８．　曲線、曲面の求め方／フィッティング(Fitting)と近似（Approximation）

８．１ Shoenberg&Reinsch の平滑化関数 (Shoenberg&Reinsch Smoothing function)

この平滑化スプラインはSpringer-Verlag出版、Carl de Boor著、「A Practical Guide to Splines」に収録されている Fortran プログラム SMOOTH を改良したものである。

Schoenberg/Reinsch の平滑化スプライン

n 点のデータポイント(τ_i, g_i)と各データポイントにおけるエラー見積もりδg_i for i=0,⋯,n-1 が与えられて、これらの点のできるだけ近くを通り、できるだけ変化が少ない３次スプライン$f(t)$（(式 4-4）を参照）を求める。このひとつの方法として、Schoenberg/Reinsch の平滑化スプラインは次式のf_pを最小化するものとして定義される。

（式8-1）
$$f_p = p\sum_{i=0}^{n-1}\left(\frac{g_i - f(\tau_i)}{\delta g_i}\right)^2 + (1-p)\int_{\tau_0}^{\tau_{n-1}}(f^{(2)}(t))^2 dt$$

スプライン$f(t)$の、点群$\{g_i\}_{i=0}^{n-1}$からの離れ$S(f)$は次式で表現される。

（式8-2）
$$S(f) = \sum_{i=0}^{n-1}\left(\frac{g_i - f(\tau_i)}{\delta g_i}\right)^2$$

スプラインの変化は二次微分値の二乗の積分値として定量化されており、p＝０の時、最も変化が少なく、p=1 の時、データポイント(τ_i, g_i)を補間する通常の３次スプライン(Cubic Spline)となる。離れ$S(f)$は p=0 から p=1 へと単調に減少する。そこで、Schoenberg/Reinsch の平滑化スプラインは、許容される離れ量 S を与えて、$S(f)$=S となる p の値を求める。

Carl de Boor 著、「A Practical Guide to Splines」では$f(t)$として自由端末条件の３次 PP 表現を求めているが、自由端末条件のスプラインはあまり好ましいものとは言えない。これは、繰り返し自由端末条件のスプラインとして$f(t)$を求めると、始終点の形がくずれ、接続を前提とした曲線には使用できないためである。そこで、ここでは、平滑化対象の曲線の始終点における１次を求め、これを端末条件とし、端末条件を固定して平滑化を実施する機能を追加する。

$c_i = \dfrac{f_i^2}{2} = \dfrac{f_i^{(2)}(\tau_i)}{2}$ として、（式 4-4）と（式 4-8）から$f(t)$の２回微分は$f_i^{(2)}(t) = f_i^2 + f_i^3(t-\tau_i) = 2c_i + \dfrac{2(c_{i+1}-c_i)}{\Delta\tau_i}(t-\tau_i)$。

これから、 $\int_{\tau_i}^{\tau_{i+1}}(f_i^{(2)}(t))^2 dt = \dfrac{4\Delta\tau_i}{3}(c_i^2 + c_i c_{i+1} + c_{i+1}^2)$、従って（式 8-1）の$f_p$は次式となる：

（式8-3）　$$f_p = p\sum_{i=0}^{n-1}\left(\frac{g_i - f(\tau_i)}{\delta g_i}\right)^2 + \frac{4(1-p)}{3}\sum_{i=0}^{n-2}\Delta\tau_i(c_i^{\ 2} + c_i c_{i+1} + c_{i+1}^{\ 2})$$

以下、F，G を長さ n のベクタで、$f_i = f(\tau_i)$ として、$F^T = [\ f_0\ \ldots\ f_{n-1}\ ]$、$G^T = [\ g_0\ \ldots\ g_{n-1}\ ]$、$D$ を n*n の帯幅１の対角行列で、$D = \begin{bmatrix} \delta g_0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \delta g_{n-1} \end{bmatrix}$ と定義する。

（１） Shoenberg&Reinsch の平滑化関数：１次微分値の端末条件を固定

始終点における端末条件として、端末における１次微分値 s_0 と s_{n-1} が与えられ、B_1（B_1 の添え字の１は１次微分値の端末条件‐ Boundary Condition を表現している）を長さ n のベクタで $B_1^T = [\ s_0\ \ 0\ \ldots\ 0\ \ -s_{n-1}\ ]$ とする。Q_1^T を n*n の帯幅３の対角行列で次のように定義すると

$$Q_1^T = \begin{bmatrix} -1/\Delta\tau_0 & 1/\Delta\tau_0 & 0 & 0 & 0 \\ 1/\Delta\tau_0 & -(1/\Delta\tau_0 + 1/\Delta\tau_1) & 1/\Delta\tau_1 & 0 & 0 \\ & & \vdots & & \\ 0 & 1/\Delta\tau_{i-1} & -(1/\Delta\tau_{i-1} + 1/\Delta\tau_i) & 1/\Delta\tau_i & 0 \\ & & \vdots & & \\ 0 & \ldots & 1/\Delta\tau_{n-3} & -(1/\Delta\tau_{n-3} + 1/\Delta\tau_{n-2}) & 1/\Delta\tau_{n-2} \\ 0 & \ldots & 0 & 1/\Delta\tau_{n-2} & -1/\Delta\tau_{n-2} \end{bmatrix}$$

点 f_i を通り、端末条件として、始終点における１次微分値 s_0 と s_{n-1} が与えられている３次スプライン $f(t)$ は、（式 4-15）または、（式 4-16）から $R_2C = H_2 = 3(Q_1^T F - B_1)$ と記述できる。

（式8-4）　$R_2C = 3(Q_1^T F - B_1)$ または、$C = 3R_2^{\ -1}(Q_1^T F - B_1)$

ここで R_2、C、H_2 を４章から再掲すると：

$$R_2 = \begin{bmatrix} 2\Delta\tau_0 & \Delta\tau_0 & & & & & \\ \Delta\tau_0 & 2(\Delta\tau_0+\Delta\tau_1) & \Delta\tau_1 & & & & \\ & & \vdots & & & & \\ & \Delta\tau_{i-1} & 2(\Delta\tau_{i-1}+\Delta\tau_i) & \Delta\tau_i & & & \\ & & \vdots & & & & \\ & & \Delta\tau_{n-3} & 2(\Delta\tau_{n-3}+\Delta\tau_{n-2}) & \Delta\tau_{n-2} & & \\ & & & \Delta\tau_{n-2} & 2\Delta\tau_{n-2} & & \end{bmatrix}$$

$$C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_i \\ \vdots \\ c_{n-2} \\ c_{n-1} \end{bmatrix}、\quad H_2 = \begin{bmatrix} 3[\tau_0,\tau_1]g - 3s_0 \\ 3[\tau_1,\tau_2]g - 3[\tau_0,\tau_1]g \\ \vdots \\ 3[\tau_i,\tau_{i+1}]g - 3[\tau_{i-1},\tau_i]g \\ \vdots \\ 3[\tau_{n-2},\tau_{n-1}]g - 3[\tau_{n-3},\tau_{n-2}]g \\ 3s_{n-1} - 3[\tau_{n-2},\tau_{n-1}]g \end{bmatrix}$$

行列の形で（式 8-3）の f_p を書き直すと次のようになる：

（式8-5）　$$f_p = p(G-F)^T D^{-2}(G-F) + \frac{2}{3}(1-p)C^T R_2 C$$

これに（式 8-4）を適用すると

$$f_p = p(G-F)^T D^{-2}(G-F) + \frac{2}{3}(1-p)3(R_2^{-1}Q_1^T F - R_2^{-1}B_1)^T R_2 3(R_2^{-1}Q_1^T F - R_2^{-1}B_1)$$
$$= p(G-F)^T D^{-2}(G-F) + 6(1-p)(R_2^{-1}Q_1^T F - R_2^{-1}B_1)^T (Q_1^T F - B_1)$$

f_p は F が、$-2pD^{-2}(G-F) + 12(1-p)((R_2^{-1}Q_1^T)^T Q_1^T F - Q_1 R_2^{-1}B_1) = 0$ のとき最小となる。これを $C = 3R_2^{-1}(Q_1^T F - B_1)$ を使って書き直すと次式となる：

（式8-6）　$$pD^{-2}(G-F) = 2(1-p)Q_1 C$$

これに $3Q_1^T D^2$ を両辺に掛けると $3pQ_1^T(G-F) = 6(1-p)Q_1^T D^2 Q_1 C$ 。（式 8-4）から $3Q_1^T F = R_2 C + 3B_1$ であるので、C に関して整理すると結局（式 8-6）は次式となる：

（式8-7）　$$(6(1-p)Q_1^T D^2 Q_1 + pR_2)C = 3p(Q_1^T G - B_1)$$

$C = 3pU$ として $U^T = [\quad u_0 \quad \dots \quad u_{n-1} \quad]$ を定義して上式を書き直すと

（式8-8）　$$(6(1-p)Q_1^T D^2 Q_1 + pR_2)U = Q_1^T G - B_1$$

この式を解いて U を求め、PP 表現を求めることができる。また、U を用いて（式 8-6）を書き

直すと次式となる：

（式8-9）　$$F = G - \frac{2(1-p)}{p} D^2 Q_1 C = G - 6(1-p) D^2 Q_1 U$$

(式 4-8)はU を用いて書き直すと次式となり、(式 4-4) から３次スプラインの表現を求めることができる。

（式8-10）
$$\begin{cases} f_i^0 = f_i(\tau_i) = f_i \\ f_i^1 = f_i'(\tau_i) = \dfrac{\Delta f_i}{\Delta \tau_i} - 2p\Delta\tau_i u_i - p\Delta\tau_i u_{i+1} \\ f_i^2 = f_i^{(2)}(\tau_i) = 6pu_i \\ f_i^3 = f_i^{(3)}(\tau_i) = \dfrac{f_i^{(2)}(\tau_{i+1}) - f_i^{(2)}(\tau_i)}{\Delta\tau_i} = \dfrac{6p(u_{i+1} - u_i)}{\Delta\tau_i} \end{cases}$$

（２）　Shoenberg&Reinsch の平滑化関数：２次微分値=0 の端末条件（自由端末条件）

始終点における端末条件として、端末における２次微分値c_0=0 とc_{n-1}=0 とする。Q_2^T を (n-2) *n の帯幅３の対角行列、で次のように定義すると

$$Q_2^T = \begin{bmatrix} 1/\Delta\tau_0 & -(1/\Delta\tau_0 + 1/\Delta\tau_1) & 1/\Delta\tau_1 & 0 & 0 \\ 0 & 1/\Delta\tau_1 & -(1/\Delta\tau_1 + 1/\Delta\tau_2) & 1/\Delta\tau_2 & 0 \\ & & \vdots & & \\ 0 & 1/\Delta\tau_{i-1} & -(1/\Delta\tau_{i-1} + 1/\Delta\tau_i) & 1/\Delta\tau_i & 0 \\ & & \vdots & & \\ 0 & 1/\Delta\tau_{n-4} & -(1/\Delta\tau_{n-4} + 1/\Delta\tau_{n-3}) & 1/\Delta\tau_i & 0 \\ & & 1/\Delta\tau_{n-3} & -(1/\Delta\tau_{n-3} + 1/\Delta\tau_{n-2}) & 1/\Delta\tau_{n-2} \end{bmatrix}$$

点 f_i を通り、端末条件として、始終点における２次微分値＝０である３次スプライン $f(t)$ は、(式 4-17) または (式 4-18) から、$R_3 C_3 = H_3$ であり、$H_3 = 3Q_2^T F$ と記述できる。

（式8-11）　$R_3 C_3 = 3Q_2^T F$ または、$C_3 = R_3^{-1} H_3 = 3R_3^{-1} Q^T F$

R_3、C_3、H_3 を４章から再掲すると：

$$R_3 = \begin{bmatrix} 2(\Delta\tau_0+\Delta\tau_1) & \Delta\tau_1 & & & & & \\ \Delta\tau_1 & 2(\Delta\tau_1+\Delta\tau_2) & \Delta\tau_2 & & & & \\ & & \vdots & & & & \\ & & \Delta\tau_{i-1} & 2(\Delta\tau_{i-1}+\Delta\tau_i) & \Delta\tau_i & & \\ & & & \vdots & & & \\ & & & \Delta\tau_{n-4} & 2(\Delta\tau_{n-4}+\Delta\tau_{n-3}) & \Delta\tau_{n-3} \\ & & & & \Delta\tau_{n-3} & 2(\Delta\tau_{n-3}+\Delta\tau_{n-2}) \end{bmatrix}$$

$$C_3 = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_i \\ \vdots \\ c_{n-3} \\ c_{n-2} \end{bmatrix}、\quad H_3 = \begin{bmatrix} 3[\tau_1,\tau_2]g - 3[\tau_0,\tau_1]g - c_0\Delta\tau_0 \\ 3[\tau_2,\tau_3]g - 3[\tau_1,\tau_2]g \\ \vdots \\ 3[\tau_i,\tau_{i+1}]g - 3[\tau_{i-1},\tau_i]g \\ \vdots \\ 3[\tau_{n-3},\tau_{n-2}]g - 3[\tau_{n-4},\tau_{n-3}]g \\ 3[\tau_{n-2},\tau_{n-1}]g - 3[\tau_{n-3},\tau_{n-2}]g - c_{n-1}\Delta\tau_{n-2} \end{bmatrix}$$

（式 8-3）の f_p を行列の形で書き直すと次のようになる：

（式8-12）　$$f_p = p(G-F)^T D^{-2}(G-F) + \frac{2}{3}(1-p)C_3^T R_3 C_3$$

これに（式 8-11）を適用すると

$$f_p = p(G-F)^T D^{-2}(G-F) + \frac{2}{3}(1-p)(3R_3^{-1}Q_2^T F)^T R_3 (3R_3^{-1}Q_2^T F)$$
$$= p(G-F)^T D^{-2}(G-F) + 6(1-p)(R_3^{-1}Q_2^T F)^T R_3 (R_3^{-1}Q_2^T F)$$

f_p は F が、$-2pD^{-2}(G-F) + 12(1-p)(R_3^{-1}Q_2^T)^T R_3 (R_3^{-1}Q_2^T)F = 0$ のとき最小となる。これを $C_3 = 3R_3^{-1}Q_2^T G$ を使って書き直すと次式となる：

（式8-13）　$$pD^{-2}(G-F) = 2(1-p)Q_2 C_3$$

これに $3Q_2^T D^2$ を両辺に掛けると $3pQ_2^T(G-F) = 6(1-p)Q_2^T D^2 Q_2 C_3$。（式 8-11）から $3Q^T F = R_3 C_3$ であるので、C_3 に関して整理すると結局（式 8-13）は次式となる：

（式8-14）　$$(6(1-p)Q_2^T D^2 Q_2 + pR_3)C_3 = 3pQ_2^T G$$

$C_3 = 3pU_2$ として $U_2^T = [\; u_1 \;\; \ldots \;\; u_{n-2} \;]$ を定義して上式を書き直すと

（式8-15）　$$(6(1-p)Q_2^T D^2 Q_2 + pR_3)U_2 = Q_2^T G$$

この式を解いてU_2を求め、PP 表現を求めることができる。また、U_2を用いて（式 8-13）を書き直すと次式となる：

（式8-16）
$$F = G - \frac{2(1-p)}{p} D^2 Q_2 C_3 = G - 6(1-p) D^2 Q_2 U_2$$

(式 4-8)をU_2を用いて書き直すと次式となり、（式 4-4）から３次スプラインの表現を求めることができる。

（式8-17）
$$\begin{cases} f_i^0 = f_i(\tau_i) = f_i \\ f_i^1 = f_i'(\tau_i) = \dfrac{\Delta f_i}{\Delta \tau_i} - 2p\Delta\tau_i u_i - p\Delta\tau_i u_{i+1} \\ f_i^2 = f_i^{(2)}(\tau_i) = 6pu_i \\ f_i^3 = f_i^{(3)}(\tau_i) = \dfrac{f_i^{(2)}(\tau_{i+1}) - f_i^{(2)}(\tau_i)}{\Delta\tau_i} = \dfrac{6p(u_{i+1} - u_i)}{\Delta\tau_i} \end{cases}$$

８.２ Shoenberg&Reinsch の平滑化関数により B-表現を求める

（１）　１次微分値の端末条件

MGLBRep の次のコンストラクタは 8.1 の“Shoenberg&Reinsch の平滑化関数：１次微分値の端末条件”により PP 表現を求め、それを B スプライン化、すなわち B 表現を求めるものである。

```
//Smooth this line by Schoenberg and Reinsch smoothing function, given
//starting and end end condions, data points x, the weight dy at the data point,
//and a mean deviation max_deviation.
//If dy[i] becomes bigger, deviation at x(i) becomes bigger.
//n can be any number greater than or equal to 2.
//1st derivatives at the start and end points will be begin and end.
//Point data are approximations.
MGLBRep(
    const MGLBRepEndC& begin,//Begin end condition
    const MGLBRepEndC& end,     //End end conditoion
    const MGNDDArray& tau, //Data point abscissa
    const MGBPointSeq& g,    //Data point ordinates.
    const double* dg,           //Weights   at tau[i] for i=0,..., tau.length()-1.
    double max_deviation,    //Maximum of the weighted MEAN DISTANCE
                             //of the approximation
    bool dev_is_mean=true    //true if max_deviation is mean, false if max_deviation
    //is maximum deviation for each point.
    //when true, Sum((max_deviation/dg[i])**2) >= Sum(((g(i)-pout(i))/dg[i])**2),
    //when false, max_deviation>=maximum of (distance from g(i) to pout(i)).
```

);

MGLBRepEndC& begin, end は始終点の１次微分値を入力する。dg には δg_i を入力する。関数の元の点からの離れの許容度を max_deviation で入力する。（式 8-2）の $S(f)$ が、dev_is_mean=true のとき、$S(f)=\sum_{i=0}^{n-1}\left(\frac{g_i - f(\tau_i)}{\delta g_i}\right)^2 <= \sum_{i=0}^{n-1}\left(\frac{\max\_deviation}{\delta g_i}\right)^2$ となるよう、また、dev_is_mean=false のとき、$g_i - f(\tau_i)$ の最大値<=max_deviation となるよう制御される。

（２）　自由端末条件（端末の２次微分値＝０とする）

次のコンストラクタは自由端末条件による Shoenberg&Reinsch の平滑化関数により B-表現を求めるものである。ここでは入力の点の移動後の点を出力することができる。

```
// Construct Line B-rep of order 4 by Schoenberg and Reinsch smoothing
// function from point data with approximation weights and max deviation.
//(tau(i), points(i,.),dp[i]) are one pair.
//Approximation is done for the following expression to hold:
//Sum((max_deviation/dp[i])**2) >=   Sum(((points(i)-pout(i))/dp[i])**2),
//where pout are smoothed data.
//If dp[i] becomes bigger, deviation at tau(i) becomes bigger.
//***End conditions are free end condition.***
MGLBRep(
    const MGNDDArray& tau,        //Data point abscissa
    const MGBPointSeq& points,    //Point seq data
    const double* dg,             //Weights for each points
    double max_deviation,         //Maximum of the weighted mean
                                  //distance of the approximation
    MGBPointSeq* pout=0);    //new points corresponding to points will be output,
                //if pout specified.
```

dg には δg_i を入力する。関数の元の点からの離れの許容度を max_deviation で入力する。（式 8-2）の $S(f)$ が、$S(f)=\sum_{i=0}^{n-1}\left(\frac{g_i - f(\tau_i)}{\delta g_i}\right)^2 <= \sum_{i=0}^{n-1}\left(\frac{\max\_deviation}{\delta g_i}\right)^2$ となるよう制御される。