Лабораторная работа №1. Часть I
Методы одномерного поиска экстремума

Цель работы

Ознакомиться с методами одномерного поиска, используемыми в многомерных методах минимизации функций переменных. Сравнить различные алгоритмы по эффективности на тестовых примерах.

Методические указания

1. Общая схема методов поиска минимума на отрезке

Пусть функция $f(x)$ унимодальна на отрезке $[a_0, b_0]$ (т.е. имеет только один минимум). Необходимо найти точку минимума функции на этом отрезке с заданной точностью ε. Все методы одномерного поиска строятся на последовательном уменьшении интервала, содержащего точку минимума.

Возьмем внутри отрезка $[a_0, b_0]$ две точки x_1 и x_1: $a_0 < x_1 < x_2 < b_0$, и вычислим значения функции в этих точках. Из свойства унимодальности функции можно сделать вывод о том, что минимум расположен либо на отрезке $[a_0, x_2]$, либо на отрезке $[x_1, b_0]$. Действительно, если $f(x_1) < f(x_2)$, то минимум не может находиться на отрезке $[x_2, b_0]$, а если $f(x_1) > f(x_2)$, то минимум не может находиться на отрезке $[a_0, x_1]$. Если же $f(x_1) = f(x_2)$, то минимум находится на интервале $[x_1, x_2]$.

Алгоритм заканчивается, когда длина интервала, содержащего минимум, становится меньше ε. Различные методы одномерного поиска отличаются выбором точек x_1 и x_1. Обычно сравнивают число вычислений функции, необходимое для достижения заданной точности.

2. Метод дихотомии (деления отрезка пополам)

Точки x_1 и x_1 выбираются на расстоянии $\delta < \varepsilon/2$ от середины отрезка:

$$x_1 = (a_i + b_i)/2 - \delta, x_2 = (a_i + b_i)/2 + \delta. \tag{1}$$

За одну итерацию интервал неопределенности уменьшается примерно в два раза (см. рис. 1). Значит, за n итераций длина интервала будет примерно равна $(b_0 - a_0)/2^n$. Для достижения точности ε потребуется $ln((b_0 - a_0)\varepsilon)/ln2$ итераций. На каждой итерации функция вычисляется два раза.

3. Метод золотого сечения

Точки x_1 и x_1 находятся симметрично относительно середины отрезка $[a_0, b_0]$ и делят его в пропорции золотого сечения (длина всего отрезка относится к длине большей его части также, как длина большей части относится к длине меньшей части):

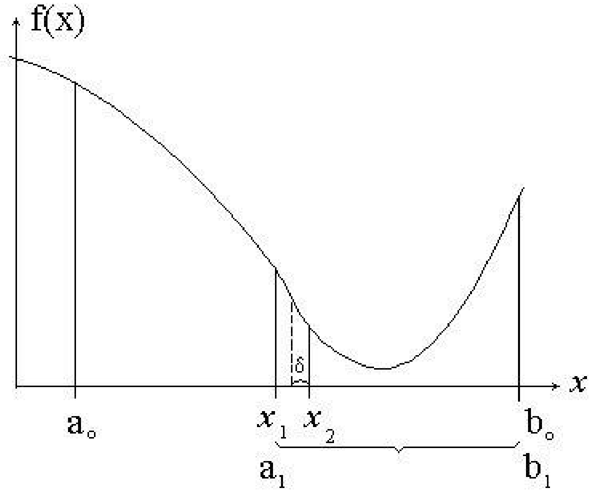

Рис. 1. Метод дихотомии

$$\frac{b_0 - a_0}{b_0 - x_1} = \frac{b_0 - x_1}{x_1 - a_0}, \tag{2}$$

$$\frac{b_0 - a_0}{x_2 - a_0} = \frac{x_2 - a_0}{b_0 - x_2}. \tag{3}$$

Отсюда

$$x_1 = a_i + \frac{3 - \sqrt{5}}{2}\left(b_i - a_i\right) = a_i + 0.381966011 * \left(b_i - a_i\right), \tag{4}$$

$$x_2 = a_i + \frac{\sqrt{5} - 1}{2}\left(b_i - a_i\right) = a_i + 0.618003399 * \left(b_i - a_i\right). \tag{5}$$

За одну итерацию интервал неопределенности уменьшается в $\frac{\sqrt{5}+1}{2} = 1.618...$ раз, однако на следующей итерации мы будем вычислять функцию только один раз, так как по свойству золотого сечения $\frac{x_2 - x_1}{b - x_1} = 0.381...$ и $\frac{b - x_2}{b - x_1} = 0.381....$ (см. рис. 2). Для достижения точности ε потребуется $\frac{ln((b_0 - a_0)/\varepsilon)}{ln 1.618}$ итераций.

Неточное задание величины $\sqrt{5}$ на ЭВМ уже при достаточно небольшом количестве итераций может приводить к погрешностям и потере точки минимума (она выпадает из интервала неопределенности). Поэтому, вообще говоря, при реализации алгоритма возможность такой ситуации должна быть предусмотрена.

4. Метод Фибоначчи

Числа Фибоначчи определяются соотношениями:

$$F_{n+2} = F_{n+1} + F_n, n = 1, 2..., F_1 = F_2. \tag{6}$$

С помощью индукции можно показать, что -е число Фибоначчи представимо в виде (формула Бинэ):

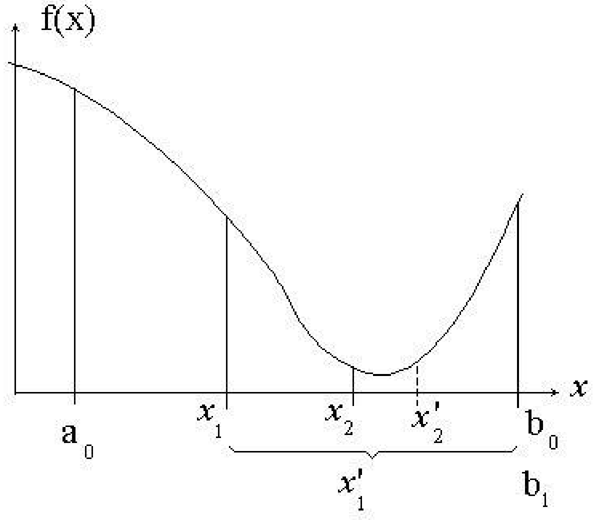

Рис. 2. Метод золотого сечения

$$F_n = \frac{1}{\sqrt{5}}\left[\left((1+\sqrt{5})/2\right)^n - \left((1-\sqrt{5})/2\right)^n\right], n = 1, 2... \tag{7}$$

Из этой формулы видно, что при больших n: $F_n \approx \left((1+\sqrt{5})/2\right)^n/\sqrt{5}$, так что числа Фибоначчи с увеличением растут очень быстро.

На начальном интервале вычисляют точки

$$x_1 = a_0 + \frac{F_n}{F_{n+2}}(b_0 - a_0), \tag{8}$$

$$x_2 = a_0 + \frac{F_{n+1}}{F_{n+2}}(b_0 - a_0), \tag{9}$$

где выбирается по формуле (18), исходя из точности и начальной длины интервала.

На k-м шаге метода будет получена тройка чисел $a_k, b_k, \bar{x_k}$, локализирующая минимум $f(x)$, такая, что

$$\Delta_k = b_k - a_k = (b_0 - a_0)\frac{F_{n-k+3}}{F_{n+2}}, 1 \leq k \leq n, a_1 = a_0, b_1 = b_0, \tag{10}$$

а точка $\bar{x_k}, a_k < \bar{x_k} < b_k$ с вычисленным значением

$$f(\bar{x_k}) = min_{1 \leq i \leq k} f(x_i), \tag{11}$$

совпадает с одной из точек

$$x_1 = a_k + \frac{F_{n-k+1}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+1}}{F_{n+2}}(b_0 - a_0), \tag{12}$$

$$x_2 = a_k + \frac{F_{n-k+2}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+2}}{F_{n+2}}(b_0 - a_0)\,, \tag{13}$$

расположенных на отрезке $[a_k, b_k]$ симметрично относительно его середины. При k=n процесс заканчивается. В этом случае длина отрезка

$$\Delta_n = b_n - a_n = (b_0 - a_0)/F_{n+2}, \tag{14}$$

а точки

$$x_n = a_n + \frac{F_1}{F_{n+2}}(b_0 - a_0)\,, \tag{15}$$

$$x_n = a_n + \frac{F_2}{F_{n+2}}(b_0 - a_0)\,, \tag{16}$$

совпадают и делят отрезок пополам.

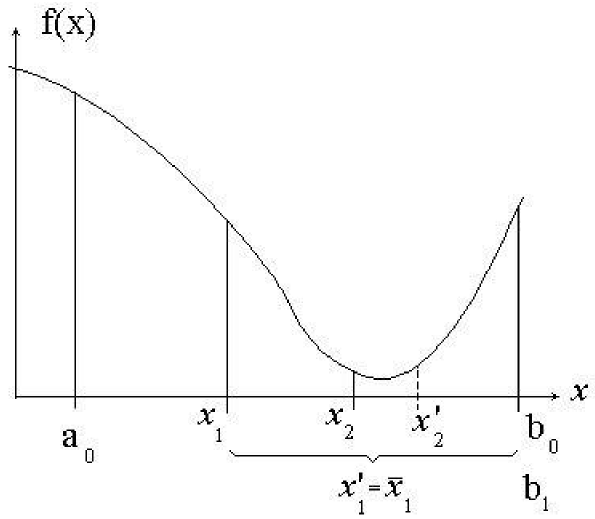

Рис. 3. Метод Фибоначчи

Следовательно

$$\frac{b_n - a_n}{2} = \frac{b_0 - a_0}{F_{n+2}} < \varepsilon. \tag{17}$$

Отсюда можно выбрать n из условия

$$\frac{b_0 - a_0}{\varepsilon} < F_{n+2}. \tag{18}$$

С ростом n, из-за того, что F_n/F_{n+2} – бесконечная десятичная дробь, происходит искажение метода. Поэтому на очередном шаге в качестве новой точки берут из (12) наиболее удалённую от $\bar{x_{k-1}}$ на предыдущем шаге.

5. Поиск минимума функции на прямой

В рассмотренных методах требуется знать начальный отрезок, содержащий точку минимума. Поиск на прямой заключатся в том, что осуществляются возрастающие по величине шаги, до тех пор, пока не будет пройдена точка минимума функции (т.е. убывание функции сменится на возрастание). На первом шаге выбираем начальную точку x_0 и определяем направление убывания функции.

Шаг 1. Если $f(x_0) > f(x_0 + \delta)$, то полагаем $x_1 = x_0 + \delta$, $h = \delta$. Иначе, если $f(x_0) > f(x_0 - \delta)$, то ,$x_1 = x_0 - \delta$, $h = -\delta$.

Шаг 2. Удваиваем h и вычисляем $x_{k+1} = x_k + h$.

Шаг 3. Если $f(x_k) > f(x_{k+1})$, то полагаем $k = k + 1$ и переходим к шагу 2. Иначе – поиск прекращаем, т.к. отрезок $[x_{k-1}, x_{k+1}]$ содержит точку минимума.

6. Поиск минимума функции n переменных в заданном направлении

Пусть требуется найти минимум функции n переменных $f(\bar{x})$, где $\bar{x} = (x_1, x_2, ..., x_n)$, в направлении вектора $\bar{s}$. Для этого нужно найти минимум функции $g(\lambda) = f(\bar{x} + \lambda\bar{s})$ рассмотренными выше методами, λ – величина шага в заданном направлении.

Порядок выполнения работы

1. Реализовать данные три метода, иследовать сходимость алгоритмов и провести их сравнение по числу вычислений функции для достижения заданной точности.

2. Реализовать алгоритм поиска минимума функции на прямой.

3. Реализовать данные алгоритмы в виде процедуры, предназначенной для минимизации функции переменных в направлении заданного вектора.

Варианты заданий

1. $f(x) = \sin(x), x \in [-\pi/2, \pi/2]$
2. $f(x) = \cos(x), x \in [0, \pi]$
3. $f(x) = (x-2)^2, x \in [-2, 20]$
4. $f(x) = (x-15)^2 + 5, x \in [2, 200]$
5. $f(x) = (x+5)^4, x \in [-10, 15]$
6. $f(x) = \exp(x), x \in [0, 100]$
7. $f(x) = x^2 + 2x - 4, x \in [-10, 20]$
8. $f(x) = x^3 - x, x \in [0, 1]$

Содержание отчета

Отчет должен содержать таблицы с результатами исследований по каждому методу, где должны быть отражены исходный и последующие интервалы, соотношение их длин, вычисляемые на них точки и значения функций в них.

Построить график зависимости количества вычислений минимизируемой функции от логарифма задаваемой точности ε.

По всем пунктам задания требуется сделать выводы.

Контрольные вопросы

1. Метод дихотомии.
2. Метод золотого сечения.
3. Метод Фибоначчи.
4. Метод квадратичной интерполяции (метод парабол).

Лабораторная работа №1. Часть II
Методы многомерного поиска экстремума. Методы первого порядка: метод наискорейшего спуска

Цель работы

Ознакомиться с методами поиска минимума функции nпеременных в оптимизационных задачах без ограничений.

Методические указания

1. Методы первого порядка

К методам первого порядка относятся методы, использующие производные первого порядка для выбора направления спуска: метод наискорейшего спуска; метод сопряженных градиентов в модификации Данилина-Пшеничного (Полака-Рибьера); метод сопряженных градиентов в модификации Флетчера-Ривса.

Пусть дана функция $f(x)$, где $\bar{x} = (x_1, x_2, ..., x_n)^T$, и задана начальная точка $\bar{x_0}$. Требуется найти минимум функции $f(\bar{x})$ с точностью ε_f – по функции, ε_i – по переменным x_i , где $i = 1, ..., n$.

На k-м шаге $(k > 0)$ определяем вектор $\bar{s_k}$, в направлении которого функция $f(\bar{x})$ уменьшается. В этом направлении делаем шаг величиной λ_k и получаем новую точку $x^{\bar{k+1}} = x^{\bar{k+1}} + \lambda_k s^{\bar{k}}$, в которой $f(\bar{x}^{k+1}) < f(\bar{x}^k)$. Поиск прекращаем как только $\left|f(\bar{x}^{k+1}) - f(\bar{x}^k)\right| < \varepsilon_f$ или для всех i верно $\left|\bar{x}^{k+1} - \bar{x}^k\right| < \varepsilon_i$.

Различные методы спуска отличаются выбором направления и величины шага. Как правило, для нахождения λ_k используется процедура одномерного поиска.

2. Алгоритм наискорейшего спуска

Градиент функции в любой точке показывает направление наибольшего локального увеличения $f(\bar{x})$. Поэтому при поиске минимума $f(\bar{x})$, следует двигаться в направлении $-\nabla f(\bar{x})$ – направлении наискорейшего спуска.

Итерационная формула процесса наискорейшего спуска имеет вид

$$\bar{x}^{k+1} = \bar{x}^k - \lambda^k \cdot \nabla f(\bar{x}^k), \tag{19}$$

или

$$\bar{x}^{k+1} = \bar{x}^k - \lambda^k \cdot \frac{\nabla f(\bar{x}^k)}{||\nabla f(\bar{x}^k)||} = \bar{x}^k - \lambda^k \cdot \bar{S}^k. \tag{20}$$

Очевидно, что в зависимости от выбора параметра λ траектории спуска будут существенно различаться. При большом значении λ траектория спуска будет представлять собой колебательный процесс, а при слишком больших λ процесс может расходиться. При выборе малых λ траектория спуска будет плавной, но и процесс будет сходиться очень медленно. Обычно λ выбирают из условия

$$\lambda^k = argmin_\lambda f\left(\bar{x}^k + \lambda^k \cdot \bar{S}^k.\right) \tag{21}$$

Если λ определяется в результате одномерной минимизации, то градиент в точке очередного приближения будет ортогонален направлению предыдущего спуска $\nabla f\left(\bar{x}^{k+1} \perp \bar{S}^k\right)$.

Вообще говоря, процедура наискорейшего спуска может закончиться в стационарной точке любого типа, в которой $\nabla f(\bar{x})$ в т.ч. и в седловой точке.

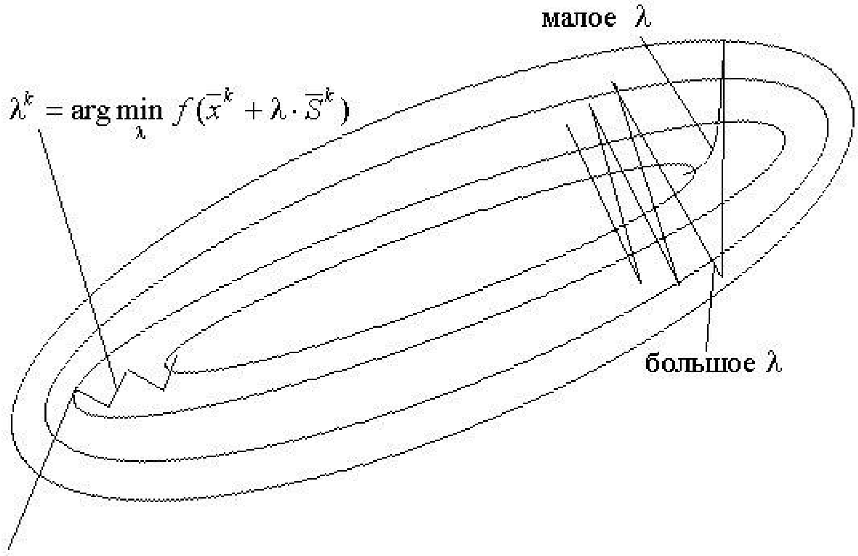

Рис. 4. Визуализация алгоритма наискорейшего спуска

Эффективность алгоритма зависит от вида минимизируемой функции. Алгоритм наискорейшего спуска сойдется за одну итерацию при любом начальном приближении для функции $f(\bar{x}) = x_1^2 + x_2^2$, но сходимость будет очень медленной для функции вида $f(\bar{x}) = x_1^2 + 100x_2^2$. Когда линии уровня функции представляет собой криволинейный "овраг эффективность алгоритма оказывается очень низкой. Очевидно, что хорошие результаты может давать предварительное масштабирование функций.

Процесс наискорейшего спуска обычно быстро сходится вдали от точки экстремума и медленно в районе экстремума. Поэтому метод наискорейшего спуска нередко используют в комбинации с другими алгоритмами.

Порядок выполнения работы

С использованием программного обеспечения исследовать алгоритм наискорейшего спуска на заданной тестовой функции, осуществляя спуск из различных исходных точек. Исследовать сходимость алгоритма, фиксируя точность определения минимума, количество итераций метода и количество вычислений минимизируемой функции в зависимости от задаваемой точности поиска. Результатом выполнения данного пункта должны быть выводы об объѐме вычислений в зависимости от задаваемой точности и начального приближения.

Сравнить результаты при различных подходах к вычислению производных - аналитическом и численном.

Варианты заданий

1. $f(\bar{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$
2. $f(\bar{x}) = (x_2 - x_1^2)^2 + (1 - x_1)^2$
3. $f(\bar{x}) = (x_2 - x_1)^2 + 100(1 - x_1)^2$
4. $f(\bar{x}) = 100(x_2 - x_1^3)^2 + (1 - x_1)^2$
5. $f(\bar{x}) = (1.5 - x_1(1 - x_2))^2 + (2.25 - x_1(1 - x_2^2))^2 + (2.625 - x_1(1 - x_2^3))^2$
6. $f(\bar{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2) + (1 - x_3)^2 + 10.2(x_2 - 1)^2 + (x_4 - 1)^2 + 19.8(x_2 - 1)(x_4 - 1)$
7. $f(\bar{x}) = (x_1 + x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$

Содержание отчета

Отчет должен содержать: титульный лист; цель работы; задание; таблицы с результатами проведенных исследований, где должны быть отражены начальное приближение x0 , задаваемая точность, количество итераций, число вычислений целевой функции, найденная точка и значение функции в ней, а также выводы о сходимости алгоритмов в зависимости от точности и начального приближения с указанием преимуществ и недостатков.

Контрольные вопросы

1. Общая характеристика методов спуска
2. Методы первого порядка
3. Математическое обоснование градиентных методов
4. Улучшения градиентных методов
5. Особенности метода наискорейшего спуска

Список литературы

[1] Аттетков, А.В., Введение в методы оптимизации. [Электронный ресурс] — Электрон. дан. — М. : Финансы и статистика, 2011. — 272 с. — Режим доступа: http://e.lanbook.com/book/53756

[2] Лесин, В.В. Основы методов оптимизации. [Электронный ресурс] / В.В. Лесин, Ю.П. Лисовец. — Электрон. дан. — СПб. : Лань, 2016. — 344 с. — Режим доступа: http://e.lanbook.com/book/86017

[3] Пантелеев, А.В. Методы оптимизации в примерах и задачах. [Электронный ресурс] / А.В. Пантелеев, Т.А. Летова. — Электрон. дан. — СПб. : Лань, 2015. — 512 с. — Режим доступа: http://e.lanbook.com/book/67460