دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

یادگیری ماشین

پاییز ۱۴۰۳

استاد: علی شریفی زارچی
مسئول تمرین: ایزدی

مهلت ارسال امتیازی: ۱۶ آذر | **تمرین چهارم** | مهلت ارسال نهایی: ۲۳ آذر

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روزهای مشخص‌شده است.
- در طول ترم، برای هر تمرین می‌توانید تا ۵ روز تأخیر مجاز داشته باشید و در مجموع حداکثر ۱۵ روز تأخیر مجاز خواهید داشت. توجه داشته باشید که تأخیر در تمرین‌های عملی و تئوری به صورت جداگانه محاسبه می‌شود و مجموع تأخیر هر دو نباید بیشتر از ۱۵ روز شود. پس از اتمام زمان مجاز، دو روز اضافی برای آپلود غیرمجاز در نظر گرفته شده است که در این بازه، به ازای هر ساعت تأخیر، ۲ درصد از نمره نهایی تمرین کسر خواهد شد.
- اگر بخش عملی یا تئوری تمرین را قبل از مهلت ارسال امتیازی آپلود کنید، ۲۰ درصد نمره اضافی به آن بخش تعلق خواهد گرفت و پس از آن، ویدیویی تحت عنوان راهنمایی برای حل تمرین منتشر خواهد شد.
- حتماً تمرین‌ها را بر اساس موارد ذکرشده درک‌شده در صورت سوالات حل کنید. در صورت وجود هرگونه ابهام، آن را در صفحه تمرین در سایت کوئرا مطرح کنید و به پاسخ‌هایی که از سوی دستیار آموزشی مربوطه ارائه می‌شود، توجه کنید.
- در صورت هم‌فکری و یا استفاده از منابع، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال را ذکر کنید.
- فایل پاسخ‌های سوالات نظری را در قالب یک فایل pdf به فرمت HW4_T_[STD_ID].pdf آماده کنید و برای سوالات عملی، هر یک را در یک فایل zip جداگانه قرار دهید. فایل مربوط به نوتبوک i ام را به فرمت HW4_P[i]_[STD_ID].zip نام‌گذاری کرده و هرکدام را به صورت جداگانه آپلود کنید

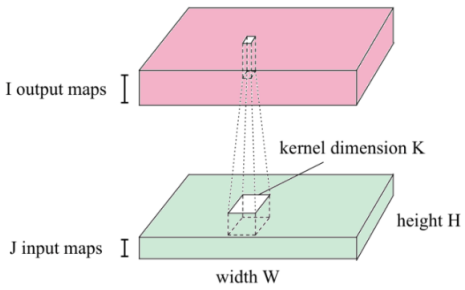
گردآورندگان تمرین: امیرحسین ایزدی، محمدحسین شالچیان، امیرعزتی، علی بختیاری، علی الوندی

سوالات نظری (۱۰۰+۱۰ نمره)

۱. (۲۲ نمره) به سوالات زیر پاسخ دهید.

الف) دو لایه متوالی از شبکه ژرفی را در نظر بگیرید به طوری که لایه ورودی دارای J و لایه خروجی دارای I ویژگی نگاشت باشد. اتصال میان این دو لایه می‌تواند یا به صورت اتصال تمام‌متصل یا به صورت پیچشی باشد. برای هر دو حالت، ویژگی نگاشت لایه ورودی دارای ابعاد $W \times H$ است و کرنل استفاده شده در لایه پیچشی دارای ابعاد $K \times K$ است. نمای کلی این دولایه در شکل ۲ قابل مشاهده هست.

- تعداد واحدهای خروجی، اتصالات و وزن‌های قابل یادگیری را در صورت امکان محاسبه؛ گزارش و با یکدیگر مقایسه کنید.(از بایاس صرف نظر کنید)
- از محاسبات قبلی می‌توان نتیجه گرفت که شبکه‌های عمیق کاملاً پیچشی نسبت به شبکه‌های کاملاً متصل عموماً نیازمند داده آموزشی کمتری برای آموزش هستند؟ آیا این نتیجه‌گیری لزوماً صحیح است و مرتبط است یا خیر؟



شکل۱ : لایه ورودی با رنگ سبز و خروجی با رنگ قرمز مشخص شده است.

ب) شبکه CNN ای را در نظر بگیرید که از بلاک‌هایی به فرم زیر استفاده می‌کند:

$$(\text{ConvLayer}) \rightarrow (\text{BatchNorm}) \rightarrow (\text{Activation})$$

- با مطالعه این مقاله نحوه انجام نرمال سازی بچ در شبکه های تماما متصل و شبکه های پیچشی را با یکدیگر مقایسه کنید
- آیا حذف بایاس b از لایه کانولوشن در کارکرد این شبکه اختلالی ایجاد می‌کند؟ چرا؟
- همچنین فرض کنید شبکه را آموزش داده‌ایم؛ آیا ضرب کردن وزن‌ها در یک عدد مانند α در زمان آزمایش (Inference)، عملکرد شبکه را تغییر می‌دهد؟ ضرب کردن این ضریب در تمام درایه‌های ورودی شبکه چطور؟
- یک سناریو مرتبط به حوزه پردازش تصویر را شرح دهید که در آن نرمال سازی دسته ای یا بچ ممکن است در آن کارایی کمتری داشته باشد و یا حتی نتیجه در معکوس دهد.در چنین مواردی چه تکنیک های جایگزینی می توانند مورد توجه قرار گیرند؟

ج) معماری Unet را در نظر بگیرید:

- تصور کنید که ابعاد تصویر ورودی ما برای این شبکه ۲۵۶ × ۲۵۶ می‌باشد. حال فرض می‌شود که در این معماری هر لایه در انکدر ابعاد را به نصف کاهش می‌دهد و در دیکدر دو برابر می‌کند. در پایین‌ترین لایه (عمیق‌ترین لایه) این معماری، فضای ویژگی ما چند پیکسل خواهد داشت؟
- فرض کنید انکودر دارای لایه‌هایی با ۶۴, ۱۲۸, ۲۵۶ و ۵۱۲ فیلتر است. اگر هر لایه کانولوشن از کرنل‌های ۳ × ۳ استفاده کند، تعداد پارامترهای لایه کانولوشن دوم انکدر را محاسبه کنید.

د) تفسیرپذیری شبکه‌های عصبی از اهمیت بالایی برخوردار است. هنگام دسته‌بندی تصاویر، علاقه‌مندیم بدانیم کدام بخش‌های تصویر در دسته‌بندی، تأثیر بیشتری داشته‌اند. در مقاله ایده‌ی شبکه‌های de-convolutional و در مقاله ایده‌ی شبکه‌های up-convolutional مطرح شده است. با بررسی این مقالات، توضیح دهید هر کدام از دو روش به چه صورت منجر به تفسیرپذیری شبکه کانولوشنی می‌شوند.

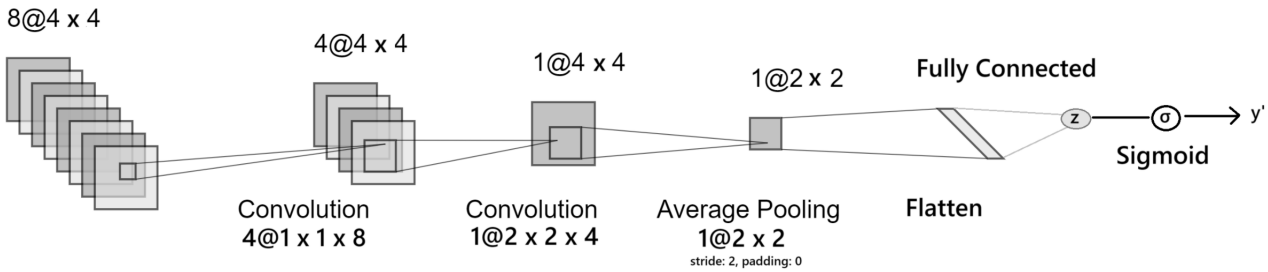
۲. (۸ نمره) فرض کنید برداری به طول N دارید و قصد دارید یک لایه کانولوشن یک بعدی روی آن اعمال کنید. حاصل اعمال یک لایه کانولوشن را از طریق رابطه‌ی زیر به‌دست آورید:

$$Z = W * X \quad \longrightarrow \quad Z_i = \sum_{j=0}^{K-1} W_j X_{i+j}$$

به دست می‌آوریم که K اندازه‌ی فیلتر را نشان می‌دهد. اگر مقدار $\frac{\partial L}{\partial Z_i}$ برای تمامی مقادیر i بدانیم، رابطه‌ی مربوط به $\frac{\partial L}{\partial W_j}$ را به طور دقیق پیدا کنید. نشان دهید این رابطه، عملاً معادل اعمال یک فیلتر کانولوشن است.

۳. (۱۵ نمره)

یکی از اساسی‌ترین اجزا در یک شبکه ژرف، انتشار به عقب یا Backpropagation است. شبکه داده شده در شکل ۲ را در نظر بگیرید. در این شکل، وزن‌های لایه اول با $W^{(1)}$ و وزن‌های لایه دوم با $W^{(2)}$ و وزن‌های لایه تمام‌متصل با $W^{(fc)}$ نمایش داده شده‌اند. همچنین، $P^{(1)}$، $P^{(2)}$ و $P^{(3)}$ به ترتیب خروجی‌های لایه اول، دوم و سوم شبکه هستند.



شکل۲ :معماری شبکه

- ورودی شبکه: یک تنسور با ابعاد ۸ × ۴ × ۴
- لایه اول: لایه کانولوشن با ۴ فیلتر ۸ × ۱ × ۱ که خروجی $P^{(1)}$ با ابعاد ۴ × ۴ × ۴ تولید می‌کند.
- لایه دوم: لایه کانولوشن با ۱ فیلتر ۴ × ۲ × ۲ که خروجی $P^{(2)}$ با ابعاد ۱ × ۴ × ۴ تولید می‌کند.
- لایه سوم: لایه Avg. Pooling با کرنل ۲ × ۲ و گام ۲، که خروجی $P^{(3)}$ با ابعاد ۱ × ۲ × ۲ ایجاد می‌کند.
- لایه چهارم: لایه Flatten که $P^{(3)}$ را به یک بردار $\mathbf{v}$ با ابعاد ۱ × ۴ تبدیل می‌کند.
- لایه پنجم: لایه تمام‌متصل با تابع فعالساز Sigmoid که خروجی نهایی را تولید می‌کند.

وزن‌های فیلترهای کانولوشن در لایه‌های اول و دوم ($W^{(1)}$ و $W^{(2)}$) و وزن‌های لایه تمام‌متصل ($W^{(fc)}$) از جمله پارامترهای قابل‌آموزش شبکه هستند. هدف شبکه، بهینه‌سازی این وزن‌ها برای کمینه‌سازی یک تابع زیان L است.

وظایف:

(آ) با استفاده از قاعده مشتق زنجیره‌ای و بر اساس مشتق $\frac{\partial L}{\partial z}$، عبارت‌های زیر را محاسبه کنید:

i. $\frac{\partial L}{\partial P^{(3)}_{i,j}}$: گرادیان زیان نسبت به خروجی لایه Average Pooling.

ii. $\frac{\partial L}{\partial P^{(1)}_{i,j,k}}$: گرادیان زیان نسبت به خروجی لایه اول کانولوشن.

(ب) گرادیان زیان نسبت به وزن $W^{(1)}_{1,1,k}$ یکی از فیلترهای لایه اول کانولوشن را محاسبه کنید:

$$\frac{\partial L}{\partial W^{(1)}_{1,1,k}}$$

(ج) عبارت گرادیان $\frac{\partial L}{\partial W^{(2)}_{i,j,k}}$، یکی از وزن‌های فیلتر کانولوشن لایه دوم را به‌دست آورید.

۴. (۲۲ نمره) در این سوال قصد داریم برخی گونه‌های مختلف شبکه‌های پیچشی و همچنین معماری‌های مبتنی بر CNN معرفی شده در دهه اخیر را مورد بررسی و تحلیل قرار دهیم.

الف) شبکه DenseNet

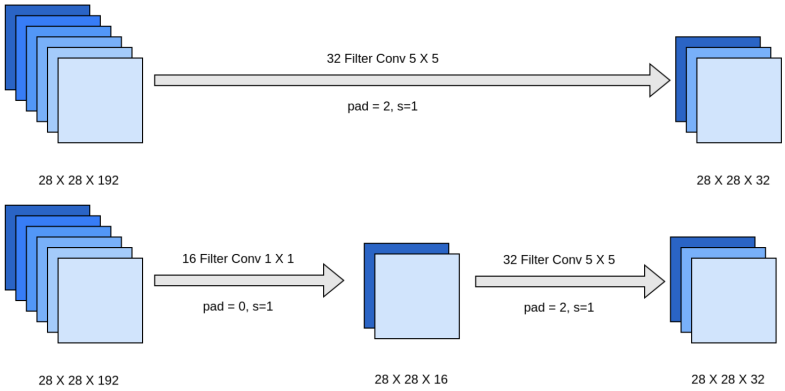
- تفاوت اصلی ResNet's residual connections و DenseNet's dense connections را بیان کنید. در مورد هر کدام نیز، ابتدا توضیح مختصری بدهید.
- توضیح دهید که DenseNet چگونه مشکل vanishing gradient را کاهش می‌دهد و مزیت محاسباتی آن چیست؟

- با در نظر گرفتن نرخ رشد k در DenseNet، اگر هر لایه k ویژگی نگاشت جدید تولید کند و ورودی یک dense block دارای ۳۲ کانال باشد، اگر $k = ۲۴$ باشد، لایه سوم در بلوک چند کانال خروجی خواهد داشت؟

ب) شبکه GoogleNet

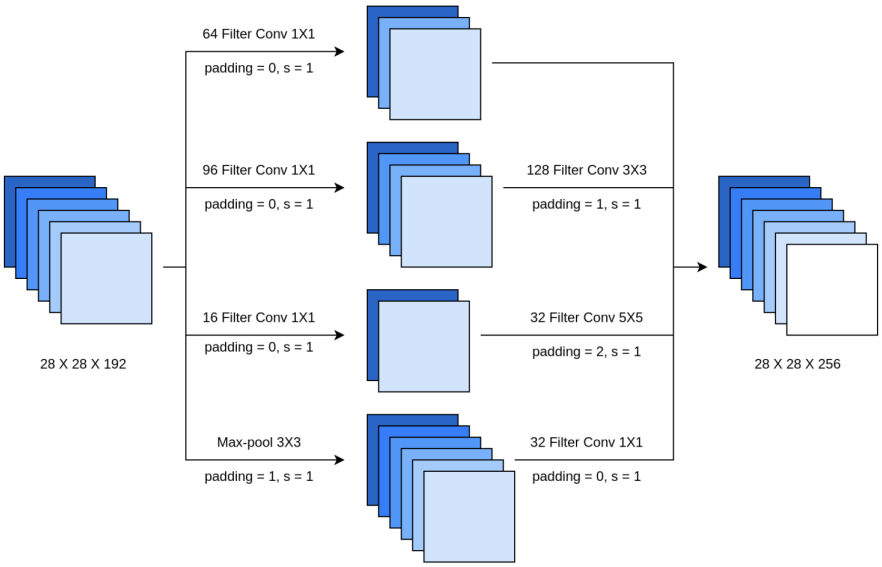
پیمانه پیدایش با هدف کاهش پیچیدگی محاسباتی به وجود آمد. قصد داریم ابتدا با پیمانه پیدایش و سپس شبکه GoogleNet آشنا شویم.

- در شکل ۳ یک لایه کانولوشن به دو صورت نشان داده شده است. ورودی هر دو حالت $۲۸ \times ۲۸ \times ۱۹۲$ و خروجی هر دو $۲۸ \times ۲۸ \times ۳۲$ است. با محاسبه تعداد کل عملیات‌های انجام شده در هر حالت، پیچیدگی محاسباتی دو حالت نشان داده شده را مقایسه کنید. مشخص کنید که افزودن فیلتر کانولوشن ۱×۱، پیچیدگی محاسباتی را چند درصد کاهش یا افزایش داده است؟



شکل۳ :لایه کانولوشن معمولی (بالا) و لایه کانولوشن با فیلتر ۱×۱ (پایین)

- در شکل ۴ یک پیمانه پیدایش به صورت کامل نشان داده شده است. ترکیب متوالی این پیمانه‌ها با هم، شبکه GoogleNet را تشکیل می‌دهند. توضیح دهید که دلیل استفاده از فیلترهایی با اندازه متفاوت (برای مثال ۱×۱، ۳×۳ و ۵×۵) و حتی فیلترهای با انواع متفاوت (استفاده از Max pooling در کنار کانولوشن) چیست؟ تعداد کل محاسبات در این شبکه را بدست آورید.



شکل۴ : شمای کلی پیمانه پیدایش

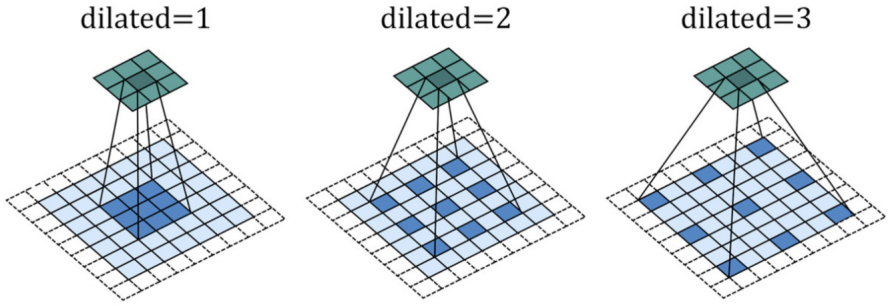
- آیا این معماری نسبت به vanishing gradient مقاوم است؟ همچنین توضیح دهید که auxiliary classifier موجود در این معماری چگونه به بهبود جریان گرادیان و پایداری بهینه‌سازی در شبکه

کمک می‌کند. در نهایت، محدودیت‌های احتمالی این طراحی در کاربردهای مدرن یادگیری عمیق را ارزیابی کنید.

ج) شبکه‌های کانولوشنی Deformable

- تفاوت شبکه‌های کانولوشنی عادی و شبکه‌های کانولوشنی Deformable را از نظر grid sampling مقایسه کنید. همچنین شبکه‌های Deformable چگونه می‌توانند نسبت به Geometric transformation در تصاویر انعطاف‌پذیر باشند؟
- مفهوم offset در این شبکه‌ها به چه معناست و چگونه محاسبه می‌شود؟

۵. (۲۱ نمره) در شبکه های پیچشی به صورت متداول از لایه های کانولوشن ساده استفاده می شود که با آن آشنا هستید. نوع دیگری از لایه ها که می توان از آنان ددر شبکه های پیچشی استفاده نمود، لایه کانولوشن گسترش یافته یا متسع است. در شکل ۵ تصویر شهودی از فیلتر کانولوشن گسترش یافته ارائه شده است، این فیلتر ها میان خانه هایی که فیلتر با استفاده از اطلاعات آن‌ها لایه بعد را محاسبه می‌کنند فاصله می‌اندازند یا به بیانی دیگر در زمان اعمال فیلتر و انجام عملیات ضرب کانولوشن، بر روی ورودی با گام (dilated) بزرگتری حرکت می کنیم، توجه کنید طول گام مفهومی متفاوت نسبت به طول گام (stride) در لایه های شبکه کانولوشن دارد.



شکل۵ : شهودی از کانولوشن گسترش یافته با گام های متفاوت

همانطور که در شکل ۱ نیز مشخص است این روش، یک روش کم هزینه برای افزایش محدوده دید شبکه های پیچشی است. کانولوشن گسترش یافته بصورت فرم بسته ریاضی زیر تعریف می‌شود.

$$(K \star_D I)(i,j) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} K(m,n) I(i + Dm, j + Dn)$$

الف) برای ورودی $I \in \mathbb{R}^{M \times N}$ و کرنل $K \in \mathbb{R}^{F \times F}$ نشان دهید خروجی عملگر کانولوشن گسترش یافته دارای ابعاد $(M - DF + D) \times (N - DF + D)$ است.

ب) نشان دهید کانولوشن گسترش یافته معادل کانولوشن با کرنل متسع شده K' است. ماتریس A را مشخص کنید. ($K' = K \otimes A$، عملگر $\otimes$ کرونکر محصول `kronecker product` است.)

ج) فرض کنید یک لایه ورودی با ابعاد $M \times N$ به یک شبکه عصبی با سه لایه کانولوشن گسترش یافته داده می‌شود. در این شبکه، هر لایه کانولوشن شامل تعدادی فیلتر است که بر روی قسمت‌های مختلف ورودی حرکت می‌کند و ویژگی‌های گوناگون آن را استخراج می‌کند. هدف ما بررسی تعداد پیکسل‌های قابل مشاهده توسط یک عنصر خاص مانند (i,j) از خروجی است. به عبارت دیگر، محدوده‌ای از ورودی که این عنصر را تحت تأثیر قرار می‌دهد را در صورت وجود امکان محاسبه به صورت پارامتری بیابید.

ابعاد فیلترها به ترتیب از چپ به راست برابر هستند با:

$$(w-۱)\times(w-۱),\quad w\times w,\quad (w+۱)\times(w+۱)$$

پارامتر گسترش این فیلترها به ترتیب از چپ به راست برابر است با:

$$d-۱,\quad d,\quad d+۱$$

د) در مورد Masked Convolution، کاربرد و محدودیت های آن تحقیق کنید و ضمن توضیح مختصری دراین باره پاسخ دهید چگونه می توان با استفاده از کانولوشن گسترش یافته محدودیت Masked Convolution را بهبود بخشید؟

۶. (۲۲ نمره) در این سوال به بررسی پدیده vanishing gradient یا "گرادیان ناپدید شونده" می‌پردازیم.

الف)(امتیازی) پیش از ابداع روش‌های مبتنی بر شبکه‌های عصبی کانولوشنی، feedforward neural network ها قادر به استخراج ویژگی‌ها از تصاویر به‌طور مستقل نبودند و معمولاً استخراج ویژگی‌ها با روش‌هایی غیر از یادگیری عمیق انجام می‌شد. از طرفی، می‌دانیم که عمیق‌تر کردن شبکه عصبی می‌تواند باعث شود مدل ویژگی‌های پیچیده‌تری را بیاموزد، اما عمیق‌تر کردن شبکه بدون استفاده از تکنیک‌های خاص مشکلاتی نیز به وجود می‌آورد، یکی از مشکلات اصلی که در این فرآیند بروز می‌کند، پدیده vanishing gradient است.
فرض کنید یک شبکه عصبی با تعداد زیادی لایه داریم. می‌دانیم که گرادیان وزن‌های لایه i از طریق عبارت زیر محاسبه می‌شود (که در آن، δ_L مشتق تابع ضرر نسبت به خروجی لایه آخر، W_k وزن لایه k، f تابع فعال‌سازی و z ورودی تابع فعال‌سازی است.)

$$\frac{\partial J}{\partial W^i}=(\delta^i)^T\times\frac{\partial z^i}{\partial W^i}=(\delta^i)^T\times a^{i-۱}$$

$$\delta_i=\delta_L\left(\prod_{k=i+۱}^{L}W_k\cdot f'_{k-۱}(z_{k-۱})\right)$$

فرض کنید بزرگ‌ترین singular value (مقدار تکین) برای همه ماتریس‌های وزن W_k کوچک‌تر از یک باشد، با فرض این‌که $\delta_L=M$ و تابع فعال‌سازی ما از نوع ReLU باشد، در این صورت نشان دهید که اگر $L\to\infty$ در این صورت، $|\delta_i|\to ۰$.
همچنین فرض کنید بزرگترین singular value همه ماتریس های وزن برابر ۰/۹ باشد، برای $L=۱۰۰$ با شرایطی که بیان شد یک حد بالا برای $|\delta_i|$ در صورتی که $i=۰$ باشد؛ پیدا کنید.

ب) یکی از راهکارها برای جلوگیری از vanishing gradient استفاده از یک initialization مناسب است. فرض کنید یک لایه کانولوشن با طول و عرض ۸×۸ داریم که یک لایه نهفته با تعداد کانال ۳ را به یک لایه نهفته با تعداد کانال ۵ متصل می‌کند. اگر از Kaiming initialization با توزیع نرمال استفاده کنیم، پارامترهای توزیع نرمال را برای مقداردهی اولیه کرنل‌های این لایه به دست آورید.
راهنمایی: تعداد واحدهای ورودی به هر کرنل، حاصل‌ضرب تعداد کانال‌های ورودی در مساحت کرنل است.

ج) در اوایل توسعه‌ی شبکه‌های عصبی، تابع فعال‌سازی Sigmoid برای بسیاری از شبکه‌ها انتخاب می‌شد. این تابع، که مقادیر ورودی را بین صفر و یک نگاشت می‌کند، مناسب به نظر می‌رسید. اما به مرور زمان و با افزایش

عمق شبکه‌ها، معلوم شد که استفاده از تابع Sigmoid نیز می‌تواند باعث بروز پدیده vanishing gradient شود، یکی از دلایلی که معماری‌های Modern Convolutional Neural Network در ابتدا نمی‌توانستند عملکرد کافی داشته باشند عدم امکان عمیق‌سازی آن‌ها به اندازه کافی بود.

در سال ۲۰۱۲ معماری AlexNet معرفی شد که نسبت به معماری مشابه قبلی خود LeNet پیشرفت بسیار قابل توجهی داشت، یکی از تفاوت‌های این معماری عمق بیشتر و استفاده از تابع فعال‌سازی ReLU به جای Sigmoid بود.

با توجه به این موضوع، پاسخ دهید: چرا استفاده از تابع ReLU به جای Sigmoid به جلوگیری از مشکل vanishing gradient کمک می‌کند؟

د) اگرچه ReLU مزایای زیادی ارائه می‌دهد، یکی از معایب آن مسئله‌ی Dead Neurons است، توضیح دهید این مشکل چیست. Leaky ReLU ، Parametric ReLU و ELU توابعی هستند که برای رفع مشکلات تابع ReLU طراحی شده اند، با استفاده از معادله و معادله مشتق این توابع بیان کنید هرکدام چه معایب و مزایایی نسبت به تابع ReLU دارند.

ه) یک شبکه با ۲ بلاک residual را در نظر بگیرید. نشان دهید چگونه این بلاک‌ها در شبکه residual کمک می‌کند تا از مشکل vanishing gradient جلوگیری شود؟ این شبکه را با شبکه‌های نرمال بدون residual blocks مقایسه کنید. برای سادگی در نوشتار، تابعی که در هر بلاک روی ورودی اعمال می‌شود را با F نشان دهید که شامل activation function هم می‌شود.