



به نام خدا

تمرین سری سوم درس بهینه‌سازی (بخش پیاده‌سازی)

(نیمسال اول ۱۴۰۲)



تمرینات پیاده‌سازی

۱- هدفی که در این سوال دنبال می‌شود پیاده‌سازی روش AdaGrad در پایتورچ است. برای این منظور می‌بایست یک کلاس تعریف کنید که از `torch.optim.Optimizer` ارث برده باشد. سپس کافی است در این کلاس توابع `__init__()` و `step()` مشابه قطعه کد زیر تعریف و تکمیل گردند.

```
import torch
import torch.nn as nn

class MyAdaGrad(torch.optim.Optimizer):

    def __init__(self, params, lr):
        super(MyAdaGrad, self).__init__(params, defaults={'lr': lr})
        pass

    def step(self):
        pass

optimizer = MyAdaGrad(model.parameters(), lr=0.001)
```

پس از تعریف این کلاس، می‌توان از آن برای آموزش یک شبکه عصبی دلتواخ در پایتورچ استفاده کرد.

روش AdaGrad را در پایتورچ پیاده‌سازی کنید و از `Optimizer` خود برای آموزش یک شبکه عصبی دولایه، به منظور دسته‌بندی داده‌های MNIST استفاده کنید و نمودار تابع خطای آن را برای داده‌های آموزشی رسم کنید. در آموزش این شبکه از تابع خطای `nn.CrossEntropyLoss()` استفاده کرده و شبکه‌ی خود را براساس قطعه کد زیر تعریف کنید. در نهایت نتایج خود را با روش GD که در تمرین سری قبل پیاده‌سازی کرده بودید مقایسه کنید.

```
class MyNet(nn.Module):

    def __init__(self):
        super(MyNet, self).__init__()
        self.fc1 = nn.Linear(28*28, 256)
        self.fc2 = nn.Linear(256, 256)
        self.fc3 = nn.Linear(256, 10)
```

```

def forward(self, x):
    x = x.view(-1, 28*28)
    x = torch.relu(self.fc1(x))
    x = torch.relu(self.fc2(x))
    x = self.fc3(x)
    return x

model = MyNet()
loss_fn = nn.CrossEntropyLoss()

```

۲- روش *barrier* را پیاده‌سازی نموده و از آن برای حل مساله‌ی

$$\begin{aligned} \min \quad & \frac{1}{2} x^T A_i x - b^T x \\ \text{s. t.} \quad & Px \preceq q \end{aligned}$$

استفاده کنید. سپس نمودار *duality gap* برحسب *newton iterations* را به ازای $\mu =$

$\{1.5, 3, 5, 10\}$ رسم کنید. در این مساله $b \in \mathbb{R}^n$ برداری است که تمام عناصر آن مقدار ۱ دارد و $i = 1, 2$

به‌صورت زیر تعریف می‌شوند. $P \in \mathbb{R}^{2n \times n}$ و $q \in \mathbb{R}^n$ هستند که به‌صورت تصادفی مقداردهی می‌شوند. (مساله

برای ماتریس با اندازه‌های $n = 100, 400$ حل گردد).

$$A_1 = \text{tridiag}(-1, 4, -1)_{n \times n} = \begin{bmatrix} 4 & -1 & 0 & 0 & \dots & 0 \\ -1 & 4 & -1 & 0 & \dots & 0 \\ 0 & -1 & 4 & -1 & \dots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -1 & 4 & -1 \\ 0 & 0 & 0 & \dots & -1 & 4 \end{bmatrix}_{n \times n}$$

$$A_2 = \text{hilb}(n) = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \dots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \dots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \ddots & \ddots & \frac{1}{n+2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \dots & \frac{1}{2n-1} \end{bmatrix}_{n \times n}$$

۳- روش Primal-dual interior-point را برای مسالهی سوال ۲ پیاده‌سازی کنید سپس نمودار surrogate duality gap برحسب تعداد تکرارها و همچنین نمودار r_{feas} برحسب تعداد تکرارها را رسم نمایید. r_{feas} مطابق با رابطه‌ی زیر تعریف می‌شود. همچنین فرض کنید $\mu = 10$ است. (مساله برای ماتریس با اندازه‌های $n = 100, 400$ حل گردد).

$$r_{feas} = \left(\|r_{pri}\|_2^2 + \|r_{dual}\|_2^2 \right)^{\frac{1}{2}}$$

۴- در این تمرین به بررسی SVM خواهیم پرداخت. مدل SVM را به‌عنوان یک تابع $f(x): \mathbb{R}^n \rightarrow \{-1, 1\}$ تصور کنید که به‌صورت زیر تعریف می‌شود:

$$f(x) = \text{sgn}(a^T x - b)$$

در این رابطه $a \in \mathbb{R}^n$ و $b \in \mathbb{R}$ پارامترهای مدل هستند که با استفاده از داده‌ها یادگرفته می‌شوند. داده‌های مفروض $X = (x_1, \dots, x_N) \subset \mathbb{R}^n$ را به همراه برچسب‌های متناظرشان $Y = (y_1, \dots, y_N) \subset \{-1, 1\}$ در نظر بگیرید. می‌دانیم در صورتی که این داده‌ها خطی جدایی‌پذیر باشند می‌توانیم ابرصفحه‌ی جداکننده با بیشترین حاشیه^۱ را با حل مسالهی QCQP زیر پیدا کنیم.

$$\begin{aligned} \max \quad & t \\ \text{subject to} \quad & y_i(a^T x_i - b) \geq t, \quad \forall i \\ & \|a\|_2^2 \leq 1 \\ & t \geq 0 \end{aligned}$$

در اینجا مقدار بهینه‌ی t نشان‌دهنده‌ی کمترین فاصله هر کدام از نقاط از ابرصفحه موردنظر است. اگر هیچ ابرصفحه جداکننده‌ای وجود نداشته باشد تنها مقادیر شدنی برای متغیرهای مساله $a = 0, b = 0, t = 0$ خواهد بود.

الف) تابعی بنویسید که ابرصفحه‌ی جداکننده را برای مجموعه داده‌ی X با برچسب‌های Y بیابد (در صورت وجود). این تابع همچنین بایستی کمترین فاصله میان ابرصفحه و نقاط را نیز برگرداند. در صورتی که پاسخی وجود نداشته مقدار None برگردانده شود.

در مرحله بعد، می‌خواهیم SVM را به‌گونه‌ای بهبود دهیم که در صورتی که داده‌ها خطی جدایی‌پذیر نباشند، بهترین ابرصفحه جداکننده را پیدا کنیم. برای این منظور مساله موردنظر را در سه گام بازنویسی می‌کنیم (توجه کنید که تمامی این مسائل معادل هستند)

^۱ Maximal margin

$$\begin{aligned} \max & \quad t \\ \text{subject to} & \quad y_i(a^T x_i/t - b/t) \geq 1, \quad \forall i \\ & \quad \|a\|_2^2 \leq 1 \end{aligned}$$

$$\begin{aligned} \max & \quad t \\ \text{subject to} & \quad y_i(a^T x_i - b) \geq 1, \quad \forall i \\ & \quad \|a\|_2^2 \leq 1/t \end{aligned}$$

$$\begin{aligned} \max & \quad \frac{1}{2} \|a\|_2^2 \\ \text{subject to} & \quad y_i(a^T x_i - b) \geq 1, \quad \forall i \end{aligned}$$

اکنون می‌خواهیم تعداد نقاطی که در سمت نادرست ابرصفحه قرار گرفته‌اند را به حداقل برسانیم. در صورتی که نقطه‌ی x_i در سمت اشتباه ابرصفحه باشد آن‌گاه:

$$y_i(a^T x_i - b) \leq 1, \quad \Leftrightarrow \quad 1 - y_i(a^T x_i - b) \geq 0$$

برای اضافه کردن جریمه بابت نقاطی که در سمت اشتباه ابرصفحه قرار گرفته‌اند می‌توانیم عبارت $\max(0, 1 - y_i(a^T x_i - b))$ را به تابع خطا اضافه کنیم. به این ترتیب مساله بدون قید زیر به دست می‌آید:

$$\text{minimize} \quad \frac{1}{2} \|a\|_2^2 + C \sum_{i=1}^N \max(0, 1 - y_i(a^T x_i - b))$$

ثابت $C > 0$ مشخص می‌کند که چقدر می‌خواهیم خطای دسته‌بندی را جریمه کنیم و مقدار آن بسته به مساله‌ای که قصد حل کردن آن را داریم تنظیم می‌شود. توجه داشته باشید که $\xi_i = \max(0, 1 - y_i(a^T x_i - b))$ کوچکترین مقداری است که به ازای آن $y_i(a^T x_i - b) \geq 1 - \xi_i$ است. به این ترتیب می‌توانیم مساله‌ی بهینه‌سازی را به فرم زیر بازنویسی کنیم:

$$\begin{aligned} \max & \quad \frac{1}{2} \|a\|_2^2 + C \sum_{i=1}^N \xi_i \\ \text{subject to} & \quad y_i(a^T x_i - b) \geq 1 - \xi_i, \quad \forall i \\ & \quad \xi_i \geq 0 \end{aligned}$$

در اینجا $\xi_i = 0$ نشان دهنده‌ی آن است که x_i در سمت درست ابرصفحه جداکننده قرار گرفته است. اگر $\xi_i > 0$ باشد مقدار آن برابر است با $1 - y_i(a^T x_i - b)$.

ب) تابعی بنویسید که به حل این مساله بهینه‌سازی بپردازد و در ورودی خود مقدار C را به عنوان پارامتر ورودی دریافت کند. این تابع بایستی مقادیر بهینه (a, b, ξ) را برگرداند.

توابع خود را با استفاده از داده‌هایی که با استفاده از کد زیر تولید می‌شود تست کنید و خط جداکننده را برای هرکدام از مجموعه داده‌ها رسم کنید. (برای حل مسائل بهینه‌سازی می‌توانید از CVX استفاده کنید).

```
from sklearn import datasets

X, y = datasets.make_blobs(n_samples=500, centers=2, n_features=2, center_box=(0, 13), random_state=42)
X, y = datasets.make_moons(n_samples=500, noise=0.1, random_state=42)
```

فرمت گزارش:

- گزارش بایستی حاوی تمام نتایج بدست آمده از شبیه‌سازی‌های کامپیوتری در قالب فایل PDF باشد. همچنین انتظار می‌رود که در این گزارش برای سوالات پیاده‌سازی، تحلیل خود را از نتایج به‌دست آمده ارائه دهید.
- در صورتی که تمرینات را به‌صورت دست‌نویس حل می‌کنید. فایل‌های عکس تمرینات را با کیفیت مناسب و به‌ترتیب سوالات در یک فایل pdf قرار دهید و درنهایت این فایل را آپلود نمایید.
- فایل گزارش خود را تنها به‌شکل StdNum.pdf نام‌گذاری کنید. (مانند 9272203.pdf)

فرمت کدها:

- برای هر تمرین شبیه‌سازی کامپیوتری بایستی فایل کد جداگانه در محیط MATLAB، Python یا R تهیه شود.
- هر فایل کد خود را به شکل k_Q نامگذاری کنید. که k بیانگر شماره سوال شبیه‌سازی خواهد بود.

نحوه تبدیل:

- فایل‌های کد و گزارش خود را که طبق فرمت‌های فوق تهیه شده‌اند، در قالب یک فایل فشرده بارگذاری نمایید. فایل فشرده را تنها به شکل StdNum.zip نامگذاری نمایید. (مانند 9272203.zip)

تذکر:

- در صورتی‌که پارامتر خاصی در سوالات مشخص نشده با توجه به اطلاعاتی که در ارتباط با محدوده پارامتر دارید، مقدار دلخواهی انتخاب کنید و آن را در گزارش توضیح دهید.
- تحویل بخش‌های تئوری و پیاده‌سازی هر تمرین به‌صورت جداگانه خواهد بود. ارسال تمرین‌های تئوری تا یک روز تاخیر بلامانع است. پس از آن پاسخ این بخش بارگذاری خواهد شد و لذا مواردی که پس از بارگذاری پاسخ‌ها ارسال شوند قابل قبول نخواهد بود.
- برای تمرین‌های بخش پیاده‌سازی در مجموع ۷ روز تاخیر مجاز است (برای کل تمرینات جمع تاخیرهای شما نباید از ۷ روز بیشتر شود). در صورت تاخیر بیشتر از ۷ روز کسر ۵ درصد نمره از نمره کل تمرینات پیاده‌سازی به ازای هر روز تاخیر مد نظر قرار خواهد گرفت.
- در صورت شبیه بودن تمارین دانشجویان، نمره تمرین بین دانشجویان با تمرین مشابه تقسیم خواهد شد.
- در صورت وجود هرگونه سوال یا ابهام با یکی از ایمیل‌های زیر ارتباط برقرار کنید.

- farzane.abdoli@aut.ac.ir
- meysam.fozi@aut.ac.ir
- b.roshanfekr@aut.ac.ir