دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

مقدمه‌ای بر یادگیری ماشین

پاییز ۱۴۰۱

اساتید:علی شریفی، بهروز آذرخلیلی

پایانترم | تاریخ برگزاری: ۴ بهمن

سوالات (۱۲۵ نمره)

۱. (۳۰ نمره) به سوالات زیر کوتاه پاسخ دهید:

- یک شبکه عصبی fully connected را در نظر بگیرید که تابع فعالسازی تمام لایه‌ها تابع sigmoid می‌باشد. برای مقداردهی اولیه وزن‌ها، همه وزن‌های شبکه را مقادیری بزرگ انتخاب می‌کنیم. آیا این ایده خوبی است؟ استفاده از این مقداردهی اولیه موجب چه پدیده‌ای می‌شود؟
- شما در حال طراحی یک سیستم یادگیری عمیق برای تشخیص سرطان سینه با استفاده از تصاویر X-ray هستید به نظر شما مناسب ترین معیار ارزیابی در این مدل چه چیزی می تواند باشد و چرا: Accuracy, Precision, Recall, F1 score.
- شما در حال طراحی یک شبکه CNN برای استفاده در یک تسک بینایی ماشین با استفاده از ماژول‌های زیر هستید:

  Layer Input → Conv. Layer → Batch Norm. → Activation → Next Layer Input

  همانطور که از درس به یاد دارید، هر لایه کانولوشن مجموعه‌ای از وزن‌ها و بایاس‌های قابل یادگیری دارد. یکی از دوستان شما به شما پیشنهاد می‌دهد که در شبکه خود بایاس‌ها را یاد نگیرید و مقدار آن‌ها را برای همیشه صفر قرار دهید. آیا عملکرد مدل در صورت استفاده از توصیه دوستتان تغییر خواهد نمود؟
- شما در حال طراحی یک مدل برای یک تسک طبقه‌بندی (classification) هستید. در ابتدا مدل خود را بر روی ۲۰ نمونه آموزش می‌دهید و مشاهده می‌کنید که با وجود همگرا شدن آموزش، خطای آموزش بر روی این نمونه‌ها زیاد است. پس در ادامه تصمیم می‌گیرید که شبکه خود را این‌بار روی ۱۰۰۰۰ نمونه آموزش دهید. آیا روش شما برای حل این مشکل صحیح است؟ اگر بلی، محتمل‌ترین نتایج مدل خود را در این حالت توضیح دهید. اگر خیر، راه‌حلی برای رفع این مشکل بیان کنید.
- هدف استفاده از کانولوشن ۱*۱ چیست؟
- به چه دلیل scale کردن (γ) و شیفت دادن (β)معمولا پس از نرمالیزه کردن استاندارد در لایه batch normalization استفاده می‌شود؟

۲. (۱۵ نمره) همانطور که می‌دانید، تابع هزینه الگوریتم خوشه‌بندی k-means با k خوشه(cluster) به صورت زیر می‌باشد:

$$L = \sum_{j=1}^{k} \sum_{x_i \in S_j} \|x_i - \mu_j\|^2$$

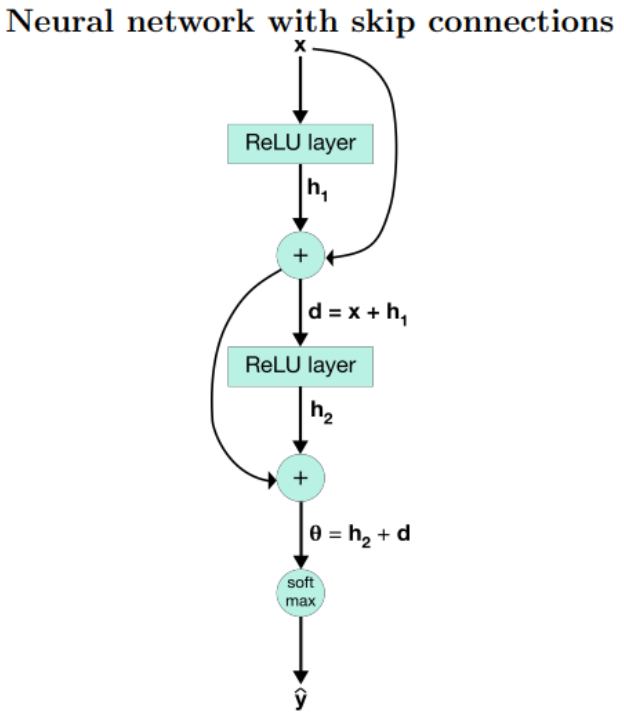
که در آن $x_1, x_2, \dots x_n$ نمونه‌ها و $\mu_1, \mu_2, \dots \mu_n$ مراکز خوشه‌ها می‌باشند. منظور از S_j نیز مجموعه‌ای از نمونه‌هاست که به مرکز μ_j نزدیکتر از مرکز هر خوشه دیگه می‌باشند.

الف) مرحله‌ای از الگوریتم را در نظر بگیرید که برچسب داده‌ها y_j ثابت است و میانگین هر خوشه μ_i آپدیت می‌شود. نشان دهید برای کمینه کردن تابع هزینه در این مرحله، کافی است میانگین هر خوشه را به عنوان مرکز آن خوشه قرار دهیم.

ب) حال حالتی را در نظر بگیرید که به جای آپدیت کردن مرکز کلاستر با میانگین اعضای آن، از batch gradient descent استفاده کنیم. ضابطه آپدیت کردن مرکز کلاستر اول یعنی μ_1 را بدست آورید. فرض کنید learning rate برابر با ϵ می‌باشد.

ج) در این قسمت قصد داریم که ارتباط بین ضابطه بدست آمده برای الگوریتم batch gradient descent را با الگوریتم استاندارد k-means بدست آوریم. همانطور که می‌دانید(و در قسمت قبل بدست آورده می‌شود.) در الگوریتم استاندارد برای آپدیت کردن مراکز خوشه‌ها از میانگین اعضای آن خوشه استفاده می‌کردیم. می‌توان نشان داد که انتخاب مقدار خاصی برای نرخ آموزشی ϵ، که می‌تواند برای خوشه‌های مختلف متفاوت باشد، باعث شود که هر دو الگوریتم ضابطه یکسانی برای آپدیت کردن μ_i داشته باشند. مقداری از ϵ را بدست آورید که هر دو الگوریتم عملکرد مشابهی در این مرحله برای آپدیت کردن μ_1 دارند.

۳. (۲۰ نمره) وقتی‌که شبکه‌های عصبی بسیار عمیق می‌شوند(لایه‌های زیادی دارند)، آموزش‌دادن آن‌ها به علت Vanishing Gradient مشکل می‌شود - همانطور که مشتقات نسبت به لایه‌های متعددی backpropagate می‌شوند، ضرب‌های پشت هم می‌تواند موجب شود مشتقات بسیار کوچک شده و در نتیجه عملکرد شبکه بهبود پیدا نمی‌کند یا حتی تضعیف می‌شود!
یک راه بهینه برای رفع این مشکل، استفاده از ResNet است که به ویژه در بینایی ماشین کاربرد دارد. ایده اصلی ResNet استفاده از skip connections است که از یک یا چند لایه پرش می‌کند. گراف محاسباتی زیر را برای عملکرد ResNet مشاهده کنید:



$$\begin{aligned}
\mathbf{z_1} &= \mathbf{W_1 x} + \mathbf{b_1} \\
\mathbf{h_1} &= \mathrm{ReLU}(\mathbf{z_1}) \\
\mathbf{d} &= \mathbf{h_1} + \mathbf{x} \\
\mathbf{z_2} &= \mathbf{W_2 d} + \mathbf{b_2} \\
\mathbf{h_2} &= \mathrm{ReLU}(\mathbf{z_2}) \\
\theta &= \mathbf{h_2} + \mathbf{d} \\
\hat{\mathbf{y}} &= \mathrm{softmax}(\theta) \\
J &= \mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}})
\end{aligned}$$

ابعاد متغیرها را به صورت $x \in \mathbb{R}^{D_x \times 1}, W_1 \in \mathbb{R}^{H \times D_x}, b_1 \in \mathbb{R}^H, W_2 \in \mathbb{R}^{D_y \times H}, b_2 \in \mathbb{R}^{D_y}$ و $\hat{y} \in \mathbb{R}^{D_y \times 1}$ در نظر بگیرید. همچنین فرض کنید که $D_x = D_y = H$ است. در این سوال قصد داریم $\frac{\partial J}{\partial \mathbf{x}}$ را محاسبه کنیم. به ترتیب مراحل زیر را برای ایجاد جواب خود طی کنید:

الف) ابتدا $\delta_1 = \frac{\partial J}{\partial \theta}$ را بدست آورید.

ب) حال $\delta_2 = \frac{\partial J}{\partial z_2}$ را بدست آورید.

ج) $\delta_3 = \frac{\partial J}{\partial d}$ را محاسبه نمایید.

د) در نهایت با توجه به نتایج بخش‌های قبل $\frac{\partial J}{\partial \mathbf{x}}$ را بدست آورید.

۴. (۲۰ نمره) می خواهیم یک شبکه CNN با معماری مشخص‌شده در جدول زیر را به منظور دسته‌بندی تصاویر بناهای تاریخی ایران به ۶ دسته‌ی معماری دوره‌ی ایلام(عیلام)، معماری دوره مادها، معماری دوره هخامنشیان، معماری دوره اشکانیان، معماری دوره ساسانیان و معماری اسلامی، ایجاد کنیم.

الف) برای هر لایه از شبکه با توجه به اطلاعات زیر، تعداد وزن‌ها، تعداد بایاس و سایز Feature map را در جدول وارد کنید.

- منظور از CONV-K-N یک لایه ی کانولوشنی با N فیلتر با سایز K*K است. Padding و Stride برای همه ی لایه های CONV به ترتیب ۰ و ۱ در نظر گرفته شده است.
- منظور از Pool-K یک K*K pooling layer است که Stride برابر با K و Padding برابر با صفر دارد.
- منظور از FC-N یک لایه ی Fully Connected با N نورون است.

Layer	Activation Map Dimension	Number of Weights	Number of Biases
Input	۱۲۸*۱۲۸*۳	۰	۰
CONV-9-32			
Pool-2			
CONV-5-64			
Pool-2			
CONV-5-64			
Pool-2			
FC-6	۶	۶(۱۲*۱۲*۶۴)	۶

ب) چه تابع فعال سازی برای لایه آخر استفاده می کنید؟ با ذکر فرمول بیان کنید چه مزیتی نسبت به سایر توابع دارد.

ج) آیا می توانیم برای بهبود یادگیری الگوهای پیچیده تر، لایه های بیشتری به شبکه اضافه کنیم؟ با این کار چه مشکلی ممکن است برای شبکه ایجاد شود؟

د) دلیل استفاده از لایه های Pooling چیست؟

ه) استفاده از Stride بزرگتر از ۱ چه فایده ای دارد؟

۵. (۲۰ نمره) الف) منظور از exploding gradient problem در RNN چیست؟ تحت چه شرایطی این پدیده بوجود خواهد آمد؟

ب) به چه دلیل vanishing gradient در شبکه‌های RNN عادی نسبت به شبکه‌های feedforward مشکل رایج‌تری می‌باشد؟

ج) دو روش برای رفع مشکل vanishing gradient در RNN نام برده و نحوه عملکرد آن‌ها را برای این مسئله توضیح دهید.

د) شبکه عصبی بازگشتی (RNN) با یک نورون پنهان و تابع فعالسازی سیگموید در نظر بگیرید.

$$h_m = \sigma(\theta h_{m-1} + x_m)$$

ابتدا شماتیک این شبکه را ترسیم کنید و سپس اثبات کنید اگر $|\theta| < 1$ باشد، مشتق جزئی $\frac{\partial h_{m+k}}{\partial h_m}$ به ازای $k \to \infty$ به سمت صفر میل می‌کند.

۶. (۲۰ نمره) تابع فعالسازی ReLU می‌تواند باعث بوجود آمدن نورون‌های مرده، یعنی نورون‌هایی که تحت هیچ ورودی فعال نمی‌شوند و خروجی آن‌ها به ازای هر ورودی صفر است شود. شبکه عصبی feedforward دو لایه را با N نورون ورودی و H نورون در لایه پنهان (hidden) در نظر بگیرید که وزن میان آن‌ها $\mathbf{W}^{(1)}$ و بایاس آن‌ها نیز $b^{(1)}$ می‌باشد. خروجی این شبکه نیز یک نورون داشته(خروجی اسکالر) و با وزن‌های $W^{(2)}$ است.

$$h_i = ReLU(W_i^{(1)}.x + b_i^{(1)}) = ReLU(\sum_{j=1}^{N} W_{ij}^{(1)} x_j + b_i^{(1)}) \quad for \quad i \in \{1, 2, ..., H\}$$

$$\hat{y} = W^{(2)}.h$$

که در آن هدف ما بهینه کردن تابع هزینه‌ی مشتق‌پذیر دلخواه $l : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ که آرگومان‌های آن برچسب واقعی داده‌ها و پیش‌بینی‌های شبکه است، می‌باشد.

الف) تحت چه شرایطی نورون h_i مرده است؟ جواب شما باید بر حسب $W_i^{(1)}$ یعنی سطری از $\mathbf{W}^{(1)}$ که متناظر نورون i می‌باشد، و $b^{(1)}$ بیان شود.

ب) فرض کنید که به ازای یک نمونه خواهیم داشت :$\frac{\partial l}{\partial y} = 1$. مشتقات جزئی $\frac{\partial l}{\partial b_i^{(1)}}$ و $\frac{\partial l}{\partial W_{ij}^{(1)}}$ را برای این نمونه بدست آورید.

ج) با توجه به نتایج خود در بخش‌های قبلی، توضیح دهید چرا یک نورون مرده نمی‌تواند زنده شود؟! منظور از زنده شدن این است که پارامترهای آن به گونه‌ای تغییر کنند که به ازای تمام ورودی‌ها خروجی این نورون صفر نباشد.

د) راه‌حلی برای رفع این مشکل تابع ReLU ارائه دهید. (لزوما جواب یکتایی برای این بخش وجود ندارد، اما باید راه‌حل پیشنهادی خود را توجیه کنید.)