

Метод Ньютона для решения систем нелинейных уравнений

Шепрут Илья

19 января 2019 г.

Всё будет описано для размерности $n = 2$, но это легко обобщается на более высокие размерности.

1. Формулировка задачи

Дана Система Нелинейных Уравнений (СНУ) в виде:

$$\begin{cases} F_1(x_1, x_2) = 0 \\ F_2(x_1, x_2) = 0 \end{cases}$$

Требуется, чтобы она имела непрерывные производные 1 порядка.

Необходимо найти такие x_1, x_2 при которых система оборачивается в нуль. Это называется решение СНУ.

2. Обозначения

Её можно преобразовать к следующему виду, заменив аргументы функции на один векторный аргумент $\mathbf{x} = (x_1, x_2)$:

$$\begin{cases} F_1(\mathbf{x}) = 0 \\ F_2(\mathbf{x}) = 0 \end{cases}$$

Далее эту СНУ можно преобразовать к одной вектор-функции \mathbf{F} , такой что $\mathbf{F} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$:

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} F_1(\mathbf{x}) \\ F_2(\mathbf{x}) \end{pmatrix}$$

Тогда исходная система уравнений примет вид:

$$\mathbf{F}(\mathbf{x}) = \mathbf{0}$$

где $\mathbf{0} = (0, 0)^T$.

3. Преобразования

3.1. Разложение в ряд Тейлора для функции многих переменных

Взято с Википедии.

Пусть дана функция $f(\mathbf{x}) : \mathbb{R}^2 \rightarrow \mathbb{R}$, $\mathbf{x} = (x_1, x_2)^T$.

Разложение функции $f(\mathbf{x})$ будет произведено в окрестности точки $\mathbf{a} = (a_1, a_2)^T$. Требуется, чтобы у функции f имелись непрерывные производные до $(2+1)$ -го порядка включительно по всем переменным.

Тогда разложение в ряд Тейлора имеет следующий вид:

$$f(\mathbf{x}) = f(\mathbf{a}) + \sum_{i=1}^n \frac{\partial f(\mathbf{a})}{\partial x_i} (x_i - a_i) + \frac{1}{2!} \sum_{i=1}^2 \sum_{j=1}^2 \frac{\partial^2 f(\mathbf{a})}{\partial x_i \partial x_j} (x_i - a_i)(x_j - a_j) + R_3(\mathbf{x})$$

Раскроем сумму первых двух членов предыдущего равенства:

$$f(\mathbf{x}) = f(\mathbf{a}) + \frac{\partial f(\mathbf{a})}{\partial x_1} (x_1 - a_1) + \frac{\partial f(\mathbf{a})}{\partial x_2} (x_2 - a_2) + R_2(\mathbf{x})$$

Где $R_n(\mathbf{x})$ — остаточный член в форме Лагранжа, $R_n(\mathbf{x}) \xrightarrow{n \rightarrow \infty} 0$.

3.2. Разложение в ряд Тейлора СНУ

Любой итерационный метод имеет начальное приближение $\mathbf{x}^{[0]}$, которое задается извне. На k -м шаге имеется текущее решение $\mathbf{x}^{[k]}$. Обозначим за $\hat{\mathbf{x}}$ искомое решение, и введем ещё обозначение $\Delta \mathbf{x}^{[k]} = \hat{\mathbf{x}} - \mathbf{x}^{[k]}$.

Разложим функцию $F_i(\mathbf{x})$ в окрестности точки $\mathbf{x}^{[k]}$ в ряд Тейлора:

$$F_i(\mathbf{x}) = F_i(\mathbf{x}^{[k]}) + \frac{\partial F_i(\mathbf{x}^{[k]})}{\partial x_1} (x_1 - x_1^{[k]}) + \frac{\partial F_i(\mathbf{x}^{[k]})}{\partial x_2} (x_2 - x_2^{[k]}) + R_2(\mathbf{x})$$

Подставив в это разложение точку $\hat{\mathbf{x}}$ получаем следующее:

$$F_i(\hat{\mathbf{x}}) = F_i(\mathbf{x}^{[k]}) + \frac{\partial F_i(\mathbf{x}^{[k]})}{\partial x_1} \Delta x_1^{[k]} + \frac{\partial F_i(\mathbf{x}^{[k]})}{\partial x_2} \Delta x_2^{[k]} + R_2(\mathbf{x}^{[k]})$$

Теперь запишем СНУ $\mathbf{F}(\hat{\mathbf{x}}) = 0$ с учетом предыдущего разложения:

$$\begin{cases} F_1(\mathbf{x}^{[k]}) + \frac{\partial F_1(\mathbf{x}^{[k]})}{\partial x_1} \Delta x_1^{[k]} + \frac{\partial F_1(\mathbf{x}^{[k]})}{\partial x_2} \Delta x_2^{[k]} + R_2^{F_1}(\mathbf{x}^{[k]}) = 0 \\ F_2(\mathbf{x}^{[k]}) + \frac{\partial F_2(\mathbf{x}^{[k]})}{\partial x_1} \Delta x_1^{[k]} + \frac{\partial F_2(\mathbf{x}^{[k]})}{\partial x_2} \Delta x_2^{[k]} + R_2^{F_2}(\mathbf{x}^{[k]}) = 0 \end{cases}$$

Или в матричном виде:

$$\mathbf{F}(\mathbf{x}^{[k]}) + \begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} \\ \frac{\partial F_2}{\partial x_1} & \frac{\partial F_2}{\partial x_2} \end{pmatrix} (\mathbf{x}^{[k]}) \cdot \Delta \mathbf{x}^{[k]} + \begin{pmatrix} R_2^{F_1}(\mathbf{x}^{[k]}) \\ R_2^{F_2}(\mathbf{x}^{[k]}) \end{pmatrix} = \mathbf{0}$$

$$\mathbf{F}(\mathbf{x}^{[k]}) + \mathbf{J}_F(\mathbf{x}^{[k]}) \cdot \Delta \mathbf{x}^{[k]} + \mathbf{R}_2(\mathbf{x}^{[k]}) = \mathbf{0}$$

$$\mathbf{J}_F(\mathbf{x}^{[k]}) \cdot \Delta \mathbf{x}^{[k]} = -\mathbf{F}(\mathbf{x}^{[k]}) - \mathbf{R}_2(\mathbf{x}^{[k]})$$

Где $\mathbf{J}_F(\mathbf{x})$ — матрица Якоби функции \mathbf{F} в точке \mathbf{x} .

Из этого уравнения можно найти $\Delta \mathbf{x}^{[k]}$, который прибавляется к текущему решению $\mathbf{x}^{[k]}$ чтобы найти $\hat{\mathbf{x}}$.

Мы не знаем как вычислять $\mathbf{R}_2(\mathbf{x})$, но можем им пренебречь, так как он стремится к нулю. Именно из-за этого пренебрежения мы не сразу получаем точное решение, а процесс становится итеративным. Итог:

$$\mathbf{J}_F(\mathbf{x}^{[k]}) \cdot \Delta \mathbf{x}^{[k]} = -\mathbf{F}(\mathbf{x}^{[k]})$$

Можно заметить, что это СЛАУ. Её можно легко решить известными методами.

4. Алгоритм

Весь метод заключается в итерационном процессе:

$$\mathbf{x}^{[k+1]} = \mathbf{x}^{[k]} + c (\mathbf{J}_F(\mathbf{x}^{[k]}))^{-1} \mathbf{F}(\mathbf{x}^{[k]}), \quad c \in (0, 1)$$

Data: $\varepsilon_1, \varepsilon_2, m, \mathbf{F}, \mathbf{x}^{[0]}$

Result: $\mathbf{x}^{[k]}$

```

1 begin
2    $\mathbf{x}^{[k]} \leftarrow \mathbf{x}^{[0]}$ 
3    $\beta \leftarrow 1$ 
4    $i \leftarrow 0$ 
5   while  $\|\mathbf{F}(\mathbf{x})\| > \varepsilon_1$  u  $\beta > \varepsilon_2$  u  $i < m$  do
6      $\mathbf{A} \leftarrow \mathbf{J}_F(\mathbf{x}^{[k]})$ 
7      $\mathbf{b} \leftarrow \mathbf{F}(\mathbf{x}^{[k]})$ 
8     Решить СЛАУ  $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$ , результат поместить в  $\mathbf{x}$ 
9      $\beta \leftarrow 1$ 
10    while  $\|\mathbf{F}(\mathbf{x}^{[k]} + \beta \mathbf{x})\| > \|\mathbf{F}(\mathbf{x}^{[k]})\|$  u  $\beta > \varepsilon_2$  do
11       $\beta \leftarrow 0.5\beta$ 
12    end
13     $\mathbf{x}^{[k]} \leftarrow \mathbf{x}^{[k]} + \beta \mathbf{x}$ 
14     $i \leftarrow i + 1$ 
15  end
16 end
```

Дополнительный цикл с переменной β создан для того, чтобы невязка всегда уменьшалась.

Заметьте, что данный алгоритм не зависит от размерности системы, а, следовательно, может быть применен к любой размерности задачи.

Замечание: в алгоритме используется матрица Якоби в известной точке, что означает, что знать её аналитическое выражение не нужно: её можно вычислять при помощи численного взятия производной в конкретной точке.

4.1. Одномерный случай

В одномерном случае это преобразуется к:

$$x^{[k+1]} = x^{[k]} + c \left(\frac{df(x^{[k]})}{dx} \right)^{-1} f(x^{[k]}) = x^{[k]} + c \frac{f(x^{[k]})}{f'(x^{[k]})}$$

5. Конкретный пример

Например, у нас имеется следующая СНУ:

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 = (r_1 + r)^2 \\ (x - x_2)^2 + (y - y_2)^2 = (r_1 + r)^2 \\ \frac{|ax + by + c|}{\sqrt{a^2 + b^2}} = r \end{cases}$$

Решением этой СНУ является окружность с центром в точке (x, y) и радиусом r , которая касается внешним образом окружностей (x_1, y_1, r_1) , (x_2, y_2, r_2) и прямой $ax + by + c = 0$.

Для этой системы получим:

$$\mathbf{F} = \begin{pmatrix} (x - x_1)^2 + (y - y_1)^2 - (r_1 + r)^2 \\ (x - x_2)^2 + (y - y_2)^2 - (r_1 + r)^2 \\ \frac{|ax + by + c|}{\sqrt{a^2 + b^2}} - r \end{pmatrix}, \mathbf{J}_{\mathbf{F}} = \begin{pmatrix} 2(x - x_1) & 2(y - y_1) & -2(r_1 + r) \\ 2(x - x_2) & 2(y - y_2) & -2(r_2 + r) \\ \frac{|a|}{\sqrt{a^2 + b^2}} & \frac{|b|}{\sqrt{a^2 + b^2}} & -1 \end{pmatrix}$$

Далее остается только применить алгоритм.

6. Когда размерность СНУ не соответствует размерности решения

Пусть дана СНУ:

$$\begin{cases} F_1(x_1, x_2, \dots, x_n) = 0 \\ F_2(x_1, x_2, \dots, x_n) = 0 \\ \dots \\ F_m(x_1, x_2, \dots, x_n) = 0 \end{cases}$$

Вышеописанный алгоритм работает только когда $m = n$, в случае же $m \neq n$ есть несколько методов приведения к системы квадратному виду.

Следующие методы показывают как надо модифицировать СНУ для конкретной точки, поэтому не получится их применить для того, чтобы один раз вычислить новую СНУ, новую СНУ придётся считать на каждой итерации заново.

6.1. $m < n$

Это означает, что решений, где СНУ оборачивается в ноль, множество.

Новая СНУ $\mathbf{G}(\mathbf{x})$ строится по алгоритму:

Data: \mathbf{F}, \mathbf{a}

Result: \mathbf{G}

```
1 begin
2    $\mathbf{A} \leftarrow \mathbf{J}_{\mathbf{F}}(\mathbf{a})$ 
3    $\mathbf{y} \leftarrow \left( \max_i |A_{i,1}|, \max_i |A_{i,2}|, \dots, \max_i |A_{i,n}| \right)$ 
4   Находятся такие  $i_j$ , что:  $y_{i_1} \geq y_{i_2} \geq \dots \geq y_{i_n}$ 
5   for  $j \leftarrow 1$  to  $n - m$  do
6      $G_{i_j} \leftarrow x_{i_j} - a_{i_j}$ 
7   end
8   for  $j \leftarrow n - m + 1$  to  $n$  do
9      $G_{i_j} \leftarrow F_{i_j}$ 
10  end
11 end
```

Обратите внимание, что в итоге $\mathbf{G}(\mathbf{x})$ является функцией $\mathbf{G} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ и для каждой различной точки \mathbf{a} функция \mathbf{G} различна.