Newtonova metoda

Newtonova metoda

Jana Ernekerová, 2014

Teoretické shrnutí

Newtonova metoda nazývaná také metoda tečen je **iterační** numerická metoda, která slouží k nalezení řešení rovnice *f(x) = 0* za předpokladu, že známe derivaci funkce *f'(x)* a dovedeme vypočítat směrnici tečny v daném bodě.

Nezbytným předpokladem je **znalost počáteční hodnoty** x_0, v jejíž blízkosti hledáme řešení. Čím blíže bude počáteční odhad hodnoty x_0, k hledanému kořeni, tím rychlejší bude nalezení kořene.

Postup

Myšlenka Newtonovy metody spočívá v konstrukci posloupnosti (x_n) aproximující řešení rovnice $f(x) = 0$. Konstrukce posloupnosti je následující:

1. Je dáno x_n.
2. Sestroj tečnu funkce f v bodě x_n,

   $$y = f(x_n) + f'(x_n) \cdot (x - x_n)$$

3. Průsečík tečny s osou x nechť je další člen posloupnosti, $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$.
4. Opakuj s x_{n+1} místo x_n.

Výhody a nevýhody

Newtonova metoda je oproti metodě půlení intervalů rychlejší (pokud je první odhad rozumně přesný), avšak metoda půlení intervalů hledání postupně zmenšuje interval, ve kterém se kořen nachází, až se k němu přiblíží se zanedbatelnou chybou, oproti tomu se při použití Newtonovy metody může stát, že se algoritmus zacyklí a nikdy požadovaný kořen nenalezneme. To se stane v případě, že námi konstruovaná posloupnost diverguje. Příkladem funkce, pro kterou Newtonova metoda takto selhává je např. $f(x) = \sqrt{|x|}$.

Metoda sečen

Na podobném principu funguje také další iterativní metoda, metoda sečen. Ke konstrukci posloupnosti aproximující řešení, ale místo tečen využívá sečny. Iterativní vzorec tedy vypadá následovně

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$$

Stejně jako Newtonova metoda je rychlejší než metoda půlení intervalů, ale zároveň sdílí i její nevýhodu, posloupnost, kterou pomocí ni zkonstruujeme ne vždy konverguje k námi hledanému řešení.

Ukázka

Mějme rovnici $x^2 = 2$. Pokud převedeme dvojku na druhou stranu, získáme rovnici $x^2 - 2 = 0$. Její řešení je průsečík grafu funkce $f(x) = x^2 - 2$ s osou x a hledáme tedy numerickou hodnotu $\sqrt{2}$.

Příklad

Najděte numerickou hodnotu $\sqrt{2}$ s přesností na 0,01.

```
def newton(x0,eps,maxiter=10):
    xold = x0
    xnew = x0 - (x0^2 - 2)/(2*x0)
    k = 1
    while abs(xnew-xold) > eps and k <= maxiter:
        xold,xnew = xnew,xnew - (xnew^2 - 2)/(2*xnew)
        k+=1
    return xnew
```

```
print newton(2.0,0.01)
print newton(2.0,0.0001)
print n(sqrt(2.0))
```

```
1.41421568627451
1.41421356237469
1.41421356237310
```

Příklad

Najděte numerickou hodnotu $5^{\frac{1}{3}}$ s přesností na 0,01.

```
def newton(x0,eps,maxiter=10):
    xold = x0
    xnew = x0 - (x0^3 - 5)/(3*x0^2)
    k = 1
    while abs(xnew-xold) > eps and k <= maxiter:
        xold,xnew = xnew,xnew - (xnew^3 - 5)/(3*xnew^2)
        k+=1
    return xnew
```

```
print newton(2.0,0.001)
print newton(2.0,0.00001)
print n(5^(1/3))
```

```
1.70997642891697
1.70997594667683
1.70997594667670
```