به نام خدا

تمرین سری سوم درس یادگیری عمیق

دکتر محمدی

محمد یارمقدم

96462104

سوال اول)

مورد یک:

در مدل های شبکه عصبی از چندین بهینه ساز می توان استفاده کرد. در این بخش به مقایسه این بهینه ساز ها می پردازیم.

در ابتدا به adam می پردازیم. این بهینه ساز یک الگوریتم است که بهینه سازی غیر محدب را به خوبی انجام می دهد. این بهینه ساز مزایایی مانند: پیاده سازی سریع، بهره محاسباتی، بهبود بهره مکانی و ... دارد و برای داده های پراکنده و با شیب زیاد و اهداف غیر ثابت مناسب است. این بهینه ساز معمولا به تنظیم کمی نیاز دارد. به علاوه می توان گفت که این بهینه ساز ترکیبی از adaGrad و RMSProp است. این اپتیمایز مزایای هر دو این موارد را دارد. علاوه بر آپدیت lr بر اساس میانگین ممان اول، این بهینه ساز میانگین ممان دوم گرادیان هم در آپدیت دخیل می کند. علی الخصوص، این الگوریتم یک میانگین نمایی از گرادیان و مشتق گرادیان کحاسبه می کند. هم چنین پارامتر های beta1 و beta2 میزان فروپاشی این میانگین های متحرک را کنترل می کنند.

طبق توضیحات بالا و نمودار سوال، RMSProp و Adadelta و Adam در شرایط مشابه عملکرد خوبی دارند. تصحیح bias در بهینه ساز adam باعث می شود که adam از RMSProp کمی بهتر باشد، به این دلیل که گرادیان ها پراکنده تر می شوند. به همین دلیل در حالت کلی adam بهترین گزینه است و هزینه آموزش کمتری دارد. گزینه جایگزین برای آن می تواند SGD + Nestrov Momentum باشد.

در حالت کلی SGD به آرامی به سمت همگرایی و جواب حرکت می کند و نوسان زیادی دارد ولی فضای کمی نیاز دارد و از آنجا که در هر زمان فقط یک نمونه آپدیت میشود برای داده های بزرگ بهتر است اما هزینه آپدیت زیاد است. پس از آن Adagrad نوسان در خلاف جهت جواب دارد اما با سرعت بیشتری از SGD به جواب می رسد. در رده بالایی RMSProp قرار دارد که هم نوسان کمتری از مورد قبل دارد هم با زمان بهتری به سمت همگرایی حرکت می کند.

اما دو بهینه ساز عملکرد متفاوتی دارند. SGD + Momentum و Nestrov + SGD در این دسته قرار دارند. به علت اینکه مومنتوم را در آپدیت متغیر ها دخیل می کنند زود تر به جواب می رسد اما در ابتدای روند نوسان نسبتا شدیدی دارند که کمتر ازSGD است ولی با سرعت بالایی از رقبای خود پیشی می گیرند و در طول زمان نوسان کم و کمتر میشود.

پس مقایسه کلی را به صورت زیر خلاصه می کنیم:

RMSProp و Adagrad نوسان کمی دارند و با روند تقریبا ثابتی به سمت هدف حرکت می کنند زیرا مومنتوم در آپدیت دخیل نیست و سرعت تغییر آنچنانی نمی کند. اما در Momentum و NAG نوسان در ابتدا زیاد است و مسیر خوبی طی نمیشود اما چون سرعت در جهت هدف بیشتر میشود در ادامه از این دو پیشی می گیرند.

Adam نیز از آنجا که ترکیب RSMProp و Adagrad است هم نوسان کم هم سرعت خوب دارد که مشکلات را به حداقل می رساند.

پس در داده های پراکنده استفاده از RMSProp و Adam و Adagrad پیشنهاد میشود. این سه موارد یکسان زیادی دارند و نتایج مشابهی اغلب دارند. هر چه پراکندگی داده بیشتر شود adam از RMSProp بهتر عمل میکند.

Adadelta هم در ابتدا ضعیف عمل می کند اما در نهایت از RMSProp نتیجه بهتری را در داده های پراکنده ثبت میکند.

منبع کمکی : https://medium.com/mlearning-ai/optimizers-in-deep-learning-7bf81fed78a0

مورد دو:

1- Sparse بودن داده ها
2- تعداد داده ها
3- همگرایی گلوبال
4- همگرایی محلی
5- نقش روش بهینه سازی اساسی
6- نقش بازگشت چند شبکه ای(multigrid recursion)
7- ویژگی های مدل بهینه سازی
8- وزن ها و پارامترهای قبلی

تعداد داده ها روی عملکرد بهینه ساز ها تاثیر دارد. به طور مثال تعداد عمل هایی که روی هر داده انجام می شود که با تعداد داده رابطه مستقیم دارد، روی عملکرد تاثیر دارد. مثلا SGD از آنجا که هزینه محاسباتی کمتری دارد روی داده های بزرگ مناسب تر است.

Sparse بودن داده نیز از آنجا روی عملکرد موثر است که بعضی موارد از بهینه ساز ها بهتر می توانند روی sparse data بهینه سازی انجام دهند، زیرا نوسان اولیه شدید و روند بهبودی این مورد باعث می شود بهتر این نوع دیتا را ارزیابی کنند. به طور مثال RMSProp و Adam بهتر روی این نوع کار می کنند.

همگرایی های محلی و گلوبال نیز بسیار تاثیر دارند. زیرا در بعضی از بهینه ساز ها فرار از این همگرایی ها سخت تر است و نوع و تعداد این موارد روی عملکرد تاثیر دارد.

ویژگی های مدل نیز در موارد بالا بیان شد.

منبع : https://link.springer.com/chapter/10.1007/0-387-29550-X_6

سوال دوم)

د)

میانگین در اوایل باید 1 باشد و در آخر باید به -1 برسد چون در اوایل مقدار ورودی ها 1 و در آخر -1 است.

در حالت اول به 0 می رسد در آخر پس دچار خطا نیز هست. پس روش اول که میانگین تمجعی خوب متوجه نشده روش خوبی برای فهمیدن میانیگن لحظه ای نیست.

در حالت دوم اولین مقدار 0.1 است و به سمت 1 می رود اما بعد از به سرعت به سمت -1 میرود. اما چون اوایل طول می کشد تا به 1 برسد مقداری از داده رو نادیده می گیرد که مناسب نیست.

روش آخر در ده epoch اول به مقدار خوب می رسد و از میانگین تجمعی بهتر است.

تغییر بتا در روش اول که تاثیری نداشت اما برای روش های بعدی باعث می شود عمده حالت فعلی با حالت قبل منتقل شود و عملا حالت فعلی تاثیری روی میانگین گیری ندارد و فقط از حالت قبل تاثیر می پذیرد که مناسب نیست. پس باید بتا مناسب ست شود که تاثیر دو حالت مشهود باشد.

سوال سوم)

الف)

این دیتاست مجموعه ای از 60000 نمونه آموزش است و 10000 نمونه تست دارد. هر عکس یک grayscale است که 28*28 است. پس در مجموع 70000 نمونه در این مجموعه موجود است. هدف این پایگاه داده جایگزینی داده های پیچیده تر MNIST (اعداد دست نویس) در منطق یادگیری ماشین است.

این مجموعه داده 10 لیبل دارد. این لیبل ها عبارت اند از:

- T-shirt/top
- Trouser
- Pullover
- Dress
- Coat
- Sandal
- Shirt
- Sneaker
- Bag
- Ankle boot

ب)

از 60000 داده آموزش، 5000 نمونه به عنوان داده اعتبارسنجی در نظر گرفتم. چون معمولا بین 10 تا 20 درصد برای این مقدار در نظر میگیرند تا هم اثر آموزش خراب نشود هم بتوان تایید آنرا بدست آورد.

ج)

دقت فاز آموزش: %96829998493194 58

دقت فاز تست: %95160001516342 16

در فاز آموزش به خوبی داده ها یادگیری شده اند که تایید آن را در داده تست می توانیم مشاهده کنیم.

د)

خیر، زیرا در بحث ساختار شبکه تغییری ایجاد نشده و آموزش باز هم با تعداد 128 نورون بهتر انجام می شود.

ه)

Adam بهترین optimizer است. زیرا بهترین دقت در داده تست و آموزش را برای ما فراهم کرد. زیرا مزیت های RMSProp و Adagrad را دارد و باید دقت بهتری را بدهد که تایید این موضوع را در شبکه دیدیم.

و)

تغییر محسوسی در دقت نمونه ها ایجاد نشد. ولی هرچه دقت learning rate کمتر باشد نتیجه بهتر خواهد بود. چون نوسان ها کمتر میشود. نمودار در نوت بوک موجود است.

ز)
بله همگرا میشود. نمودار در نوت بوک رسم شده است.

ح)
آخرین قسمت نوت بوک بیانگر این قسمت است.