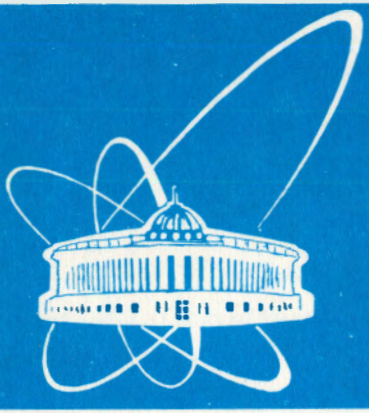


ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

Дубна

98-45

Р11-98-45

В.Б.Злоказов

МЕТОД НАИМЕНЬШИХ КВАДРАТОВ ДЛЯ РЕГРЕССИЙ СО СЛУЧАЙНЫМИ ИЛИ НЕОПРЕДЕЛЕННЫМИ АРГУМЕНТАМИ

Направлено в журнал «Computer Physics Communications»

1998

Данная работа представляет собой обобщение метода и алгоритмов, реализованных в [1], на случай подгонки распределений в менее типичных задачах, чем классические, но в то же время представляющих большой практический интерес. Математический формализм в [1] был следующий: дана регрессия $y(x_j), j = 1, ..., m$, и дана модель $f(x_j, P)$, зависящая от параметров P. Их оценки находились минимизацией функционала

$$\sum_{j=1}^{m} w_j (y_j - f(x_j, P))^2, \tag{1}$$

при условии

$$h(x, y, P) \leq 0, \tag{2}$$

где w_j - веса (в частности, обобщенные).
К новым задачам относятся:

- Аргументы регрессий x_j - случайные величины.
- Аргументы регрессий x_j - неопределенные величины, зависящие от каких-либо дополнительных условий как формульного, так и неформульного типа, например, являющиеся целочисленными величинами из заданных множеств.

Все эти задачи могут быть решены с помощью минимизации МНК-функционала типа (1)-(2), но программа GFIT, опубликованная в [1], требует некоторых дополнений. Главное дополнение было обусловлено необходимостью сочетать метод наименьших квадратов и метод статистических испытаний: последний является генератором начальных значений параметров, уточняемых далее с помощью МНК-процедуры, которая, в случае сходимости, дает одну из возможных оптимальных оценок, а из них выбирается самая лучшая, которая и будет окончательной МНК-оценкой. Если начальные значения берутся из очень большой области, то, как правило, МНК-процесс не будет сходящимся. Для экономии времени несходящиеся итерации нужно прекращать как можно раньше или менять стратегию (шаг, демпфер, веса) минимизации. Именно такие добавления и были внесены в программу GFIT.
Ниже приведены примеры 4 типичных задач, иллюстрирующих работу программы GFIT в новых условиях.

1. Случайный аргумент. Пусть задана регрессия $f(x, P)$, где $x \in X$, n-мерный вектор P является множеством его параметров. Наблюдение этой регрессии дает множество значений y_j, $j = 1, 2, ..., m$, которое мы относим к значениям аргумента x_j. Последние могут быть неизвестными точно, но полученными экспериментально и представлять точные значения лишь с определенной статистической погрешностью. Требуется на основании регрессионной модели $f(x, P)$ и множества экспериментальных данных y_j, x_j, которые предполагаются взаимно независимыми, получить оценки параметров P.
В конфлюентном анализе исследовалась задача для простейшего случая линейной регрессии с нормально распределенными x, y; данная работа предлагает универсальный и практически эффективный метод для произвольной нелинейной регрессии.
Будем считать, что математические ожидания x_j и y_j суть $\bar{x}_j$ и $f(x, P_0)$, а дисперсии - v_{xj} и v_{yj} соответственно. Здесь P_0 является истинным значением параметров.
Нам нужно построить одновременно оценки как параметров P, так и аргументов x_j.

Возьмем $n+m$-мерный вектор Q

$$q_i = p_1, p_2, ..., p_n, x_1, ..., x_m,$$

и $m+m$-мерный вектор T; его компоненты t_j суть

$$t_j = y_1, y_2, ..., y_m, x_1, ..., x_m.$$

Его моделью является $m+m$-мерный вектор S

$$s_j = f_1, f_2, ..., f_m, \bar{x}_1, ..., \bar{x}_m,$$

где каждая $f_j = f(x_j, P)$, а дисперсия

$$W = v_{y1}, v_{y2}, ..., v_{ym}, v_{x1}, ..., v_{xm}.$$

Далее мы можем действовать как обычно и строить МНК-оценки вектора Q минимизацией выражения

$$\sum_{j=1}^{2m} \frac{1}{w_j}(t_j - s_j(Q))^2 \qquad (3)$$

по параметрам q_k, $k = 1, 2, ..., n+m$. (3) является обычным МНК-функционалом, и дальнейшие действия являются стандартными: дифференцируя (3) по q_k, мы получаем нормальную систему уравнений для МНК-оценок. Частными производными $s_j(q_i)$ по q_k являются

$$\frac{\partial s_j}{\partial q_k} = \begin{cases} \frac{\partial f_j}{\partial p_k} & \text{if } k = 1, 2, ..., n; j = 1, 2, ..., m; \\ \frac{\partial f_j}{\partial x_k} & \text{if } k = n+1, ..., m+n; j = 1, 2, ..., m; \\ 1 & \text{if } k = n+1, ..., m+n; j = m+1, ..., 2m; \\ 0 & \text{if } k = 1, ..., n; j = m+1, ..., 2m. \end{cases}$$

Аналогичным образом метод может быть распространен на многомерный случай. Пусть функция $f(X, P)$ задана, где X является k-мерным вектором. Обобщение вектора Q есть $n + k \cdot m$-мерный вектор

$$q_i = p_1, p_2, ..., p_n, x_{11}, ..., x_{m1}, x_{12}, ..., x_{mk},$$

или, иначе записывая (группируя координаты точек),

$$q_i = p_1, p_2, ..., p_n, x_{11}, x_{12}, ..., x_{1k}, x_{21}, ..., x_{mk}.$$

Уравнение (3) выглядит следующим образом:

$$\sum_{j=1}^{(k+1)m} \frac{1}{w_j}(t_j - s_j(Q))^2. \qquad (4)$$

Затем мы получаем МНК-оценки вектора Q минимизацией (4) по $n+km$ параметрам: n параметров p_i и km параметров - X-координаты.
Возникает вопрос, каково качество получаемых оценок. Поскольку величины X являются уже не независимыми аргументами, но параметрами, регрессия $f(X, P)$ становится нелинейной, даже если она линейна по P. Поэтому мы уже не можем

гарантировать несмещенность оценок параметров P, минимальность их дисперсий и их состоятельность. Тем не менее на практике существует большое количество случаев, когда метод работает хорошо. В [2] даны условия для вышеупомянутого хорошего качества МНК-оценок параметров нелинейных регрессий, по крайней мере, в асимптотике. Их общий смысл: малость дисперсий данных и близость начальных параметров к истинным значениям.
Следует учитывать, что в общем случае регрессия со случайным аргументом содержит меньше информации, чем регрессия с известными аргументами. Иногда произвольная фиксация аргумента может привести к меньшим систематическим ошибкам, чем общая очень большая дисперсия регрессии со случайным аргументом.
Поэтому нижеследующий метод является альтернативой к вышеописанному подходу.
Зафиксируем x_j, но повысим дисперсии y_j по формуле

$$\tilde{v}_{yj} = \frac{\partial f(x_j, P)}{\partial x}^2 v_{xj} + v_{yj}.$$

Тогда мы имеем обычную регрессию

$$y(x_j) = f(x_j, P) + e(x_j),$$

где $e(x)$ является ошибкой с дисперсией $\tilde{v}(x)$, и далее используется обычный МНК-метод: минимизировать

$$\sum_{j=1}^{m} \frac{1}{\tilde{v}_j}(y_j - f_j(P))^2 \qquad (5)$$

по параметрам p_k, $k = 1, 2, ..., n$.

2. Неопределенный аргумент. Пусть дана регрессия $y = f(x, P)$, где n-мерный вектор P - вектор параметров, но для экспериментально измеренных ординат $y_j, j = 1, 2, ..., m$, их значения x являются неизвестными ни точно, ни приближенно, а известно лишь то, что $x \in X$, где X - дискретное множество значений $x_k, k = k_1, k_2, ..., k_l$.
Рассмотрим следующий реальный пример.
Пусть из калибровочного измерения известно, что изотоп имеет следующий список энергий гамма-переходов: $E_i, i = 1, 2, ..., i_m$. Другое измерение дает список каналов $x_j, j = j_1, j_2, ..., j_m$, в которых были зарегистрированы пики интенсивности, и известно, что эти каналы соответствуют части калибровочного списка энергий. Требуется:

1. определить автоматически соответствующие друг другу пары "энергия - канал измерения";
2. осуществить калибровку энергия - канал , т.е. оценить параметры P калибровочной формулы $E(x) = f(x, P)$.

Задачу можно решить с помощью следующего итерационного процесса. Пусть на k-й итерации мы имеем оценку параметров P_k; мы формируем вектор аргументов из условия

$$x_i = ARG(MIN\|E_i - f(x, P_k)\|), i = 1, 2, ..., i_m,$$

где x_i - аргумент минимума по x - ищется среди $(x_j), j = j_1, ..., j_m$. Далее, как обычно, с помощью МНК получается уточненное значение параметров P_{k+1}, т.е. минимизацией

$$\sum_{i=1}^{m} w_i (E_i - f(x_i, P))^2. \quad (6)$$

Успех этой процедуры зависит от близости оценки параметров к истинным значениям. Обычно область возможных значений параметров может быть задана неравенствами

$$p_{il} \le p_i \le p_{iu} \quad (7)$$

и начальный вектор параметров берется из этой области.
Однако эта область может оказаться слишком большой и произвольный исходный вектор может не привести к сходимости итераций. В этом случае необходима серия испытаний - последовательность случайно выбранных исходных векторов из (7) и их пробное итерационное уточнение, что является единственным средством для успешного решения задачи.

3. Распознавание точечных образов кривых на плоскости. Пусть задано множество точек на плоскости $(T_j), j = 1, ..., m$, где каждая точка имеет координаты x_j, y_j и либо принадлежит (в пределах определенной погрешности) одной из кривых

$$f_i(x, y, P_i) = 0,$$

$i = 1, ..., n$, зависящих от неизвестных параметров P_i, либо является случайной. Задача заключается:

- в определении того, какой из одной или нескольких кривых из заданного множества принадлежит (или не принадлежит ни одной) каждая точка T_j;
- в оценке параметров кривых P_i.

Для того, чтобы решить эту задачу с помощью программы GFIT, надо сформулировать ее как задачу минимизации квадратичного функционала интегрального типа в пространстве значений параметров.
Выведем предварительно меру евклидовой близости произвольной точки x_1, y_1 на плоскости к кривой, заданной уравнением $f(x, y) = 0$.
Очевидно, что такая точка будет удовлетворять следующему требованию:

$$min\ ((x - x_1)^2 + (y - y_1)^2)$$

$$\text{при условии } f(x, y) = 0$$

Применение к ней метода Лагранжа приводит к очень сложной системе уравнений относительно координат и множителя, поэтому мы поступим иначе. Предположим, что точка x_1, y_1 находится в малой окрестности кривой, так что допустима линеаризация условия

$$f(x, y) \sim f(x_1, y_1) + \frac{\partial f(x_1, y_1)}{\partial x} h_x + \frac{\partial f(x_1, y_1)}{\partial y} h_y = 0,$$

где $h_x = x - x_1; h_y = y - y_1$. Тогда применение метода Лагранжа приводит к следующей системе уравнений:

$$\frac{\partial f(x_1, y_1)}{\partial y} h_x - \frac{\partial f(x_1, y_1)}{\partial x} h_y = 0,$$

$$\frac{\partial f(x_1, y_1)}{\partial x} h_x - \frac{\partial f(x_1, y_1)}{\partial y} h_y = f(x_1, y_1).$$

Ее решение позволяет получить точку x, y (которую, если надо, можно уточнить итерационным путем):

$$x = x_1 - \frac{f_1 f'_x}{D}; y = y_1 - \frac{f_1 f'_y}{D},$$

где $f_1 = f(x_1, y_1)$, а $D = f'^2_x + f'^2_y$ в точке x_1, y_1: и норму расстояния от точки до кривой

$$r^2(x_1, y_1, P_1) = r_1^2 = \frac{f_1^2}{D}. \quad (8)$$

Если бы мы знали, к каким кривым принадлежат точки T_j, то функционал качества для уточнения параметров мы могли бы взять в таком виде:

$$F(P_1, P_2, ...) = \sum_{i \in I_1} r_1^2 + \sum_{i \in I_2} r_2^2 + ... + \sum_{i \in I_n} r_n^2, \quad (9)$$

где I_i - множество индексов точек, лежащих на i-й кривой, а r_i - выражение типа (8). Если какие-либо точки не принадлежат ни одной кривой f_i, они в сумме (9) не участвуют.
Функционал (9) не является квадратичным по параметрам, но он может быть сделан таким, если подставить некоторую априорную оценку параметров в знаменатели D выражений (8) и тем самым зафиксировать их. Выражение (9) становится тогда каноническим МНК-функционалом, в котором значения регрессии равны нулю, $f_i(T_j, P_i)$ играют роль моделей регрессии, а величины $\frac{1}{D_j}$ являются весами.
Окончательный алгоритм решения задачи будет выглядеть так.

1. Выбирается некоторое начальное значение параметров P_i, например, случайным выбором из области возможных значений этих параметров (чем уже эта область, тем лучше, но годится, конечно, любая).

2. С помощью какого-либо алгоритма сортировки точки приписываются к кривым f_i либо объявляются неидентифицированными.

3. Делается стандартный шаг МНК-процедуры, с помощью которого уточняются параметры P_i, либо, если процесс не сходится, МНК-процедура с данными параметрами обрывается. Обрыв делается также, если сортировка оказывается плохой (слишком мало идентифицированных точек).

4. Если процесс сходится, то пересчитываются веса $\frac{1}{D_j}$, делаются новая сортировка и новый шаг МНК-процедуры, пока не закончится уточнение.

5. Для каждой кривой проверяется, обеспечивают ли уточненные параметры лучшую близость точек к этой кривой в смысле (8), и если да, то эти параметры запоминаются.

6. Затем повторяются шаги 1-5 до тех пор, пока не будет получено удовлетворительное решение задачи распознавания.

Простейшим алгоритмом сортировки является следующий.

- Вычисляются расстояния (8) от j-й точки до i-й кривой $r(j,i)$, и определяются медианы этих расстояний для каждой кривой $m_i = MEDIAN_j(r(j,i))$.
- j-я точка относится к i-й кривой, если: 1/ (8) для нее является минимальным; 2/ оно меньше m_i. Иначе точка считается неидентифицированной.

Данный алгоритм является, как и в предыдущем случае, комбинацией метода наименьших квадратов и метода статистических испытаний, но имеет особенность, вызванную тем, что здесь все параметры могут быть разбиты на независимые группы, и заломинать, в случае успеха, имеет смысл не весь вектор параметров, а только те группы, по которым получено улучшение. Это существенно экономит испытания.

О математической корректности задачи можно сказать следующее. Помимо обычных условий: невырожденность МНК-матрицы и регулярность ошибок,здесь есть два дополнительных условия:

- к каждой кривой в ходе сортировки должно быть приписано количество точек A_i, не меньшее, чем заданные априори величины a_i:

$$A_i \geq a_i;$$

- кривые f_i должны быть значимо различимы.

Иначе задача минимизации не может иметь корректного решения.

Описанный алгоритм является универсальным и в то же самое время очень мощным средством распознавания точечных образов самых разнообразных кривых и отличной альтернативой или дополнением к таким методам, как методы перцептрона, нейронных сетей и т.д.
Описанный алгоритм допускает также простое и естественное обобщение на случай n-мерных кривых, причем получается практически весьма эффективный метод.

4. Энтропийные метрики. Методы получения оценок параметров с помощью принципа минимальной (или максимальной, в зависимости от знака) энтропии математически разработаны вообще-то для функций и плотностей распределения случайных величин, но их популярность (независимо от того, обоснована она или нет) делает актуальным вопрос о способах перенесения этого принципа на случай регрессий. Более точной и общей постановкой задачи будет, по-видимому, такая: построить метрику (или иную меру близости), которая в приемлемой степени уступала бы МНК-метрике в смысле эффективности, но превосходила бы ее в отношении других качеств (например, робастности).
Пусть $a(x)$ и $b(x)$ - элементы метрического пространства, $\epsilon(x)$ - "ошибка" одного из элементов a, b с нулевым средним и дисперсией $V(x)$, а $x = x_i, i = 1, 2, ..., m$. Рассмотрим выражение

$$\rho(a,b) = \sum_{i=1}^{m} \frac{(a(x_i) - b(x_i))^2}{V(x_i)} exp(-\frac{(a(x_i) - b(x_i))^2}{V(x_i)}). \qquad (10)$$

Выражение (10) можно рассматривать как метрический аналог негэнтропии [3]. Будучи ограниченной, эта метрика, естественно, более робастна, чем МНК-метрика. В нее можно включить весовую функцию $w(x_i)$, а в последнюю, после соответствующей нормировки, и множитель

$$exp(-\frac{(a(x_i) - b(x_i))^2}{V(x_i)}),$$

и, следовательно, минимизацию (или максимизацию) в этой метрике можно осуществлять с помощью программы GFIT.
Представляют интерес и другие меры близости, являющиеся функциями от $f(a(x_i) - b(x_i))ln(f(a(x_i) - b(x_i)))$, например:

$$\rho_1(a,b) = \sum_{i=1}^{m} \frac{\|a(x_i) - b(x_i)\|}{\sqrt{V(x_i)}} ln((\frac{\|a(x_i) - b(x_i)\|}{\sqrt{V(x_i)}})).$$

Величина ρ_1 похожа на метрику, но отличается от нее, например, неединственностью нуля: $\rho_1(a, a + e) = 0$, хотя $a + e$ не равно a.
Другими словами, она равна нулю в двух случаях:

1. если a равно b;
2. если a отличается от b на стандартное отклонение.

Мы назовем ρ_1 квазиметрикой. Она не является ограниченной, но она мала в области $a \pm 2e$ и растет существенно медленнее, чем МНК-метрика, и, следовательно, будет более робастной, чем последняя.
Эту и любую интегральную меру близости $M(a(x) - b(x))$ можно представить приближенно так:

$$\sum_x W(\hat{a}(x), \hat{b}(x))(a(x) - b(x))^2, \qquad (11)$$

где $W(u,v)$ - весовая функция, учитывающая как норму погрешности $a(x), b(x)$ (вместо них можно поставить их оценки), так и нормированные компоненты меры M, "мешающие" ей быть квадратичной. Это делает процедуру частным случаем обобщенной подгонки. Она может быть осуществлена программой GFIT с использованием гибких весов.

Примеры

1. Исследуется зависимость температуры сверхпроводящего перехода от концентрации допирующего элемента. Формула регрессии

$$y(x) = a(1 - b(x - c)^2),$$

где a, b, c - параметры, а x - аргумент, являющийся случайной величиной.

2. Автоматическое построение калибровочной кривой "энергия - канал". Дана таблица известных калибровочных линий ряда изотопов и результаты обработки гамма-спектра, содержащего среди прочих и линии калибровочных изотопов. Требуется определить, какие линии принадлежат имеющимся в таблице изотопам и одновременно оценить параметры калибровочной кривой.

3. Даны 50 точек, из них 20 лежат (неточно) на части эллипса $\frac{y-y_0}{w_y} = \sqrt{1-\frac{x-x_0}{w_x}^2}$, 20 других (тоже неточно) на фрагменте кривой $y = asin(wx) + c$, 10 точек - случайные. Программа в основном правильно рассортировала точки и весьма точно оценила параметры $x_0, w_x, y_0, w_y, a, w, c$. Несколько точек, одинаково близких к обеим кривым, были отнесены к одной из них произвольно.

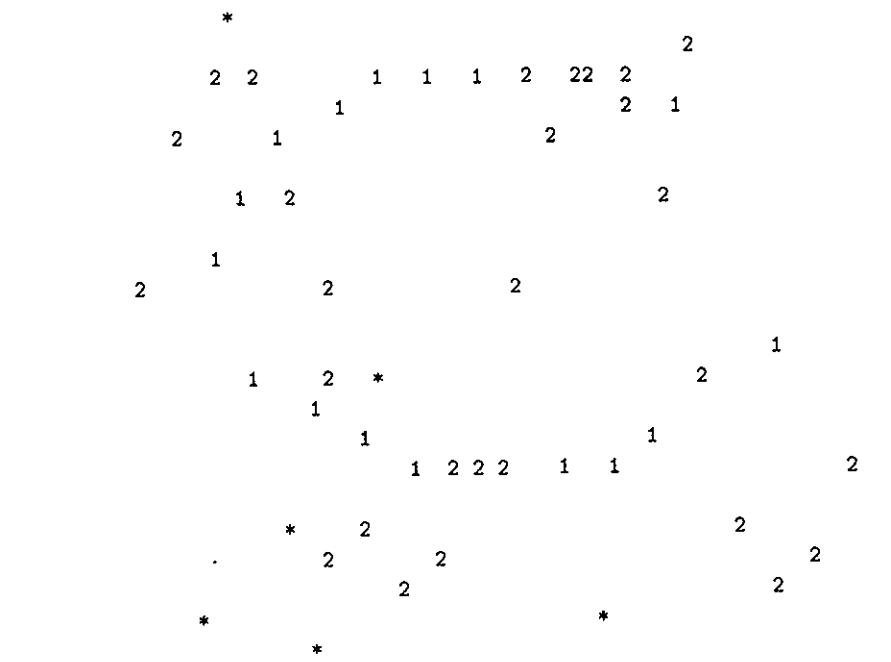
4. Параметры линейной регрессии с выбросами оценивались с помощью подгонки в метриках: энтропийной, наименьших квадратов и МНК с обобщенными весами. Точные ответы: 1,10. Видно, что лучшие ответы получены в третьем случае.

Ниже приведен краткий листинг выдачи программы GFIT.


```
**********      RANDOM  ARGUMENT      ****************
Ch2/df=    .31  eps=   .3    3 + 0 iter.  RF=1.00  Chdev= .71  Ch+= .45
 Parameters and their errors
    97.4282      .1280      .1407
     1.2543      .0059      .0131
 Parameter b =      50.500      9.376
 Experimental and fitted arguments and their errors
   .0570  .0600  .1200  .1240  .1400  .1900  .1940
   .0560  .0614  .1186  .1244  .1394  .1831  .2016
   .0045  .0044  .0077  .0089  .0092  .0100  .0111

**********      UNORDERED ARGUMENT    ****************
Ch2/df=   1.00  eps=   .2   10 + 0 iter.  RF= .90  Chdev= .50  Ch+= .28
 Found channels
  279.103   548.540   767.232   985.708  1719.957
 1912.933  2125.412  2394.927  2449.159  3097.047
 Parameters and their errors
       .456    -5.735
       .000      .330
     Data     Fit    Chi2                Graph
 ------------------------------------------------------------------------
    121.78  121.61     .11 .   *
    244.69  244.55     .08 .       *
    344.27  344.34     .02 .           *
    443.94  444.03     .03 .             *
    778.91  779.05     .07 .                       *
    867.38  867.10     .32 .                         *
    964.07  964.05     .00 .                             *
   1085.83 1087.03    5.67 .                               *
   1112.06 1111.77     .35 .                                 *
   1407.97 1407.39    1.36 .                                          *
```

```
**********      POINT  PATTERN  RECOGNITION   ********
   3000 tries,     chi1, chi2 =       .008       .002
```



```
      1 - points of ellipse,  2 - of sine, * - random
```

```
Obtained parameters
    4.94   4.01   6.02   2.97       4.01   1.25   5.09
True parameters
    5.00   4.00   6.00   3.00       4.00   1.26   5.00
```



```
**********      MAXIMUM  ENTROPY  FITTING    *********
   ..............The fitting quality is not good
   ..............Minimization accuracy not reached
Ch2/df=   5.76  eps=  1.6    7 + 5 iter.  RF= .60  Chdev= .33  Ch+= .87
 Parameters and their errors
      1.141     8.262
       .044      .577
    Data     Fit    Chi2                Graph
 ------------------------------------------------------------------------
     12.49    9.40   12.73 .   * +
     10.99   10.54     .26 .    *
     13.25   11.68    3.26 .    *+
     13.71   12.82    1.06 .     *
       .00   13.96  260.01 .+    *
     16.19   15.11    1.57 .     *+
     17.62   16.25    2.51 .      *
     16.82   17.39     .43 .      *
     20.23   18.53    3.88 .       *
    100.00   19.67 8604.45 .       *                              +
     19.89   20.81    1.13 .       **
     21.25   21.95     .66 .        *
     23.39   23.09     .12 .        *+
     23.75   24.23     .31 .         *
       .00   25.37  858.19 .+        *
     26.58   26.51     .01 .          *
     27.88   27.65     .07 .          *
     26.96   28.79    4.47 .          *
     30.27   29.93     .15 .           *
     31.23   31.07     .03 .           *
 ------------------------------------------------------------------------
   ..............The fitting quality is not good
Ch2/df= 531.05  eps=   .0    7 + 5 iter.  RF= .25  Chdev= .33  Ch+= .85
 Parameters and their errors
       .917    13.000
       .034      .402
   ..............The fitting quality is not good
Ch2/df=   3.23  eps=   .0    7 + 5 iter.  RF= .70  Chdev= .33  Ch+= .39
 Parameters and their errors
      1.026     9.812
       .035      .425
```

Литература

1. Zlokazov V.B.
GFIT - generalized quadratic approximation of functions
under constraints.
Computer Physics Communications,1989,v.54,p.371-379.

2. Злоказов В.Б.
О детерминистской интерпретации метода наименьших квадратов.
ОИЯИ, Р10-86-502, Дубна, 1986.

3. Белашов Б.З.,Сороко Л.М. Препринт ОИЯИ 10-86-100, Дубна, 1986.

Рукопись поступила в издательский отдел
12 марта 1998 года.