**به نام آنکه جان را فکرت آموخت**

**دانشکده مهندسی برق – دانشگاه صنعتی شریف**

آزمون میان ترم درس یادگیری عمیق

زمان: ۱۲۰+۱۵

نام و نام خانوادگی:

شماره دانشجویی:

جواب هر سوال می تواند یک، دو، سه، یا تمامی گزینه ها باشد.

آزمون نمره منفی ندارد ولی اگر یک گزینه درست باشد و شما دو گزینه (شامل گزینه درست) را انتخاب کنید، نمره تعلق نمی گیرد.

اگر در سوالی شرح خواسته شده است، در همان فضا شرح دهید.

**برای پاسخ به ۱۰ سوال اول همین فایل را تکمیل و آپلود کنید – اگر می توانید تبدیل به pdf کنید.**

**برای پاسخ به سه مسئله آخر از برگه سفید استفاده و پاسخ خود را اسکن کنید.**

**سوال اول:** می دانیم لایه Max-pooling از میان تعدادی سلول، سلول با حداکثر فعالیت را انتخاب، و مابقی را حذف می کند (مثلاً اگر ساختار ۲×۲ را انتخاب کنیم، در لایه بعدی اندازه از هر طرف نصف می شود). آیا برای اجرای الگوریتم Error Back Propagation نیاز داریم، اندیس سلول برنده (با حداکثر فعالیت) را ذخیره کنیم؟

☒بله

☐خیر

دلیل (یک خط): برای اصلاح ضرایب ماقبل و ادامه مسیر الگوریتم پس انتشار خطا، نیار به دانستن اندیس سلول برنده داریم.

**سوال دوم:** یک شبکه CNN تک لایه داریم، که ورودی آن یک تصویر است، با اعمال کدام گزینه خروجی عوض **می شود.**

☒دوران دادن تصویر

☒قرینه آینه ای تصویر

☒بزرگ کردن تصویر از دو طرف

☐انتقال تصویر

دلیل (یک خط): ضرایب لایه کانولوشنی در هر لایه ثابت هستند و از آنجاییکه که فقط یک لایه داریم. با انتقال مشکلی نخواهیم داشت.

**سوال سوم:** الگوریتم Stochastic Gradient Descent با ایده Minibatch با احتمال **کمتری** نسبت به Full Batch Gradient Descent در نقاط زینی (Saddle point) گیر می کند.

☒صحیح

☐غلط

دلیل (یک خط): روش SGD ایجاد Randomness بالایی می کند و به تغییرات تصادفی و گیرنیفتادن در نقاط زینی و حداقل های محلی کمک فراوانی می کند.

**سوال چهارم:** ورودی یک شبکه CNN یک تصویر ۵ کاناله ۵ با ابعاد ۵×۱۰۰۰×۱۰۰۰ است. اگر ۱۰ فیلتر به سایز ۷×۷ داشته باشیم، تعداد پارامترهای این لایه را حساب کنید.

پاسخ و محاسبه:

$$\underbrace{(7\times 7)}_{Kernel}\times \underbrace{(5)}_{Channels}\times \underbrace{(10)}_{filters}+\underbrace{(10)}_{Bias/filter}=2460$$

**سوال پنجم:** یک مهندس ادعا می کند بجای اینکه در خروجی هر لایه یک شبکه CNN با چند لایه یک تابع ReLu قرار دهیم، بین لایه ها تابع فعالیت خطی قرار داده و در انتهای شبکه و برای لایه آخر CNN یک تابع فعالیت با میزان غیر خطی بودن بالاتر نسبت به ReLu قرار دهیم. آیا با بکار بردن این ایده به شبکه بهتری (به لحاظ قابلیت و توانایی بیشتر یا مساوی شبکه اول) دست پیدا می کنیم.؟

☐صحیح

☒غلط

دلیل (یک خط): در خروجی هر لایه شبکه CNN یک پیچیدگی ReLu داریم که توالی آنها به پیچیدگی زیادی منجر می شود، ولی در روش پیشنهادی مهندس مسئله ما توالی لایه های خطی، هیچ اثری ندارد. وتابع فعالیت غیزخطی تر، نمی تواند عملکرد تعداد بالای پارامتر نگاشت غیرخطی متعارف را جبران نماید.

**سوال ششم:** فاصله بین دقت یک شبکه پایه برای داده های آموزشی (train) و داده های آزمایشی (test) بالا است کدام سازوکار (**تنها یک پاسخ**) می تواند مشکل را برطرف کند. **گزینه ای را پاسخ دهید که به عنوان یک متخصص اول سراغ آن می روید.**

☐تغییر تابع فعالیت از ReLu به tanh

☐استفاده از روش بهینه سازی دیگری

☒استفاده از Dropout

☐استفاده از لایه Batch Normalization

دلیل (یک خط): این روش نوعی Regularization است که با ایجاد Randomness بالا نوعی می تواند جلوی بیش برازش را بگیرد.

**سوال هفتم:**کدام ابزار می تواند برای مقابله با overfitting بکار رود

☐Dropout

☐Early Stopping

☐ Data Augmentation

☒ هر سه

دلیل (سه خط): هر سه روش نوعی Regularization هستند که برای مقابله با بیش برازش بکار می روند.

**سوال هشتم:** کدام تابع فعالیت می تواند منجر به Gradient Vanishing بشود.

☒ sigmoid

☒ tanh

☐ Leaky ReLu

☐ ReLU

☐ $log(1+e^x)$

دلیل (یک خط): توابع sigmoid و tanh اشباع شوند (مشتق نزدیک صفر) هستند ولی سه تابع بعدی رفتار همانند ReLu دارند.

**سوال نهم:** کدام گزینه در مورد لایه Batch Normalization درست است:

☐ اثری همانند Dropout دارد.

☐ با تبدیلی غیر خطی متوسط داده ها را صفر و واریانس انها را یک می کند

☒ باعث افزایش سرعت یادگیری می شود.

☐ در بخشی از آن متوسط گیری روی ابعاد بردار ویژگی هر نمونه ورودی انجام می شود.

دلیل (یک خط): BN لایه خطی است، متوسط گیری روی نمونه ها است و نه ابعاد ویژگی، شباهتی به DO ندارد.

**سوال دهم:** کدام روش Regularization منجر به تنک شدن ضرایب شبکه می شود

☐ نرم ۲ یا همان L2

☐ Dropout

☒ نرم یک یا همان L1

☐ توقف زود هنگام در ترکیب با Dropout

دلیل (یک خط): در کلاس بصورت تحلیل ثابت شده است.

**مسئله اول)** تابع خطا در یک شبکه با اعمال Dropout گوسی-ضربی به شرح زیر است.

$$J_1 = 0.5\left(y_d - \sum_{k=1}^{n} \delta_k w_k x_k\right)^2$$

که در آن توزیع $\delta_k \sim Normal\left(1,\sigma^2\right)$ می باشد، مقدار امید ریاضی گرادیان تابع هدف نسبت به متغیر $w_i$ را نوشته و تا حد ممکن ساده کنید. **(۲۰)**

$$E\left\{\frac{\partial J_1}{\partial w_i}\right\}$$

$$J_1 = 0.5\left(y_d - \sum_{k=1}^{n} \delta_k w_k x_k\right)^2 \rightarrow \frac{\partial J_1}{\partial w_i} = -\delta_i x_i \left(y_d - \sum_{k=1}^{n} \delta_k w_k x_k\right) = -\delta_i x_i y_d + \sum_{k=1}^{n} \delta_k \delta_i w_k x_k x_i$$

$$E\left\{\frac{\partial J_1}{\partial w_i}\right\} = -x_i y_d + \left(1^2 + \sigma^2\right) w_i x_i^2 + \sum_{k \neq i}^{n} 1 \times 1 \times w_k x_k x_i = -x_i y_d + \sigma^2 w_i x_i^2 + \sum_{k=1}^{n} w_k x_k x_i$$

آیا می توانید تعبیری از Regularization با استفاده از این نوع Dropout ارایه هید؟ در صورت امکان تابع هدف Non-Regularized را هم معرفی کنید. **(۱۰)**

$$J_2 = 0.5\left(y_d - \sum_{k=1}^{n} w_k x_k\right)^2 + 0.5\sigma^2 \sum_{k=1}^{n} w_i^2 x_i^2$$

**مسئله دوم)** روابط حاکم بر یک شبکه به شرح زیر است، به موارد خواسته شده پاسخ دهید:

| | |
|---|---|
| الف) در این شبکه از چه روش ویژه ای (نسبت به ساختار پایه AlexNet) استفاده شده است.<br>ب) مشتقات زیر را محاسبه کنید: **(۴۰)**<br>$\frac{\partial J}{\partial W_1}, \quad \frac{\partial J}{\partial W_2}$<br>CE=Cross Entropy | $\mathbf{z_1} = \mathbf{W}_1\mathbf{x} + \mathbf{b_1}$<br>$\mathbf{h_1} = \mathrm{ReLU}(\mathbf{z_1})$<br>$\mathbf{d} = \mathbf{h_1} + \mathbf{x}$<br>$\mathbf{z_2} = \mathbf{W}_2\mathbf{d} + \mathbf{b_2}$<br>$\mathbf{h_2} = \mathrm{ReLU}(\mathbf{z_2})$<br>$\theta = \mathbf{h_2} + \mathbf{d}$<br>$\hat{\mathbf{y}} = \mathrm{softmax}(\theta)$<br>$J = \mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}})$ |

$$J = -\sum_i y_i \log \hat{y}_i = -\sum_i y_i \log \frac{e^{\theta_i}}{\sum_n e^{\theta_n}} = -\sum_i y_i \left(\theta_i - \log \sum_n e^{\theta_n}\right) = -\sum_i y_i \theta_i + \left(\log \sum_n e^{\theta_n}\right) \sum_i (y_i)$$

$$\frac{\partial J}{\partial \theta_k} = -y_k + \frac{e^{\theta_k}}{\sum_n e^{\theta_n}} \underbrace{\sum_i (y_i)}_{=1} = \hat{y}_k - y_k$$

$$\sigma_1 = \frac{\partial J}{\partial \theta} = \hat{y} - y$$

$$\sigma_2 = \frac{\partial J}{\partial z_2} = \frac{\partial J}{\partial \theta}\frac{\partial \theta}{\partial h_2}\frac{\partial h_2}{\partial z_2} = \sigma_1 \otimes u(z_2), \quad u(\alpha) = \begin{cases} 1 & \alpha \geq 0 \\ 0 & \alpha < 0 \end{cases}$$

$$\sigma_3 = \frac{\partial J}{\partial d} = \frac{\partial J}{\partial z_2}\frac{\partial z_2}{\partial d} + \frac{\partial J}{\partial \theta}\frac{\partial \theta}{\partial d} = W_2^T \sigma_2 + \sigma_1$$

$$\frac{\partial J}{\partial W_1} = \frac{\partial J}{\partial h_1}\frac{\partial h_1}{\partial W_1} = \frac{\partial J}{\partial d}\frac{\partial d}{\partial h_1}\frac{\partial h_1}{\partial W_1} = x^T \sigma_3 \otimes u(z_1)$$

$$\frac{\partial J}{\partial W_2} = \frac{\partial J}{\partial z_2}\frac{\partial z_2}{\partial W_2} = d^T \sigma_1 \otimes u(z_2)$$

**مسئله سوم)** در یک محیط محاسباتی فقط تابع فعالیت ReLu تعریف شده است؛ اما در لایه آخر (تمام اتصال) نیاز به تابع فعالیت tanh داریم،

الف) پیشنهاد شده است که با استفاده از تابع ReLu تابع tanh را با دقت مناسب ایجاد و استفاده کنیم. این کار چگونه ممکن است؟ **(۱۰)**

**تخمین تابع tanh با استفاده از تابع ReLu (از طریق پیش آموزش یک شبکه عصبی)**

ب) یک شبکه خیلی ساده با شباهت نسبی به این منظور طراحی کنید (لازم است وزنها و بایاسها و ... را بطور عددی مشخص کنید). منظور یادگیری وزنها نمی باشد، بلکه باید مقادیری برای وزنها و بایاسها پیشنهاد دهید. **(۲۰)**

**تابع tanh(x) در 3- حدوداً برابر ۱- و در ۳ حدوداً برابر یک است و مشتق آن در مبدا برابر یک است. بسته به نوع تقریب (حفظ مشتق در مبدا) یا مقدا در کرانهای اشباع می توان آن را با تابع خطی/متقارن/اشباع شونده به شرح زیر تخمین زد. (a=1 or 3)**

$$\tanh(x) \simeq \begin{cases} -1 & x < -a \\ \frac{1}{a}x & -a \leq x \leq a = -1 + \frac{1}{a}\mathrm{ReLu}(x+a) - \frac{1}{a}\mathrm{ReLu}(x-a) \\ 1 & x > a \end{cases}$$

**مشخصات شبکه:**

**یک ورودی**

**یک لایه مخفی حاوی دو نرون با تابع فعالیت ReLu**

**یک خروجی با تابع فعالیت خطی**