به نام خدا

**درس مبانی یادگیری عمیق**

**پاسخنامه تمرین سری دوم**

استاد درس : دکتر مرضیه داوودآبادی

دستیاران : سحرسرکار، فائزه صادقی، حسن حماد

دانشگاه علم و صنعت ایران، دانشکده مهندسی کامپیوتر

نیمسال اول تحصیلی ۱۴۰۲ - ۱۴۰۳

---

## پاسخ های مناسب برای هر سوال، لزوما یکتا نیستند

۱. به سوالات زیر پاسخ دهید. (۱۵ نمره)

الف) مشکل بیش‌برازش[1] و کم‌برازش[2] در شبکه‌های عصبی را توضیح دهید.

بیش‌برازش به معنای این است که مدل ما روی داده های آموزشی بیش از اندازه یاد گرفته باشد. به عبارت دیگر، پارامتر های قابل یادگیری خود را به گونه ای تنظیم کرده که فقط برای داده های آموزشی مناسب باشد و این باعث می شود که مدل ما قابلیت تعمیمش را از دست دهد و برای داده های تست به نتایج مطلوبی نرسیم. به عبارت دیگر، مدل ما به گونه ای تنظیم شده که داده های آموزشی را که نمایشگر داده های عمومی نیستند را یاد گرفته. در بیش‌برازش مدل ما نسبت به نقاط حساسیت بالایی دارد و داده های نویزی را هم به خوبی یاد میگیرد و این باعث میشود که کوچکترین تغییری در ورودی باعث تغییر شدید در خروجی شود. پس دارای واریانس زیاد است. از طرفی چون میانگین خروجی ها به جواب درست نزدیک است پس دارای بایاس کم است.

کم‌برازش به معنای این است که مدل ما قادر به یادگیری کافی از داده‌های آموزشی نیست به عبارت دیگر، مدل قادر نیست رابطه‌ی میان متغیرهای ورودی و خروجی را به طور دقیق ثبت کند. این خطا ایجاد خطای بالایی در هر دو مجموعه‌ی آموزشی و مجموعه‌ی داده‌های (تست) را رقم می‌زند. این امر زمانی رخ می‌دهد که مدل بیش‌ازحد ساده باشد؛ به‌این معنا که مدل به زمان آموزش بیشتر، ویژگی‌های ورودی بیشتر یا تنظیم ($Regularization$) کمتر نیاز دارد. مدل در کم‌برازش نمی‌تواند الگوی غالب را در داده‌ها را تشخیص دهد؛ درنتیجه، این امر افزایش خطا و عملکرد ضعیف مدل را به همراه دارد. یک مدل کم‌برازش دارای بایاس بالا و واریانس کم است.

ب) فرض کنید مدلی داریم که قبلاً آموزش دیده است. چگونه می‌توانیم بیش‌برازش مدل را تشخیص دهیم؟

[1] Overfitting

[2] Underfitting

معیار بیش‌برازش مدل از مقایسه نتایج فرایند های تست و آموزش به دست می آید. در این حالت یک مدلی داریم که قبلا آموزش دیده شده بود. به عبارت دیگر، به اطلاعات و نتایج فرایند آموزش مدل دسترسی نداریم. بنابراین نمی توانیم مشخص بکنیم که مدل بیش‌برازش شده است یا نه. لازم است که حتما اطلاعاتی از فرایند های آموزش و تست مدل داشته باشیم تا بتوانیم در مورد بیش‌برازش مدل قضاوت کنیم.

پ) یکی از راه‌های جلوگیری از بیش‌برازش استفاده از $Dropout$ است. فرض کنید مقادیر یکی از لایه‌های یک شبکه عصبی و ماسک $Dropout$ به صورت زیر باشند. مقادیر نهایی این لایه بعد از اعمال $Dropout$ را در مرحله‌ی آموزش و آزمون محاسبه کنید.

جدول ۱: $Output$

| | | | |
|---|---|---|---|
| ۱.۹ | -۰.۲ | -۰.۷ | ۱.۶ |
| -۰.۹ | ۲.۵ | ۲.۵ | -۲.۳ |
| -۰.۴ | ۳.۷ | ۳.۲ | -۰.۵ |
| ۱.۲ | -۲.۶ | -۰.۴ | ۱.۳ |

جدول ۲: $Dropoutmask$

| | | | |
|---|---|---|---|
| ۱ | ۰ | ۰ | ۱ |
| ۰ | ۱ | ۱ | ۰ |
| ۰ | ۱ | ۱ | ۰ |
| ۱ | ۰ | ۰ | ۱ |

ابتدا مقدار پارامتر $P$ را محاسبه می کنیم:

$$P = \frac{\text{تعداد سلول های که مقدارشان ۱ است}}{\text{تعداد کل سلول ها}} = \frac{8}{16} = 0.5$$

$Dropout$

مرحله آموزش:

$$TrainOutput = output \times dropoutmask$$

مرحله آزمون:

$$TestOutput = output \times P$$

جدول ۳: مرحله آموزش

| | | | |
|---|---|---|---|
| ۱.۹ | ۰ | ۰ | ۱.۶ |
| ۰ | ۲.۵ | ۲.۵ | ۰ |
| ۰ | ۳.۲ | ۳.۷ | ۰ |
| ۱.۲ | ۰ | ۰ | ۱.۳ |

جدول ۴: مرحله آزمون

| | | | |
|---|---|---|---|
| ۱.۹ | -۰.۱ | -۰.۳۵ | ۰.۸ |
| -۰.۴۵ | ۱.۲۵ | ۱.۲۵ | -۱.۱۵ |
| -۰.۲ | ۱.۸۵ | ۱.۶ | -۰.۲۵ |
| ۰.۶ | -۱.۳ | -۰.۲ | ۰.۶۵ |

$InvertedDropout$

مرحله آموزش:

$$TrainOutput = output \times dropoutmask \times \frac{1}{P}$$

مرحله آزمون: هیچ تغییری ایجاد نمی کند.

$$TestOutput = output$$

جدول ۵: مرحله آموزش

| | | | |
|---|---|---|---|
| ۳.۸ | ۰ | ۰ | ۲.۳ |
| ۰ | ۵ | ۵ | ۰ |
| ۰ | ۶.۴ | ۷.۴ | ۰ |
| ۲.۶ | ۰ | ۰ | ۲.۴ |

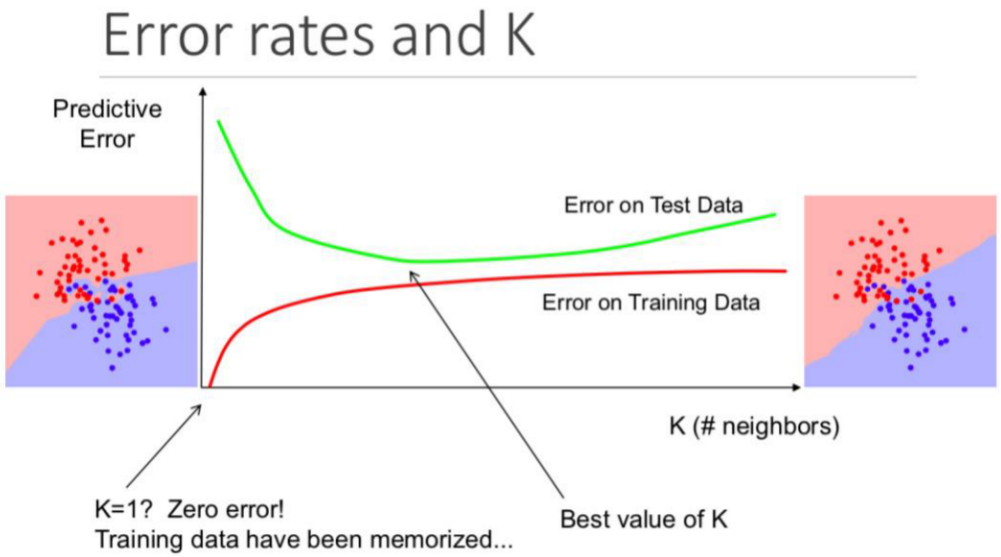
جدول ۶: مرحله آزمون

| | | | |
|---|---|---|---|
| ۱.۹ | -۰.۲ | -۰.۷ | ۱.۶ |
| -۰.۹ | ۲.۵ | ۲.۵ | -۲.۳ |
| -۰.۴ | ۳.۷ | ۳.۲ | -۰.۵ |
| ۱.۲ | -۲.۶ | -۰.۴ | ۱.۳ |

۲. لطفا سوالات زیر را به صورت کامل پاسخ دهید. (۲۰ نمره)

الف) یکی از الگوریتم‌هایی که در حوزه یادگیری ماشین مورد استفاده قرار می‌گیرند، الگوریتم نزدیک‌ترین همسایگی[۳] است. برای مطالعه بیشتر درباره‌ی این الگوریتم می‌توانید به این لینک مراجعه کنید. توضیح دهید که با تغییر مقدار $K$، بایاس و واریانس چه تغییری می‌کنند.

چنانچه $K$ مقدار کمی باشد فرض کنید $K$ برابر ۱ باشد، بنابراین مدل داده آموزشی ما را به طور کامل یاد خواهد گرفت و این احتمال $overfit$ شدن را بالا می‌برد که یعنی بایاس کم و واریانس زیاد ($highvariance$). چنانچه مقدار $k$ بسیار بالا باشد، در واقع در داده آموزشی با خطا روبرو هستیم (بایاس زیاد) و بعد از مدتی کاهش در خطای تست (کم شدن واریانس) بعد از حدی از $k$ دوباره خطای تست افزایش میابد (واریانس زیاد می‌شود).




ب) درست یا غلط بودن گزاره‌های زیر را مشخص کنید و دلیل پاسخ خود را نیز بیان کنید.

- استفاده از منظم‌سازی، ممکن است باعث تضعیف عملکرد مدل شود.

بله، اگر مقدار لاندای انتخاب شده کوچک باشد، عملاً باز تاثیر بهینه‌سازی پارامترها در تابع خطا کم است و $overfit$ می‌شویم. همچنین با وزن بیشتر دادن به بهینه‌سازی پارامترها $underfit$ ایجاد می‌شود.

- اضافه کردن تعداد زیاد ویژگی‌های[۴] جدید، باعث جلوگیری از بیش‌برازش می‌شود.

خیر، افزایش بیش از حد ویژگی باعث گسترده شدن و پراکندگی دادگان می‌شود که مدل نیز برای یادگیری آن‌ها پیچیده‌تر خواهد شد و این باعث به وجود آمدن $overfit$ می‌شود.

[3] K-Nearest Neighbors (KNN)

[4] Features

- با زیاد کردن ضریب منظم‌سازی، احتمال بیش‌برازش بیشتر می‌شود.

خیر، زیاد کردن ضریب، وزن بهینه‌سازی پارامترها را بیشتر می‌کند که این باعث جلوگیری از $over fitting$ می‌شود. هر چند این مقوله می‌تواند منجر به $under fitting$ شود.

پ) فرض کنید یک مدلی داریم و برای جلوگیری از بیش‌برازش می‌خواهیم از منظم‌سازی $L1$ و $L2$ استفاده کنیم. برای این کار چهار آزمایش اجرا کرده و نتایج به دست آمده به صورت زیر است. با توجه به این نتایج، مشخص کنید در هر آزمایش از کدام منظم‌سازی استفاده شده است (دلیل انتخاب خود را توضیح دهید).

- $W_{exp1} = [0.26, 0.25, 0.25, 0.25]$
- $W_{exp2} = [1, 0, 0, 0]$
- $W_{exp3} = [13.3, 23.5, 53.2, 5.1]$
- $W_{exp4} = [0.5, 1.2, 8.5, 0]$

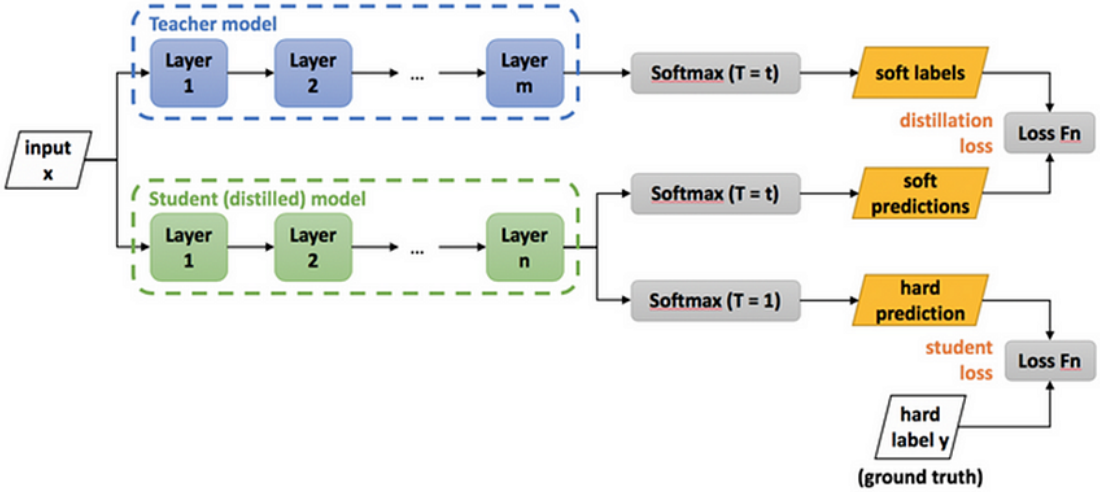
از بین موارد ۳ و ۴، مورد ۴ مربوط به منظم‌سازی است؛ زیرا در منظم‌سازی ضرایب کمتر می‌شوند و در این جا منظم‌سازی $L1$ استفاده شده است. زیرا در این منظم‌سازی ضرایب صفر هم داریم.

۳. به سوالات زیر پاسخ دهید. (۱۵ نمره)

الف) فرایند تقطیر دانش[5] چیست و به چه منظور استفاده می‌شود؟

به فرایند استخراج دانش یادگرفته شده از شبکه های عمیق و انتقال دانش به شبکه با ابعاد کوچکتر تقطیر دانش گویند. این فرایند به منظور کاهش توان محاسباتی مورد نیاز برای اجرای شبکه و پیاده سازی بر روی دستگاه هایی با توان پردازشی کمتر صورت میگیرد.

ب) معماری شکل زیر برای استخراج دانش از شبکه‌ی *teacher* و انتقال آن به شبکه‌ی *student* پیشنهاد شده است. روند یادگیری آن را توضیح دهید.




در این معماری ۲ شبکه وجود دارند که قصد داریم دانش را از یکی به دیگری منتقل کنیم. دانش یادگرفته شده از *teacher* به *student* منتقل خواهد شد. در طول فرایند یادگیری وزن های شبکه آموزگار تغییر نمیکند و ثابت باقی خواهد ماند. آموزش به این صورت است که یک ورودی به هر دو شبکه داده خواهد شد و خروجی برای هر شبکه محاسبه میشود. برای *student* دو تابع خطا در نظر گرفته میشود یکی برای بررسی صحت دسته بندی که می تواند *categorical crossentropy* باشد و دیگری مقایسه شباهت ویژگی های استخراج شده با شبکه *teacher* است. این دو خطا با هم جمع شده و وارد شبکه *student* می شوند. وزن های شبکه با توجه به این دو خطا بروزرسانی خواهند شد.

پ) وزن‌های شبکه‌ی *student* باتوجه به کدام تابع ضرر[6] به‌روزرسانی خواهند شد؟

همانطور که در بخش قبل گفتیم، خطای دسته بندی و خطای شباهت ویژگی های استخراج شده با شبکه *teacher* هم جمع می شوند و وارد *student* می شوند. وزن های شبکه با توجه به این دو خطا بروزرسانی خواهند شد.

[5] Knowledge Distillation

[6] Loos Function

۴. در نوت‌بوک پیوست شده، کد آماده‌ای قرار داده شده که تنها نیاز است سلول‌ها را اجرا کنید و نتایج به دست آمده از بهینه‌سازهای متفاوت را با هم مقایسه کرده و تحلیل نمایید (لطفا تحلیل خود را به علاوه توابع استفاده شده به صورت کامل توضیح دهید). (۱۰ نمره)

:$SGD$

$SGD$ یک الگوریتم پایه است که تنها به جهت گرادیان نگاه می‌کند و نرخ یادگیری پایین در این مثال ۰.۱۰۰ سرعت همگرایی را کند می‌کند. نرخ یادگیری از یک حدی بیشتر در اینجا ۱ باعث واگرایی می‌شود. در این مسئله نرخ یادگیری ۰.۰۱ و ۰.۱ برای $SGD$ مناسب به نظر می‌آید.

:$Momentum$

در $Momentum$ علاوه بر جهت گرادیان تکانه اول نیز محاسبه می‌شود تا در صورت اطمینان از جهت حرکت درست سرعت گام بیشتری بر دارد و می بینیم در نرخ یادگیری ۱ دیگر واگرا نشده است. البته که هنوز نوسانات شدیدی وجود دارد و برای ۰.۱ نیز بسیار خطا بالا و پایین شده است. به نظر نرخ یادگیری ۰.۰۱ از همه مناسب‌تر است.

۵. در نوت‌بوک داده شده، موارد خواسته شده را پیاده‌سازی کنید. در این نوت‌بوک هدف آموزش یک شبکه ساده $MLP$ برای یادگیری مجموعه داده $FashionMNIST$ می‌باشد. (۴۰ نمره)

الف) در این بخش مدل خود را تعریف کنید. تعداد لایه‌ها و تعداد نورون‌های هر لایه بر عهده شماست. برای تابع ضرر از $CrossEntropy$ و برای بهینه‌ساز از $SGD$ استفاده کنید. سپس قسمت آموزش مدل را تکمیل کنید. در قسمت تست نیز خروجی مدل برای چند عکس موجود در داده‌های تست را به دست آورده و با برچسب واقعی مقایسه کنید.

ب) مدل تعریف شده در قسمت الف را تغییر دهید تا شبکه شما دچار بیش‌برازش شود. دلیل بیش‌برازش شبکه در این مرحله را توضیح دهید. همچنین نموداری رسم کنید که میزان خطای حین آموزش و آزمون را با هم مقایسه کند.

پ) حال تلاش کنید فقط با داده‌افزایی[۷]، شبکه بیش‌برازش شده را بهبود دهید. برای مطالعه بیشتر درباره‌ی داده‌افزایی در $PyTorch$ می‌توانید از این لینک راهنمایی بگیرید. حداقل دو مورد از تبدیلات توضیح داده شده را پیاده‌سازی کرده و نتایج حاصل را تحلیل کنید.

ت) با استفاده از منظم‌سازی $L1$ یا $L2$ (به انتخاب خود) شبکه را بهبود دهید و نتایج را تحلیل کنید.

[7] Data Augmentation

ث) (**امتیازی**) با استفاده از ترکیبی از داده‌افزایی، منظم‌سازی و $Dropuot$ شبکه را بهبود و بیان کنید که چه ترکیبی از این‌ها باعث بهبود حداکثری می‌شود. (۱۵ نمره)

با توجه به آنکه پیاده‌سازی های مختلف برای این سوال می تواند درست باشد، پاسخی برای این سوال نمی آوریم