به نام خدا

یادگیری ژرف

نیم‌سال دوم ۱۴۰۱-۱۴۰۲

مدرس: دکتر مهدیه سلیمانی

طرح تمرین: محمدرضا فریدونی، علیرضا سخایی‌راد، حسام اسداله زاده

دانشکده‌ی مهندسی کامپیوتر

تمرین سری اول (۱۰۰ نمره) | **مرور مفاهیم پایه** | زمان تحویل: ۱۰ اسفند (نظری)، ۱۲ اسفند (عملی)

لطفا نکات زیر را رعایت کنید:

- سوالات خود را از طریق پست مربوط به تمرین در Quera مطرح کنید.
- در هر کدام از سوالات، اگر از منابع خارجی استفاده کرده‌اید باید آن را ذکر کنید. در صورت همفکری با افراد دیگر هم باید نام ایشان را در سوال مورد نظر ذکر نمایید.
- پاسخ ارسالی واضح و خوانا باشد. در غیر این صورت ممکن است منجر به از دست دادن نمره شود.
- پاسخ ارسالی باید توسط خود شما نوشته شده باشد. به اسکرین‌شات از منابع یا پاسخ افراد دیگر نمره‌ای تعلق نمی‌گیرد.
- در صورتی که بخشی از سوال‌ها را جای دیگری آپلود کرده و لینک آن را قرار داده باشید، حتما باید تاریخ آپلود مشخص و قابل اعتنا باشد.
- تمام پاسخ‌های خود را در یک فایل با فرمت HW#_[SID]_[Fullname].zip روی کوئرا قرار دهید.
- برای ارسال هر تمرین تا ساعت ۲۳:۵۹ روز ددلاین فرصت دارید. مهلت تاخیر (مجاز و غیر مجاز) برای این تمرین، ۱۰ روز است.

سوال ۱: مرور جبرخطی (۲۰ نمره)

(آ) نشان دهید ماتریس Hessian یک تبدیل مثل $y = \psi(u, v, z)$ را می‌توان به صورت ماتریس ژاکوبی گرادیان این تبدیل نوشت. متغیرهای u, v, w را تک‌بعدی و y را تابعی برحسب آن‌ها در نظر بگیرید.

(ب) فرض کنید یک ماتریس دلخواه $A_{m\times n}$ داریم. همچنین بردارهای y و x که به ترتیب m و n بعدی اند به صورت $y = Ax$ به هم مرتبط می‌شوند. مشتق جزیی y نسبت به x به صورت زیر تعریف می شود.

$$\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \dots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \dots & \frac{\partial y_2}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \dots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

از روی فرم باز شده‌ی یک ضرب ماتریسی یعنی $y_i = \sum_{k=1}^{n} a_{ik}x_k$ عبارات زیر را بدست آورید

(i) $\frac{\partial y}{\partial x} = A$

(ii) اگر x یک تابع از z و A مستقل از z باشد، ثابت کنید $\frac{\partial y}{\partial z} = A\frac{\partial x}{\partial z}$

(iii) اگر تعریف کنیم $\alpha = y^TAx$ ثابت کنید $\frac{\partial \alpha}{\partial x} = y^TA$ و $\frac{\partial \alpha}{\partial y} = x^TA^T$

(iv) اگر x و y دو بردار n بعدی که تابعی از متغیر z اند و $\alpha = y^Tx$ ثابت کنید $\frac{\partial \alpha}{\partial z} = x^T\frac{\partial y}{\partial z} + y^T\frac{\partial x}{\partial z}$

(v) اگر $A_{m\times m}$ یک ماتریس non singular باشد که درایه‌های آن تابعی‌هایی از مقدار اسکالر α باشند، ثابت کنید

$$\frac{\partial A^{-1}}{\partial \alpha} = -A^{-1}\frac{\partial A}{\partial \alpha}A^{-1}$$

(ج) اگر حاصل‌جمع عناصر همه‌ی سطرهای یک ماتریس $n\times n$ برابر با m باشد، نشان دهید m یک مقدار ویژه برای این ماتریس می‌باشد.

(د) برخی از انواع مجموعه دادگان مانند تصاویر صورت یا برخی از انواع سری‌های زمانی دارای یک شبه‌تقارن ذاتی هستند (یعنی در این داده‌ها تقارنی وجود دارد که شاید بتواند به ما در آموزش یک مدل بهتر برای دسته‌بندی داده‌ها کمک کند). برای سادگی فرض کنید یک عکس کوچک با ابعاد 2×1 در اختیار داریم. ترم رگولاریزیشن L_2 به صورت $R(w) = w^T w = w^T I w$ تعریف می‌شود. ماتریس S را طوری بیابید که $R(w) = w^T S w$ از نامتقارن شدن وزن‌ها جلوگیری کند.

سوال ۲: بهینه‌سازی (۲۵ نمره)

- بهینه‌سازی در توابع **محدب**:

تابع زیر را در نظر بگیرید:

$$f(\underline{x}) = 3x_1^2 + 2x_2^2 - 3x_1x_2 + 4x_1^3 + x_1^4 \quad (۱)$$

تمام نقاط ایستای تابع را به دست آورید و نوع آن را مشخص کنید. راهنمایی:

نقاط ایستا در جاهایی رخ می‌دهند که $\nabla f = 0$ باشد. سه نوع نقطه‌ی ایستا وجود دارد: نقطه‌ی بیشینه، نقطه‌ی کمینه و نقطه‌های زینی.

در هر نقطه‌ی ایستا، مقدار $D = f_{x_1x_1} f_{x_2x_2} - (f_{x_1x_2})^2$ را محاسبه کنید. سپس هر نقطه‌ی ایستا را به ترتیب زیر آزمایش کنید:

- اگر $D < 0$ باشد، نقطه‌ی ایستا یک **نقطه‌ی زینی** است.
- اگر $D > 0$ و $\frac{\partial^2 f}{\partial x_1^2} > 0$ باشد، نقطه‌ی ایستا یک **کمینه‌ی محلی** است.
- اگر $D > 0$ و $\frac{\partial^2 f}{\partial x_1^2} < 0$ باشد، نقطه‌ی ایستا یک **بیشینه‌ی محلی** است.
- اگر $D = 0$ باشد، آزمون نامشخص است (نیاز به آزمون جایگزین داریم).

- بهینه‌سازی در توابع **غیرمحدب**:

تابع زیر را در نظر بگیرید:

$$f(\underline{x}) = 2x_1^2 + 2x_2^2 - 17x_2 \cos(0.2\pi x_1) - x_1x_2 \quad (۲)$$

الف: (**نظری** - روش نیوتن) در آنالیز عددی روش نیوتن ، که همچنین به عنوان روش نیوتن–رافسون (Newton-Raphson method) نیز شناخته می‌شود. برای بهینه‌سازی یک تابع در فضای یک بعدی داریم:

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}$$

در فضاهای با ابعاد بالاتر داریم:

$$f'(x) = \nabla f, f''(x) = \nabla^2 f = H(x), x_{n+1} = x_n - \alpha H^{-1}(x_n) f'(x_n)$$

با شروع از نقطه‌ی $(0, 0)$، جهت گرادیان نزولی (Gradient Descent) را پیدا کنید. سپس با استفاده از روش نیوتن، مقدار جدید نقطه شروع را در یکبار به‌روزرسانی به صورت تحلیلی به دست آورید.

ب: (**عملی** - روش نیوتن) با استفاده از روش نیوتن و با شروع از نقطه‌ی $(1, 3)$، نقطه کمینه این تابع را با استفاده از شبیه‌سازی کامپیوتری به دست آورید. همچنین به ازای تمامی نقاط $-5 < x_1 < 5, 0 < x_2 < 10$ با $stepSize = 0.5$، مقدار کمینه را به دست آورید و آن‌ها را بر اساس فاصله‌شان تا -36.4، به سه دسته نزدیک، دور و دورتر تقسیم‌بندی کنید و نمودار این نقاط را همانند شکل۱ رسم کنید و نتایج خود را به صورت مختصر توضیح دهید. برای حل این بخش، نوتبوک NewtonMethod_TODO را تکمیل کنید.

ج: (**نظری** - روش آدام) الگوریتم $Adam$ برای آموزش وزن‌های شبکه عصبی به صورت تکراری پله‌های ذیل را اجرا می‌کند:

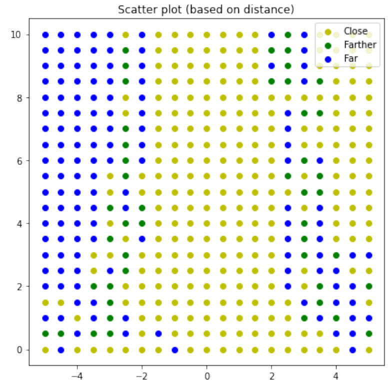
۱. $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (گرادیان تابع هدف تصادفی در زمان t)

۲. $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$

۳. $v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$

۴. $\hat{m}_t \leftarrow \frac{m_t}{1-\beta_1^t}$

۵. $\hat{v}_t \leftarrow \frac{v_t}{1-\beta_2^t}$



شکل ۱: نتیجه‌ی نمونه

۶. $\theta_t \leftarrow \theta_{t-1} - \frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t}+\epsilon}$

توضیح دهید که هر خط این الگوریتم چه عملی انجام می‌دهد؟ نشان دهید چرا مقادیر m_t به سمت صفر بایاس دارند و چرا مقدار $\hat{m}_t$ که به شکل $\frac{m_t}{1-\beta_1^t}$ محاسبه می‌شود (در tهای با مقادیر کمتر)، با این مشکل روبرو نمی‌شود؟ (توجه کنید که مقدار اولیه‌ی $m_0 = 0$ است)

سوال ۳: انتشار رو به عقب (۲۰ نمره)

(آ) یک شبکه‌ی عصبی با ورودی x را در نظر بگیرید. برای بدست آوردن خروجی محاسبات زیر بر روی x انجام می‌شود.

$$z = wx + b$$

$$y = \sigma(z)$$

$$\mathcal{L} = \frac{1}{2}(y-t)^2$$

$$\mathcal{R} = \frac{1}{2}w^2$$

$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

گراف محاسباتی این مسئله را رسم کنید و مشتقات $\mathcal{L}_{reg}$ را نسبت به همه‌ی متغیرها بدست آورید.

(ب) پارامترهای یک شبکه‌ی عصبی در ابتدا به صورت تصادفی و با مقادیر کوچک مقداردهی می‌شوند. توضیح دهید در صورت عدم رعایت این دو ویژگی در مقداردهی چه مشکلاتی بروز پیدا می‌کند

(ج) وزن‌های شبکه‌ی عصبی بدست آمده در قسمت اول سوال را با مقادیر تصادفی دلخواه مقداردهی کنید و برای یک ورودی دلخواه، با توجه به مشتقاتی که در قسمت اول بدست آوردید، با اعمال بهینه‌سازی gradient descent برای یک ایپاک با نرخ یادگیری 0.1 وزن‌های شبکه را آپدیت کنید.

سوال ۴: مقدمات پایتورچ - عملی (۱۵ نمره)

در این سوال با مقدمات فریمورک پایتورچ و عملیات پایه‌ای روی تنسورها آشنا می‌شوید. برای تکمیل این سوال لازم است تا در فایل q۱.zip به مطالعه‌ی نوتبوک موجود بپردازید و در هر قسمت، تابع متناظری که در فایل hw1_basic.py وجود دارد را تکمیل کنید.

سوال ۵: پیاده‌سازی MLP - عملی (۲۰ نمره)

در این سوال برای ابرهای دوبعدی داده، الگوریتم Multi-Layer Perceptron را پیاده‌سازی می‌کنید. در ابتدا به کمک numpy این کار را انجام می‌دهید و در ادامه با فریمورک پایتورچ و روند معمول این فریمورک این پیاده‌سازی را انجام می‌دهید. برای پیاده‌سازی این سوال به نوت‌بوک hw1_mlp.ipynb مراجعه کنید.