

5. Bスプライン(B-Spline)

5.1 Bスプライン関数

(定義)

ノットベクトルと呼ばれる $(n+k)$ 個の非減少な実数列 $\mathbf{t} = (t_j)_{j=0}^{n+k-1}$ が与えられたとき、ノットベクトル \mathbf{t} に対するオーダ k の i 番目の B スプラインは次式で定義される：

$$(式2-1) \quad B_{i,k,t}(x) = (t_{i+k} - t_i)[t_i, \dots, t_{i+k}]_+(\cdot - x)_+^{k-1} \quad \text{for } i=0, \dots, n-1$$

ここで、 $[t_i, \dots, t_{i+k}]_+(\cdot - x)_+^{k-1}$ は関数 $(\cdot - x)_+^{k-1}$ の差分商(divided difference)。関数 $(\cdot - x)_+^{k-1}$ は切断幂関数(truncated power function)。

切断幂関数 $(\cdot - x)_+^{k-1}$ は次のように定義される。ここでは汎関数である差分商のパラメータとしているのでその変数は特に重要ではなく、プレースホルダー (\cdot) で表現している— x は B スプラインの変数であるが、切断幂関数の変数ではない。次の説明ではこれを表現するのに、変数 χ を用いている。

$$(式2-2) \quad (\cdot - x)_+ = (\chi - x)_+ = \begin{cases} \chi - x & \chi \geq x \\ 0 & \chi < x \end{cases}$$

$$(式2-3) \quad (\cdot - x)_+^{k-1} = ((\cdot - x)_+)^{k-1}$$

以上を用いて差分商 $[t_i, \dots, t_{i+k}]_+(\cdot - x)_+^{k-1}$ は次のように定義される。

$$(式2-4) \quad [t_i, \dots, t_{i+k}]_+(\cdot - x)_+^{k-1} = \frac{[t_{i+1}, \dots, t_{i+k}]_+(\cdot - x)_+^{k-1} - [t_i, \dots, t_{i+k-1}]_+(\cdot - x)_+^{k-1}}{t_{i+k} - t_i}$$

⋮

$$(式2-5) \quad [t_i, t_{i+1}]_+(\cdot - x)_+^{k-1} = \frac{(t_{i+1} - x)_+^{k-1} - (t_i - x)_+^{k-1}}{t_{i+1} - t_i}$$

$B_{i,k,t}(x)$ はノットベクトルを明示する必要がない場合は $B_{i,k}(x)$ と記述され、さらにオーダも明示する必要がない場合は $B_i(x)$ と簡略化して記述される。

5.2 Bスプラインの性質 1

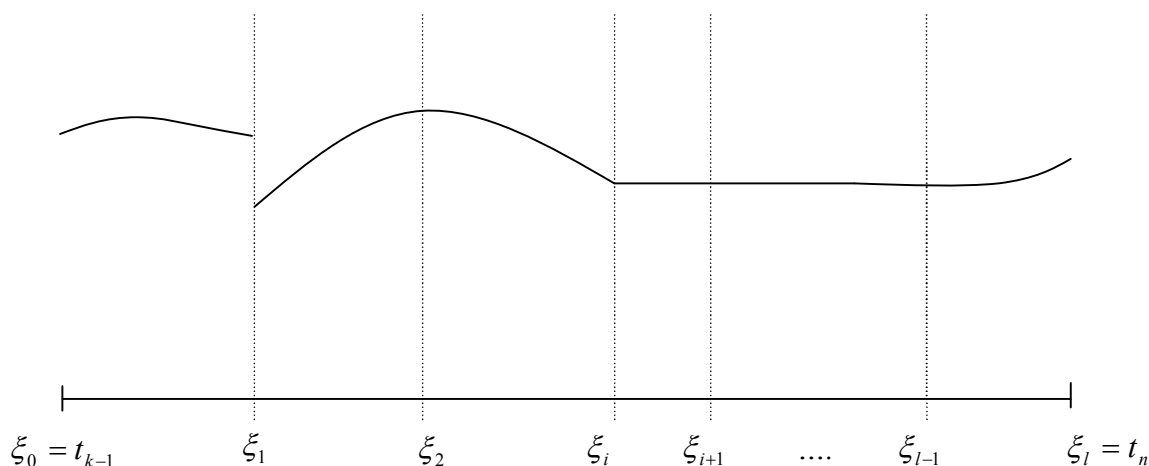
$0 \leq B_i(x) \leq 1$ である。

あるパラメータ値の B スプライン関数の総和は 1 である。すなわち、パラメータ値 x

($t_{k-1} \leq x < t_n$) が与えられたとき、 $\sum_{i=0}^{n-1} B_i(x) = 1$

B スプライン列 $\{B_i(x)\}_0^{n-1}$ は $t_\mu \leq x < t_{\mu+1}$ である x が与えられれば $i < \mu - k + 1$ 、または $\mu < i$ である $B_i(x)$ はすべて 0 (ゼロ) であり、 k 個の B スプライン関数 $\{B_i(x)\}_{i=\mu-k+1}^\mu$ だけが非ゼロの可能性はある。また、 $\sum_{i=\mu-k+1}^\mu B_i(x) = 1$ である。

5. 3 Curry, Shoenberg の定理(PP 表現の B スプラインによる表現)



単調増加な実数列 $\xi = (\xi_i)_0^l$ (分割点・break points と呼ばれる) と ξ 内部の各分割点 ξ_i $1 \leq i \leq l-1$ における連続性を表現する整数値 $\nu = (\nu_i)_1^{l-1}$ 、 $1 \leq \nu_i \leq k$ (ξ_i における連続性は $C^{(k-\nu_i-1)}$) for $1 \leq i \leq l-1$ が与えられ、 $n = k + \sum_{i=1}^{l-1} \nu_i$ とする。また、 $\mathbf{t} = (t_j)_{j=0}^{n+k-1}$ を次の条件を満たす非減少な実数列とする

$$t_0 \leq t_1 \leq \dots \leq t_{k-1} \leq \xi_0, \text{ かつ、 } \xi_l \leq t_n \leq \dots \leq t_{n+k-1}$$

$i=1, \dots, l-1$ に対して、 ξ_i はベクトル \mathbf{t} 内で ν_i 回現れる

このとき、オーダー k (または次数 $k-1$) で、ノットベクトル \mathbf{t} の B スプライン列 $(B_i)_{i=0}^{n-1}$ ($B_0, \dots,$

B_{n-1}) は実数区間 $[t_{k-1}, t_n]$ をパラメータ区間とする関数空間 $P_{k, \xi, \nu}$ の基底(basis)である。こ

こで、 $P_{k, \xi, \nu}$ はブレイクポイント $\xi = (\xi_i)_0^l$ 上の各点における連続性の定義を伴った、オーダー k の

区分多項式関数空間。 $P_{k,\xi,\nu}$ は、オーダ k 、B表現次元 n 、ノットベクトル \mathbf{t} から表現される区分多項式表現全体が構成する（関数空間としての）線形空間である。
別の言い方をすれば、

オーダ（多項式の次数を定義する）

分割点のパラメータ値

各分割点における連続性

の三つが定義された区分的多項式表現の関数は、すべて B スプライン列 $(B_i)_{i=0}^{n-1}$ の一次結合により一意的に表現可能である。

5.4 B 表現の定義

多項式のオーダを表現する k および（関数空間としての）線形空間の次元数を表現する

$$n = k + \sum_{i=1}^{l-1} \nu_i \quad (\text{B 表現次元と呼ばれる})$$

$$\text{ノットベクトル } \mathbf{t} = (t_j)_{j=0}^{n+k-1}$$

B スプライン列 $(B_i)_0^{n-1}$ に関する、関数 f の係数ベクトル $\alpha = (\alpha_i)_0^{n-1}$ （制御点列（Control Points）と呼ばれる）

関数 $f \in P_{k,\xi,\nu}$ に対する B 表現は上の各データから次のように定義される：

$$(式2-6) \quad f(x) = \sum_{i=0}^{n-1} \alpha_i B_i(x)$$

特に $k-1 \leq \mu < n$ である μ に対して $t_\mu \leq x < t_{\mu+1}$ であれば、

$$(式2-7) \quad f(x) = \sum_{i=0}^{n-1} \alpha_i B_i(x) = \sum_{i=\mu-k+1}^{\mu} \alpha_i B_i(x)$$

である。すなわち、B スプライン列 $(B_i)_0^{n-1}$ は $t_\mu \leq x < t_{\mu+1}$ であれば $i < \mu - k + 1$ 、または $\mu < i$

である $B_i(x)$ はすべて 0（ゼロ）となる。 k 個の B スプライン $(B_i)_{i=\mu-k+1}^{\mu}$ だけが非ゼロの可能性はある。

ノットベクトル $\mathbf{t} = (t_j)_{j=0}^{n+k-1}$ には B 表現次元 n 、オーダ k の情報が含まれるとして、B 表現はノットベクトルと制御点列の対 (\mathbf{t}, α) により表現されると考えてよい。

5.5 B スプラインの性質 2

B スプライン関数は強凸包 (strict convex hull) である。 x が $t_\mu \leq x < t_{\mu+1}$ を満たすようなパラメータ値であるとき、 $f(x) = \sum_{i=\mu-k+1}^{\mu} \alpha_i B_i(x)$ であり、 $i = \mu - (k-1), \mu - (k-2), \dots, \mu$ の k 個の制御点 α_i だけの一次結合で表わされる。

B 表現は変動減少 (Variation Diminishing) 特性を持つ。 B 表現は制御点列を直線で結んだ多角形 (定義多角形) の変動よりも変動が少ない。

5.6 B スプライン関数の評価 - 1

オーダ k のスプライン列 $(B_{i,k})_{i=0}^{n-1}$ は、 $t_\mu \leq x < t_{\mu+1}$ であるパラメータ値 x が与えられた時、すなわち、

パラメータ値 x が与えられ

x がノットベクトルの $t_\mu \leq x < t_{\mu+1}$ を満たすような、どこに位置するかを求め

そのノットベクトルの位置を μ とする時

k 個の非ゼロである (非ゼロの可能性のある) B スプライン列 $\{B_{i,k}(x)\}_{i=\mu-k+1}^{\mu}$ は、次のように帰納的に求めることができる。

$$(式2-8) \quad B_{\mu,1}(x) = \begin{cases} 1 & \text{for } t_\mu \leq x < t_{\mu+1} \\ 0 & \text{for } x < t_\mu \text{ or } t_{\mu+1} \leq x \end{cases}$$

$$(式2-9) \quad B_{i,j+1}(x) = \frac{x-t_i}{t_{i+j}-t_i} B_{i,j}(x) + \frac{t_{i+j+1}-x}{t_{i+j+1}-t_{i+1}} B_{i+1,j}(x)$$

$$i = \mu - j, \dots, \mu \text{ for } j=1, \dots, k-1$$

(式 2-9) を、 a を添え字として、書き直せば、

$$(式2-10) \quad B_{\mu-j+a,j+1}(x) = \frac{x-t_{\mu-j+a}}{t_{\mu+a}-t_{\mu-j+a}} B_{\mu-j+a,j}(x) + \frac{t_{\mu+1+a}-x}{t_{\mu+1+a}-t_{\mu+1-j+a}} B_{\mu-j+a+1,j}(x)$$

$$a = 0, \dots, j \text{ for } j=1, \dots, k-1$$

MGKnotVector::eval_coef()では(nderiv=0 として)、非ゼロの可能性のある k 個の係数を領域 coef[.]に

$$\text{coef}[a] = B_{\mu+1-k+a,k}(x) \text{ for } a=0, \dots, k-1$$

として求めている。

5.7 B スプライン関数の評価－2 (微分値の評価)

B スプライン関数の微分は $t_\mu \leq x < t_{\mu+1}$ であるとき

$$\begin{aligned} \text{(式2-11)} \quad f'(x) &= \left(\sum_{i=0}^{n-1} \alpha_i B_{i,k}(x) \right)' = \left(\sum_{i=\mu-k+1}^{\mu} \alpha_i B_{i,k}(x) \right)' = \sum_{i=\mu-k+2}^{\mu} (k-1) \frac{\alpha_i - \alpha_{i-1}}{t_{i+k-1} - t_i} B_{i,k-1}(x) \\ &= \sum_{i=\mu-k+1}^{\mu} \alpha_i (k-1) \left(\frac{B_{i,k-1}(x)}{t_{i+k-1} - t_i} - \frac{B_{i+1,k-1}(x)}{t_{i+k} - t_{i+1}} \right) \end{aligned}$$

すなわち

$$\text{(式2-12)} \quad B'_{i,k}(x) = (k-1) \left(\frac{B_{i,k-1}(x)}{t_{i+k-1} - t_i} - \frac{B_{i+1,k-1}(x)}{t_{i+k} - t_{i+1}} \right) \quad i = \mu - k + 1, \dots, \mu$$

一般に微分回数を nd ($nd \geq k$ の時、評価結果はすべて 0 であり、 $nd < k$ であるとしている) としたとき、

$$\text{(式2-13)} \quad B_{i,k-nd}^{(0)}(x) = B_{i,k-nd}(x), \quad i = \mu - (k - nd) + 1, \dots, \mu$$

$$\text{(式2-14)} \quad B_{i,k-nd+1}^{(1)}(x) = (k - nd) \left(\frac{B_{i,k-nd}^{(0)}(x)}{t_{i+k-nd} - t_i} - \frac{B_{i+1,k-nd}^{(0)}(x)}{t_{i+k-nd+1} - t_{i+1}} \right)$$

$$i = \mu - (k - nd), \dots, \mu$$

⋮

$$\text{(式2-15)} \quad B_{i,k}^{(nd)}(x) = (k-1) \left(\frac{B_{i,k-1}^{(nd-1)}(x)}{t_{i+k-1} - t_i} - \frac{B_{i+1,k-1}^{(nd-1)}(x)}{t_{i+k} - t_{i+1}} \right) \quad i = \mu - k + 1, \dots, \mu$$

(式 2-13) (式 2-14) (式 2-15) を、 a を添え字として書き直すと、

$$\text{(式2-16)} \quad B_{\mu+1-k+nd+a,k-nd}^{(0)}(x) = B_{\mu+1-k+nd+a,k-nd}(x), \quad a = 0, \dots, k - nd - 1$$

$$(式2-17) \quad B_{\mu-k+nd+a, k-nd+1}^{(1)}(x) = (k-nd) \left(\frac{B_{\mu-k+nd+a, k-nd}^{(0)}(x)}{t_{\mu+a} - t_{\mu-k+nd+a}} - \frac{B_{\mu+1-k+nd+a, k-nd}^{(0)}(x)}{t_{\mu+1+a} - t_{\mu+1-k+nd+a}} \right),$$

$$a = 0, \dots, k-nd$$

⋮

$$(式2-18) \quad B_{\mu+1-k+a, k}^{(nd)}(x) = (k-1) \left(\frac{B_{\mu+1-k+a, k-1}^{(nd-1)}(x)}{t_{\mu+a} - t_{\mu+1-k+a}} - \frac{B_{\mu+2-k+a, k-1}^{(nd-1)}(x)}{t_{\mu+1+a} - t_{\mu+2-k+a}} \right)$$

$$a = 0, \dots, k-1$$

MGKnotVector::eval_coef では (式 2-16) のオーダ(k・nd)の B スプライン関数の値を、(k・nd) 個領域 coef[.]に求めた後、(式 2-18) により、

$$\text{coef}[a] = B_{\mu+1-k+a, k}^{(nd)}(x) \quad \text{for } a=0, \dots, k-1$$

として求めている。

プログラムリスト1 Program code MGKnotVector::eval_coef

```
#define MAX_ORDER 20
// Function's return value id is the index of B-coefficients that
// should be multiplied to.
// coef[a] is for (id+a)-th B-Coefficients, 0<= a <=order-1 .
// Multiplication with coef should be done like:
//
// data=0.;
// for(int a=0; a<order; a++) data+=coef[a]*B_coef[id+a].
//
//left indicates whether left continuous(left=true), or right continuous
//(left=false) evaluation.
int MGKnotVector::eval_coef(
    double x, // Parameter value to evaluate
    double* coef, // coef's to mutiply are returned.
    unsigned nderiv, //order of derivative, =0 means position
    int left //Left continuous(left=true)
                //or right continuous(left=false).
)const{
    int k=m_order;
    int mu=locate(x,left); // k-1 <= mu <=bdim()-1 is guaranteed.
    int mup1=mu+1;
    int id=mup1-k;
    if(nderiv>=unsigned(k)){
```

```

    for(int a=0; a<k; ++a)
        coef[a]=0.f;
    return id;
}

double delta_buf[MAX_ORDER], deltar_buf[MAX_ORDER];
double* delta=delta_buf; double* deltar=deltar_buf;
if(k>MAX_ORDER){
    delta=new double[k]; deltar=new double[k];
}

//deltar[j] will be t[mu+j+1]-x
//delta[j] will be x-t[mu-j]    for j=0, ... ,k-1-nderiv
const MGKnotVector& t=*this;
int kmnd = k-nderiv;
coef[0] = 1.f;//=Bmu,1(x)
int j;
for(j=1; j<kmnd; j++){
    int jm1=j-1;
    int mup1mj = mup1-j;
    deltar[jm1] = t[mu+j]-x;
    delta[jm1] = x-t[mup1-j];
    double saved = 0.f;
    for(int a=0; a<j; ++a) {
        double term = coef[a]/(t[mup1+a]-t[mup1mj+a]);
        coef[a] = saved+deltar[a]*term;
        saved = delta[jm1-a]*term;
    }
    coef[j] = saved;
}
//Here coef[a] contains Bmu-j+a,j+1(x)
//(B-coefficient of (mu-j+a)th and of order j+1) for a=0, ... , j.
}
if(k>MAX_ORDER){
    delete[] delta; delete[] deltar;
}

//NOW B-SPLINE coef(a) 0<=a<=K-JDERIV(ORDER OF K-JDERIV)-1 ARE OBTAINED
//GET DERIVATIVE PART OF THE COEFFICIENTS.
//Here j == kmnd, coef[a] contains Bmu-k+nderiv+1+a,k-nderiv(x)
//(B-coefficient of (mu-k+nderiv+1+a)th and of order k-nderiv)
//for a=0, ... , k-nderiv-1.
for(j<k;j++){
    double fj = double(j);

```

```
int mup1mj = mup1-j;
double saved = 0.f;
for(int a=0; a<j; ++a){
    double term=coef[a]*fj/(t[mup1+a]-t[mup1mj+a]);
    coef[a] = saved-term;
    saved =term;
}
coef[j] = saved;
}

return id;
}
```

5.8 ベーゼ曲線 (Bezier Spline)