Министерство образования, науки и молодежной политики
Краснодарского края
Государственное бюджетное профессиональное образовательное учреждение
Краснодарского края «Ейский полипрофильный колледж»

ЛАБОРАТОРНАЯ РАБОТА №3

по теме:

Интерполирование функций с помощью полинома Лагранжа

Выполнил:
студент ЕПК, группа И-22
Синицын М.В.

Проверил:
преподаватель численных
методов
Фомин А. Т.

Ейск 2023

1 Постановка задачи

Существует прибор, который в дискретные моменты времени выдаёт сигнал по закону f(t). Наблюдатель зарегистрировал 4 отсчёта в моменты времени t_i (i = 0 .. 3). Задачей наблюдателя (который не знает закона выдачи сигнала) является получение приближённого значения функции на отрезке [0; 1] в любой момент времени.

1. Описать метод аппроксимации Лагранжа;
2. Выполнить интерполирование функции f(t) с помощью полинома Лагранжа;
3. Построить в GNU Octave графики функций f(t) и L(t) в одной системе координат;
4. Определить значение функции L(t) в любой точке на отрезке [0; 1];
5. Определить погрешность результата вычислений.
6. Сделать вывод

Вариант 21

$$f(t)=\frac{e^t}{3} \tag{1}$$

Таблица 1. Узлы интерполяции

i	0	1	2	3
x_i	0,25	0,375	0,5	0,625
y_i	0,428	0,485	0,549	0,623

2 Интерполяция

2.1 Описание полинома Лагранжа

Интерполяцио́нный многочле́н Лагра́нжа — многочлен минимальной степени, принимающий заданные значения в заданном наборе точек, то есть решающий задачу интерполяции.

Ж. Л. Лагранж предложил следующий способ вычисления таких многочленов:

$$L_n(x)=\sum_{i=0}^{n} y_i \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} \tag{2}$$

Полиномы Лагранжа используются для интерполяции, а также для численного интегрирования.

Многочлен в форме Лагранжа в явном виде содержит значения функций в узлах интерполяции, поэтому он удобен, когда значения функций меняются, а узлы интерполяции неизменны. Число арифметических операции, необходимых для построения многочлена Лагранжа, пропорционально n^2 и является наименьшим для всех форм записи. К недостаткам этой формы записи можно отнести то, что при построении полинома степени n+1 полностью теряется информация о предыдущем полиноме степени n, т.е. с изменением числа узлов приходится все вычисление выполнить заново.

2.1 Интерполяция функции варианта

Выполнил интерполяцию задананной функции(1)

$$\begin{gathered}
L_3(x)=0{,}428\frac{(x-0{,}375)(x-0{,}5)(x-0{,}625)}{(0{,}25-0{,}375)(0{,}25-0{,}5)(0{,}25-0{,}625)}+\\
+0{,}485\frac{(x-0{,}25)(x-0{,}5)(x-0{,}625)}{(0{,}375-0{,}25)(0{,}375-0{,}5)(0{,}375-0{,}625)}-\\
+0{,}549\frac{(x-0{,}25)(x-0{,}375)(x-0{,}625)}{(0{,}5-0{,}25)(0{,}5-0{,}375)(0{,}5-0{,}625)}+\\
+0{,}623\frac{(x-0{,}25)(x-0{,}375)(x-0{,}5)}{(0{,}625-0{,}25)(0{,}625-0{,}375)(0{,}625-0{,}5)}=\\
0{,}256\,x^3-0{,}064\,x^2+0{,}42\,x+0{,}323.
\end{gathered} \tag{3}$$

2.2 Построение графиков функций в GNU Octave

Листинг 1. Построение графиков

```
x = -4:0.1:6.5;
y1 = exp(x)./3;
y2 = 0.323 + 0.42 .*x - 0.064 .*x.^2 + 0.256 .*x.^3
graf = plot(x, y1, 'r', x, y2, 'b--');
lg = legend('x sin x', 'polinom');
h  = title('Interpolation 21');
```

```
ax = gca();
set(ax, 'xlim', [-4 4], 'ylim', [-4 4]);
set(ax, 'xtick', [-4:0.5:4], 'ytick', [-4:0.5:4]);
set(ax, 'color', 'w');
set(ax, 'xcolor', 'k', 'ycolor', 'k');
set(ax, 'xgrid', 'on', 'ygrid', 'on');
set(ax, 'gridcolor', [1 1 1]);
set(ax, 'gridlinestyle', '-');
set(ax, 'gridalpha', 0.15);
set(ax, 'fontsize', 12);
set(ax, 'xlabel', 'x', 'ylabel', 'f(x)');
 set(h, 'fontsize', 14);
set(graf, 'LineWidth', 2);
set(lg, 'edgecolor', 'k')
set(lg, 'fontsize', 9, 'fontweight', 'bold');
```

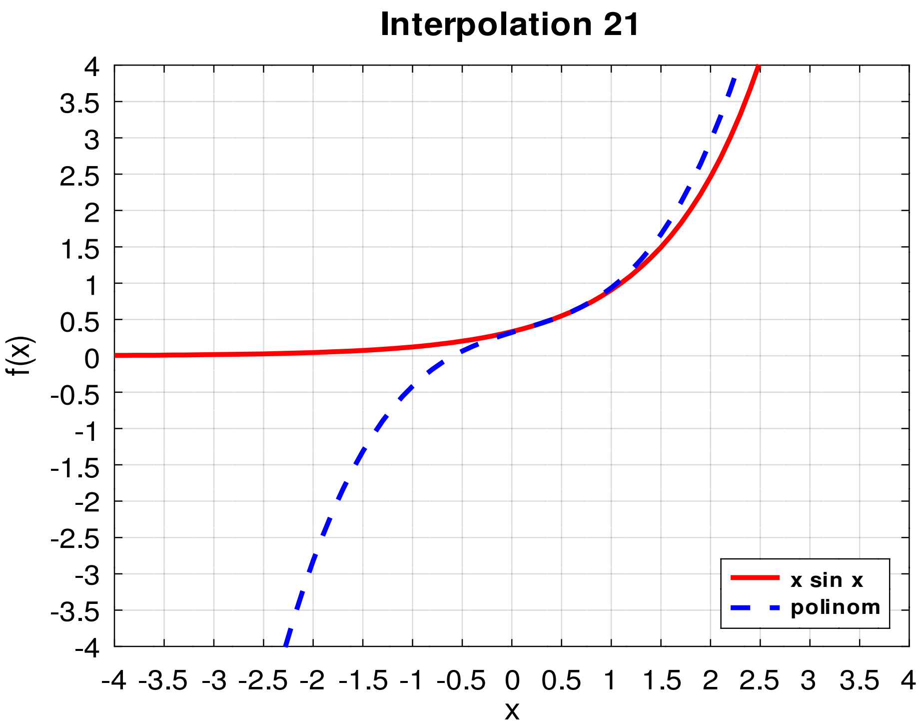

Рис. 1. Графическое представление интерполяции

2.3 Интерполирование

В качестве точки, для выполнения интерполирования я выбрал точку x = 0,4. на промежутке [0;1]

Получил:

$$\begin{aligned} L_3(0,4)={}&0,428\frac{(0,4-0,375)(0,4-0,5)(0,4-0,625)}{(0,25-0,375)(0,25-0,5)(0,25-0,625)}+\\ &+0,485\frac{(0,4-0,25)(0,4-0,5)(0,4-0,625)}{(0,375-0,25)(0,375-0,5)(0,375-0,625)}-\\ &+0,549\frac{(0,4-0,25)(0,4-0,375)(0,4-0,625)}{(0,5-0,25)(0,5-0,375)(0,5-0,625)}+\\ &+0,623\frac{(0,4-0,25)(0,4-0,375)(0,4-0,5)}{(0,625-0,25)(0,625-0,375)(0,625-0,5)}=0,497144. \end{aligned} \tag{4}$$

Абсолютная погрешность составляет:

$|f(0,4)-L_3(0,4)|=|0,49727-0,497144|=0,000125$

2.4 Проверка вычислений с помощью программы на C#

Листинг 2. Алгоритм интерполирования функции

```
namespace ConsoleApp
{
class Program
{
static void Main()
{
Console.Write("Задайте степень полинома\nn = ");
int size = Convert.ToInt32(Console.ReadLine());
size++;
var xValues = new double[size];
var yValues = new double[size];
Console.WriteLine($"x[i]:");
for (int i = 0; i < size; i++)
{
xValues[i] = Convert.ToSingle(Console.ReadLine());;
yValues[i] = TestF(xValues[i]);
}
Console.Write("Задайте значение для интерполирования: ");
double x = Convert.ToDouble(Console.ReadLine());
```

```
Console.WriteLine(InterpolateLagrangePolynomial(x, xValues,
yValues, size));
}
static double InterpolateLagrangePolynomial (double x, double[]
xValues, double[] yValues, int size)
{
double lagrangePol = 0;
for (int i = 0; i < size; i++)
{
double basicsPol = 1;
for (int j = 0; j < size; j++)
{
if (j != i)
{
basicsPol *= (x - xValues[j])/(xValues[i] - xValues[j]);
}
}
lagrangePol += basicsPol * yValues[i];
}
return lagrangePol;
}
static double TestF(double x)
{
return 0.256*Math.Pow(x, 3)- 0.064*Math.Pow(x, 2) + 0.42*x
+0.323;
}
}
}
```

Вывод:

```
n = 3
0,497144
```

Вывод

В ходе ходе данной работы мы выполнили апроксимацию функции и получили из исходной функции интерполяционный поленом Лагранжа третьей степени, построив графики исходной функции и поленома мы получили пересечение в узловых точках. Отсюда следует, что апроксимация функции была выполнена верно и полученный поленом решает задачу интерполяции.