به نام خدا

محمد زیاری (۹۷۲۲۲۰۴۷)

تاریخ: ۱۴۰۰/۱۱/۰۵

گزارش تمرین سری ۵ – سوالات تئوری

درس: مبانی علوم داده

استاد درس: دکتر خردپیشه

۱.

a) اگر برای محاسبه loss مدل از negative log likelihood استفاده کنیم، درواقع در مرحله اول loss را منفی کردیم. چرا اینکار را انجام میدهیم؟ زیرا اکثر framework های یادگیری عمق، با بهینه گر هایی که Minimizeمیکنند سر و کار دارند اما در این مواقع ما میخواهیم احتمال انتخاب شدن کلاس درست را ماکسیمم کنیم. به همین علت از منفی استفاده میکنیم. همچنین عملیات log برای آن است که مقادیر کوچک تری را داشته باشیم و به جای عملیات ضرب، جمع انجام دهیم تا مدل سریع تر عمل کند.Likelihood نیز برای آن استفاده میشود که پارامتر های مدل طوری انتخاب شوند که نزدیک به الگوی داده ها باشند تا خیلی از مدل و داده ها فاصله نگیریم.
پس استفاده از negative log likelihood بسیار به ما کمک میکند تا نتایج بهتری از مدل داشته باشیم و محاسبات loss را سریعتر میکند. هرچه این مقدار کمتر باشد یعنی پیشبینی ما دقیق تر بوده و نزدیک تر به جواب اصلی است. نحوه محاسبه loss توسط این الگو بسیار دقیق است و اگر کلاس بندی ما بسیار به مقدار واقعی نزدیک باشد مقدار کمتری نمایش خواهد داد.

b) دو روش L1 Regularization یا Lasso و L2 Regularization یا Ridge، برای کنترل کردن وزن ها و تعداد فیچرهای مدل استفاده میشوند. درواقع با اضافه شدن یک جمله پنالتی به cost function میتوانند تاثیر در عملکرد مدل داشته باشند.

برای Lasso داریم:

$$\text{Cost} = \sum_{i=0}^{N}(y_i - \sum_{j=0}^{M} x_{ij}W_j)^2 + \lambda \sum_{j=0}^{M}|W_j|$$

همانطور که میبینیم، یک جمله که حاصل جمع قدرمطلق وزن های مدل است، با ضریب ثابت λ به cost اضافه شده است. تغییرات وزن ها و همچنین تغییرات تابع cost به ضریب λ وابسته است. به طوری که اگر λ صفر باشد که این پنالتی درواقع بی اثر است اما اگر این مقدار بسیار بزرگ باشد، باعث میشود تا ضرایب نیز بزرگ شوند. پس باید مقدار λ به درستی انتخاب شود تا تاثیر گذار باشد. اگر λ درست انتخاب شود آنگاه این جمله باعث خواهد شد تا برخی فیچر هایی که از اهمیت کمتری در مدل برخوردار هستند، از آن حذف شوند. زیرا ضرایب آنها را صفر میکند و انتخاب ویژگی انجام میدهد.

برای Ridge داریم:

$$\text{Cost} = \sum_{i=0}^{N}(y_i - \sum_{j=0}^{M} x_{ij}W_j)^2 + \lambda \sum_{j=0}^{M} W_j^2$$

همانطور که میبینیم، در اینجا جمله ای که اضافه شده است، مجموع مربعات وزن ها است که باز هم با ضریب λ به مدل اضافه میشود. در این حالت، باز هم تعیین λ بسیار مهم است اما باید توجه داشته باشیم که برخلاف Lasso در این regularization، وزن ها به سمت صفر میل نخواهند کرد بلکه تنها باعث میشود که این وزن ها کوچک بمانند و زیاد بزرگ نشوند تا از overfitting و یا underfitting جلوگیری شود.

مقایسه Lasso و Ridge:

- تفاوت اصلی این دو در آن است که Lasso درواقع سعی دارد میانه داده را پیدا کند در حالی که Ridge با پیدا کردن میانگین آنها سعی دارد تا از overfit شدن جلوگیری کند.
- روش Lasso وزن ها را صفر میکند، در حالی که Ridge تنها آنها را کوچک نگه میدارد.
- روش Lasso یک نوع feature selection است.
- با استفاده از Ridge تنها یک یادگیری برای مدل ایجاد میشود.
- روش Lasso عملکرد خوبی برروی دادهه ای outlier نیز دارد درصورتی که Ridge اینگونه نمیباشد.

c) در عملیات آپدیت کردن وزن ها در مدل های شبکه های عصبی، momentum میتواند نقش مهمی داشته باشد. از این ضریب میتوان در مواقعی استفاده کرد. Momentum به ما کمک میکند تا در local minimum ها گیر نکنیم. این روش m برابر وزن قبلی را به وزن جدید اضافه میکند تا آن را آپدیت کند. انتخاب m بسیار مهم است. باید طوری انتخاب شود که خیلی زیاد نباشد که global minimum را رد کنیم و خیلی کم نیز نباشد تا بتواند سرعت عملیات را افزایش دهد و باعث گیر افتادن در local minimum ها نشود.
زمانی که بردار گرادیان به یک جهت اشاره میکند، momentum گام ها رو بزرگتر میکند تا به مینمم مورد نظر برسد، با این روش سرعت مدل افزایش میابد. همچنین زمانی که گرادیان بسیار زیاد تغییر جهت میدهد، momentum کمک میکند تا این تغییر جهات کمتر شود.

۲.

(a

در شکل داده شده یک لایه hidden داریم و ۲ نورون در هر لایه، برای مرحله feed forward باید محاسبه کنیم که چه مقدار از ورودی مدل به لایهhidden و سپس به خروجی میرود. برای اینکار نیاز است که وزن مربوطه را در ورودی ضرب کنیم تا نتایج حاصل در لایه hidden و سپس در خروجی را بدست آوریم. ابتدا مقادیر لایه hidden را محاسبه میکنیم:

برای هر مرحله ابتدا مقدار حاصل ضرب وزن در مقدار ورودی (به اضافه bias) را محاسبه کرده و سپس با اعمال تابع sigmoid بر آن، خروجی آن را تعیین میکنیم.

نورون h1:

$$net_{h1} = w_1 * i_1 + w_2 * i_2 + b_1 = 0.15 * 0.05 + 0.20 * 0.10 + 0.35 = 0.3775$$

$$out_{h_1} = \frac{1}{1 + e^{-net_{h1}}} = 0.593269$$

نورون *h2*:

$$net_{h2} = w_3 * i_1 + w_4 * i_2 + b_1 = 0.05 * 0.25 + 0.10 * 0.30 + 0.35 = 0.3925$$

$$out_{h2} = \frac{1}{1 + e^{-net_{h2}}} = 0.596884$$

سپس نیاز است تا برای o1 و o2 نیز با توجه به مقادیر بدست آمده در لایه hidden، همین محاسبات را انجام دهیم

برای o1:

$$net_{o1} = w_5 * out_{h1} + w_6 * out_{h2} + b_2 = 0.40 * 0.593269 + 0.45 * 0.596884 + 0.60 = 1.1059054$$

$$out_{o1} = \frac{1}{1 + e^{-net_{o1}}} = 0.7513649$$

برای o2:

$$net_{o2} = w_7 * out_{h1} + w_8 * out_{h2} + b_2 = 0.50 * 0.593269 + 0.55 * 0.596884 + 0.60 = 1.2249207$$

$$out_{o2} = \frac{1}{1 + e^{-a_{o2}}} = 0.77292834$$

در مرحله آخر پس از بدست آوردن مقادیر o1 وo2 توسط فرمول mean squared error خطا را محاسبه میکنیم:

$$E_{o1} = \frac{1}{2}(target_{o1} - out_{o1})^2 = \frac{1}{2}(0.01 - 0.7513649)^2 = 0.2748109575$$

$$E_{o2} = \frac{1}{2}(target_{o2} - out_{o2})^2 = \frac{1}{2}(0.99 - 0.77292834)^2 = 0.0235600528$$

به طور کلی خطا برابر است با:

$$E_{total} = E_{o1} + E_{o2} = 0.298371$$

(b

عمل backpropagation را از لایه آخر آغاز میکنیم، بدین ترتیب که از لایه خروجی به لایه hidden باز میگردیم و با اسفتاده از مشتق های جزئی و زنجیره‌ای سعی میکنیم که وزن های مدل را در هر مرحله آپدیت کنیم و با اینکار وزن ها را کاهش میدهیم.

با استفاده از این روش وزن ها پس از چندین بار تکرار، بسیار کاهش می یابند و باعث میشود تا خطای مدل نیز کاهش یابد.

ابتدا وزن های w5 تا w8 را آپدیت میکنیم.

آپدیت وزن w5:

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_{o1}} * \frac{\partial out_{o1}}{\partial a_{o1}} * \frac{\partial net_{o1}}{\partial w_5}$$

با توجه به معادلات E_{total} , out_{o1}, net_{o1} مشتقات زیر را محاسبه میکنیم:

$$\frac{\partial E_{total}}{\partial out_{o1}} = -(target_{o1} - out_{o1}) = -(0.01 - 0.7513649) = 0.7413649$$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}\,(1 - out_{o1}) = 0.7513649\,(1 - 0.7513649) = 0.1868156$$

$$\frac{\partial net_{o1}}{\partial w5} = out_{h1} = 0.593269$$

پس مقدار مشتق زنجیره‌ای برابر است با:

$$\frac{\partial E_{total}}{\partial w5} = 0.7413649 * 0.1868156 * 0.593269 = 0.082166$$

حال وزن w_5 را با استفاده از معادله زیر آپدیت میکنیم:

$$w_5' = w_5 - \eta * \frac{\partial E_{total}}{\partial w_5}\ , \eta = 0.3$$

$$w_5' = 0.40 - 0.3 * 0.082166 = 0.3753502$$

برای وزن w_6 تنها مقدار جمله آخر تغییر میکند:

$$\frac{\partial E_{total}}{\partial w_6} = 0.7413649 * 0.1868156 * 0.596884 = 0.082667$$

$$w_6' = 0.45 - 0.3 * 0.082667 = 0.425199$$

اما برای محاسبه w_7 , w_8 نیاز است تا تغییراتی در این مشتق گرفتن ایجاد شود

$$\frac{\partial E_{total}}{\partial w_7} = \frac{\partial E_{total}}{\partial out_{o2}} * \frac{\partial out_{o2}}{\partial net_{o2}} * \frac{\partial net_{o2}}{\partial w_7}$$

$$\frac{\partial E_{total}}{\partial out_{o2}} = -(target_{o2} - out_{o2}) = -(0.99 - 0.77292834) = 0.21707166$$

$$\frac{\partial out_{o2}}{\partial net_{o2}} = out_{o2}(1 - out_{o2}) = 0.77292834(1 - 0.77292834) = 0.1755101$$

$$\frac{\partial a_{o2}}{\partial w_7} = out_{h1} = 0.593269$$

$$\frac{\partial E_{total}}{\partial w_7} = -0.21707166 * 0.1755101 * 0.593269 = -0.02260252$$

$$w_7' = 0.50 + 0.3 * 0.02260252 = 0.506780$$

برای w_8 نیز تنها مقدار جمله آخر را تغییر میدهیم:

$$\frac{\partial E_{total}}{\partial w_8} = -0.21707166 * 0.1755101 * 0.596884 = -0.022740247$$

$$w_8' = 0.55 + 0.3 * 0.022740247 = 0.556822074$$

وزن های w1 w2 w3 w4 را باید با محاسباتی از لایه hidden بدست بیاوریم، برای اینکار به صورت زیر عمل میکنیم:

محاسبه وزن w_1:

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1}$$

برای محاسبه جمله اول ابتدا نیاز است که E_{total} را به صورت جمع E_{o1} و E_{o2} بنویسیم، تا بتوانیم هرکدام را جداگانه محاسبه کرده و سپس جمع کنیم:

$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

محاسبه بخش آبی:

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial out_{h1}}$$

این مقادیر را از محاسبات بالا داریم:

$$\frac{\partial E_{o1}}{\partial net_{o1}} = \frac{\partial E_{o1}}{\partial out_{o1}} * \frac{\partial out_{o1}}{\partial net_{o1}} = 0.7413649 * 0.1868156 = 0.138498 \quad , \quad \frac{\partial net_{o1}}{\partial out_{h1}} = w_5 = 0.40$$

$$\frac{\partial E_{o1}}{\partial out_{h1}} = 0.138498 * 0.40 = 0.055399$$

محاسبه بخش قرمز:

$$\frac{\partial E_{o2}}{\partial out_{h1}} = \frac{\partial E_{o2}}{\partial net_{o2}} * \frac{\partial net_{o2}}{\partial out_{h1}}$$

$$\frac{\partial E_{o2}}{\partial net_{o2}} = \frac{\partial E_{o2}}{\partial out_{o2}} * \frac{\partial out_{o2}}{\partial net_{o2}} = -0.21707166 * 0.1755101 = -0.038098 \quad , \quad \frac{\partial net_{o2}}{\partial out_{h1}} = w_7 = 0.50$$

$$\frac{\partial E_{o2}}{\partial out_{h1}} = -0.038098 * 0.50 = -0.019049$$

در نتیجه:

$$\frac{\partial E_{total}}{\partial out_{h1}} = 0.055399 - 0.019049 = 0.03635$$

$$\frac{\partial out_{h1}}{\partial net_{h1}} = 0.241300$$

$$\frac{\partial net_{h1}}{\partial w1} = i_1 = 0.05$$

$$\frac{\partial E_{total}}{\partial w_1} = 0.0363 * 0.241300 * 0.05 = 0.000438$$

$$w_1' = 0.15 - 0.3 * 0.000438 = 0.14986$$

محاسبات برای w_2 نیز به همین صورت است تنها جمله آخر تفاوت دارد

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2}$$

$$\frac{\partial E_{total}}{\partial out_{h1}} = 0.03635$$

$$\frac{\partial out_{h1}}{\partial net_{h1}} = 0.241300$$

$$\frac{\partial net_{h1}}{\partial w_2} = i_2 = 0.10$$

$$\frac{\partial E_{total}}{\partial w_2} = 0.0363 * 0.241300 * 0.10 = 0.0008759$$

$$w_2' = 0.20 - 0.3 * 0.0008759 = 0.199737$$

برای w_3 و w_4 باید از جمله net_{h2} مشتق بگیریم، پس معادله کمی تغییر می کند:

$$\frac{\partial E_{total}}{\partial w_3} = \frac{\partial E_{total}}{\partial out_{h2}} * \frac{\partial out_{h2}}{\partial net_{h2}} * \frac{\partial net_{h2}}{\partial w_3}$$

مانند قبل جمله اول را دو قسمت میکنیم:

$$\frac{\partial E_{o1}}{\partial out_{h2}} = \frac{\partial E_{o1}}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial out_{h2}}$$

$$\frac{\partial E_{o1}}{\partial net_{o1}} = \frac{\partial E_{o1}}{\partial out_{o1}} * \frac{\partial out_{o1}}{\partial net_{o1}} = 0.7413649 * 0.1868156 = 0.138498$$

$$\frac{\partial net_{o1}}{\partial out_{h2}} = 0.45$$

$$\frac{\partial E_{o1}}{\partial out_{h2}} = 0.138498 * 0.45 = 0.0623241$$

$$\frac{\partial E_{o2}}{\partial out_{h2}} = \frac{\partial E_{o2}}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial out_{h2}} = 0.0761739$$

$$\frac{\partial E_{total}}{\partial out_{h2}} = 0.0623241 + 0.0761739 = 0.138498$$

$$\frac{\partial out_{h2}}{\partial net_{h2}} = = 0.2406134$$

$$\frac{\partial net_{h2}}{\partial w_3} = i_1 = 0.05$$

$$\frac{\partial E_{total}}{\partial w_3} = 0.138498 * 0.2406134 * 0.05 = 0.0016662$$

$$w_3' = 0.25 - 0.3 * 0.0016662 = 0.249500$$

برای w_4 :

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial E_{total}}{\partial out_{h2}} * \frac{\partial out_{h2}}{\partial net_{h2}} * \frac{\partial net_{h2}}{\partial w_4}$$

$$\frac{\partial net_{h2}}{\partial w_4} = i_2 = 0.10$$

$$\frac{\partial E_{total}}{\partial w_4} = 0.138498 * 0.2406134 * 0.10 = 0.0033324$$

$$w_4' = 0.30 - 0.3 * 0.0033324 = 0.2990002$$

.۳

Try better random initialization for the weights: بله - تاثیر مثبتی دارد . قاعدتا اگر وزن ها از ابتدا میزان رندم با متر و معیاریهتری داشته باشند می تواند باعث شود زودتر به مقدار بهینه برسیم.

Try mini-batch gradient descent: بله - زمانی که از batch های کوچکتر استفاده می کنیم در مدت زمان کوتاه تری خطایمان مینیمم می شود.

Try using Adam: بله - به طور کلی با استفاده از اپتیمایزر آدام زودتر به جواب می رسیم

Try initializing all the weights to zero : خیر - این روش نمی تواند همیشه جواب درست را به ما دهد و ممکن است با 0 کردن تمامی وزن ها خیلی دیر به جواب برسیم.

Try tuning the learning rate α : بله - تنظیم کردن میزان لرنینگ ریت و تغییر ان می تواند نتیجه مثبتی داشته باشد و باید لرنینگ ریتی را انتخاب کنیم که در عین حال که زود به جواب می رسیم به طور دقیق هم جواب را برایمان پیدا کند.