Решение системы линейных уравнений методом наискорейшего градиентного спуска

ИВТ-22

6 мая 2025 г.

1 Описание метода

Метод наискорейшего градиентного спуска является итерационным методом решения систем линейных уравнений. Метод применяется для решения систем вида

$$Ax = b, \tag{1}$$

где A — положительно определенная симметричная матрица коэффициентов, b — вектор правых частей системы, x — вектор неизвестных.

Метод заключается в построении последовательности векторов $x_k,\ k = 0, 1, 2, \ldots$, сходящейся к вектору ξ — решению системы (1). Рассмотрим функцию

$$f(x) = (Ax, x) - 2(x, b) \tag{2}$$

Данная функция является квадратичной относительно координат вектора x, так как содержит члены второго порядка по этим координатам:

$$f(x) = \sum_i \sum_j a_{ij} x_i x_j - 2 \sum_i x_i b_i$$

Покажем, что точка минимума функции $f(x)$ существует и является решением ξ системы уравнений (1).

Так как матрица системы (1) положительно определена, то определитель матрицы системы не равен нулю. Следовательно, система имеет единственное решение. Пусть $x = \xi + \Delta$ — произвольный вектор, тогда значение функции в x равно:

$$\begin{aligned} f(x) = f(\xi + \Delta) &= (A(\xi + \Delta), \xi + \Delta) - 2(\xi + \Delta, b) \\ &= (A\xi, \xi) + (A\xi, \Delta) + (A\Delta, \xi) + (A\Delta, \Delta) - 2((\xi, b) + (\Delta, b)) \\ &= f(\xi) + (A\xi - b, \Delta) + (A\Delta, \Delta) \geq f(\xi) \end{aligned}$$

Таким образом, $f(x)$ ограничена снизу значением $f(\xi)$.

Введем понятие невязки, показывающее, насколько на i-м шаге Ax_i отличается от b, то есть

$$r_i = b - Ax_i \tag{3}$$

Найдем градиент ∇f функции $f(x)$. Пусть p — произвольный единичный вектор. Тогда,

$$f(x + tp) = f(x) + 2tf(Ax - b, p) + t^2(Ap, p).$$

Найдем значение, при котором производная достигает максимума, то есть найдем производную функции одной переменной $f(t)$, при условии, что $t = 0$:

$$\frac{df(x + tp)}{dt}\Big|_{t=0} = 2(Ax - b, p) = 2|Ax - b| \cos\varphi, \tag{4}$$

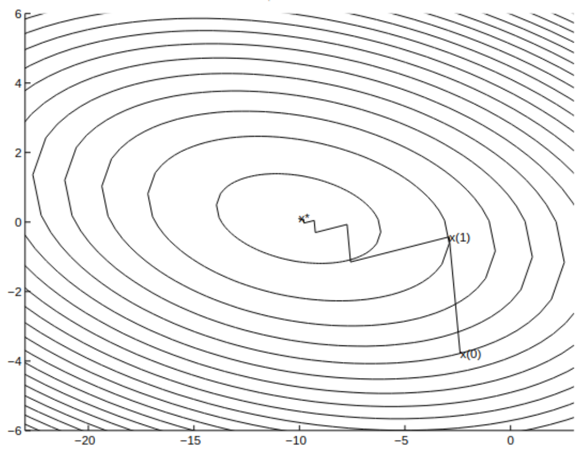
где φ — угол между $Ax - b$ и p, и заметим, что при $\varphi = 0$ производная достигает максимального значения, то есть, когда векторы p и $Ax - b$ сонаправлены. Таким образом, градиент функции:

$$\nabla f = Ax - b = -r.$$

Введем итерационный процесс:

$$x_{i+1} = x_i + \alpha_i \cdot p_i = x_i + \alpha_i \nabla f = x_i + \alpha_i r_i \tag{5}$$

Функция $f(x)$ убывает при движении в направлении вектора r_i от точки x_i до тех пор, пока луч $x_i + t \cdot r_i,\ t > 0$, не достигнет более низкого множества уровня функции $f(x)$. В такой точке касания x_{i+1} функция $f(x)$ достигает минимального значения локально, при этом $f(x_i + 1) < f(x_i)$.



Для определения коэффициента шага в (5) найдем минимум функции $f(x_i + \alpha_i r_i)$:

$$\frac{df(x_i + \alpha_i r_i)}{d\alpha_i} = 0$$

$$d(f(x_i) + 2\alpha_i(Ax_i, r_i) + t^2(Ar_i, r_i) - 2\alpha_i(b, r_i)) = 0$$

$$2(Ax_i - b, r_i) + 2\alpha_i(Ar_i, r_i) = 0$$

$$\alpha_i = \frac{(r_i, r_i)}{(Ar_i, r_i)}$$

Если матрица не симметрична, то части равенства $Ax = b$ можно домножить на транспонированную матрицу A^T:

$$A^T Ax = A^T b,$$

получив равносильную систему, матрица которой симметрична и положительно определена.

2 Алгоритм

Объединив полученные формулы, имеем:

$$r_i = b - Ax_i \tag{6}$$

$$\alpha_i = \frac{(r_i, r_i)}{(Ar_i, r_i)} \tag{7}$$

$$x_{i+1} = x_i + \alpha_i \cdot r_i \tag{8}$$

Данный рекуррентный алгоритм требует два произведения матрица-вектор за итерацию. Вычислительная сложность алгоритма, зависящая от этой операции, может быть упрощена, если в уравнении (8) умножить обе части равенства на $-A$ и прибавить b:

$$r_{i+1} = r_i - \alpha_i A r_i. \tag{9}$$

Произведение Ar, которое используется в уравнениях (7) и (9), можно вычислить только один раз. Недостатком такой оптимизации является то, что в уравнении (9) не используется значение x_i, то есть отсутствует обратная связь по x. Таким образом накопление ошибки округления может привести к сходимости последовательности x_i к точке, не совпадающей с решением системы уравнений. Этого можно избежать, если раз в определенное число итераций пересчитывать Ax.

Список литературы

[1] https://books-vuzi.narod.ru/olderfiles/1/VICH_MATH_MAXIMA.PDF

[2] https://users.cs.utah.edu/~haocheng/notes/NoteonConjugateGradientMethod.pdf

[3] https://antonleykin.math.gatech.edu/math2605fall10/PROJECTS/SteepestDescent.pdf

[4] https://ocw.mit.edu/courses/18-409-topics-in-theoretical-computer-science-an-algorithmists-toolki
68f1b1ba4d419ee8f3b67a41ebf258e8_MIT18_409F09_scribe21.pdf

[5] https://antonleykin.math.gatech.edu/math2605fall10/PROJECTS/SteepestDescent.pdf

[6] https://www.cs.umd.edu/users/oleary/a600/cgnotes.pdf

[7] https://people.eecs.berkeley.edu/~satishr/cs270/sp11/rough-notes/Linear-Equations.pdf

[8] https://www.phys.uconn.edu/~rozman/Courses/m3511_18s/downloads/steepest-descent.pdf