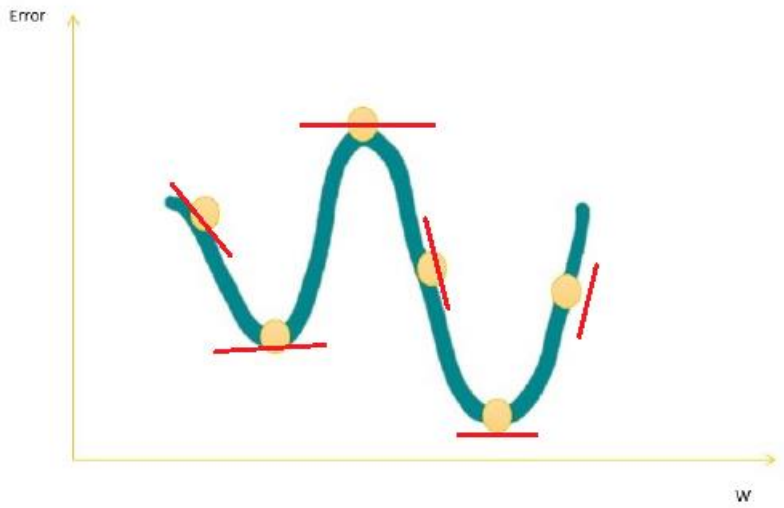
به نام خدا

تمرین دو درس شبکه های عصبی — توحید عابدینی ۹۹۷۲۲۱۵۳

سوال ۱)

بخش a)



- در نقطه یک (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمت **جلو** برویم.
- در نقطه دو (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمت **عقب** برویم که البته تغییر زیادی نخواهیم داشت و در یک مینیمم محلی گیر کرده ایم. در صورتی که مقدار گرادیان به طور دقیق صفر باشد الگوریتم ما حرکت نمیتواند بکند و واضح است که مقدار گام حرکتی ما متناسب با مقدار گرادیان است.
- در نقطه سه (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمتی برویم که شیب خط نشان میدهد که در این حالت شیب بسیار نزدیک به صفر است ولی چون در نقطه ماکسیمم سراسری هستیم اینطور نیست که بگوییم الگوریتم متوقف میشود و باید به یک سمت حرکت کند. بسته به علامت شیب خط رسم شده که به وضوح در کل قابل تشخیص نیست و مقدار عددی بسیار پایین و نزدیک به صفر دارد، شاهد تغییر وزن ها خواهیم بود. در صورتی که مقدار گرادیان به طور دقیق صفر باشد الگوریتم ما حرکت نمیتواند بکند و واضح است که مقدار گام حرکتی ما متناسب با مقدار گرادیان است.
- در نقطه چهار (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمت **جلو** برویم.
- در نقطه پنج (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمتی برویم که شیب خط نشان میدهد که در این حالت شیب بسیار نزدیک به صفر است و عملا در مینمم سراسری هستیم و اگر الگوریتم را متوقف نکنیم فقط **مقداری حرکت لرزشی به چپ و راست** خواهیم داشت (بسته به علامت شیب خط رسم شده) مگر اینکه مقدار گرادیان به طور دقیق صفر باشد و واضح است که مقدار گام حرکتی ما متناسب با مقدار گرادیان است.
- در نقطه شش (از چپ به راست) اگر بخواهیم در جهت کاهش گرادیان حرکت کنیم باید به سمت **عقب** برویم.

بخش b)

به نقاط ۲ و ۳ به ترتیب مینیمم محلی و ماکسیمم محلی (یا سراسری – بسته به وضعیت نمودار) میگویند.

یک راه حلی پیشنهادی برای رفع این مشکل استفاده از مامنتوم (Momentum) میباشد.

به طور خلاصه مامنتوم یک ضریبی از وزن های قبلی را به وزن جدید اضافه میکند که از گیر کردن الگوریتم در مینیمم یا ماکسیمم محلی جلوگیری میکند. همچنین مامنتوم باعث سریع تر شدن همگرایی و حرکت منسجمانه (با نویز کمتر) میشود.

روابط محاسبه ی مامنتوم و اعمال تغییرات پیشنهادی به شکل زیر است:

$$W_{new} = W_{old} - lr * (\nabla_W L)_{W_{old}}$$

$$V_{new} = \alpha * V_{old} + lr * (\nabla_W L)_{W_{old}}$$

$$W_{new} = W_{old} - (\alpha * V_{old} + lr * (\nabla_W L)_{W_{old}})$$

$$0 < \alpha < 1$$

مامنتوم به کمک وزن دهی به تمامی آپدیت های اخیر در مقایسه با آپدیت قبلی محاسبه میشود که به سریع تر شدن همگرایی کمک میکند. V_{old} همان بخش موثر از آپدیت های اخیر است که به کمک ما می‌آید.

مقدار مناسب آلفا هم طبق تجربه مقدار بالایی نظیر 0.9 یا 0.95 است.

منابع:

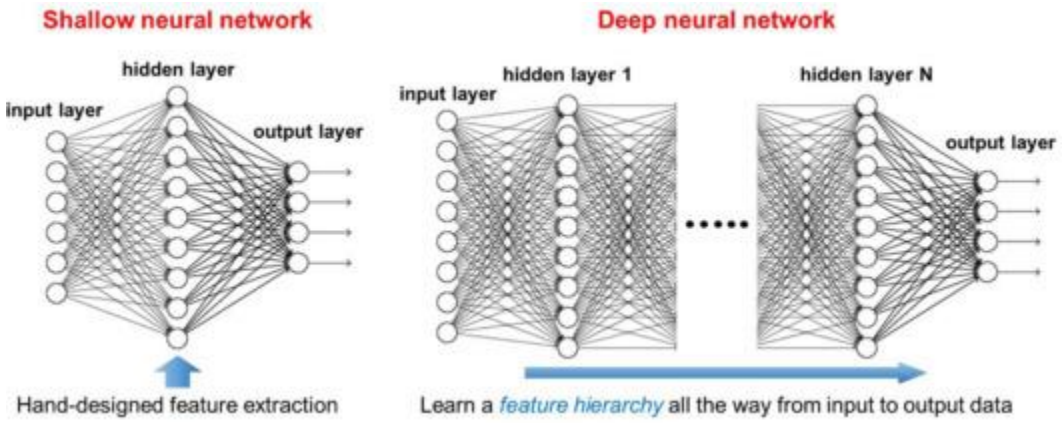
https://towardsdatascience.com/improving-vanilla-gradient-descent-f9d91031ab1d
https://towardsdatascience.com/overview-of-various-optimizers-in-neural-networks-17c1be2df6d5

سوال ۲)

تفاوت اصلی دو نوع شبکه عمیق و کم عمق در تعداد لایه ها و همچنین تعداد نورون های هر لایه است. با تنظیم این دو هایپرپارامتر میتوان عملکرد شبکه را بهتر کرد و برای هر کدام مقداری مناسب به دست آورد. برخی مقادیر تجربی که کارآیی خوبی دارند در هر نوع مختلف از شبکه ها وجود دارد که به عنوان مثال به این که تعداد نورون های هر لایه در شبکه های feed-forward بهتر است توانی از دو باشد، اشاره کرد. یا اینکه تعداد نورون ها در لایه های میانی بهتر است در ابتدا زیاد تر شود و سپس کمتر شود (یا حداقل برای مدتی ثابت بماند). داشتن لایه های میانی متعدد باعث استخراج ویژگی ها به صورت سلسله مراتبی (از ساده به پیچیده) میشود که در حل مساله برای ما کمک کننده خواهد بود. البته تا حدی که بیش از حد پیچیده نشود و دچار بیش برازش نشویم.

تعریف شبکه عصبی عمیق و کم عمق:

شبکه های عصبی به صورت کلی میتوانند به دو دسته کم عمق و عمیق تقسیم شوند. به شبکه ای که ۱ یا ۲ لایه مخفی داشته باشد شبکه عصبی کم عمق گفته میشود که با درک درست ساختار شبکه میتوانیم راجع به نحوه کار شبکه های عصبی عمیق هم درک خوبی داشته باشیم. شبکه ی عصبی با تعداد لایه های مخفی بیشتر (۳ و بالا تر) با نام شبکه های عصبی عمیق شناخته میشوند. در زیر یک تصویر برای درک بصری بهتر از معماری هر دو نوع شبکه آورده میشود. شایان ذکر است که امروزه به دلیل وجود سخت افزار مناسب و الگوریتم های متعدد و کارا، شبکه های عمیق کاربرد بسیار بیشتر و قدرتمند تری در دنیای واقعی دارند. البته حتما باید مراحل انتخاب مدل مناسب از مدل ساده تر به مدل پیچیده تر انجام شود تا پیچیدگی اضافی نداشته باشیم. و صد البته نیاز به داده یادگیری مناسب هم یکی از مسائل مهم در یادگیری شبکه های عمیق است.



منابع:

https://www.i2tutorials.com/explain-deep-neural-network-and-shallow-neural-networks

سوال ۳)

فیچر خای ۱*۶۴ گفته شده برای حروف تصویر:

حرف ب:

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,
0,1,0,0,0,0,0,0,0,0,0,0,0]

حرف گ:

[0,0,0,0,0,1,0,1,0,0,0,0,1,0,1,0,
0,0,0,0,0,1,0,0,1,0,0,0,1,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

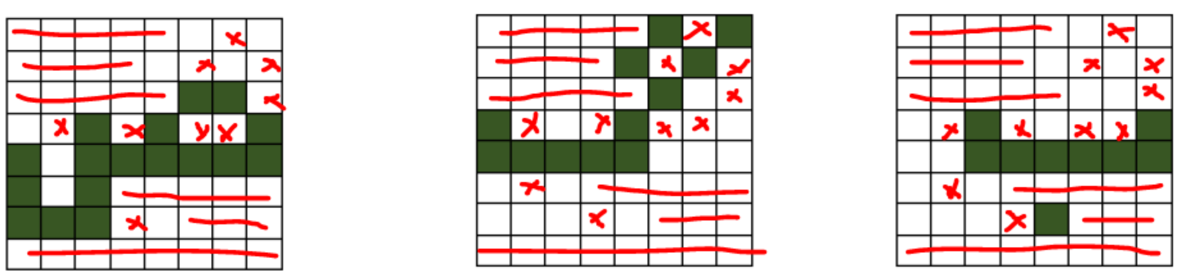
حرف ص:

[0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,1,0,1,0,0,1,1,0,1,1,1,1,1,1,1,0,1,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,
0,0,0,0,]

توضیح در باره ی معماری شبکه:

با دو رویکرد میتوان به مساله نگاه کرد:

۱. در صورتی که مطمئن باشیم هیچ گونه نویزی روی ورودی های ما نخواهد بود و مساله ی پیچیدگی مدل برای ما اهمیت دارد دست به کاهش ابعاد ورودی میزنیم. پیسکل هایی که در تمامی سه عکس ثابت هستند را از ورودی ها حذف میکنیم. که به صورت زیر خواهد بود. پیکسل هایی که با رنگ قرمز علامت زده شده اند از ورودی های ما حذف خواهند شد:



که در مجموع ۲۴ ویژگی باقی خواهد ماند که نسبت به ۶۴ ویژگی اصلی کاهش چشمگیری دارد.

معماری پیشنهادی شبکه در این حالت:

برای این حالت یک پرسپترون با ۲۴ نورون در لایه ورودی و ۳ نورون در لایه خروجی و بدون لایه میانی پیشنهاد میشود. با توجه به وضعیت مساله و پیچیدگی کم آن نیازی به اضافه کردن لایه های میانی نیست و برای تفکیک سه الگوی داده شده کافی است.

۲. در صورتی که امکان وقوع نویز داشته باشیم و باید قابلیت تحمل نویز در مدل ایجاد کنیم و پیچیدگی مدل در حدی نیست که هزینه زیادی برای ما داشته باشد، تعداد ابعاد ورودی را کاهش نمیدهیم.

معماری پیشنهادی شبکه در این حالت:
برای این حالت یک پرسپترون با ۶۴ نورون در لایه ورودی و ۳ نورون در لایه خروجی پیشنهاد میشود. با توجه به وضعیت مساله و پیچیدگی کم آن نیازی به اضافه کردن لایه های میانی نیست و برای تفکیک سه الگوی داده شده کافی است.

تست معماری برای حالت با ۶۴ فیچر در فایل نوت بوک Problem3 - architecture test آورده شده است که نشان میدهد شبکه همگرا میشود و بنابراین برای تعداد فیچر های کمتر یعنی حالت ۲۴ ویژگی نیز همین معماری از پیچیدگی کافی برخوردار است.