

گزارش تمرین اول پایتون

زهرا نیازی

اطلاعات گزارش	چکیده
تاریخ:	هدف از این گزارش، بررسی عوامل موثر بر عملکرد یک شبکه عصبی تعلیم دیده شده برای طبقه‌بندی دیتاست MNIST می‌باشد. به خصوص تاثیر تغییر تعداد لایه‌های شبکه، تعداد نرون‌های هر لایه، و optimizer انتخاب شده بر عملکرد شبکه را بررسی می‌کنیم.
واژگان کلیدی:	
Neural Network	
MLP	
Multilayer Perceptron	
MNIST	
Accuracy	

1- مقدمه

در این گزارش، عملکرد یک شبکه عصبی تعلیم دیده شده برای طبقه‌بندی دیتاست MNIST، با توجه به عوامل مختلفی که بر آن تاثیر می‌گذارند، بررسی می‌شود. عواملی که در این بررسی مورد توجه قرار می‌گیرند، تعداد لایه‌های شبکه، تعداد نرون‌های هر لایه، و optimizer انتخاب شده هستند. هدف از این بررسی، تحلیل شبکه‌های مختلفی که به دست می‌آیند و تاثیر آنها در بهبود عملکرد شبکه برای طبقه‌بندی داده‌های MNIST است. در این گزارش، به صورت خلاصه، اهمیت و ضرورت تغییر این عوامل برای بهبود عملکرد شبکه مورد بحث قرار می‌گیرد.

به طور کلی اما افزایش تعداد لایه‌ها، باعث افزایش قدرت پردازش شبکه می‌شود. اما نکته‌ای که باید به آن توجه کرد این است که هرچه از لایه آخر به سمت لایه اول می‌آییم، اندازه گرادیانی که backpropagate می‌شود به دلیل متوسط‌گیری متعدد کمتر و کمتر می‌شود و vanishing gradient رخ می‌دهد. از طرفی لایه اول لایه بسیار مهمی است که ورودی‌ها به آن متصل هستند. اما اگر عمق شبکه خیلی زیاد باشد، باعث می‌شود نسبت به لایه اول تعلیمی انجام نشود. پس بهتر است تعداد لایه‌های شبکه را عدد خیلی بزرگی در نظر نگیریم.

2-2- بررسی تاثیر تغییر تعداد نرون‌ها در لایه مخفی و optimizer

حال برای بررسی تاثیر تعداد نرون‌ها در تعلیم شبکه، حالت‌های مختلفی را آزمایش می‌کنیم. هنگامی که حداکثر تعداد نرون‌های لایه یک شبکه ۳ لایه‌ای ۱۲۸ باشد، تعداد پارامترهای شبکه حدود ۶۰,۰۰۰ می‌باشد. در این حالت دقت تعلیم و تست Adagrad optimizer از SGD بیشتر است. همچنین دقت Adam در تمام حالت‌هایی که خواهیم داشت به مراتب پایین‌تر از این دو می‌باشد.

اگر حداکثر تعداد نرون‌های لایه یک شبکه ۳ لایه‌ای را ۲ برابر کنیم تعداد پارامترها به ۸۰۰,۰۰۰ تا افزایش پیدا می‌کنند. و اگر حداکثر تعداد نرون‌های سک شبکه ۳ لایه ۲۰۴۶ باشد، تعداد پارامترهای شبکه به ۳,۰۰۰,۰۰۰ افزایش پیدا می‌کند. در این دو حالت SGD optimizer نسبت به Adagrad عملکرد بهتری دارد.

با تصحیح و خطا می‌توان به بهترین عدد برای تعداد نرون‌های هر لایه رسید. اما به طور کلی، اگر در نظر بگیریم

2- بخش اول

2-1- بررسی تاثیر تغییر تعداد لایه‌ها و optimizer

در این بخش ابتدا شبکه را به ترتیب با ۱، ۲ و ۳ لایه و با ۳ optimizer متفاوت تعلیم می‌دهیم. برای هر یک از این حالت‌ها، میزان accuracy و loss را هم نمایش می‌دهیم. هنگام بررسی optimizerها برای شبکه تک‌لایه، مشاهده می‌کنیم که Adagrad عملکرد بهتری هنگام تعلیم و تست دارد.

با افزایش تعداد لایه‌ها به ۲، مشاهده می‌کنیم که Adagrad همچنان عملکرد بهتری دارد اما عملکرد Adam به شدت افت کرده است.

هنگامی که تعداد لایه‌ها ۳ است، عملکرد Adagrad بهتر از دیگر optimizerهاست.

* zahra.niazi2001@gmail.com

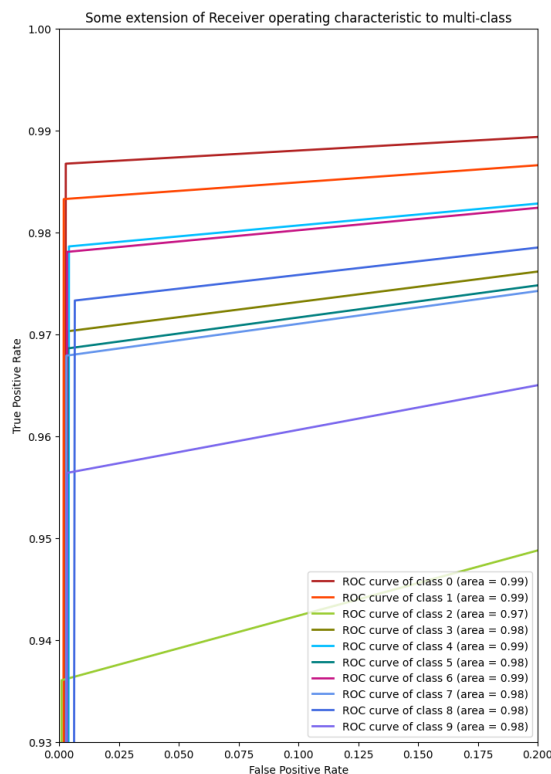
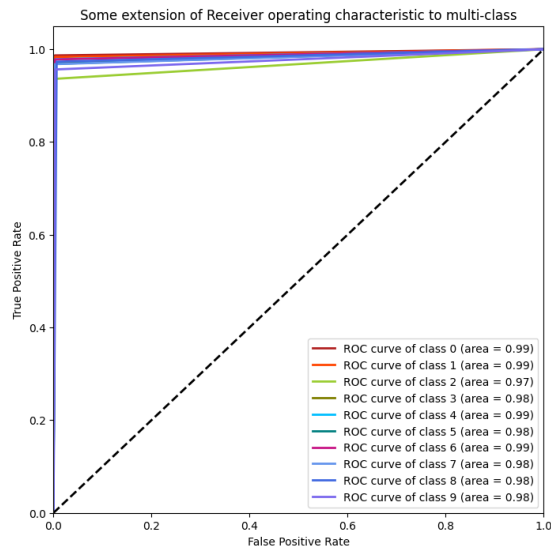
2-4- محاسبه پارامترهای مختلف برای بهترین شبکه

حال recall, precision, ROC curve, learning curve را برای این شبکه به دست می‌آوریم.

accuracy = 0.9699

precision = 0.9701289568362584

recall = 0.9699



2-بخش دوم

حال از ما خواسته شده داده های ۵ کلاس اول را به عنوان داده های آموزش و داده های ۵ کلاس بعدی را به عنوان داده های تست استفاده بکنیم.

که تعداد مجهولات همان تعداد پارامترهای شبکه، و تعداد معلومات برابر با تعداد الگوها ضرب در تعداد خروجی‌ها باشد، قاعده‌ای که می‌توان به عنوان یاندی روی تعداد پارامترها در نظر گرفت، به این صورت است که تعداد مجهولات باید کمتر مساوی نصف تعداد معلومات باشد. تعداد معلومات اینجا $60,000 = 10 * 6,000$ می‌باشد پس بهتر است تعداد پارامترهای شبکه تا حداکثر ۳۰,۰۰۰ باشد.

هرچه تعداد پارامترها از این تعداد بالاتر برود، پیچیدگی ساختاری شبکه بیشتر می‌شود و به تبع آن بعد از تعلیم می‌توان انتظار داشت خطای تعلیم کمتر شود. اما برای ما خطای تخمین مهم است که با افزایش پیچیدگی ساختاری، افزایش می‌یابد و باعث افزایش ریسک ساختاری می‌شود و overfitting رخ می‌دهد. که در این حالت ما شبکه‌ای خواهیم داشت که قابلیت یادگیری مسائل پیچیده را دارد، اما قاعده کلی بین مسائل را یاد نمی‌گیرد چون شبکه بزرگتر از حد نیاز است و generalization ندارد. پس همه مسئله‌ها را حفظ کردن اما یک قاعده کلی را یاد نمی‌گیرد.

2-3- بررسی تاثیر تغییر پارامتر optimizer

همچنین در هر دو حالت تغییر تعداد لایه‌ها و تغییر تعداد نرون‌ها، optimizer شبکه‌ای که بهترین عملکرد داشته را انتخاب کرده، learning rate آن را تغییر می‌دهیم تا تاثیر این تغییرات را بررسی کنیم.

به طور کلی الگوریتم backpropagation، الگوریتمی کند است. به خصوص در نواحی که مقدار گرادیان کم باشد این الگوریتم بسیار کند عمل می‌کند. همچنین به یک حداقل محلی همگرا می‌شود. این الگوریتم به مقدار طول گامی (learning rate) که در جهت گرادیان برمی‌دارد نیز حساس است. اگر این طول گام به اندازه کافی کوچک باشد، الگوریتم همگرا است و در هر گام خطا کاهش می‌یابد اما ممکن است کند باشد چون تعداد تکرارها زیاد است و قدم‌هایی که برمی‌دارد کوچک‌اند. همچنین اگر این طول گام بزرگ باشد ممکن است همگرایی و کاهش بودن الگوریتم از بین برود اما گام‌هایی که برمی‌دارد بزرگ هستند و سرعت هم زیاد می‌شود.

در اینجا مشاهده می‌کنیم که پس از تغییر learning rate، سرعت همگرایی الگوریتم کاهش یافته به طوری که در تعداد iteration مساوی با حالت قبلی، دقت شبکه به مراتب کمتر است.

در انتها برحسب نتایجی که به دست آورده‌ایم شبکه‌ای با ۳ لایه و تعداد پارامترهای حدود ۶۰,۰۰۰ و Adagrad optimizer بهترین نتیجه‌ایست که به آن رسیده‌ایم.

طبیعتاً به دلیل اینکه شبکه داده های آن ۵ کلاس دوم را ندیده است، خطای تعلیم آن زیاد نخواهد بود اما خطای تخمین آن به شدت زیاد می باشد.

لینک نوتبوک:

https://colab.research.google.com/drive/12sHss_zFuO-1G_ILa8uEoib5PvCqqmDQ?usp=sharing