统计推断在数模转换系统中的应用

组号：42　姓名：朱邦圻 学号：5130309684，姓名：李援曦 学号：5130309683

摘要：本文主要是对某些样品的某些参数进行检测，但是输出的参数呈现很明显的非线性的数学特性，本文着重于对数学模型的建立，结合统计推断的有关知识，完成选点、拟合、评估等过程，并最终得出相对比较完善的拟合方式。

关键词：遗传算法，曲线拟合，统计推断

ABSTRACT: We will text some samples' some parameters,but the output is clearly nonliner,we concentrate on the set-up of the math model,including some math knowledge to complete choosing the points,choosing the lines,evaluating,and get the final way.

Keywords:genetic algorithm,curve fitting,statistical inference

1. 引言:

在电子元器件完成生产的过程之后，检测其性能并在直角坐标系中将其参数拟合成直观的曲线图形式提供给用户,将能使客户更清晰地了解产品的性能。但是由于在现实情况当中，流水化的生产规模往往过于庞大,逐一逐个地去检查校验参数往往会花去大量的时间空间以及人力的成本，所以人们在长时间的生产生活中总结经验，学会了用几个有代表性的点去作为代表，用它们拟合的方式来拟合出最后的曲线的方式，在这个过程中，使用的点越多成本往往越大，但是也更加精确，使用的点越少成本就会控制的越小，但是拟合出的曲线往往并不是很精确。所以，我们必须取出既能不失精确性，又能极好的控制成本的一些点来作为代表来拟合。

2 评价标准的构建

为评估和比较不同的校准方案，特制定以下成本计算规则。

2.1 单点定标误差成本

$$s_{i,j}=\begin{cases}0 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|\le 0.5\\ 0.5 & \text{if } 0.5<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 1\\ 1.5 & \text{if } 1<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 2\\ 6 & \text{if } 2<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 3\\ 12 & \text{if } 3<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 5\\ 25 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|>5\end{cases} \quad (2\text{-}1)$$

单点定标误差的成本按式（2-1）计算，其中 $y_{i,j}$ 表示第 i 个样本之第 j 点 Y 的实测值，$\hat{y}_{i,j}$ 表示定标后得到的估测值（读数），该点的相应误差成本以符号 $s_{i,j}$ 记。

2.2 单点测定成本

实施一次单点测定的成本以符号 q 记。本课题指定 q=12。

2.3 某一样本个体的定标成本

$$S_i = \sum_{j=1}^{51} s_{i,j} + q \cdot n_i \qquad (2\text{-}2)$$

对样本 i 总的定标成本按式（2-2）计算，式中 n_i 表示对该样本个体定标过程中的单点测定次数。

2.4 校准方案总体成本

按式（2-3）计算评估校准方案的总体成本，即使用该校准方案对标准样本库中每个样本个体逐一定标，取所有样本个体的定标成本的统计平均。

$$C = \frac{1}{M} \sum_{i=1}^{M} S_i \qquad (2\text{-}3)$$

总体成本较低的校准方案，认定为较优方案。

3 数学模型和解决方案

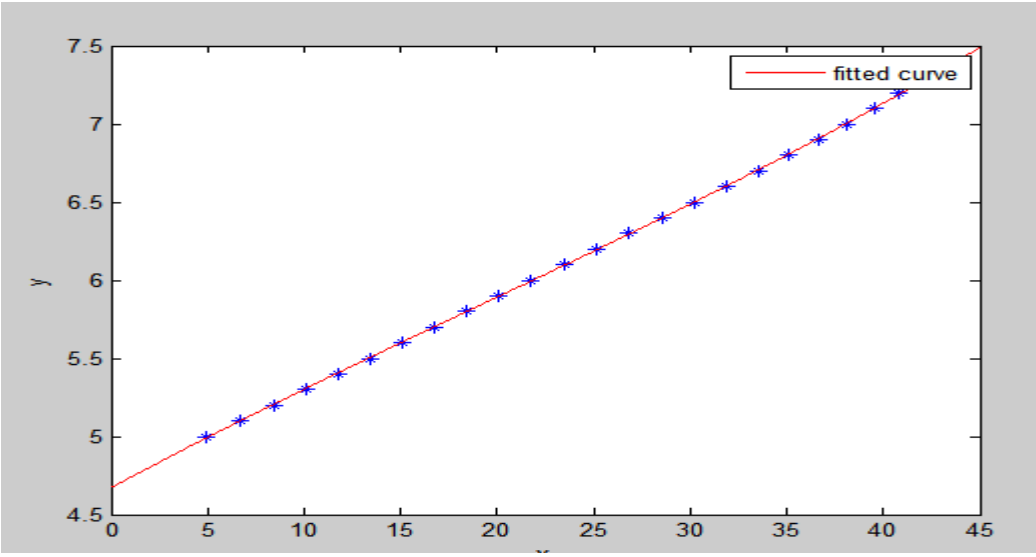
3.1 初步方案

3.1.1 数学模型

这个问题抽象为一个数学问题就是在一个器件 D-U 曲线未知的前提下，求取 7 个电压值进行测量，得出 7 个(Un,Dn)的点，通过这 7 个点拟合出 D-U 曲线。这 7 个电压值是根据抽样调查器件得到得若干样本进行统计推断后来求取的。而不是事先已知一组(Un,Dn)点，在这些点中寻找 7 个点来拟合曲线。

3.1.2 方案概述

本来分成三段，分别为（1,23）（24,37）（38,51）分别选点 3,2,2 个。



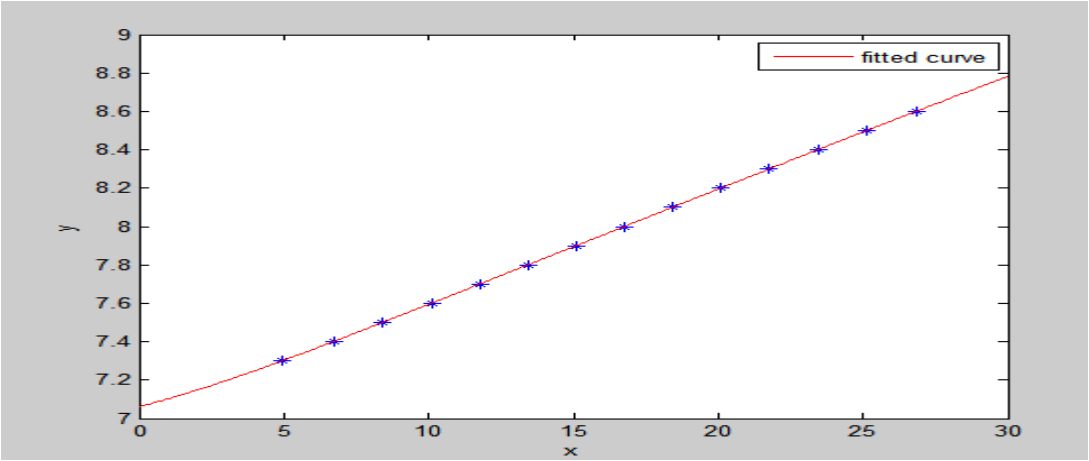
图一　前二十三个点的拟合

前二十三个点：　成本=1.094398986313637e+02

拟合曲线:

0.000007502013900x*3-0.000423230969466x*2+ 0.066446657246462x+4.670600690447534

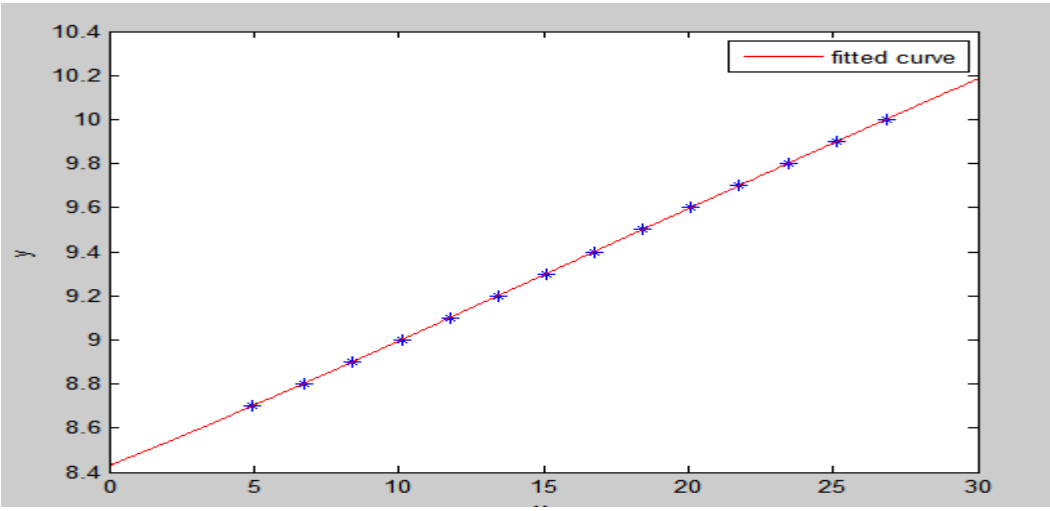
选出点（横坐标）（2,16,9）



图二　中间点的拟合

中间成本：56.079188976461268

选出点（横坐标）（24,25）



图三　　最后一段的拟合

最后一段成本：　57.096079572368851

选出点（横坐标）（42,48）

拟合曲线

0.000000203445845x*4 -0.000019808258347x*3+0.000627545500484x*2

+0.052004348572132x+8.429100144776932

所以总成本约为 222，选取的值为：[2,9,16,24,25,42,48]

3.1.3 方案总结

此方案虽然选点简介明了，运行速度极佳，但是不可否认的是，成本已经超过 200，拟合曲线很复杂，而且，中间段的三次样条法选出了前两个点，这种选点方式根本无法代表整条曲线，所以，这种方法是有着极大的局限性的。

3.2 改进方案

经过了一次次探讨以及小组面谈，和其他人交流之后，我们决定不再使用上述方式进行拟合，而是使用传统的遗传算法进行选点描图计算，而我们重新使用的拟合算法是三次样条算法。

3.2.1 遗传算法[2]

遗传算法（Genetic Algorithm）是一类借鉴生物界的进化规律（适者生存，优胜劣汰遗传机制）演化而来的随机化搜索方法。其主要特点是直接对结构对象进行操作，不存在求导和函数连续性的限定;具有内在的隐并行性和更好的全局寻优能力;采用概率化的寻优方法，能自动获取和指导优化的搜索空间，自适应地调整搜索方向，不需要确定的规则。遗传算法的这些性质，已被人们广泛地应用于组合优化、机器学习、信号处理、自适应控制和人工生命等领域。

遗传算法是解决搜索问题的一种通用算法，对于各种通用问题都可以使用。搜索算法的共同特征为：

（1）首先组成一组候选解；

（2）依据某些适应性条件测算这些候选解的适应度；

（3）根据适应度保留某些候选解，放弃其他候选解；

（4）对保留的候选解进行某些操作，生成新的候选解。

遗传算法还具有以下几个特点：

（1）遗传算法从问题解的串集开始搜索，而不是从单个解开始。这是遗传算法与传统优化算法的极大区别。传统优化算法是从单个初始值迭代求最优解的;容易误入局部最优解。遗传算法从串集开始搜索，覆盖面大，利于全局择优。

（2）遗传算法同时处理群体中的多个个体，即对搜索空间中的多个解进行评估，减少了陷入局部最优解的风险，同时算法本身易于实现并行化。

（3）遗传算法基本上不用搜索空间的知识或其它辅助信息，而仅用适应度函数值来评估个体，在此基础上进行遗传操作。适应度函数不仅不受连续可微的约束，而且其定义域可以任意设定。这一特点使得遗传算法的应用范围大大扩展。

3.2.2 方案实施

在Matlab遗传算法的实现中，需要确定一些初始化的参数，如编码串长度、种群大小、交叉和变异概率，为保证算法的运行效率和群体的多样性，我们选取交叉概率pc=0.9，突变概率pm=0.01，进化代数n=100。

在运算过程中，在适应度方面，我们采用了代价的相反数作为适应度的方式，来选取成本比较小，而适应度比较大的方案留下，出于拟合精确度考虑，我们直接先选取第1个点与第51个点，且本次使用三次插值法进行拟合。

表一 演化过程中特征点与成本的变化

代数	特征点	成本
1	1 2 8 12 18 27 30 32 43 45 51	137.8891
3	1 2 8 12 18 27 32 43 45 51	126.4446
4	1 2 8 12 18 27 30 43 45 51	121.9488
10	1 8 18 27 32 45 51	97.6077
20	1 8 18 26 32 45 51	97.2239
50	1 8 18 26 32 45 51	97.2239
100	1 8 19 26 34 45 51	95.9211

运行过几次后，用各种数据来进行对比，并选出最小的成本

表二 几组方案以及成本

组数	特征点	成本
1	1 8 19 26 34 45 51	95.9211
2	1 9 21 28 35 43 51	95.9733
3	1 12 21 25 33 44 51	98.0629
4	1 8 16 27 34 44 51	97.8166

5	1 11 21 29 38 47 51	98.4371

比较过后，选择的特征点为[1,8,19,26，34,45,51]成本为 95.9211。

3.2.3 方案结论

进行三次插值拟合，选择的特征点为[1,8,19,26，34,45,51]成本为 95.9211。

4. 鸣谢

特别感谢袁焱老师和李老师的指导，对我们的算法和小论文提出了至关重要的意见。

5. 参考文献

［1］袁焱老师. 统计推断课程讲座讲义. 上海：上海交通大学 电子工程系

[2]百度百科 词条“遗传算法”。

6 附录（MATLAB 运行代码）

6.1 初步方案代码

6.1.1 第一段拟合

```
format long
e=xlsread('20141010dataformgai.csv');
a=e(:,1:23);
b = mean (a)';c = var(a)';d = [5:0.1:7.2]';
b = [b,c,d];

 x = b (:,1)';y = b(:,3)'
ft = fittype('poly3');
yourLine = fit(x',y',ft);
plot(x,y,'*');
hold on
plot(yourLine)
coeffvalues(yourLine)
d = [5:0.1:7.2];
c = 0;
 for  n = 1:68
c  = c +sum (( yourLine (a(n,:))'-d).^2);
end
c
 A=(yourLine (b (:,1))'-d).^2;
[B,id]=sort(A)
```

6.1.2 第二段拟合

```
format long
e=xlsread('20141010dataformgai.csv');
a=e(:,1:14);
b = mean (a)';c = var(a)';d = [7.3:0.1:8.6]';
b = [b,c,d];
```

```
 x = b (:,1)';y = b(:,3)'
ft = fittype('spline');
yourLine = fit(x',y',ft);
plot(x,y,'*');
hold on
plot(yourLine)
coeffvalues(yourLine)
d = [7.3:0.1:8.6];
c = 0;
 for  n = 1:68
c  = c +sum (( yourLine (a(n,:))'-d).^2);
end
c
 A=(yourLine (b (:,1))'-d).^2;
[B,id]=sort(A)
```

6.1.3 最后一段拟合

```
format long
e=xlsread('20141010dataformgai.csv');
a=e(:,1:14);
b = mean (a)';c = var(a)';d = [8.7:0.1:10]';
b = [b,c,d];

 x = b (:,1)';y = b(:,3)'
ft = fittype('poly4');
yourLine = fit(x',y',ft);
plot(x,y,'*');
hold on
plot(yourLine)
coeffvalues(yourLine)
d = [8.7:0.1:10];
c = 0;
 for  n = 1:68
c  = c +sum (( yourLine (a(n,:))'-d).^2);
end
c

 A=(yourLine (b (:,1))'-d).^2;
[B,id]=sort(A)
```

6.2 改进方案代码

```
function out = select(gene,cost,pop)
%  自然选择,out(1)为父代最优予以保留
out=zeros(pop,51);
cost0=max(cost)-cost;
```

```
s0=sum(cost0);
s=zeros(pop+1);
s(1)=0;
s(pop+1)=1;
s(2:pop)=sum(cost0(1:pop-1))/s0;
for i=2:pop
    t=rand();
    j=search(t,s,1,pop+1);
    out(i,:)=gene(j,:);
end
sort0=[[1:pop]',cost];
sort0=sortrows(sort0,2);
out(1,:)=gene(sort0(1,1),:);
end

function [out] = search(in,s,l,r)
%  二分法查找
mid=floor((l+r)/2);
if in<=s(mid)
    if in>s(mid-1)
        out=mid-1;
    else
        out=search(in,s,l,mid);
    end
else
    if in<=s(mid+1)
        out=mid;
    else
        out=search(in,s,mid,r);
    end
end
end

function out = mutate(gene,pop,pm)
%  变异
out=gene;
for i=2:pop;
    for j=2:50;
        t=rand();
        if t<=pm
            out(i,j)=~out(i,j);
        end
    end
end
```

```
end

function out = generate(gene,pop,pc)
%  交叉,保留 gene(1)
%  i 与 pop-i+2 配对
for i=2:floor(pop/2+1)
    out=gene;
    mid=floor(rand()*50)+1;
    t=rand();
    if t<=pc
        out(i,1:mid)=gene(pop-i+2,1:mid);
        out(pop-i+2,1:mid)=gene(i,1:mid);
        out(i,mid+1:51)=gene(pop-i+2,mid+1:51);
        out(pop-i+2,mid+1:51)=gene(i,mid+1:51);
    end
end
end

function out = geneinit(pop)
%  随机产生初始种群
out=round(rand(pop,51)-0.2);
out(:,1)=1;
out(:,51)=1;
end

function [out] = errorcost(dy)
%  单个个体平均误差成本函数
t=abs(dy);
t0=sum(sum(t<=0.5));
t1=sum(sum(t<=1))-t0;
t2=sum(sum(t<=2))-t0-t1;
t3=sum(sum(t<=3))-t0-t1-t2;
t4=sum(sum(t<=5))-t0-t1-t2-t3;
t5=sum(sum(t>5));
out=0.5*t1+1.5*t2+6*t3+12*t4+25*t5;
end

function [out] =assess(in,y)
%   计算成本
out=length(in)*12;
x=5:0.1:10;
xx=5+(in-1)*0.1;
yy=y(:,in);  %  测试点 y 值矩阵
f=spline(xx,yy);
```

```
dy=ppval(f,x)-y;  %  理论值与实际值的差
out=out+errorcost(dy)/469;
fid=fopen('answer.txt','a');
fprintf(fid,'position: [ ');
fprintf(fid,'%2d ',in);
fprintf(fid,']    mean_cost: %7f\n\n',out);
fclose(fid);
end

function out = adapt(gene,y,pop)
%  计算每个个体平均成本
out=zeros(pop,1);
x=5:0.1:10;
for i=1:pop
    c=sum(gene(i,:)==1);  %  测试点数量
    pos=find(gene(i,:)==1);  %  测试点位置
    xx=5+(pos-1)*0.1;  %  测试点 x 值
    yy=y(:,pos);  %  测试点 y 值矩阵
    f=spline(xx,yy);
    dy=ppval(f,x)-y;  %  理论值与实际值的差
    out(i)=12*c+errorcost(dy)/469;  %  单个个体平均成本
end
min(out)
mean(out)
end

%  main()主函数
data=csvread('20141010dataform.csv');
pop=100;  %  种群数量
pc=0.9;  %  交叉概率
pm=0.01;  %  突变概率
n=100;  %  进化代数
y=zeros(469,51);
y(1:469,:)=data(2:2:938,:);
gene=geneinit(pop);
for g=1:n
    display(g);
    cost=adapt(gene,y,pop);
    gene=select(gene,cost,pop);
    gene=generate(gene,pop,pc);
    gene=mutate(gene,pop,pm);
    display(find(gene(1,:)==1));

display('-----------------------------------------------------')
```

```
end
xx=find(gene(1,:)==1);
assess(xx,y);  %  计算测试点组合的平均成本
```