

به نام خدا



دانشکده مهندسی کامپیوتر

تمرینات درس یادگیری عمیق

پاسخنامه تمرین سری پنجم

دکتر محمدرضا محمدی

دانشجو: مجتبی نافذ

96431335

نیمسال دوم ۹۹-۱۰۰

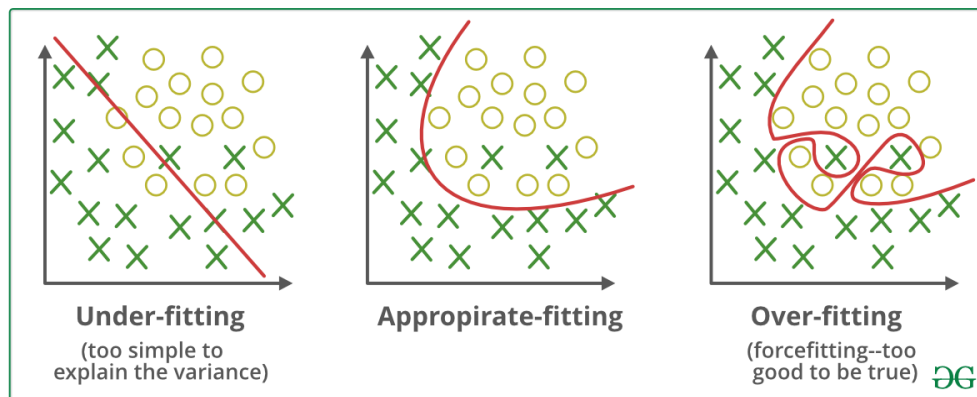
۱. به طور خلاصه مفاهیم **overfitting** و **underfitting** را توضیح دهید و راه حل های رفع این مشکلات را توضیح دهید. (می توانید از کتاب یا اسلایدها و یا این [لینک](#) استفاده کنید)

توضیح **underfitting** : زمانی که مدل شبکه نمی تواند یک راه حل خوب برای داده ها پیدا کند و در واقع توانایی یادگیری ویژگی های مناسبی از داده های ورودی را ندارد و پترن های داده های **train** را خوب ارزیابی نمی کند و به دقت مناسبی نمی رسد شبکه به اصطلاح **underfit** کرده است. (البته در ابتدای کار که هنوز شبکه خوب **train** نشده هم شبکه در حالت **underfit** می باشد)

معمولا زمان هایی رخ می دهد که یا داده های آموزشی ما کم است. یا از شبکه خطی برای داده های غیر خطی استفاده کرده ایم

روش جلوگیری: ۱- افزایش **complexity** مدل ۲- افزایش تعداد ویژگی ها با انجام **feature engineering** (داده آموزشی رو بهتر کنیم). ۳- حذف نویز از داده (مثلا در داده های خطی چند داده ی پراکنده را حذف کنیم) ۴- تعداد **epoch** ها را افزایش دهیم. توضیح **overfitting** : اگر مدل ما با داده های آموزشی خیلی بیش از حد **train** شود (یا تعداد داده ی آموزشی زیاد است یا ما تعداد **epoch** را زیاد گرفتیم) و ما نویز ها و پترن های غیر مهم داده ی آموزشی را به عنوان ویژگی و پترن یاد بگیریم در واقع روی داده ی آموزشی **overfit** کرده ایم و دقت ما روی داده های آموزشی خیلی خوب ولی روی داده های تست که تاکنون ندیده ایم افت خواهد کرد.

روش جلوگیری: ۱- افزایش داده های آموزشی ۲- کاهش **complexity** مدل (مثل کم کردن لایه ها و کم کردن تعداد نرون های لایه ها) ۳- توقف زود هنگام (زمانی که در طول فاز آموزش دقت روی داده های تست ما کاهشی بود ادامه ندهیم و آموزش را متوقف کنیم). ۴- استفاده از **regularization** (سعی دارد با اضافه کردن اطلاعات اضافی وزن سایر پارامترها را کاهش بدهد). ۵- استفاده از **dropout** برای شبکه



۲. در [نوتبوک](#) تهیه شده مدل ساده‌ای را بر روی یک مجموعه داده آموزش داده‌ایم ولی به دلیل سادگی مدل **underfit** شده است. ابتدا با تعریف مدل‌های بهتر این مشکل را حل کنید. سپس، برای جلوگیری از **overfitting** از **regularization** برای وزن‌ها و همچنین از **Dropout** استفاده کنید و در نهایت بهترین مدل برای آموزش این مجموعه داده را استفاده کنید. نتایجی که به دست می‌آورید را مقایسه کنید و توضیح دهید. (در مورد لایه‌ها فقط می‌توانید از لایه‌های **Dense** و **Dropout** استفاده کنید. نتایج تست چند مدلی را که می‌خواهید در مقایسه خود بیاورید در نوتبوک نگه دارید و حذف نکنید.)

برای این سوال تقریباً همه‌ی موارد را درون **notebook** ضمیمه شده گذاشته‌ام و با نمودار کامل قابل مقایسه است .

برای حل **underfitting** کمی تعداد لایه‌ها و نوروں‌های لایه‌ها را بیشتر کرده‌ام.

که در این حالت دچار **overfit** شده و روی داده‌های آموزشی بالا ۹۹ درصد دقت به دست آمد اما برای داده‌های تست دقت کمتر از ۶۵ درصد میباشد.

مدل را انواع **L1, L2 regularization** دوباره **train** کردم , تا حدی از **overfitting** روی **train** جلوگیری شده دقت روی داده

های آموزشی حدود ۷۰ درصد شد و روی تست تقریباً حدود ۶۸ تا ۶۹ درصد

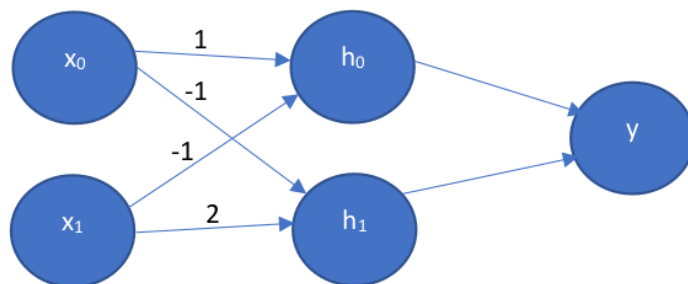
اما **dropout** توانست مدل را تا ۷۰ درصد هم رو داده‌های تست به دقت برساند

مدل‌ها ترکیبی از **dropout , regularization** هم تا حد ۷۰ من دقت دادند.

در کل مدل‌ها بر پایه‌ی **dropout** دقتی بالاتر از بقیه به من دادند که در **notebook** ضمیمه آن را هم اجرا کرده‌ام.

۳. مطابق شکل، شبکه‌ای با دو ورودی (ویژگی) و یک لایه مخفی با دو نورون و یک خروجی داریم (با تابع فعالسازی ReLU). اگر برای آموزش این شبکه دو داده آموزشی به صورت جدول زیر در اختیار داشته باشیم و از تابع ضرر MSE و از بهینه‌ساز GD استفاده شود، پارامترهای شبکه را در دو تکرار با فرض استفاده از $\text{Dropout} = 0.5$ (در تکرار اول نورون h_0 و در تکرار دوم نورون h_1 را حذف کنید) محاسبه کنید. (مقدار اولیه بایاس را صفر در نظر بگیرید و مقدار اولیه وزن‌ها بر روی شکل نشان داده شده است)

الف) از رگولاریزیشن نرم ۲ استفاده کنید. (فرمول بهینه‌سازی در اسلایدها)
 ب) از رگولاریزیشن نرم ۱ استفاده کنید. (فرمول بهینه‌سازی در اسلایدها)



| n | X_n | y_n |
|---|--------|-------|
| 1 | 5, 4 | 10 |
| 2 | 15, 12 | 20 |

(1)

فرض $\epsilon = 0.01$ و $\alpha = 0.01$

سوال سه

الف / ب / ج

اول / دوم

$b_1 = 0$

50% 2

$$\text{فرض} = w_0 x + b_0 \leq \epsilon x + 6(-1) = 3$$

باز

$$\frac{\partial L}{\partial w_0} = -2(10-3)(-1) = 14$$

$$\frac{\partial L}{\partial w_1} = -2(10-3)2 = -14$$

$$\frac{\partial L}{\partial b_1} = -2(10-3) = -14$$

$$Loss = (y - \text{Relu}(w_0 x + b_0))^2$$

$$\frac{\partial L}{\partial w_1} = \begin{cases} -2(y - \text{Relu}(w_0 x + b_0))w_1 & : w_0 x + b_0 > 0 \\ 0 & : w_0 x + b_0 \leq 0 \end{cases}$$

$$\frac{\partial L}{\partial b_1} = \begin{cases} -2(y - \text{Relu}(w_0 x + b_0)) & : w_0 x + b_0 > 0 \\ 0 & : w_0 x + b_0 \leq 0 \end{cases}$$

$$\text{فرض: } w \leftarrow (1 - \epsilon \alpha)w - \epsilon \frac{\partial L}{\partial w}(\theta; x, y)$$

$$w_0 = 0.999 \times (-1) - 0.01 \times 14 = -1.01499$$

$$w_1 = 0.999 \times 2 + 0.01 \times 14 = 2.01499$$

$$b_1 = 0 - 0.01 \times 14 = -0.14$$

50% 2

$$\text{فرض} = 10(-1.01499) + 2(2.01499) + (-0.14) = 10.1499$$

$$\frac{\partial L}{\partial w_0} = +2(10 - 10.1499)(-1.01499) = 2.1111$$

$$\frac{\partial L}{\partial w_1} = -2(10 - 10.1499)(2.01499) = -4.2222$$

$$\frac{\partial L}{\partial b_1} = -2(10 - 10.1499) = -0.298$$

$$w_0 = 0.999 \times (-1.01499) - 0.01 \times 2.1111 = -1.0171$$

$$w_1 = 0.999 \times (2.01499) + 0.01 \times 4.2222 = 2.0192$$

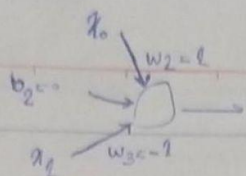
$$b_1 = 0.999 \times (-0.14) + 0.01 \times 0.298 = -0.1394$$

(۲)

Subject:

Date:

1/11



$$\begin{bmatrix} 0.999 \\ 1 \end{bmatrix}$$

$$y = w \cdot x + b = 0 - 1 + 0 = -1$$

$$\frac{\partial L}{\partial w_2} = -1(1 - 1) = -1$$

$$w_2 = 0.999(1) - 0.01(-1) = 1.009$$

$$\frac{\partial L}{\partial w_3} = -1(1 - 1) = 0$$

$$w_3 = 0.999(-1) - 0.01(1) = -1.009$$

$$\frac{\partial L}{\partial b} = -1(1 - 1) = -1$$

$$b = 0.999 \times 0 - 0.01(-1) = 0.01$$

$$y = 1 \times 1.009 + 1 \times (-1.009) + 0.01 = 0.01$$

$$\frac{\partial L}{\partial w_2} = -1(1 - 0.01) = -0.99$$

$$\frac{\partial L}{\partial w_3} = +1(1 - 0.01) = 0.99$$

$$\frac{\partial L}{\partial b} = -1(1 - 0.01) = -0.99$$

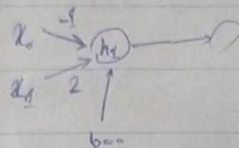
$$w_2 = 0.99(1.009) - 0.01(-0.99) = 1.0089$$

$$w_3 = 0.99(-1.009) - 0.01 \times 0.99 = -1.0089$$

$$b = 0.99(0.01) + 0.01 \times 0.99 = 0.0198$$

$$W \leftarrow W - \epsilon (\alpha \text{sign}(W) + \nabla_W J(\theta; x, y)) \quad \text{: 1st iteration}$$

$$1st \text{ iteration: } x=0, y=0, \epsilon=0.01$$



$$y_{\text{pred}} = w_0 x_0 + b_0 = -0 + 0 = 0$$

$$\frac{\partial L}{\partial w_0} = +10$$

$$\frac{\partial L}{\partial w_1} = -10$$

$$\frac{\partial L}{\partial b_0} = -10$$

$$w_0 = 0 - 0.01 (0 \times 1 + 10) = -0.1$$

$$w_1 = 0 - 0.01 (0 \times 1 - 10) = 0.1$$

$$b_0 = 0 - 0.01 (0 + (-10)) = 0.1$$

$$2nd \text{ iteration: } x=1, y=0 \Rightarrow y_{\text{pred}} = (-0.1 \times 1) + (0.1 \times 0) + 0.1 = 0$$

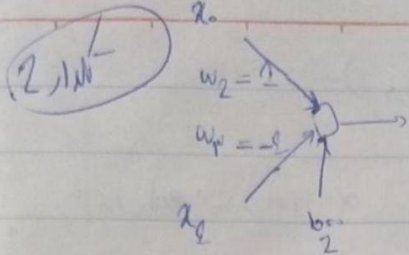
$$= -0.4 \quad \text{Loss}$$

$$\frac{\partial L}{\partial w_0} = 0 \Rightarrow w_0 = -0.1 - 0.01 (0 \times 1 + 10) = -0.2$$

$$\frac{\partial L}{\partial w_1} = 0 \Rightarrow w_1 = 0.1 - 0.01 (0 \times 1 - 10) = 0.2$$

$$\frac{\partial L}{\partial b_0} = 0 \Rightarrow b_0 = 0.1 - 0.01 (0 + (-10)) = 0.2$$

$$b_0 = 0.2$$



$$1.0.15: w_0 x + b = 0 - 1 + 2 = 1$$

$$\frac{\partial L}{\partial w_2} = -1(1 - 0.1) = -0.9$$

$$w_2 = 0.1 - 0.1(0.1(1) + (-1)) = 0.1$$

$$\frac{\partial L}{\partial w_3} = +1$$

$$w_3 = -1 - 0.1(0.1(1) + 1) = -1.1$$

$$\frac{\partial L}{\partial b} = -1$$

$$b_2 = 0 - 0.1(0.1 \times 1) - 1 = -0.1$$

$$2.0.15: \text{Cost} = 10 \times (0.1 \times 0.1) + 1 \times (-1.1 \times 0.1) + 0.1 = -0.4$$

$$\frac{\partial L}{\partial w_2} = -1(1 - 0.1) = -0.9 \quad \text{Relu}$$

$$\frac{\partial L}{\partial w_3} = -1(1 - 0.1) = -0.9 \quad \text{Relu}$$

$$\frac{\partial L}{\partial b} = -1(1 - 0.1) = -0.9 \quad \text{Relu}$$

$$w_2 = 0.1 - 0.1(0.1 \times 0.1) = 0.09$$

$$w_3 = -1.1 - 0.1(0.1 \times 1.1) = -1.11$$

$$b_2 = -0.1 - 0.1(0.1 \times 0.1) = -0.101$$