СОДЕРЖАНИЕ

1 Методы локальной оптимизации

Оптимизация – это задача нахождения экстремума (минимума или максимума) целевой функции в некоторой области конечномерного векторного пространства, ограниченной набором линейных и/или нелинейных равенств и/или неравенств.

Во многих практически важных случаях для целевой функции многих переменных $f(\boldsymbol{x})$ задача оптимизации может быть сформулирована в виде:

$$f(\boldsymbol{x}) \to \min,$$

где $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$ – вектор неизвестных (управляющих параметров); min – минимальное значение функции в ограниченной или неограниченной области изменения неизвестных.

Для нахождения абсолютного минимума целевой функции $f(\boldsymbol{x})$ существует только один способ: найти все локальные минимумы этой функции, сравнить их и выбрать из них тот, в котором функция принимает наименьшее значение.

1.1 Минимум функции одного переменного

Для функции одной переменной $f(x)$, задача нахождения минимума эквивалента задачи нахождения корней уравнения:

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = 0 \qquad (1)$$

Эта одномерная задача нередко возникает в практических приложениях. Кроме того, большинство методов решения многомерных задач сводится к поиску одномерного минимума.

Предположим, что $f(x)$ задана и кусочно-непрерывна на отрезке $x \in [a, b]$, и имеет на этом отрезке (включая его концы) только один локальный минимум. Построим итерационный процесс, сходящийся к этому минимуму.

Вычислим значение функции на концах отрезка $x = a$ и $x = b$, а также в двух внутренних точках $x_1 < x_2$. Так так функция $f(x)$ имеет минимум на отрезке $x \in [a, b]$, то справедливо утверждение:

$$f(a) \geqslant f(x_1), \quad f(x_2) \leqslant f(b)$$

Сравним все четыре значения функции между собой $f(a)$, $f(x_1)$, $f(x_2)$ и $f(b)$ и выберем среди них наименьшее.

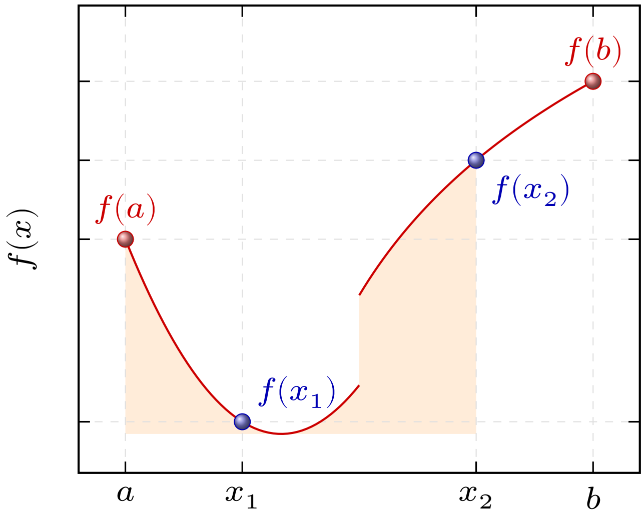


Рисунок 1 – График кусочно-непрерывной функции $y = f(x)$,
имеющей минимум на отрезке $x \in [a, b]$

Из рисунка **??** видно, что наименьшее значение функция достигает в точке $x = x_1$:

$$f(x_1) < f(a) < f(x_2) < f(b)$$

Очевидно, что минимум функции $f(x)$ расположен в одном из прилегающих к точке $x = x_1$ отрезков, то есть минимум находится либо в пределах отрезка $[a, x_1]$, либо в $[x_1, x_2]$ (рисунок **??**, выделенная область).

Поэтому на первом шаге итерационного процесса отбрасывается отрезок $[x_2, b]$, и для поиска минимума функции $f(x)$ рассматривается отрезок $[a, x_2]$, при этом область поиска минимума функции сужается:

$$|a - x_2| < |a - b|\,, \quad \text{так как} \quad x_2 < b.$$

Полагая $b = x_2$, на новом отрезке $[a, b]$ вновь необходимо выбрать две внутренние точки, вычислить в них и на концах отрезка значения функции $f(x)$, и сделать следующий шаг итерационного процесса.

Критерием остановки итерационного процесса является условие выполнения неравенства, которое гарантирует малость размера области поиска минимума по сравнению с заранее заданной погрешность метода:

$$(b-a) \leqslant \epsilon,$$

где ϵ – погрешность метода.

Симметричный метод поиска минимума функции одной переменной $f(x)$ основан на выборе внутренних точек x_1 и x_2 отрезка $[a, b]$, которые равноудалены от концов этого отрезка. Например, если точки x_1 и x_2 делят отрезок $[a, b]$ на три равные части (рисунок **??**), то координаты этих точек могут быть определены из соотношений:

$$x_1 = a + \frac{b-a}{3} = \frac{2a+b}{3}, \quad x_2 = b - \frac{b-a}{3} = \frac{a+2b}{3}.$$

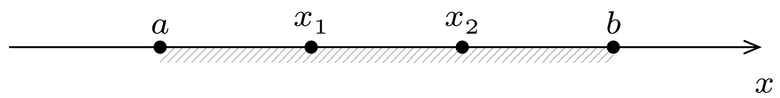


Рисунок 2 – Схематическое изображение точек деления отрезка $[a, b]$

Оценка длины отрезка после первого итерационного шага составит:

$$\ell_1 = (b-a) - \frac{b-a}{3} = \frac{2}{3} \cdot (b-a),$$

после второго шага:

$$\ell_2 = \ell_1 - \frac{\ell_1}{3} = \frac{2}{3} \cdot \ell_1 = \left(\frac{2}{3}\right)^2 \cdot (b-a),$$

а после k-ого итерационного шага:

$$\ell_k = \left(\frac{2}{3}\right)^k \cdot (b-a).$$

Таким образом, чтобы погрешность вычисления ℓ_k была менее ϵ, для числа итераций k справедлива оценка:

$$\left(\frac{2}{3}\right)^k \cdot (b-a) \leqslant \epsilon \quad \rightarrow \quad k = \left\lceil \frac{\ln(b-a) - \ln(\epsilon)}{\ln(3) - \ln(2)} \right\rceil$$

Симметричный метод поиска минимума функции является аналогом метода дихотомии для нахождения корня уравнения $f(x) = 0$. Метод применим к недифференцируемым функциям и всегда сходится. Следует отметить, что если на отрезке $[a, b]$ функция $f(x)$ имеет несколько локальных минимумов, то итерационный процесс сойдется к одному из этих минимумов, но не обязательно к наименьшему.

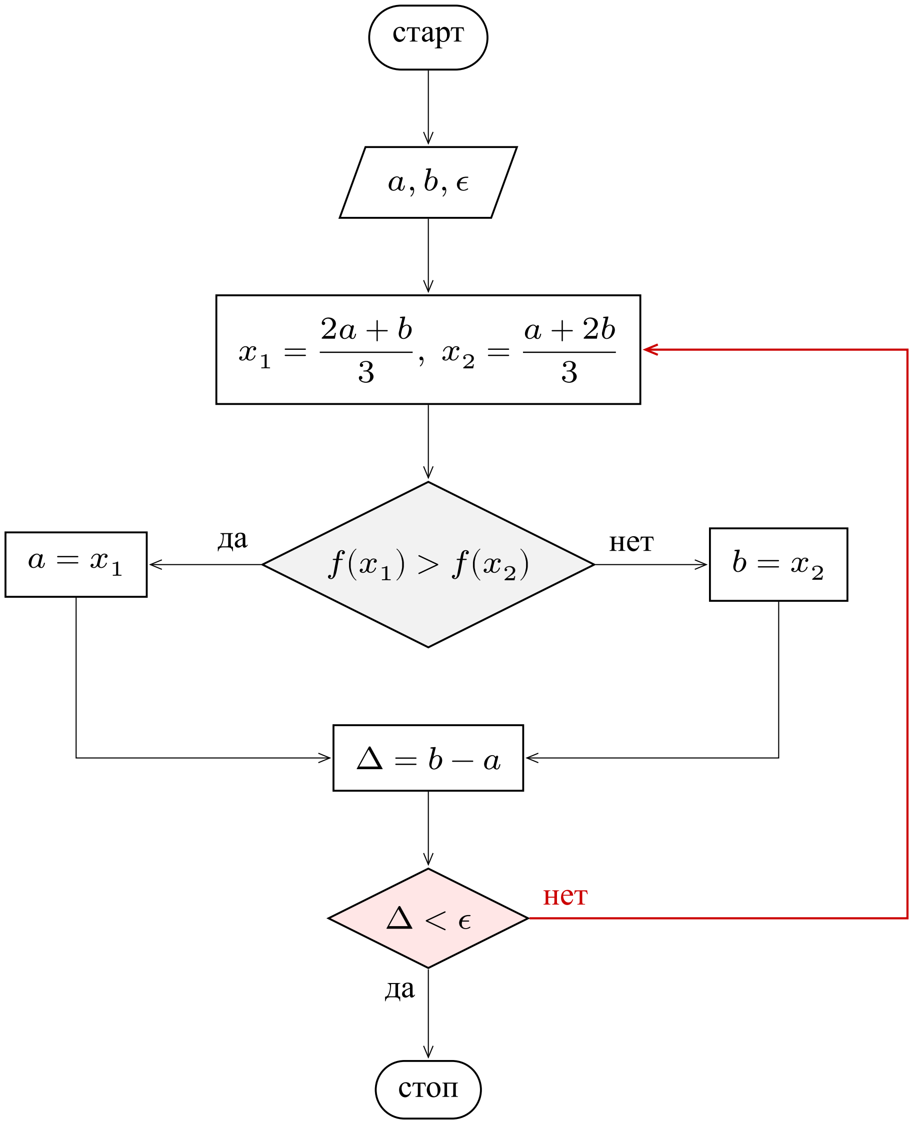


Рисунок 3 – Блок-схема алгоритма нахождения минимума функции $f(x)$ одного переменного

1.2 Минимизация функций многих переменных

Задача безусловной минимизации (оптимизации) состоит в нахождении минимума или максимума функции $f(\boldsymbol{x})$ в отсутствие каких-либо ограничений на область изменения переменных задачи $\boldsymbol{x} \in (-\infty, +\infty)$.

Большинство практических задач оптимизации содержит ограничения, которые обусловлены технико-экономическим смыслом решаемой задачи, однако многие алгоритмы решения задач с ограничениями предполагают сведение ее к последовательности задач безусловной оптимизации.

1.2.1 Спуск по координатам

Идея метода по координатного спуска заключается в том, что задача поиска минимума функции $f(x_1, x_2, \dots, x_n) \to \min$ разделяется на серию задач *одномерной минимизации* этой функции вдоль направления одной из координатных осей:

$$
\begin{aligned}
&f(x_1, x_2 = \text{const}, \dots, x_n = \text{const}) \to \min \\
&f(x_1 = \text{const}, x_2, \dots, x_n = \text{const}) \to \min \\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
&f(x_1 = \text{const}, x_2 = \text{const}, \dots, x_n) \to \min
\end{aligned}
$$

1) Выбирают нулевое приближение $\boldsymbol{x}_0 = (x_{10}, x_{20}, \dots, x_{n0})$.
2) Фиксируют (считают постоянными) значения всех координат кроме x_1. Тогда функция $f(\boldsymbol{x})$ будет зависеть только от одной переменной x_1:

$$\phi_1(x_1) = f(x_1, x_{20} = \text{const}, \dots, x_{n0} = \text{const})$$

3) Используя метод одномерной минимизации, находится минимум функции одной переменной $\phi_1(x_1) \to \min$, который можно обозначить через m_1.
4) Сделан переход из начальной точки $\boldsymbol{x}_0$ в точку “частного“ минимума по направлению, параллельному оси x_1:

$$(x_{10}, x_{20}, \dots, x_{n0}) \to (m_1, x_{20}, \dots, x_{n0}),$$

и значение функции уменьшается:

$$f(x_{10}, x_{20}, \dots, x_{n0}) > f(m_1, x_{20}, \dots, x_{n0})$$

5) Затем из новой точки $(m_1, x_{20}, \dots, x_{n0})$ осуществляется спуск по направлению, параллельному оси x_2, то есть находится минимум функции:

$$\phi_2(x_2) = f(m_1 = \text{const}, x_2, \dots, x_n = \text{const}),$$

который обозначим m_2.

6) Таким образом, сделан переход во вторую точку “частного“ минимума по направлению, параллельному оси x_2:

$$(m_1, x_{20}, \dots, x_{n0}) \to (m_1, m_2, \dots, x_{n0}),$$

и значение функции уменьшается:

$$f(m_1, x_{20}, \dots, x_{n0}) > f(m_1, m_2, \dots, x_{n0})$$

7) Процесс спуска по координатам повторяется для всех переменных задачи $x_3, x_4, \dots, x_n$, а приход в точку $\boldsymbol{x}_m = (m_1, m_2, \dots, m_n)$ завершает цикл спусков.
8) Конечную точку цикла спусков можно принять за нулевое приближение $\boldsymbol{x}_0 = \boldsymbol{x}_m$ и повторить цикл спусков по координатам $x_1, x_2, \dots, x_n$ до тех пока, не выполнено условие останова итерационного процесса.

Практически можно задать некоторое число $\varepsilon > 0$, связанное с выбранной точностью вычислений, и проводить итерации до тех пор, пока на k-ой итерации не будут выполнены одно или несколько неравенств вида:

$$\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\| < \varepsilon_1, \quad \|f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k-1})\| < \varepsilon_2 \tag{2}$$

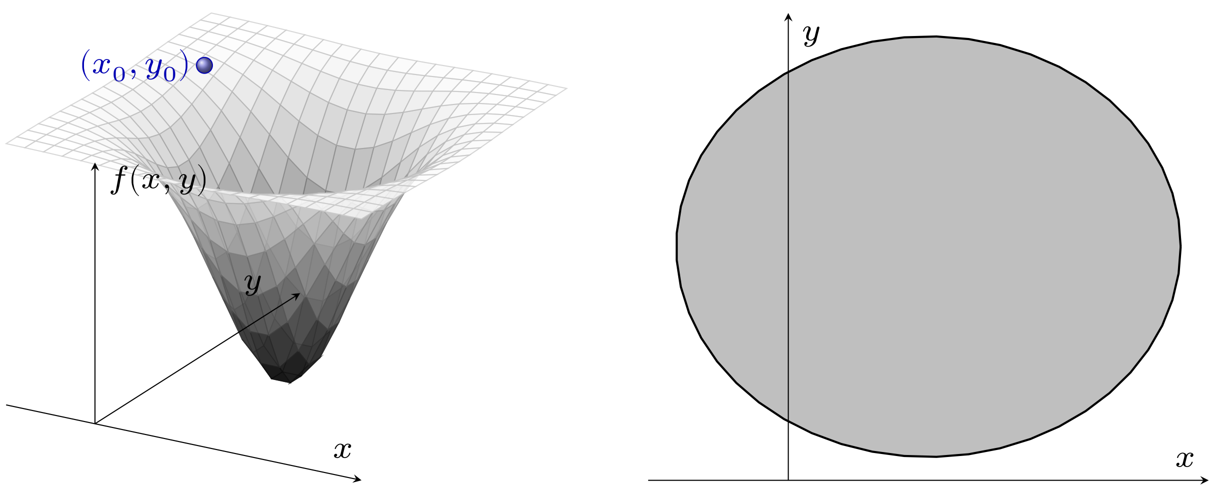


Рисунок 4 – График функции $f(x, y) = 1 - x \cdot e^{-(x-1)^2 - y^2}$

1.2.2 Метод градиентного спуска

Градиентный спуск – метод нахождения локального экстремума (минимума или максимума) функции многих переменных $f(x_1, x_2, \ldots, x_n)$ с помощью движения вдоль градиента этой функции. Это наиболее простой в реализации из всех методов локальной оптимизации, но имеет относительно малую (линейную) скорость сходимости.

Градиент ∇ это вектор, указывающий направление наибольшего возрастания некоторой функции f, значение которой меняется от одной точки пространства к другой (скалярного поля), а по величине (модулю) равный скорости роста этой величины в этом направлении. Компонентами вектора градиента являются частные производные f по всем её аргументам:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}\right) \tag{3}$$

Для случая трёхмерного пространства градиентом скалярной функции $f(x, y, z)$ называется векторная функция:

$$\operatorname{grad} f = \nabla f,$$

где ∇ – векторный дифференциальный оператор набла, компоненты которого являются частными производными по координатам:

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)$$

Следует отметить, что оператор набла не принадлежит тому же пространству, что и обычные векторы, а говоря точнее, скалярное и векторное произведение для него определено с некоторыми отличиями. Оператор ∇ действует на те скалярные поля, что стоят от него справа, и не действует на стоящие от него слева. Поэтому скалярное и векторное произведение с участием ∇ *не коммутативны* и не антикоммутативны, как это свойственно для таких произведений обычных векторов.

Минимизация целевой функции $f(\boldsymbol{x})$ сводится к итерационному процессу последовательного выбора нового вектора неизвестных $\boldsymbol{x}_{k+1}$, такого чтобы значение функции в новой точки было меньше чем в предыдущих:

$$f(\boldsymbol{x}_0) > f(\boldsymbol{x}_1) > \ldots > f(\boldsymbol{x}_k) > f(\boldsymbol{x}_{k+1}) > \ldots$$

Предполагая, что новый вектор неизвестных мало отличается от предыдущего ($\boldsymbol{x}_{k+1} - \boldsymbol{x}_k \approx 0$), можно воспользоваться линейным приближением для разложения в ряд Тейлора целевой функции:

$$f(\boldsymbol{x}_{k+1}) = f(\boldsymbol{x}_k) + \left(\nabla f(\boldsymbol{x}_k), \boldsymbol{x}_{k+1} - \boldsymbol{x}_k\right), \tag{4}$$

где k – номер итерационного шага процесса; $\boldsymbol{x}_k$ – значение неизвестных на k-ой итерации.

Если в качестве нового вектора неизвестных выбрать:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \lambda \cdot \nabla f(\boldsymbol{x}_k), \tag{5}$$

то из (**??**) получим:

$$f(\boldsymbol{x}_{k+1}) = f(\boldsymbol{x}_k) - \lambda \cdot \|\nabla f(\boldsymbol{x}_k)\|^2 \quad \rightarrow \quad f(\boldsymbol{x}_{k+1}) \leqslant f(\boldsymbol{x}_k) \tag{6}$$

где $\lambda > 0$ – малое положительное число (параметр метода), имеющий смысл скорости градиентного спуска; $\|\nabla f(\boldsymbol{x}_k)\| \geqslant 0$ – норма вектора градиента (неотрицательное число):

$$\|\nabla f\| = \sqrt{(\nabla f, \nabla f)}$$

Таким образом, выбор нового вектора неизвестных $\boldsymbol{x}_{k+1}$ в соответствии с выражением (**??**), гарантирует монотонное убывание целевой функции $f(\boldsymbol{x})$ в каждой итерации. Поэтому основная идея метода градиентного спуска заключается в том, чтобы последовательно идти в направлении наибольшего уменьшения целевой функции, которое задаётся антиградиентом $-\nabla f(\boldsymbol{x})$.

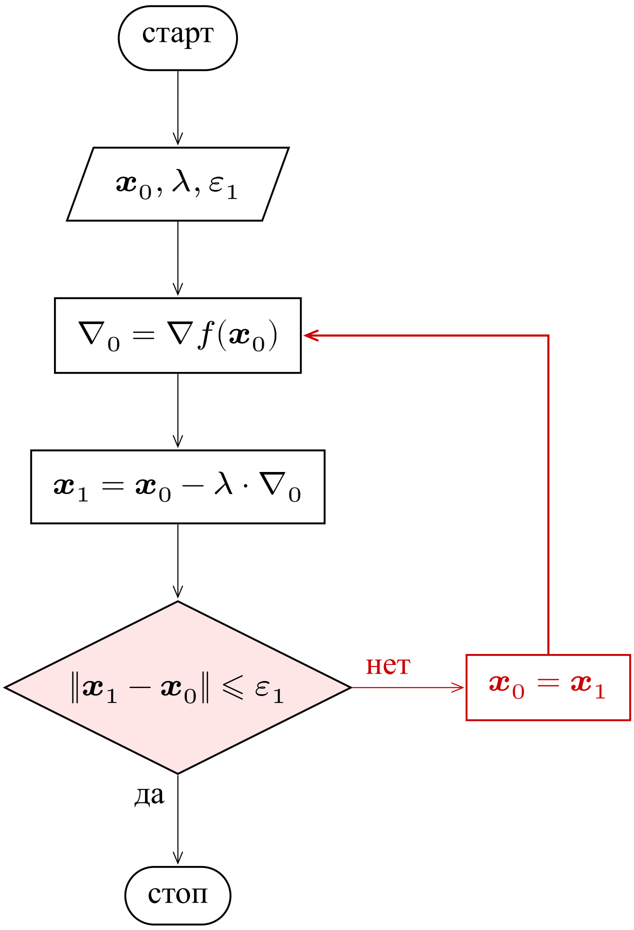


Рисунок 5 – Блок-схема алгоритма нахождения минимума функции $f(\boldsymbol{x})$ многих переменных методом градиентного спуска

1.2.3 Метод тяжелого шара

Поиск минимума функции многих переменных $f(\boldsymbol{x})$ методом "тяжелого шара" основан на аналогии движения материальной частицы массой m в консервативном силовом поле $\boldsymbol{F}(\boldsymbol{x})$ в вязкой среде.

В соответствии с принципом минимальной энергии тело смещается в положение, которое минимизирует общую потенциальную энергию системы $f(\boldsymbol{x}) \to \min$. Поэтому если предположить, что функция $f(\boldsymbol{x})$ является потенциальной энергией частицы в консервативном силовом поле $\boldsymbol{F}(\boldsymbol{x}) = -\nabla f(\boldsymbol{x})$, и частица перемещается в пространстве $\boldsymbol{x}$ минимизируя свою энергию, то уравнение движения этой частицы можно записать в виде:

$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{v} \\ m\,\dfrac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{F} - \alpha \cdot \boldsymbol{v} \end{cases} \tag{7}$$

где $\boldsymbol{x}$ – положение частицы в выбранной системе координат; $\boldsymbol{v}$ и α – скорость и коэффициент вязкого трения частицы в среде, соответственно.

Этот метод используется в методе стохастического градиентного спуска и в качестве расширения алгоритмов обратного распространения ошибок для обучения искусственных нейронных сетей.

Поиск минимума данным методом начинается с задания начальных условий, которые, как правило, формулируются в виде:

$$\begin{cases} \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{v}(0) = \boldsymbol{v}_0 \end{cases}, \tag{8}$$

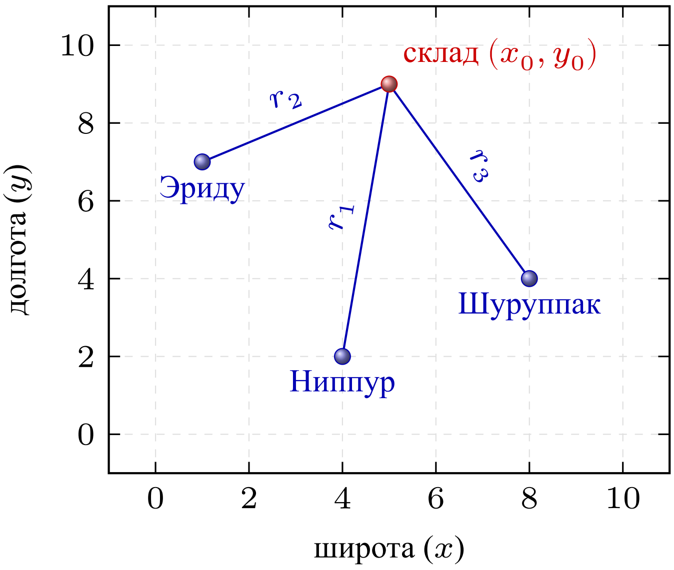
где $\boldsymbol{x}_0$ – начальное приближения для поиска минимума функции; $\boldsymbol{v}_0$ – “начальная скорость“ в пространстве неизвестных.

Масса частицы m и коэффициент вязкого трения α являются эвристическими параметрами метода и выбираются произвольным образом, отражающим специфику решаемой задачи.

1.3 Поиск оптимального положения склада

Рассмотрим задачу по нахождению географических координат расположения склада готовой продукции, таких чтобы суммарное расстояние от склада до потребителей продукции было минимальным.

Известны географические координаты трех городов: Ниппур $(4, 2)$, Эриду $(1, 7)$ Шуруппак $(8, 4)$.



1) Обозначим неизвестные:
   x – географическая широта положения склада;
   y – географическая долгота положения склада.
2) Целевая функция – суммарное расстояние от склада до всех магазинов:

$$f = r_1 + r_2 + r_3,$$

   где r_1, r_2 и r_3 – расстояние от *склада* до городов Ниппур, Эриду и Шуруппак, соответственно.
3) Поверхность планеты Земля будем считать “плоской“ в пределах области поиска положения Склада. Поэтому для нахождения расстояния от склада до каждого города воспользуемся теоремой Пифагора Самосского:

$$r_1 = \sqrt{(x_1 - x)^2 + (y_1 - y)^2}$$
$$r_2 = \sqrt{(x_2 - x)^2 + (y_2 - y)^2}$$
$$r_3 = \sqrt{(x_3 - x)^2 + (y_3 - y)^2},$$

   где x_1 и y_1 – географическая широта и долгота города Ниппур; x_2 и y_2 – географическая широта и долгота города Эриду; x_3 и y_3 – географическая широта и долгота города Шуруппак.
   Таким образом, целевая функция – суммарное расстояние от склада до всех городов, с учетом данных задания о географических координатах городов

(Ниппур, Эриду и Шуруппак), запишется в виде:

$$\begin{aligned} f(x,y) = & \sqrt{(4-x)^2+(2-y)^2} \\ & +\sqrt{(1-x)^2+(7-y)^2} \\ & +\sqrt{(8-x)^2+(4-y)^2} \end{aligned}$$

4) Определим градиент целевой функции $\nabla f(x,y)$:

$$\nabla f(x,y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

Для этого найдем частные производные целевой функции от широты (x) и долготы (y) положения склада:

$$\begin{aligned} \frac{\partial f}{\partial x} = & -\frac{4-x}{\sqrt{(4-x)^2+(2-y)^2}} \\ & -\frac{1-x}{\sqrt{(1-x)^2+(7-y)^2}} \\ & -\frac{8-x}{\sqrt{(8-x)^2+(4-y)^2}} \end{aligned}$$

$$\begin{aligned} \frac{\partial f}{\partial y} = & -\frac{2-y}{\sqrt{(4-x)^2+(2-y)^2}} \\ & -\frac{7-y}{\sqrt{(1-x)^2+(7-y)^2}} \\ & -\frac{4-y}{\sqrt{(8-x)^2+(4-y)^2}} \end{aligned}$$

5) Выбираем (в общем случае, произвольно) начальные координаты склада, например, $x_0 = 5$ и $y_0 = 9$, скорость градиентного спуска $\lambda = 2$ и точность расчёта $\varepsilon_1 = 0.25$ (единиц измерения).
6) Текущее суммарное расстояние от склада до всех городов:

$$\begin{aligned} R_0 = f(5,9) = & \sqrt{(4-5)^2+(2-9)^2} \\ & +\sqrt{(1-5)^2+(7-9)^2} \\ & +\sqrt{(8-5)^2+(4-9)^2} = 17.37 \end{aligned}$$

Определим градиент целевой функции в начальной точке положения склада

(x_0, y_0):

$$\left.\frac{\partial f}{\partial x}\right|_{\substack{x\,=\,5\\y\,=\,9}} = -\frac{4-5}{\sqrt{(4-5)^2+(2-9)^2}} -\frac{1-5}{\sqrt{(1-5)^2+(7-9)^2}} -\frac{8-5}{\sqrt{(8-5)^2+(4-9)^2}} = 0.52$$

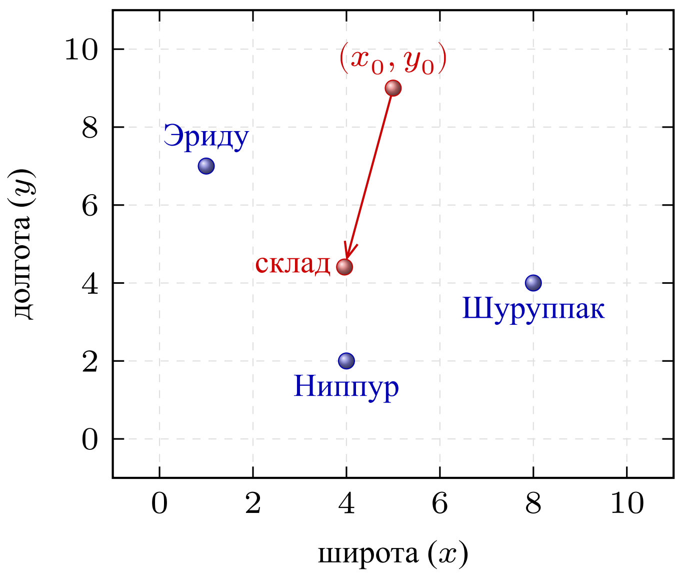
$$\left.\frac{\partial f}{\partial y}\right|_{\substack{x\,=\,5\\y\,=\,9}} = -\frac{2-9}{\sqrt{(4-5)^2+(2-9)^2}} -\frac{7-9}{\sqrt{(1-5)^2+(7-9)^2}} -\frac{4-9}{\sqrt{(8-5)^2+(4-9)^2}} = 2.29$$

Зная градиент целевой функции в начальной точке $\nabla f(x_0, y_0) = (0.52, 2.29)$, определим новые географические координаты склада:

$$x_1 = x_0 - \lambda \cdot \left.\frac{\partial f}{\partial x}\right|_{\substack{x=5\\y=9}} = 5 - 2 \cdot 0.52 = 3.96$$

$$y_1 = y_0 - \lambda \cdot \left.\frac{\partial f}{\partial y}\right|_{\substack{x=5\\y=9}} = 9 - 2 \cdot 2.29 = 4.41$$

Новые географические координаты склада $(x_1 = 3.96; y_1 = 4.41)$.



Рассчитаем величину “шага“ – расстояния между двумя последовательными положениями склада:

$$r = \sqrt{(5-3,96)^2 + (9-4,41)^2} = 4,71$$

Сравниваем величину текущего “шага“ r и заданную точность расчетов ε_1:

$$r = 4.71 > 0.25 = \varepsilon_1$$

Величина текущего “шага“ r больше заданной точности расчетов ε_1, следовательно, *итерационный процесс продолжаем*!

7) Текущее суммарное расстояние от склада до всех городов:

$$\begin{aligned} R_1 = f(3.96, 4.41) = &\sqrt{(4-3.96)^2 + (2-4.41)^2} \\ &+\sqrt{(1-3.96)^2 + (7-4.41)^2} \\ &+\sqrt{(8-3.96)^2 + (4-4.41)^2} = 10.41 \end{aligned}$$

Суммарное расстояние уменьшилось:

$$R_1 = 10.41 < 17.37 = R_0.$$

Рассчитаем градиент целевой функции в новой точке положения склада $x_1 = 3.96$ и $y_1 = 4.41$:

$$\begin{aligned} \left.\frac{\partial f}{\partial x}\right|_{\substack{x_1 = 3.96 \\ y_1 = 4.41}} = &-\frac{4-3.96}{\sqrt{(4-3.96)^2 + (2-4.41)^2}} \\ &-\frac{1-3.96}{\sqrt{(1-3.96)^2 + (7-4.41)^2}} \\ &-\frac{8-3.96}{\sqrt{(8-3.96)^2 + (4-4.41)^2}} = -0.26 \end{aligned}$$

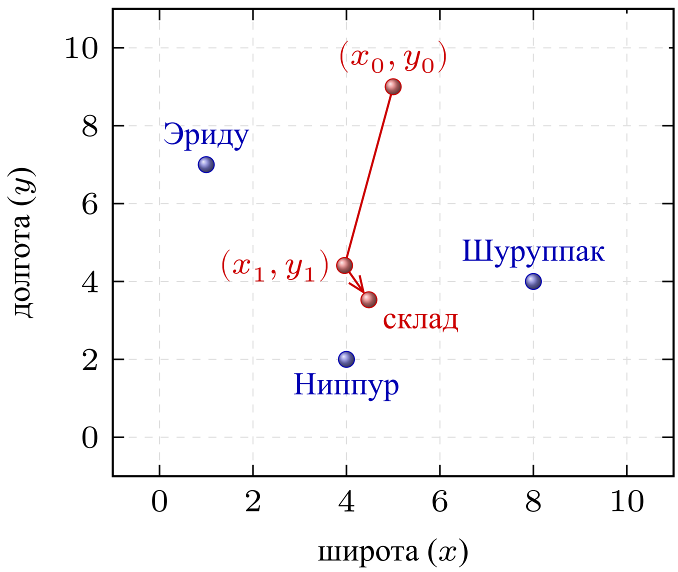
$$\begin{aligned} \left.\frac{\partial f}{\partial y}\right|_{\substack{x_1 = 3.96 \\ y_1 = 4.41}} = &-\frac{2-4.41}{\sqrt{(4-3.96)^2 + (2-4.41)^2}} \\ &-\frac{7-4.41}{\sqrt{(1-3.96)^2 + (7-4.41)^2}} \\ &-\frac{4-4.41}{\sqrt{(8-3.96)^2 + (4-4.41)^2}} = 0.44 \end{aligned}$$

Зная градиент целевой функции в текущей точке $\nabla f(x_1, y_1) = (-0.26, 0.44)$, определим новые географические координаты склада:

$$x_2 = x_1 - \lambda \cdot \left.\frac{\partial f}{\partial x}\right|_{\substack{x=3.96 \\ y=4.41}} = 3.96 - 2 \cdot (-0.26) = 4.48$$

$$y_2 = y_1 - \lambda \cdot \left.\frac{\partial f}{\partial y}\right|_{\substack{x=3.96 \\ y=4.41}} = 4.41 - 2 \cdot 0.44 = 3.53$$

Новые географические координаты склада $(x_2 = 4.48, y_2 = 3.53)$.



Рассчитаем величину “шага“ – расстояния между двумя последовательными положениями склада:

$$r = \sqrt{(3.96 - 4.48)^2 + (4.41 - 3.53)^2} = 1.03$$

Сравниваем величину текущего “шага“ r и заданную точность расчетов ε_1:

$$r = 1.03 > 0.25 = \varepsilon$$

Величина текущего “шага“ r больше заданной точности расчетов ε_1, следовательно, *итерационный процесс продолжаем*!

8) Текущее суммарное расстояние от склада до всех городов:

$$\begin{aligned} R_2 = f(4.48, 3.53) = &\sqrt{(4-4.48)^2+(2-3.53)^2} \\ &+\sqrt{(1-4.48)^2+(7-3.53)^2} \\ &+\sqrt{(8-4.48)^2+(4-3.53)^2} = 10.07 \end{aligned}$$

Рассчитаем градиент целевой функции в новой точке положения склада $x_2 = 4.48$ и $y_2 = 3.53$:

$$\begin{aligned} \left.\frac{\partial f}{\partial x}\right|_{\substack{x_2\,=\,4.48\\ y_2\,=\,3.53}} = &-\frac{4-4.48}{\sqrt{(4-4.48)^2+(2-3.53)^2}} \\ &-\frac{1-4.48}{\sqrt{(1-4.48)^2+(7-3.53)^2}} \\ &-\frac{8-4.48}{\sqrt{(8-4.48)^2+(4-3.53)^2}} = 0.02 \end{aligned}$$

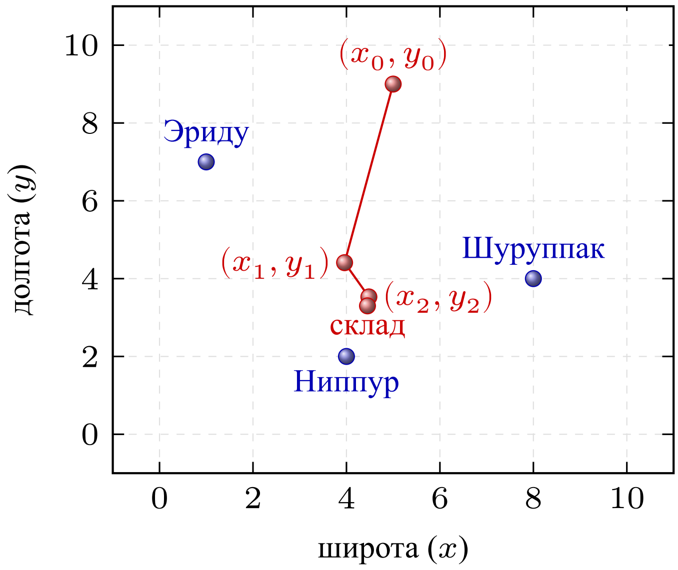
$$\begin{aligned} \left.\frac{\partial f}{\partial y}\right|_{\substack{x_2\,=\,4.48\\ y_2\,=\,3.53}} = &-\frac{2-3.53}{\sqrt{(4-4.48)^2+(2-3.53)^2}} \\ &-\frac{7-3.53}{\sqrt{(1-4.48)^2+(7-3.53)^2}} \\ &-\frac{4-3.53}{\sqrt{(8-4.48)^2+(4-3.53)^2}} = 0.11 \end{aligned}$$

Зная градиент целевой функции в текущей точке $\nabla f(x_2, y_2) = (0.02, 0.11)$, определяют новое географическое положение склада:

$$x_3 = x_2 - \lambda \cdot \left.\frac{\partial f}{\partial x}\right|_{\substack{x=4.48\\ y=3.53}} = 4.48 - 2 \cdot 0.02 = 4.45$$

$$y_3 = y_2 - \lambda \cdot \left.\frac{\partial f}{\partial y}\right|_{\substack{x=4.48\\ y=3.53}} = 3.53 - 2 \cdot 0.11 = 3.30$$

Новые географические координаты склада $(x_0 = 4.45, y_0 = 3.30)$.



Рассчитаем величину "шага" – расстояния между двумя последовательными положениями склада:

$$r = \sqrt{(4.48 - 4.45)^2 + (3.53 - 3.30)^2} = 0.23$$

Сравниваем величину текущего "шага" r и заданную точность расчетов ε:

$$r = 0.23 < 0.25 = \varepsilon$$

Величина текущего "шага" r меньше заданной точности расчетов ε_1, поэтому итерационный процесс поиска положения склада *останавливаем*.

9) Определим минимальное расстояние от склада до всех городов:

$$\begin{aligned} R_4 = f(4.45, 3.30) = & \sqrt{(4 - 4.45)^2 + (2 - 3.30)^2} \\ & + \sqrt{(1 - 4.45)^2 + (7 - 3.30)^2} \\ & + \sqrt{(8 - 4.45)^2 + (4 - 3.30)^2} = 10.05 \end{aligned}$$

Таким образом, с заданной точностью определены оптимальные координаты склада ($x_{opt} = 4.45, y_{opt} = 3.30$), при которых общее расстояние от всех городов до склада будет минимальным и составит $R_{\min} = 10.05$: