

به نام خدا



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده برق و کامپیوتر



درس سیستم‌های هوشمند

تمرین شماره یک

مهر ۱۴۰۰

فهرست سوالات

- سوال ۱ : بهینه‌سازی در توابع محدب ۳
- الف: تحلیل نقاط ایستا ۳
- ب: روش مرتبه اول (گرادیان) ۳
- ج: روش مرتبه دوم (نیوتن) ۳
- د: تابع درجه دو ۳
- سوال ۲ : بهینه‌سازی در توابع غیرمحدب ۴
- الف: روش گرادیان نزولی ۴
- ب: طول گام در روش گرادیان نزولی ۴
- ج: روش فراابتکاری ۴
- سوال ۳: ماشین بردار پشتیبان ۶
- الف: تحلیلی ۶
- ب: شبیه‌سازی ۶
- پیاده‌سازی بدون استفاده از کتابخانه: ۷
- پیاده‌سازی با استفاده از کتابخانه: ۷
- نکات تحویل تمرین: ۸

سوال ۱: بهینه‌سازی در توابع محدب

تابع زیر را در نظر بگیرید:

$$f(x_1, x_2) = 3x_1^2 + 12x_1 + 8x_2^2 + 8x_2 + 6x_1x_2; \quad (1-1)$$

الف: تحلیل نقاط ایستا

تمام نقاط ایستای تابع را به دست آورید و نوع آن‌ها را نیز مشخص کنید.

ب: روش مرتبه اول (گرادیان)

از نقطه اولیه (۱, ۱) شروع کنید و جهت گرادیان نزولی^۲ را پیدا کنید. سپس، با جست‌وجوی خط^۳ مقدار بهینه طول گام (α) را، با روش تحلیلی بیابید. یک بار به صورت دستی دو تکرار^۴ را محاسبه کرده، یک بار هم با در نظر گرفتن تابع خطا مناسب محاسبات را به صورت شبیه‌سازی کامپیوتری پیاده کنید.

ج: روش مرتبه دوم (نیوتن)

آیا روش نیوتن در این سوال کارایی دارد؟ با اعمال دو تکرار به صورت دستی نشان دهید.

د: تابع درجه دو

برای تابع درجه دو^۵ زیر هم نقاط ایستا را بیابید و نوع آن را تعیین کنید (محاسبات را به صورت ماتریسی انجام دهید). چه تفاوتی با قسمت الف (تابع ۱-۱) دارد؟ دلیل آن چیست؟ تفسیر کنید. راهنمایی: تابع قسمت الف را نیز به شکل ماتریسی در آورید و سپس تفاوت را توجیه کنید.

$$f(x_1, x_2) = \frac{1}{2} \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 6 & 20 \\ -8 & 16 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 12 & 8 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix};$$

^۱ Stationary Points

^۲ Gradient Descent

^۳ Line Search

^۴ Iteration

^۵ Quadratic Function

سوال ۲: بهینه‌سازی در توابع غیرمحدب

تابع زیر را در نظر بگیرید: (در این سوال نیازی به محاسبات دستی تکرارها نیست و شبیه‌سازی کامپیوتری کفایت می‌کند)

$$f(x) = x_1^2 - 10x_2 \cos(0.2\pi x_1) + x_2^2 - 15x_1 \cos(0.4\pi x_2);$$

الف: روش گرادیان نزولی

با استفاده از روش گرادیان نزولی با شروع از دو نقطه متفاوت مقدار کمینه محلی^۶ تابع را با شرط زیر به دست آورید. توجه داشته باشید که در این قسمت محاسبه طول گام^۷ مطرح نیست و با استفاده از روش‌های مختلف (طول گام ثابت، کاهش طول گام با افزایش تعداد تکرار و ...) می‌توانید الگوریتم را اجرا کنید.

$$^8(x_1^0, x_2^0) \in [-15, 15]; \quad (1-2)$$

ب: طول گام در روش گرادیان نزولی

با شروع از نقطه^۹ $(0, 0)$ طول گام بهینه (α) را یک بار به روش تحلیلی و بار دیگر با استفاده از روش آرمیجو^۹ به دست آورید و با هم مقایسه کنید. (حداقل دو تکرار)

ج: روش فراابتکاری

با استفاده از یکی از روش‌های فراابتکاری (تبرید شبیه‌سازی^{۱۰} شده یا ژنتیک^{۱۱}) مقدار کمینه محلی تابع را با شرط $(1-2)$ به دست آورید. در صورت استفاده از هر کدام از روش‌ها مولفه‌ها^{۱۲} را با توجه به شرایط مسئله در نظر بگیرید و علت انتخاب خود را به صورت مختصر بیان کنید.

^۶ Local Minimum

^۷ Step Length

^۸ بالانویس n به معنی مقدار در تکرار n ام است و در صورت قرار دادن مقدار صفر به معنی مقدار اولیه می‌باشد.

^۹ Armijo

^{۱۰} Simulated Annealing

^{۱۱} Genetic

^{۱۲} Parameters

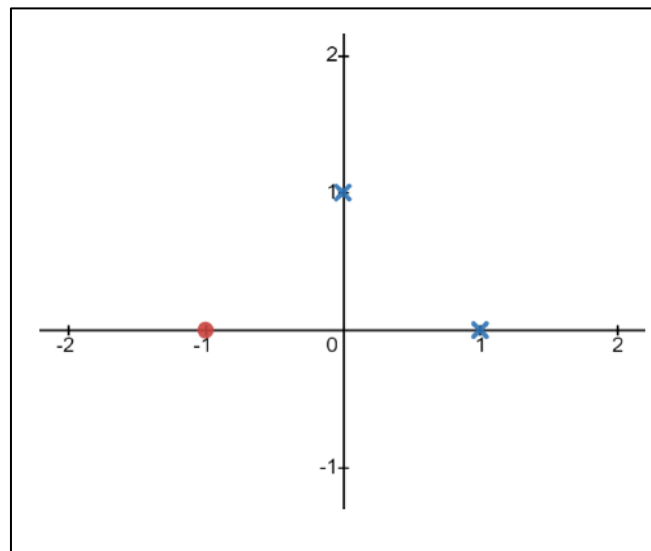
نکات پیاده‌سازی سوال ۲:

- مقدار بهینه تابع در حدود $۷۵/۶$ - است.
- در تمامی قسمت‌ها امکان به دام افتادن الگوریتم در کمینه‌های محلی وجود دارد، لذا با تغییر نقاط اولیه حداقل دو نوع از رفتار تابع را بررسی کنید؛ لیکن توجه داشته باشید که میزان خطا (با توجه به تابع خطای انتخابی شما) باید تا حد امکان کاهش پیدا کند.
- در قسمت (الف) برای رسیدن به همگرایی می‌توانید از مقادیر کوچک برای طول گام اولیه استفاده کنید. اما به این سوال پاسخ دهید که تا چه میزان می‌توان این طول گام را افزایش داد.
- در قسمت (ب) برای روش تحلیلی رسیدن به همگرایی اولویت نیست اما برای حداقل دو نقطه اولیه محاسبات حداقل دو تکرار را انجام دهید. (می‌توانید برای حل معادلات از هسته محاسبات سمبولیک متلب نیز استفاده کنید).
- در قسمت (ب) برای روش آرمیجو حداقل از گام دوم، در تعیین طول اولیه گام از روش‌های کاربردی که در کلاس توضیح داده شده‌اند، استفاده کنید. برای این روش بر خلاف روش تحلیلی همگرایی اولویت ماست، لذا از این نظر روش را بقیه روش‌ها مقایسه کنید.
- در قسمت (ج) میزان فاصله از مقدار بهینه به صورتی حدودی قابل قبول است.

سوال ۳: ماشین بردار پشتیبان

الف: تحلیلی

داده‌های شکل ۱-۳ به دو گروه تقسیم شده‌اند. با نوشتن روابط مربوطه، معادله جداسازی ماشین بردار پشتیبان^{۱۳} را برای آن‌ها پیدا کنید.



شکل ۱-۳: نقاط داده متشکل از دو گروه قرمز و آبی مربوط به بخش تحلیلی سوال ماشین بردار پشتیبان

ب: شبیه‌سازی

با استفاده از ماشین بردار پشتیبان و روش یک طبقه در قیاس با بقیه^{۱۴}، با استفاده از ویژگی‌های طول کاسبرگ^{۱۵} و عرض کاسبرگ^{۱۶} از داده‌گان iris، ناحیه طبقه‌های مختلف را مشخص و رسم کنید. علاوه بر رسم نواحی مختلف در هر دو قسمت، دقت داده آموزش^{۱۷}، ماتریس آشفتگی^{۱۸} و ماتریس اطمینان^{۱۹} را نیز در گزارش خود بیاورید و روابط میان آن‌ها را تفسیر کنید.

نکات مربوط به استفاده از داده‌گان:

برای استفاده از داده‌گان ذکر شده، می‌توانید از کد زیر در ابتدای کدهای خود استفاده کنید:

^{۱۳} Support Vector Machine (SVM)

^{۱۴} One Vs. Rest

^{۱۵} Sepal Length

^{۱۶} Sepal Width

^{۱۷} Train

^{۱۸} Confusion Matrix

^{۱۹} Confidence Matrix

```
from sklearn import datasets
iris = datasets.load_iris()
data = iris.data[:, :2] #data
label = iris.target #label
```

با توجه به ویژگی‌های خواسته شده و اطلاعاتی که در مورد دیتاست داریم، فقط ۲ ستون اول را برای پیاده‌سازی لازم داریم، در ادامه اطلاعات ستونی داده‌گان آورده شده است:

جدول ۱-۳ : مشخصات داده‌گان Iris، مورد استفاده مربوط به [شیبه‌سازی ماشین بردار پشتیبان](#)

شماره ستون	۱	۲	۳	۴
عنوان (مقادیر به سانتی‌متر)	طول کاسبرگ	عرض کاسبرگ	طول گلبرگ ^{۲۰}	عرض گلبرگ ^{۲۱}

پیاده‌سازی بدون استفاده از کتابخانه:

در این قسمت فقط اجازه استفاده از کتابخانه‌های ابتدایی را خواهید داشت. یک بار با استفاده از گرادیان نزول تصادفی^{۲۲} و یک بار با استفاده از الگوریتم تندترین نزول^{۲۳} با کمک قاعده آرمیجو طبقه‌بند را پیاده‌سازی کنید.

پیاده‌سازی با استفاده از کتابخانه:

[قسمت قبل](#) را با استفاده از کتابخانه‌های موجود پیاده‌سازی کرده و نتایج را مقایسه کنید.

راهنمایی: می‌توانید از کتابخانه sklearn برای این منظور استفاده کنید. به خاطر داشته باشید که ماشین بردار پشتیبان مورد نظر ما خطی است.

^{۲۰} Petal length

^{۲۱} Petal width

^{۲۲} Stochastic Gradient Descent (SGD)

^{۲۳} Steepest Descent (SD)

نکات تحویل تمرین:

- مهلت تحویل این تمرین، جمعه ۷ آبان است.
- انجام این تمرین به صورت یک نفره است.
- برای انجام تمرین‌ها فقط مجاز به استفاده از زبان‌های برنامه‌نویسی Python و MATLAB خواهید بود. در سؤالاتی که از شما خواسته شده است یک الگوریتم را پیاده‌سازی کنید مجاز به استفاده از توابع آماده نمی‌باشید مگر اینکه در صورت سوال مجاز بودن استفاده از این توابع یا کتابخانه‌ها صریح ذکر شده باشد.
- کدهای مربوط به هر تمرین می‌بایست در پوشه‌ای با نام Codes در کنار گزارش کار شما موجود باشد. این کدها باید خوانا و به صورت مرتبط نام‌گذاری شده باشند، لذا توضیحات لازم را به صورت یادداشت^{۲۴} در کدهای خود قرار دهید.
- گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است. لطفاً تمامی نکات و مفروضاتی که برای پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید را در گزارش ذکر کنید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست، ولیکن تحلیل و تفسیر نتایج بدست آمده الزامی است.
- گزارش‌ها تنها در قالب تهیه شده که روی صفحه درس در سامانه Elearn بارگذاری شده، تصحیح خواهند شد و به قالب‌های دیگر نمره‌ای تعلق نخواهد گرفت.
- در گزارش استفاده از زیرنویس برای تصاویر و بالانویس برای جداول الزامی است.
- در صورت مشاهده تقلب نمرات تمامی افراد شرکت‌کننده در آن ۱۰۰- لحاظ می‌شود.
- لطفاً گزارش، فایل کدها و سایر ضامین مورد نیاز را با ترتیب نام‌گذاری زیر در صفحه درس در سامانه یادگیری الکترونیکی بارگذاری نمایید.

HW[HW Number]_[LastName]_[StudentNumber].zip

- در صورت وجود هرگونه ابهام یا مشکل می‌توانید از طریق رایانامه‌های زیر با دستیاران آموزشی مربوطه در تماس باشید:

سرکار خانم مسعود – سوال ۱ – رایانامه { mahsamassoud@gmail.com }

سرکار خانم جعفریان – سوال ۲ – رایانامه { jafarian.fateme7899@gmail.com }

سرکار خانم نورزاد – سوال ۳ – رایانامه { njnoorzad@gmail.com }

^{۲۴} comment