问题1 ：

为了研究三种不同细菌对于小白鼠存活天数的影响，分三组实验，实验数据如下：

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 细菌类型 | 存活天数 | | | | | | | | | |
| 细菌1 | 7 | 11 | 6 | 6 | 7 | 9 | 5 | 10 | 6 | 3 |
| 细菌2 | 2 | 4 | 3 | 2 | 4 | 7 | 7 | 2 | 5 | 4 |
| 细菌3 | 5 | 6 | 8 | 5 | 10 | 7 | 12 | 6 | 6 | 8 |

试检验不同细菌对于小白鼠存活天数有无显著影响？

**解决思路：**

**1、分析问题**

本问题中：

|  |  |  |
| --- | --- | --- |
| 关心的指标 | 影响因素 | 影响因素的水平 |
| 存活天数 | 细菌类型 | 3 |

**2 将样本数据导入到matlab中**

所用代码：

[x,y]=xlsread('b.xls');

bac = x(:,1);

day = x(:,2);

**3、检验样本数据是否满足方差分析的基本假定**

所用代码：

for i=1:3

dayi = day(bac==i);

[h,p]=lillietest(dayi);

result(i,:)=p;

end

result

运行结果：

result =

0.3042

0.3042

0.1997

结果说明：

p值均大于0.05，所以接受原假设，即符合正态分布

**4、检验不同细菌对存活天数有无显著性影响？**

所用代码：

p = vartestn(day, bac)

运行结果：



结果说明：

经过正态性和方差齐性检验之后，认为3组数据服从方差相同的正态分布

**5、如果有显著性影响，任意两个细菌对存活天数是否有显著性不同？**

所用代码：

[p,table,stats] = anova1(day, bac)

运行结果：



结果说明：

p<0.05 拒绝原假设，表明3组数据有非常显著的差别

问题2

神经网络小问题

实例2

利用三层BP神经网络来完成非线性函数的逼近任务，其中隐层神经元个数为五个。

样本数据：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 输入X | 输出D | 输入X | 输出D | 输入X | 输出D |
| -1.0000 | -0.9602 | -0.3000 | 0.1336 | 0.4000 | 0.3072 |
| -0.9000 | -0.5770 | -0.2000 | -0.2013 | 0.5000 | 0.3960 |
| -0.8000 | -0.0729 | -0.1000 | -0.4344 | 0.6000 | 0.3449 |
| -0.7000 | 0.3771 | 0 | -0.5000 | 0.7000 | 0.1816 |
| -0.6000 | 0.6405 | 0.1000 | -0.3930 | 0.8000 | -0.3120 |
| -0.5000 | 0.6600 | 0.2000 | -0.1647 | 0.9000 | -0.2189 |
| -0.4000 | 0.4609 | 0.3000 | -0.0988 | 1.0000 | -0.3201 |

利用双极性Sigmoid函数作为激励函数。

程序如下：

input = (-1:0.1:1);

output = [-0.9602, -0.5770, -0.0729, 0.3771, 0.6405, 0.6600, 0.4609, 0.1336, -0.2013, -0.4344, -0.5000, -0.3930, -0.1647, -0.0988, 0.3072, 0.3960, 0.3449, 0.1816, -0.3120, -0.2189, -0.3201];

input\_train=input;

output\_train=output;

input\_test=input;

output\_test=output;

[inputn,inputps]=mapminmax(input\_train);

[outputn,outputps]=mapminmax(output\_train);

net=newff(inputn,outputn,5, {'tansig','purelin'});

net.trainParam.epochs=100;

net.trainParam.lr=0.1;

net.trainParam.goal=0.00004;

net=train(net,inputn,outputn);

inputn\_test=mapminmax('apply',input\_test,inputps);

an=sim(net,inputn\_test);

BPoutput=mapminmax('reverse',an,outputps);

figure(1)

plot(BPoutput,':og')

hold on

plot(output\_test,'-\*');

legend('预测输出','期望输出')

title('BP网络预测输出','fontsize',12)

ylabel('函数输出','fontsize',12)

xlabel('样本','fontsize',12)

error=BPoutput-output\_test;

figure(2)

plot(error,'-\*')

title('BP网络预测误差','fontsize',12)

ylabel('误差','fontsize',12)

xlabel('样本','fontsize',12)

figure(3)

plot((output\_test-BPoutput)./BPoutput,'-\*');

title('神经网络预测误差百分比')

errorsum=sum(abs(error))

net.IW{1,1}

net.B{1}

net.LW{2,1}

net.B{2}

所得结果：

输入层到中间层的权值：

中间层各神经元的阈值：

中间层到输出层的权值：

输出层各神经元的阈值：

ans =

-16.6064

-7.2251

6.9186

5.7439

-5.0588

ans =

12.0048

1.5566

-0.2046

1.6553

-4.2007

ans =

0.4057 -0.5953 -0.0137 -0.8715 -1.3034

ans =

-0.7837