



۱. (۱۵ نمره)

(آ) می‌دانیم که یک رابطه منطقی را می‌توان به صورت جمع p_i هایی نوشت که هر یک به صورت ضرب a_j ها و not آن‌ها نوشته می‌شوند. نورون‌های ورودی را a_j ها تعریف می‌کنیم که در صورتی که خودشان به کار رفته باشند، با وزن یک و در صورتی که not شان به کار رفته باشد با وزن ۱ - مشخص می‌کنیم و اگر آن ورودی در عبارت ضربی اصلاً به کار نرفته باشد، وزن آن را صفر قرار می‌دهیم. بایاس هر جمله هم به اندازه یکی کمتر از تعداد نورون‌هایی است که در ساخت جمله ضربی کاربرد دارند البته به صورت منفی. در لایه اول ضرب‌ها یعنی p_i ها را می‌سازیم. در لایه دوم هم از هر p_i با وزن یک به نورون خروجی متصل می‌کنیم و بایاس نمی‌گذاریم تا در صورتی که حداقل یکی از p_i ها یک شد، مقدار خروجی مثبت شود.

(ب) حاصل XOR وقتی یک است که تعداد فردی از متغیرهای ما یک باشند و آن را می‌توان به صورت SOP نوشت بنابراین طبق الف ساخته می‌شود. تعداد جملاتی که در ساخت XOR دخالت دارند، معادل خواهد بود با تعداد زیر مجموعه‌های فرد عضوی، یعنی دو به دو به توان $n - 1$ عبارت ضربی در لایه اول داریم که همگی به خروجی متصل هستند. پس در مجموع تعداد پارامترهای شبکه از اردر دو به توان n خواهد بود

(ج) برای این قسمت، جملات را دو به دو با هم XOR می‌کنیم، و نتیجه‌ها رو دو به دو با هم و به همین منوال ادامه می‌دهیم. به ازای هر مرحله از XOR هم که به دو لایه نیاز داریم، پس نهایتاً لایه های شبکه از اردر $2 * \log(n)$ خواهد شد. اما تعداد پارامترها:

وقتی دو متغیر اول را با هم XOR می‌کنیم، برای ساخت عبارات ضربی به ۶ پارامتر (با احتساب بایاس) و برای جمع زدن آن‌ها به ۳ پارامتر نیاز داریم. و چون متغیرها را دو به دو با هم به این شکل جمع می‌کنیم، در کل $n/2$ تا قسمت داریم که در لایه اول و دوم هر یک ۹ پارامتر دارند. پس در لایه اول و دوم $9n/2$ ، در لایه سوم و چهارم $9n/4$ و به همین ترتیب تعداد پارامترها مشخص می‌شود که جمع این‌ها از اردر n است.

۲. (۲۰ نمره) پیش از هرچیز به این نکته دقت می‌کنیم که برای انتخاب x_1 به عنوان ماکسیموم و یک شدن y_1 باید هر دو شرط زیر برقرار باشند:

$$\begin{aligned}x_1 &\geq x_2 \\x_1 &\geq x_3\end{aligned}$$

پس اگر بتوانیم g_1 و g_2 را به گونه‌ای تشکیل دهیم که هر کدام، یکی از دو شرط بالا را ایجاد کنند، (یعنی با برقرار شدن شرط مقدار ۱ و در غیر این صورت مقدار ۰ برگردانند) کافی است که بر روی g_1 و g_2 تابع and پیاده سازی کنیم.

برای این کار کافی است که به g_1 و g_2 وزن یک بدهیم، و از بایاس ۱.۵ - استفاده کنیم. به این ترتیب، تنها در شرایطی که هر دوی آن‌ها یک شده باشند، این خروجی یک می‌شود.

حال باید سعی کنیم g_1 و g_2 را به گونه‌ای بسازیم که شرط مورد نظر را ایجاد کنند. نحوه ایجاد g_1 را بررسی می‌کنیم، مشابه آن برای g_2 هم برقرار است. فرض کنید که h_1 و h_2 را به این شکل تعریف کنیم:

$$h_1 = \text{ReLU}(x_1 - x_2)$$

$$h_2 = \text{ReLU}(x_1 - x_2 + 1)$$

و g_1 را نیز به صورت $\text{ReLU}(h_2 - h_1)$ تعریف می‌کنیم.

با توجه به آن که می‌دانیم ورودی‌های ما اعداد صحیح هستند، می‌توانیم بگوییم که در صورتی که $x_1 \geq x_2$ باشد اختلاف این دو ورودی یک عدد مثبت است پس بعد از عبور از تابع فعالسازی، مقدارش خودش باقی می‌ماند و خروجی h_1 و h_2 هر دو مقادیر مثبتی هستند. پس با کم کردن این دو از هم، حاصل یک می‌شود و با عبور از تابع فعالسازی یک باقی می‌ماند.

همچنین در صورتی که $x_2 > x_1$ باشد، خروجی h_1 و h_2 صفر می‌شود. این کار را برای تشکیل بقیه خروجی‌ها هم تکرار می‌کنیم.