РАБОТА № 1

Интерполяция зависимостей

1. Постановка задачи

Основу математических моделей многих процессов и явлений в физике, химии, биологии, экономике и др. областях составляют уравнения различного вида: нелинейные уравнения, обыкновенные дифференциальные уравнения первого и второго порядков, дифференциальные уравнения в частных производных и т.д. Для решения подобных уравнений необходимо иметь возможность вычислять значения функций, входящих в описание математической модели рассматриваемого процесса или явления, при произвольном значении аргумента.

Используемые в математических моделях функции могут быть заданы как аналитическим способом, так и табличным, при котором функция известна только при дискретных значениях аргумента. В частности если функциональная зависимость получена в результате расчетов, проведенных на ЭВМ или в процессе измерений, осуществленных в рамках какого-либо эксперимента, то она оказывается заданной именно табличным способом. В этом случае возникает проблема в вычислении значений функций при произвольном значении аргумента.

Пусть функция *y(x)* задана множеством своих значений для дискретного набора точек (т.е. таблицей):

x	x_0	x_1	…	x_n
y(x)	y_0	y_1	…	y_n

Требуется найти значения функции *y(x)* для $x \neq x_i$.

Поставленная проблема решается путем приближенной замены функции *y(x)* другой функцией $\varphi(x)$, заданной аналитическим выражением, которую можно вычислить при любом значении аргумента *x* в заданном интервале его изменения.

Приближение функции *y(x)* более простой функцией $\varphi(x)$ называется *аппроксимацией* (от латинского *approximo* – приближаюсь).

Часто аппроксимирующую функцию $\varphi(x)$ строят таким образом, чтобы ее значения в узловых точках x_i $(i = 0, 1, 2, \ldots, n)$ совпадали с табличными значениями заданной функции *y(x)*:

$$\varphi(x_0) = y_0, \quad \varphi(x_1) = y_1, \ldots, \quad \varphi(x_n) = y_n. \tag{1}$$

Такой способ введения аппроксимирующей функции называют лагранжевой интерполяцией, а условия (1) – условиями Лагранжа. Аппроксимирующая функция $\varphi(x)$, удовлетворяющая условиям Лагранжа, называется *интерполяционной функцией*.

Задача интерполяции состоит в нахождение приближенных значений табличной функции при аргументах *x*, не совпадающих с узловыми, путем вычисления значений интерполяционной функции $\varphi(x)$. Если значение аргумента расположено внутри интервала $[x_0, x_n]$, то нахождение приближенного значения функции *y(x)* называют *интерполяцией*, если аппроксимирующую функцию вычисляют вне интервала $[x_0, x_n]$, то процесс называют *экст-*

раполяцией. Происхождение этих терминов связано с латинскими словами: *inter* – между, внутри, *extra* – вне, *pole* – узел.

2. Интерполяция полиномом степени *n*

2.1. Полином в каноническом виде

Выберем в качестве аппроксимирующей функции $\varphi(x)$ полином степени *n* в каноническом виде

$$\varphi(x) = P(x) = c_0 + c_1 x + c_2 x^2 + \ldots + c_n x^n \ . \qquad (2)$$

Коэффициенты полинома c_i определяются из условий Лагранжа $P(x_i) = y_i$, $i = 0, 1, \ldots, n$, что с учетом выражения (2) дает систему уравнений с *n+1* неизвестными:

$$\begin{aligned} c_0 + c_1 x_0 + c_2 x_0^2 + \ldots + c_n x_0^n &= y_0 \\ c_0 + c_1 x_1 + c_2 x_1^2 + \ldots + c_n x_1^n &= y_1 \\ \cdots\cdots\cdots\cdots \\ c_0 + c_1 x_n + c_2 x_n^2 + \ldots + c_n x_n^n &= y_n \end{aligned} \ . \qquad (3)$$

Систему уравнений (3) можно кратко записать следующим образом

$$\sum_{k=0}^{n} c_k x_i^k = y_i \ , \quad i = 0, 1, \ldots n \qquad (3')$$

или в матричной форме

$$\mathbf{A} \cdot \mathbf{c} = \mathbf{y} \ ,$$

где **c** – вектор-столбец, содержащий неизвестные коэффициенты c_i, **y** – вектор-столбец, составленный из табличных значений функции y_i, а матрица **A** имеет вид

$$\mathbf{A} = \begin{pmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^n \\ 1 & x_1 & x_1^2 & \ldots & x_1^n \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & x_n & x_n^2 & \ldots & x_n^n \end{pmatrix} .$$

Система линейных алгебраических уравнений (3) относительно неизвестных c_i будет иметь решение, если определитель системы (определитель матрицы **A**) отличен от нуля. Определитель матрицы **A**, известный в алгебре как определитель Вандермонда, имеет аналитическое выражение:

$$\det \mathbf{A} = \prod_{\substack{i,j=0 \\ (i \neq j)}}^{n} (x_i - x_j) \neq 0 \ .$$

Из этого выражения видно, что $\det \mathbf{A} \neq 0$, если среди узлов x_i нет совпадающих.

Решение системы уравнений (3) представляет собой самостоятельную и достаточно трудоемкую вычислительную задачу. Но использование математических пакетов, в частности – Mathcad, позволяет решить ее удивительно легко и изящно. Достаточно, просто записать решение в матричном виде:

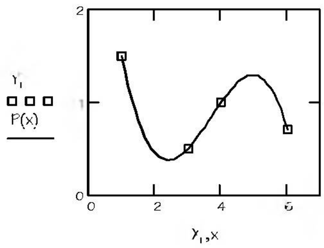
$$\mathbf{c} = \mathbf{A}^{-1} \cdot \mathbf{y},$$

где $\mathbf{A}^{-1}$ – обратная матрица ($\mathbf{A}^{-1} \cdot \mathbf{A} = \mathbf{I}$, $\mathbf{I}$ – единичная диагональная матрица).

Пример полиномиальной интерполяции:

$$X := \begin{pmatrix} 1 \\ 3 \\ 4 \\ 6 \end{pmatrix} \quad Y := \begin{pmatrix} 1.5 \\ 0.5 \\ 1 \\ 0.7 \end{pmatrix} \quad i := 0..3 \quad j := 0..3 \quad x := 1, 1.1..6$$

$$A_{i,j} := (X_i)^j \qquad c := A^{-1} \cdot Y \qquad P(x) := \sum_{k=0}^{3} c_k \cdot x^k$$



2.2. Интерполяционный полином Лагранжа

Для интерполяционного полинома $P(x)$ можно получить выражение в явном виде:

$$P(x) \equiv L(x) = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}. \tag{4}$$

Полином, записанный в форме (4) называется *интерполяционным полиномом Лагранжа*.

Старшая степень аргумента x в полиноме Лагранжа равна n, так как каждое произведение в формуле (4) содержит n сомножителей $x - x_i$. В узлах $x = x_i$ выполняются условия Лагранжа, потому что в сумме остается по одному слагаемому y_i, остальные обращаются в нуль за счет нулевых сомножителей в произведениях.

В отличии от интерполяционного полинома в канонической форме для вычисления значений полинома Лагранжа не требуется предварительно определять коэффициенты полинома путем решения системы уравнений. Однако для каждого значения аргумента x полином Лагранжа приходится пересчитывать вновь, коэффициенты же канонического полинома вычисляются только один раз. Поэтому практическое применение полинома Лагранжа оправдано только в том случае, когда интерполяционная функция вычисляется в сравнительно небольшом количестве точек x.

4. Многоинтервальная интерполяция

Рассмотренная выше полиномиальная интерполяция не всегда дает удовлетворительные результаты. Несмотря на выполнение условий Лагранжа в узлах, интерполяционная функция может иметь значительное отклонение от аппроксимируемой кривой между узлами. С увеличением количества узлов возрастает и степень интерполяционного полинома, что приводит к резкому увеличению погрешности в результате возникновения так называемого явления волнистости. Для того чтобы избежать высокой степени полинома отрезок интерполяции разбивают на несколько частей и на каждом частичном интервале строят самостоятельный полином невысокой степени. Ниже рассматриваются наиболее часто используемые виды многоинтервальной интерполяции, а также способы их реализации в Mathcad'е.

4.1. Кусочно-линейная интерполяция

Кусочно-линейная интерполяция является простейшим видом многоинтервальной интерполяции, при которой исходная функция на каждом частичном интервале $[x_i, x_{i+1}]$ аппроксимируется отрезком прямой, соединяющим точки (x_i, y_i) и (x_{i+1}, y_{i+1}):

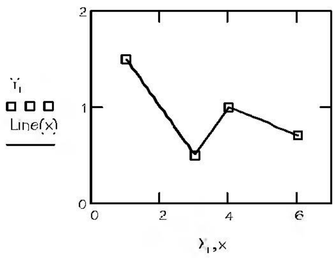
$$y(x) \approx y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i).$$

В Mathcad'е имеется встроенная функция, специально предназначенная для такого вида интерполяции – **linterp(vx, vy, x)** – линейная интерполяция. Аргументами этой функции являются два вектора **vx** и **vy**, содержащие исходные данные и независимая переменная **x**.

Пример использования функции **linterp**:

$$X := \begin{pmatrix} 1 \\ 3 \\ 4 \\ 6 \end{pmatrix} \quad Y := \begin{pmatrix} 1.5 \\ 0.5 \\ 1 \\ 0.7 \end{pmatrix} \quad i := 0..3 \quad x := 1, 1.1..6$$

$$\mathrm{Line}(x) := \mathrm{linterp}(X, Y, x)$$



4.2. Кусочно-квадратичная интерполяция

Если рассмотреть интервал, содержащий три узловых точки, например, x_{i-1}, x_i и x_{i+1}, то аналогично можно построить интерполяционный полином второй степени (т.е. параболу):

$$y(x) \approx y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i) + \frac{y_{i+1} - 2y_i + y_{i-1}}{2(x_i - x_{i-1})^2}(x - x_i)(x - x_{i+1}).$$

4.3. Сплайн-интерполяция

Существенным недостатком кусочной интерполяции является то, что в точках стыка разных интерполяционных полиномов оказывается разрывной их первая производная. Этот недостаток устраняется при использовании особого вида многоинтервальной интерполяции – интерполяции сплайнами (англ. *spline* – рейка, линейка).

Сплайн – это функция, которая на каждом частичном интервале представляется полиномом некоторой степени, а на всем заданном отрезке непрерывна вместе с несколькими своими производными. На практике широкое применение получили сплайны третьей степени (кубические сплайны).

На интервале $[x_{i-1}, x_i]$ кубический сплайн можно представить в следующем виде

$$s_i(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3, \tag{5}$$

где a_i, b_i, c_i, d_i – коэффициенты сплайнов; $i = 1, 2, \ldots, n$ – номер сплайна (интервала).

Коэффициенты сплайнов определяются из условия сшивания соседних сплайнов в узловых точках:

1) условия Лагранжа

$$s_i(x_i) = y_i, \quad s_i(x_{i-1}) = y_{i-1} \tag{6}$$

2) непрерывность первой и второй производной сплайнов в узлах

$$s_i'(x_i) = s_{i+1}'(x_i), \quad s_i''(x_i) = s_{i+1}''(x_i) \tag{7}$$

Кроме этого необходимо задать дополнительные условия на концах интервала, т.е. в точках x_0 и x_n. Если потребовать нулевой кривизны сплайна в этих точках, то дополнительными условиями будут являться равенства нулю вторых производных сплайнов на концах интервала интерполяции:

$$s_1''(x_0) = 0, \quad s_n''(x_n) = 0 \tag{8}$$

Дополнительные условия могут быть и иными поскольку их выбор, в общем случае, зависит от конкретной задачи.

Подстановка выражения (5) в условия (6), (7) и (8) дает полную систему из $4n$ уравнений относительно коэффициентов сплайнов a_i, b_i, c_i, d_i.

Интерполяция сплайнами имеет очень простую и наглядную физико-механическую аналогию. Если попытаться совместить упругую металлическую линейку с узловыми точками, то форма, которую примет в этом случае линейка будет совпадать с графиком кубического сплайна (сплошная линия на рис.1). Вне узловых точек, где линейка свободна, она описывается уравнением прямой. Соответствующее поведение сплайна обеспечивается условием (8), в связи с чем, его иногда называют "условием свободных концов сплайнов". Если к свободным концам линейки подвесить небольшие грузы, то линейка деформируется (пунктирная линия на рис. 1) и ее поведение вне узловых точек может быть описано, например, уравнением параболы. В этом случае условия (8) будут совершенно иными ("условия нагруженных сплайнов").

Mathcad содержит в себе четыре функции, обеспечивающие интерполяцию кубическими сплайнами:

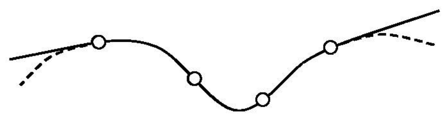


Рис. 1.

lspline(vx, vy)
pspline(vx, vy)
cspline(vx, vy) – функции, возвращающие коэффициенты сплайнов

interp(vs, vx, vy, x) – функция, возвращающая значение сплайна в точке **x** по исходным векторам **vx** и **vy** и по коэффициентам сплайна **vs**.

Функции **lspline**, **pspline** и **cspline** отличаются тем, что в окрестностях границ интервала точек интерполяция (и экстраполяция) осуществляется, соответственно, линейным, параболическим или кубическим полиномом. Внутри интервала точек интерполяция ведется кубическим полиномом во всех трех случаях.

Пример сплайн-интерполяции:

$$X := \begin{pmatrix} 1 \\ 3 \\ 4 \\ 6 \\ 7 \\ 10 \end{pmatrix} \quad Y := \begin{pmatrix} 0.7 \\ 0.6 \\ 1.1 \\ 0.7 \\ 0.8 \\ 1.2 \end{pmatrix}$$

$i := 0..5$

$x := 0.5, 0.6..12$

$C_l := \mathrm{lspline}(X, Y) \qquad S_l(x) := \mathrm{interp}(C_l, X, Y, x)$

$C_p := \mathrm{pspline}(X, Y) \qquad S_p(x) := \mathrm{interp}(C_p, X, Y, x)$

$C_c := \mathrm{cspline}(X, Y) \qquad S_c(x) := \mathrm{interp}(C_c, X, Y, x)$