تمرین شماره سوم درس یادگیری ماشین

پریا پاسه‌ورز

شماره دانشجویی: 810101393

سوال دوم)

الف) روش نیوتون برای بهینه‌سازی را بیان کنید.

روش نیوتون یک الگوریتم iterative برای محاسبه مینیمم تابع f است و و با استفاده از رابطه زیر محاسبه می‌شود:

$$x_{k+1} = x_k - H_k^{-1} \nabla f(x_k)$$

ب) مشکلات روش نیوتون را بیان کنید.

1. اگر ماتریس هسین مثبت معین نباشد، جهت حرکت ممکن است به سمت مینیمم نباشد و در نهایت این الگوریتم converge نکند.
2. محاسبه ماتریس هسین و وارون آن از لحاظ محاسباتی سنگین است و باعث می‌شود این الگوریتم از ($O(n^3$ باشد.
3. ممکن است ماتریس هسین اصلا وارون‌پذیر نباشد.
4. اگر حدس اولیه خوبی نداشته باشیم، این الگوریتم به نقطه مینیمم converge نمی‌کند.

پ) روش های نیوتون تصحیح شده را بیان کنید.

1. Quasi Newton method

گفتیم که یکی از مشکلات روش نیوتون این است که محاسبه ماتریس هسین و وارون آن هزینه زیادی دارد. در این روش ماتریس هسین از طریق یکی الگوریتم iterative تقریب زده می‌شود.

روابط آن به تفصیل در قسمت بعد آورده شده‌اند.

2. Damped Newton’s method

در این روش به update rule در روش نیوتون، یک ضریب اضافه می‌شود، با کمک این ضریب، مقدار update step را می‌توان کنترل کرد:

$$x_{k+1} = x_k - \alpha_k H^{-1}(x_k) \nabla f(x_k)$$

گفتیم که یکی از مشکلات روش نیوتون این است که اگر حدس اولیه خوب نباشد و در هر مرحله گامهای بزرگی برداریم، ممکن است converge نکند. حال با استفاده از این damping factor می‌توان مقدار گام را به مقدار درستی ست کرد. این مقدار می‌توان ثابت یا متغییر باشد، یعنی اول مقادیر بزرگی اختیار کند و رفته رفته کوچک شود.

3. Trust region Newton method

این روش نیز سعی می‌کند convergence در الگوریتم نیوتون را بهبود ببخشد. به همین دلیل به جای step ای که در الگوریتم نیوتون مشخص شده است، این متود جهت جستجو را به سمت یک trust region محدود می‌کند.

1. از یک نقطه اولیه مثل x_0 شروع می‌کنیم.
برای trust region هم یک مقدار اولیه مثبت مثل Δ در نظر می‌گیریم.
برای جهت اولیه حرکت (p)، روش‌های مختلفی وجود دارد. می‌توان مقدار آن را صفر در نظر گرفت.

2. تعریف می‌کنیم:

$$m_k(p) = f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T B_k p$$

که در اینجا B_k ماتریس هسین یا تقریبی از ماتریس هسین است.

3. حال مسئله بهینه‌سازی زیر را حل می‌کنیم:

$$p_k = \arg\min_p m_k(p), \quad \text{subject to } |p| \le \Delta_k$$

4. سپس ضریب ρ_k را محاسبه می‌کنیم:

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}$$

اگر این مقدار حدودا 1 شود، یعنی مدل به خوبی عمل می‌کند و اگر 0 شود، یعنی عملکرد آن ضعیف است.

5. حال مقدار trust region را با توجه به این ضریب آپدیت می‌کنیم:

- If $\rho_k > \eta_1$ (good agreement):

  Increase Δ_k

- If $\rho_k < \eta_2$ (poor agreement):

  Decrease Δ_k

$\eta_1 = 0.75, \quad \eta_2 = 0.25$

6. سپس نقطه کنونی را آپدیت می‌کنیم. اگر $\rho_k > \eta_2$:

$$x_{k+1} = x_k + p_k$$

در غیر این صورت:

$$x_{k+1} = x_k$$

7. آنقدر این کار را تکرار می‌کنیم تا اندازه $\nabla f(x_k)$ از یک threshold مثل ε کمتر شود.

4. Line search Newton method

در این روش نیز با تعیین مقدار ضریب، اطمینان حاصل می‌کنیم که objective function در هر مرحله کاهش پیدا می‌کند و این یعنی مسئله نهایتا converge می‌کند. همچنین مقدار step size را می‌توان تغییر داد.

مقدار این ضریب را می‌توان از طریق یکی از روش‌هایی که در سوال یک بیان شد، مثل golden section algorithm یا gradient descent algorithm تعیین کرد.

$$x_{k+1} = x_k - \alpha_k H^{-1}(x_k)\nabla f(x_k)$$

5. Modified Hessian Newton method

گفتیم یکی از مشکلات متود نیوتون این است که ممکن است ماتریس هسین مثبت معین نباشد و به همین دلیل جهت حرکت کاهشی نباشد. از طرف دیگر، ممکن است این ماتریس وارون‌پذیر نیز نباشد. به همین دلیل ماتریس هسین را مشابه روبه‌رو تغییر می‌دهیم:

$$\widetilde{H_k} = H_k + \lambda I$$

λ کوچک‌ترین مقداری است که با استفاده از آن اطمینان حا صل می‌کنیم ماتریس جدید وارون‌پذیر است. سپس در الگوریتم نیوتون از این ماتریس به جای ماتریس هسین استفاده می‌کنیم.

6. Conjugate Gradient Newton method

گفتیم یکی از مشکلات روش نیوتون این است که ممکن است converge نکند. در روش گرادیان مزدوج، جهت حرکت را مستقیما پیدا می‌کنیم:

$$x_{i+1} = x_i - \alpha p_i$$

$$p_0 = 0, \;\; p_{i+1} = g_i - \beta_i p_i$$

$$\beta_i = \frac{{g_i}^T g_i}{{g_{i-1}}^T g_{i-1}}$$

ت) مسئله بهینه سازی روش‌های شبه نیوتون (DFB, BFGS) را بیان کنید.

BFGS (Broyden–Fletcher–Goldfarb–Shanno):

این روش نمی‌تواند در یک گام همگرا شود، ولی ثابت می‌شود که در فضای n بعدی در n گام همگرا می‌شود.

تقریب ماتریس هسین در این روش مثبت معین است، پس تضمین می‌شود که در جهت نزول حرکت می‌کنیم.

1. با یک حدس اولیه مثل x_0 برای مسئله شروع می‌کنیم.
   مقدار اولیه ماتریس هسین را نیز I در نظر می‌گیریم.
   ماتریس زیر را تعریف می‌کنیم:
$$g_k(x) = \nabla f(x_k)$$

2. حال می‌خواهیم با کمک الگوریتم line search، step size را پیدا کنیم:

$$x_{k+1} = x_k + \alpha_k p_k$$

در اینجا جهت را با کمک رابطه نیوتون محاسبه می‌کنیم، ولی به جای ماتریس هسین، تقریب کنونی‌مان از مقدار آن را قرار می‌دهیم:

$$p_k = -Q_k g_k$$
$$x_{k+1} = x_k - \alpha_k Q_k g_k$$

3. نهایتا برای تقریب ماتریس هسین:

$$Q_{k+1} = \left(I - \frac{\nabla x_k \nabla g_k^T}{\nabla x_k^T \nabla g_k}\right) Q_k \left(I - \frac{\nabla x_k \nabla g_k^T}{\nabla x_k^T \nabla g_k}\right)^T + \frac{\nabla x_k \nabla x_k^T}{\nabla x_k^T \nabla g_k}$$

4. آنقدر این کار را تکرار می‌کنیم تا اندازه $\nabla f(x_k)$ از یک threshold مثل ε کمتر شود.

DFP (Davidon–Fletcher–Powell):

تقریب ماتریس هسین در این روش مثبت معین است، پس تضمین می‌شود که در جهت نزول حرکت می‌کنیم.

1. با یک حدس اولیه مثل x_0 برای مسئله شروع می‌کنیم.
   مقدار اولیه ماتریس هسین را نیز I در نظر می‌گیریم.
   ماتریس زیر را تعریف می‌کنیم:
$$g_k(x) = \nabla f(x_k)$$

2. حال می‌خواهیم با کمک الگوریتم line search، step size را پیدا کنیم:

$$x_{k+1} = x_k + \alpha_k p_k$$

در اینجا جهت را با کمک رابطه نیوتون محاسبه می‌کنیم، ولی به جای ماتریس هسین، تقریب کنونی‌مان از مقدار آن را قرار می‌دهیم:

$$p_k = -Q_k g_k$$
$$x_{k+1} = x_k - \alpha_k Q_k g_k$$

3. نهایتا برای تقریب ماتریس هسین:

$$Q_{k+1} = Q_k + \frac{\nabla x_k \nabla x_k^T}{\nabla x_k^T \nabla g_k} - \frac{Q_k \nabla g_k \nabla g_k^T Q_k}{\nabla g_k^T Q_k \nabla g_k}$$

4. آنقدر این کار را تکرار می‌کنیم تا اندازه $\nabla f(x_k)$ از یک threshold مثل ε کمتر شود.