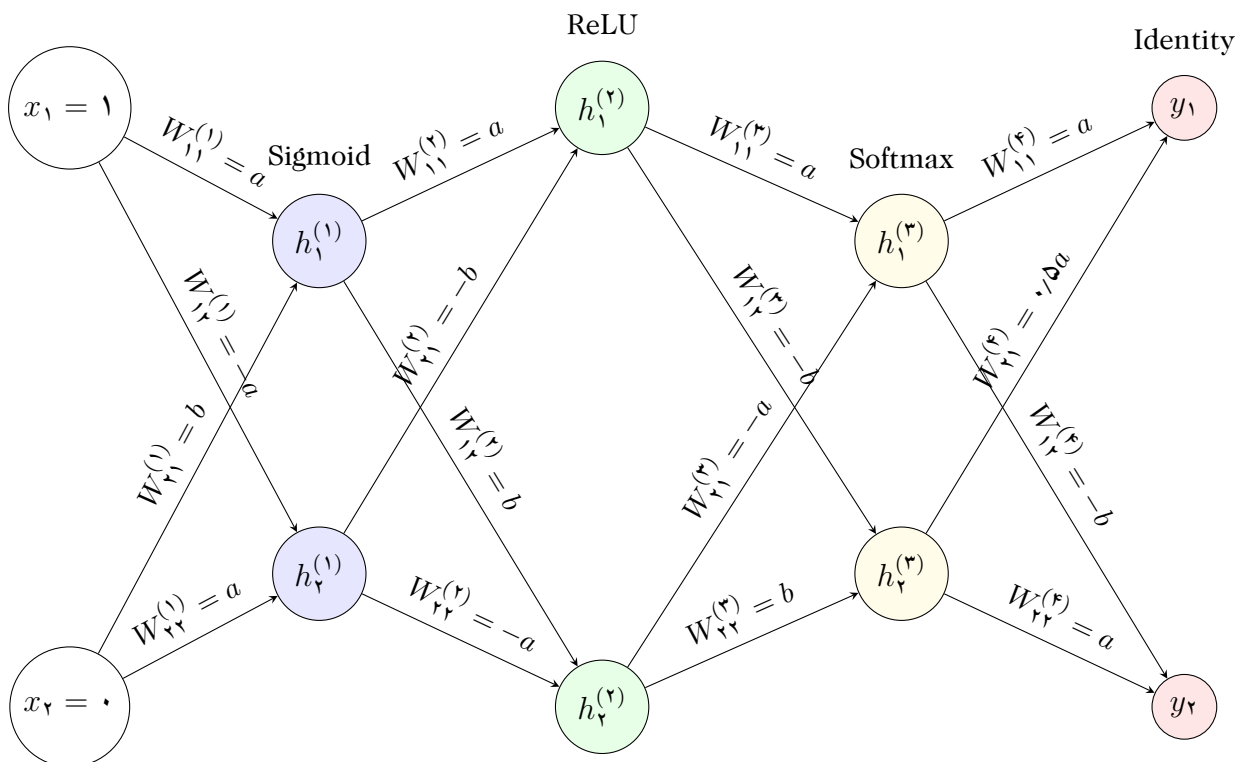
۴. مقدار $\frac{\partial z_2}{\partial a_1}$ چیست؟ این مقدار را $\delta_3^{(i)}$ بنامید.

۵. مقدار $\frac{\partial a_1}{\partial z_1}$ چیست؟ این مقدار را $\delta_4^{(i)}$ بنامید.

۶. مقدار $\frac{\partial z_1}{\partial W_1}$ چیست؟ این مقدار را $\delta_5^{(i)}$ بنامید. $x^{(i)T}$

۷. مقدار $\frac{\partial J}{\partial W_1}$ چیست؟ می‌توانید از نتایج قبلی استفاده کنید.

۴. (۳۰ نمره) یک شبکه عصبی feedforward با معماری زیر را در نظر بگیرید:



- لایه ورودی: ۲ نورون
- لایه مخفی ۱: ۲ نورون، با استفاده از تابع فعال‌سازی Sigmoid
- لایه مخفی ۲: ۲ نورون، با استفاده از تابع فعال‌سازی ReLU
- لایه مخفی ۳: ۲ نورون، با استفاده از تابع فعال‌سازی Softmax
- لایه خروجی: ۲ نورون، با استفاده از تابع فعال‌سازی Identity

وزن‌ها و بایاس‌های اولیه شبکه به صورت زیر است:
وزن‌ها و بایاس‌ها از لایه ورودی به لایه مخفی ۱ (لایه ۱):

$$W^{(1)} = \begin{bmatrix} a & -a \\ b & a \end{bmatrix}, \quad b^{(1)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

وزن‌ها و بایاس‌ها از لایه مخفی ۱ به لایه مخفی ۲ (لایه ۲):

$$W^{(2)} = \begin{bmatrix} a & b \\ -b & -a \end{bmatrix}, \quad b^{(2)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

وزن‌ها و بایاس‌ها از لایه مخفی ۲ به لایه مخفی ۳ (لایه ۳):

$$W^{(3)} = \begin{bmatrix} a & -b \\ -a & b \end{bmatrix}, \quad b^{(3)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

وزن‌ها و بایاس‌ها از لایه مخفی ۳ به لایه خروجی (لایه ۴):

$$W^{(4)} = \begin{bmatrix} a & -b \\ 0.5a & a \end{bmatrix}, \quad b^{(4)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

بردار ورودی x و خروجی هدف t به صورت زیر است:

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad t = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

با استفاده از نرخ یادگیری (η) ، یک بار فرآیند **backpropagation** را انجام دهید تا وزن‌ها و بایاس‌های شبکه به‌روزرسانی شوند. از توابع فعال‌سازی زیر در هر لایه استفاده کنید:

- لایه ۱ (Sigmoid): $\sigma(x) = \frac{1}{1 + e^{-x}}$
- لایه ۲ (ReLU): $\text{ReLU}(x) = \max(0, x)$
- لایه ۳ (Softmax): $\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$
- لایه ۴ (Identity): $f(x) = x$

فرض کنید تابع خطا به صورت میانگین مربعات (MSE) تعریف شده است:

$$E = \frac{1}{2} \sum_{k=1}^2 (a_k^{(4)} - t_k)^2$$

که در آن $a^{(4)}$ خروجی شبکه است.

سوال:

- الف) یک forward pass از شبکه انجام دهید تا خروجی $a^{(4)}$ را محاسبه کنید.
ب) خطای E را با استفاده از تابع خطای MSE محاسبه کنید.

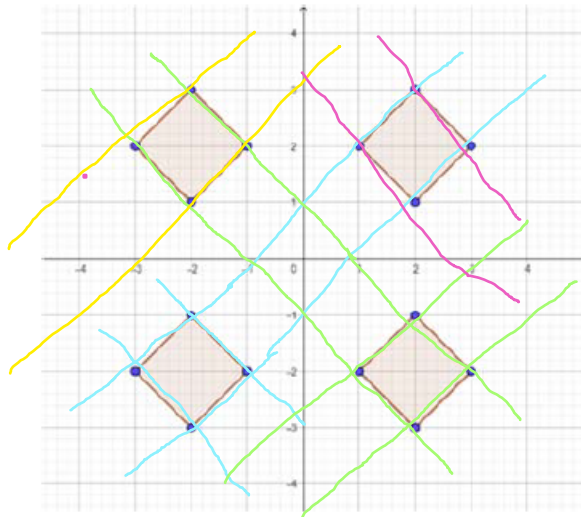
ج یک backward pass انجام دهید تا گرادیان‌های خطا نسبت به تمام وزن‌ها و بایاس‌ها را محاسبه کنید.

د وزن‌ها و بایاس‌ها را با استفاده از gradient descent و نرخ یادگیری η به‌روزرسانی کنید.

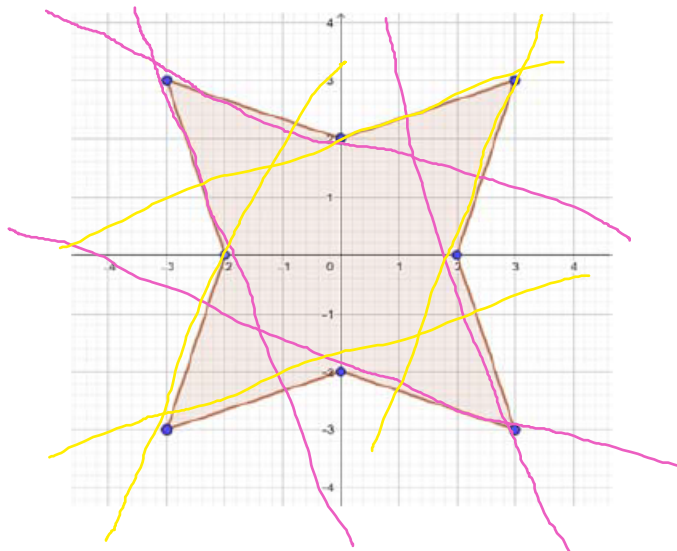
۵. (۱۵ نمره)

Threshold Logic Unit

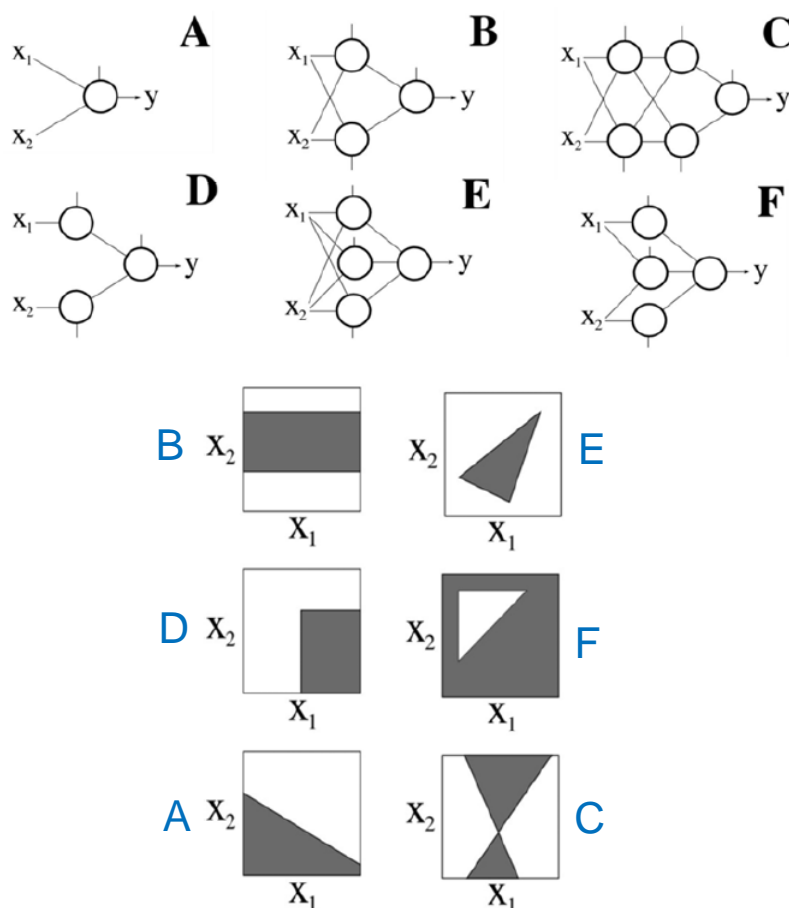
الف) یک شبکه با حداقل تعداد TLU به گونه‌ای طراحی کنید که برای ورودی‌های داخل ناحیه‌های شکل زیر خروجی یک بدهد و برای خارج آن صفر. ساختار شبکه به همراه وزن‌ها و بایاس‌های آن را مشخص کنید.



ب) آیا می‌توان با استفاده از شبکه طراحی شده در قسمت (الف) و صرفاً با تغییر پارامترهای آن، طبقه‌بندی برای شکل زیر بدست آورد؟ توضیح دهید.



ج) شبکه‌های TLU زیر را در نظر بگیرید. ابتدا مشخص کنید هر کدام از پاسخ‌ها توسط کدام شبکه تولید شده است و سپس علت را توضیح دهید. فرض کنید که هر کدام از شبکه‌ها فقط یک پاسخ را ایجاد کرده است.



۶. (امتیازی-۱۵ نمره) شبکه عصبی با معماری A را در نظر بگیرید. این شبکه معادل با تابع $F_A(x; W)$ است که در آن x ورودی و W تمام پارامترهای شبکه می‌باشد. فرض کنید تابع activation این شبکه ReLU باشد. آنگاه برای شبکه مفهومی به نام activation pattern تعریف می‌کنیم. activation pattern یک رشته از ۰ و ۱ به طول تعداد نورون‌های شبکه است. ۰ بودن به معنای غیرفعال بودن نورون و ۱ بودن به معنای فعال بودن است که در تابع ReLU معادل صفر یا غیرصفر بودن مقدار هر نورون است. تعداد این pattern ها را با $\mathcal{A}(F_A(x; W))$ نشان می‌دهیم.

الف) شبکه عصبی $F_A(\mathbb{R}^m; W)$ را با ReLU به عنوان تابع activation در نظر بگیرید. نشان دهید فضای ورودی با استفاده از این شبکه به polytope (اشکال هندسی با سطوح صاف) های convex تقسیم می‌شود که $F_A(\mathbb{R}^m; W)$ متناظر با یک تابع خطی در نواحی مختلف این فضا می‌باشد.

ب) معماری $A_{(n,k)}$ را در نظر بگیرید که معادل یک شبکه عصبی fully connected با n لایه با عرض k می‌باشد. ثابت کنید $\mathcal{A}(F_{A_{(n,k)}}(\mathbb{R}^m; W))$ دارای حد بالای $O(k^{mn})$ با در نظر گرفتن تابع ReLU به عنوان activation می‌باشد.

راهنمایی: فرض کنید k ابرصفحه در فضای \mathbb{R}^m وجود دارند که هر کدام متناظر با یک معادله به فرم $a_i^T x = b_i$ هستند. اگر تعداد نواحی تشکیل شده در فضا $r(k, m)$ باشد، خواهیم داشت:

$$r(k, m) \leq \sum_{i=0}^m \binom{k}{i}$$