

1 Numerická integrace

1.1 Simpsonova metoda

1.1.1 Odvození

Pokud při výpočtu určitého integrálu

$$I = \int_a^b y(x) dx$$

neznáme primitivní funkci, můžeme použít některou z přibližných numerických metod. Integrační interval $\langle a, b \rangle$ rozdělíme na n menších intervalů pomocí ekvidistantních bodů

$$a = x_0 < x_1 < x_2 < \dots < x_n = b$$

a na každém z nich nahradíme integrovanou funkci mnohočlenem. Při lichoběžníkové metodě použijeme mnohočlen prvního stupně. Pro dosažení vyšší přesnosti můžeme použít mnohočlen druhého stupně. Taková metoda se nazývá Simpsonova metoda. Odvodíme si způsob výpočtu a ukážeme si, jak se používá. Bez újmy na obecnosti můžeme uvažovat uzlové body

$$x_0 = -h, \quad x_1 = 0, \quad x_2 = h.$$

Nejdříve si spočteme integrál

$$\begin{aligned} \int_{-h}^h (a_0 + a_1 x + a_2 x^2) dx &= \left[a_0 x + a_1 \frac{x^2}{2} + a_2 \frac{x^3}{3} \right]_{-h}^h = \\ &= a_0 h + a_1 \frac{h^2}{2} + a_2 \frac{h^3}{3} - (a_0(-h) + a_1 \frac{(-h)^2}{2} + a_2 \frac{(-h)^3}{3}) = 2a_0 h + 2a_2 \frac{h^3}{3}. \end{aligned}$$

Nyní chceme najít hodnoty a_0, a_1, a_2 , aby se integrovaná funkce y rovnala v bodech $x = -h, x = 0$ a $x = h$ polynomu. Označme funkční hodnoty v těchto třech bodech y_0, y_1, y_2 . Tedy aby platilo

$$\begin{aligned} a_0 - a_1 h + a_2 h^2 &= y_0 \\ a_0 &= y_1 \\ a_0 + a_1 h + a_2 h^2 &= y_2 \end{aligned}$$

Z druhé rovnice známe hodnotu $a_0 = y_1$. Sečtením první a třetí rovnice vyločíme a_1 a dostaneme

$$2a_0 + 2a_2 h^2 = y_0 + y_2$$

tedy

$$a_2 = \frac{y_0 + y_2 - 2y_1}{2h^2}$$

a integrál lze aproximovat výrazem

$$\begin{aligned} I &\doteq I_S = 2a_0h + 2a_2\frac{h^3}{3} = 2y_1h + 2\frac{y_0 + y_2 - 2y_1}{2h^2}\frac{h^3}{3} = \\ &= 2y_1h + (y_0 + y_2 - 2y_1)\frac{h}{3} = \frac{h}{3}(y_0 + 4y_1 + y_2). \end{aligned}$$

1.1.2 Použití

Ukážeme si použití lichoběžníkové a Simpsonovy metody na jednoduchém příkladě

$$I = \int_1^2 \frac{1}{x} dx = \left[\ln |x| \right]_1^2 = \ln 2 \doteq 0.693147.$$

Tento integrál lze spočítat přesně, protože známe primitivní funkci. My si na něm ukážeme použití přibližných metod a můžeme přibližné výsledky porovnat s přesnou hodnotou.

Připravíme si tabulku funkčních hodnot pro pár hodnot argumentů, které budeme potřebovat

x	y
1	1
$\frac{5}{4}$	$\frac{4}{5}$
$\frac{3}{2}$	$\frac{2}{3}$
$\frac{7}{4}$	$\frac{4}{7}$
2	$\frac{1}{2}$

Lichoběžníkové pravidlo pro $n = 1$, tedy $h = 1$:

$$I_{L,1} = \frac{h}{2}(y_0 + y_1) = \frac{1}{2}\left(1 + \frac{1}{2}\right) = \frac{3}{4} = 0.75.$$

Lichoběžníkové pravidlo pro $n = 2$, tedy $h = \frac{1}{2}$:

$$I_{L,2} = \frac{h}{2}(y_0 + 2y_1 + y_2) = \frac{1}{4}\left(1 + \frac{4}{3} + \frac{1}{2}\right) = \frac{17}{24} \doteq 0.708.$$

Lichoběžníkové pravidlo pro $n = 4$, tedy $h = \frac{1}{4}$:

$$I_{L,4} = \frac{h}{2}(y_0 + 2y_1 + 2y_2 + 2y_3 + y_4) = \frac{1}{8}\left(1 + \frac{8}{5} + \frac{4}{3} + \frac{8}{7} + \frac{1}{2}\right) = \frac{1171}{1680} \doteq 0.697024.$$

A nyní se podívejme, jaké výsledky nám dá Simpsonova metoda pro $n = 2$, tedy $h = \frac{1}{2}$:

$$I_{S,2} = \frac{h}{3}(y_0 + 4y_1 + y_2) = \frac{1}{6}\left(1 + \frac{8}{3} + \frac{1}{2}\right) = \frac{25}{36} \doteq 0.694444.$$

Simpsonova metoda pro $n = 4$, tedy $h = \frac{1}{4}$:

$$I_{S,4} = \frac{h}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + y_4) = \frac{1}{12}\left(1 + \frac{16}{5} + \frac{4}{3} + \frac{16}{7} + \frac{1}{2}\right) = \frac{1747}{2520} \doteq 0.693254.$$

```
In[1]:= a = 1; b = 2; f[x_] := 1.0/x; ip = Integrate[f[x], {x, a, b}]
```

```
Out[1]= 0.693147
```

```
In[2]:= it[n_] := (h = (b - a) / n;
  s = 0;
  For[i = 0, i ≤ n, i++, s += f[a + i * h] * If[i == 0 || i == n, 1, 2]]];
  s * h / 2);
is[n_] := (h = (b - a) / n;
  s = 0;
  For[i = 0, i ≤ n, i++, s += f[a + i * h] * If[i == 0 || i == n, 1, 2 + 2 * Mod[i, 2]]];
  s * h / 3);
```

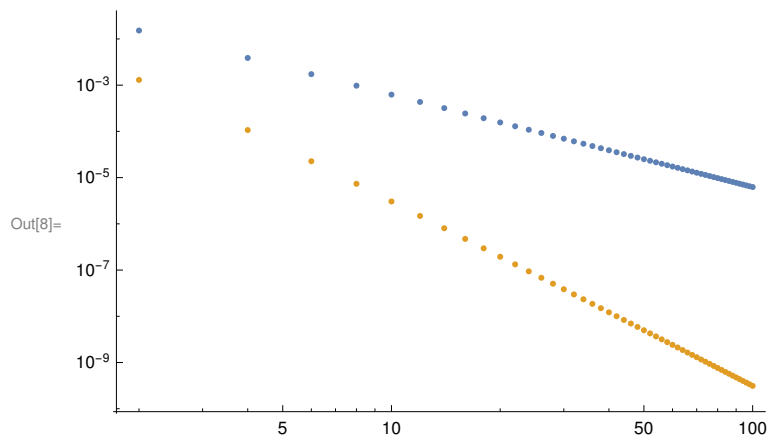
```
In[4]:= it[4]
```

```
Out[4]= 0.697024
```

```
In[5]:= is[4]
```

```
Out[5]= 0.693254
```

```
In[6]:= tt = Table[{n, it[n] - ip}, {n, 2, 100, 2}];
ts = Table[{n, is[n] - ip}, {n, 2, 100, 2}];
ListLogLogPlot[{tt, ts}]
```



```
In[9]:= Fit[Log[tt], {1, x}, x]
```

```
Out[9]= -2.78449 - 1.99701 x
```

```
In[10]:= Fit[Log[ts], {1, x}, x]
```

```
Out[10]= -3.64984 - 3.95393 x
```

Porovnání lichoběžníkové a Simpsonovy metody ukazuje obrázek. Zde jsou vyčísleny hodnoty integrálu přesně, dále lichoběžníkovou metodou pro $n = 4$ a Simpsonovou metodou pro $n = 4$. V grafu je v logaritmickém měřítku vynesena závislost chyby pro různé hodnoty n pro obě tyto metody. Pokud těmito body proložíme přímkou, její směrnice nám říká, s jakou mocninou klesá chyba pro rostoucí n . Pro lichoběžníkovou metodu je chyba přibližně úměrná $\frac{1}{n^2}$, zatímco pro Simpsonovu metodu chyba klesá rychleji, je úměrná $\frac{1}{n^4}$.

1.2 Zobecnění

Při obdélníkové metodě nahradíme integrovanou funkci konstantou, tedy mnohočlenem stupně nula. A graf integrované funkce nahradíme vodorovnou úsečkou. Při lichoběžníkové metodě nahradíme integrovanou funkci mnohočlenem stupně jedna. A graf integrované funkce nahradíme vhodně skloněnou úsečkou. Podobně při Simpsonově metodě nahradíme integrovanou funkci mnohočlenem stupně dva. A graf integrované funkce nahradíme parabolou. Nabízí se otázka, jestli lze tento postup zobecnit a nahradit integrovanou funkci mnohočlenem vyššího řádu. Odpověď zní ano. A čtenář, který zná násobení matice a vektoru a inverzní matici, bude překvapen elegancí a stručností odvození a zápisu této zobecněné metody.

Očekáváme, že dostaneme opět pravidlo, kde se funkční hodnoty integrované funkce vyčíslené v ekvidistantních uzlových bodech vynásobí vhodnými koeficienty. Jak nalézt tyto koeficienty?

Pro pevné přirozené $n \geq 2$ uvažujme n uzlových bodů $x_i = i = 1 \dots n$. Těmito body proložíme křivku, graf mnohočlenu $P(x)$ stupně nejvýše $n - 1$

$$P(x) = \sum_{j=1}^n a_j x^{j-1}.$$

Integrál funkce f budeme aproximovat integrálem mnohočlenu P

$$\int_1^n f(x) dx \doteq \int_1^n P(x) dx.$$

Označme

$$I = \int_1^n P(x) dx = \int_1^n \sum_{j=1}^n a_j x^{j-1} dx = \sum_{j=1}^n a_j \left[\frac{x^j}{j} \right]_1^n = \sum_{j=1}^n a_j \frac{n^j - 1}{j}.$$

Koeficienty v mnohočlenu, tedy čísla a_j pro $j = 1 \dots n$ můžeme považovat za složky sloupcového vektoru $\vec{a}$. Podobně čísla $\frac{n^j - 1}{j}$ pro $j = 1 \dots n$ můžeme považovat za složky sloupcového vektoru $\vec{b}$. Tedy

$$b_j = \frac{n^j - 1}{j}.$$

Pravým horním indexem T budeme značit transponování, tedy operaci, která převádí sloupcový vektor na řádkový (a naopak). Pak počítaný integrál lze psát jako skalární součin těchto dvou vektorů

$$I = \vec{a}^T \vec{b} = \sum_{j=1}^n a_j b_j.$$

Zbývá ještě najít vztah mezi koeficienty a_j v mnohočlenu P a funkčními hodnotami $f_i = f(x_i) = f(i)$. Chceme, aby platilo

$$P(x_i) = f_i, \quad \text{pro } i = 1 \dots n.$$

Tedy

$$\sum_{j=1}^n a_j i^{j-1} = f_i.$$

Výraz na levé straně připomíná násobení matice a vektoru. Zavedeme-li matici X velikosti $n \times n$ s prvky

$$X_{ij} = i^{j-1},$$

pak

$$(X\vec{a})_i = \sum_{j=1}^n X_{ij} a_j = \sum_{j=1}^n a_j i^{j-1}.$$

Zavedeme-li vektor $\vec{f}$ o složkách f_i , kde $i = 1 \dots n$, můžeme podmínku psát ve tvaru

$$X\vec{a} = \vec{f}.$$

Uved'me zde bez důkazu, že $\det X \geq 1$, tedy matice X je regulární a tedy existuje k ní inverzní matice X^{-1} . Takže

$$\vec{a} = X^{-1} \vec{f}$$

a

$$\vec{a}^T = \vec{f}^T (X^{-1})^T.$$

Tedy náš integrál lze psát ve tvaru

$$I = \vec{a}^T \vec{b} = \vec{f}^T (X^{-1})^T \vec{b} = \vec{f}^T \vec{c},$$

kde vektor

$$\vec{c} = (X^{-1})^T \vec{b}$$

je hledaný vektor koeficientů pro naši obecnou integrační formuli. Pro úplnost připomínáme, že

$$X_{ij} = i^{j-1}$$

a

$$b_j = \frac{n^j - 1}{j}.$$

Vyzkoušejme tento právě odvozený vztah pro některé hodnoty n . Můžeme s výhodou použít počítačový algebraický systém *Mathematica*. Pro $n = 2$, tedy dvěma body proložíme přímku, tedy mnohočlen stupně jedna, můžeme postupovat takto:

Příkazy

```
n = 2
x = Table[i^(j-1), {i, n}, {j, n}]
```

si připravíme matici

$$X = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}.$$

Příkazem

```
xi = Inverse[x]
```

spočteme inverzní matici

$$X^{-1} = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}.$$

a příkazem

```
b = Table[(n^j-1)/j, {j, n}]
```

si připravíme vektor

$$\vec{b} = \begin{pmatrix} 1 \\ \frac{3}{2} \end{pmatrix}.$$

A nakonec příkazem

```
c = Transpose[xi].b
```

najdeme vektor

$$\vec{c} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix},$$

tedy koeficienty lichoběžníkové metody.

A nebo si můžeme připravit funkci (nazvěme ji např. k), která všechny tyto úkony provede. Funkce k bude mít jeden argument, dimenzi n . To lze provést např. takto:

```
k[n_] := (
  x = Table[i^(j-1), {i, n}, {j, n}];
  xi = Inverse[x];
  b = Table[(n^j-1)/j, {j, n}];
  c = Transpose[xi].b
);
```

Tuto funkci pak zavoláme s jedním argumentem, počtem bodů. Celý výpočet pak může vypadat např. takto:

```
In[1]:= k[n_]:= (
      x = Table[i^(j-1),{i,n},{j,n}];
      xi = Inverse[x];
      b = Table[(n^j-1)/j,{j,n}];
      c = Transpose[xi].b
    );
```

```
In[2]:= k[2]
```

```
      1  1
Out[2]= {-, -}
      2  2
```

```
In[3]:= k[3]
```

```
      1  4  1
Out[3]= {-, -, -}
      3  3  3
```

```
In[4]:= k[9]
```

```
      3956  23552  3712  41984  3632  41984  3712  23552  3956
Out[4]= {-----, -----, -(-----), -----, -(-----), -----, -(-----), -----, -----}
      14175  14175  14175  14175  2835  14175  14175  14175  14175
```

6.1.2019

Pavel.Pokorny@vscht.cz