Introduction to Machine Learning - HW3 - Q1

Professors: Abolghasemi & Arabi
Student: Mohamad Mahdi Samadi
Student ID: 810101465

سوال اول:

الف) فرم کلی بهینه سازی الگوریتم های جستجوی خط [1] را بیان کنید.

ب) درالگوریتم جستجوی خط تعریف جهت نزول چیست؟

پ) برای بدست آوردن طول پله در الگوریتم های جستجوی خط از چه روش هایی استفاده می شود ؟

ت) تابع $f(x_1, x_2) = (x_1 + x_2^2)^2$ را در نظر بگیرید. در نقطه $x^T = (0\ ,1)$ جهت جستجوی $p^T = (-1, -1)$ را در نظر می‌گیریم. نشان دهید که p یک جهت نزول است و تمامی مقادیر مینیم کننده مسئله زیر را در هر تکرار t پیدا کنید:

$$\min_{\alpha>0} f\ (x^t + \alpha p^t)$$

الف: در مسائل بهینه‌سازی برای انتخاب حرکت بعدی دو موضوع را باید مشخص کنیم. جهت حرکت و مقدار حرکت در آن جهت. بعد از مشخص کردن جهت حرکت، پیدا کردن مقدار حرکت یک مسئله line search می‌شود.

$$\theta_{i+1} = \theta_i + \alpha_{i+1} \times v_{i+1}$$

که در آن x_i نقطه کنونی، x_{i+1} نقطه جدید، v_{i+1} جهت مشخص شده حرکت و α_{i+1} همان جهت حرکت و هدف الگوریتم line ssearch هستند. لزوما به دنبال بهترین مقدار حرکت ممکن نیستیم. بلکه یک α که جواب نسبتا خوبی هم باشد کافی‌ست. توجه کنید که چون جهت حرکت در جهت حرکت تابع هدف به سمت مقدار local/global optimum تعیین شده است، باید یک α نامنفی پیدا کنیم. به این منظور می‌توان از α کوچک شروع کرده و تا جایی ادامه دهیم که دیگر مقدار $f(\theta_{i+1})$ نسبت به $f(\theta_i)$ بهتر نباشد. منظور از بهتر، "کمتر" در مسائل کمینه‌سازی و "بیشتر" در مسائل بیشینه‌سازی است. البته می‌توان تا جایی پیش رفت که تغییرات از حد مشخصی کمتر شود و در نتیجه به نقطه local\global optimum نزدیک شده‌ایم.

تفسیر هندسی این روش بدین صورت است که در فضای ممکن حرکت، بعد از مشخص شدن v انگار یک hyperplane رسم کرده‌ایم که تابع هدف را به شکل یک منحنی قطع کرده و باید بهترین نقطه به روی آن منحنی را پیدا کنیم.

ب: در مسائل کمینه‌سازی، جهت کاهش جهتی است که مقدار تابع هدف در نقطه جدید از مقدار آن در نقطه کنونی کمتر شود. به عبارتی رابطه زیر برقرار باشد:

$$f(\theta_{i+1}) < f(\theta_i)$$

برای این که متوجه شویم یک جهت کاهش است یا خیر رابطه زیر را برای آن چک می‌کنیم:

$$\nabla^T f(\theta_i) v_i < 0$$

در هر الگوریتم بیهنه‌سازی اشاره شده در درس این جهت به روش خاصی پیدا می‌شود که به چند مورد آنان اشاره می‌کنیم:

- Simplex search (Nelder-Mead) method:

  اگر در فضای سه بعدی سه نقطه را A، B و C بنامیم و بدون کم شدن از کلیت مسئله فرض کنیم بیشینه این نقاط A است، جهت کاهش در راستای عمود منصف BC است. البته در این روش خاص line search نداریم و میزان حرکت مشخص است. این مورد را می‌توان به فضای n بعدی نیز تعمیم داد.

- Hooke-Jeeves method:

  در این روش برای فضای n بعدی، n پایه متعامد (و ترجیحا یکه) می‌ساختیم که برای هر کدام از پایه‌ها با چک کردن اینکه حرکت در جهت آن یا در مخالف جهت آن به بهتر شدن جواب کمک می‌کند جهت را انتخاب می‌کردیم.

- Steepest descent:

  جهت حرکت را طبق فرمول زیر برابر منفی گرادیان تابع هدف در مسائل کمینه‌سازی می‌گذاریم.

$$\theta_{i+1} = \theta_i - \alpha \nabla f(\theta)_{\theta=\theta_i}$$

$$f(\theta_{i+1}) = f(\theta_i) - \alpha \, |\nabla f(\theta)|^2_{\theta=\theta_i}$$

- Newton’s method:

  این روش علاوه بر گرادیان تابع از وارون ماتریس هسین نیز استفاده می‌کند. البته step size نداریم و مستقیما همین جهت را پیش می‌رویم.

$$\theta_{i+1} = \theta_i - H_f^{-1}(\theta)_{\theta=\theta_i} \times \nabla f(\theta)_{\theta=\theta_i}$$

- Quasi-Newton’s method:

  در این روش تقریبی از ماتریس هسین می‌زنیم تا حجم محاسبات کمتر شود.

پ:

چند روش را بررسی و شرح می‌دهیم:

- Exact search:

به مانند روش Exhaustive search، تمام حالات مختلف در نظر گرفته شده و بهترین طول گام انتخاب می‌شود. همانطور که بیان کردیم نقطه جدید از رابطه زیر به دست می‌آید:

$$\theta_{i+1} = \theta_i + \alpha_{i+1} \times v_{i+1}$$

در مسائل کمینه‌سازی، α به صورت زیر انتخاب می‌شود:

$$\alpha_i = \arg\ \min\ f(\theta_{i+1}) = \arg\ \min\ f(\theta_i + \ \alpha v_i) \quad \forall \alpha \geq 0$$

و در مسائل بیشینه‌سازی به این صورت:

$$\alpha_i = \arg\ \max\ f(\theta_{i+1}) = \arg\ \max\ f(\theta_i + \alpha\ v_i) \quad \forall \alpha \geq 0$$

- Inexact search:

در این روش‌ها لزوما به دنبال بهترین جواب ممکن نیستیم. بلکه در گام فعلی سعی در بهتر کردن جواب تا جای قابل قبولی داریم و میان هزینه محاسبات و پیشرفت حرکت تعادل برقرار می‌کنیم. حال چگونه مشخص کنیم بهتر شدن تا کجا قابل قبول است؟ شرایط Wolf به ما کمک می‌کند.

$$\theta_{i+1} = \theta_i + \alpha_i v_i$$
$$f(\theta_{i+1}) \leq f(\theta_i) + c_1 \alpha_i \nabla^T f(\theta_i) v_i$$
$$\nabla^T f(\theta_{i+1}) v_i \geq c_2 \nabla^T f(\theta_i) v_i$$

رابطه اول اطمینان حاصل می‌کند که حرکت در جهت و اندازه انتخاب شده به اندازه کافی مقدار تابع را کاهش دهد. رابطه دوم اطمینان حاصل طول گام میزان پیشرفت در جهت انتخاب شده را در قدم‌های آتی به میزان چشمگیری کاهش ندهد.

- Fixed step size search:

یک مقدار ثابت برای گام حرکت انتخاب شده و از آن تبعیت می‌کنیم. مانند روش Grid search.

- Heuristic search:

در این روش‌ها، در ابتدا با طول گامی شروع کرده و در گذر زمان بسته به ماهیت مسئله آن را کاهش/افزایش می‌دهیم. به طور مثال از فرمول زیر برای کاهش استفاده می‌کنیم.

$$\alpha_i = \frac{\alpha}{1+\lambda i}$$

که در آن α برابر طول گام اولیه، λ برابر ضریب کاهشی ثابت (بین صفر تا یک) و i برابر شماره iteration هستند.

ت: از رابطه‌ای که در بخش ب بیان کردیم استفاده می‌کنیم:

$$\nabla^T f(\theta_i) v_i < 0$$
$$\nabla^T f(\theta) = (\frac{\partial f}{x_1}, \frac{\partial f}{x_2}) = [2x_1 + 2{x_2}^2, 4x_2(x_1 + {x_2}^2)] \rightarrow \nabla^T f(\theta_i) v_i = [2x_1 + 2{x_2}^2, 4x_2(x_1 + {x_2}^2)] v$$

نقطه شروع و جهت را در رابطه بالا جایگذاری می‌کنیم:

$x_1 = 0,\ x_2 = 1,\ v^T = [-1,\ -1] \rightarrow \nabla^T f(\theta_i) v_i = [2,\ 4][-1, -1]^T = -6 < 0$

پس این جهت، جهت نزول است.

اگر تمامی مقادیر کمینه مسئله را بخواهیم از همان رابطه بخش ب که $f(\theta_{i+1}) < f(\theta_i)$ بود استفاده می‌کنیم.

با فرض اینکه منظور حرکت در همان جهت است مسئله را حل می‌کنیم.

$\theta_{i+1} = \theta_i + \alpha v = [0, 1]^T - \alpha[1, 1]^T = [-\alpha,\ 1 - \alpha]^T$

$f(\theta_{i+1}) = (-\alpha + (1 - \alpha)^2)^2 = (\alpha^2 - 3\alpha + 1)^2$

به دنبال کمینه‌سازی تابع بالا هستیم. پس گرادیان برحسب α را صفر می‌کنیم.

Minimize $f(\theta_{i+1}) \rightarrow \frac{\partial f(\theta_{i+1})}{\partial \alpha} = 2(\alpha^2 - 3\alpha + 1)(2\alpha - 3) = 0 \rightarrow \alpha_1 = \frac{3+\sqrt{5}}{2},\ \alpha_2 = \frac{3-\sqrt{5}}{2},\ \alpha_3 = \frac{3}{2}$