



به نکات زیر توجه کنید:

فایل پاسخ تمرین خود را به صورت یک فایل فشرده به همراه کدهای مربوط در Quera.ir بارگذاری نمایید. فایل فشرده را به صورت `HW1_[firstName]_[lastName]_[StudentID]` نامگذاری کنید. هر سوال عملی را داخل یک Jupyter Notebook جدا تحویل دهید. سوالات خود را در Quera مطرح کنید.

مسئله ۱. Linear Regression (۱۵ نمره)

مجموعه داده‌ها $S = \{(x^{(i)}, y^{(i)})\}_{i=1}^n$ را در نظر بگیرید به گونه‌ای که $x \in \mathbb{R}^d$ و $y \in \mathbb{R}$ است. برای تخمین برچسب یک نمونه x از رابطه رگرسیون خطی به صورت $\hat{y} = \sum_{j=1}^d w_j x_j + b$ و برای آموزش مدل از تابع هزینه SSE به صورت $\mathcal{L}(w, b) = \frac{1}{2n} \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2$ استفاده می‌کنیم. حال به سوالات زیر پاسخ دهید.

(الف) با استفاده از گرادینت کاهشی رابطه به‌روز رسانی برای وزن‌ها ارائه دهید.

(ب) حال رابطه محاسبه وزن‌ها را به صورت فرم بسته ارائه دهید.

(ج) با فرض اینکه گرادینت کاهشی بعد از m به‌روز رسانی به پاسخ بهینه می‌رسد، این دو روش را از نظر مرتبه زمانی با یکدیگر مقایسه کنید.

مسئله ۲. Activation Function (۱۰ نمره)

به سوالات زیر پاسخ دهید.

(الف) ابتدا مشکل vanishing gradient را توضیح دهید. سپس توابع فعال‌سازی ReLU و sigmoid با بررسی مشتق از این نظر مقایسه کنید.

(ب) ابتدا مقداردهی Xavier را توضیح دهید و سپس بررسی کنید که چگونه به مشکل محو شدن کمک می‌کند.

(ج) فرایند آموزش یک شبکه عصبی با تابع فعال‌سازی sigmoid را در صورتی که مقداردهی اولیه وزن‌ها بزرگ است، بررسی کنید.

مسئله ۳. Regularization & Optimization (۲۰ نمره)

به سوالات زیر پاسخ دهید.

(الف) چرا منظم‌ساز L_2 معمولاً روی Bias شبکه اعمال نمی‌شود؟ همچنین توضیح دهید چرا منظم‌ساز L_1 منجر به صفر شدن برخی از وزن‌ها می‌شود؟

ب) مسئله رگرسیون خطی بر روی n داده با تابع هزینه $\mathcal{L}(w) = \frac{1}{2n} \sum_{i=1}^n (y^{(i)} - x^{(i)T} w)^2$ را در نظر بگیرید. اثبات کنید اضافه کردن نویز از توزیع $\mathcal{N}(0, \sigma^2 I)$ به داده‌های ورودی معادل استفاده از منظم‌ساز L_2 در تابع هزینه است.

ج) توضیح دهید که Batch Normalization چگونه فرایند آموزش را سرعت می‌بخشد. همچنین توضیح دهید زمانی که سائز batch کوچک باشد، استفاده از Batch Normalization چه تاثیری در فرایند آموزش دارد.

د) رابطه گشتاور اول بهینه‌ساز آدام را به صورت $m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_{\theta} J(\theta_t)$ در نظر بگیرید. نشان دهید چرا مقادیر m_t گرایش به صفر دارند و چرا مقدار $\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$ که به صورت محاسبه می‌شود، با این مشکل روبرو نمی‌شود.

مسئله ۴. Backpropagation (۲۰ نمره)

الف) یک شبکه عصبی feedforward را با دولایه نهان با تابع فعال‌سازی sigmoid برای مسئله دسته‌بندی دودویی در نظر بگیرید. لایه اول نهان شامل ۴ نورون و لایه دوم شامل ۳ نورون است. ابعاد ورودی دلخواه در نظر گرفته می‌شود. در ابتدا تمامی وزن‌ها و بایاس شبکه صفر مقداردهی می‌شوند. به‌ازای یک ورودی، شبکه چه مقداری را خروجی دهد؟ بعد از یک بار به‌روز رسانی وزن‌ها با استفاده از SGD، بررسی کنید مقادیر وزن‌ها چگونه تغییر می‌کند.

ب) شبکه زیر را در نظر بگیرید.

$$\begin{aligned} h &= W^T x \\ u &= W'^T h \\ \hat{y} &= \text{Softmax}(u) \\ \mathcal{L}(W, W') &= -y^T \log \hat{y} \end{aligned}$$

که در آن $x \in \mathbb{R}^d$ و $h \in \mathbb{R}^H$ و $\hat{y} \in \mathbb{R}^d$ و y برچسب one-hot شده است. با در نظر گرفتن تابع هزینه برای یک نمونه ورودی، با استفاده از روش گرادیان کاهشی روابط به‌روز رسانی وزن‌ها را با نوشتن جزئیات مراحل بنویسید.

مسئله ۵. دسته‌بندی تصویر (عملی) (۴۰ نمره)

در این مسئله قصد داریم تصاویر مجموعه دادگان fashion-MNIST را دسته‌بندی کنیم. برای دریافت مجموعه دادگان می‌توانید از torchvision.datasets یا keras.datasets استفاده نمایید.

الف) ابتدا تابعی بنویسید که با هر بار اجرا ۴ تصویر این مجموعه دادگان را به صورت تصادفی در یک گرید ۲ در ۲ نمایش دهد.

ب) یک شبکه MLP با دو لایه نهان با اندازه‌های ۱۲۸ و ۶۴ پیاده‌سازی کنید. برای اینکار تنها اجازه دارید از کتابخانه numpy استفاده کنید. در اینجا نیاز دارید تا تمامی مراحل Backpropagation را پیاده‌سازی کنید و وزن‌ها را بر اساس گرادیان‌های محاسبه شده، با استفاده از الگوریتم SGD به‌روز رسانی کنید. از تابع هزینه Cross entropy و همچنین تابع فعال‌سازی sigmoid استفاده کنید. نمودار تابع هزینه در هنگام آموزش را بر روی مجموعه دادگان آموزش و آزمون رسم کنید و همچنین دقت دسته‌بندی که بر روی مجموعه دادگان آموزش و آزمون بدست آورده‌اید را گزارش کنید.

ج) مرحله (ب) را برای تابع فعال‌سازی ReLU تکرار کنید.

د) مرحله (ب) را با استفاده از الگوریتم Momentum برای به‌روز رسانی وزن‌ها تکرار کنید.

ه) مرحله (ب) را برای یک شبکه MLP با سه لایه نهان با اندازه‌های ۱۲۸ و ۶۴ و ۶۴ تکرار کنید.

و) مرحله (ب) را با استفاده از کتابخانه Pytorch پیاده‌سازی کنید.

نکته: در تمامی مراحل (ج) تا (و) نمودارها و نتایج مانند (ب) گزارش شوند.