标准曲线拟合算法分析报告

**报告制作：崔志雷**

**日期：2013年7月20日**

**摘要**：

标准曲线拟合算法是化学发光仪上位机软件的理论基础和核心功能，然而旧版软件所采用的拟合算法是参考其他软件代码编写的，由于缺乏严谨的数学理论依据，导致旧版软件出现的计算bug极其难以调试。

三次样条插值算法和四参数Logistic模型是化学发光免疫分析技术中最常用的标准曲线计算方法。本报告不分析两种模型的优劣性，只是对这两种模型的数值计算过程进行详细分析和描述。

**三次样条插值算法:**

1. **模型简介**

三次样条插值是一种分段多项式插值逼近方法，在相邻的各对节点之间使用三次多项式去逼近，并且假定在各节点上是二阶平滑的。

三次样条函数的定义：

给定在上定义的函数和一组节点的三次样条插值是满足下列条件的函数：

1. 在子区间上是三次多项式，记为
2. ，
3. ，
4. ，
5. ，
6. 自由边界条件：

或固支边界条件： 和

本软件采用自由样条函数拟合标准曲线，即满足上述f条件的自由边界条件。样条曲线以浓度的对数为x值，以发光值为y值。

1. **计算过程推导**

我们利用的二阶导数值.由于在区间上是三次多项式，故在上是线性函数，可表示为

对积分两次并利用=及=，可定出积分常数，于是得三次样条表达式：

= 0，1，…，.

其中表示第j个插值节点的x坐标，

表示第j个插值节点的y坐标，

表示样条曲线在第j点处的二阶导数值,是待求的未知数

= -

对求导得：

由此可求得

利用

其中 = ,

= 1 - =

= 　=

自由边界条件：

= 0

= 0

= 0

= 0

以上可得求解 ~ 的矩阵形式方程组

=

上述对角占优的三角线方程组可用追赶法求解：

求解过程

1. （中间变量） 递推：
2. （中间变量） 递推：

1. (方程解) 递推：

~ 即得到三次样条函数

1. **详细算法描述（c++伪代码）**

声明自然样条类：

class 样条类

{

public :

样条类(数组 输入插值节点[]); //构造对象时,需要输入插值节点

double GetYByX(double x) const;//根据给定的x坐标值,取得该曲线上的y坐

double GetXByY(double y) const;//根据给定的y坐标值,逆推x值,即求方程S(x)-y=0的根 int 拐点数() const; //取得样条曲线拐点数

private:

; //曲线方程系数数组,保存~

成员插值节点[]; //保存插值节点

private:

内联函数 h(j); //

内联函数μ(j); //

内联函数λ(j); //

内联函数 d(j); //

内联函数 ; //样条函数表达式

内联函数 ; //

}

实现样条函数

样条类:: 样条类(数组 输入插值节点[])

{

1.成员插值节点[]← 输入插值节点[]；

2.按x值从小到大排序；

3.定义临时数组变量，Y[]

4.；

for(i = 1;i<节点个数；i++)

{

}

5.

for(i = 1;i<节点个数；i++)

{

}

6. = 0;

for(i = n-1;i>0；i--)

{

}

}

double 样条类:: GetYByX(double x)

{

return S(x)

}

double 样条类:: GetXByY(double y)

{

// 用牛顿下山迭代法逆推x值

1. 定义迭代精度；
2. 定义临时变量x = ；
3. do

{

* 1. 定义临时变量p = x;
  2. 定义临时变量下山因子c =1.0;

do

{

x p –(S(p) - y)/;

c c / 2;

}while( |S(x) – y| > |S(p) – y|)

}while(|x – p|> )

1. return x;

}

int 样条类:: 拐点数()

{

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// 拐点处的二阶导数值为0，

// 插值函数在每段函数上的二阶导函数是一条直线

// 如果此段两个端点的二阶导数符号互异，则必有拐点

// 对于自由样条曲线，因为两端点都是二阶导数为零的，

// 所以第一段函数区间和最后一段函数区间不可能有拐点

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int 拐点数 = 0;

for(int i=1; i < 节点个数 - 2;i++)

{ if( == 0) // 如果 == 0 说明此点是拐点

拐点数++;

if(\* < 0) 拐点数++; }

return 拐点数;

}

**四参数Logistic模型:**

1. **Logistic模型简介**

Logistic回归模型是离散选择模型之一，是对因变量为定性变量的回归分析。所谓Logistic模型，就是人们想为两分类的应变量作一个回归方程出来，可概率的取值在0~1之间，回归方程的应变量取值可是在实数集中，直接做会出现0~1范围之外的不可能结果，因此可以将概率做了一个Logit变换，这样取值区间就变成了整个实数集，作出来的结果就不会有问题了，从而该方法就被叫做了Logistic回归。

对于发生概率为P的二分类事件，其Logistic回归模型为：

Logistic() =

1. **标准曲线的四参数Logistic模型简介**

设为标准品发光值与最大发光值之比值，为最大浓度的发光值，为零标准品的发光值，为标准品浓度，为常数，相当于发光值为

或

1. **非线性最小二乘曲线拟合算法**

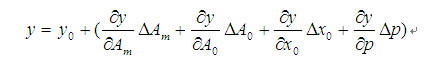
非线性拟合问题是指拟合函数模型关于拟合系数C是非线性关系

将标准点的x，y值代入拟合模型可得非线性方程组。

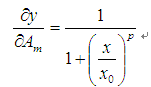
非线性最小二乘问题是无法直接求解的，可将用泰勒公式展开替代，局部线性化。那么非线性最小二乘问题就化成了线性最小二乘问题来求解。

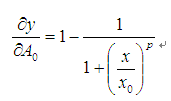
给定参数矩阵的初值,，代入上述用泰勒公式展开的线性方程组，用最小二乘法求得**C**的增量，然后把与原**C**叠加，代入继续运算。如此反复迭代，最后求得**C**的最佳解。这就是高斯—牛顿迭代法的原理。

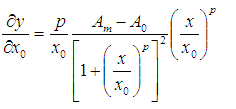
对于四参数模型方程：

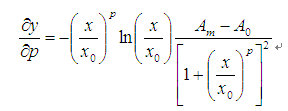


其中









令, 即表示迭代k此后的参数矩阵

令, 即表示迭代k此后的雅克比矩阵

令 ，即表示第k迭代的增量矩阵

令，即表示y值拟合误差矩阵

于是求最小二乘线性方程组就是求以下方程组的解：

（3.1）

上述3.1表示的超定方程组的最小二乘解用矩阵形式表示为：

由于可能为病态矩阵，会导致计算逆矩阵严重偏差。我们可以通过正则化手段来改善的形态。寻找一个合适阻尼因子λ，令

这就是**LM**（**Levenberg-Marquart）**算法

1. **详细算法描述（c++伪代码）**

class 矩阵类

{

说明：本类定义和实现矩阵的各种运算操作，如矩阵四则运算，矩阵求行列式，求逆，转置，求秩等等。实现算法本处不做介绍。

}

class 四参数拟合类

{

public :

四参数算法类(数组 输入拟合节点[]); //构造对象时,需要输入节点

double GetYByX(double x) const;//根据给定的x坐标值,取得该曲线上的y坐

double GetXByY(double y) const;//根据给定的y坐标值推x值

int 相关系数() const; //取得拟合点与测量点的相关数

private:

;

;

;

;

成员拟合节点[]; //保存插值节点

private:

内联函数 (); //Am的偏导数

内联函数 (); //A0的偏导数

内联函数 (); //x0的偏导数

内联函数 (); //p的偏导数

内联函数 矩阵类 ;

内联函数 double ();

内联函数 double ();

}

样条类:: 样条类(数组 输入插值节点[])

{

1. 成员插值节点[]← 输入插值节点[]；
2. 按x值从小到大排序；
3. 选定参数的初始值
4. 定义临时矩阵变量

//

//

//

//

1. 其他定义临时变量

double 阻尼系数

double终止常数

double增长因子

double 新y值残差 =(),旧y值残差 = 新y值残差；

1. LM法迭代循环

while（迭代次数 < 最大迭代次数）

{

6.1 ；

6.2 定义临时变量，保存叠加前的四参数和拟合误差：

OldAm = ，OldA0 = ，OldX0 = ，OldAm = ，Old；

6.3 **=**  //

6.4新y值残差 = ()；

If(新y值残差 == ())

{

增长因子 //

}

6.5雅克比矩阵；

6.6

If(新y值残差 < 旧y值残差)

{

If(拟合误差矩阵的2-范数值 < 终止常数)

return;

阻尼因子; //

}

Else

{

恢复6.2保存的到旧参数和旧的拟合误差；

阻尼因子； //

}

6.7 迭代次数++；

}

**参考文献：**

[1]李庆扬.数值分析（第五版）.北京：清华大学出版社.2008-12

[2]张明等.应用数值分析（第四版）.北京：石油工业出版社.2012-8

[3杨利国等.酶免疫测定技术.江苏：南京大学出版社.1998-7

[4]庄楚强等.应用数理统计（第三版）.广东：华南理工大学.2006-2

[5]Richard L.Burden等著.冯烟利等译.数值分析（第七版）.北京：高等教育出版社

[6]王腾蛟.基于Levenberg-Marquardt算法图像拼接研究.长沙：国防科技大学

[7]魏木生.广义最小二乘问题的理论和计算.北京：科学出版社.2006-9