

Метод SLSQP

- Метод SLSQP использует последовательное квадратичное программирование для минимизации функций нескольких переменных с любой комбинацией границ, а также ограничений на равенство и неравенство.
- Метод включает в себя подпрограмму оптимизации SLSQP, первоначально реализованную Дитером Крафтом (Dieter Kraft).
- Обратите внимание, что оболочка обрабатывает бесконечные значения в границах, преобразуя их в большие значения с плавающей точкой.

Метод SLSQP работает с задачей минимизации с ограничениями в следующей форме:

$$\begin{aligned}
 & \min_x f(x) \\
 & \text{при } c_j(x) = 0, \quad j \in \mathcal{E} \\
 & \quad c_j(x) \geq 0, \quad j \in \mathcal{I} \\
 & \quad lb_i \leq x_i \leq ub_i, \quad i = 1, \dots, N.
 \end{aligned} \tag{1}$$

Здесь \mathcal{E} и \mathcal{I} – множества индексов, содержащих ограничения по равенству и неравенству.

Метод SLSQP

`scipy.optimize.minimize(method='SLSQP')`

Основные параметры:

`fun: callable`

Объектная функция для минимизации.

`fun(x, *args) -> float`

где `x` – одномерный массив размера `(n,)`; `args` – кортеж параметров, необходимых для вызова функции.

`x0: ndarray, shape (n,)`

Начальное приближение. Массив элементов размера `(n,)`, где `n` – количество независимых переменных.

`args: tuple, optional`

Дополнительные аргументы, передаваемые объектной функции и ее производным (функциям `fun`, `jac` и `hess`)

`bounds: sequence, optional`

Предельные значения для переменных.

`constraints: {Constraint, dict} or List of {Constraint, dict}, optional`

Описание функций, накладывающих ограничения на решение.