**重 庆 大 学**

**学 生 实 验 报 告**

**实验课程名称 数学实验**

**开课实验室 DS1407**

**学 院 计算机 年级 2021 专业班 计科05、计科03**

**学 生 姓 名 李程程 学 号 20214582**

**学 生 姓 名 张梓健 学 号 20214578**

**学 生 姓 名 任俊璇 学 号 20215149**

**开 课 时 间 2022 至 2023 学年第 二 学期**

|  |  |
| --- | --- |
| **总 成 绩** |  |

|  |
| --- |
| **第一次作业**  **题目1**  在同一个坐标下作出y1=ex,y2=1+x,y3=1+x+(1/2)x2,y4= 1+x+(1/2)x2+(1/6)x3这四条曲线的图形，要求在图上加各种标注，同时用subplot作出这四条曲线，为每幅图形加上标题。  **程序**  x = linspace(-4,3,30);  y1 = exp(x);  y2 = 1 + x;  y3 = 1 + x + (1/2) \* x.^2;  y4 = 1 + x + (1/2) \* x.^2 + (1/6) \* x.^3;  hold on  plot(x,y1,'y-');  plot(x,y2,'m-.');  legend("1 + x");  plot(x,y3,'b--');  plot(x,y4,'-g.');  xlabel("x");  title("在同一个坐标下作出四条曲线的图形");  legend("exp(x)","1+x","1 + x + (1/2) \* x^2","y4 = 1 + x + (1/2) \* x^2 + (1/6) \* x^3");  hold off  hold on;  figure(2);  subplot(2,2,1)  plot(x,y1,'y-')  xlabel("x"); ylabel("y1"); title("exp(x)");  subplot(2,2,2)  plot(x,y2,'m-.')  xlabel("x"); ylabel("y2"); title("1+x");  subplot(2,2,3)  plot(x,y3,'b--')  xlabel("x"); ylabel("y3"); title("1 + x + (1/2) \* x^2")  subplot(2,2,4)  plot(x,y4,'-g.')  xlabel("x"); ylabel("y4"); title("y4 = 1 + x + (1/2) \* x^2 + (1/6) \* x^3")  hold off  **结果**  在同一个坐标下作出y1=ex,y2=1+x,y3=1+x+(1/2)x2,y4= 1+x+(1/2)x2+(1/6)x3这四条曲线的图形：  用subplot作出这四条曲线：  **分析**  这道题是二维绘图，所以要用到plot这个函数，第一个要求是在一幅图里同时画四条曲线，这要用到hold on和hold off，同时为了区分各条曲线，在使用plot绘制每条曲线时使用不同的参数来绘制以区分，并且使用legend和label给图加上标注。第二个要求是使用subplot在每一个子图里绘制一条曲线，这时可以使用figure（2）另起一幅图，然后使用subplot在新图一一绘制每个子图，同时使用title加上标题。在绘制过程中遇到了一个问题是exp（x）在x > 0 时增长速率远高于另外三个函数，导致如果在一幅图中同时画四条曲线的话，y轴单位长度过大，其他三条曲线看上去几乎为一条直线，所以我将x范围定在原点附近的一小段区域，如果x要求取到较大正数，可以使用plotyy，将exp（x）的y轴放到右边，我实验过确实可行，不过看起来比较乱，题目也没这样要求，所以最后考虑还是使用缩小x范围的效果进行展示。  **题目2**  绘制如下函数 曲面图。  **程序**  x = linspace(-5,5,30);  y = linspace(-5,5,30);  [X,Y] = meshgrid(x,y);  Z = y.^2 + 3 .\* X .\* Y + X.^2;  mesh(X,Y,Z);  xlabel("x"); ylabel("y"); zlabel("z");  title("Z = y^2 + 3 \* X \* Y + X^2")  **结果**  **分析**  这道题是绘制三维曲面，我选择使用mesh函数来绘制，需要先按照x和y生成网格矩阵X和Y，然后使用X和Y求得Z，然后使用mesh绘制三维曲面。  **题目3**  编写函数M-文件sq.m：用迭代法求的值。求平方根的迭代公式为    迭代的终止条件为前后两次求出的x的差的绝对值小于10−5。  **程序**  x = 1;  y = 0;  a = c; //c为一个输入数值  i = 1;  disp("x = a^(1/2)");  fprintf("a = %i\n",a);  fprintf("x初始值 = %i\n", x);  while abs(x - y) >= 0.00001  y = x;  x = 0.5 \* (x + a / x);  fprintf("x%i = %f\n",i,x);  i = i + 1;  end  **结果**  **分析**  这道题是方程求解，使用点迭代法，迭代公式题目已经给出，按照要求用while来判断何时终止迭代，并在迭代过程中将每次迭代的x值打印输出以便后续观察和检验，题目中a的取值没有给出，我将a分别赋值为2和16进行了测试，结果证明程序可以正常执行，并且功能符合题目要求。  **题目4**  将方程改写成各种等价的形式进行迭代，观察迭代是否收敛，并给出解释。  **程序**  fc = "x^5 + 5x^3 - 2x + 1 = 0";  x = 1;  x1 = x; x2 = x; x3 = x;  fprintf("x初始值 = %i\n",x);  disp("求解方程为 x^5 + 5x^3 - 2x + 1 = 0");  disp("第一个迭代方程为 x = (1/2)x^5 + (5/2)x^3 + 1/2");  disp("第二个迭代方程为 x = (-1/5)x^3 + (2/5)x^-1 - (1/5)x^-2");  disp("第三个迭代方程为 x = -5x^-1 + 2x^-3 - x^-4");  for i = 1:20  x1 = (1/2) \* x1^5 + (5/2) \* x1^3 + 1/2;  x2 = (-1/5) \* x2^3 + (2/5) \* x2^-1 -(1/5) \* x2^-2;  x3 = (-5) \* x3^-1 + 2 \* x3^-3 - x3^-4;  fprintf("x1 = %f x2 = % f x3 = %f\n",x1,x2,x3);  end  disp("第一个加速迭代方程为 x = (-4\*x^5-10\*x^3+1)/(2-5\*x^4-15\*x^2)");  disp("第二个加速迭代方程为 x = (2\*x^3+4\*x^-1-3\*x^-2)/(5+3\*x^3+2\*x^-2-2\*x^-3)");  disp("第三个加速迭代方程为 x = (-10\*x^-1+8\*x^-3-5\*x^-4)/(1-5\*x^-2+6\*x^-4-4\*x^-5)");  x1 = x; x2 = x; x3 = x;  for i = 1:20  x1 = (-4 \* x1^5 - 10 \* x1^3 + 1)/(2 - 5 \* x1^4 - 15 \* x1^2);  x2 = (2 \* x2^3 + 4 \* x2^-1 - 3 \* x2^-2)/(5 + 3 \* x2^3 + 2 \* x2^-2 - 2 \* x2^-3);  x3 = (-10 \* x3^-1 + 8 \* x3^-3 - 5 \* x3^-4)/(1 - 5 \* x3^-2 + 6 \* x3^-4 - 4 \* x3^-5);  fprintf("x1 = %f x2 = % f x3 = %f\n",x1,x2,x3);  end  **结果**  **分析**  这道题也是求解方程，原方程可以改写成三种迭代方程以及相应的加速迭代方程，对x赋初始值为1进行测试，结果显示三个迭代方程和第二个加速迭代方程不收敛，第一个和第三个加速迭代方程收敛。从打印输出的每次迭代值可以看出，第一个和第二个迭代方程x绝对值变化极大，没经过几次迭代就变成了无穷大或无穷小，所以x无法收敛；第三个迭代方程x每次迭代改变正负，正值和负值分别收敛一个值；第二个加速迭代方程x反复变化，并不收敛；第一个和第三个加速收敛方程收敛于-0.768463，并且第三个收敛速度快于第一个，经验证该收敛值是方程的近似解。 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **第二次作业**  **题目1**  用向前欧拉公式和改进的欧拉公式求方程,，，的数值解，要求编写程序，并比较两种方法的计算结果，说明了什么问题？  **模型1**    欧拉法求解微分方程数值解。  **程序1**  %求解析解  y=dsolve('Dy=y-2\*x/y','y(0)=1','x');  y  x1(1)=0;y1(1)=1;y2(1)=1;h=0.1;  for k=1:10  %向前欧拉公式  x1(k+1)=x1(k)+h;  y1(k+1)=y1(k)+h\*(y1(k)-2\*x1(k)/y1(k));  %改进欧拉公式  k1=y2(k)-2\*x1(k)/y2(k);  k2=(y2(k)+h\*k1)-2\*x1(k+1)/(y2(k)+h\*k1);  y2(k+1)=y2(k)+(h/2)\*(k1+k2);  end  x1,y1,y2  x=0:0.1:1;  y=(2\*x+1).^(1/2);  plot(x,y,'b-\*');  hold on  plot(x1,y1,'k-.',x1,y2,'r');  legend('y:解析解','y1:向前欧拉公式','y2:改进欧拉公式');  **结果1**      **分析1**  如上图为向前欧拉公式与改进欧拉公式所计算的结果，分析数据和图像可以得出，改进欧拉公式得到的结果精度更高，更接近于解析解。  **题目2**  Rossler微分方程组：  当固定参数b=2, c=4时，试讨论随参数*a*由小到大变化（如*a*∈(0,0.65))而方程解的变化情况，并且画出相图，观察相图是否形成混沌状？  **模型2**  Rossler微分方程组，ode45求解微分方程数值解。  **程序2**  %rossler.m文件：  function xdot = rossler(t,x)  xdot = [-x(2)-x(3);x(1)+a\*x(2);2+x(1)\*x(3)-4\*x(3)];  %主程序main.m：  [t,x]=ode45('rossler',[0,20],[0 0 0]);  plot(t,x(:,1),'b:',t,x(:,2),'r.',t,x(:,3),'g')  legend('x','y','z');  figure(2)  plot3(x(:,1),x(:,2),x(:,3))  xlabel('x'); ylabel('y'); zlabel('z');  grid on  **结果2**  a = 0.1时  a = 0.3时  a = 0.6时  **分析2**  我们使用ode45函数来求解Rossler微分方程组，t取[0,20],将a的值分别设为0.1，0.3，0.6进行实验，方程解变化以及相图如上图所示，可见，随着a变大，方程解的波动也随之变大，相图形成混沌状。  **题目3**  增加生产、发展经济所依靠的主要因素有增加投资、增加劳动力以及技术革新等，在研究国民经济产值与这些因素的数量关系时，由于技术水平不像资金、劳动力那样容易定量化，作为初步的模型，可认为技术水平不变，只讨论产值和资金、劳动力之间的关系。在科学技术发展不快时，如资本主义经济发展的前期，这种模型是有意义的。  用*Q，K，L*分别表示产值、资金、劳动力，要寻求的数量关系*Q(K,L)*。经过简化假设与分析，在经济学中，推导出一个著名的Cobb-Douglas生产函数：  *Q(K,L) = aKαLβ*， 0<α,β<1 （\*）  式中α,β*，a*要由经济统计数据确定。现有美国马萨诸塞州1900—1926年上述三个经济指数的统计数据，如下表，试用数据拟合的方法，求出式（\*）中的参数α*,*β*，a*。  表1   |  |  | | --- | --- | | *t Q K L* | *t Q K L* | | 1900 1.05 1.04 1.05  1901 1.18 1.06 1.08  1902 1.29 1.16 1.18  1903 1.30 1.22 1.22  1904 1.30 1.27 1.17  1905 1.42 1.37 1.30  1906 1.50 1.44 1.39  1907 1.52 1.53 1.47  1908 1.46 1.57 1.31  1909 1.60 2.05 1.43  1910 1.69 2.51 1.58  1911 1.81 2.63 1.59  1912 1.93 2.74 1.66  1913 1.95 2.82 1.68 | 1914 2.01 3.24 1.65  1915 2.00 3.24 1.62  1916 2.09 3.61 1.86  1917 1.96 4.10 1.93  1918 2.20 4.36 1.96  1919 2.12 4.77 1.95  1920 2.16 4.75 1.90  1921 2.08 4.54 1.58  1922 2.24 4.54 1.67  1923 2.56 4.58 1.82  1924 2.34 4.58 1.60  1925 2.45 4.58 1.61  1926 2.58 4.54 1.64 |   **模型3**  非线性拟合，最小二乘法  **程序3**  %curvefun.m  function f=curvefun(x,KLdata)  f=x(1)+(KLdata(1,:).^x(2)).\*(KLdata(2,:).^x(3));  %main.m  KLdata=[1.04,1.06,1.16,1.22,1.27,1.37,1.44,1.53,1.57,2.05,2.51,2.63,2.74,2.82...  3.24,3.24,3.61,4.10,4.36,4.77,4.75,4.54,4.54,4.58,4.58,4.58,4.54;...  1.05,1.08,1.18,1.22,1.17,1.30,1.39,1.47,1.31,1.43,1.58,1.59,1.66,1.68...  1.65,1.62,1.86,1.93,1.96,1.95,1.90,1.58,1.67,1.82,1.60,1.61,1.64];  Qdata=[1.05 1.18 1.29 1.30 1.30 1.42 1.50 1.52 1.46 1.60 1.69 1.81 1.93 1.95...  2.01 2.00 2.09 1.96 2.20 2.12 2.16 2.08 2.24 2.56 2.34 2.45 2.58];  x0=[0.1,0.1,0.1];  x=lsqcurvefit('curvefun',x0,KLdata,Qdata)  **结果3**  结果从左到右以此为a，α,β    **分析3**  通过lsqcurvefit函数求非线性最小二乘拟合，得到最终结果为  *Q(K,L) = 1.2246K0.4612L-0.1277*  **题目4**  收集重庆市的人口数据，采用数据拟合预测2030年重庆市的人口数**。**  **模型4**  线性最小二乘拟合。  **程序4**  %多项式次数为5时  %renkou.m：  function num = renkou(a,m)  num = a(1)\*m^5 + a(2)\*m^4 + a(3)\*m^3 + a(4)\*m^2 + a(5)\*m + a(6);  x = 0:1:21;  y = [2848.8 2829.2 2814.8 2803.2 2793.3 2798 2808 2816 2839 2859 2884.6 2944.4 2974.9 3011 3043.5 3070 3110 3163.1 3187.8 3208.9 3212.4 3213.3];  a = polyfit(x,y,5)  z = polyval(a,x);  plot(x,y,'k\*',x,z,'r-')  ylabel('单位万');  legend('实际人口数','拟合人口数');  num\_2030 = renkou(a,30)  %多项式次数为10时  %renkou.m：  function num = renkou(a,m)  num = a(1)\*m^10 + a(2)\*m^9 + a(3)\*m^8 + a(4)\*m^7 + a(5)\*m^6 + a(6)\*m^5 + a(7)\*m^4 + a(8)\*m^3 + a(9)\*m^2 + a(10)\*m + a(11);  x = 0:1:21;  y = [2848.8 2829.2 2814.8 2803.2 2793.3 2798 2808 2816 2839 2859 2884.6 2944.4 2974.9 3011 3043.5 3070 3110 3163.1 3187.8 3208.9 3212.4 3213.3];  a = polyfit(x,y,5)  z = polyval(a,x);  plot(x,y,'k\*',x,z,'r-')  ylabel('单位万');  legend('实际人口数','拟合人口数');  num\_2030 = renkou(a,30)  **结果4**  多项式次数为5时  多项式次数为10时    **分析4**  我们使用polyfit函数作线性最小二乘拟合，多项式的次数分别设为5和10进行实验，结果如上图所示，可见选取的多项式拟合函数不同，拟合结果会有差别。  **第三次作业**  **题目1**  求解无约束优化  1) 画出该曲面图形,直观地判断该函数的最优解;  2) 使用fminunc命令求解,能否求到全局最优解?  **模型1**  无约束非线性优化。  **程序1**  wuyueshuyouhua.m文件:  function f = wuyueshuyouhua(x)  f = -20.\*exp(-0.2.\*(0.5\*(x(1).^2+x(2).^2).^0.5))-exp(0.5.\*(cos(2.\*pi.\*x(1))  +cos(2.\*pi.\*x(2))))+22.713;  x1 = linspace(-5,5,100);  x2 = linspace(-5,5,100);  [X1,X2]=meshgrid(x2,x2);  Y = -20\*exp(-0.2\*((0.5\*(X1.^2+X2.^2)).^0.5))-exp(0.5.\*(cos(2.\*pi.\*x1)+cos(2.\*pi.\*x2)))+22.713;  surf(X1,X2,Y);  xlabel('X1');  ylabel('X2');  zlabel('f(X1,X2)');  options=optimset('display','iter');  X0=[0,0];  [x,fval]=fminunc('wuyueshuyouhua',X0,options);  x  fval  **结果1**      **分析1**  上面的曲线图是用surf函数画出的该函数的曲面图形，由于精度限制，在图形中大致可以观察出该函数最小值的最优解应该是X1和X2取0左右,f(X1,X2)最小值也在0左右。用fminunc函数求解，x初始值设为[0 0]，求得该函数最小值得的最优解如图所示，为X1=-1.535\*10^-16, X2=-1.535\*10^-16, f(X1,X2)最小值=-0.0053，结合图形判断改解即为全局最优解。  **题目2**  请自行查询某商业银行的整存整取年利率，填入下表：   |  |  |  |  | | --- | --- | --- | --- | | 一年期 | 二年期 | 三年期 | 五年期 | |  |  |  |  |   现有1笔本金，准备30年后使用，若此期间利率不变，问应该采用怎样的存款方案？  **模型2**  约束非线性优化。  查询到的中国农业银行整存整取年利率：  1.65%，2.15%，2.6%，2.65%  根据题目以及查询到的中国农业银行整存整取年利率构造数学模型minProfit(x)=-(1+1.65%)^x1\*(1+2\*2.15%)^x2\*(1+3\*2.6%)^x3\*(1+5\*2.65%)^x4;  st: k1+2\*k2+3\*k3+5\*k4=30;  0<=x1<=30;  0<=x2<=15;  0<=x3<=10;  0<=x4<=6;  **程序2**  profit.m文件：  function f = profit(x)  f = -(1+0.0167)^x(1)\*(1+2\*0.0215)^x(2)\*(1+3\*0.026)^x(3)\*(1+5\*0.0265)^x(4);  x0 = [0 0 10 0];  A = [1 0 0 0;0 1 0 0;0 0 1 0;0 0 0 1];  B = [30 15 10 6];  Aeq = [1 2 3 5];  beq = 30;  lb = [0 0 0 0];  options = optimset('largescal','off');  [x,fval] = fmincon('profit',x0,A,B,Aeq,beq,lb);  x  fval  **结果2**    **分析2**  根据题意以及查询到的中国农业银行整存整取年利率构造出计算总利率的数学模型，因为该数学模型是约束非线性的，所以使用fmincon函数来求得最优解，如上图所示，最优解为  x3=9.9991，x4=0.0005，profit=2.1193，因为题目要求x都为整数，所以取x3=10来计算得profit约等于2.1193，所以应采用的存款方案为选择10次三年期的整存整取，能获得最大收益约为本金的2.1193倍。  **题目3**  A公司面临破产，只余下100种物品，表1中给出了每种物品的数量，现有1000名公司债权人，表格中给出了债权人对不同物品的偏好（数值越大越喜欢），要求你们对这些资产进行处置，应该如何安排呢？  **模型3**  0, 债权人i为得到物品j；  由题设**xij= i=1,2…1000;j=1,2,…100;**  1, 债权人i得到物品j；  由题可知物品数量共3007件，债权人有1000人，因此公平起见，每个债权人至少分得1件，至多分得4件物品。  使用lingo求解，偏好值最大时的分配方案。  **程序3**  定义债权人和物品：  sets:  person/1..1000/;  item/1..100/:b;  link(person,item):c,x;  endsets  导入数据如图（仅展示部分）：    enddata  目标及约束段：  max=@sum(link:c\*x);  @for(link(i,j):@bin(x(i,j)));  @for(person(i):@sum(item(j):x(i,j))>=1);  @for(person(i):@sum(item(j):x(i,j))<=4);  @for(item(j):@sum(person(i):x(i,j))=b(j));  end  **结果3**    X(i,j)部分结果：    **分析3**  根据题意使用lingo求解，偏好值最大时的分配方案。通过物品总数量与债权人数量确定每个债权人应分得的物品数量范围。  通过lingo解得最大偏好值总和为23390.  **第四次作业**  **题目1**  火车行驶的路程、速度数据如表1，计算从静止开始20 分钟内走过的路程。  表1   |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | *t*(分) | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | | *v*(km/h) | 10 | 18 | 25 | 29 | 32 | 20 | 11 | 5 | 2 | 0 |   **模型1**  一维插值。  **程序1**  t = [2 4 6 8 10 12 14 16 18 20];  v = [10 18 25 29 32 20 11 5 2 0];  plot(t,v)  xlabel('t(分)')  ylabel('v(千米每小时）')  title('速度曲线')  t = t \* 60;  v=v/3.6;  s = cumsum(v) .\* diff([0 t]);  t\_new = 0:1:1200;  s\_new = interp1(t, s, t\_new, 'linear');  distance = s\_new(end);  **结果1**      **分析1**  利用线性插值的方法每秒插值一次，用interp1函数实现这个结果，在这个过程中注意单位之间的换算，将分化为秒千米每小时化为米每秒计算。  **题目2**  确定地球与金星之间的距离  天文学家在1914年8月份的7次观测中，测得地球与金星之间距离（单位：米），并取其常用对数值，与日期的一组历史数据如表2。  表2   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 日期（号） | 18 | 20 | 22 | 24 | 26 | 28 | 30 | | 距离对数 | 9.9617724 | 9.9543645 | 9.9468069 | 9.9390950 | 9.9312245 | 9.9231915 | 9.9149925 |   由此推断何时金星与地球的距离（米）的对数值为9.9351799？  **模型2**  由于日期与距离的对数成近似线性关系，所以采用三次样条插值进行求解  **程序2**  date=18:2:30;  distance=[9.9617724 9.9543645 9.9468069 9.9390950 9.9312245 9.9231915 9.9149925];  date1=interp1(distance,date,9.9351799,'spline')  plot(date,distance,'--x',date1,9.9351799,'o');  xlabel('日期');  ylabel('距离对数');  **结果2**      **分析2**  如图所示，通过spline方法进行插值，得出当金星与地球的距离（米）的对数值为9.9351799时，日期为25日。  **题目3**  山区地貌图 在某山区（平面区域（0，2800）×（0，2400）内，单位：米）测得一些地点的高程（单位：米）如表3，试作出该山区的地貌图和等高线图。  表3   |  |  | | --- | --- | | 2400  2000  1600  1200  800  400  0 | 1430 1450 1470 1320 1280 1200 1080 940  1450 1480 1500 1550 1510 1430 1300 1200  1460 1500 1550 1600 1550 1600 1600 1600  1370 1500 1200 1100 1550 1600 1550 1380  1270 1500 1200 1100 1350 1450 1200 1150  1230 1390 1500 1500 1400 900 1100 1060  1180 1320 1450 1420 1400 1300 700 900 | | Y/X | 0 400 800 1200 1600 2000 2400 2800 |   **模型3**  二维插值。  **程序3**  x=0:400:2800;  y=0:400:2400;  z=[1180 1320 1450 1420 1400 1300 700 900;...  1230 1390 1500 1500 1400 900 1100 1060;...  1270 1500 1200 1100 1350 1450 1200 1150;...  1370 1500 1200 1100 1550 1600 1550 1380;...  1460 1500 1550 1600 1550 1600 1600 1600;...  1450 1480 1500 1550 1510 1430 1300 1200;...  1430 1450 1470 1320 1280 1200 1080 940];  xi=0:50:2800;  yi=0:50:2400;  figure(1);  subplot(2,2,1);  meshz(x,y,z)  xlabel('X'),ylabel('Y'),zlabel('Z'),title("原数据（单位：米）")  subplot(2,2,2);  z1i=interp2(x,y,z,xi,yi','nearest');  surfc(xi,yi,z1i)  xlabel('X'),ylabel('Y'),zlabel('Z'),title("最邻近插值（单位：米）")  subplot(2,2,3);  z2i=interp2(x,y,z,xi,yi');  surfc(xi,yi,z2i)  xlabel('X'),ylabel('Y'),zlabel('Z'),title("双线性插值（单位：米）")  subplot(2,2,4);  z3i=interp2(x,y,z,xi,yi','cubic');  surfc(xi,yi,z3i)  xlabel('X'),ylabel('Y'),zlabel('Z'),title("双三次插值（单位：米）")  figure(2)  subplot(2,2,1),contour(x,y,z,10,'r')  xlabel('X'),ylabel('Y'),title("原数据（单位：米）")  subplot(2,2,2),contour(xi,yi,z1i,10,'r')  xlabel('X'),ylabel('Y'),title("最邻近插值（单位：米）")  subplot(2,2,3),contour(xi,yi,z2i,10,'r')  xlabel('X'),ylabel('Y'),title("双线性插值（单位：米）")  subplot(2,2,4),contour(xi,yi,z3i,10,'r')  xlabel('X'),ylabel('Y'),title("双三次插值（单位：米）")  **结果3**      **分析3**  根据题意使用二维插值，借助matlab函数interp2并且使用三种二维插值方法与原数据进行比较，画出的地貌图和等高线图如上图所示，三种插值方法得到的数据逐渐光滑。  **第五次作业**  **题目1**    完整数据如下表：      **模型1**  多元线性回归模型**。**  **程序1**  y1=[22.1 15.4 11.7 10.3 11.4 7.5 13.0 12.8 14.6 ...  18.9 19.3 30.1 28.2 25.6 37.5 36.1 39.8 44.3] ;  y2=[7.2 5.4 7.6 2.5 2.4 1.7 4.3 3.7 3.9 ...  7.0 6.8 10.1 9.4 7.9 14.1 14.5 14.9 15.6];  x1=[1.89 1.94 1.95 1.82 1.85 1.78 1.76 1.76 1.75...  1.74 1.70 1.70 1.68 1.60 1.61 1.64 1.67 1.68];  x2=[6.1 6.2 6.3 8.2 9.8 10.3 10.5 8.7 7.4 6.9...  5.2 4.9 4.3 3.7 3.6 3.1 1.8 2.3];  X=[ones(18,1),x1',x2'];  [b1, bint1, r1, rint1, s1]=regress (y1' , X) ;  figure(1);  b1,bint1,s1  rcoplot(r1,rint1)  [b2, bint2, r2, rint2, s2]=regress (y2' , X) ;  figure(2);  b2,bint2,s2  rcoplot(r2,rint2)  **结果1**  原始数据得到的结果和残差及其置信区间图形如下：          删除了异常点后，得到的结果和残差及其置信区间图形如下：          **分析1**  最终得到的普通型汽车模型的系数的置信区间都不包含零点，所以模型有效，且R2较大，s2较小，说明模型精度较高；而豪华型汽车模型β1的置信区间包含了零点，所以该模型无效。  **题目1**    **模型2**  多元线性回归模型。  **程序2**  y = [22.1 15.4 11.7 10.3 11.4 7.5 13.0 12.8 14.6 ...  18.9 19.3 30.1 28.2 25.6 37.5 36.1 39.8 44.3 ...  7.2 5.4 7.6 2.5 2.4 1.7 4.3 3.7 3.9 ...  7.0 6.8 10.1 9.4 7.9 14.1 14.5 14.9 15.6];  x1 = [1.89 1.94 1.95 1.82 1.85 1.78 1.76 1.76 1.75...  1.74 1.70 1.70 1.68 1.60 1.61 1.64 1.67 1.68];  x2 = [6.1 6.2 6.3 8.2 9.8 10.3 10.5 8.7 7.4 6.9...  5.2 4.9 4.3 3.7 3.6 3.1 1.8 2.3];  x3 = [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ...  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ];  x4 = [x1,x1];  x5 = [x2,x2];  X=[ones(36,1),x4',x5',x3'];  [b,bint,r,rint,s]=regress(y',X);  b,bint,s  **结果2**      **分析2**  最终得到的回归模型为**y=64.5753-16.1436x1-2.3322x2-14.4222x3**。b1的置信区间包含了零点，所以该模型无效。将x3=0代入，得到豪华车的二元模型**y=64.5753-16.1436x1-2.3322x2；**x3=1代入，得到普通车的二元模型**y=50.1531-16.1436x1-2.3322x2，**与第一题得到的模型系数差别较大。  **题目3**    **模型3**  多元线性回归模型，散点图。  **程序3**  figure(1)  scatter(x4,r)  scatter(x4,rint)  title("x1与残差散点图")  xlabel("x1"); ylabel("残差")  figure(2)  scatter(x5,r)  scatter(x5,rint)  title("x2与残差散点图")  xlabel("x2"); ylabel("残差")  **结果3**    **分析3**  使用scatter函数分别作出x1和x2与残差的散点图，由图可见，两种类型的汽车残差分布不同，即车的类型对x1和x2都有影响，所以统一模型中应加入交互性。  **题目4**    **模型4**  多元线性回归模型，逐步回归。  **程序4**  y = [22.1 15.4 11.7 10.3 11.4 7.5 13.0 12.8 14.6 ...  18.9 19.3 30.1 28.2 25.6 37.5 36.1 39.8 44.3 ...  7.2 5.4 7.6 2.5 2.4 1.7 4.3 3.7 3.9 ...  7.0 6.8 10.1 9.4 7.9 14.1 14.5 14.9 15.6];  x1 = [1.89 1.94 1.95 1.82 1.85 1.78 1.76 1.76 1.75...  1.74 1.70 1.70 1.68 1.60 1.61 1.64 1.67 1.68 ...  1.89 1.94 1.95 1.82 1.85 1.78 1.76 1.76 1.75...  1.74 1.70 1.70 1.68 1.60 1.61 1.64 1.67 1.68];  x2 = [6.1 6.2 6.3 8.2 9.8 10.3 10.5 8.7 7.4 ...  6.9 5.2 4.9 4.3 3.7 3.6 3.1 1.8 2.3 ...  6.1 6.2 6.3 8.2 9.8 10.3 10.5 8.7 7.4 ...  6.9 5.2 4.9 4.3 3.7 3.6 3.1 1.8 2.3];  x3 = [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ...  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ];  x4 = x1 .\* x2;  x5 = x1 .\* x3;  x6 = x2 .\* x3;  x7 = x1 .^2;  x8 = x2 .^2;  x9 = x3 .^2;  X = [x1', x2', x3', x4', x5', x6', x7', x8', x9'];  stepwise(X,y,[1,2])  **结果4**    **分析4**  对统一模型加入二次项和交互项，然后使用stepwise函数进行逐步回归，最终得到的结果如上图，模型为 y = 55.4129 – 7.71189x2 + 2.28504x2x3 + 0.31x22 – 28.2975x32,与此前的统一模型相比，系数置信区间都不包含0，模型有效，F更大，模型可信度更高，R2更大，模型精度更高，可见加入二次项和交互性后得到的模型有所改进。 |

备注：