دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

بهینه‌سازی محدب

نیم‌سال بهار ۹۹

مدرس:دکتر مهدی جعفری سیاوشانی

پروژه تاریخ سررسید: ۱۳۹۹/۰۴/۳۱

برای حل این تمرین می‌توانید از کتاب‌خانه‌هایی که دارای امکان automatic differentiation هستند مانند PyTorch و jax استفاده کنید. در آخر موارد خواسته شده در هر بخش را به صورت یک گزارش به همراه کدهای خود فشرده و آپلود کنید.

روش‌های درجه دو

می‌خواهیم مسالهٔ دسته بندی به کمک logistic regression را برای داده‌های ساخته‌شده بررسی کنیم.
داده‌ها را به این صورت تولید می‌کنیم:

- دستهٔ صفر: صد داده از توزیع: $\mathcal{N}(\begin{bmatrix}-1\\-1\end{bmatrix}, I)$
- دستهٔ یک: صد داده از توزیع: $\mathcal{N}(\begin{bmatrix}1\\1\end{bmatrix}, I)$

سپس داده‌ها را به صورت تصادفی مخلوط می‌کنیم و ۱۵۰ نقطهٔ اول را به عنوان دادهٔ آموزش و ۵۰ نقطهٔ دیگر را برای ارزیابی جدا می‌کنیم. برای سادگی پیاده‌سازی logistic regression حالت بدون intercept آن را در نظر می‌گیریم یعنی:

$$X : \text{input features with shape (M, 2)} \quad (۱)$$
$$y : \text{target labels with shape (M, 1)} \quad (۲)$$
$$w = \begin{bmatrix}w_1\\w_2\end{bmatrix} \quad (۳)$$
$$\hat{y} = \sigma(Xw) \quad (۴)$$
$$loss = \frac{-1}{M}[y^T log(\hat{y}) + (1-y)^T log(1-\hat{y})] \quad (۵)$$

که در آن σ تابع سیگموید است. در هر بخش بعد از بهینه‌سازی روی دادهٔ آموزش میزان دقت روی دادهٔ ارزیابی را هم گزارش کنید.

۱. دربارهٔ محدب بودن تابع هزینه نسبت به بردار پارامترها (w) بحث کنید.

۲. نمودار سطح تابع هزینه را برای مقادیر $-2 < w_1 < 2, -2 < w_2 < 2$ رسم کنید.

۳. با شروع از $w_s = \begin{bmatrix}0\\0\end{bmatrix}$ روش gradient descent را با learning rate $\lambda = 0.2$ برای پنج گام اجرا کنید.

۴. با شروع از w_s این بار روش Newton را برای پنج گام اجرا کنید.

۵. روش Natural Gradient یک روش بهینه‌سازی از خانوادهٔ steepest descent است که در آن نرم استفاده شده، فاصلهٔ KL توزیع likelihood بین دو بردار پارامتر است. دربارهٔ این روش مطالعه کنید و به این سوال‌ها پاسخ دهید:

(آ) منظور از فضای توزیع (distribution space) در این روش چیست؟
(ب) رابطهٔ این روش با Fisher Information Matrix چیست؟

۶. با شروع از w_s روش Natural Gradient را با پارامتر یادگیری $\alpha = 0.2$ و با استفاده از Emperical Fisher Information Matrix پیاده‌سازی کنید و برای پنج گام اجرا کنید. عملیات به روزرسانی به صورت:

$$w_{next} = w - \alpha F^{-1} \nabla_w loss(w)$$

خواهد بود.

۷. حال می‌خواهیم میزان حساس بودن روش‌های بالا را به تغییرات در فضای پارامترها بررسی کنیم. برای این کار رابطهٔ ۴ را به صورت زیر تغییر می‌دهیم:

$$\hat{y} = \sigma(0.01 \times Xw)$$

پس از این تغییر دوباره نمودار سطح تابع هزینه را برای مقادیر $-2 < w_1 < 2, -2 < w_2 < 2$ رسم کنید.

۸. روش Gradient Descent را با learning rate $\lambda = 0.2$ برای پنج گام برای روابط جدید اجرا کنید. در یک نمودار تغییرات loss این حالت را به همراه تغییرات loss قسمت ۴ رسم کنید و مقایسه کنید.

۹. روش Newton را برای پنج گام برای روابط جدید اجرا کنید. در یک نمودار تغییرات loss این حالت را به همراه تغییرات loss قسمت ۵ رسم کنید و مقایسه کنید.

۱۰. روش Natural Gradient را با پارامتر یادگیری $\alpha = 0.2$ برای پنج گام برای روابط جدید اجرا کنید. در یک نمودار تغییرات loss این حالت را به همراه تغییرا loss قسمت ۶ رسم کنید و مقایسه کنید.

۱۱. نتایج بخش‌های ۸ و ۹ و ۱۰ را توضیح دهید. در هر حالت چرا این مشاهده حاصل شد؟

۱۲. در روش Natural Gradient در هر گام توزیع likelihood را برای قبل و بعد از به‌روز کردن پارامترها در یک نمودار رسم کنید.

۱۳. مزایا و معایب این سه روش را بررسی کنید. در بهینه‌سازی شبکه‌های عصبی برای این که مزیت‌های روش‌های درجه دو را داشته باشیم ولی معایب آن‌ها را نداشته باشیم از چه روش‌هایی استفاده می‌شود؟

روش‌های کاهش واریانس

می‌خواهیم مسالهٔ دسته‌بندی چندکلاسه به کمک multiclass logistic regression به همراه L2 Regularization برای ارقام دادگان MNIST بررسی کنیم.
برای کم‌شدن زمان اجرا $M = 6000$ نمونهٔ تصادفی از دادگان آموزش انتخاب می‌کنیم و به عنوان دادهٔ آموزش استفاده می‌کنیم. برای ارزیابی هم از داده‌های ارزیابی استفاده می‌کنیم. روابط این مساله برای یک دستهٔ k تایی به این صورت است:

X : input features with shape (k, 28*28)
y : target labels with shape (k,)
W : with shape (28*28, 10)
b : with shape (10,)

$$\hat{y} = softmax(XW + b)$$

$$loss = \frac{-1}{k} \sum_{i=1}^{k} log(\hat{y}[i, y[i]]) + \frac{1}{2}\gamma(trace(W^T W) + b^T b)$$

که در آن γ ضریب regularization است که به صورت $\gamma = 10^{-4}$ به آن مقدار می‌دهیم.

۱. روش SGD خام (batch_size = 1) را برای $t = M * 100$ قدم با learning rate های $\lambda = \{0.01, 0.05\}$ اجرا کنید. پس از هر M قدم loss روی کل دادگان آموزش، دقت روی دادهٔ ارزیابی، و اندازهٔ واریانس قدم‌های برداشته شدهٔ آن M قدم را محاسبه کنید (جمع واریانس بعدها). نمودار تغییرات این سه پارامتر را رسم کنید. برای نمودار تغییرات واریانس از نمودار نیمه‌لگاریتمی استفاده کنید.

۲. روش mini-batch SGD با (batch_size = 4) را برای $t = \frac{M}{4} * 100$ قدم با learning rate های $\lambda = \{0.01, 0.04, 0.2\}$ اجرا کنید. پس از هر $\frac{M}{4}$ قدم، loss روی کل دادگان آموزش، دقت روی دادهٔ ارزیابی، و اندازهٔ واریانس قدم‌های برداشته شدهٔ آن $\frac{M}{4}$ قدم را محاسبه کنید. نمودار تغییرات loss ، دقت، و واریانس را به همراه تغییرات مربوط به بخش قبل رسم کنید و مقایسه کنید.

۳. یکی از روش‌های کاهش واریانس روش SVRG[1] است. این روش را مطالعه کنید و به سوالات زیر پاسخ دهید:

(آ) مقدار $\mathbb{E}[w^{(t)}|w^{(t-1)}]$ را حساب کنید. این مقدار چه رابطه‌ای با روش SGD دارد؟

(ب) با توجه به مشاهدهٔ بخش (آ) چه انتظاری از نمودار تغییرات loss برای SGD و SVRG با نرخ یادگیری یکسان داریم؟

(ج) SVRG چرا باعث کاهش واریانس می‌شود؟

۴. روش SVRG را با learning rate های $\lambda = \{0.01, 0.05\}$ اجرا کنید. متغیر m که در الگوریتم باید مشخص شود را به صورت $m = M$ مقداردهی کنید. همین‌طور می‌توانید به انتخاب خودتان از option I یا option II استفاده کنید. پس از هر بار اجرا شدن حلقهٔ داخلی مقدار loss روی کل دادگان آموزش، دقت روی دادهٔ ارزیابی، و اندازهٔ واریانس قدم‌های برداشته شده را محاسبه کنید. نمودار تغییرات این سه پارامتر را به همراه نمودارهای بخش‌های قبلی ترسیم و مقایسه کنید.

۵. مزیت‌ها و معایب روش SVRG چیست؟

۶. زیاد و کم کردن مقدار m چه تفاوت‌هایی در عملکرد الگوریتم ایجاد می‌کند؟

سلامت باشید

[1] Stochastic Variance Reduced Gradient مقاله