Лабораторная работа №4

Интерполяция функций

Цель работы: изучение методов решения задачи интерполяции; приобретение навыков программирования методов интерполяции; приобретение навыков использования стандартных средств системы Matlab для проведения интерполирования.

Теоретические сведения

Постановка задачи интерполяции. Пусть на отрезке $a \le x \le b$ задана сетка $\overline{\omega} = \{x_0 = a < x_1 < x_2 < ... < x_n = b\}$ и в ее узлах x_i заданы значения функции $y(x)$, равные

$$y(x_0) = y_0, ..., y(x_i) = y_i, ..., y(x_n) = y_n .$$

Требуется построить интерполянту - функцию $f(x, \mathbf{c})$, совпадающую с функцией $y(x)$ в узлах сетки:

$$f(x_i, \mathbf{c}) = y_i, \quad i = 0,1,...,n, \tag{1}$$

где $\mathbf{c} = \{c_0, ..., c_n\}$ - некоторые неизвестные параметры. Основная цель решения этой задачи состоит в том, чтобы иметь возможность вычисления значений функции $y(x)$ для значений x, не содержащихся в таблице данных. Основным вопросом интерполяции является выбор интерполянты $f(x, \mathbf{c})$ и оценка погрешности интерполяции, т.е. величины $y(x) - f(x, \mathbf{c})$. Математически он заключается в определении неизвестных параметров $\mathbf{c}$ при выборе определенного вида функциональной зависимости.

Рассмотрим линейную зависимость функции $f(x, \mathbf{c})$ от параметров $\mathbf{c}$, т.е. будем считать, что она представима в виде обобщенного многочлена

$$f(x, \mathbf{c}) = \sum_{k=0}^{n} c_k \varphi_k(x) .$$

Для определения неизвестных коэффициентов c_k необходимо решить систему линейных алгебраических уравнений:

$$\sum_{k=0}^{n} c_k \varphi_k(x_i) = y_i, \quad i = 0,1,...,n .$$

Данная задача имеет единственное решение только в том случае, когда определитель системы уравнений $\det[\varphi_k(x_i)]$ отличен от нуля. Система функций, удовлетворяющая такому требованию, называется чебышевской. В качестве $\varphi_k(x)$ чаще всего выбирают: степенные $\varphi_k(x) = x^k$ (тогда $f(x) = P_n(x)$ - полином степени n), тригонометрические $\{\varphi_k(x)\} = \{\cos kx, \sin kx\}$ (в этом случае $f(x)$ - тригонометрический полином), экспоненциальные и другие системы линейно-независимых функций.

Любая непрерывная на отрезке $[a, b]$ функция $f(x)$ может быть хорошо приближена некоторым полиномом $P_n(x)$:

$$P_n(x) = \sum_{k=0}^{n} c_k x^k , \tag{2}$$

где $\mathbf{c} = \{c_0, ..., c_n\}$ - неопределенные коэффициенты. Согласно основному условию интерполяций (1) для нахождения неизвестных коэффициентов $\mathbf{c}$ имеем систему линейных алгебраических уравнений:

$$\begin{aligned} &c_0 + c_1 x_0 + c_2 x_0^2 + ... + c_n x_0^n = y_0, \\ &c_0 + c_1 x_1 + c_2 x_1^2 + ... + c_n x_1^n = y_1, \\ &. \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \\ &c_0 + c_1 x_n + c_2 x_n^2 + ... + c_n x_n^n = y_n. \end{aligned}$$

Данная система имеет единственное решение, так как ее определителем является отличный от нуля определитель Вандермонда. Отсюда следует, что интерполяционный полином (2) существует и единственен.

Интерполяционный многочлен Лагранжа. Будем строить многочлен n-й степени, который исторически обозначается $L_n(x)$ и называется многочленом Лагранжа, в виде линейной комбинации специальных (базисных) многочленов n-й степени $l_i(x)$ при $i = 0, 1, ..., n$:

$$L_n(x) = \sum_{i=0}^{n} c_i l_i(x). \tag{3}$$

Для того чтобы такой многочлен был интерполяционным для функции $y(x)$, потребуем выполнения условий интерполяции $L_n(x_i) = y(x_i) = y_i$. Эти условия будут выполнены, если $c_i = y_i$, а базисные многочлены $l_i(x)$ удовлетворяют условиям

$$l_i(x_j) = \left\{\begin{matrix} 0, \text{ если } j \neq i \\ 1, \text{ если } j = i \end{matrix}\right| \forall i, j \in \{0, 1, ..., n\}. \tag{4}$$

Положим

$$l_i(x) = \frac{(x - x_0)...(x - x_{i-1})(x - x_{i+1})...(x - x_n)}{(x_i - x_0)...(x_i - x_{i-1})(x_i - x_{i+1})...(x_i - x_n)}, \; i = 0, 1, 2, ..., n. \tag{5}$$

Нетрудно проверить, что многочлены $l_i(x)$ удовлетворяют условиям (4). Подставляя (5) в (3), получаем

$$L_n(x) = \sum_{i=0}^{n} \frac{(x - x_0)...(x - x_{i-1})(x - x_{i+1})...(x - x_n)}{(x_i - x_0)...(x_i - x_{i-1})(x_i - x_{i+1})...(x_i - x_n)} y_i \tag{6}$$

или в более компактной форме

$$L_n(x) = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{(x - x_j)}{(x_i - x_j)}.$$

Погрешность интерполяции степенными функциями. Рассмотрим в качестве погрешности величину остаточного члена интерполяции в произвольной точке $x \in [a, b]$:

$$R_n(x) = y(x) - L_n(x).$$

Оценку величины $R_n(x)$ будем проводить в предположении, что функция $f(x)$ имеет $(n + 1)$ непрерывную производную на отрезке $[a, b]$.

Введем многочлен (n + 1)-й степени $\Pi_{n+1}(x)$, определенный через узлы $x_0, x_1, ..., x_n$:

$$\Pi_{n+1}(x) = \prod_{i=0}^{n}(x - x_i) = (x - x_0)(x - x_1)...(x - x_n).$$

Тогда абсолютную погрешность интерполяционной формулы Лагранжа (6) в произвольной точке $\hat{x} \in [a, b]$ можно оценить с помощью неравенства

$$\left|R_n(\hat{x})\right| \le \frac{M}{(n+1)!}\left|\Pi_{n+1}(\hat{x})\right|, \qquad (7)$$

где $M = \max\limits_{x\in[a,b]}\left|f^{(n+1)}(x)\right|$.

Рассмотрим, как ведет себя погрешность интерполяции при увеличении числа узлов. Выражение для погрешности (7) состоит из трех различных частей; факториал и произведение разностей с увеличением n уменьшают ошибку, а порядок производной при этом растет. Для многих функций величина M_{n+1} увеличивается быстрее, чем (n + 1)!. В результате полиномиальные интерполянты редко сходятся к обычной непрерывной функции при $n \to \infty$. Наблюдаемый на практике эффект проявляется в том, что интерполяционный полином высокой степени может вести себя «плохо» в точках, отличных от узлов интерполяции (x_i, y_i), $i = \overline{0,n}$. Поэтому на практике часто используют интерполянты степени не выше 5-6.

Примером может служить функция Рунге вида $r(x) = 1/(1 + 25x^2)$. С увеличением порядка интерполирующего полинома при равномерном распределении узлов интерполяции на интервале [–1, 1] происходит ухудшение качества приближения на краях интервала. Это объясняется тем, что производные $r(x)$, которые фигурируют в выражении для погрешности интерполяции, быстро растут с увеличением числа n. Таким образом, точность приближения зависит не только от числа узлов интерполяции (т.е. порядка интерполирующего полинома), но и от их расположения на интервале [a, b].

Задача о наилучшем выборе узлов интерполирования была решена Чебышевым. Наилучшие узлы интерполирования выбираются равными корням «полинома, наименее отклоняющегося от нуля» на отрезке интерполирования. Полином, наименее отклоняющийся от нуля на отрезке [–1, 1], был найден Чебышевым и назван его именем. Полином Чебышева определяется следующим выражением:

$$T_n(t) = \cos(n \arccos t), \qquad (8)$$

где $n = 0, 1, 2,...$ Эта тригонометрическая функция является многочленом при любом n. При $n = 0$ и $n = 1$ непосредственно из (8) получаем $T_0(t) = 1$, $T_1(t) = t$. Далее, обозначая $\alpha = \cos t$, имеем $T_1(t) = \cos\alpha$, $T_n(t) = \cos n\alpha.$ Так как по правилу сложения косинусов $\cos(n+1)\alpha + \cos(n-1)\alpha = 2\cos\alpha \cos n\alpha$, справедливо равенство

$$T_{n+1}(t) + T_{n-1}(t) = 2T_1(t)T_n(t). \qquad (9)$$

Из (9) следует, что последовательность функций $T_n(t)$, определяемая рекуррентно, представляет собой многочлен степени n: $T_2(t) = 2t^2 - 1$, $T_3(t) = 4t^3 - 3t$, $T_4(t) = 8t^4 - 8t^2 + 1$ и т.д.

Из всех многочленов степени n со старшим коэффициентом 1 нормированный многочлен Чебышева $\hat{T}_n(t) = 2^{-n+1}T_n(t)$ наименее уклоняется от нуля на отрезке [–1, 1]. Последнее фактически означает, что среди всех многочленов степени n вида

$$P_n(t) = t^n + a_1 t^{n-1} + a_2 t^{n-2} + ... + a_n$$

именно нормированный многочлен $\widehat{T}_n(t)$ минимизирует максимальное расстояние от графика многочлена до оси абсцисс при $t \in [-1, 1]$. Таким образом, максимальная погрешность интерполирования будет минимальной, если в качестве многочлена $\Pi_{n+1}(t)$ взять нормированный многочлен Чебышева $\widehat{T}_n(t)$, так как именно он наименее уклоняется от нуля на отрезке $[-1, 1]$. В этом случае точки t_0, t_1,…, t_n будут корнями многочлена Чебышева $T_{n+1}(t)$ или $\widehat{T}_{n+1}(t)$, которые имеют вид

$$t_k = \cos\frac{2k+1}{2n}\pi, \text{ где } k = 0, 1, ..., n-1.$$

Обычно эти узлы называются чебышевскими узлами интерполяции.

На интервале [a, b] эти корни можно представить по формуле Чебышева

$$x_{i+1} = \frac{a+b}{2} + \frac{b-a}{2}\cos\left(\frac{(2i+1)\pi}{2n+2}\right),\ i = \overline{0, n}.$$

Интерполяционная функция с использованием именно этих точек в качестве узлов дает наилучшее приближение к интерполируемой функции.

Средства Matlab для проведения интерполяции. Для интерполирования функций в **Matlab** можно использовать операторы **polyfit** и **polyval**.

Оператор **polyfit**(x,y,n) находит массив коэффициентов a длиной (n + 1) полинома степени n по массивам длиной m узлов x и значений функции y в узлах x, m ≥ n + 1. Этот полином аппроксимирует функцию y(x) по методу наименьших квадратов. Предполагается, что полином задается в виде

$$y(x) = a_1 x^n + a_2 x^{n-1} + ... + a_n x + 1.$$

Если $m = n + 1$, то программа возвращает коэффициенты интерполяционного полинома.

Оператор **polyval**(a,x) возвращает значение полинома в точке x, коэффициенты которого определены в векторе **a**.

Кроме этого, существуют операторы, позволяющие проводить интерполяцию одно- и многомерных данных:

- **interpft**(y,n,dim) **-** аппроксимация периодической функции на основе быстрого преобразования Фурье (y **-** одномерный массив значений функции; n **-** число узлов в массиве значений);

- **spline**(x,y,z) **-** интерполяция y=y(x) кубическим сплайном и вывод соответствующих значений в точках z. Для получения большей информации используется конструкция pp=**spline**(x,y), здесь командой v=**ppval**(pp,z) можно найти значения в точках z, а командой [xs,Coef,m,L]=**unmkpp**(pp) получить данные о векторе разбиений аргумента xs, коэффициентах Coef, m=**length**(xs), L=**length**(Coef)/m;

- **interp1**(x,y,z), **interp1**(x,y,z,’method’) **-** одномерная табличная интерполяция (если y **-** двумерный массив, интерполяция ведется по каждому столбцу; значения z должны входить в диапазон значений x). Можно указать метод интерполяции **-** кусочно-линейной (linear по умолчанию), ступенчатой (nearest), кубической (cubic), кубическими сплайнами (spline). Функция **interp1q**(x,y,z) реализует быструю линейную интерполяцию на неравномерной сетке;

- **interp2**(x1,x2,y,z1,z2), **interp1**(x1,x2,y,z1,z2,’method’) **-** двумерная табличная интерполяция y=y(x1,x2), аргументы должны меняться монотонно и быть заданы в формате функции meshgrid;
- **interp3**(x1,x2,x3,y,z1,z2,z3), **interp3**(...,’method’) **-** трехмерная табличная интерполяция y=y(x1,x2,x3);
- **interpn**(x1,x2,...,y,z1,z2,...), **interp3**(...,’method’) **-** многомерная табличная интерполяция y=y(x1,x2,...);
- **griddata**(x1,x2,y,z1,z2), **griddata**(x1,x2,y,z1,z2,’method’) **-** двумерная табличная интерполяция на неравномерной сетке.

Для реализации программы полезными являются следующие операторы:

- **sum**(x) и **prod**(x) **-** суммирование и произведение элементов вектора (для двумерного массива выполняется поиск сумм и произведений по столбцам);
- **diff**(x), **diff**(x,n), **diff**(x,n,dim) **-** вычисление конечных разностей (первых, n-го порядка или по указанному измерению); если x **-** массив, берутся разности между столбцами:

Порядок выполнения работы

Написать соответствующие m-файлы сценарии и m-файлы функции для реализации следующих задач.

1. Провести интерполяцию функции Рунге на отрезке [–1, 1] по формуле Лагранжа для n = 11 при равномерном распределении узлов интерполяции.
2. Провести интерполяцию функции Рунге на отрезке [–1, 1] по формуле Лагранжа для n = 11 для чебышевских узлов.
3. Построить графики функции Рунге и ее интерполянт не менее чем в 100 узлах. Сравнить результаты.
4. Выбрать функцию согласно номеру компьютера и провести ее интерполяцию по формуле Лагранжа при равномерном распределении узлов на заданном интервале для n = 11 и n = 6.
5. Провести интерполяцию по тем же узлам, используя стандартные функции Matlab.
6. Построить графики исходной функции и интерполянт не менее чем в 100 узлах. Сравнить результаты.
7. Вычислить и построить графики функций ошибок интерполяции.

Содержание отчета

Отчет должен содержать:

1) цель работы, задание;
2) описание метода решения, краткие сведения из теории (формулы, алгоритм и т.п.);
3) программу (распечатку), ее описание;
4) сравнение результатов расчета;
5) краткие выводы.

Задание

Номер варианта	$y(x)$	a	b
1	$\sqrt{(x-2)^3(5-x)}$	2	5
2	$(1+x)(x^2+3)^{-1}$	–9	9
3	$10e^{-x}(x^3-2x+1)$	–2	2
4	$\left(1+x^2\right)\sin 2x$	–1	5
5	$\exp(-0,5x\cos x)$	–5	3
6	$\exp(2,5\sin x)\cos x$	1	3
7	$x^2\sin(2x-3)$	0	4
8	$x(x^2+3)^{-1}$	–5	5
9	$(1+x^2)(1+x^3)^{-1}$	0	4
10	$(x-3)\cos^2 x$	–3	3
11	$(1-x)\cos x$	0	4
12	$\sqrt{x}-\cos(1,5x)$	0	8
13	$\sin 2x-0,2x^2$	0	6
14	$(x^2+1)^{-1}\sin(x+1)$	–3	2
15	$0,5x^2+8x^{-1}+8$	–4	–1
16	$x^2-2x+16(x-1)^{-1}-13$	2	5
17	$2x(2x+3)(x^2+4x+5)^{-1}$	–2	2
18	$(1+x)\sqrt{8-x}$	–3	3
19	$(x^2+7x+7)(x^2-2x+2)^{-1}$	2	5
20	$(2x^2+6)(x^2-2x+5)^{-1}$	–3	3
21	$x(2x^2+1)^{-1}$	–1	4
22	$(1+x)\sin x$	0	5
23	$\exp(-x\sin 2x)$	–2	2
24	$(1-x)(x^2+4)^{-1}$	–2	3
25	$\sqrt{x}\cos 2x$	0	4
26	$x^2\cos(x+1)$	–3	3
27	$x+3\cos^2 x$	–3	3
28	$x^2+2(x+0,5)^{-1}$	0	4
29	$(x^2+1)^{-1}\cos x$	–2	3
30	$(1+x^3)e^{-x}$	–2	2

Контрольные вопросы

1. Сформулируйте постановку задачи интерполяции.
2. Сформулируйте постановку задачи полиномиальной интерполяции со степенным базисом.
3. Выведите интерполяционную формулу Лагранжа. Приведите примеры для интерполяции по 2 и 3 точкам.
4. Приведите оценку погрешности интерполяционной формулы Лагранжа.
5. Дайте определение полиномов Чебышева. Выведите рекуррентное соотношение для их вычисления.
6. Как осуществляется интерполяция по чебышевским узлам?
7. Дайте определение интерполяционного полинома Ньютона. Укажите условия практического применения многочленов Ньютона и Лагранжа.
8. Дайте определение разделенных и конечных разностей. Укажите их свойства.
9. Постройте таблицу разделенных разностей для пяти точек.
10. Поставьте задачу кусочно-полиномиальной интерполяции.

Литература

Гончаров В.А., Земсков В.Н., Яковлев В.Б. Лабораторный практикум по курсу «Вычислительная математика». – М.: МИЭТ, 2008.