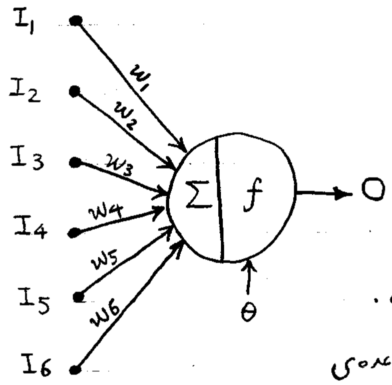
هوش مصنوعی پیشرفته

تکلیف شماره‌ی ۵ — شبکه‌های عصبی مصنوعی (۱)



۱) الف) پرسپترون

۱۴ مثال ورودی - خروجی داریم.

هرکدام را یک بار به شبکه نشان می‌دهیم

(به ترتیب جدول ستون به ستون از چپ به راست)

هر ورودی یک بردار ستونی 6×1 و خروجی اسکالر 1×1 است.

با هر بار نشان دادن یک ورودی (مثال) به شبکه، وزن‌ها با قاعده‌ی یادگیری پرسپترون هنگام می‌شوند:

$$w_i \leftarrow w_i + \alpha I_i e \qquad i = 1, 2, \dots, 6$$

نرخ یادگیری α را $\alpha = 0.1$ انتخاب می‌کنیم.

خطای e برای هر مثال برابر با تفاضل خروجی مطلوب (هدف): T و خروجی واقعی شبکه است O:

$$e = T - O$$

تابع فعال‌سازی f هم یک hardlim با آستانه‌ی θ است. مقدار θ هم مشابه سایر وزن‌ها هنگام می‌شود:

$$\theta \leftarrow \theta + \alpha.(-1).e = \theta - \alpha e$$

از مقادیر وزن و آستانه‌ی صفر شروع می‌کنیم:

با اجرای الگوریتم فوق برای 10 ایپک، به میزان خطای صفر می‌رسیم و وزن‌های نهایی عبارتند از

$w_1 = 0.4 \quad w_2 = 0 \quad w_3 = 0.2 \quad w_4 = -0.1 \quad w_5 = -0.2 \quad w_6 = 0.2$

$\theta = 0.2$

کد MATLAB پیوست را ملاحظه کنید perceptron_test_assign4.m

ب) درخت تصمیم :

برای ایجاد درخت تصمیم از الگوریتم DECISION_TREE_LEARNING استفاده می‌کنیم.

برای این کار ابتدا باید ریشه‌ی درخت براساس تعیین مهم‌ترین خصیصه انتخاب شود. این انتخاب به کمک مفهوم آنتروپی و تعیین میزان بهره‌ی اطلاعاتی یک خصیصه صورت می‌گیرد.

آنتروپی یک متغیر تصادفی بولی که با احتمال q برابر با true است، عبارت است از:

$$B(q) = -(q \log_2 q + (1-q)\log_2(1-q))$$

وقتی مجموعه‌ی داده‌ی ما دارای p مثال مثبت $(T = 1)$ و n مثال منفی $(T = 0)$ است، آنتروپی خصیصه‌ی هدف بر روی کل مجموعه می‌شود:

$$H(Goal) = B\left(\frac{p}{p+n}\right)$$

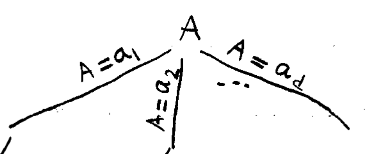
وقتی خصیصه‌ی A با d مقدار متمایز مجموعه‌ی آموزشی E را به زیرمجموعه‌های E_1، E_2، ...، E_d تقسیم کند و هر زیرمجموعه‌ی E_k دارای p_k مثال مثبت و n_k مثال منفی باشد، در این صورت اگر در سمت شاخه‌ی E_k جلو برویم به $B\left(\frac{p_k}{p_k+n_k}\right)$ بیت اضافی برای پاسخ به پرسش نیاز داریم.

یک مثال انتخاب شده به صورت تصادفی از مجموعه داده‌ی آموزشی دارای kامین مقدار آن خصیصه است با احتمال

$$\frac{p_k + n_k}{p+n}$$

پس آنتروپی متوسط باقیمانده پس از آزمایش خصیصه‌ی A می‌شود:

$$Remainder(A) = \sum_{k=1}^{d} \frac{p_k + n_k}{p+n} B\left(\frac{p_k}{p_k+n_k}\right)$$



بهره‌ی اطلاعاتی که میزان اهمیت یک خصیصه را نشان می‌دهد، کاهش مورد انتظار در آنتروپی پس از آزمایش A است:

$$IMPORTANCE(A) = Gain(A) = B\left(\frac{p}{p+n}\right) - Remainder(A)$$

پس خصیصه‌ای به عنوان ریشه انتخاب می‌شود که Gain آن بیشتر باشد.

در این تمرین یکی از خصیصه‌های I_1 تا I_6 باید انتخاب شود تا در ریشه‌ی درخت تصمیم قرار گیرد.

$$H(T) = B\left(\frac{7}{7+7}\right) = B(0.5) = 1$$

بر اساس مقدار I_1 (0 یا 1) دو گروه داریم: گروه ۱) $p_1 = 6 , n_1 = 1$ ؛ گروه ۲) $p_2 = 1 , n_2 = 6$

$$Gain(I_1) = information\_gain(\overset{p_1}{[6} \ \overset{p_2}{1]}, \overset{n_1}{[1} \ \overset{n_2}{6]}) = 0.408$$

$$Gain(I_2) = information\_gain([3 \ \ 4], [4 \ \ 3]) = 0.015$$

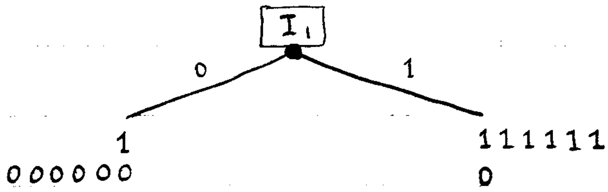
$$Gain(I_3) = information\_gain([5 \ \ 2], [3 \ \ 4]) = 0.061$$

$$Gain(I_4) = information\_gain([3 \ \ 4], [4 \ \ 3]) = 0.015$$

$$Gain(I_5) = information\_gain([3 \ \ 4], [4 \ \ 3]) = 0.015$$

$$Gain(I_6) = information\_gain([3 \ \ 4], [4 \ \ 3]) = 0.015$$

مشخصاً I_1 دارای بالاترین بهره است و به عنوان ریشه انتخاب می‌شود.



حال هر یک از زیر درخت‌ها با مجموعه مثال‌های خودشان به عنوان یک درخت تصمیم مستقل یاد گرفته می‌شوند. برای هر زیر درخت، فقط خصیصه‌های در نظر گرفته می‌شوند که در مسیر ریشه‌ی درخت اصلی تا آن زیر درخت بررسی نشده‌اند.

مثلاً برای زیر درخت سمت چپ I_1 داریم:

$$Gain(I_2) = information\_gain([1 \ \ 0], [4 \ \ 2]) = 0.076$$

$Gain(I_3) = 0.128$ ✓

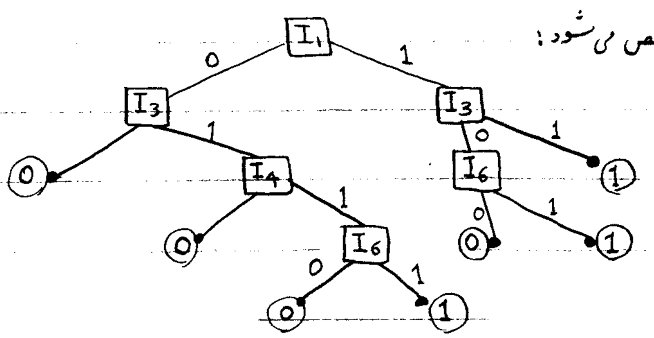
$Gain(I_4) = 0.076$

$Gain(I_5) = 0.128$ ✓

$Gain(I_6) = 0.076$

بین I_3 و I_5 یکی به دلخواه می‌تواند به عنوان ریشه‌ی سمت چپ زیر درخت I_1 انتخاب شود.

و به همین ترتیب سایر گره‌های درخت تصمیم مشخص می‌شود:



* کدهای MATLAB ضمیمه را ملاحظه کنید.

۲) برای سادگی فرض می‌کنیم تابع فعال‌سازی برای همه‌ی گره‌ها مشابه باشد:

$$g(x) = cx + d$$

بحث انجام شده برای c_i و d_i مختلف، مشابه است اما کمی شلوغ‌تر می‌شود!

الف) خروجی لایه‌ی پنهان می‌شود:

$$H_j = g\left(\sum_k W_{k,j} I_k\right) = c \sum_k W_{k,j} I_k + d$$

خروجی نهایی می‌شود:

$$O_i = g\left(\sum_j W_{j,i} H_j\right) = c\left(\sum_j W_{j,i}\left(c \sum_k W_{k,j} I_k + d\right)\right) + d$$

$$= c^2 \sum_k I_k \sum_j W_{k,j} W_{j,i} + d\left(1 + c \sum_j W_{j,i}\right)$$

ملاحظه می‌شود که خروجی O_i نسبت به ورودی‌های I_k خطی است.

پس می‌توانیم این تابع را در قالب یک پرسپترون بدون لایه‌ی میانی هم پیاده‌سازی کنیم که وزن‌های آن

$$W_{k,j} = \sum_j W_{k,j} W_{j,i}$$

است و تابع فعال‌سازی آن به صورت خطی

$$g(x) = \underbrace{c^2}_{\alpha} x + \underbrace{d\left(1 + c \sum_j W_{j,i}\right)}_{\beta} = \alpha x + \beta$$

می‌باشد.

ب) کاهش فوق می‌تواند به طور مستقیم برای کاهش یک شبکه‌ی n لایه به یک شبکه‌ی $n-1$ لایه استفاده شود. با استقراء شبکه n لایه می‌توان به یک شبکه‌ی تک‌لایه تبدیل شود.

پس نتیجه می‌گیریم توابع فعال‌سازی خطی، شبکه‌های عصبی چندلایه را محدود می‌کند تا فقط توابع خطی را بازنمایی کند.

۳) به طور شهودی این داده‌ها بیان می‌کند که احتمال 1 بودن خروجی 80% است (پیش‌بینی 0.8 = (P(Output = 1)
شبکه وزن‌هایش را به گونه‌ای تنظیم کرده که خطا به حداقل برسد. تابع خطا عبارت است از :

$$E = \frac{1}{2} \sum_i (\underbrace{y_i}_{\text{target}} - \underbrace{a_i}_{\text{actual output}})^2 = \frac{1}{2}\left[80(1-a_1)^2 + 20(0-a_1)^2 \right]$$

$$= 40(1-a_1)^2 + 10(0-a_1)^2$$

$$= 40(a_1^2 - 2a_1 + 1) + 10a_1^2$$

$$= 50a_1^2 - 80a_1 + 40$$

$$\frac{dE}{da_1} = 100a_1 - 80 = 0 \implies a_1 = 0.8$$

پس خروجی شبکه احتمال 1 بودن خروجی است.

۴) تابع فعال‌سازی خطی $g(n)=n$:

لایه‌ی پنهان :

$$a_1 = w_1 p + b_1$$

$$a_2 = w_2 p + b_2$$

لایه‌ی خروجی :

$$a = w_3 a_1 + w_4 a_2$$

هدف، می‌نیمم‌سازی تابع خطا است :

روی همه‌ی N نمونه مثال :

$$E = \frac{1}{2} \sum_{i=1}^{N} (y(i) - a(i))^2$$

و با استفاده از روش تندترین شیب، روابط زیر را داریم :

$$w_1 \leftarrow w_1 - \alpha \frac{\partial E}{\partial w_1}$$

$$w_2 \leftarrow w_2 - \alpha \frac{\partial E}{\partial w_2}$$

$$w_3 \leftarrow w_3 - \alpha \frac{\partial E}{\partial w_3}$$

$$w_4 \leftarrow w_4 - \alpha \frac{\partial E}{\partial w_4}$$

$$b_1 \leftarrow b_1 - \alpha \frac{\partial E}{b_1}$$

$$b_2 \leftarrow b_2 - \alpha \frac{\partial E}{b_2}$$

کافی است مشتقات فوق را محاسبه کنیم. از لایه‌ی آخر شروع می‌کنیم و از قاعده‌ی زنجیره‌ای مشتق کمک می‌گیریم :

$$\frac{\partial E}{\partial w_3} = \frac{\partial}{\partial w_3}\left(\frac{1}{2} \sum_{i=1}^{N} \left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)^2\right)$$

$$= -\sum_{i=1}^{N} a_1(i)\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)$$

مشابه با w_3 :

$$\frac{\partial E}{\partial w_4} = -\sum_{i=1}^{N} a_2(i)\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)$$

برای لایه‌ی میانی

$$\frac{\partial E}{\partial w_1} = \frac{\partial E}{\partial a_1}\frac{\partial a_1}{\partial w_1} = -\sum_{i=1}^{N} w_3\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)p$$

$$\begin{cases} \dfrac{\partial E}{\partial a_1} = -\sum_{i=1}^{N} w_3\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right) \\ \dfrac{\partial a_1}{\partial w_1} = \dfrac{\partial}{\partial w_1}(w_1 p + b) = p \end{cases}$$

و به طور مشابه :

$$\frac{\partial E}{\partial w_2} = \frac{\partial E}{\partial a_2}\frac{\partial a_2}{\partial w_2} = -\sum_{i=1}^{N} w_4\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)p$$

و

$$\frac{\partial E}{\partial b_1} = \frac{\partial E}{\partial a_1}\cancelto{1}{\frac{\partial a_1}{\partial b_1}} = -\sum_{i=1}^{N} w_3\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)$$

$$\frac{\partial E}{\partial b_2} = \frac{\partial E}{\partial a_2}\cancelto{1}{\frac{\partial a_2}{\partial b_2}} = -\sum_{i=1}^{N} w_4\left(y(i) - (w_3 a_1(i) + w_4 a_2(i))\right)$$