Advanced Optimization

Homework 4.

—

Ali Izadi

810199102

سوالات تحلیلی 2
سوال ۱ 2
سوالات پیاده سازی 5
سوال ۱ 5
سوال ۲ 12

سوالات تحلیلی

سوال ۱

الف) ابتدا تابع لاگرانژی را مینویسم.

$$L(x_1,\, x_2,\, \lambda) \;=\; x_1 -\; \lambda(-|x_1| - |x_2| + 1) \;=\; x_1 \;+\; \lambda|x_1| + \;\lambda|x_2| \;-\lambda$$

در نتیجه تابع دوگان مطابق با زیر است.

$$q(\lambda) = \inf_{x_1,\, x_2} L(x_1,\, x_2,\, \lambda)$$

تابع لاگرانژی مشتق پذیر نیست بنابراین نمیتوان برای پیدا کردن آن از مشتق برابر صفر استفاده کرد. این حالت حتما باید مینیمم داشته باشد بنابراین حالت های مختلف لامبدا را بررسی میکنیم که به ازای کدام لامبدای بزرگتر از یک تابع q مینیمم دارد.

$$L(x_1,\, x_2,\, \lambda) = \; x_1 + \lambda|x_1| + \lambda|x_2| - \lambda$$

هدف مینیمم کردن تابع بر حسب x1 و x2 است.

ترم |x2| مثبت است بنابراین همیشه بزرگتر از صفر است و تابع را در x2=0 مینیمم میکند.

ترم منفی لامبدا هم که constant است. بنابراین دو حالت زیر برای بقیه ترم ها پیش می آید.

۱- اگر $0 \le \lambda < 1$ آنگاه ترم اول یعنی $x_1 + \lambda|x_1|$ minmum ندارد زیرا مقدار منفی بی نهایت میتواند بگیرد. بنابراین تابع q در این حالت infimum ندارد.

۲- اگر $\lambda \ge 1$ آنگاه ترم اول یعنی $x_1 + \lambda|x_1|$ به ازای x1=0 مقدار آن مینیمم میشود.

بنابراین در این حالت تابع q را به ازای x1=0 مینیمم میکند.

در نتیجه به ازای x1=0 و x2=0 تابع دوگان را محاسبه میکنیم که به دست می آید.

$$q(\lambda) = \inf_{x_1,\, x_2} L(x_1,\, x_2,\, \lambda) \;=\; \cancel{x_1}^{0} + \lambda\, |\cancel{x_1}|^{0} + \lambda\, |\cancel{x_2}|^{0} - \lambda$$
$$q(\lambda) = \; -\lambda$$

در نتیجه مسئله دوگان برابر است با:

$$\max\, q(\lambda) \;= -\lambda$$
$$where\, \lambda \geq 1$$

که جواب مسئله دوگان و اصلی برابر است با $d^\star = f^\star = -1$

ب)

ابتدا تابع لاگرانژی را مینویسم.

$$L(x_1,\, x_2,\, \lambda) \;=\; x_1 - \lambda_1(-|x_1| - |x_2| + 1) - \lambda_2(-|x_1| + 1) - \lambda_3(-|x_2| + 1)$$
$$= x_1 + \lambda_1|x_1| + \lambda_2|x_1| + \lambda_1|x_2| + \lambda_3|x_2| - \lambda_1 - \lambda_2 - \lambda_3$$

در نتیجه تابع دوگان مطابق با زیر است.

$$q(\lambda) = \inf_{x_1,\, x_2} L(x_1,\, x_2,\, \lambda)$$

مطابق با قسمت الف ترم های لامبدا سمت راست تاثیری در مینیمم کردن ندارند.

هم چنین چون |x2| مثبت است بنابراین در x2=0 تابع را مینیمم میکند.

ترم باقیمانده یعنی $x_1 + \lambda_1|x_1| + \lambda_2|x_1|$ را باید حالت های مختلف برای آن در نظر گرفت که مطابق قسمت الف زمانی که $\lambda_1 \geq 0.5,\ \lambda_2 \geq 0.5$ باشد تابع در x1=0 مقدار مینیمم دارد.

در نتیجه به ازای x1=0 و x2=0 تابع دوگان را محاسبه میکنیم که به دست می آید.

$$q(\lambda) = \inf_{x_1,\, x_2} L(x_1,\, x_2,\, \lambda_1,\; \lambda_2,\, \lambda_3) \;=\; -\lambda_1 - \lambda_2 - \lambda_3$$

در نتیجه مسئله دوگان برابر است با:

$$\max\ q(\lambda)\ = -\lambda_1 - \lambda_2 - \lambda_3$$

$$where\, \lambda_1 \geq 0.5,\ \lambda_2 \geq 0.5,\ \lambda_3 \geq 0$$

که جواب مسئله دوگان و اصلی برابر است با $d^\star = f^\star = -1$

سوالات پیاده سازی

سوال ۱

روش پنالتی:

روش penalty با اضافه کردن constraintها به عنوان یک ترم پنالتی به تابع اصلی و تبدیل مسئله به یک مسئله بدون قید عمل مطابق با زیر عمل میکند. ضریب mu باید در طول الگوریتم به بی نهایت میل کند.

$$Q(x;\mu) \stackrel{\text{def}}{=} f(x) + \frac{\mu}{2}\sum_{i\in\mathcal{E}} c_i^2(x),$$

پیاده سازی الگوریتم مطابق با الگوریتم ۱۷.۱ کتاب nocedal که در زیر آورده شده است انجام شده است.

Framework 17.1 (Quadratic Penalty Method).

Given $\mu_0 > 0$, a nonnegative sequence $\{\tau_k\}$ with $\tau_k \to 0$, and a starting point x_0^s;
for $k = 0, 1, 2, \ldots$
  Find an approximate minimizer x_k of $Q(\cdot;\mu_k)$, starting at x_k^s,
    and terminating when $\|\nabla_x Q(x;\mu_k)\| \le \tau_k$;
  if final convergence test satisfied
    stop with approximate solution x_k;
  end (if)
  Choose new penalty parameter $\mu_{k+1} > \mu_k$;
  Choose new starting point x_{k+1}^s;
end (for)

- مقدار mu ابتدایی ۱ در نظر گرفته شده است و هر بار در 1.2 ضرب میشود و در این صورت در آزمایش های انجام شده به بی نهایت میل میکند.

- در هر iteration نیاز به حل مسئله بدون قید است که از روش نیوتن برای حل این زیر مسئله استفاده کردیم.

  هم چنین مقدار alpha یا طول پله زیر مسئله نیوتن عدد ثابت 0.01 در نظر گرفته شد.

  هر زیر مسئله نیوتن هم شرط هم گرایی آن بر اساس کمتر شدن مقدار گرادیان تابع Q از epsilon است و که این مقدار 0.001 در نظر گرفته شد.

  هر زیر مسئله طبق شرط بالا نیز حتما باید همگرا شود تا به نتیجه دلخواه برسیم.

پیاده سای قسمت اصلی الگوریتم در زیر مشاهده میشود. که گرادیان و hessian تابع Q در کد آورده شده است.

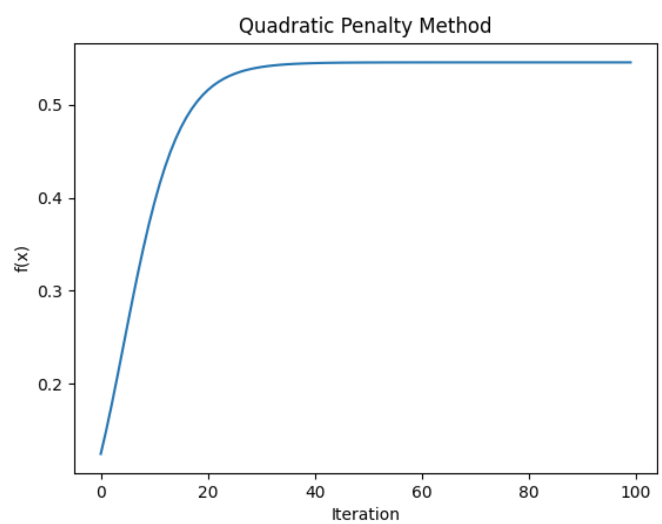
```
x0 = np.array([0.1, 0.2, 0.7])
x = x0

epsilon = 0.001
mu = 1

xs = []
fs = []

for i in range(100):
    if i % 10 == 0:
        print(i, mu)
    for j in range(10000):
        g_Q = grad_Q_penalty(x, mu)
        H_Q = hessian_Q_penalty(x, mu)
        x = x - 0.01 * np.dot(np.linalg.inv(H_Q), g_Q)

        if np.linalg.norm(g_Q) < epsilon:
            print('sub-problem converged')
            break

    mu = mu * 1.2
```

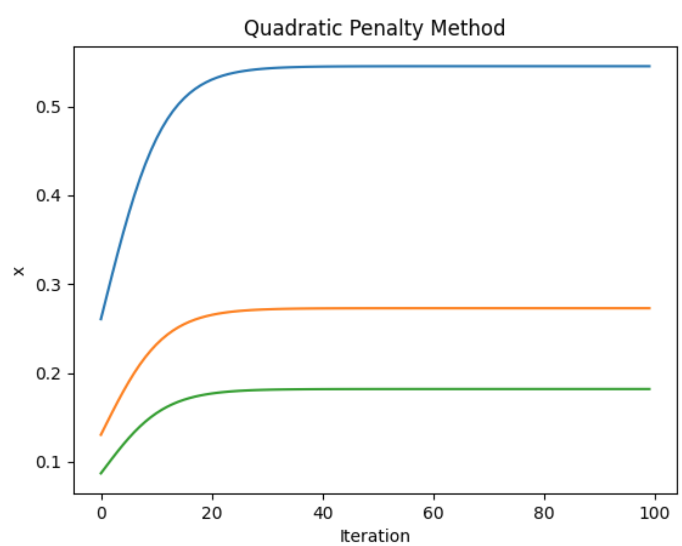
نتایج:

در زیر نمودار f و مقادیر x1 در زیر آورده شده است که نشان از همگرایی بهینه سازی دارد. صعودی بودن مقدار f نیز به خاطر همگرا شدن به جایی است که قید برقرار باشد و همان طور که در نمودار xها مشاهده میشود xها به سمت برقرار قید مساوی ۱ همگرا شده اند.

در زیر x نهایی نیز آورده شده است.

[0.18195525 0.2728545 0.54519023]





روش ضرایب

روش ضرایب علاوه بر اضافه کردن قید پنالتی تقریبی از ضرایب لاگرانژ را نیز به دست می آورد و تابع بدون قید زیر را بهینه میکند.

$$\mathcal{L}_A(x, \lambda; \mu) \stackrel{\text{def}}{=} f(x) - \sum_{i \in \mathcal{E}} \lambda_i c_i(x) + \frac{\mu}{2} \sum_{i \in \mathcal{E}} c_i^2(x),$$

پیاده سازی الگوریتم مطابق با الگوریتم ۱۷.۳ کتاب nocedal که در زیر آورده شده است انجام شده است.

Framework 17.3 (Augmented Lagrangian Method-Equality Constraints).

Given $\mu_0 > 0$, tolerance $\tau_0 > 0$, starting points x_0^s and λ^0;
for $k = 0, 1, 2, \ldots$
  Find an approximate minimizer x_k of $\mathcal{L}_A(\cdot, \lambda^k; \mu_k)$, starting at x_k^s,
    and terminating when $\|\nabla_x \mathcal{L}_A(x_k, \lambda^k; \mu_k)\| \le \tau_k$;
  if a convergence test for (17.1) is satisfied
    stop with approximate solution x_k;
  end (if)
  Update Lagrange multipliers using (17.39) to obtain λ^{k+1};
  Choose new penalty parameter $\mu_{k+1} \ge \mu_k$;
  Set starting point for the next iteration to $x_{k+1}^s = x_k$;
  Select tolerance τ_{k+1};
end (for)

این الگوریتم دقیقا مشابه با الگوریتم پنالتی است با این تفاوت که مقدار lambda آن دیگر صفر نیست و در هر مرحله طبق رابطه زیر به روز رسانی میشود.

$$\lambda_i^{k+1} = \lambda_i^k - \mu_k c_i(x_k), \qquad \text{for all } i \in \mathcal{E}.$$

- نحوه پیاده سازی و پارامترهای استفاده شده در این مسئله نیز مطابق با قبل است با این تفاوت که در هر مرحله mu در 1.1 ضرب میشود.

- در هر iteration مشابه با قبل نیاز به حل مسئله بدون قید است که از روش نیوتن برای حل این زیر مسئله استفاده کردیم.

  هم چنین مقدار alpha یا طول پله زیر مسئله نیوتن عدد ثابت 0.01 در نظر گرفته شد.

  هر زیر مسئله نیوتن هم شرط هم گرایی آن بر اساس کمتر شدن مقدار گرادیان تابع Q از epsilon است و که این مقدار 0.001 در نظر گرفته شد.

  هر زیر مسئله طبق شرط بالا نیز حتما باید همگرا شود تا به نتیجه دلخواه برسیم.

پیاده سای قسمت اصلی الگوریتم در زیر مشاهده میشود. که گرادیان و hessian تابع Q در کد آورده شده است.

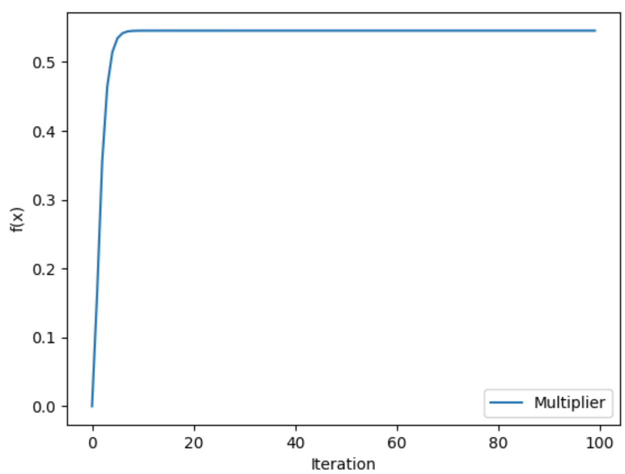
```
x0 = np.array([0.1, 0.2, 0.7])
x = x0

epsilon = 0.001
mu = 1
lambda_ = 1

xs = []
fs = []

for i in range(100):
    if i % 10 == 0:
        print(i, mu)
    for j in range(10000):
        g_Q = grad_Q_multiplier(x, lambda_, mu)
        H_Q = hessian_Q_multiplier(x, lambda_, mu)
        p = - np.dot(np.linalg.inv(H_Q), g_Q)
        x = x - 0.01 * -p

        if np.linalg.norm(g_Q) < epsilon:
            print('sub-problem converged')
            break

    mu = mu * 1.1
    lambda_ = lambda_ + mu * constraint(x)
```

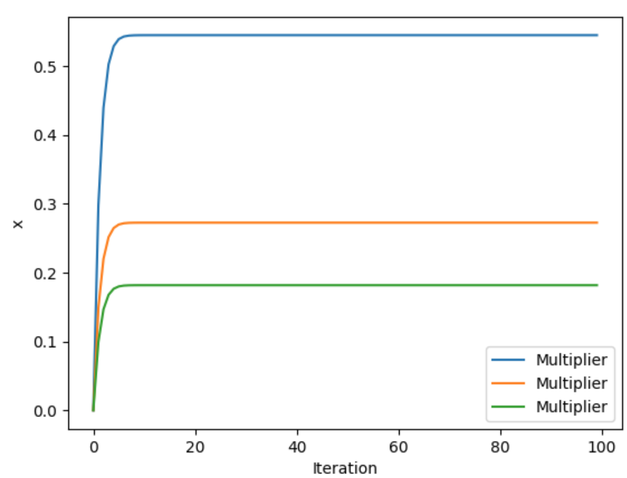
نتایج:

در زیر نمودار f و مقادیر x1 در زیر آورده شده است که نشان از همگرایی بهینه سازی دارد. صعودی بودن مقدار f نیز به خاطر همگرا شدن به جایی است که قید برقرار باشد و همان طور که در نمودار xها مشاهده میشود xها به سمت برقرار قید مساوی ۱ همگرا شده اند.

در زیر x نهایی نیز آورده شده است.

[0.18187384 0.27278559 0.54534057]



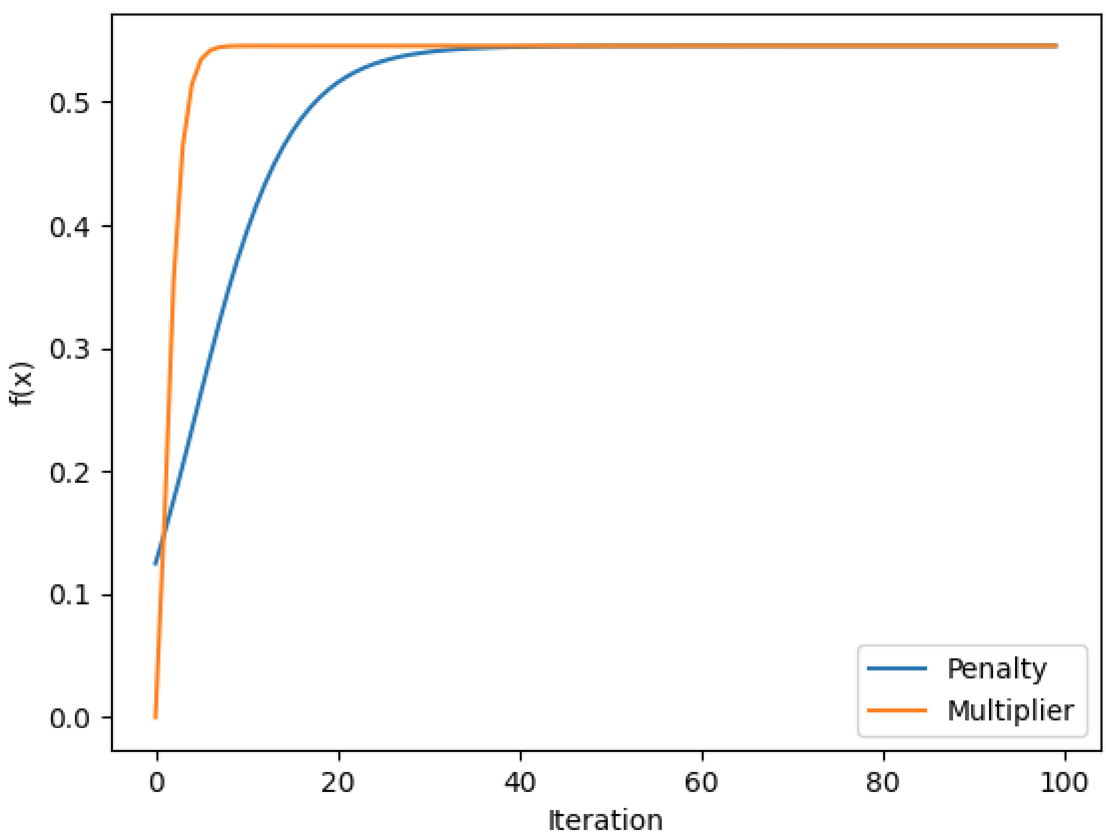


مقایسه دو روش پنالتی و ضرایب.

همان طور که از جواب نهایی مشاهده شد دو روش تقریبا به جواب یکسان همگرا شده اند.

در زیر نمودار همگرایی تابع برای این دو روش آورده شده است که نشان از همگرایی سریع تر روش ضرایب دارد.

روش ضرایب به خاطر اضافه کردن ترم لاگرانژی و به خاطر حفظ smoothness مشکل ill conditioning در بهینه سازی به استفاده از روش پنالتی را برطرف میکند و همگرایی سریعتری دارد.



سوال ۲

برای پیاده سازی از روش حائل یا barrier که یک روش interior point است استفاده کرده ایم.

هدف از این روش حل مسئله بهینه سازی زیر است.

$$\min f(x), \quad \text{s.t. } c_i(x) \geq 0, i \in \mathcal{I}, x \in \mathcal{X}$$

که یک تابع حائل بر روی قیود مطابق با زیر تعریف میکند.

$$B(x) = -\sum_{i \in \mathcal{I}} \ln(c_i(x))$$

و مسئله بهینه سازی با قید را به یک مسئله بدون قید مطابق با زیر تبدیل میکند.

$$x_k = \arg\min_{x \in \mathcal{S}} \{f(x) + \epsilon_k B(x)\}$$
$$0 < \epsilon_{k+1} < \epsilon_k, \quad \epsilon_k \to 0$$

پیاده سازی مطابق با الگوریتم ۱۱.۱ کتاب boyd که الگوریتم آن در زیر آورده شده است انجام شده است.

Algorithm 11.1 *Barrier method.*

given strictly feasible x, $t := t^{(0)} > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

repeat

1. *Centering step.*
   Compute $x^\star(t)$ by minimizing $tf_0 + \phi$, subject to $Ax = b$, starting at x.
2. *Update.* $x := x^\star(t)$.
3. *Stopping criterion.* **quit** if $m/t < \epsilon$.
4. *Increase* t. $t := \mu t$.

در هر iteration یک زیر مسئله بهینه سازی بدون قید باید انجام شود که از روش steepest descent برای حل آن استفاده کرده ایم.

در ادامه روابط بالا و هم چنین مشتقات تابع به دست آمده نسبت به پارامترهای خواسته شده را نشان خواهیم داد.

مسئله SVM مطابق با زیر است:

$$f = \min_{W,\, b,\, \xi} 0.5||W||^2 + C\sum_{i=1}^{n}\xi_i$$
$$s.t\{y_i\left(w^T x_i + b\right) - 1 + \xi_i \geq 0,\ \xi_i \geq 0,\ i = 1, \ldots, n\}$$

تابع حائل مطابق با زیر است:

$$B(x) = -\sum_{i=1}^{n}\ln\left[y_i\left(w^T x_i + b\right) - 1 + \xi_i\right] - \sum_{i=1}^{n}\ln\xi_i$$

و در نتیجه تابع بدون قید مسئله اصلی برابر است با

$$f(x) + \epsilon B(x)$$

حال از این تابع نسبت به سه پارامتر خواسته شده مشتق میگیریم تا از آن در روش steepest descent برای به روزرسانی این پارامترها استفاده کرد.

$$\frac{\mathrm{d}L}{\mathrm{d}w} = w - \epsilon\sum_{i=1}^{n}\frac{y_i x_i}{y_i(w^T x_i + b) - 1 + \xi_i}$$

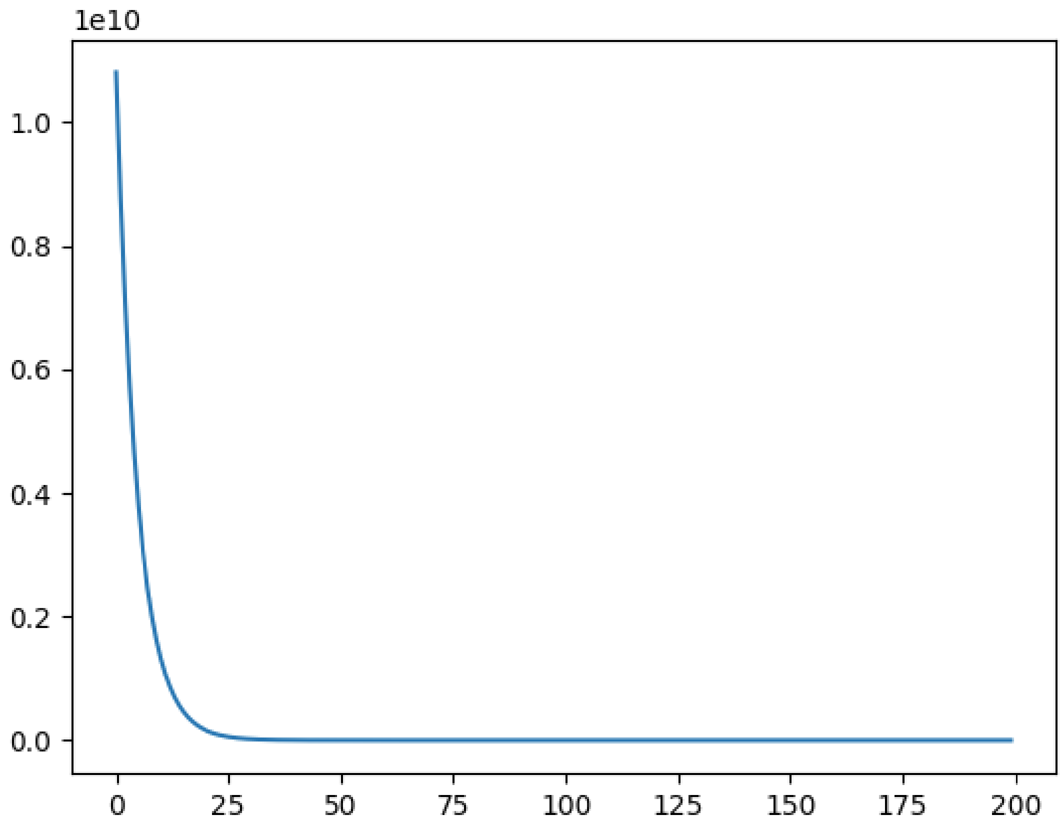
$$\frac{\mathrm{d}L}{\mathrm{d}b} = -\epsilon\sum_{i=1}^{n}\frac{1}{\xi_i}$$

$$\frac{\mathrm{d}L}{\mathrm{d}\xi_i} = C - 2\epsilon$$

در نتیجه در هر مرحله بر اساس مشتقات بالا و با طول پله انتخابی 0.1 بر اساس روش steepest descent پارامترها به روز رسانی میشوند.

مقدار ϵ نیز با ضرب در 0.1 کاهش میابد تا به صفر میل کند.

نمودار تابع f در زیر آورده شده است.



هم چنین مقدار CCR عدد 0.259 به دست آمده است.