Лабораторна робота № 7

Інтерполяція і наближення функцій

Мета роботи: отримання практичних навичок в побудові формул інтерполювання для функцій однієї змінної, заданих на відрізку у вигляді таблиці.

Короткі теоретичні відомості

7.1. Постановка задачі наближення функцій

На практиці часто доводиться заміняти одну функцію $f(x)$ деякою функцією $\varphi(x)$, близькою до $f(x)$, яка має визначені властивості. Така заміна дозволяє виконувати над нею ті чи інші аналітичні або обчислювальні операції. У тому випадку, якщо $f(x)$ задана таблицею значень для деякої кінцевої множини аргументів x: $f(x_0)$, $f(x_1),\ldots,$ $f(x_n)$, і в процесі вирішення задачі необхідно використовувати значення $f(x)$ для проміжних значень аргументу, функцію $\varphi(x)$ будують таким чином, щоб у заданих точках $x_0, x_1,\ldots, x_n$ вона мала значення, що збігаються зі значеннями $f(x_0), f(x_1),\ldots, f(x_n)$, а в інших точках відрізка $[a,b]$, що належить області визначення $f(x)$, представляла функцію $f(x)$ з тим чи іншим ступенем точності. Тоді при розв’язуванні задачі замість функції $f(x)$ оперують з функцією $\varphi(x)$ а задача побудови функції $\varphi(x)$ називається задачею наближення. Найчастіше функцію $\varphi(x)$, яка використовується під час наближення, будують у вигляді алгебраїчного багаточлена

$$\varphi(x) = c_0 + c_1 x + c_2 x^2 + \ldots + c_m x^m. \tag{7.1}$$

Для знаходження коефіцієнтів c_i, i=0,1,2,…,m використовують вимогу рівності $\varphi(x_j) = f(x_j)$, j=0,1,2,…,n і формують систему з (n+1) лінійних алгебраїчних рівнянь на множині точок $x_0, x_1,\ldots, x_n$. За умови $n = m$ система рівнянь має єдиний розв’язок у випадку, коли вектори $\varphi_i(x_j)$, i, j = 0,1,2,…,n лінійно незалежні. Виникаюча при цьому задача наближення називається задачею інтерполяції. Якщо $m < n$ то система рівнянь може бути розв’язана методом найменших квадратів для мінімізації нев’язки.

Для випадку $n = m$ і вибору в якості *базисних* функцій поліномів коефіцієнти c_i можна одержати з такої системи рівнянь:

$$\begin{cases} c_0 + c_1 x_0 + c_2 x_0^2 + \ldots + c_n x^n{}_0 = f(x_0); \\ c_0 + c_1 x_1 + c_2 x_1^2 + \ldots + c_n x^n{}_1 = f(x_1); \\ \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\ c_0 + c_1 x_n + c_2 x_n^2 + \ldots + c_n x^n{}_n = f(x_n). \end{cases} \tag{7.2}$$

7.2. Інтерполяційний багаточлен Лагранжа

Виходячи з того, що шуканий поліном $\varphi(x)$ повинен мати в заданих вузлах $x_0, x_1,\ldots, x_n$ значення, що збігаються зі значеннями $f(x_0), f(x_1),\ldots, f(x_n)$ можна записати $\varphi(x)$ у вигляді:

$$\varphi(x) = \sum_{j=0}^{n} y_i \, \Phi_j(x), \tag{7.3}$$

де $\Phi_j(x)$ – багаточлен ступеня n, що задовольняє умовам:

$$\Phi_j(x_i) = \begin{cases} 1, & if \quad i = j; \\ 0. & if \quad i \neq j. \end{cases}$$

Такий варіант запису багаточлена $\varphi(x)$ називають інтерполяційним поліномом Лагранжа. Для пошуку $\Phi_j(x)$ знаходять багаточлен степеня n, що обертається в нуль у вузлах інтерполяції x_0, $x_1,\ldots,x_{j-1},x_{j+1},\ldots$, x_n і дорівнює 1 у точці x_j. Багаточлен, що задовольняє цим вимогам, може бути записаний у вигляді:

$$\Phi_j(x) = \frac{(x-x_0)(x-x_1)\ldots(x-x_{j-1})(x-x_{j+1})\ldots(x-x_n)}{(x_j-x_0)(x_j-x_1)\ldots(x_j-x_{j-1})(x_j-x_{j+1})\ldots(x_j-x_n)}.$$

Тоді, якщо вирази $\Phi_j(x)$ визначені зазначеним вище чином, то інтерполяційний багаточлен (7.3) називається інтерполяційним багаточленом Лагранжа. Його позначають як $L_n(x)$ для того, щоб відрізняти цю формулу від інших випадків інтерполяції, остаточно:

$$L_n(x) = \sum_{j=0}^{n} y_j \frac{(x-x_0)(x-x_1)\ldots(x-x_{j-1})(x-x_{j+1})\ldots(x-x_n)}{(x_j-x_0)(x_j-x_1)\ldots(x_j-x_{j-1})(x_j-x_{j+1})\ldots(x_j-x_n)}. \quad (7.4)$$

Якщо значення x_i є рівновіддаленими (як у випадку приклада 7.1), тобто $x_i = x_0 + jh$, $i = 1,2,\ldots,n$, то задача інтерполяції значно спрощується. Можна ввести позначення

$$\frac{x-x_0}{h} = t,$$

і інтерполяційний поліном буде мати вигляд:

$$L_n(x) = (-1)^n \frac{t(t-1)\cdots(t-n)}{n!} \sum_{i=0}^{n} (-1)^i \frac{C_n^i y_i}{t-i}. \quad (7.5)$$

де коефіцієнти перед знаком суми

$$(-1)^n \frac{t(t-1)\ldots(t-n)}{(t-i)n!} C_n^i,$$

не залежать ні від значень функції $f(x)$, ні від відстані між вузлами інтерполяції h. Їх називають коефіцієнтами Лагранжа. Розглянемо можливість застосування коефіцієнтів Лагранжа для прикладу 7.1 і порівняємо результати.

Приклад 7.1

Нехай деяка функція задана в вузлах своїми значеннями:

```
X={1.,1.5,2.,2.5,3.};
F={5.,3.5,1.25,1.,1.5};
n =5;
j=Table[i,{i,0,n}];
G=(-1)^(n-j)*t*Product[(t-i),{i,1,n}]/(Factorial[j]*Factorial[n-
j]*(t-j))
Y=Sum[G[[i]]*F[[i]],{i,1,5}]

0.208333 (-4 + t) (-3 + t) (-2 + t) (-1 + t) - 0.583333 (-4 + t)
(-3 + t) (-2 + t) t + 0.3125 (-4 + t) (-3 + t) (-1 + t) t -
0.166667 (-4 + t) (-2 + t) (-1 + t) t + 0.0625 (-3 + t) (-2 + t)
(-1 + t) t
```

Замінимо в отриманому виразі параметр t на значення `(x - x`$_0$`)/h = t,` де `x0=1` і `h=0.5` будемо мати:

```
Y1=Y/.t->(x-1)/0.5

0.208333 (-4 + 2. (-1 + x)) (-3 + 2. (-1 + x)) (-2 + 2. (-1 + x))
(-1 + 2. (-1 + x)) - 1.16667 (-4 + 2. (-1 + x)) (-3 + 2. (-1 + x))
(-2 + 2. (-1 + x)) (-1 + x) + 0.625 (-4 + 2. (-1 + x)) (-3 + 2. (-
1 + x)) (-1 + 2. (-1 + x)) (-1 + x) -  0.333333 (-4 + 2. (-1 + x))
(-2 + 2. (-1 + x)) (-1 + 2. (-1 + x)) (-1 + x) + 0.125 (-3 + 2. (-
1 + x)) (-2 + 2. (-1 + x)) (-1 + 2. (-1 + x)) (-1 + x)
```

або після спрощення

```
Expand[Y1]
```

$-25.25 + 75.9167\ x - 65.3333\ x^2 + 22.3333\ x^3 - 2.66667\ x^4$

Використаємо засоби пакета Mathematica

```
TA={{1.,5.},{1.5,3.5},{2,1.25},{2.5,1.},{3.,1.5}};
yp[z_]: = InterpolatingPolynomial[TA,z];
yp[z]

1.5+(-1.75+(2.+(2.33333-2.66667(-1.5+z))(-2+z))(-1.+z))(-3.+z)
```

і порівняємо результат з `Y1`, що був отриманий вище

```
Expand[yp[z]]
```

$-25.25 + 75.9167\ z - 65.3334\ z^2 + 22.3334\ z^3 - 2.66667\ z^4$

Поліноми збіглися, що свідчить про правильність визначення функції. Побудуємо графіки отриманого багаточлена (товста лінія) і тих вузлів, які були задані за допомогою таблиці TA

```
p1=Plot[-25.25 + 75.9167*x - 65.3333*x^2  + 22.3333*x^3  -
2.66667*x^4,{x,1,3}];p2=ListPlot[{{1.,5.},{1.5,3.5},{2,1.25},{2.5,
1.},{3.,1.5}}];
Show[p1,p2]
```

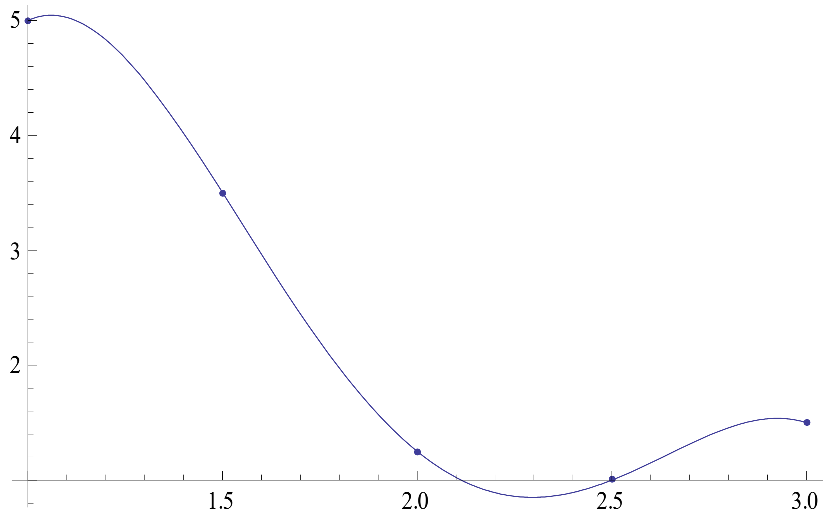


Рис. 7.1. Інтерполювання таблиці за допомогою багаточлена Лагранжа

Приклад 7.2.

Інтерполювати функцію

f(x)=e^x + x^2-2.

Спочатку виберемо вузли інтерполяції і побудуємо компоненти таблиці TA:

```
K1=Exp[x]+x^2-2;
x={0,0.5,1,1.5,2,2.5};
Y1=N[K1]

{-1.,-0.101279,1.71828,4.73169,9.38906,16.4325}
```

Знаходимо інтерполюючий поліном:

```
TA=Table[{x[[i]],Y1[[i]]},{i,1,6}]

{{0,-1.},{0.5,-0.101279},{1,1.71828},{1.5,4.73169},{2,9.38906},
{2.5,16.4325}}

Y2=InterpolatingPolynomial[TA,z]
Y2= Expand[Y2]

16.4325 +(-2.5+z) (6.973 +(2.83648 +(0.720462 +(0.148708
+0.0306379 (-0.5+z)) (-2+z)) (-1+z)) z)
```

$-1.+1.01601\ z+1.42887\ z^2+0.27788\ z^3-0.0351192\ z^4+0.0306379\ z^5$

Обчислюємо значення полінома в вузлах:

```
z= {0,0.5,1.,1.5,2.,2.5};
Y2

{-1.,-0.101279,1.71828,4.73169,9.38906,16.4325}
```

Оцінимо похибку:

```
Y1-Y2
```

$\{0.,7.49401\times10^{-16},-6.66134\times10^{-16},0.,-1.77636\times10^{-15},0.\}$

Як видно з результату, значення похибки нехтовно мале. Побудуємо графіки початкової функції (тонка лінія), отриманого багаточлена (товста лінія) і тих вузлів, які були задані за допомогою таблиці TA

```
Clear[x,z];
p1=K1;
p2=InterpolatingPolynomial[TA,x];
p12=Plot[{p1,p2},{x,0,2.5}];
p3=ListPlot[TA];
Show[p12,p3]
```

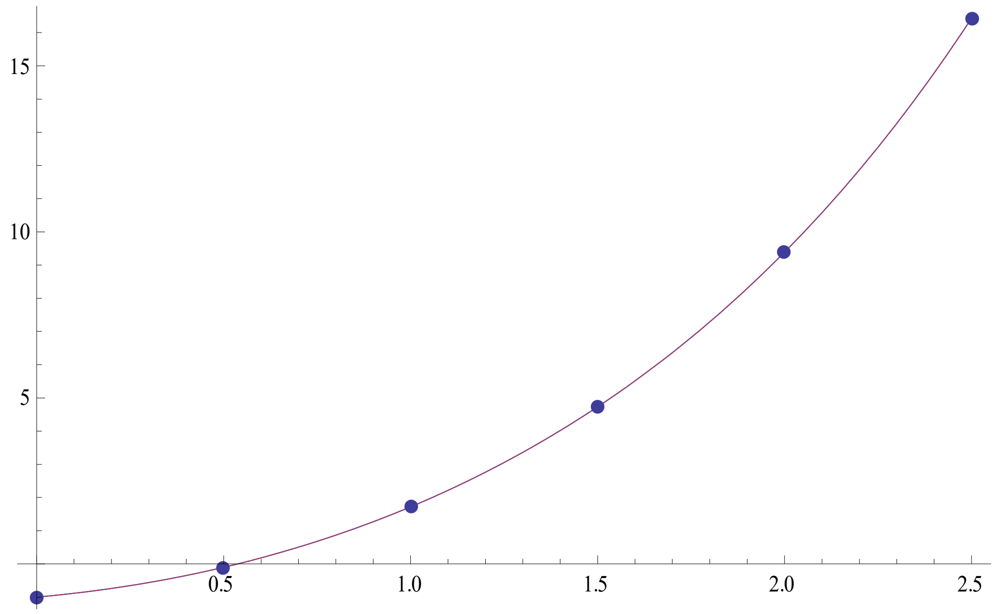


Рис. 7.2. Інтерполювання функції багаточленом Лагранжа

Різницю між функцією $f(x)$ і її інтерполяційним наближенням $L_n(x)$ називають залишковим членом інтерполяційної формули або похибкою інтерполяції:

$$r_n(x) = f(x) - L_n(x). \tag{7.6}$$

Зрозуміло, що у вузлах інтерполяції ця похибка дорівнює нулю, тому вона оцінюється в інших точках відрізка інтерполяції. У загальному випадку похибку інтерполяції полінома Лагранжа можна оцінити в такий спосіб:

$$|f(x) - L_n(x)| \leqslant \frac{M_{n+1}}{(n+1)!} \cdot |\omega(x)|, \tag{7.7}$$

де

$$\omega(x) = (x - x_0)(x - x_1)...(x - x_n); \tag{7.8}$$

$$M_{n+1} = \sup_{x \in [a,b]} \left| f^{(n+1)}(x) \right|.$$

Якщо $f(x)$ є алгебраїчним поліномом ступеня n, інтерполяція, проведена по будь-яких точках $x_0, x_1, \dots, x_n$, здійснюється точно, тобто

$$L_n(x) \equiv f(x).$$

Залежність (7.7) дає можливість провести апріорну оцінку похибки, тобто для випадку аналітично заданої функції $f(x)$ провести оцінку до початку обчислень.

7.2. Інтерполяційні формули Ньютона

Формули Ньютона є іншою формою запису багаточлена Лагранжа, і відрізняються ці формули тільки позначенням (за умови, що в них використані ті ж самі вузли інтерполяції). Існує дві формули Ньютона – для інтерполяції вперед – *перша формула Ньютона*, для інтерполяції назад – *друга формула Ньютона*. Вибір однієї з них обумовлений тим, що звичайно буває зручніше вести обчислення, якщо при інтерполяції спочатку використовуються найближчі до x вузли, а потім підключаються більш віддалені. При цьому перші члени інтерполяційних формул дадуть основний внесок у шукану величину, а інші будуть давати лише невелику поправку. У цьому випадку легше встановити скільки членів у формулі варто використовувати.

Використовуючи розділені різниці, можна одержати 1-*у формулу Ньютона* для нерівних проміжків у вигляді:

$$\begin{aligned} H_n(x) = f(x_0) + (x - x_0) f(x_0; x_1) + (x - x_0)(x - x_1) f(x_0; x_1; x_2) + \ldots \\ \ldots + (x - x_0)(x - x_1)\ldots(x - x_{n-1}) f(x_0; x_1; \ldots; x_n), \end{aligned} \tag{7.9}$$

де розділені різниці будь-якого порядку знаходяться за допомогою формули:

$$f(x_{i-1}; x_i; ...; x_{i+k}) = \frac{f(x_i; x_{i+1}; \ldots; x_{i+k}) - f(x_{i-1}; x_i, \ldots, x_{i+k-1})}{x_{i+k} - x_{i-1}}.$$

У тому випадку, якщо шукані точки розташовані ближче до кінця таблиці, використовується *2-а формула Ньютона* для інтерполяції назад**:**

$$\begin{aligned} H_n(x) = f(x_n) + (x - x_n) f(x_n; x_{n-1}) + \\ +(x - x_n)(x - x_{n-1}) f(x_n; x_{n-1}; x_{n-2}) + \ldots + \\ +(x - x_n)(x - x_{n-1})\ldots(x - x_1) f(x_n; x_{n-1}; \ldots; x_1; x_0). \end{aligned} \tag{7.10}$$

Однак дуже часто ці вузли обрано з постійним кроком. Якщо відстань між сусідніми вузлами $x_i - x_{i-1} = h$, $i = 1,2,\ldots,n$, можна замінити розділені різниці через скінчені за співвідношенням:

$$f(x_0; x_1; \ldots, x_k) = \frac{\nabla^k f(x_0)}{k! h^k}. \tag{7.11}$$

Після заміни $t = (x - x_0)/h$ формула Ньютона для інтерполяції вперед з рівновіддаленими вузлами буде мати вигляд:

$$\begin{aligned} H_n(t) = f(x_0) + t\nabla f(x_0) + \frac{t(t-1)}{2!}\nabla^2 f(x_0) + \ldots \\ \ldots + \frac{t(t-1)\ldots(t-(n-1))}{n!}\nabla^n f(x_0). \end{aligned} \tag{7.12}$$

Формула Ньютона для інтерполяції назад з рівновіддаленими вузлами, якщо прийняти, що $t = (x - x_m)/h$ така

$$H_n(t) = f(x_n) + t \cdot \nabla f(x_{n-1}) + \frac{t(t+1)}{2!}\nabla^2 f(x_{n-2}) + \ldots$$
$$\ldots + \frac{t(t+1)\ldots(t+n-1)}{n!}\nabla^n f(x_0). \qquad (7.13)$$

Розглянемо реалізацію методів Ньютона за допомогою наступного прикладу.

Приклад 7.3.

Визначимо функцію, що обчислюється по заданій таблиці TA (для якої раніше ми вже будували поліном Лагранжа в прикладі 7.1) розділену різницю *n*-го порядку:

```
TA={{1.,5.},{1.5,3.5},{2,1.25},{2.5,1.},{3.,1.5}};
rd[ta_,n_]:=Sum[ta[[i,2]]Product[If[j≠i,(ta[[i,1]]-ta[[j,1]]),1],{
j,1,n}],{i,1,n}];
```

Визначимо інтерполяційний поліном Ньютона наступною формулою:

```
New[ta_,n_,x_]:=Sum[rd[ta,k]*Product[(x-ta[[i,1]]),{i,1,k-1}],{k,1,
n}];
```

Отримаємо інтерполяційний поліном по таблиці TA:

```
q = New[TA, Length[TA], x];
Expand[q]
```

$$-25.25 + 75.9167\, z - 65.3334\, z^2 + 22.3334\, z^3 - 2.66667\, z^4$$

Порівнюючи результати прикладів 7.1 і 7.3, можна переконатися, що обидва методи створюють той самий інтерполяційний поліном. Оскільки поліноми Лагранжа і Ньютона побудовані по одній і тій же таблиці, відрізняються тільки формою запису, представлення похибки у вигляді (7.7) справедливо як для формули Лагранжа, так і для формул Ньютона .

7.3. Вибір вузлів інтерполяції

Похибку інтерполяції можна зменшити за допомогою залежностей Чебишева, які дозволяють обрати вузли інтерполяції x_i (при заданому числі вузлів n) у такий спосіб, щоб поліном $\omega_{n+1}(x)$ мав найменше максимальне значення по абсолютній величині на відрізку:

$$x_i = \frac{b+a}{2} + \frac{b-a}{2}\xi_i, \qquad (7.16)$$

де

$$\xi_i = -\cos\frac{2i+1}{2n+2}\pi, \quad (i = 1, 2, \ldots, n).$$

У цьому випадку можна стверджувати, що

$$|\omega_{n+1}(x)| \leqslant ((b-a)/4)^{n+1}. \qquad (7.17)$$

За допомогою запропонованого принципу вибору вузлів інтерполяції можна зменшити похибку.

Приклад 7.4

Обчислимо розташування вузлів за Чебишевим, скориставшись даними приклада 7.2 для наближення аналітично заданої функції:

```
n=6;
a=1;
b=2.5;
Do[y[i] = -Cos[Pi*(2*(i) + 1)/(2*n + 2)],{i,1,n}]
Do[Uzel[i] = N[(b + a)/2 + y[i]*(b-a)/2], {i,1,n }]
TA = Table[Uzel[i], {i,n}]

{1.16363, 1.42459, 1.75, 2.07541, 2.33637, 2.4812}
```

Обчислимо значення функції в вузлах Чебишева:

```
K2=Exp[x] + x^2-2;
x=TA;
Y2=N[K2]

{2.55555,4.18559,6.8171,10.2752,13.8023,16.1119}
```

Знаходимо інтерполяційний поліном:

```
TA1=Table[{TA[[i]],Y2[[i]]},{i,1,6}];
G=InterpolatingPolynomial[TA1,z]

16.1119 +(10.2889 +(4.13194 +(1.10373 +(0.251364 +0.0550376 (-
1.42459+z)) (-2.07541+z)) (-1.75+z)) (-1.16363+z)) (-2.4812+z)

Y2=Expand[G]
```

$-1.25024+1.83275\ z+0.380134\ z^2+0.93788\ z^3-0.238186\ z^4+0.0550376\ z^5$

Обчислюємо значення отриманого полінома в вузлах Чебишева:

```
z= TA;
Y2

{2.55555,4.18559,6.8171,10.2752,13.8023,16.1119}
```

Вибір вузлів за Чебишевим забезпечує краще наближення функції ніж рівномірно відділені вузли. В цьому можна впевнитися, якщо визначити значення поліномів з регулярним розташуванням вузлів і з вузлами Чебишева в проміжних точках, наприклад, `x=1.25` і `x=2.75`. Побудуємо графіки первинної функції, її вузлів та інтерполяційного полінома.

```
Clear[z];
p1=Plot[Y2,{z,0,2.5}];
p2=ListPlot[TA1];
p3=Plot[Exp[z]+z^2-2,{z,0,2.5}];
Show[p1,p2,p3]
```

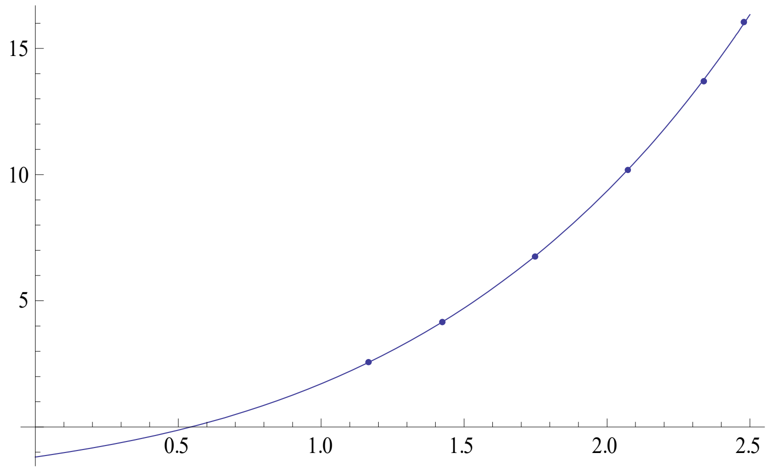


Рис. 7.3. Графік багаточлена , побудованого для функції $f(x)$ = `e^x + x^2-2` з чебишевським розташуванням вузлів

7.4. Інтерполяційні сплайни

Збільшуючи число точок інтерполяції, не завжди можна знизити похибку. Це має місце в тих випадках, коли проміжок $[a,b]$, на якому потрібно наблизити функцією $f(x)$ функцією $\varphi(x)$, великий і немає підстав вважати дану функцію $f(x)$ достатньо гладкою. Тоді немає сенсу підвищувати точність поліноміальної апроксимації за рахунок використання в якості базисних функцій поліноми високих ступенів. У цих умовах більш перспективним є підхід, що використовує кусково-поліноміальну апроксимацію. Ідея такого підходу полягає в наступному: вихідний відрізок $[a,b]$ розбивається на кілька ділянок, а потім на кожній ділянці виконується інтерполяція. Через те, що на окремих ділянках вузлів гладкість функцій є кращою, ніж на усьому вихідному відрізку, то інтерполяція може здійснюватися поліномами меншого ступеня, що значно спрощує задачу. При цьому звичайно вимагають, щоб у точках з'єднання сусідніх ділянок самі поліноми і похідні від них до деякого порядку мали однакові значення.

Приклад 7.5.

Складемо програму, що визначає сплайн-функцію для таблиці `TA` з прикладу 7.1 для таблично заданої функції. Оскільки ми маємо 5 вузлів, то можлива апроксимация двома сплайнами другого порядку, які мають однин спільний вузол `z=2`:

```
TA={{1.,5.},{1.5,3.5},{2,1.25},{2.5,1.},{3.,1.5}};
Y=Interpolation[TA,InterpolationOrder->2]

InterpolatingFunction[{{1., 3.}}, <>]
```

Перевіримо результат у проміжних точках:

```
Y[1.25]

4.34375

Y[2.75]
```

```
1.15625
```

В прикладі 7.1 для цих аргументів можна отримано значення

```
{4.6719, 1.39008}
```

У вузлах значення функції і її наближення співпадають:

```
Y[1.]

5.

Y[.5]

1.5
```

Перша парабола проходить через перші три точки

a0 + a1* z + a2* z^2= 5 в вузлі z= 1
a0 + a1* z + a2* z^2= 3.5 в вузлі z= 1.5
a0 + a1* z + a2* z^2=1.25 в вузлі z= 2

Друга парабола проходить через останні три вузли:

b0 + b1* z + b2* z^2 = 2 в вузлі z= 2
b0 + b1* z + b2* z^2 = 1 в вузлі z= 2.5
b0 + b1* z + b2* z^2 = 1.5 в вузлі z=3

З цих рівнянь можна знайти коефіцієнти парабол. Для першої параболи:

```
z={1.,1.5,2.};
y=z^2

{1., 2.25, 4.}
```

Складемо матрицю Вандермонда і визначимо коефіцієнти:

```
V={{1.,1.,1.},{1.,1.5,2.25},{1.,2.,4.}};
G={5.,3.5,1.25};
B= LinearSolve[V,G]

{5.75, 0.75, -1.5}
```

Перевірка:

```
Y3= 5.75 + 0.75 z - 1.5 z^2;
z=1.25;
Y3

4.34375
```

Для другої параболи:

```
z={2.,2.5,3.};
y=z^2

{4., 6.25, 9.}
```

Складемо матрицю Вандермонда і визначимо коефіцієнти:

```
V1={{1.,2.,4.},{1.,2.5,6.25},{1.,3.,9.}};
G1={1.25,1.,1.5};
B= LinearSolve[V1,G1]

{9.75, -7.25, 1.5}
```

Перевірка:

```
Y4= 9.75 - 7.25 z +1.5 z^2;
z=2.75;
Y4

1.15625
```

Таким чином сплайн- інтерполяція двома параболами для нашої задачі виглядає так:

$1\le z\le 2$	Y3= 5.75 + 0.75 z - 1.5 z^2
$2\le z\le 3$	Y4= 9.75 – 7.25 z +1.5 z^2

7.5. Метод найменших квадратів

Для відновлення функції або заміни складної функції більш простою, а також при обробці експериментальних даних і для розв'язування перевизначених лінійних алгебраїчних систем можна використовувати метод найменших квадратів. Використання цього методу рекомендується і в тому випадку, коли вихідні дані для виявлення тих чи інших закономірностей містять похибку через неточність вимірювальних приладів, неповторюваність умов спостережень, випадкових помилок і т.д.

Задача відновлення деякої функції $f(x)$ методом найменших квадратів вимагає, щоб міра відхилення експериментальних значень від обраної функції була мінімальною в заданих $n + 1$ точках (x_i, y_i), i = 0, 1,.., n.

$$I(f)=\sum_{i=0}^{n}(f(x_i)-\varphi(x_i))^2\to\min. \qquad (7.18)$$

Якщо $\varphi(x)$ має форму полінома, тоді говорять, що функція $\varphi(x)$ є поліномом найкращого середньоквадратичного наближення:

$$\varphi(x)=a_0+a_1x+a_2x^2+\ldots+a_kx^k,\quad k<n, \qquad (7.19)$$

де a_0, a_1,…, a_k – довільні дійсні числа (коефіцієнти узагальненого полінома). Побудова полінома найкращого середньоквадратичного наближення зводиться до знаходження оптимального набору коефіцієнтів згідно з формулою (7.19) на основі методу найменших квадратів, тобто до розв'язку задачі мінімізації:.

$$I(x)=\sum_{i=0}^{n}[a_0+a_1x_i+a_2x_i^2+\ldots+a_kx_i^k-y_i]^2\to\min. \qquad (7.20)$$

Якщо $k = n$, то розглядається задача інтерполяції, при цьому поліном $\varphi(x)$ збігається з інтерполяційним поліномом Лагранжа, побудованим на тих же вузлах. Для вирішення задачі знайдемо часткові похідні від функції $I(a_0, a_1..., a_n)$ за всіма змінними a_j, $j = 0,1,\ldots,n$ і прирівняємо кожну з них до нуля

$$\begin{cases} \dfrac{1}{2}\dfrac{\partial I}{\partial a_0} = \sum\limits_{i=1}^{n}(a_0 + a_1x_i + a_2x_i^2 + ... + a_mx_i^m - y_i)*1 = 0; \\ \dfrac{1}{2}\dfrac{\partial I}{\partial a_1} = \sum\limits_{i=1}^{n}(a_0 + a_1x_i + a_2x_i^2 + ... + a_mx_i^m - y_i)*x_i = 0; \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \dfrac{1}{2}\dfrac{\partial I}{\partial a_m} = \sum\limits_{i=1}^{n}(a_0 + a_1x_i + a_2x_i^2 + ... + a_mx_i^m - y_i)*x_i^m = 0. \end{cases} \quad (7.21)$$

Введемо позначення

$$s_k = \sum_{i=1}^{n} x_i^k / n, k = \overline{0,2m}; \quad b_k = \sum_{i=1}^{n} x_i^k * y_i / n, k = \overline{0,m}. \quad (7.22)$$

Перетворимо систему (7.21), використовуючи введені позначення:

$$\begin{cases} a_0 + a_1s_1 + a_2s_2 + ... + a_ms_m = b_0; \\ a_0s_1 + a_1s_2 + a_2s_3 + ... + a_ms_{m+1} = b_1; \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_0s_m + a_1s_{m+1} + a_2s_{m+2} + ... + a_ms_{2m} = b_m. \end{cases} \quad (7.23)$$

Рішення системи (7.23) визначає значення коефіцієнтів a_i багаточлену (7.19).

Приклад 7.6.

Визначимо по заданій таблиці багаточлен якнайкращого середньоквадратичного наближення. Хай таблична функція задана двома таблицями ta ={xi} і tb ={yi} :

```
ta={1.1,1.2,1.5,2,4,2.7,2.4};
tb={1,2.5,3.4,4.3,5.1,4.2,3.7};
```

Задамо кількість коефіцієнтів системи (ступінь багаточлена буде на одиницю менше) і опишемо допоміжний масив для обчислення коефіцієнтів системи sk

```
m=4;S=Array[s,{m,m}];n=Length[ta];
```

Обчислимо всі sk за формулою (7.22) і утворюємо з них матрицю системи рівнянь (7.23)

```
X=Table[Sum[ta[[i]]^j/n,{i,n}],{j,0,2*(m-1)}];
TA=Table[s[i,j]=X[[i+j-1]],{i,m},{j,m}];
TableForm[TA]
```

1	2.12857	5.42143	15.9916
2.12857	5.42143	15.9916	52.4174
5.42143	15.9916	52.4174	184.401
15.9916	52.4174	184.401	679.239

Обчислимо праві частини системи рівнянь b_k за формулою (7.22)

```
b=Table[Sum[ta[[i]]^j*tb[[i]]/n,{i,n}],{j,0,(m-1)}]
```

```
{3.45714,8.34571,23.3129,73.1062}
```

і отримаємо її рішення

```
A=Inverse[TA].b

{-13.4621,21.5347,-8.4563,1.05846}
```

Запишемо отриманий багаточлен

```
Clear[x]
p[x_]:=Sum[A[[i]]*x^(i-1),{i,m}];
p[x]
```

$-13.4621+21.5347\ x-8.4563\ x^2+1.05846\ x^3$

Побудуємо графік отриманого багаточлена, на який нанесемо задані табличні значення. Складемо з двох таблиць ta і tb таблицю $\{x_i,y_i\}$

```
TT=Transpose[{ta,tb}]

{1.1,1},{1.2,2.5},{1.5,3.4},{2,4.3},{4,5.1},{2.7,4.2},{2.4,3.7}
```

Побудуємо графік табличної функції

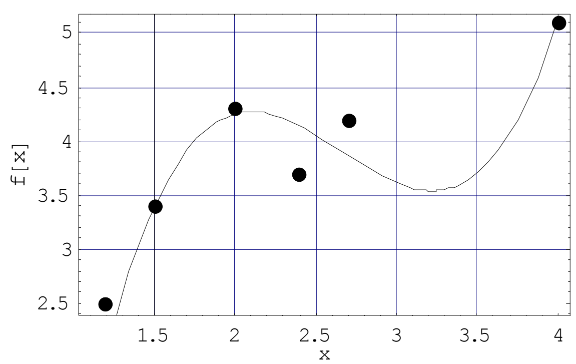
```
g1=ListPlot[TT,PlotStyle->PointSize[0.03],DisplayFunction-
>Identity];
```

Побудуємо графік отриманого багаточлена

```
g2=Plot[p[x],{x,1.1,4},DisplayFunction->Identity];
```

Тепер виведемо обидва графіки

```
Show[g1,g2,GridLines→Automatic,Frame→True,FrameLabel→{"x","f[x]
"},DisplayFunction→$DisplayFunction]
```



Знайдемо відносну середньоквадратичну похибку

```
Is=Sum[(p[TT[[i,1]]]-TT[[i,2]])^2,{i,1,n}]/n

0.0989264
```

Приклад 7.7.

Скориставшись даними прикладу 7.1 ,виконаємо наближення функції, що задана таблицею, за допомогою метода найменших квадратів, обираючи для цього ступені полінома від першого до максимально можливого. Поліном найкращого середньоквадратичного наближення на всіх вузлах має вид:

$$c0 + c1 * z + c2 * z^2 = g.$$

Наведемо приклад програми, що визначає по заданих даних параметри полінома.

```
z={1.,1.5,2.,2.5,3.};
y=z^2;
A={{1.,1.,1.},{1.,1.5,2.25},{1.,2.,4.},{1.,2.5,6.25},{1.,3.,9}};
AT = Transpose[A];
g={5.,3.5,1.25,1.,1.5};
G:= AT.g;
A2=AT.A;
B= LinearSolve[A2,G]

{12.25, -8.75714, 1.71429}
```

Тобто **c0= 12.25, c1= -8.75714, c2=1.71429.**

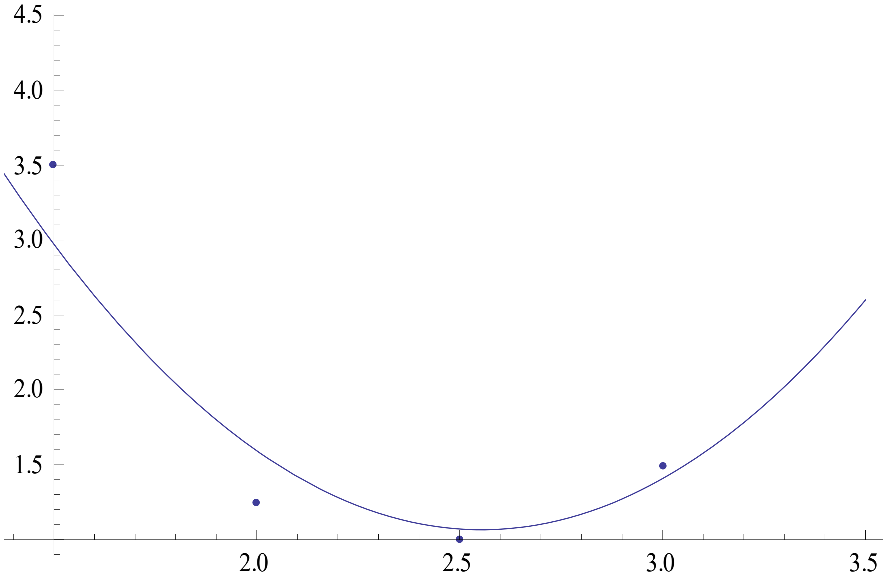
Запишемо отриманий поліном:

```
Y5[x_]= Sum[B[[i]]*x^(i-1), {i, 3}];
Y5[x]
```

$12.25 - 8.75714\ x + 1.71429\ x^2$

Побудуємо графік отриманого полінома найкращого середньоквадратичного наближення з нанесеними координатами вузлів, через які тепер графік не проходить:

```
g1=ListPlot[{{1.,5.},{1.5,3.5},{2,1.25},{2.5,1.},{3.,1.5}}];
g2=Plot[Y5[x],{x,0.5,3.5}]; Show[g1,g2]
```



Оцінимо відхилення в вузлах:

```
x=z;
Y6=N[Y5[x]]
```

```
{5.20714,2.97143,1.59286,1.07143,1.40714}
```

З прикладу 7.1 відомі значення функції в вузлах F={5.,3.5,1.25,1.,1.5}, тому можна підрахувати середньоквадратичне відхилення.

```
Y7={5.,3.5,1.25,1.,1.5};
Y8=Y6-Y7

{0.207143,-0.528571,0.342857,0.0714286,-0.0928571}
```

Обчислимо середньоквадратичне відхилення:

```
M=Sum[Y8[[i]]^2,{i,5}]/5

0.0907143
```

Скористаємося стандартним оператором пакета Mathematica *PolinomialFit* для побудови всіх можливих поліномів для нашої задачі:

```
TA={{1.,5.},{1.5,3.5},{2,1.25},{2.5,1.},{3.,1.5}};
Fit[ TA,{1,x},x]

6.25 - 1.9 x

Fit[ TA,{1,x,x^2},x]

12.25 - 8.75714 x + 1.71429 x²

Fit[ TA,{1,x,x^2,x^3},x]

5.95 + 2.39286 x - 4.28571 x²  + 1. x³

Fit[ TA,{1,x,x^2,x^3,x^4},x]

-25.25 + 75.9167 x - 65.3333 x²  + 22.3333 x³  - 2.66667 x⁴
```

ПОРЯДОК ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

1. Згідно з варіантом за даними таблиці 7.1 побудувати інтерполяційний багаточлен. По тим же точкам, використавши засоби пакета Mathematica, за допомогою функції *InterpolatingPolynomial* отримати поліном і порівняти з побудованим раніше.

2. Обчислити значення функції у проміжних точках.

3. Побудувати графіки отриманих функцій і нанести на них початкові дані з таблиці.

4. По аналітично заданій функції (табл. 7.2) сформувати таблицю вузлів з постійним кроком Δx, що не перевищує 1, $\Delta x \leq 1$, діапазон апроксимації обирається з інтервалу [1;1000]. Побудувати за отриманими даними інтерполяційний поліномом і оцінити отриману похибку, порівнявши на інтервалі початкову аналітично задану функцію і значення поліному. Визначити максимальну розбіжність.

5. Для функції з п.4 розташувати ту ж кількість вузлів за допомогою формул Чебишева. Порівняти розбіжності, що були отримані двома способами.

6. Побудувати графіки функції, значень у вузлах і інтерполяційного полінома на одному рисунку.

7. По даним таблиці 7.1 сформувати систему лінійних рівнянь і виконати сплайн-інтерполяцію. Побудувати графіки отриманих залежностей і полінома з п.1 на одному графіку. Визначити різницю функцій у проміжних точках

8. Виконати наближення функції, що задана таблицею, за допомогою метода найменших квадратів, обираючи для цього степені полінома від першого до максимального можливого. Побудувати графіки. Визначити для кожного випадку значення середньоквадратичного відхилення.

9. Скласти звіт, що складається з отриманих результатів, математичних формул використаних методів по кожному пункту завдання, дати оцінку порівняльної точності отриманих рішень різними методами.

ВАРІАНТИ ЗАВДАНЬ

Табл.7. 1

№ вар.	X_i Yi	точка 0	точка 1	точка 2	точка 3	точка 4	Формула
1	X_i Yi	0.00 1.00	0.50 1.50	1.00 1.75	1.50 2.75	2.00 6.00	Ньютона 1-а
2	Xi Yi	0.00 0.00	0.50 2.00	1.00 2.25	1.50 3.00	2.00 3.25	Ньютона 2-а
3	Xi Yi	1.00 4.00	1.50 2.00	2.00 1.75	2.50 1.25	3.00 1.75	Лагранжа
4	Xi Yi	1.00 3.00	1.40 3.25	1.80 2.75	2.20 2.50	2.60 1.00	Ньютона 1-а
5	Xi Yi	0.10 -0.80	0.20 -0.40	0.30 -0.30	0.40 -0.15	0.50 -0.20	Ньютона 2-а
6	Xi Yi	0.20 1.10	0.40 1.15	0.60 1.175	0.80 1.35	1.00 1.425	Лагранжа
7	Xi Yi	1.00 1.10	1.50 1.30	2.00 1.325	2.50 1.40	3.00 1.375	Ньютона 1-а
8	Xi Yi	0.40 1.50	0.60 1.75	0.80 1.75	1.00 2.00	1.20 3.50	Ньютона 2-а
9	Xi Yi	0.20 0.30	0.60 0.35	1.00 0.275	1.40 0.25	1.80 0.10	Лагранжа
10	Xi Y_i	1.00 -1.00	1.40 -3.00	1.80 -3.25	2.20 -2.75	2.60 -2.90	Ньютона 1-а
11	Xi Yi	0.00 0.80	0.20 0.85	0.40 0.825	0.60 0.90	0.80 0.50	Ньютона 2-а
12	Xi Yi	0.60 2.00	0.90 3.25	1.20 3.00	1.50 1.75	1.80 0.50	Лагранжа
13	Xi Yi	0.20 1.50	0.40 1.275	0.60 1.225	0.80 1.125	1.00 1.10	Ньютона 1-а
14	Xi Yi	0.50 1.00	1.50 2.75	2.00 3.00	2.50 3.50	3.00 3.75	Ньютона 2-а
15	Xi Yi	0.40 0.50	0.60 1.375	0.80 1.50	1.00 1.75	1.20 1.625	Лагранжа

16	Xi Yi	1.00 3.75	1.40 4.25	1.80 4.00	2.20 2.50	2.60 2.00	Ньютона 1-а
17	Xi Yi	3.00 2.75	3.50 3.25	4.00 3.00	4.50 2.75	5.00 3.50	Ньютона 2-а
18	Xi Yi	-2.60 1.50	-2.20 1.00	-1.80 1.25	-1.40 3.50	-1.00 5.00	Лагранжа
19	Xi Yi	1.00 1.00	1.50 3.00	2.00 3.25	2.50 4.00	3.00 3.75	Лагранжа
20	Xi Yi	0.80 0.60	1.20 0.40	1.60 0.70	2.00 1.40	2.40 1.50	Ньютона 1-а
21	Xi Yi	2.00 1.50	2.30 1.00	2.60 1.75	2.90 2.00	3.20 3.00	Ньютона 2-а
22	Xi Yi	0.40 -4.00	0.80 -4.25	1.20 -3.00	1.60 -1.00	2.00 -0.75	Лагранжа
23	Xi Yi	2.00 0.75	2.50 2.00	3.00 1.75	3.50 1.80	4.00 1.00	Ньютона 1-а
24	Xi Yi	1.00 4.00	1.40 2.00	1.80 1.75	2.20 2.50	2.60 2.75	Ньютона 2-а
25	Xi Yi	2.00 3.00	2.50 3.25	3.00 2.50	3.50 2.25	4.00 1.50	Лагранжа
26	Xi Yi	1.20 1.75	1.60 3.00	2.00 2.75	2.40 2.80	2.80 2.00	Ньютона 1-а
27	Xi Yi	-3.00 4.25	-2.50 4.50	-2.00 3.50	-1.50 1.75	-1.00 1.50	Ньютона 2-а
28	Xi Yi	1.00 1.00	1.30 0.75	1.60 1.50	1.90 1.60	2.20 3.25	Лагранжа
29	Xi Yi	2.00 3.75	2.40 3.50	2.80 1.75	3.20 1.25	3.60 1.50	Ньютона 2-а

Табл. 7.2

№ вар.	Формула	№ вар.	Формула
1	$x-\sin(x)=0.25$	15	$\lg(x) - 1/x^2=0$
2	$x^2-20*\sin(x)=0$	16	$x-\lg(x+2)=0$
3	$x^2-\sin(5x)=0$	17	$e^{-x} + x^2=2$
4	$1.8x^2-\sin(10x)=0$	18	$e^{-x} + (x-1)^2=0$
5	$2-x*e^x=0$	19	$x^2-\cos(x^2)=6$
6	$x-\cos(x)=0$	20	$e^x + (x-1)^2=0$
7	$x^2-\cos(4x)=0$	21	$Ln(x)+(x+1)^3=0$
8	$x-\cos^2(2x)=0$	22	$3x+1-1/x=0$
9	$x^2-\cos^2(3x)=0$	23	$3*x-\cos(x)=1$
10	$2*x-\cos(x)=0$	24	$x-0.2*\sin(x+0.5)=0$
11	$x-2*\cos(x)=0$	25	$x+\lg(x)+0.5=0$
12	$2^x-4*x=0$	26	$x*\lg(x)+0.125=0$
13	$2*\ln(x)-1/x=0$	27	$x*\ln(x)-100=0$
14	$2*\lg(x)-x/2=-1$	28	$2-x=\ln(x)$
		29	$(x-1)^2-0.5*e^x=0$