БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Механико-математический факультет

Кафедра веб-технологий и компьютерного моделирования

СМОЛЕНСКИЙ ЗАХАР ВАСИЛЬЕВИЧ

ИНТЕРПОЛИРОВАНИЕ ФУНКЦИЙ АЛГЕБРАИЧЕСКИМИ СПЛАЙНАМИ

Курсовая работа, 2 курс

Руководитель
доцент, канд. физ.-мат. наук
ИГНАТЕНКО М. В.

Минск
2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

При решении задач с научными и инженерными расчётами часто приходится оперировать наборами значений, полученных опытным путём или методом случайной выборки. Как правило, на основании этих наборов требуется построить функцию, на которую могли бы с высокой точностью попадать другие получаемые значения. Такая задача называется приближением функций. Интерполяцией называют такую разновидность приближения функций, при которой кривая построенной функции проходит точно через имеющиеся точки данных.

Построение интерполяционного многочлена Лагранжа и Ньютона с использованием большого числа узлов интерполирования на отрезке $[a, b]$ может привести к плохому приближению интерполируемой функции из-за возрастания вычислительной погрешности. Кроме того, построенный многочлен будет иметь высокую степень, что тоже весьма нежелательно. Этих неприятностей можно избежать, разбив отрезок $[a, b]$ на частичные отрезки и построив на каждом из них многочлен невысокой степени, так или иначе приближенный к заданной функции $f(x)$.

Одна из возможностей преодоления этого недостатка заключается в применении сплайн-интерполяции.

Данная курсовая работа посвящена алгебраическому сплайн-интерполированию функций и состоит из двух глав. В первой главе, которая носит реферативный характер, формулируется задача интерполирования алгебраическими сплайнами, рассматриваются ее частные случаи и описывается механизм работы встроенной функции *Spline* компьютерной алгебры *Mathematica*.

Во второй главе представлены результаты самостоятельных исследований: приведены примеры построения интерполяционных сплайнов, выполненных как в среде *Mathematica*, так и аналитически. В заключении изложены выводы, касающиеся результатов самостоятельных исследований.

ГЛАВА 1. ЗАДАЧА АЛГЕБРАИЧЕСКОГО СПЛАЙН–ИНТЕРПОЛИРОВАНИЯ ФУНКЦИЙ

1.1 Постановка задачи

Пусть множество точек $\Delta_N = \{a = \overline{x}_0 < \overline{x}_1 < \ldots < \overline{x}_N = b\}$ – разбиение отрезка $[a,\ b] \subset R$. Функция $S_{m,\tilde{k}}(x)$ называется [1–4] *сплайном* степени m с *дефектом* $\tilde{k}$ и узлами Δ_N, если:

1) на каждом частичном отрезке $[\overline{x}_k,\ \overline{x}_{k+1}]$, $k = \overline{0, N-1}$, функция $S_{m,\tilde{k}}(x)$ является многочленом степени не выше m, и эта степень равна m хотя бы на одном из отрезков;

2) $S_{m,\tilde{k}}(x) \in C^{(m-\tilde{k})}[a,\ b]$, при этом $S_{m,\tilde{k}}^{(m-\tilde{k}+1)}(x)$ не является непрерывной функцией на отрезке $[a,\ b]$.

В определении сплайна число $\tilde{k}$ может принимать значения 1, 2, …, $m+1$. В случае $m - \tilde{k} = -1$ под $C^{(-1)}[a,\ b]$ понимается линейное пространство кусочно-непрерывных функций с точками разрыва первого рода. Заметим, что число точек разбиения $N+1$ не связано со степенью сплайна m.

Если при решении задачи приближения функции $f(x)$, заданной на отрезке $[a,\ b]$, сплайном $S_{m,\tilde{k}}(x)$ потребовать выполнения условия

3) $S_{m,\tilde{k}}(x_k) = f(x_k)$, $x_k \in [\overline{x}_k,\ \overline{x}_{k+1}]$, $k = \overline{0, N-1}$, то такой *сплайн* называется *интерполяционным* для функции $f(x)$.

Для единственности сплайна $S_{m,\tilde{k}}(x)$ следует задать $(m-\tilde{k}) + n(\tilde{k}-1)$ условий, которые обычно задаются на концах отрезка $[a,\ b]$. Вопрос о существовании и единственности сплайна решается в каждом конкретном случае отдельно.

1.2 Частные случаи интерполирования сплайнами

1.2.1 Интерполяционные сплайны произвольной фиксированной степени и соответствующего дефекта

Пусть $m,\ \tilde{k}$ – заданные натуральные числа. Тогда для интерполяционного сплайна $S_{m,\tilde{k}}(x)$, проходящего через точки $\left(x_k, f(x_k)\right)$, $k=\overline{0,n}$, используем представление

$$S_{m,\tilde{k}}(x)=\left\{P_k(x)=f(x_k)+(x-x_k)Q_k(x),\quad x\in[x_k,\ x_{k+1}],\quad k=\overline{0,n-1},\right.$$

где $Q_k(x)$ – многочлен степени $m-1$ с m неизвестными коэффициентами.

Такая схема построения интерполяционного сплайна $S_{m,\tilde{k}}(x)$ обеспечивает выполнение условий 1) и 3) в его определении. Для выполнения условия 2) потребуем, чтобы $P_k^{(i)}(x_{k+1})=P_{k+1}^{(i)}(x_{k+1})$, $i=0,\ 1,\ \ldots,\ m-\tilde{k}$.

Как было отмечено выше, для единственности сплайна $S_{m,\tilde{k}}(x)$ необходимо задать $\left(m-\tilde{k}\right)+n\left(\tilde{k}-1\right)$ дополнительных условий. В результате получим m уравнений для нахождения m неизвестных коэффициентов многочлена $Q_k(x)$.

1.2.2 Линейные интерполяционные сплайны первого дефекта

Пусть степень $m=1$, дефект $\tilde{k}=1$. Требуется построить сплайн $S_{1,1}(x)$ – ломаную с вершинами в заданных точках $\left(x_k, f(x_k)\right)$, $k=\overline{0,n}$ (рис. 1).

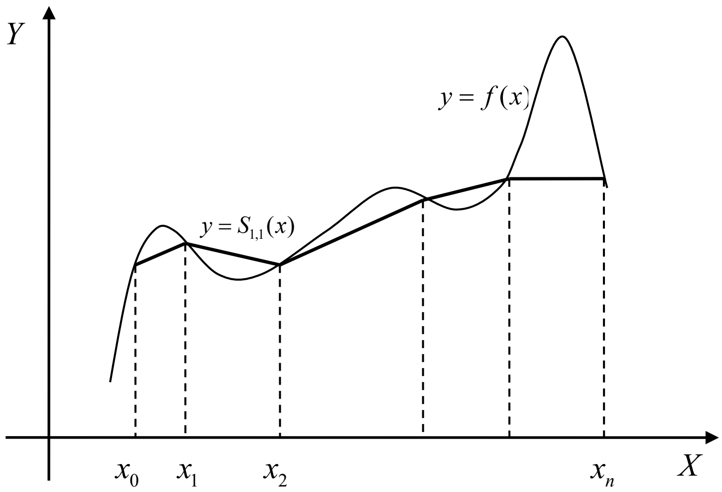


Рис. 1. Пример линейного сплайна

По условию 1) и 3) в определении интерполяционного сплайна можем записать

$$S_{1,1}(x) = \left\{ P_k(x) = f(x_k) + (x - x_k) A_k, \quad x \in [x_k,\, x_{k+1}], \quad k = \overline{0, n-1}, \right.$$

где A_k – многочлен $(m-1)$ степени или нулевой. По условию 2) определения сплайна

$$P_k(x_{k+1}) = P_{k+1}(x_{k+1}) \equiv f(x_{k+1}),$$

т. е. $f(x_k) + (x_{k+1} - x_k) A_k = f(x_{k+1})$, следовательно, $A_k = \dfrac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}$, а искомый сплайн имеет вид

$$S_{1,1}(x) = \left\{ f(x_k) + (x - x_k) \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}, \quad x \in [x_k,\, x_{k+1}], \quad k = \overline{0, n-1}. \right.$$

Построенный сплайн определен однозначно, он является многочленом первой степени на любом из частичных отрезков, представляет собой непрерывную функцию и удовлетворяет интерполяционным условиям [3]. Графиком сплайна $S_{1,1}(f,\, x)$ является ломаная.

Совпадение дефекта сплайна с его степенью обеспечивает просто непрерывность сплайна. Интерес представляет построение сплайнов с большей гладкостью, т.е. с малым дефектом. Такие сплайны являют собой дальнейшее совершенствование идеи кусочно-полиномиальной аппроксимации. Рассмотрим их в следующих пунктах.

1.2.3 Кубические интерполяционные сплайны дефекта 2 (эрмитовы кубические сплайны)

Одним из наиболее распространенных интерполяционных сплайнов является *кубический сплайн.* Рассмотрим эрмитовы кубические сплайны. Для этого дополним данные из пункта 1.2.1 значениями производной функции в узлах сетки $f_i' = f'(x_i)$, $i = \overline{0, N}$

Кубическим интерполяционным сплайном дефекта 2 (*эрмитовым кубическим сплайном*) будем называть функцию $S_{3,2}(f, x) = S_{3,2}(x)$, удовлетворяющую следующим условиям:

1) $S_{3,2}(x) = a_{i0} + a_{i1}(x - x_i) + a_{i2}(x - x_i)^2 + a_{i3}(x - x_i)^3$, $\forall x \in [x_i;\ x_{i+1}]$;
2) $S_{3,2}(x_i) = f_i$; $S'_{3,2}(x_i) = f_i'$, $i = \overline{0, N}$;
3) $S_{3,2}(x) \in C^1[a;\ b]$.

Вторая производная эрмитова кубического сплайна, вообще говоря, разрывна в узлах сетки Δ. Учитывая условия интерполяции для каждого из коэффициентов сплайна $a_{i\alpha}$, $\alpha = \overline{0,3}$, $i = \overline{0, N-1}$, при каждом i имеем следующую систему уравнений:

$$S_{3,2}(x_i) = f_i;\ S'_{32}(x_i) = f_i';\ S_{3,2}(x_{i+1}) = f_{i+1};\ S'_{3,2}(x_{i+1}) = f'_{i+1}.$$

Решив эту систему на $[x_i, x_{i+1}]$ получаем

$$S_{3,2}(x) = \varphi_1(t) f_i + \varphi_2(t) f_{i+1} + \varphi_3(t) h_i f_i' + \varphi_4(t) h_i f'_{i+1},$$

причем

$$h_i = x_{i+1} - x_i,\ t = \frac{x - x_i}{h_i},$$

$$\varphi_1(t) = (1-t)^2(1+2t);\quad \varphi_2(t) = t^2(3-2t);$$

$$\varphi_3(t) = t(1-t)^2;\quad \varphi_4(t) = -t^2(1-t).$$

Данная формула удобна для теоретических исследований. Для практического вычисления сплайна в $\forall x \in [x_i;\ x_{i+1}]$ более выгодны с точки зрения количества арифметических операций следующие формулы:

$$S_{3,2}(x) = f_i + (x - x_i)\left[f_i' + t(B + tA)\right]$$

где

$$A = -2(f_{i+1} - f_i)/h_i + (f_i' + f_{i+1}'),\quad B = -A + (f_{i+1} - f_i)/h_i - f_i'.$$

Для вычисления сплайна в одной точке достаточно выполнить 16 арифметических операций.

Данный алгоритм вычисления сплайна устойчив к погрешности исходных данных, но при вычислении производных эрмитова сплайна не следует использовать слишком густые сетки.

1.2.4 Интерполяционные кубические сплайны третьей степени первого дефекта (сплайны Шенберга)

Рассмотрим [5] наиболее известный и применяемый интерполяционный кубический сплайн дефекта 1.

Пусть на отрезке $[a;\ b]$ в узлах сетки Δ заданы значения некоторой функции $f_i = f(x_i),\ i = \overline{0, N}.$

Интерполяционным кубическим сплайном $S_{3,1}(f,\ x)$ называется сплайн, удовлетворяющий условиям

$$S_{3,1}(f,\ x_i) = f_i,\ i = \overline{0, N}. \tag{1.1}$$

Для построения сплайна $S_{3,1}(f,\ x)$ на отрезке $[a;\ b]$ необходимо определить $4N$ коэффициентов. Условие $S_{3,1}(f,\ x) \in C^2[a;\ b]$ эквивалентно требованию непрерывности сплайна и его производных $S_{3,1}^{(r)}(f,\ x)$, $r = 0,\ 1,\ 2$ во всех внутренних узлах x_i, $i = \overline{1, N-1}$ сетки Δ, что дает $3(N–1)$ равенств. Таким образом, вместе с равенствами (1.1) получается $4N–2$ соотношений. Два дополнительных условия обычно задаются в виде ограничений на значения сплайна и его производных на концах отрезка $[a;\ b]$ (или вблизи концов) и называются краевыми условиями (граничными), из которых наиболее употребительными являются следующие типы:

I. $S'_{3,1}(f,\ a) = f'(a);\ S'_{3,1}(f,\ b) = f'(b);$

II. $S''_{3,1}(f,\ a) = f''(a);\ S''_{3,1}(f,\ b) = f''(b);$

III. $S_{3,1}^{(r)}(f,\ a) = S_{3,1}^{(r)}(f,\ b),\ r = 1,\ 2;$

IV. $S'''_{3,1}(f,\ x_p + 0) = S'''_{3,1}(f,\ x_p - 0),\ p = 1,\ N-1.$

Интерполяционный кубический сплайн $S_{3,1}(f,\ x)$, удовлетворяющий условиям (1.1) и одному из типов краевых условий I–IV, существует и единственен.

Рассмотрим набор граничных условий и выясним какие из них более предпочтительны и в каких случаях.

Выбор краевых условий осуществляется в зависимости от того, какими данными мы располагаем об интерполируемой функции $f(x)$.

1. Если $f(x)$ – периодическая функция, то следует использовать периодические краевые условия (тип III).

2. Когда известны значения $f'(x)$ или $f''(x)$ в точках a и b, то естественно воспользоваться краевыми условиями типа I или II. При этом, если существует возможность выбора между ними, то предпочтительнее условия типа I.

3. Наибольшие трудности возникают, когда заданы только узловые значения f_i. Рассмотрим некоторые из известных рекомендаций.

а) Часто предлагается использовать естественные краевые условия: $S''_{3,1}(a) = S''_{3,1}(b) = 0$. Однако такое решение весьма неудачно. В данном случае точность приближения кубическим сплайном снижается до точности приближения $S_{1,1}(f, x)$.

б) При другом подходе используются краевые условия типа I или II, но необходимые значения производных заменяются подходящими разностными аппроксимациями. Недостаток этого в том, что приходится применять односторонние аппроксимации, которые, как известно, имеют невысокую точность.

в) В большинстве случаев кардинальным решением вопроса является применение краевых условий типа IV. Этот вывод подтверждает и практический опыт расчетов.

1.3 Функция Spline компьютерной алгебры Wolfram Mathematica

В компьютерной алгебре *Wolfram Mathematica* имеется встроенная функция *Spline*, которая позволяет строить график и получать значение функции сплайна путем введения некоторых параметров. Рассмотрим [6, 7] эту функцию и ее варианты подробнее.

Сама функция *Spline* является частью пакета *Splines* программы *Mathematica*. Она имеет следующий вид:

Spline[{*pt1*,*pt2*,…}, *type*].

Функция принимает множество точек {*pt1, pt2*, …} и тип сплайна *type* в качестве параметра, а на выходе возвращает графический примитив, который представляет сплайн заданного типа, проходящий через заданные точки.

Возможные следующие настройки для *type*:

1) кубический (Cubic) – сплайн степени 3 и дефекта 1 $S_{31}\left(f, x\right)$;

2) сплайны Безье (Bezier) – тип сплайнов, который интерполирует только конечные точки, остальные контролируют форму сплайна создавая выпуклую оболочку;

3) составной сплайн Безье (CompositeBezier) – сплайн, созданный наборами кривых Безье третьего порядка с непрерывной первой производной. Он интерполирует нечетные точки, а остальные служат в качестве параметров. Если количество точек четное, то конечная и предпоследняя вершина меняются местами.

В частности, кубический сплайн состоит из кусочно-непрерывных многочленов третьего порядка. Отметим, что в случае использования настройки *Cubic*, по умолчанию вторая производная сплайна в конечных точках имеет значение 0.

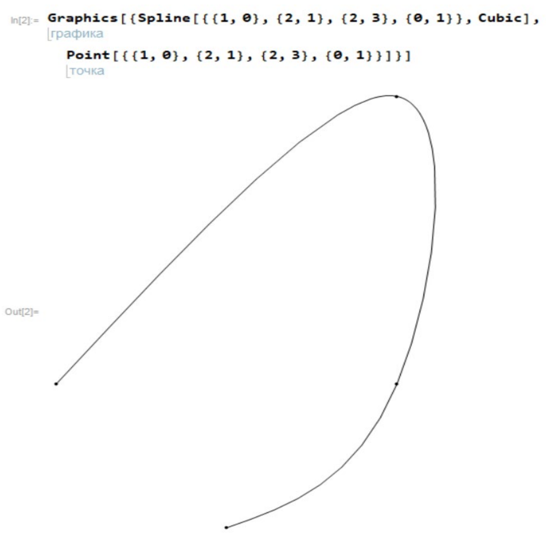


Рис. 2. Кубический сплайн по четырем точкам

На рисунке 2 представлен пример построения графика кубического сплайна по четырем точкам и условием по умолчанию, а в Приложении Д – сплайнов Безье для единичной окружности с шагом $\frac{\pi}{4}$ с помощью операторов *Graphics* и *Spline*.

ГЛАВА 2. ПОСТРОЕНИЕ АЛГЕБРАИЧЕСКИХ ИНТЕРПОЛЯЦИОННЫХ СПЛАЙНОВ

В данной главе будут рассмотрены практические примеры построения различных сплайнов для некоторой функции как аналитически, так и с помощью компьютерной алгебры *Mathematica*.

2.1 Линейные интерполяционные сплайны первого дефекта

Пример 2.1. Пусть функция $f(x)$ задана следующей таблицей.

k	x_k	$f(x_k)$
0	0	1,5
1	2	2,3
2	4	3,4

Построим интерполяционный сплайн $S_{1,1}(x)$ первой степени с дефектом 1 для функции $f(x)$. На каждом из отрезков $[0,\ 2]$ и $[2,\ 4]$ искомый сплайн имеет вид многочлена первой степени $P_k(x) = f(x_k) + \dfrac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}(x - x_k)$ для $k = 0,1$ соответственно, т. е.

$$S_{1,1}(x) = \begin{cases} P_0(x) = 1,5 + 0,4x,\ x \in [0,\ 2]; \\ P_1(x) = 1,2 + 0,55x,\ x \in [2,\ 4]. \end{cases}$$

Для полученного сплайна $S_{1,1}(x)$ выполнены интерполяционные условия:

$$S_{1,1}(0) = P_0(0) = 1,5 = f(0),$$

$$S_{1,1}(2) = P_0(2) = P_1(2) = 2,3 = f(2),\quad S_{1,1}(4) = P_1(4) = 3,4 = f(4).$$

Пример 2.2. Пусть функция $f(x)$ задана таблицей ниже ($k = 0,1,2,3$).

x_k	0,1	0,3	0,7	1,4
$f(x_k)$	1	-7	15	25

Программная реализация задачи построения интерполяционного сплайна $S_{1,1}(x)$ первой степени с дефектом 1 для данной функции выполнена в среде *Mathematica* и представлена в Приложении А. На каждом из частичных отрезков искомый сплайн имеет вид многочлена первой степени. Для полученного сплайна $S_{1,1}(x)$ выполнены интерполяционные условия. График сплайна $S_{1,1}(x)$ представляет собой ломаную линию, проходящую через заданные точки.

2.2 Квадратичные интерполяционные сплайны первого дефекта

Пример 2.3. Построим интерполяционный сплайн $S_{2,1}(x)$ второй степени с дефектом 1, который удовлетворяет дополнительному условию $S'_{2,1}(0) = 0$, для функции $f(x)$, заданной следующей таблицей.

k	x_k	$f(x_k)$
0	0	0
1	1	1
2	2	5

Для сплайна $S_{2,1}(x)$ используем представление

$$S_{2,1}(x) = \begin{cases} P_0(x) = f(x_0) + (x - x_0)(A_0 x + B_0) = 0 + x(A_0 x + B_0), \; x \in [0,\, 1]; \\ P_1(x) = f(x_1) + (x - x_1)(A_1 x + B_1) = 1 + (x - 1)(A_1 x + B_0 x), \; x \in [1,\, 2] \end{cases}$$

с неизвестными коэффициентами A_0, A_1, B_0 и B_1.

Так как дополнительное условие задано в точке x_0 из отрезка $[0, 1]$, то оно равносильно условию $P_0'(x_0)=0$. Учитывая, что $P_0'(x)=2A_0x+B_0$, а $P_1'(x)=2A_1x+B_1-A_1$, запишем систему из четырех уравнений для определения неизвестных коэффициентов:

$$\begin{cases} P_0'(x_0)=0, \\ P_0(x_1)=P_1(x_1), \\ P_0'(x_1)=P_1'(x_1), \\ P_1(x_2)=f(x_2). \end{cases} \text{Следовательно, } \begin{cases} B_0=0, \\ A_0=1, \\ A_1+B_1=2, \\ 1+2A_1+B_1=5, \end{cases} \text{т. е. } \begin{cases} B_0=0, \\ A_0=1, \\ A_1=2, \\ B_1=0. \end{cases}$$

Таким образом, интерполяционный сплайн $S_{2,1}(x)$ для заданной функции $f(x)$ имеет вид

$$S_{2,1}(x)=\begin{cases} P_0(x)=x^2,\ x\in[0, 1]; \\ P_1(x)=2x^2-2x+1,\ x\in[1, 2]. \end{cases}$$

Пример 2.4. Пусть функция $f(x)$ задана таблицей из примера 2.2. Программная реализация задачи построения интерполяционного сплайна $S_{2,1}(x)$ второй степени с дефектом 1, который удовлетворяет дополнительному условию $S_{2,1}'(a)=0$, для данной функции выполнена в среде *Mathematica* и представлена в Приложении Б. На каждом из частичных отрезков искомый сплайн имеет вид многочлена второй степени. Для полученного сплайна $S_{2,1}(x)$ выполнены интерполяционные условия. График сплайна $S_{2,1}(x)$ представляет собой параболы, концы которых соединяют заданные точки.

2.3 Сплайны Шенберга

Пример 2.5. Пусть функция $f(x)$ снова задана таблицей из примера 2.2. Программная реализация задачи построения интерполяционного сплайна $S_{3,1}(x)$ третьей степени с дефектом 1 (Шенберга), который удовлетворяет двум дополнительным условиям $S''_{3,1}(a) = S''_{3,1}(b) = 0$, для рассматриваемой функции выполнена в среде *Mathematica* и представлена в Приложении В.

2.4 Эрмитовы кубические сплайны

Пример 2.6. Пусть функция $f(x)$ снова задана таблицей из примера 2.2., которую дополним значениями производной. В результате исходные данные примут следующий вид:

x_k	0,1	0,3	0,7	1,4
$f(x_k)$	1	-7	15	25
$f'(x_k)$	-7	10	22	7

Программная реализация задачи построения эрмитова кубического сплайна $S_{3,2}(x)$ третьей степени с дефектом 2, который удовлетворяет дополнительным условиям $S'_{32}\left(x_i\right) = f'_i;\quad \forall x \in \left[x_i;\, x_{i+1}\right],\ i = \overline{0, N-1}$, для рассматриваемой функции выполнена в среде *Mathematica* и представлена в Приложении Г. Выполнено сравнение полученного результата с графиком, полученным через встроенную функцию Spline. В итоге построенный сплайн лишь частично совпадает с проверочным, что происходит из-за различия в дефекте (функция Spline строит с дефектом 1) и заранее заданных первых производных в узлах сетки.

ЗАКЛЮЧЕНИЕ

В ходе самостоятельных исследований по теме курсовой работы получены следующие результаты.

1. Изучен и изложен теоретический материал [3, с. 48–52], [4, с. 27–36], касающийся задачи алгебраического интерполирования сплайнами: сформулирована интерполяционная задача, рассмотрены ее частные случаи, кратко описан механизм работы встроенной функции Spline компьютерной алгебры *Mathematica* (Глава 1) и приведены примеры ее работы в Приложении Д.

2. Построены различные интерполяционные сплайны для конкретных функций как аналитически (Глава 2), так и программным путем в среде *Mathematica* (Приложения А–Г).

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. *Крылов В. И., Бобков В. В., Монастырный П. И.* Начала теории вычислительных методов. Интерполирование и интегрирование. – Мн.: Наука и техника, 1983.
2. *Мысовских И. И.* Лекции по методам вычислений: учебное пособие. – СПб.: Изд–во С. Петерб. ун–та, 1998.
3. *Игнатенко М. В.* Методы вычислений. Интерполирование и интегрирование: курс лекций. – Минск: БГУ, 2006.
4. Монастырный П. И., Азаров А. И., Игнатенко М. В. [и др.]. Интерполирование функций и численное интегрирование: вычислительный практикум для студентов мех.-мат. фак. спец. 1-31 03 01 «Математика (по направлениям)», 1-31 03 02 «Механика (по направлениям)». – Минск: БГУ, 2008.
5. *Завьялов, Ю. С.* Методы сплайн-функций / Ю. С. Завьялов, Б. И. Квасов, В. Л. Мирошниченко. – М.: Наука, 1980.
6. *Воробьев Е. М.* Введение в систему «Математика». – М.: Финансы и статистика, 1998.

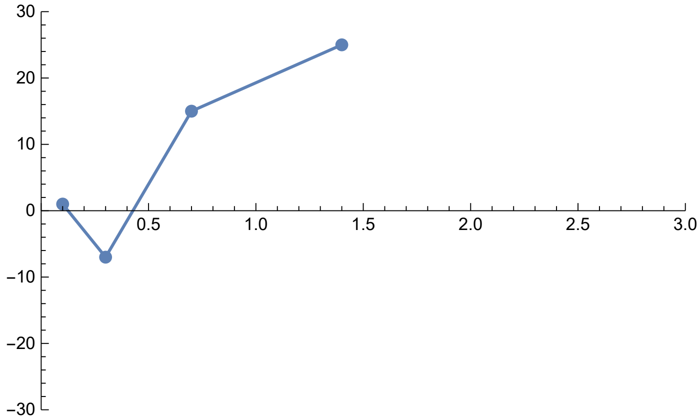
ПРИЛОЖЕНИЕ А

```
In[1]:= Needs["Splines`"]
        необходимо

In[2]:= (*Введём начальные значения, по которым будем строить сплайн*)

In[3]:= n = 3;

In[4]:= Uzl = {0.1, 0.3, 0.7, 1.4};

In[5]:= Znach = {1, -7, 15, 25};

In[6]:= Do[{x_i = Uzl[[i + 1]], f_i = Znach[[i + 1]]}, {i, 0, n}]
        оператор цикла

In[7]:= (*Построим линейный сплайн с дефектом 1 - S_{1,1}(x)*)

In[8]:= P_k_[X_] = f_k + (X - x_k) * A_k;

In[9]:= (*Составим систему уравнений для нахождения коэффициентов*)

In[10]:= eq1 = Table[P_k[x_{k+1}] == P_{k+1}[x_{k+1}], {k, 0, n - 2}];
               таблица значений

In[11]:= eq2 = {P_{n-1}[x_n] == f_n};

In[12]:= eq = Join[eq1, eq2];
              соединить

In[13]:= (*Найдём их, построим график полученного сплайна*)

In[14]:= koef = Solve[eq, {}] // Flatten
                решить уравнения  уплостить

Out[14]= {A_0 -> -40., A_1 -> 55., A_2 -> 14.2857}

In[15]:= Spl[X_] = Table[P_k[X] /. koef, {k, 0, n - 1}] // Expand
                   таблица значений                    раскрыть

Out[15]= {5. - 40. X, -23.5 + 55. X, 5. + 14.2857 X}

In[16]:= Gr1 = ListPlot[Table[{x_i, f_i}, {i, 0, n}], PlotStyle -> {PointSize[0.02]},
               диаграмм… таблица значений              стиль графика  размер точки
           PlotRange -> {{0, 3}, {-30, 30}}];
           отображаемый диапазон графика

In[17]:= Gr2 = Table[Plot[Spl[X][[k]], {X, x_{k-1}, x_k}], {k, 1, n}];
               табл… график функции
```

In[18]:= `Show[Gr1, Gr2]`

показать

Out[18]=

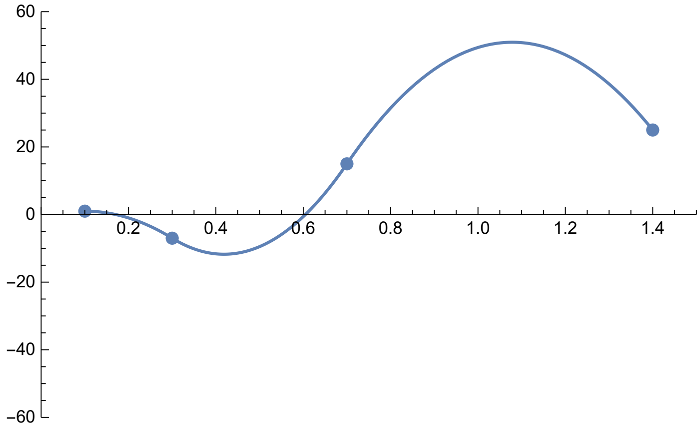


ПРИЛОЖЕНИЕ Б

```
In[1]:= Needs["Splines`"];
        необходимо

In[2]:= (*Введеём начальные значения, по которым будем строить сплайн*)

In[3]:= n = 3;

In[4]:= Uzl = {0.1, 0.3, 0.7, 1.4};

In[5]:= Znach = {1, -7, 15, 25};

In[6]:= Do[{x_i = Uzl[[i + 1]], f_i = Znach[[i + 1]]}, {i, 0, n}]
        оператор цикла

In[7]:= Gr1 = ListPlot[Table[{x_i, f_i}, {i, 0, n}], PlotStyle → {PointSize[0.02]},
              диаграмм··  таблица значений              стиль графика  размер точки
           PlotRange → {{0, 1.5}, {-60, 60}}];
           отображаемый диапазон графика

In[8]:= (*Построим сплайн второй степени с дефектом 1 - S_{2,1}(x)*)

In[9]:= P_k_[X_] = f_k + (X - x_k) * (A_k * X + B_k);

In[10]:= eq1 = Table[P_k[x_{k+1}] == P_{k+1}[x_{k+1}], {k, 0, n - 2}];
               таблица значений

In[11]:= eq2 = {P_{n-1}[x_n] == f_n};

In[12]:= eq3 = Table[P_k'[x_{k+1}] == P_{k+1}'[x_{k+1}], {k, 0, n - 2}];
               таблица значений

In[13]:= eq4 = {P_0'[x_0] == 0};

In[14]:= eq = Join[eq1, eq2, eq3, eq4];
              соединить

In[15]:= koef  =  Solve[eq, {}] // Flatten
                  решить уравнения  уплостить

Out[15]= {A_0 → -200., A_1 → 337.5, A_2 → -251.02, B_0 → 20., B_1 → -181.25, B_2 → 365.714}

In[16]:= Spl[X_] = Table[P_k[X] /. koef, {k, 0, n - 1}] // Expand
                   таблица значений                       раскрыть

Out[16]= {-1. + 40. X - 200. X^2, 47.375 - 282.5 X + 337.5 X^2, -241. + 541.429 X - 251.02 X^2}

In[17]:= Gr3 = Table[Plot[Spl[X][[k]], {X, x_{k-1}, x_k}, PlotRange → All], {k, 1, n}];
               табл···  график функции                    отображаем··· всё
```

In[18]:=
```
Show[Gr1, Gr3]
```
показать

Out[18]=

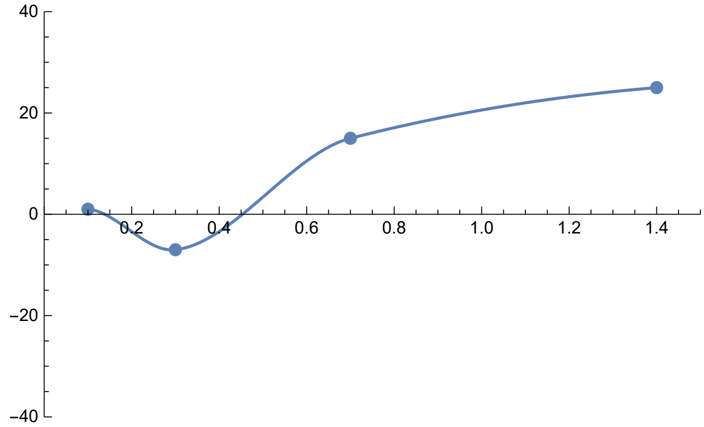


ПРИЛОЖЕНИЕ В

```
In[1]:= Needs["Splines`"];
        необходимо

In[2]:= (*Введеём начальные значения, по которым будем строить сплайн*)

In[3]:= n = 3;

In[4]:= Uzl = {0.1, 0.3, 0.7, 1.4};

In[5]:= Znach = {1, -7, 15, 25};

In[6]:= dF = {-7, 10, 22, 7};

In[7]:= Do[{x_i = Uzl[[i + 1]], f_i = Znach[[i + 1]], df_i = dF[[i + 1]]}, {i, 0, n}]
        оператор цикла

In[8]:= Gr1 = ListPlot[Table[{x_i, f_i}, {i, 0, n}], PlotStyle → {PointSize[0.02]},
              диаграмм·· таблица значений            стиль графика  размер точки
          PlotRange → {{0, 1.5}, {-40, 40}}];
          отображаемый диапазон графика

In[9]:= (*Построим сплайн Эрмита - S_{3,2}(x)*)

In[10]:= Do[t_i[X_] = (X - x_i)/(x_{i+1} - x_i), {i, 0, n - 1}];
         оператор цикла

In[11]:= Do[{A_k = -2 (f_{k+1} - f_k) / (x_{k+1} - x_k) + (df_k + df_{k+1}), B_k = -A_k + (f_{k+1} - f_k) / (x_{k+1} - x_k) - df_k},
         оператор цикла
           {k, 0, n - 1}];

In[12]:= P_k_[X_] := f_k + (X - x_k) (df_k + t_k[X] (B_k + t_k[X] A_k));

In[13]:= P[X_] = Table[P_k[X], {k, 0, n - 1}] // Expand
                 таблица значений               раскрыть

Out[13]= {-6.175 + 171.25 X - 1202.5 X^2 + 2075. X^3,
          30.8375 - 306.125 X + 746.25 X^2 - 487.5 X^3, -6.4 + 39.5714 X - 13.4694 X^2 + 0.874636 X^3}

In[14]:= G32 = Table[Plot[P[X][[k]], {X, x_{k-1}, x_k}, PlotStyle → PointSize[0.05],
               табл··· график функции                   стиль графика размер точки
            PlotRange → {{0, 3}, {-45, 30}}], {k, 1, n}];
            отображаемый диапазон графика
```

In[15]:=
```
Show[Gr1, G32]
```
показать

Out[15]=



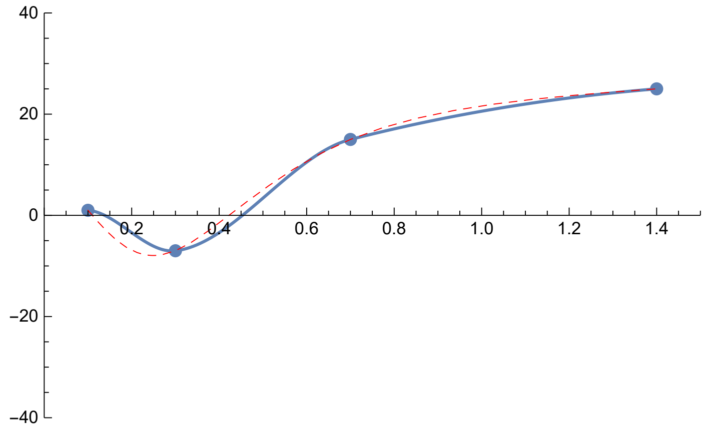
In[16]:=
```
(*Сравним построенный сплайн с результатом
 встроенной функцией Spline(показана красной штриховкой)*)
```

In[17]:=
```
Points = {{0.1, 1}, {0.3, -7}, {0.7, 15}, {1.4, 25}};
```

In[18]:=
```
GrH1 = Graphics[{Red, Dashed, Spline[Points, Cubic]}];
```
графика кр⋯ штриховой пунктир

In[19]:=
```
Show[Gr1, G32, GrH1]
```
показать

Out[19]=



ПРИЛОЖЕНИЕ Г

In[1]:=
```
Needs["Splines`"];
```
необходимо

In[2]:=
```
(*Введеём начальные значения, по которым будем строить сплайн*)
```

In[3]:=
```
n = 3;
```

In[4]:=
```
Uzl = {0.1, 0.3, 0.7, 1.4};
```

In[5]:=
```
Znach = {1, -7, 15, 25};
```

In[6]:=
```
dF = {-7, 10, 22, 7};
```

In[7]:=
```
Do[{x_i = Uzl[[i + 1]], f_i = Znach[[i + 1]], df_i = dF[[i + 1]]}, {i, 0, n}]
```
оператор цикла

In[8]:=
```
Gr1 = ListPlot[Table[{x_i, f_i}, {i, 0, n}], PlotStyle → {PointSize[0.02]},
   PlotRange → {{0, 1.5}, {-40, 40}}];
```
диаграмм·· | таблица значений | стиль графика | размер точки

отображаемый диапазон графика

In[9]:=
```
(*Рассмотрим построение сплайна Шенберга - S_{3,1}(x)*)
```

In[10]:=
```
P_k_[X_] := f_k + (X - x_k) * (A_k * X^2 + B_k X + C_k);
```

In[11]:=
```
eq1 = Table[P_k[x_{k+1}] == P_{k+1}[x_{k+1}], {k, 0, n - 2}];
```
таблица значений

In[12]:=
```
eq2 = {P_{n-1}[x_n] == f_n};
```

In[13]:=
```
eq3 = Table[P_k'[x_{k+1}] == P_{k+1}'[x_{k+1}], {k, 0, n - 2}];
```
таблица значений

In[14]:=
```
eq4 = {P_0''[x_0] == 0};
```

In[15]:=
```
eq5 = Table[P_k''[x_{k+1}] == P_{k+1}''[x_{k+1}], {k, 0, n - 2}];
```
таблица значений

In[16]:=
```
eq6 = {P_{n-1}''[x_n] == 0};
```

In[17]:=
```
eq = Join[eq1, eq2, eq3, eq4, eq5, eq6]
```
соединить

Out[17]=

$$\{1 + 0.2\,(0.09\,A_0 + 0.3\,B_0 + C_0) == -7.,\; -7 + 0.4\,(0.49\,A_1 + 0.7\,B_1 + C_1) == 15.,\; 15 + 0.7\,(1.96\,A_2 + 1.4\,B_2 + C_2) == 25,$$
$$0.09\,A_0 + 0.3\,B_0 + 0.2\,(0.6\,A_0 + B_0) + C_0 == 0. + 0.09\,A_1 + 0.3\,B_1 + C_1,$$
$$0.49\,A_1 + 0.7\,B_1 + 0.4\,(1.4\,A_1 + B_1) + C_1 == 0. + 0.49\,A_2 + 0.7\,B_2 + C_2,\; 0. + 0.4\,A_0 + 2\,B_0 == 0,$$
$$1.6\,A_0 + 2\,B_0 == 0. + 1.2\,A_1 + 2\,B_1,\; 3.6\,A_1 + 2\,B_1 == 0. + 2.8\,A_2 + 2\,B_2,\; 7.\,A_2 + 2\,B_2 == 0\}$$

In[18]:=
```
koef = Solve[eq, {}] // Flatten
```
решить уравнения | уплостить

Out[18]=

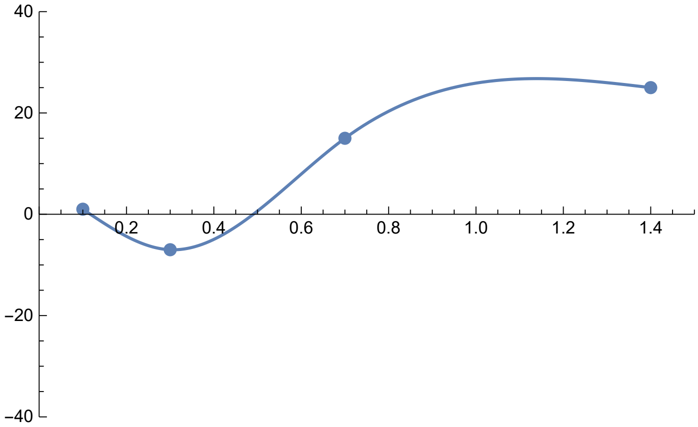
$$\{A_0 \to 454.205,\; A_1 \to -314.66,\; A_2 \to 50.0329,\; B_0 \to -90.841,$$
$$B_1 \to 461.319,\; B_2 \to -175.115,\; C_0 \to -53.6262,\; C_1 \to -113.74,\; C_2 \to 161.382\}$$

```
In[19]:= Spl[X_] = Table [Pk[X] /. koef, {k, 0, n - 1}] // Expand
                   таблица значений                      раскрыть
```

Out[19]= $\{6.36262 - 44.5421\,X - 136.262\,X^2 + 454.205\,X^3,\ 27.122 - 252.136\,X + 555.717\,X^2 - 314.66\,X^3,\ -97.9677 + 283.963\,X - 210.138\,X^2 + 50.0329\,X^3\}$

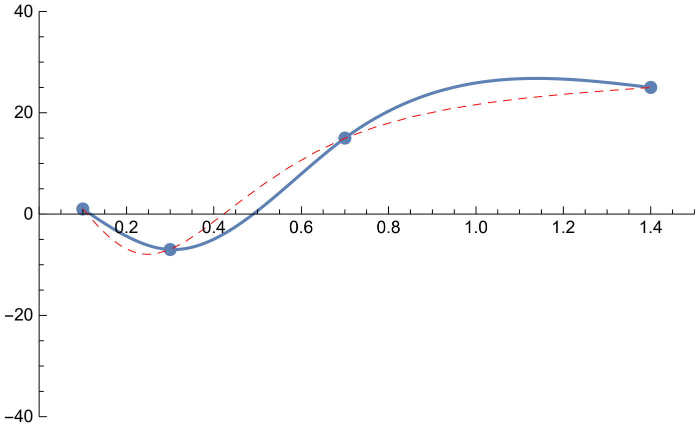
```
In[20]:= Gr4 = Table[Plot[Spl[X][[k]], {X, x_{k-1}, x_k}], {k, 1, n}];
               табл··· график функции

In[21]:= Show[Gr1, Gr4]
         показать
```

Out[21]=




```
In[22]:= (*Сравним построенный сплайн с результатом
          встроенной функцией Spline(показана красной штриховкой)*)

In[23]:= Points = {{0.1, 1}, {0.3, -7}, {0.7, 15}, {1.4, 25}};

In[24]:= GrH1 = Graphics[{Red, Dashed, Spline[Points, Cubic]}];
                графика    кр··· штриховой пунктир

In[25]:= Show[Gr1, Gr4, GrH1]
         показать
```

Out[25]=



ПРИЛОЖЕНИЕ Д

In[1]:=
```
Needs["Splines`"]
```
необходимо

In[2]:=
```
(*Применение встроенной функции Spline*)
```

In[3]:=
```
arr = Table[{Cos[t], Sin[t]}, {t, 0, 2 Pi - Pi / 4, Pi / 4}]
```
таблиц··· косинус синус ч··· число··· число пи

Out[3]= $\left\{\{1, 0\}, \left\{\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right\}, \{0, 1\}, \left\{-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right\}, \{-1, 0\}, \left\{-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right\}, \{0, -1\}, \left\{\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right\}\right\}$

In[4]:=
```
points = ListPlot[arr, PlotRange → {{-1.2, 1.2}, {-1.2, 1.2}}];
```
диаграмма раз··· отображаемый диапазон графика

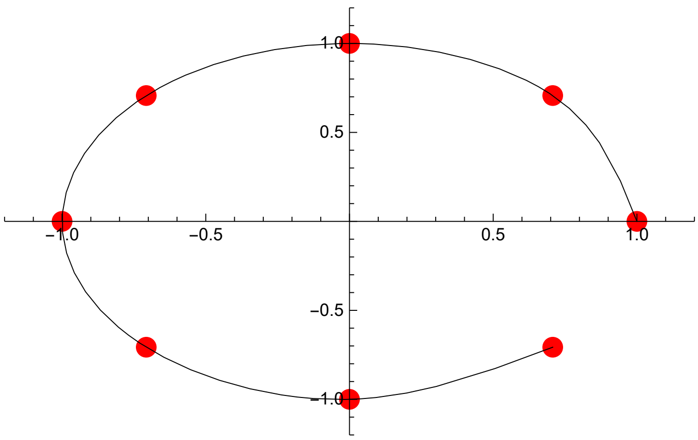
In[5]:=
```
cubic = Graphics[Spline[arr, Cubic, SplineDots → Automatic], PlotRange → Automatic];
```
графика автоматичес··· отображаем··· автоматический

In[6]:=
```
bezier = Graphics[Spline[arr, Bezier, SplineDots → Automatic], PlotRange → Automatic];
```
графика автоматичес··· отображаем··· автоматический

In[7]:=
```
compositeBezier = Graphics[Spline[arr, CompositeBezier, SplineDots → Automatic],
  PlotRange → Automatic];
```
графика автоматический отображаем··· автоматический

In[8]:=
```
(*Пример кубического сплайна*)
```

In[9]:=
```
Show[points, cubic]
```
показать

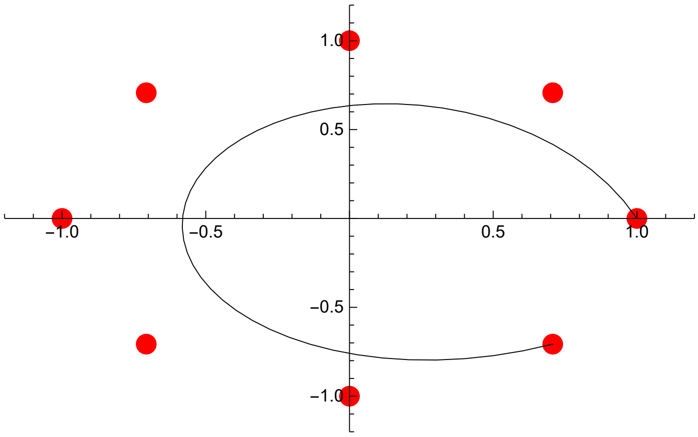
Out[9]=


In[10]:=
```
(*Пример сплайна Безье*)
```

In[11]:= `Show[points, bezier]`

показать

Out[11]=



In[12]:= `(*Пример составного сплайна Безье*)`

In[13]:= `Show[points, compositeBezier]`

показать

Out[13]=