Finite Newton method for unconstrained optimization problem

Neha Goyal, Kapil

नेहा गोयल, कपिल

National Institute of Technology, Kurukshetra

राष्ट्रीय प्रोद्योगिकी संस्थान, कुरुक्षेत्र

Abstract

Newton method is most famous method for optimization problem. It is an iterative approach for optimization type problem. In this tutorial we are focusing on finite newton method to solve unconstrained optimization problem. We have also discussed the constrained optimization problem and Lagrangain multiplier method briefly to convert constrained optimization problem to unconstrained optimization problem. As an example, minimization problem is also solved using a finite newton method.

सारांश

Optimization problem के लिए न्यूटन विधि सबसे प्रसिद्ध विधि है। यह optimization प्रकार की समस्या के लिए एक iterative approach है। इस tutorial में हम unconstrained optimization problem को हल करने के लिए सीमित न्यूटन विधि पर ध्यान केंद्रित कर रहे हैं। हमने constrained optimization problem और Lagrangian multiplier method, constrained optimization problem ko unconstrained problem में परिवर्तित करने के लिए संक्षेप में चर्चा की है। उदारण स्वरुप एक unconstrained optimization problem को newton विधि से solve किया गया है|

Keyword: Optimization problem, Constraint, Lagrangian multiplier, Finite Newton method

Introduction

optimization को constraint (बाधाओं) को पूरा करते हुए उपलब्ध संसाधनों के साथ सर्वोत्तम परिणामों को प्राप्त करने के रूप में परिभाषित किया गया है। हम अपने दैनिक जीवन में अधिकांश चीजों को अनुकूलित करते हैं। प्रकृति ने लगभग हर चीज को अनुकूलित किया है। यह किसी के पास जीवित रहने और सर्वोत्तम होने के बारे में है। optimization जीवन का एक तरीका है! optimization के साथ optimization बहुत कुछ करना है। "Optimists view

ISSN: 2229-6913(Print), ISSN: 2320-0332(Online)

Article Received: 25 September 2019 Revised: 28 December 2019 Accepted: 24 March 2020 Publication: 24 June 2020

the proverbial glass half full and not half empty.” किसी भी बाधाओं को देखते हुए, वे स्थिति से बाहर निकलने का प्रयास करते हैं।

गणित और कंप्यूटर विज्ञान में, एक optimization समस्या (optimization problem) वह समस्या है जो सभी संभावित समाधान (feasible solution) से सर्वोत्तम समाधान की खोज करती है।अपने सबसे सरल रूप में, इसे एक निर्धारित function के भीतर व्यवस्थित रूप से input मानों का चयन करके और function के मान की गणना करके वास्तविक function को अधिकतम करने और कम करने के रूप में वर्णित किया जा सकता है। सामान्य रूप से, optimization में दिए गए domain और विभिन्न प्रकार के objective function के अनुसार कुछ objective function के निष्कर्ष "सर्वोत्तम variables" मान (value) शामिल होते हैं।

Given a function f:

$$f: A \to R, x_0 \in A$$

Either, $f(x_0) \le f(x) \;\; \forall\, x \in A$ (minimization)

Or, $f(x_0) \ge f(x) \; \forall\, x \in A$ (maximization)

आम तौर पर, $A \subset R^n$ constraint (समानता (equality / असमानता (inequality)) के सेट के रूप में विशिष्ट होता है जिसे $x \in A$ को संतुष्ट करना होता है। f को objective function / loss function / लागत (cost) function/ उपयोगिता (utility) function/ fitness function के रूप में परिभाषित किया जा सकता है। optimization में यह सबसे महत्वपूर्ण बात है; हमें यह जानने की जरूरत है कि हम चरम(extreme) पर क्या सुधारना चाहते हैं। पहचान किए गए उद्देश्य के आधार पर चरम अधिकतम या न्यूनतम हो सकता है। यदि यह एक लागत है, जो कम से कम की जाती है; और यदि यह लाभ है, तो हम अधिकतम करते हैं, अक्सर, हम इसे एक उद्देश्य कार्य (objective function) कहते हैं क्योंकि इसे अनुकूलित करने के लिए कुछ चरों (variables) पर निर्भर होना पड़ता है । Objective function , optimization variables का एक function होना चाहिए। Unconstrained optimization problem कुछ अनुप्रयोगों (experiments) में सीधे उत्पन्न होती हैं लेकिन वे अप्रत्यक्ष रूप से सीमित optimization problem के सुधारों से उत्पन्न होती हैं। प्रायः objective function में penalized terms के साथ constraints को प्रतिस्थापित (placements of constraints with penalized terms in objective function) करके एक unconstrained optimization problem के रूप में हल करना व्यावहारिक है। यदि constraint optimization problem में केवल समानता बाधाएं(equality constraint) हैं तो "Lagrangian Multiplier” तकनीक को इसे unconstraint optimization problem में बदलने के लिए उपयोग किया जा सकता है। unconstraint optimization problem में चर की संख्या objective function और constrains की संख्या, मूल चर(variables) की संख्या होगी। यदि unconstraint optimization problem में inequality constraint है तो इसे ज्यामितीय optimal condition के संदर्भ में, KKT condition के प्रयोग से unconstrained optimization में परिवर्तित किया जा सकता

General constrained optimization problem:

$$\min_{x_1, x_{2,\dots,} x_n} f(x_1, x_2, x_3, \dots, x_n)$$

Subject to:

$$g_{1(x_1, x_2, \dots x_n)} \leq 0$$

$$g_{2(x_1, x_2, \dots x_n)} \leq 0$$

$$\dots$$

$$g_{m(x_1, x_2, \dots x_n)} \leq 0$$

Lagrange multiplier तकनीक constraints का वर्णन करने वाले variables को जोड़कर objective function को संशोधित करती है। The objective function $f(x)$ is augmented by the constraint equations through a set of non-negative multiplicative Lagrange multipliers, $\lambda_j \geq 0$. The augmented objective function, L is a function of the n design variables and m Lagrange multipliers.

Lagrangian of f को निम्नलिखित तरीके से व्यक्त किया जा सकता है

$$L(x_1, x_2, \dots, x_n, \alpha_1, \alpha_2 \dots \alpha_m) = f(x_1, x_2, \dots, x_m) + \sum_{j=1,2,3 \dots m} \alpha_j \, g_j(x_1, x_2, \dots . x_n)$$

Newton method for unconstrained optimization problem

Isaac newton और joseph Raphson के नाम पर Newton Raphson विधि, polynomial equation की roots को खोजने के लिए एक लोकप्रिय iterative approach है। इसे न्यूटन की विधि भी कहा जाता है। Taylor series के पहले कुछ terms के आधार पर, न्यूटन-रैफसन विधि का अधिक उपयोग किया जाता है जब दिए गए फ़ंक्शन / समीकरण का first derivative एक बड़ी value होती है। यह अक्सर numerical methods

में अन्य root finding विधियों का उपयोग कर प्राप्त values के मूल्य में सुधार करने के लिए प्रयोग किया जाता है।

Newton method, दिए गए समीकरण $(f'(x) = 0)$ का समाधान खोजने के लिए पुनरावृत्ति दृष्टिकोण (iterative approach) है| दिए गया समीकरण का twice differentiable होना अति आवश्यक है| यह समाधान अधिकतम, न्यूनतम या saddle point हो सकता है।

मान लीजिए, हम निम्नलिखित optimization problem को हल करना चाहते हैं

$$(P:) \quad min f(x)$$
$$x \in R^n$$

At $x = x', f(x)$ can be approximated by

$$f(x) \approx h(x) := f(x') + \nabla f(x')^T (x - x') + \frac{1}{2}(x - x')^t H(x')(x - x')$$

It is a Taylor expansion of $f(x)$ at $x = x'$, here $\nabla f(x)$ is gradient of $f(x)$ and $H(x)$ is hessian of $f(x)$.

यहाँ, h(x) एक द्विघातीय समीकरण(quadratic equation) है; जो $\nabla h(x) = 0$को हल करके कम किया जाता है| Gradient of $h(x)$ को इस प्रकार दिया गया है|

$$\nabla h(x) = \nabla f(x') + H(x')(x - x')$$

इसीलिए ,
$$\nabla f(x') + H(x')(x - x') = 0$$
$$x - x' = -H(x')(x - x')$$

$-H(x')(x - x')$ direction, को newton direction कहा जाता है|

Algorithm

Step 1 Given x_0, set $k \to 0$

Step 2 $d^k = -H(x^k)^{-1} \nabla f(x^k)$. If $d^k = 0$ then stop.

Step3 choose step size $\alpha^k = 1$

Step 4 Set $x^{k+1} \leftarrow x^k + \alpha^k d^k, k \leftarrow k + 1$. Go to step 2

Finite newton method के लिए assumption

1. $H(x^k)$ is non-singular for each iteration
2. It may be possible that $f(x^{k+1}) \leq f(x^k)$

ISSN: 2229-6913(Print), ISSN: 2320-0332(Online)
Article Received: 25 September 2019 Revised: 28 December 2019 Accepted: 24 March 2020 Publication: 24 June 2020

उदाहरण 1: let $f(x) = 9x - 4log(x-7)$. Suppose we want to minimize the given function.

दिया हुआ:

$$f(x) = 9x - 4log(x-7).$$

X के लिए newton direction

$$f'(x) = 9 - \frac{4}{x}$$
$$f''(x) = \frac{4}{x^2}$$

X के लिए newton direction

$$d = -f''(x)^{-1}f'(x)$$

$$d = -\frac{x^2}{4}\left(9 - {}^4\!/_x\right) = -{}^9\!/_4\, x^2 + x$$

Newton's method will generate the sequence of iterates {x} satisfying:

$$x^{k+1} = x^k + d$$

$$= x^k + (-f''(x^k)^{-1}f'(x^k))$$

$$= x^k + (x^k - {}^9\!/_4\,(x^k)^2)$$

$$= 2x^k - {}^9\!/_4\,(x^k)^2)$$

X की प्रारंभिक विभिन्न values के लिए इस विधि द्वारा generated sequence के कुछ उदाहरण नीचे table में दिए गए हैं।

iteration	x=0.01	X=0.1	X=0	X=7	X=10	X=100
1	0.009775	0.0775	0	-103.25	-215	-22400
2	0.018895	0.10661	0	-20940	-94761	-1.1189e+09
3	0.035306	0.10249	0	-9.9574e+08	-2.0292e+10	-2.8171e+18
4	0.061663	0.050313	0	-2.2309e+18	-9.2647e+20	-1.7856e+37
5	0.094243	0.0073999	0	-1.1199e+37	-1.9313e+42	-7.1739e+74
6	0.11098	0.00012745	0	-2.8217e+74	-8.3923e+84	-1.158e+150
7	0.079442	3.657e-08	0	-1.7914e+149	-1.5847e+170	-3.0169e+300
8	0.02283	2.9976e-15	0	-7.2206e+298	-Inf	-Inf
9	0.0013075	0	0	-Inf	-Inf	-Inf

10	3.869e-06	0	0	-Inf	-Inf	-Inf

खमिया

1 .सबसे पहले, function के first और second derivative दोनों की आवश्यकता है।

2. न्यूटन चरण की दिशा(newton direction) में objective को बढ़ाने के लिए संभव नहीं हो सकता है। यह केवल तभी guaranteed है जब $f''(x)$ negative definite हो| इस कारण से, Newton-Raphson तकनीक बहुत कम ही उपयोग में लायी जाती है और तब, यदि objective, globally concave हो|

External links (संदर्भ)

1. Mangasarian, O. L. (2002). A finite Newton method for classification. *Optimization Methods and Software*, *17*(5), 913-929.
2. Fung, G., & Mangasarian, O. L. (2003). Finite Newton method for Lagrangian support vector machine classification. *Neurocomputing*, *55*(1-2), 39-55.
3. https://en.wikipedia.org/wiki/Newton%27s_method_in_optimization
4. https://ocw.mit.edu/courses/sloan-school-of-management/15-084j-nonlinear-programming-spring-2004/lecture-notes/lec3_newton_mthd.pdf
5. http://web.mit.edu/~jadbabai/www/EE605/lectures/unconstrained.pdf
6. Zheng, S. (2018). A fast-iterative algorithm for support vector data description. *International Journal of Machine Learning and Cybernetics*, 1-15.