1 B-Spline

B-Spline – функция вида

$$f(x) = \sum_j c_j B_{j,k}(x) \tag{1}$$

где $c_j \in V$ – коэффициенты сплайна (контрольные точки), а $B_{j,k} : K \to K$ – полиномы порядка k от x, определяемые рекурентным соотношением

$$\begin{aligned} B_{j,0} &= \begin{cases} 1 & x \in [t_j, t_{j+1}) \\ 0 \end{cases} \\ B_{j,r} &= \frac{x - t_j}{t_{j+r} - t_j} B_{j,r-1} + \frac{t_{j+r+1} - x}{t_{j+r+1} - t_{j+1}} B_{j+1,r-1}. \end{aligned}$$

Здесь $t_j \in K$ – узлы сплайна (неубывающая последовательность чисел). Узлы обычно задаются в виде последовательности

$$t = \left(\underbrace{t_0, t_0, ..., t_0}_{k-\text{раз}}, t_0, t_1, t_2, ..., t_n, \underbrace{t_n, t_n, ..., t_n}_{k-\text{раз}} \right) \in K^{n+1+2k},$$

а коэффициенты:

$$c = (c_{-k}, \ldots, c_{-2}, c_{-1}, c_0, c_1, \ldots, c_{n-1}) \in V^{n+k}.$$

Здесь V – векторное пространство над полем K. Это может быть множество матриц заданной размерности, множество вещественных чисел и т.д. Такой подход позволяет несколько упростить алгоритм вычисления значения сплайна и зафиксировать значения производных на концах.

Для конкретного x из всей суммы ненулевые значения $B_{j,k}(x) \neq 0$ имеют только $k+1$ функций: $B_i, B_{i-1}, \ldots B_{i-k}$ (i удовлетворяет $x \in [t_i, t_{i+1})$), поэтому для вычисления выражения (1) нет необходимости считать всю сумму, достаточно вычислить лишь частичную сумму из $k+1$ слагаемых.

Если для сплайна определены N узлов, то количество контрольных точек будет $N - k - 1$.

Почему контрольные точки $c_{-k}, \dots, c_{n-1}$? Потому, что максимально возможный индекс $i = n-1$. В этом случае значение сплайна

$$f = c_{n-1}B_{n-1} + c_{n-2}B_{n-2} + \dots + c_{n-k-1}B_{n-k-1}$$

будет зависеть от $c_{n-1}, \dots, c_{n-1-k}$.

1.1 Алгоритм вычисления значения сплайна в заданной точке

1. Для данного x найдём i, который удовлетворяет

   $$i \quad : \quad x \in [t_i, t_{i+1})\,,$$

   или если x вне интервала $[t_0, t_n]$, то

   $$i \quad = \quad \begin{cases} 0 & x < t_0 \\ n-1 & x \geq t_n \end{cases}$$

   В силу того, что последовательность монотонная, i будет единственным. Поиск будет иметь логарифмическую от длины t сложность.

2. Выберем $B_{i,0} = 1$ (все остальные $B_{j,0} = 0$, $j \neq i$), вычислим

   $$B_{i,1} = \frac{x - t_j}{t_{j+1} - t_j} B_{i,0} = \frac{x - t_j}{t_{j+1} - t_j}$$
   $$B_{i-1,1} = \frac{t_{i+1} - x}{t_{i+1} - t_i} B_{i,0} = \frac{t_{i+1} - x}{t_{i+1} - t_i}$$

   Затем $B_{j,2}$ для $j = i-2 \dots i$, затем $B_{j,3}$ для $j = i-3 \dots i$ и т.д. В итоге получим следующую таблицу

   $$D := \begin{pmatrix} 0 & & 0 & 0 & B_{i,0} \\ 0 & \dots & 0 & B_{i-1,1} & B_{i,1} \\ & & B_{i-2,2} & B_{i-1,2} & B_{i,2} \\ & \iddots & & & \vdots \\ B_{i-k,k} & B_{i-k+1,k} & \dots & & B_{i,k} \end{pmatrix}$$

где $B_{i-\alpha,r}$ вычисляется по формуле

$$B_{i-\alpha,r} = \frac{x - t_{i-\alpha}}{t_{i-\alpha+r} - t_{i-\alpha}} B_{i-\alpha,r-1} + \frac{t_{i-\alpha+r+1} - x}{t_{i-\alpha+r+1} - t_{i-\alpha+1}} B_{i-\alpha+1,r-1}$$

при $\alpha \in 1 \dots r-1$ и для $\alpha \in \{0, r\}$ по формулам

$$B_{i-r,r} = \frac{t_{i+1} - x}{t_{i+1} - t_{i-r+1}} B_{i-r+1,r-1}$$
$$B_{i,r} = \frac{x - t_i}{t_{i+r} - t_i} B_{i,r-1}$$

3. Последняя строка таблицы D содержит ненулевые функции $B_{j,k}\left(x\right)$. Для нахождения значения сплайна в точке x достаточно вычислить

$$f\left(x\right) = \sum_{\beta=0}^{k} c_{i-k+\beta} B_{i-k+\beta}\left(x\right)$$

(всего $k+1$ слагаемых). Стоит отметить, что функция $B_{i,k}$ определяется $k+1$ узлами $t_i \dots t_{i+k}$, а функция $B_{i-k,k}$ – узлами $t_{i-k+1} \dots t_{i+1}$. Значит, на значение функции $f\left(x\right)$ влияние оказывают узлы $t_{i-k+1} \dots t_{i+k}$ ($2k$ штук) и коэффициенты $c_{i-k} \dots c_i$ ($k+1$ штук).

1.2 Сложность алгоритма

Зачастую порядок сплайна выбирают $k = 3$ или $k = 5$. В этом случае для заполнения таблицы D необходимо вычислить $\frac{k(k+1)}{2}$ базисных функций – то есть 6 (для $k = 3$) или 15 (для $k = 5$) значений. Сами вычисления довольно простые – они содержат 5 сложений, 2 деления и 2 умножения. Вычисление самой функции требует вычисления $k+1$ умножения и k сложений над пространством V). В итоге

- $\frac{k(k+1)}{2} \cdot 9$ арифметический операций над полем K
- $2k+1$ арифметических операций над пространством V.

Алгоритмом Де-Бура, в свою очередь, требует $\sim \frac{(k+1)(k+2)}{2}$ арифметических операций над пространством V. В случае, если V – множество матриц большой размерности, это может быть существенно.

2 Вычисление производной сплайна

2.1 Производная первого порядка

Первая производная сплайна

$$\frac{df(x)}{dx} = \sum_j c_j \frac{dB_{j,k}(x)}{dx} \qquad (2)$$

определяется через производные базисных функций $\frac{dB_{j,k}}{dx}$, которые выражаются через базисные функции порядка $k-1$:

$$\frac{dB_{j,k}}{dx} = \frac{k}{t_{j+k}-t_j}B_{j,k-1} - \frac{k}{t_{j+k+1}-t_{j+1}}B_{j+1,k-1}.$$

Заметим, что значения функций $B_{i-k+1,k-1}\ldots B_{i,k-1}$ содержатся в $k-1$ строке таблицы D. Значит, для вычисления первой производной достаточно заполнить таблицу D (по аналогии с прердыдущим алгоритмом) до строки $k-1$, а строку k посчитать в соответствии с формулами

$$\frac{dB_{i-\alpha,k}}{dx} = \frac{k}{t_{i-\alpha+k}-t_{i-\alpha}}B_{i-\alpha,k-1} - \frac{k}{t_{i-\alpha+k+1}-t_{i-\alpha+1}}B_{i-\alpha+1,k-1}$$

для $\alpha \in 1\ldots k-1$ и

$$\frac{dB_{i-k,k}}{dx} = -\frac{k}{t_{i+1}-t_{i-k+1}}B_{i-k+1,k-1}$$

$$\frac{dB_{i,k}}{dx} = \frac{k}{t_{i+k}-t_i}B_{i,k-1}$$

для $\alpha \in \{k,0\}$ соответственно.

2.2 Производная порядка d

Для вычисления производной порядка d необходимо заполнить $k-d$ строк таблицы D, оставшиеся строки заполнить по правилу

$$B^{(r-k+d)}_{i-\alpha,r} = \frac{r}{t_{i-\alpha+r}-t_{i-\alpha}}B^{(r-k+d-1)}_{i-\alpha,r-1} - \frac{r}{t_{i-\alpha+r+1}-t_{i-\alpha+1}}B^{(r-k+d-1)}_{i-\alpha+1,r-1}$$

(для краткости введено обозначение $B^{(\sigma)}_{j,r} := \frac{dB^{\sigma}_{j,r}}{dx^{\sigma}}$) при $\alpha \in 1 \dots r-1$ и

$$\begin{aligned} B^{(r-k+d)}_{i-r,r} &= -\frac{r}{t_{i+1}-t_{i-r+1}} B^{(r-k+d-1)}_{i-r+1,r-1} \\ B^{(r-k+d)}_{i,r} &= \frac{r}{t_{i+r}-t_i} B^{(r-k+d-1)}_{i,r-1} \end{aligned}$$

при $\alpha \in \{r, 0\}$ соответственно. Получим таблицу вида

$$D^d := \begin{pmatrix} & 0 & & 0 & 0 & B_{i,0} \\ & 0 & \dots & 0 & B_{i-1,1} & B_{i,1} \\ & & & & & \\ & & B_{i-k+d,k-d} & B_{i-k+d+1,k-d} & \dots & B_{i,k-d} \\ & \frac{dB_{i-k+d-1,k-d+1}}{dx} & \frac{dB_{i-k+d,k-d+1}}{dx} & & \dots & \frac{dB_{i,k-d+1}}{dx} \\ & & & & \vdots & \\ \frac{d^d B_{i-k,k}}{dx^d} & \frac{d^d B_{i-k+1,k}}{dx^d} & & & \dots & \frac{d^d B_{i,k}}{dx^d} \end{pmatrix}$$

Используя эту таблицу легко вычислить значение производной функции – для этого достаточно вычислить сумму

$$\frac{d^d f(x)}{dx^d} = \sum_{\beta=0}^{k} c_{i-k+\beta} \frac{d^d B_{i-k+\beta}(x)}{dx^d}$$

3 Алгоритм Де Бура

3.1 Значение функции

Создаём массив из $k+1$ элементов $d_0, d_1, ..., d_k$, заполняя его значениями

$$d^0_j = c_{i-k+j} \quad \text{for}\, j = 0..k$$

Пробегаемся по массиву справа налево, перезаписывая $d^0_j \mapsto d^1_j$ по правилу

$$\begin{aligned} \alpha^r_j &= \frac{x - t_{i-k+j}}{t_{i+j+1-r} - t_{i-k+j}} \\ d^1_j &= \left(1-\alpha^1_j\right) d^0_{j-1} + \alpha^1_j d^0_j \end{aligned}$$

для всех $j = k..1$.

$$d^r_j = \left(1-\alpha^r_j\right) d^{r-1}_{j-1} + \alpha^r_j d^{r-1}_j$$

3.2 Производная

4 Сплайн апроксимация

Необходимо найти слайн (его коэффициенты c, узлы заданы изначально), который минимизирует критерий качества

$$\Phi = \sum_i \left[\left\| y_i - f\left(c, x_i\right) \right\|^2 + s \int_0^T \left\| f''\left(c, x\right) \right\|^2 dx \right]$$
$$c = \arg\min_c \Phi\left(c\right)$$

для заданного набора пар (x_i, y_i). Рассмотрим случай $x_i, y_i \in \mathbb{R}$.

4.1 Случай $s = 0$

Применяем МНК

$$\Phi = \sum_i \left(\sum_j c_j B_{j,k}\left(x_i\right) - y_i \right)^2$$
$$\frac{\partial \Phi}{\partial c_l} = 2 \sum_i \left(\sum_j c_j B_{j,k}\left(x_i\right) - y_i \right) B_{l,k}\left(x_i\right)$$

получаем СЛАУ

$$\sum_j \left[\sum_i B_{j,k}\left(x_i\right) B_{l,k}\left(x_i\right) \right] c_j = \sum_i y_i B_{l,k}\left(x_i\right)$$

в матричном виде

$$Ac = h,$$
$$A_{lj} = \sum_i B_{j,k}\left(x_i\right) B_{l,k}\left(x_i\right),$$
$$h_l = \sum_i y_i B_{l,k}\left(x_i\right).$$

4.2 Случай $s \neq 0$

$$\Phi = \sum_i \left(\sum_j c_j B_{j,k}(x_i) - y_i \right)^2 + s \int_0^T \left(f''(c,x) \right)^2 dx$$

$$\frac{\partial \Phi}{\partial c_l} = 2 \sum_i \left(\sum_j c_j B_{j,k}(x_i) - y_i \right) B_{l,k}(x_i) + 2s \sum_j \left[\int_0^T B''_{l,k}(x)\, B''_{j,k}(x)\, dx \right] c_j$$

получаем СЛУ

$$\sum_j \left[\sum_i B_{j,k}(x_i)\, B_{l,k}(x_i) \right] c_j + s \sum_j \left[\int_0^T B''_{l,k}(x)\, B''_{j,k}(x)\, dx \right] c_j = \sum_i y_i B_{l,k}(x_i)$$