

Глава 11. СГЛАЖИВАНИЕ РЕЗУЛЬТАТОВ ИЗМЕРЕНИЙ МЕТОДОМ НАИМЕНЬШИХ КВАДРАТОВ

11.1. Одна содержательная задача

Описание проблемы, которой посвящена эта глава, мы начнем с простейшего примера – измерения сопротивления резистора методом амперметра и вольтметра. Метод этот, как известно, состоит в том, что одновременно измеряются: ток J , текущий через резистор, и U – падение напряжения на нем (рис.11.1).

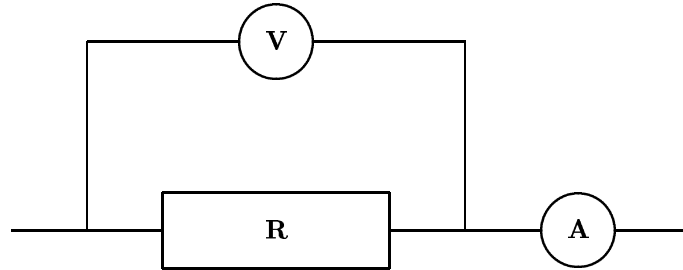


Рис.11.1

Если пренебречь током, текущим через вольтметр, то, в соответствии с законом Ома, искомое сопротивление находится по формуле

$$R = \frac{U}{J}. \quad (11.1)$$

Дело, однако, осложняется тем, что, повторяя измерения, получают каждый раз новое значение сопротивления. Возможны два различных толкования этого экспериментального факта.

1. Сопротивление резистора не постоянно, а изменяется со временем.
2. Сопротивление постоянно, но данные измерений содержат ошибки (погрешности приборов, субъективные ошибки наблюдателя и пр.)

Следует ясно понимать, что без выбора математической модели эксперимента его результаты обрабатывать нельзя.

В нашей задаче мы постулируем неизменность сопротивления резистора во времени (выяснение правомерности такого постулата выходит, естественно, за рамки курса математики) и взваливаем ответственность за его наблюдающиеся изменения на погрешности. Тогда, проведя серию из n измерений, мы можем записать их результаты в виде

$$\begin{cases} J_1 R = U_1 \\ \dots\dots\dots \\ J_n R = U_n \end{cases},$$

т.е. в виде системы из n линейных уравнений с одной переменной – искомым сопротивлением резистора. Поскольку эта система, очевидно, несовместна, будем искать ее псевдорешение. Несмотря на простоту задачи, проделаем все операции подробно, чтобы еще раз продемонстрировать методику.

Запишем систему в виде

$$JR = U, \quad (11.1.2)$$

где

$$J = [J_1, \dots, J_n]^T, \quad U = [U_1, \dots, U_n]^T.$$

Матрица $J^*J = [J_1^2 + \dots + J_n^2]$ имеет размер 1×1 , а матрица

$$JJ^* = \begin{bmatrix} J_1^2 & J_1 J_2 & \dots & J_1 J_n \\ \dots\dots\dots \\ J_n J_1 & J_n J_2 & \dots & J_n^2 \end{bmatrix} \quad -$$

размер $n \times n$.

Так как J^*J имеет размер 1×1 , сингулярное число оказывается единственным - это положительный корень уравнения

$$\det(J^*J - \sigma^2 I_1) = 0 \quad \text{или} \quad \sum_{k=1}^n J_k^2 - \sigma^2 = 0, \quad \text{т.е.} \quad \sigma = \left(\sum_{k=1}^n J_k^2 \right)^{1/2}.$$

Соответствующие сингулярные базисы находятся так: *правый* (состоящий из одного вектора) - путем решения однородной "системы"

$$(J^*J - \sigma^2 I_1)v^{(1)} = \theta_1 \quad \text{или} \quad \theta_1 \cdot v^{(1)} = \theta_1 \quad \text{т.е.} \quad v^{(1)} = [1]$$

(напомним, что сингулярный вектор нормирован). Соответствующий *левый* сингулярный вектор $w^{(1)}$ - это нормированный образ вектора $v^{(1)}$ при отображении, порожденном матрицей J : $w^{(1)} = \frac{1}{\sigma} J v^{(1)} = \frac{1}{\sigma} J$. Этот вектор можно было бы дополнить до ортонормированного базиса в \mathbb{R}^n , но остальные векторы нам в дальнейшем не понадобятся!

Сингулярное разложение матрицы J имеет вид $J = W \Sigma V^*$, где $V^* = V = [1]$, $W = [w^{(1)}, \dots, w^{(n)}]$ - унитарная матрица порядка n , а $\Sigma = [\sigma, 0, \dots, 0]^T$ - столбец высоты n .

Матрица $\Sigma^+ = [1/\sigma, 0, \dots, 0]$ - строка ширины n . Поэтому псевдообратная матрица -

$$J^+ = V \Sigma^+ W^* = \frac{1}{\sigma} w^{(1)*} = \frac{1}{\sigma^2} J^*.$$

Теперь можно по формуле (10.3.1) найти псевдорешение:

$$\begin{aligned} \tilde{R} = J^+ U &= \frac{1}{\sigma^2} J^* U = \frac{1}{\sigma^2} (J_1 U_1 + \dots + J_n U_n) = \\ &= \frac{J_1 U_1 + \dots + J_n U_n}{J_1^2 + \dots + J_n^2}. \end{aligned}$$

Замечание. Внимательный читатель заметит, что тот же результат можно было получить, перейдя к нормальным уравнениям, т.е. домножив (11.1.2) слева на J^* :

$$(J^* J) \tilde{R} = J^* U \implies \tilde{R} = \frac{J^* U}{J^* J} = \frac{J_1 U_1 + \dots + J_n U_n}{J_1^2 + \dots + J_n^2}.$$

Но, во-первых, мы уже объяснили, почему не следует пользоваться нормальными уравнениями (хотя в нашей задаче это было допустимо), а во-вторых, нашей целью была демонстрация построения псевдорешения системы линейных уравнений с помощью сингулярного разложения матрицы ее коэффициентов

Полученное число можно истолковать как значение сопротивления резистора, "наилучшим образом согласующееся" сразу со всеми результатами измерения (при этом степень согласованности понимается так, как было сказано в замечании п.10.2). Можно полагать, что таким образом мы избавились от "случайных" погрешностей в результатах измерений - *сгладили* эти результаты.

Серьезное предупреждение. Подчеркнем еще раз, что приведенные выше рассуждения имеют смысл *только в рамках принятой математической модели* - сопротивление резистора предполагается постоянным во времени. Если же наша модель не верна, и сопротивление на самом деле изменялось во время измерений, то такое "сглаживание" превращается в искажение реально наблюдаемого явления.

11.2 Полиномиальное сглаживание

Рассмотрим (с меньшей конкретизацией ее содержательной части) еще одну распространенную прикладную задачу.

Информационно-измерительная система (ИИС) фиксирует в равноотстоящие моменты времени t_1, \dots, t_n значения некоторой измеряемой величины y_1, \dots, y_n .

Требуется (как обычно говорят прикладники) "подобрать какую-нибудь *простую и удобную* функцию, которая *хорошо описывала* бы полученные результаты".

Попытаемся придать точный смысл этой туманной фразе.

По-видимому слова "хорошо описывает" можно понимать только в одном смысле: поскольку ИИС никаких сведений об измеряемой величине, кроме пар чисел (t_i, y_i) , $i = 1, \dots, n$, не имеет, подобранная нами (мы будем называть ее *аппроксимирующей*) функция в точках t_1, \dots, t_n должна принимать значения y_1, \dots, y_n соответственно.

²Экономия памяти бесплатно не достигается: теперь вместо выборки числа y_k приходится вычислять значение полинома $S(k)$ – расходуя машинное время!

Качество аппроксимации оценивается *среднеквадратической погрешностью*

$$\left(\frac{1}{n} \sum_{k=1}^n (S(k) - y_k)^2 \right)^{1/2}$$

(по построению сглаживающий полином минимизирует именно эту погрешность на множестве всех полиномов порядка m).

Серьезное предупреждение. Часто делаются попытки использовать сглаживающий полином для работы с ним *между* узлами таблицы. Не имея возможности запретить такого рода операции, хотим заранее снять с математики ответственность за возможное "качество" их результатов.

11.3. Сглаживание полиномами, ортогональными на сетке

Мы показали, что построение сглаживающего полинома на равномерной сетке сводится к нахождению нормального псевдорешения системы линейных уравнений

$$\tilde{T}s = y. \quad (11.3.1)$$

В п.10.3 указано, что нормальное псевдорешение этой системы является решением системы нормальных уравнений. Но нормальные уравнения выгодны, если столбцы матрицы \tilde{T} попарно ортогональны.

Известно, что матрица Вандермонда

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & 2^{n-1} \\ \dots & \dots & \dots & \dots \\ 1 & n & \dots & n^{n-1} \end{bmatrix}$$

не вырождена. Следовательно, множество ее столбцов линейно независимо. Поэтому линейно независима и его часть – множество столбцов матрицы \tilde{T} . А любое линейно независимое множество векторов можно с помощью алгоритма Грама-Шмидта преобразовать в ортогональное.

Положим, согласно п.7.7, $f^{(1)} = \tilde{T}^{(1)} = [1, \dots, 1]^T$. Далее

$$f^{(2)} = \tilde{T}^{(2)} - \alpha_{12}f^{(1)}, \quad \text{где} \quad \alpha_{12} = \frac{\langle \tilde{T}^{(2)}, f^{(1)} \rangle}{\langle f^{(1)}, f^{(1)} \rangle} = \frac{n+1}{2} \quad \text{и т.д..}$$

На языке полиномов этот процесс выглядит так.

Вектор $\tilde{T}^{(j)}$ – это значения полинома $E^{(j)}(t) = t^{j-1}$ на стандартной сетке $\{1, \dots, n\}$. Поэтому $f^{(j)}$ – значения на той же сетке полинома

$$F^{(j)}(t) = E^{(j)}(t) - \alpha_{1j}F^{(1)}(t) - \dots - \alpha_{j-1,j}F^{(j-1)}(t).$$

Например,

$$F^{(1)}(t) \equiv 1, \quad F^{(2)}(t) = t - \frac{n+1}{2}, \dots$$

Очевидно, что $F^{(j)}(t)$ – полином степени $j-1$ со старшим коэффициентом, равным единице. Условие $\langle f^{(i)}, f^{(j)} \rangle = 0$ при $i \neq j$ на языке полиномов запишется так

$$\sum_{k=1}^n F^{(i)}(k) \cdot \overline{F^{(j)}(k)} = 0.$$

Определение. Если $P^{(i)}(t), P^{(j)}(t) \in \mathcal{P}_n$ (полиномы порядка n), то число

$$\langle P^{(i)}, P^{(j)} \rangle = \sum_{k=1}^n P^{(i)}(k) \cdot \overline{P^{(j)}(k)} \quad (11.3.2)$$

называется *скалярным произведением* этих полиномов на стандартной сетке.

Замечание. Очевидно, что $\langle P, P \rangle \geq 0$ для любого $P \in \mathcal{P}_n$. Если $\langle P, P \rangle = 0$, то, очевидно, $P(k) = 0$ при $k = 1, \dots, n$, т.е. полином порядка n имеет по крайней мере n корней. Но полином с такими свойствами только один – нулевой. Итак, если $\langle P, P \rangle = 0$, то $P = \theta$.

Читателю предоставляется возможность самому проверить, что введенное на пространстве \mathcal{P}_n по формуле (11.3.2) "скалярное произведение" удовлетворяет и всем остальным свойствам скалярного произведения.

Естественно назвать полиномы $F^{(j)}$, $j = 1, \dots, n$ *полиномами, ортогональными на стандартной сетке*. Они, очевидно, образуют базис пространства \mathcal{P}_n (ортогональный в смысле скалярного произведения (11.3.2)).

Будем теперь искать сглаживающий полином в виде линейной комбинации полиномов из построенного базиса:

$$P(t) = p_1 F^{(1)}(t) + \dots + p_m F^{(m)}(t)$$

(так как степень полинома $F^{(j)}$ равна $j - 1$, $P(t)$ полином порядка m).

Для определения коэффициентов имеем, аналогично предыдущему пункту, систему линейных уравнений

$$fp = y, \quad (11.3.3)$$

где $f = [f^{(1)}, \dots, f^{(m)}]$ – матрица размера $n \times m$ с попарно ортогональными столбцами, p – искомый вектор коэффициентов, y – вектор результатов измерений.

Умножим теперь (11.3.3) слева на f^*

$$(f^* f)p = f^* y.$$

Решение этой системы (с диагональной матрицей) согласно формулам (10.3.3) имеет вид

$$p_k = \frac{\langle y, f^{(k)} \rangle}{\|f^{(k)}\|^2}, \quad k = 1, \dots, m.$$

Таким образом, при заготовленной заранее стандартной (при заданных n и m) матрице построение сглаживающего полинома сводится фактически к умножению матрицы на полученный в эксперименте столбец и делению координат столбца-произведения на также заранее заготовленные числа.

Мы рассмотрели два способа построения сглаживающего полинома: с помощью сингулярного разложения матрицы коэффициентов системы и с помощью ортогональных на сетке полиномов.

И тот и другой способ требует выполнения большой подготовительной работы: в первом случае – построения сингулярного разложения, во втором – построения ортогональных на сетке полиномов. И та и другая операция поддерживается устойчивыми вычислительными алгоритмами, реализованными как в средах конечного пользователя, так и в библиотеках программ на Фортране.

Мы не будем пытаться дать сравнительную оценку рассмотренных методов. Отметим лишь два факта.

1. Если мы построили сглаживающий полином, но среднеквадратическая погрешность оказалась велика и следует увеличить порядок полинома, то в случае сингулярного разложения придется пересчитывать все заново. При использовании же ортогональных на сетке полиномов потребуются добавить лишь еще одно слагаемое – уже сосчитанные коэффициенты разложения сохранятся. Это – полезное свойство *ортогональных* систем функций.

2. При использовании сингулярного разложения сглаживающий полином получается в стандартной форме,

$$P(t) = p_1 + p_1 t + \dots + p_m t^{m-1},$$

которая допускает применение схемы Горнера.

При использовании ортогональных на сетке полиномов сглаживающий полином получается в виде их линейной комбинации

$$P(t) = p'_1 F^{(1)}(t) + \dots + p'_m F^{(m)}(t).$$

Не следует приводить подобные члены в правой части (преобразовывать этот полином в стандартную форму), так как существует простое обобщение схемы Горнера, позволяющее работать с полиномом, разложенным по ортогональным на сетке полиномам.

11.4 Дискретное преобразование Фурье

Идея построения сглаживающего полинома с помощью ортогональных на сетке полиномов может быть обобщена. В самом деле, кроме полиномов существуют и другие системы "простых и удобных функций". В этом пункте мы рассмотрим одну такую систему.

Пусть $w^{(r)}(t) = \exp\left(i \frac{2\pi r}{n} t\right)$, $r \in \mathbb{Z}$. Покажем, что эти функции попарно ортогональны на стандартной сетке $\{1, \dots, n\}$. Действительно,

$$\langle w^{(j)}, w^{(r)} \rangle = \sum_{k=1}^n w^{(j)}(k) \overline{w^{(r)}(k)} = \sum_{k=1}^n \exp\left(i \frac{2\pi j}{n} k\right) \cdot \overline{\exp\left(i \frac{2\pi r}{n} k\right)} =$$

$$= \sum_{k=1}^n \exp\left(i \frac{2\pi}{n}(j-r)k\right) = \sum_{k=1}^n \left(\exp\left(i \frac{2\pi}{n}(j-r)\right)\right)^k.$$

Если $j = r$, то $\exp\left(i \frac{2\pi}{n}(r-r)\right) = 1$, и, следовательно,

$$\langle w^{(r)}, w^{(r)} \rangle = n. \quad (11.4.1)$$

Если же $j \neq r$, то, просуммировав геометрическую прогрессию, имеем

$$\begin{aligned} \langle w^{(j)}, w^{(r)} \rangle &= \exp\left(i \frac{2\pi}{n}(j-r)\right) \cdot \frac{\left(\exp\left(i \frac{2\pi}{n}(j-r)\right)\right)^n - 1}{\exp\left(i \frac{2\pi}{n}(j-r)\right) - 1} = \\ &= \frac{\exp\left(i \frac{2\pi}{n}(j-r)\right)}{\exp\left(i \frac{2\pi}{n}(j-r)\right) - 1} \cdot \left(\exp\left(i \frac{2\pi n}{n}(j-r)\right) - 1\right) = 0. \end{aligned}$$

Отметим некоторые свойства рассматриваемых функций.

1. Функции $w^{(r)}$ n -периодичны:

$$\begin{aligned} w^{(r)}(t+n) &= \exp\left(i \frac{2\pi r}{n}(t+n)\right) = \\ &= \exp\left(i \left(\frac{2\pi r}{n}t + 2\pi r\right)\right) = \exp\left(i \frac{2\pi r}{n}t\right) = w^{(r)}(t). \end{aligned}$$

2. На \mathbb{Z} существует только n различных функций $w^{(r)}$:

$$\begin{aligned} k \in \mathbb{Z} \implies w^{(r+n)}(k) &= \exp\left(i \frac{2\pi(r+n)}{n}k\right) = \\ &= \exp\left(i \left(\frac{2\pi r}{n}k + 2\pi k\right)\right) = \exp\left(i \frac{2\pi r}{n}k\right) = w^{(r)}(k). \end{aligned}$$

3. $w^{(n-r)}(k) = w^{(-r)}(k) = \exp\left(-i \frac{2\pi r}{n}k\right) = \overline{w^{(r)}(k)}$.

4. $w^{(n)}(k) \equiv w^{(0)}(k) \equiv 1$ при $k \in \mathbb{Z}$.

Сгладим теперь результаты наблюдений (п.11.2) линейной комбинацией функций $w^{(1)}, \dots, w^{(m)}$, $m \leq n$.

$$y_k = q_1 w^{(1)}(k) + \dots + q_m w^{(m)}(k). \quad (11.4.2)$$

В матричной форме система уравнений (11.4.2) имеет вид $Wq = y$, где

$$W = \begin{bmatrix} w^{(1)}(1) & \dots & w^{(m)}(1) \\ \dots & \dots & \dots \\ w^{(1)}(n) & \dots & w^{(m)}(n) \end{bmatrix} -$$

$(n \times m)$ -матрица с попарно ортогональными столбцами.

Переходя к нормальным уравнениям, получим $(W^*W)q = W^*y$ или, учитывая (11.4.1), $n \cdot q = W^*y$. Отсюда

$$q = \frac{1}{n} W^*y \quad \text{или} \quad q_r = \frac{1}{n} \langle y, w^{(r)} \rangle, \quad r = 1, \dots, m. \quad (11.4.3)$$

Если в сглаживании участвуют все функции ($m = n$), то нормальное псевдорешение превращается в решение – получаем разложение вектора в \mathbb{C}^n по ортогональному базису:

$$y_k = \sum_{r=1}^n q_r \exp\left(i \frac{2\pi r}{n}k\right), \quad k = 1, \dots, n, \quad (11.4.4)$$

где

$$q_r = \sum_{k=1}^n y_k \exp\left(-i \frac{2\pi r}{n}k\right), \quad r = 1, \dots, n, \quad (11.4.5)$$

Формулы (11.4.4) и (11.4.5) задают так называемое *дискретное преобразование Фурье*³, причем форму-

³Жан Батист Жозеф ФУРЬЕ (1768-1830) – французский математик, член Парижской АН, почетный член Петербургской АН.

ла (11.4.5), с помощью которой определяются координаты вектора q , называется *прямым преобразованием*, а формула (11.4.4), восстанавливающая исходный вектор y – *обратным преобразованием*.

Числа q_r называются *коэффициентами Фурье* вектора y или его *комплексным Фурье-спектром*. Отметим, что при аппроксимации *вещественного* вектора целесообразно выбирать попарно сопряженные на стандартной сетке функции $w^{(r)}$.

11.5. Быстрое преобразование Фурье

Для вычисления одного коэффициента Фурье по формуле (11.4.5) или для восстановления одной компоненты исходного вектора по формуле (11.4.4) требуется (при готовой матрице W) выполнить n пар "условных операций" (умножение+сложение). Таким образом, весь процесс вычисления дискретного преобразования Фурье потребует выполнения n^2 условных операций. При $n \approx 10^6 - 10^7$ (а такие массивы в приложениях не редки) обработка результатов измерений становится недоступной для современных ЭВМ.

В этом пункте рассматривается один из вариантов так называемого быстрого преобразования Фурье (БПФ). Будем считать, что $n = 2^p$, $p \in \mathbb{N}$.

Обозначим $z = \exp\left(i\frac{2\pi}{n}\right)$. Тогда, очевидно, $z^{n/2} = -1$, $z^n = 1$.

Формула (11.4.4) примет вид

$$y_k = \sum_{r=1}^n q_r z^{kr}, \quad k = 1, \dots, n. \quad (11.5.1)$$

Соберем в правой части (11.5.1) отдельно слагаемые с четными и с нечетными номерами (по предположению $n = 2^p$ – четное число):

$$\begin{aligned} y_k &= \sum_{r=1}^{n/2} (q_{2r} z^{2kr} + q_{2r-1} z^{k(2r-1)}) = \\ &= \sum_{r=1}^{n/2} (q_{2r} z^{2kr} + z^{-k} q_{2r-1} z^{2kr}) = x_k^{(e)} + z^{-k} x_k^{(o)}, \end{aligned} \quad (11.5.2)$$

где

$$x_k^{(e)} = \sum_{r=1}^{n/2} q_{2r} (z^2)^{kr}, \quad x_k^{(o)} = \sum_{r=1}^{n/2} q_{2r-1} (z^2)^{kr}. \quad (11.5.3)$$

Очевидно, что при $k = 1, \dots, n/2$

$$x_{k+n/2}^{(e)} = \sum_{r=1}^{n/2} q_{2r} (z^{2(k+n/2)})^r = \sum_{r=1}^{n/2} q_{2r} z^{nr} (z^2)^{kr} = x_k^{(e)}$$

и, аналогично, $x_{k+n/2}^{(o)} = x_k^{(o)}$.

Формулы (11.5.3) показывают, что столбцы $x_k^{(e)}$ и $x_k^{(o)}$ (высоты $n/2$ являются дискретными Фурье-образами столбцов, составленных соответственно из четных и нечетных компонент столбца q .

Преобразуем формулу (11.5.2):

$$\begin{cases} y_k = x_k^{(e)} + z^{-k} x_k^{(o)}, \\ y_{k+n/2} = x_{k+n/2}^{(e)} + z^{-n/2} z^{-k} x_{k+n/2}^{(o)} = x_k^{(e)} - z^{-k} x_k^{(o)} \end{cases}, \quad k = 1, \dots, n/2.$$

Видно, что для получения столбца y (высоты n) достаточно вычислить столбцы $x_k^{(e)}$ и $x_k^{(o)}$ (высоты $n/2$ и выполнить еще n условных операций.

Обозначим $O(p)$ количество условных операций, необходимых для вычисления дискретного преобразования Фурье столбца высоты $n = 2^p$. Тогда

$$O(p) = 2 \cdot O(p-1) + 2^p.$$

Учитывая, что $O(0) = 0$, методом математической индукции можно убедиться, что

$$O(p) = p \cdot 2^p = n \cdot \log_2(n).$$

При $n = 2^{20} \approx 10^6$ $n \cdot \log_2(n) = 20 \cdot 2^{20} < 2.1 \cdot 10^7$ – при помощи БПФ вычисления выполняются за несколько секунд, в то время как $n^2 = 2^{40} > 10^{12}$ – без использования БПФ требуемое время увеличивается почти в 10^5 раз!

Алгоритм БПФ реализован в средах конечного пользователя и в библиотеках Фортран-программ. Имеются варианты алгоритма, в которых высота столбца не обязана быть степенью двойки.