最优化方法

授课教师：徐海涛

2024年秋

第二章 无约束下的一维极小化问题

2.6 多项式拟合法

- 如果不确定性区间比较小，且函数比较平滑，则可利用多项式进行拟合。
- 拟合多项式的极小点位于不确定性区间中，是函数极小点的一个很好的近似。

❶如果函数只有在某些离散点上的值已知，比如在三个点上的值已知，则可利用二阶多项式进行拟合。

❷如果有两个点的函数值及其导数已知，则可以利用三阶多项式进行拟合。

- 基本思路就是从初始的不确定性区间出发，利用多项式信息不断压缩区间长度。结合试探法和分割法，多项式拟合方法能够获得非常稳健的性能。

2.6.1 二阶多项式拟合算法

- 获得一个不确定性区间，采用的是三点模式。三个点为$[x_1, x_2, x_3]$，对应的函数值分别为$[f_1, f_2, f_3]$，且有$x_1 < x_2 < x_3$，$f_2 \le \min(f_1, f_3)$
- 利用一个经过这三个点的二阶多项式对函数进行拟合。称为拉格朗日多项式

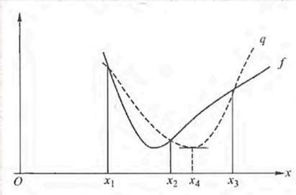


图 2.12 三点模式下的二次多项式拟合

$$q(x) = f_1\frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + f_2\frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + f_3\frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \quad (2.22)$$

2.6.1 二阶多项式拟合算法

$$q(x)=f_1\frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}+f_2\frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}+f_3\frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \qquad (2.22)$$

令$q(x)$ 的极小点为x_4，该点处对应的导数为零，即$dq/dx=0$，

$$x_4=\frac{\frac{f_1}{A}(x_2+x_3)+\frac{f_2}{B}(x_1+x_3)+\frac{f_3}{C}(x_1+x_2)}{2\left(\frac{f_1}{A}+\frac{f_2}{B}+\frac{f_3}{C}\right)}$$

$A=(x_1-x_2)(x_1-x_3)$，$B=(x_2-x_1)(x_2-x_3)$

$C=(x_3-x_1)(x_3-x_2)$

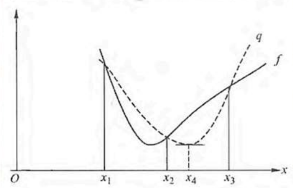


图 2.12　三点模式下的二次多项式拟合

2.6.1 二阶多项式拟合算法

计算f_4处的函数值$f(x_4)$之后，与f_2进行比对，利用图2.8所示的方法可得新的不确定性区间

$$[x_1,x_2,x_3]_{\text{new}}=\begin{cases}[x_1,x_2,x_4] & \text{当}\quad x_4>x_2\ \text{且}\ f(x_4)\geqslant f_2\\ [x_2,x_4,x_3] & \text{当}\quad x_4>x_2\ \text{且}\ f(x_4)\quad f_2\\ [x_4,x_2,x_3] & \text{当}\quad x_4<x_2\ \text{且}\ f(x_4)\geqslant f_2\\ [x_1,x_4,x_2] & \text{当}\quad x_4<x_2\ \text{且}\ f(x_4)<f_2\end{cases}$$

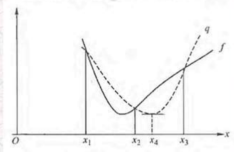


图 2.12　三点模式下的二次多项式拟合

- 重复以上过程，直到满足停止规则。
- 但是，这种简单的二阶多项式拟合方法特别容易失败
- 在算法中必须增加保护措施

2.6.1 二阶多项式拟合中的保护措施

为了保证二次多项式的稳健性，应该增加以下三种保护措施：

（1）点 x_4 必须位于区间$[x_1,x_3]$中。理论上说，按照前面给出的前提条件，这一点总是能够满足，但是在压缩过程中必须时刻注意确认。

（2）点 x_4 不能与其余三个点的距离过近，否则后续拟合过程将出现病态。如果是在 n 维优化问题（见第 3 章）求解过程中调用了多项式拟合方法，那么这一点应该尤其注意。解决的办法为增加一个尺度变量 δ，使得点 x_4 与距离过近的点之间的距离增加 δ。具体的实现方式为

$$\text{如果}\begin{cases}|x_4-x_1|<\delta,\ \text{令}\ x_4=x_1+\delta\\ |x_4-x_3|<\delta,\ \text{令}\ x_4=x_3-\delta\\ |x_4-x_2|<\delta,\ \text{且}\ x_2>0.5^*(x_1+x_3),\ \text{令}\ x_4=x_2-\delta\\ |x_4-x_2|<\delta,\ \text{且}\ x_2\leqslant 0.5^*(x_1+x_3),\ \text{令}\ x_4=x_2+\delta\end{cases}$$

$\delta>0$ 是一个足够小的数。比如，可令 $\delta=s\min\{|x_3-x_2|,|x_2-x_1|\}$，$0<s<1/2$（程序 Quadfit 采用了这种方法，参数 s 在程序中对应为 sfrac，默认为 1/8）。

2.6.1 二阶多项式拟合中的保护措施

（3）应该避免收敛速度过慢。这可以通过对比相邻两次的区间长度进行判断。图 2.13 演示了收敛速度过慢的情况。原来的三个点为$[x_1,x_2,x_3]$，采用二阶多项式进行拟合，并求极小点，可得 x_4。如果$f(x_4)<f(x_2)$，则新的三个点为$[x_2,x_4,x_3]$。但是，由于 x_2 和 x_4 非常接近，故$L_{\text{new}}/L_{\text{old}}$接近于 1。比如，如果这个比值为 0.95，则意味着这一次迭代只压缩了原来区间的 5%。

前面已经提到过，黄金分割法可以按照比例 $\tau=0.618$ 对区间进行压缩。因此，如果在某次迭代中，有 $L_{\text{new}}/L_{\text{old}}>\tau$，可以采用黄金分割法替代该步骤，如下所示：

如果$x_2\leqslant(x_1+x_3)/2$则

$x_4=x_2+(1-\tau)(x_3-x_2)$;

否则

$x_4=x_3-(1-\tau)(x_2-x_1)$;

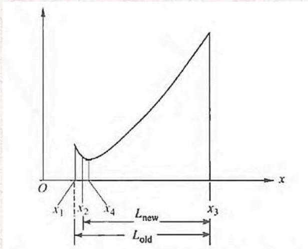


图 2.13　多项式拟合算法中收敛速度慢的问题

2.6.1 二阶多项式拟合中的保护措施

停止规则

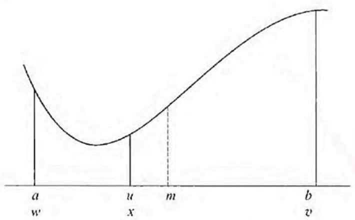
（i）超出了预先设定的计算函数值的总次数。

（ii）基于 x 的值：如果区间长度足够小，即小于等于 $xtol=\sqrt{\varepsilon_m}(1+\mathrm{abs}(x_2))$，迭代停止；如果相邻两次迭代，有 $\mathrm{abs}(x_3-x_1)\leqslant 2.0*xtol$，迭代停止。

（iii）基于函数值 f：如果在一次迭代中，区间内函数值的平均值变化足够小，即小于等于 $ftol=\sqrt{\varepsilon_m}(1+\mathrm{abs}(f_2))$；如果相邻两次迭代中，有 $\mathrm{abs}(\bar{f}_{new}-\bar{f}_{old})\leqslant ftol$，则迭代停止。

2.6.1 布伦特二阶多项式拟合 - 分割算法

- 布伦特算法的基本思路是如果二阶多项式拟合适用，则利用其进行拟合，并取多项式极小点作为新点。
- 二阶多项式是否适用，视其是否满足某个规则。如果二阶多项式不适用，则用黄金分割法替代



$$m=\frac{a+b}{2}$$

$$e=\begin{cases} b-x & 当\ x<m \\ a-x & 当\ x\geqslant m \end{cases}$$

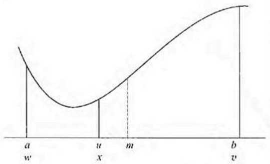
图 2.14 布伦特的多项式拟合算法

2.6.1 布伦特二阶多项式拟合 - 分割算法

- 从一个包含极小点的不确定性区间出发，在每次迭代中，都同时保留五个点a,b,x,v,w。
- 这些点之间可能存在重合
- 点a和b形成的区间包含极小点，x 对应的函数值最小，w 对应的函数值仅比x对应的函数值大，v是上一次迭代中的w（函数值最大）。u表示最新执行完函数值计算的点。
- 无论x,v和w是否存在重合点，都以它们为基础构建二阶多项式，对应的极小点为

$$q=x-0.5\frac{(x-w)^2[f(x)-f(v)]-(x-v)^2[f(x)-f(w)]}{(x-w)[f(x)-f(v)]-(x-v)[f(x)-f(w)]} \tag{2.24}$$

极小点 q 落入区间 $[a,b]$ 的可能性很大。如果落入该区间中，说明二阶多项式拟合方法没有出现问题，保留点 q；否则，利用黄金分割法引入一个新点，记为 u。在原来的五个点 a,b,x,v,w 中引入点 u，构造出一组新的 a,b,x,v,w，保证算法继续进行。



$$m=\frac{a+b}{2}$$

$$e=\begin{cases} b-x & 当\ x<m \\ a-x & 当\ x\geqslant m \end{cases}$$

图 2.14 布伦特的多项式拟合算法

2.6.1 布伦特二阶多项式拟合 - 分割算法

极小点求解的布伦特算法

1. 初始区间为三点模式：a,b 和 x，a 和 b 构成区间，x 是最小点。
2. 初始化点 w 和点 v，将其与点 x 重合。
3. 如果点 x，w 和 v 互不重合，则转向第 5 步。
4. 确定区间 $x-a$ 和区间 $x-b$ 中较大的一个，利用黄金分割法在较大的区间中计算点 u。转向第 7 步。
5. 对点 x，w 和 v 进行二阶多项式拟合，如果多项式的极小点落在区间 $a-b$ 中，则将极小点作为 u。
6. 如果点 u 与 a,b 或 x 特别接近，则将 u 调整到区间 $x-a$ 或 $x-b$ 中较大的一个，使得 u 与 x 的距离至少为 tol，tol 是基于机器精度确定的。
7. 计算点 u 处的函数值。
8. 从点 a,b,x,w,v 和 u 中，确定新的 a,b,x,w,v。
9. 如果区间 $x-a$ 或 $x-b$ 中较大的一个小于 $2*\mathrm{tol}$，则意味着迭代收敛，算法结束；否则，转向第 3 步。

布伦特的算法是以区间长度作为是否收敛的判别标准。在程序 BRENTGLD 中，还引入了基于函数值的收敛判别准则。

2.6.2 求极小点的三阶多项式拟合方法

如果函数的导数可用，则可利用三阶多项式进行拟合，构造两点模式区间

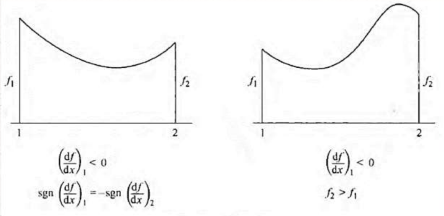


图 2.15 两点模式

2.6.2 求极小点的三阶多项式拟合方法

为了便于后续处理，先将区间归一化

$$\xi = \frac{x - x_1}{x_2 - x_1}$$

记 $f' = \frac{df}{d\xi}$，有

$$f' = \frac{df}{d\xi} = (x_2 - x_1)\frac{df}{dx}$$

拟合的三阶多项式可表示为

$$f = a\xi^3 + b\xi^2 + f_1'\xi + f_1$$

$$a = f_2' + f_1' - 2(f_2 - f_1)$$

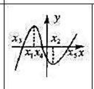
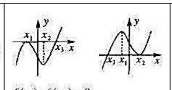
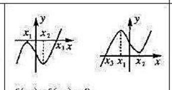
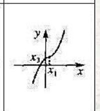
$$b = 3(f_2 - f_1) - f_2' - 2f_1'$$

由于多项式的极小点满足 $\frac{df}{d\xi} = 0$ 且 $\frac{d^2f}{d\xi^2} > 0$，故可得极小点：

$$\xi_p = \frac{-b + \sqrt{b^2 - 3af_1'}}{3a}$$

注意，结合三次函数的图像，根据a和b的取值，来确定是否有极小点

2.6.2 求极小点的三阶多项式拟合方法

	$b^2-3ac>0$			$b^2-3ac\le 0$
图像	 $f(x_1)\cdot f(x_2)<0$	 $f(x_1)\cdot f(x_2)=0$	 $f(x_1)\cdot f(x_2)>0$	
$f(x)=0$ 根的个数	三实根	两实根	一实根	一实根
与 x 轴的交点	三交点	两交点	一交点	一交点
单调性	在 $(-\infty,x_1)$ 和 $(x_2,+\infty)$ 上为增函数.，在 (x_1,x_2) 上为减函数			在 R 上为增函数
极值	有两个极值，一个极大值 $f(x_1)$，一个极小值 $f(x_2)$			无极值

2.6.2 求极小点的三阶多项式拟合方法

由于多项式的极小点满足 $\frac{df}{d\xi} = 0$ 且 $\frac{d^2f}{d\xi^2} > 0$，故可得极小点：

$$\xi_p = \frac{-b + \sqrt{b^2 - 3af_1'}}{3a}$$

- ◆ 求得极小点之后，按照 $x_p = x_1 + (x_2 - x_1)\xi_p$ 的方式引入一个新点，并保证新点与区间的两个端点之间至少保持一定的距离。
- ◆ 如果 $\text{sgn}(f') = \text{sgn}(f'_1)$，则新点命名为点1，否则命名为点2。
- ◆ 通过计算区间 $|x_2 - x_1|$ 的长度或者根据两个相邻的函数值，即 $0.5(f_1 + f_2)$ 的大小来判断是否收敛。

2.8 利用MATLAB 求函数极小点

在MATLAB 中，单变量函数的极小点求解函数为fminbnd

对于函数 $f=2-2x+e^{x}$ ，初始区间为[0，2]，首先创建一个M文件，命名为getfun.m

```
function [f] = getfun(x)
f = 2 - 2*x + exp(x);
```

```
[xopt, fopt, ifl, out] = fminbnd('getfun',0,2)
```

在MATLAB中，fminbnd 函数用于寻找单变量函数在指定区间上的局部最小值。该函数采用的是黄金分割搜索（Golden Section Search）方法，结合抛物线插值（parabolic interpolation）来更精确地逼近最小值点

明理，精工，笃学，致远　17

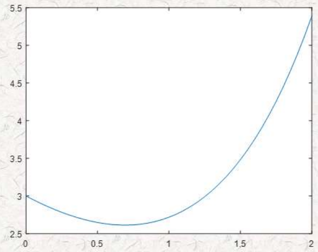
2.8 利用MATLAB 求函数极小点

```
xopt =

  0.6932

fopt =

  2.6137
```



明理，精工，笃学，致远　18

作业

必做：P2.2，P2.4，P2.5，P2.7

选做：P2.11，P2.12，P2.13

提示：

计算题可先采用fminbnd函数求解，先获取最小值，然后再思考计算条件和方法

明理，精工，笃学，致远　19

作业

必做：P2.2，P2.4，P2.5，P2.7

P2.2　试写出函数f局部极大点的必要条件和充分条件。

P2.4　某个零部件是在工厂车床上生产的，成本包括机器成本、工具有关的成本以及停工成本。机器成本与切割速度 V（m/min）成反比；工具有关的成本与 $V^{1.5}$ 成正比。总成本 c（美元）的公式为

$$c=\frac{240}{V}+10^{-4}V^{1.5}+0.45$$

试利用最优性条件式（2.5）和式（2.6）确定最优的切割速度，使得成本最小，并求对应的最小成本。

P2.5　在氯化钾溶液中，两个原子之间的平衡距离 r（单位：纳米）可保证总能量 E（单位：电子伏特）

$$E=-\frac{1.44}{r}+\frac{5.9\times10^{-6}}{r^{9}}$$

达到最小。可以看出，总能量是吸引能与排斥能之和。试利用最优性条件式（2.5）和式（2.6）确定平衡距离及其对应的总能量。

P2.7　某函数为 $f=x_1^4+x_2^4$，自变量 x_1 和 x_2 为实数。令 $\nabla f=\mathbf{0}$ 得到极小点 $\mathbf{x}^*=[0,0]^{\mathrm{T}}$，这是一个全局极小点，请给出理由。

明理，精工，笃学，致远　20

作业

选做：P2.11，P2.12，P2.13

P2.11 利用程序 FIBONACI 求解 P2.4。
P2.12 利用程序 GOLDLINE 和 GOLDINTV 求解 P2.5。
P2.13 利用 MATLAB 函数 fminbnd 求解 P2.4。

明理，精工，笃学，致远 21

本节结束，谢谢

Thank you for listening

课程名称：最优化方法

明理，精工，笃学，致远