Метод наименьших квадратов

Задонский М.А.

20 ноября 2008

1 Введение

В задаче Наименьших квадратов функция $f(x)$ представляется в специальной форме:

$$f(x) = \frac{1}{2}\sum_{j=1}^{m} r_j^2(x) \tag{1}$$

где $r_j^2(x)$ - гладкая функция из $\mathbf{R^n}$ в $\mathbf{R}$. Функции $r_j(x)$ рассматриваются как остаточные. Будем считать, что $m \geq n$

Обозначим вектор остаточных функций в форме:

$$r(x) = (r_1(x), \ldots, r_m(x))^T$$

При таком обозначении функцию f можно записать в виде $f(x) = \frac{1}{2}\|r\|_2^2$. Производные f могут быть записаны в терминах Якобиана функции r. Обозначим

$$J(x) = \left[\frac{\partial r_j}{\partial x_i}\right]$$

Тогда получаем:

$$\nabla f(x) = \sum_{j=1}^{m} r_j(x)\nabla r_j(x) = J(x)^T r(x) \tag{2}$$

$$\nabla^2 f(x) = \sum_{j=1}^{m} \nabla r_j(x)\nabla r_j(x)^T + \sum_{j=1}^{m} r_j(x)\nabla^2 r_j(x) = J(x)^T J(x) + \sum_{j=1}^{m} r_j(x)\nabla^2 r_j(x) \tag{3}$$

2 Алгоритмы для нелинейного МНК

Хотя для минимизации функции (3) можно использовать универсальные методы, как правило, этого не делают, а обращаются к алгоритмам, разработанным специально для этой задичи. Большенство алгоритмов решения задачи о наименьших квадратах опирается на предположение о том, что слагаемое $J^T J$ в формуле (3) рано или поздно станет доминирующим. Это предноложение не соблюдается, если норма $\|f(x_k)\|$ сравнима с максимальным собственным значением матрицы $J(x_k)^T J(x_k)$. Но в большенстве случаев это не так и применение этого предположения оправданно [2].

2.1 Метод Ньютона-Гаусса

Рассмотрим метод минимизации нелинейной функции (1). При этом используется информация о градиенте $\nabla f(x)$ и гессиане $\nabla^2 f(x)$. Метод Ньютона-Гаусса можно рассматривать как модификацию метода Ньютона. Вместо того, что бы определять направление поиска из уравнения $\nabla^2 f(x_k)p = -\nabla f(x)$, рассмотрим уравнение

$$J_k^T J_k p_k^{GM} = -J_k^T r_k$$

Эта простая модификация имеет преимущество перед обычным методом Ньютона. Во-первых, использование приближения

$$\nabla^2 f_k \approx J_k^T J_k$$

избавляет нас от необходимости вычислить отдельно Гессианы $\nabla^2 r(x)$, которые нужны в формуле (3). Якобинан $J(x)$ вычисляется в ходе нахождения градиента $\nabla f(x)$ по формуле (2).

Во-вторых, когда в формуле (3) $J^T J$ намного больше, чем второе слагаемое, метод работает почти также, как и метод Ньютона, даже если второе слагаемое $\sum_{j=1}^m r_j(x)\nabla^2 r_j(x)$ опустить.

Теорема 1 *Пусть все остаточные функции $r_j(x)$ Липшиц-непрерывны с константой N и Якобиан $J(x)$ имеют полный ранг. Тогда, если x_k выбраны методом Гаусса-Ньютона с длинами шага α_k, такими, что*

$$\begin{gathered} f(x_k + \alpha_k p_k) \leq f(x_k) + C_1 \alpha_k \nabla f_k^T p_k \\ \nabla f(x_k + \alpha_k p_k)^T \geq C_2 \nabla f_k^T p_k \\ 0 < C_1 < C_2 < 1 \end{gathered} \tag{4}$$

тогда

$$\lim_{k\to\infty} J_k^T r_k = 0$$

Доказательство см. [1] стр. 261.

2.2 Метод Левенберга-Маркардта

Распространенный альтернативой методу Гаусса-Ньютона является метод Левенберга-Маркардта. В нем направление поиска определяется как решение системы уравнений вида

$$(J^T(x_k)J(x_k) + \lambda_k I)p_k = -J^T(x_k)f(x_k)$$

где $\lambda_k \geq 0$. В этом методе шаг вдоль p_k всегда полагается единичным, то есть $x_{k+1} = x_k + p + k$. Метод Левенберга-Маркардта к классу методов доверительной окрестности. Монотонное убывание минимизируемой функции достигается в нем за счет подбора «хороших» значений λ_k. При $\lambda_k = 0$, p_k будет направлением метод Ньютона-Гаусса.

3 Примеры

Входные данные:

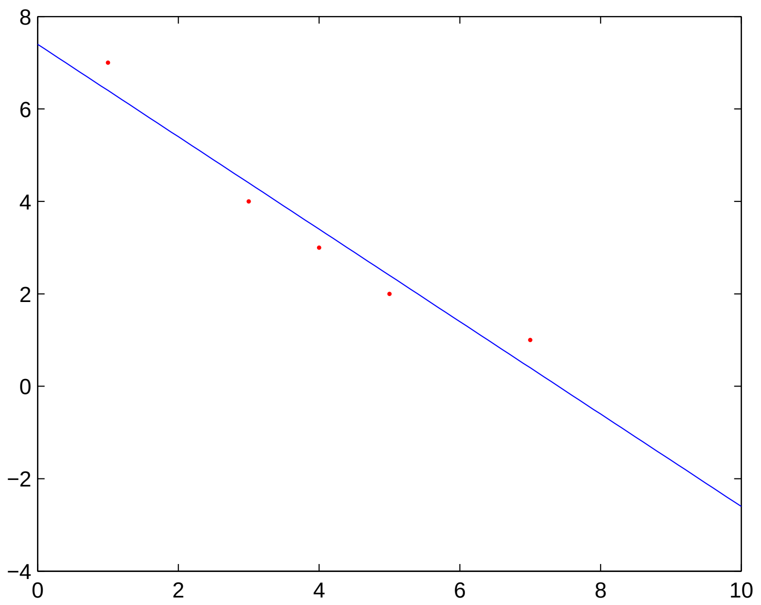
t_j	1	3	4	5	7
y_j	7	4	3	2	1

Минимизируемая функция:

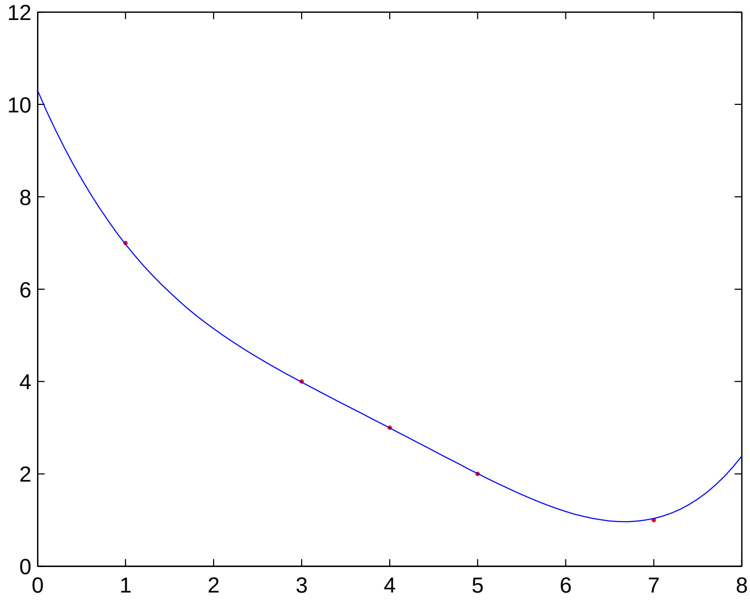
$$f(x) = \frac{1}{2}\sum_{j=1}^{m}[y_j - \varphi(x,t_j)]^2$$

Проведем минимизацию методом Гаусса-Ньютона. Рассмотрим 3 примера функции $\varphi(x,t_j)$

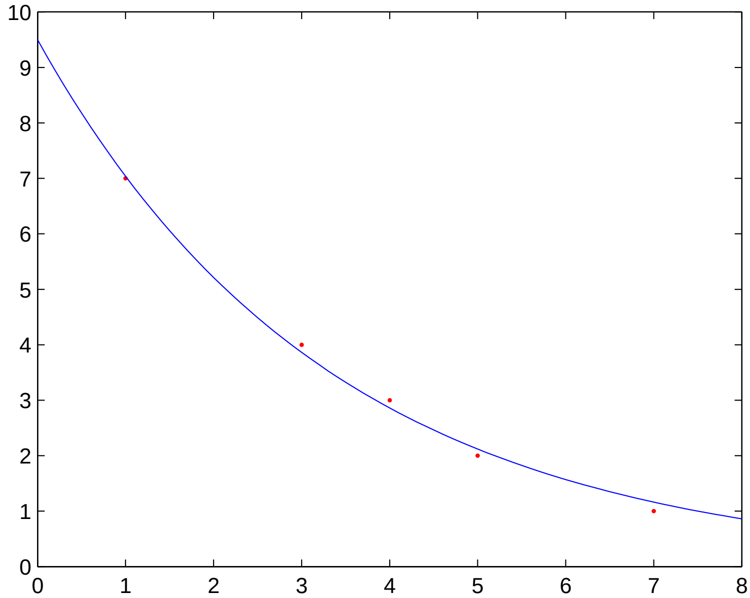
$$\varphi(x,t_j) = P_1[t]$$



$$\varphi(x,t_j) = P_4[t]$$



$$\varphi(x, t_j) = a_1 e^{-a_2 t}$$



4 Рекомендации программисту

В программе используется библиотека MathAdd.dll. Функцию $\varphi(x, t_j)$ можно менять, изменяя якобиан J(CMatrix* getJ (CMatrix *x)) и функцию r_i (double getRi(CMatrix *x,int i)). Программа выдает вектор параметров x.

Список литературы

[1] Numrerical Optimization, Jorge Nocedal, Stephen J. Wright 1999

[2] Practical Optimization, Philip E Gill,Walter Murray 1981