Тема 2. Аппроксимация функций и смежные вопросы.

Введение.

Этот раздел посвящен некоторым аспектам теории приближения функций. Термин «аппроксимация» (от латинского approximare – приближаться) в данном контексте трактуется как «замена».

Прежде, чем излагать какие-либо подходы к решению этой проблемы, целесообразно рассмотреть следующие вопросы. *Что нужно заменять и зачем заменять? Чем заменять? Как количественно оценивать погрешность замены? Наконец, если есть несколько вариантов замены, то, как выбрать из них наилучший?*

Итак, что заменять. На математическом языке это означает замену одной функциональной зависимости другой. Исходная функция чаще всего задается в одном из следующих видов:

аналитически,

графически,

таблично,

алгоритмически.

В последнем случае подразумевается, что аналитический вид функции неизвестен, но задан алгоритм (быть может, весьма сложный и трудоемкий!), ставящий в соответствие любому значению аргумента x из области определения значение функции $f(x)$. Так как на практике часто возникает потребность дифференцировать, интегрировать $f(x)$ или использовать ее в различных расчетах, то целесообразность замены $f(x)$ другой функцией, имеющей аналитический вид, сомнений не вызывает. Даже если $f(x)$ задана аналитически, она может оказаться весьма сложной с позиций решаемой задачи, и требуется другая заменяющая ее функция. При выборе аппроксимирующей функции обычно руководствуются ее простым видом.

Теоретическим обоснованием выбора заменяющей функции может служить *теорема Вейерштрасса:* Если $f(x)$ – произвольная непрерывная на конечном замкнутом интервале $[a, b]$ функция, то для любого $\varepsilon > 0$ найдется такой полином $P_n(x)$ степени $n = n(\varepsilon)$, что

$$\max\left| f(x) - P_n(x)\right| < \varepsilon .$$

Однако, эту теорему следует отнести к чистым теоремам существования. Она не дает гарантии, что такой полином можно построить с помощью практичного алгоритма. Тем не менее, именно полиномы являются наиболее популярными заменяющими функциями, поскольку они удобны в работе и их свойства хорошо известны. Наряду с ними используют тригонометрические, экспоненциальные функции и т.д. В общем случае требования к простоте выдвигает решаемая задача.

Для того, чтобы можно было сравнивать различные варианты аппроксимации, следует ввести критерий близости. В частности, им может быть максимум модуля отклонения исходной функции $f(x)$ от аппроксимирующей $g(x)$ на заданном промежутке

$$\delta = \max_{x\in[a,b]} \left| f(x) - g(x) \right| \tag{1}$$

или так называемый «среднеквадратичный критерий»

$$\rho^2 = \int_a^b (f(x) - g(x))^2 dx \tag{2}$$

В том случае, когда $f(x)$ определена таблично на заданном наборе точек, может быть использован дискретный аналог критерия (2)

$$\rho^2 = \sum_{k=1}^{m} (f(x_k) - g(x_k))^2 \tag{3}$$

Лучшей оказывается аппроксимирующая функция, обладающая наименьшей величиной δ или ρ^2. Для понимания дальнейшего полезно ответить на следующие вопросы.

***Вопрос* 1**. *Не являются ли критерии* (1) *и* (2) *весьма похожими и если аппроксимирующая функция* $g1(x)$ *лучше, чем* $g2(x)$ *по одному из критериев, то окажется ли она лучше и по другому?*

Легко заметить, что вопрос носит провокационный характер, о чем свидетельствует пример на рис.1, когда лучшими по каждому критерию оказываются различные функции.

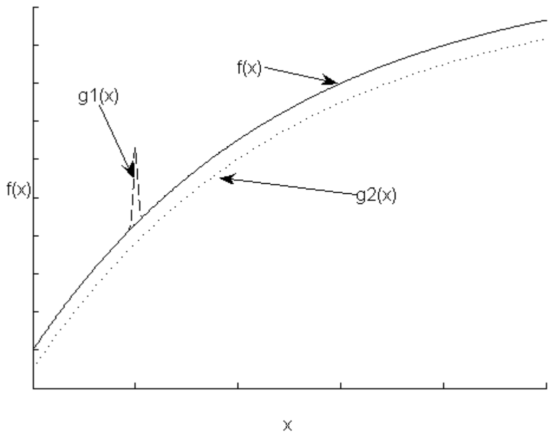

Рис 1. Различные варианты аппроксимации

***Вопрос* 2**. *Ну а на практике, какая из аппроксимирующих функций на рис.*1 *предпочтительнее, g1*(x) *или g2*(x)*?*

И опять вопрос носит провокационный характер, так как словосочетание «на практике» еще не задает критерий. Только решаемая задача диктует выбираемый критерий близости, который, в свою очередь, позволяет выбрать лучшую аппроксимацию.

Теперь приступим к подробному изложению задачи интерполяции, основанной на интерполяционном критерии близости.

Постановка задачи интерполирования.

Будем приближать исходную функцию, заданную таблично

x	x_0	x_1	x_2	…	x_m
$f(x)$	$f(x_0)$	$f(x_1)$	$f(x_2)$	…	$f(x_m)$

обобщенным многочленом

$$Q_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \ldots + a_m\varphi_m(x) = \sum_{k=0}^{m} a_k\varphi_k(x), \tag{1}$$

где $\{\varphi_k\}$ – заданный набор линейно независимых функций, а коэффициенты a_k подлежат определению. При этом в качестве критерия близости выбирается совпадение значений $f(x)$ и $Q_m(x)$ в узлах таблицы

$$Q_m(x_i) = f(x_i), \qquad i = 0,1,\ldots,m. \tag{2}$$

Тогда $Q_m(x)$ называется *интерполяционным многочленом*, а x_k – *узлами интерполирования*.

Равенства (2) представляют собой систему линейных алгебраических уравнений относительно искомых коэффициентов обобщенного многочлена $a_0, a_1, \ldots, a_m$. Эта система имеет единственное решение, если ее определитель отличен от нуля:

$$\det\begin{pmatrix} \varphi_0(x_0) & \varphi_1(x_0) & \ldots & \varphi_m(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \ldots & \varphi_m(x_1) \\ \ldots & \ldots & \ldots & \ldots \\ \varphi_0(x_m) & \varphi_1(x_m) & \ldots & \varphi_m(x_m) \end{pmatrix} \neq 0$$

Наиболее популярной является полиномиальная аппроксимация, когда $\varphi_k(x) = x^k$, $Q_m(x) = a_0 + a_1x + a_2x^2 + \ldots + a_mx^m$, а определитель системы (2) приобретает вид

$$\begin{vmatrix} 1 & x_0 & x_0^2 & . & x_0^n \\ 1 & x_1 & x_1^2 & . & x_1^n \\ 1 & x_2 & x_2^2 & . & x_2^n \\ . & . & . & . & . \\ 1 & x_n & x_n^2 & . & x_n^n \end{vmatrix}. \qquad (3)$$

Этот определитель называется определителем Вандермонда. Он отличен от нуля, и задача имеет единственное решение, если узлы интерполирования $x_0, x_1, ..., x_m$ различны.

Интерполяционный полином Лагранжа. Остаточный член полинома Лагранжа.

Непосредственное численное решение линейной системы (2) представляет значительные трудности. С одной стороны, это связано с заметным объемом вычислений для нахождения a_k. С другой стороны, малое изменение данных таблицы (x_k , $f(x_k)$) часто приводит к сильному изменению решения (особенно для близко расположенных узлов интерполирования). В связи с этим, естественно попытаться построить интерполяционный полином, не решая системы (2).

С этой целью введем следующие функции

$$\omega(x) = (x - x_0)(x - x_1)...(x - x_m)$$

$$\omega_k(x) = \frac{\omega(x)}{(x - x_k)} = (x - x_0)(x - x_1)...(x - x_{k-1})(x - x_{k+1})...(x - x_m)$$

В этих обозначениях запишем следующий полином

$$Q_m(x) = \sum_{k=0}^{m} \frac{\omega_k(x)}{\omega_k(x_k)} f(x_k) \qquad (4)$$

По построению это многочлен степени m. Определим его значения в узлах интерполирования x_i . Так как для $x = x_i$ полином $\omega_k(x)$ равен нулю, если только $i \neq k$, то для $Q_m(x_i)$ получаем

$$Q_m(x_i) = \sum_{k=0}^{m} \frac{\omega_k(x_i)}{\omega_k(x_k)} f(x_k) = \frac{\omega_i(x_i)}{\omega_i(x_i)} f(x_i) = f(x_i)\ , \quad i=0,\ 1, ...m$$

Таким образом, $Q_m(x)$ – интерполяционный полином, получивший название *интерполяционный полином Лагранжа*.

***Вопрос* 3**. *По заданной таблице при отсутствии ошибок округления построили два полинома. Коэффициенты первого нашли решением системы* (2). *Второй был непосредственно воспроизведен по формуле* (4). *Какой из этих полиномов лучше приближает исходную функцию?*

Интерполяционные многочлены можно строить и по другим системам базисных функций, например, таким как

$$1, \cos x, \sin x, \cos 2x, \sin 2x, \ldots, \cos nx, \sin nx, \ldots$$

$$1, e^{\alpha_1 x}, e^{\alpha_2 x}, \ldots e^{\alpha_n x} \ldots$$

Теперь обратимся к погрешности интерполяционного полинома. Исходная функция $f(x)$ может быть представлена в виде

$$f(x) = Q_m(x) + R_m(x), \qquad (5)$$

где $Q_m(x)$ – интерполяционный полином, а $R_m(x)$ носит название *остаточного члена интерполяционного полинома.* Вид $R_m(x)$ получается на основе известной теоремы Ролля, которая утверждает следующее.

Если $f(x)$ непрерывна на отрезке $[a, b]$, имеет первую производную в каждой точке внутри этого отрезка, и значения функции на концах этого промежутка равны, т.е. $f(a)=f(b)$, то внутри отрезка найдется, по крайней мере, одна такая точка $x = c$, что $f'(c) = 0$.

<u>*Теорема*</u>: Если $f(x)$ на промежутке $[a, b]$ имеет непрерывные производные вплоть до $m+1$ порядка, то остаточный член $R_m(x)$ можно представить в виде:

$$R_m(x) = f(x) - Q_m(x) = \frac{f^{(m+1)}(\eta)}{(m+1)!}\omega(x), \qquad \eta \in [a, b]. \qquad (6)$$

При этом $\omega(x)$, определяется, как и прежде.

Доказательство. Рассмотрим вспомогательную функцию

$$\varphi(z) = f(z) - Q_m(z) - K\omega(z), \qquad (7)$$

где K – некоторая постоянная. Пусть x_k – узлы интерполирования, а x – точка, в которой оценивается погрешность $\left(x \neq x_k\right)$. Легко заметить, что функция $\varphi(z)$ равна нулю во всех узлах интерполирования. Выберем константу K так, чтобы $\varphi(x) = 0$

$$K = \frac{f(x) - Q_m(x)}{\omega(x)} = \frac{R_m(x)}{\omega(x)}. \qquad (8)$$

Таким образом, $\varphi(z)$ имеет по меньшей мере $m+2$ нуля (все узлы интерполирования и точка x). Тогда по теореме Ролля первая производная $\varphi(z)$ имеет по меньшей мере $m+1$ нуль, вторая производная – не менее m нулей, а $(m+1)$-я производная $\varphi^{(m+1)}(z)$ имеет по меньшей мере один нуль. Обозначим точку, где $(m+1)$-я производная обращается в нуль за η. Тогда, последовательно дифференцируя (7), получаем

$$\varphi^{(m+1)}(\eta) = f^{(m+1)}(\eta) - 0 - K(m+1)! = 0.$$

Подставляя в это равенство выражение (8) для K, получаем формулу для $R_m(x)$, совпадающую с (6).

Формула (6) позволяет сделать очевидный, но важный вывод. Пусть $f(x)$ – это полином степени m. Тогда $f^{(m+1)}(\eta) = 0$. Следовательно, полином степени m *однозначно* воспроизводится интерполяционным полиномом по $m+1$ точке.

Ясно также, что остаточный член во всех узлах интерполирования равен нулю.

В заключение отметим тот факт, что, хотя о расположении точки η на промежутке интерполирования ничего неизвестно, очевидна зависимость величины η, как от узлов интерполирования, так и от точки x, где оценивается погрешность, т.е. $\eta = \eta(x)$.

Остаточный член позволяет оценивать отклонение $L_m(x)$ от $f(x)$ для дифференцируемых функций тогда, когда удается оценить $f^{(m+1)}(x)$. Полагая $M_{m+1} = \max\left|f^{(m+1)}(x)\right|$, , получим $\left|R_m(x)\right| \le \dfrac{M_{m+1}}{(m+1)!}\left|\omega(x)\right|$.

Выбор узлов интерполирования.

Для уменьшения погрешности интерполирования обратимся к формуле (6) при заданной степени полинома m. Поскольку величиной $f^{(m+1)}(\eta)$ трудно управлять, и возможна лишь оценка пределов ее изменения, задача уменьшения погрешности сводится к управлению величиной $\left|\omega(x)\right|$ за счет выбора узлов интерполирования. Рассмотрим два типичных на практике случая.

Случай 1. Задана степень полинома m и имеется таблица достаточно большой длины. Точка x^*, в которой вычисляется значение полинома, заранее известна. Требуется выбрать $m+1$ узел так, чтобы величина $\left|\omega(x^*)\right|$ была бы минимальна.

Результат очевиден. Нужно выбирать узлы интерполирования из таблицы, *ближайшие* к x^*. Использование любого другого узла вместо ближайшего неизбежно увеличивает значение

$$\left|\omega(x^*)\right| = \left|\left(x^* - x_0\right)\left(x^* - x_1\right)\ldots\left(x^* - x_m\right)\right|.$$

Случай 2. Заданы степень полинома m и промежуток интерполирования $[a, b]$. Точка x^*, в которой вычисляется значение полинома, заранее не известна. Требуется выбрать узлы интерполирования так, чтобы в самом неблагоприятном случае расположения x^* погрешность была бы минимальна (так называемый *минимаксный критерий*)

$$\max_{x^* \in [a,b]} \left|\omega(x^*)\right| \to \min.$$

***Вопрос* 4**. *С учетом того, что величина x^* заранее неизвестна и нет оснований отдавать предпочтение какой-либо части промежутка, как следует задавать узлы x_k на $[a, b]$?*

Интуитивно часто напрашивающееся предложение о равномерном задании узлов на промежутке оказывается ошибочным. Так как значения $|\omega(x)|$ в узлах интерполирования равны нулю, график $|\omega(x)|$ напоминает «колокольчики», максимум которых достигается между узлами интерполирования. Одинакова ли высота этих «колокольчиков» на различных промежутках между узлами? Оказывается, что нет. Выбор узлов равноотстоящими с попыткой уравновесить погрешность на отдельных участках промежутка $[a, b]$, оказывается здесь неудачным и точки, расположенные ближе к центру промежутка, попадают в привилегированное положение. Поясним это на примере.

Расположим на промежутке [0, 5] равномерно 6 узлов с шагом h=1 ($x_k=k$, k=0,1,2,3,4,5).

Для точки x^* = 2.5 такие узлы, как это рассматривалось в *случае 1*, оказываются наилучшими при единичном шаге $\left(|\omega(2.5)|=2.5*1.5*0.5*0.5*1.5*2.5 \approx 3.52\right)$. В то же время для x^* = 0.5 получаем $|\omega(0.5)| = 0.5*0.5*1.5*2.5*3.5*4.5 \approx 14.77$. Здесь в соответствии с рекомендациями к *случаю* 1 вместо узлов $x_4 = 4$ и $x_5 = 5$ следовало бы взять ближайшие узлы $x_4 = -1$ и $x_5 = -2$, но они в таблице отсутствуют. График $|\omega(x)|$, представленный на рис.2, не только отражает сказанное, но подсказывает выход из создавшейся ситуации. Узлы интерполирования нужно симметрично сместить ближе к концам промежутка. Тогда высота центрального «колокольчика» увеличится, в то время как высота крайних уменьшится. Оптимальный выбор узлов интерполирования отвечает нулям так называемых ортогональных полиномов Чебышева (о них речь пойдет ниже), когда все «колокольчики» будут одинаковыми по высоте. Соответствующий график представлен на рис.3, а значение x_k определяются по формулам

$$x_i = \frac{b+a}{2} + \frac{b-a}{2}\xi_i, \text{ где } \xi_i = -\cos\frac{2i+1}{2m+2}\pi, \; i=0..m$$

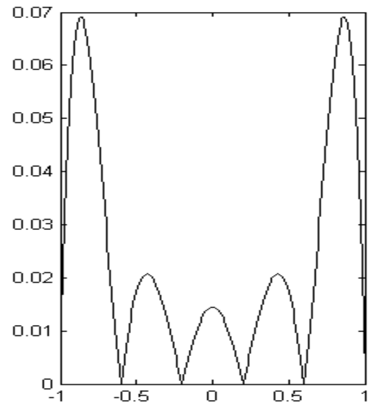

Рис.2. Равноотстоящие узлы

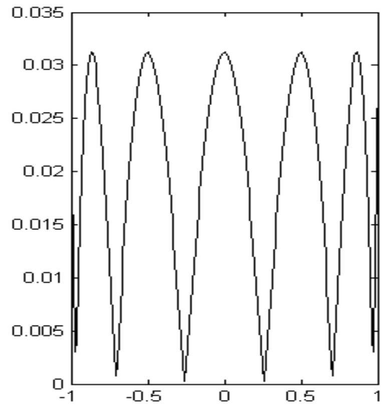

Рис.3. Чебышевские узлы

Интерполирование сплайнами.

На практике интерполяционные полиномы высоких степеней строят крайне редко. В первую очередь, это связано с тем, что их коэффициенты очень чувствительны к погрешностям исходных данных. Сравнительно малое изменение узлов интерполирования x_k или значений функции $f(x_k)$ приводит к сильному изменению вида самого полинома. В такой ситуации одним из возможных вариантов аппроксимации является разбиение большой исходной таблицы на участки, для каждого из которых строится интерполяционный полином относительно невысокой степени. Этот подход используется, например, при получении составных квадратурных формул, рассматриваемых ниже. Однако, в целом ряде приложений требуется, чтобы аппроксимирующая функция была гладкой, а функция, составленная из различных полиномов, в узлах сопряжения не имеет производной. Выходом из создавшегося положения является использование сплайн-интерполяции.

Сплайн (*spline* – англ.) – это длинная гибкая тонкая рейка, используемая чертежниками в качестве лекала для проведения гладких кривых через заданные точки. Математическое осмысление этого инструмента и породило теорию сплайнов, аппарат которой заметно выходит за рамки описания механического сплайна. Расположив чертеж в вертикальной плоскости и закрепив рейку, в узлах интерполяции к ней подвешивают грузила и добиваются, чтобы деформированная рейка совместилась со всеми точками. Сплайн принимает форму, отвечающую минимуму его потенциальной энергии. Функция, описывающая профиль сплайна, является кубическим полиномом между любыми двумя соседними точками. Кроме этого, соседние полиномы соединяются непрерывно и гладко так же, как и их первые и вторые производные (рейка не разламывается).

От механической иллюстрации перейдем к формальному аппарату сплайн-интерполяции. Обратимся к таблично заданной функции

x	x_1	x_2	x_3		…	x_N
$f(x)$	$f(x_1)$	$f(x_2)$	$f(x_3)$		…	$f(x_N)$

Число узлов равно N, а их нумерация начинается с единицы. На каждом промежутке $[x_k , x_{k+1}]$ будем строить интерполяционный полином третьей степени

$$S_k(x) = a_k + b_k\left(x - x_k\right) + c_k\left(x - x_k\right)^2 + d_k\left(x - x_k\right)^3. \qquad (14)$$

Количество полиномов, как и промежутков, равно N–1 и каждый полином имеет 4 параметра. Таким образом, всего в наличии $4N$–4 параметра. Потребуем, чтобы во всех внутренних точках были равны значения соседних полиномов, их первых и вторых производных

$$S_k(x_{k+1}) = S_{k+1}(x_{k+1}); \quad S'_k(x_{k+1}) = S'_{k+1}(x_{k+1}); \quad S''_k(x_{k+1}) = S''_{k+1}(x_{k+1}); \; k=1,\ldots N\text{–}2, \qquad (15)$$

то есть выполнялись суммарно $3(N–2) = 3N–6$ уравнений. Еще N уравнений отражают требования интерполирования

$$S_k(x_k) = f_k, \quad k = 1, 2, ..., N-1; \quad S_{N-1}(x_N) = f_N \tag{16}$$

и общее число задаваемых условий достигает $4N–6$. При наличии $4N–4$ параметров появляется возможность выполнить еще два условия. В их задании нет острой необходимости, так как требования интерполирования и сопряжения соседних полиномов уже выполнены, но это целесообразно сделать для однозначного решения задачи. Различные кубические сплайны отличаются друг от друга заданием этих двух требований, которые, как правило, записываются для двух крайних точек x_1 и x_N. К этим двум дополнительным условиям целесообразно предъявить следующие два требования. С одной стороны, их лучше задавать так, чтобы полная система уравнений решалась по возможности более просто. С другой стороны, они должны максимально соответствовать характеру поведения функции в начале и в конце промежутка интерполирования. Приведем два примера.

Пример 1. $S_1''(x_1) = 0$; $S_{N-1}''(x_N) = 0$. Этот сплайн получил название *естественного кубического сплайна*. Однако, такое название оправдывается, и выполнение второго требования имеет место только в механике, пока анализируется упомянутый механический сплайн. В общем случае равенство нулю второй производной на краях промежутка не является обязательным свойством экспериментальных данных, отражаемых таблицей.

Пример 2. По первым четырем точкам таблицы строится интерполяционный полином третьей степени $Q_3(x)$, и его третья производная приравнивается третьей производной $S_1(x)$. Аналогично по последним четырем точкам строится интерполяционный полином $\tilde{Q}_3(x)$, и его третья производная приравнивается третьей производной последнего полинома $S_{N-1}(x)$

$$Q_3'''(x_1) = S_1'''(x_1); \qquad \tilde{Q}_3'''(x_N) = S_{N-1}'''(x_N). \tag{17}$$

Такие условия не только отвечают характеру поведения функции в начале и в конце промежутка интерполирования, но и достаточно просты (третья производная от полинома третьей степени постоянна). Именно они и учитываются в рассматриваемых программах ***SPLINE*** и ***SEVAL***. Процедура решения системы линейных алгебраических уравнений (15) – (17) для $4N–4$ неизвестных коэффициентов полиномов (14) может быть заметно упрощена. Подробнее с этим можно ознакомиться по учебному пособию.

Программное обеспечение, с текстами которого, как и с другими программами, обсуждаемыми в лекциях, можно ознакомиться в работе [14], состоит из двух процедур.

Первая из них – ***SPLINE(N, X, F, B, C, D)***, оформленная как процедура, решает систему уравнений относительно b_k, c_k, d_k.

Здесь ***N*** – число точек;

X и ***F*** – векторы, элементами которых являются x_k и f_k;

B, C, D – векторы с коэффициентами b_k, c_k, d_k полиномов (14) – результаты работы ***SPLINE***.

Вторая программа ***SEVAL(N, U, X, F, B, C, D)***, оформленная как функция, использует результаты работы ***SPLINE*** и вычисляет значение сплайна в заданной точке ***U***.

Вопрос. *Почему в списке параметров процедур отсутствуют коэффициенты* $\boldsymbol{a_k}$ *полиномов (14)?*

Это следует из специальной формы записи (14), позволяющей сразу сократить число неизвестных на четверть. Требование интерполирования сразу позволяет найти $\boldsymbol{a_k}$

$$S_k(x_k) = a_k = f_k, \qquad k = 1, 2, ..., N-1. \tag{18}$$

Интерполяционный полином Эрмита

Все сказанное до этого относилось к задаче интерполирования по значениям функции. Если в таблице помимо значений функции присутствуют ее производные и от интерполяционного полинома требуется совпадение с данными этой таблицы, то такая задача называется *интерполированием по Эрмиту*. Познакомимся с ней на отдельных примерах.

Для следующих исходных данных

x	x_0	x_1	x_2
$f(x)$	$f(x_0)$	$f(x_1)$	$f(x_2)$
$f'(x)$	$f'(x_0)$	$f'(x_1)$	-----
$f''(x)$	------	$f''(x_1)$	-----

требуется построить интерполяционный полином $H(x)$, удовлетворяющий всем условиям таблицы. Прежде, чем это будет сделано, целесообразно ответить на следующие вопросы.

Вопрос 6. *Какой степени получится полином Эрмита? Как найти его коэффициенты? Если по этим же трем узлам построить полином Лагранжа, используя только значения* f_k, *то какой из двух полиномов будет приближать функцию успешнее и почему?*

Потребуем от $H(x)$ выполнения условий таблицы

$$H(x_k) = f(x_k), \qquad k = 0,\ 1,\ 2$$
$$H'(x_k) = f'(x_k), \qquad k = 0,\ 1; \tag{19}$$

$H''(x_k) = f''(x_k), \quad k = 1.$

Система (19) содержит шесть уравнений. Для ее однозначного решения полином $H(x)$ должен иметь 6 коэффициентов, т.е. быть полиномом пятой степени. Общее правило очевидно: *степень интерполяционного полинома Эрмита на единицу меньше общего числа условий таблицы*.

В задаче интерполирования по значениям функции уже возникала система уравнений, аналогичная системе (19). Тогда оказалось возможным построить, например, полином Лагранжа, не решая этой системы. Нельзя ли и в случае полинома Эрмита воспроизвести его сразу в готовом виде, не решая системы, подобной (19)? Ответ на этот вопрос положительный. Однако общая формула является громоздкой, и мы не будем ее приводить. При этом форма записи полинома будет проще, если исходная таблица симметричная, т.е. число и вид условий во всех узлах одинаковые (в отличие от системы (19)).

В качестве еще одной иллюстрации обратимся к разложению функции $f(x)$ в степенной ряд Тейлора в точке x_0

$$f(x) = f(x_0) + \frac{x - x_0}{1!} f'(x_0) + \frac{(x - x_0)^2}{2!} f''(x_0) + \frac{(x - x_0)^3}{3!} f'''(\eta) = H_2(x) + \frac{(x - x_0)^3}{3!} f'''(\eta).$$

Полином второй степени $H_2(x)$,

$$H_2(x) = f(x_0) + \frac{x - x_0}{1!} f'(x_0) + \frac{(x - x_0)^2}{2!} f''(x_0) \qquad (20)$$

с одной стороны, является частной суммой ряда Тейлора, а с другой стороны, удовлетворяет условиям

$H_2(x_0) = f(x_0), \qquad H_2'(x_0) = f'(x_0), \qquad H_2''(x_0) = f''(x_0),$

что позволяет назвать его одновременно и интерполяционным полиномом Эрмита с одним узлом интерполирования.

В заключение настоящего параграфа зададим еще один вопрос, дополняющий *вопросы 3* и *5*.

***Вопрос** 7. На промежутке* $[x_0, x_2]$ *по трем узлам интерполирования* x_0, x_1, x_2 *построим интерполяционный полином Лагранжа второй степени* $Q_2(x)$, *а затем построим полином Эрмита* $H_2(x)$, *в соответствии с формулой* (20). *Оба полинома являются полиномами второй степени. Какой из них приближает исходную функцию* $f(x)$ *лучше и почему?*

Обратная интерполяция

До настоящего параграфа решалась прямая задача интерполирования. По известной таблице

x	x_0	x_1	x_2	…	x_m
$f(x)$	$f(x_0)$	$f(x_1)$	$f(x_2)$	…	$f(x_m)$

и заданному значению x^* требуется оценить значение функции $f(x^*)$. В обратной задаче для такой же таблицы требуется восстановить значение аргумента x^*, для которого функция принимает заданное значение f^*. На практике чаще всего используется один из следующих очевидных способов.

Способ 1. Меняются местами строки таблицы, в качестве узлов интерполирования выбираются значения f_k, а в качестве значений функции x_k, и строится интерполяционный полином для обратной функции. Подставляя в него f^*, находим желаемое x^*. Такой подход вполне жизнеспособен, если существует обратная функция, то есть исходная функция является строго монотонной. В противном случае необходимо делить таблицу на части с участками строгой монотонности.

Например, обратный квадратичный полином для фрагмента таблицы:

$f(x)$	$f(x_{k-2})$	$f(x_{k-1})$	$f(x_k)$
x	x_{k-2}	x_{k-1}	x_k

будет иметь вид

$$L_2(f) = \frac{(f - f_{k-1})(f - f_k)}{(f_{k-2} - f_{k-1})(f_{k-2} - f_k)} x_{k-2} + \frac{(f - f_{k-2})(f - f_k)}{(f_{k-1} - f_{k-2})(f_{k-1} - f_k)} x_{k-1} + \frac{(f - f_{k-2})(f - f_{k-1})}{(f_k - f_{k-2})(f_k - f_{k-1})} x_k$$

Способ 2. По исходной таблице строится обычный интерполяционный полином $Q_m(x)$ с узлами x_k, а затем решается уравнение $Q_m(x) = f^*$. Для полиномов до четвертой степени ответ может быть получен даже аналитически, а в других случаях это уравнение решается численно, например, одним из методов решения нелинейных уравнений, рассматриваемых далее. Здесь нет необходимости принимать во внимание монотонность. Для немонотонной функции уравнение будет иметь несколько корней на промежутке интерполирования, и из них лишь нужно выбрать отвечающий поставленной задаче.