**Matlab科学计算语言及应用**

**21221学期**

第2次

实验报告

学号：3020234369

姓名 : 李佳林

班级 : 通信3班

题目：1 Semilog plot.

代码：

clear; clc;

figure(1)

year = 1:6;

numStudents = [15 25 55 115 144 242];

semilogy(year,numStudents,'s--','MarkerFaceColor','m');

xlabel('years');

ylabel('numStudents');

title('trend of increase');

xlim([1 6]);

实验结果及分析：  
图表, 折线图

描述已自动生成

得到数据图像大致为直线，题中假设指数增长关系大致成立。

题目：2 Subplot and axis modes

代码：

load Map

figure(2);

subplot(221);

imagesc(map);

colormap(cMap);

axis square;

title('square');

subplot(222);

imagesc(map);

colormap(cMap);

axis tight;

title('tight');

subplot(223);

imagesc(map);

colormap(cMap);

axis equal;

title('equal');

subplot(224);

imagesc(map);

colormap(cMap);

axis xy

title('xy');

实验结果及分析：

地图

描述已自动生成

观察四个图像的不同。

题目：3 Bar graph

代码：

rvector = rand(1,5);

X = categorical({'1','2','3','4','5'});

X = reordercats(X,{'1','2','3','4','5'});

figure(3);

bar(X,rvector,0.8,'r');

ylim([0 1]);

title('Bar Graph of 5 Random Values');

实验结果及分析：

图表, 条形图

描述已自动生成

柱状图格式符合题中示例要求

题目：4 nterpolation and surface plots

代码：

clf; clear; clc;

Z0 = rand(5);

[X0,Y0] = meshgrid(1:5);

[X1,Y1] = meshgrid(1:0.1:5,1:0.1:5);

Z1 = interp2(X0,Y0,Z0,X1,Y1,'cubic');

surf(X1,Y1,Z1);

zlim([-0.2 1.2])

colormap(hsv);

shading interp

hold on

contour(X1,Y1,Z1,15)

c.LineWidth = 5;

colorbar

caxis([0 1])

实验结果及分析：

图表, 表面图

描述已自动生成

图像符合题中要求

题目：5 Fun with find

代码：

function ind = findNearest(x, desiredVal)

temp = x-desiredVal;

Tvector = temp(:);

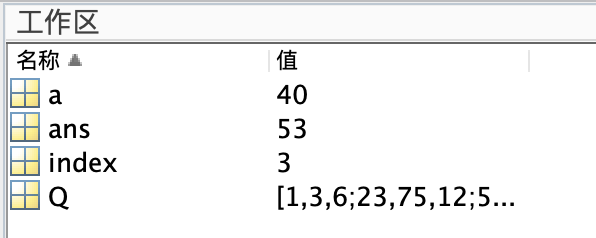
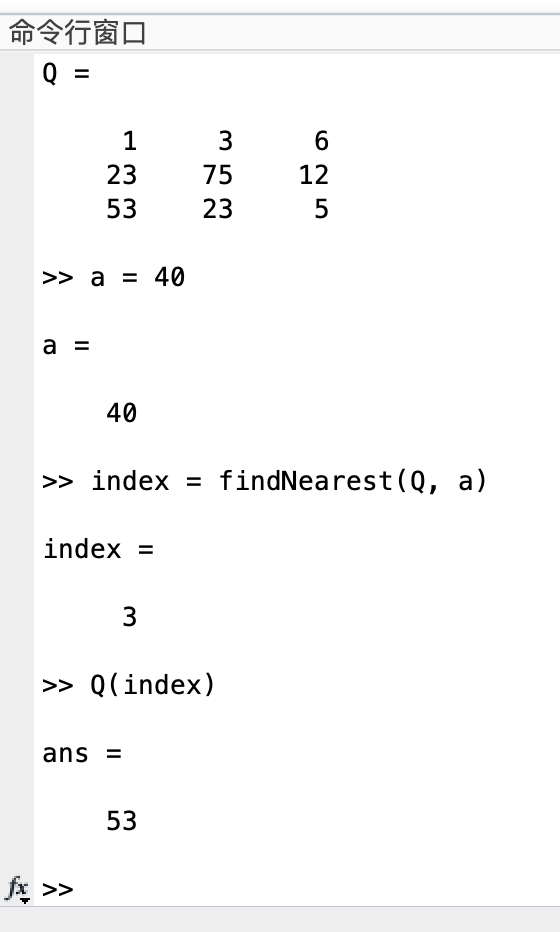
Tvector = abs(Tvector);

minOne = min(Tvector);

ind = find(Tvector==minOne);

end

实验结果及分析：



输入测试数据证明函数功能正确无误

题目：6 Loops and flow control

代码：

function loopTest(N)

for n=1:N

if mod(n,2) == 0 && mod(n,3) == 0

disp([num2str(n) ' is divisible by 2 AND 3.']);

elseif mod(n,3) == 0

disp([num2str(n) ' is divisible by 3.']);

elseif mod(n,2) == 0

disp([num2str(n) ' is divisible by 2.']);

else

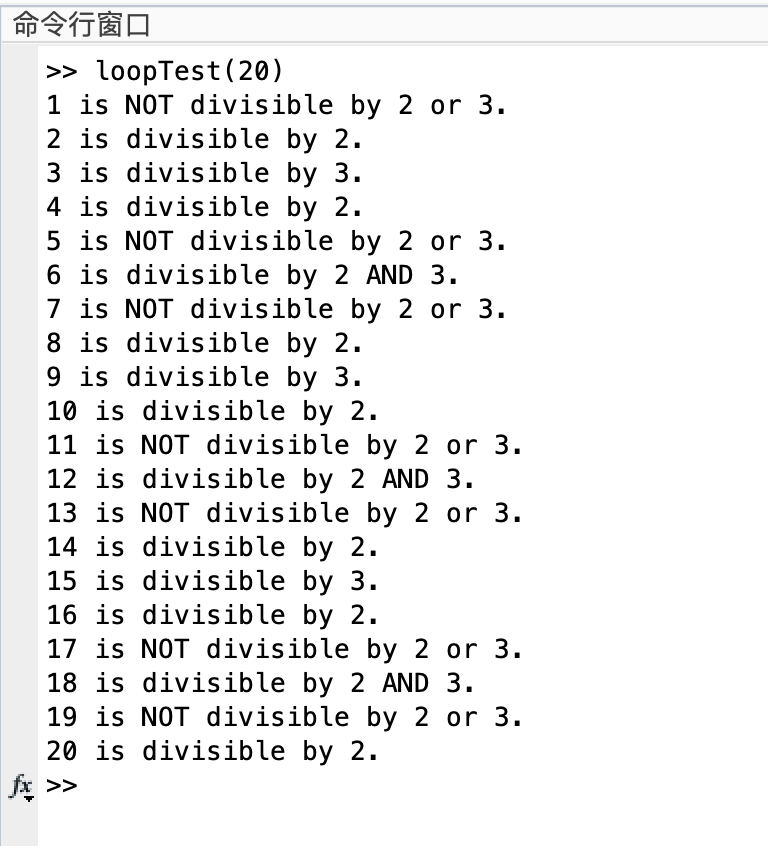
disp([num2str(n) ' is NOT divisible by 2 or 3.'])

end

end

end

实验结果及分析：



实验数据显示函数编写正确

题目：7 Smoothing filter

代码：

A．使用卷积：

function smoothed=rectFilt(x,width)

smoothed = x;

if mod(x, 2) == 0

width = width+1;

disp('width must an odd!')

end

conver = (conv(ones(1,width),x)./width);

smoothed(ceil(width/2):(length(x)-floor(width/2))) = conver(width:end-width+1);

end

B.使用循环

function smoothed=rectFilt(x,width)

smoothed = ones(length(x),1);

if mod(x, 2) == 0

width = width+1;

disp('width must an odd!')

end

for n=ceil(width/2):(length(x)-floor(width/2))

smoothed(n) = mean(x(n-floor(width/2):n+floor(width/2)));

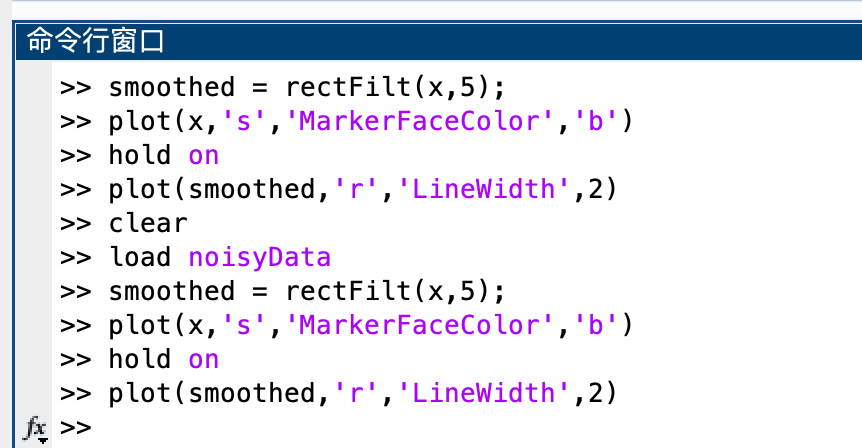
end

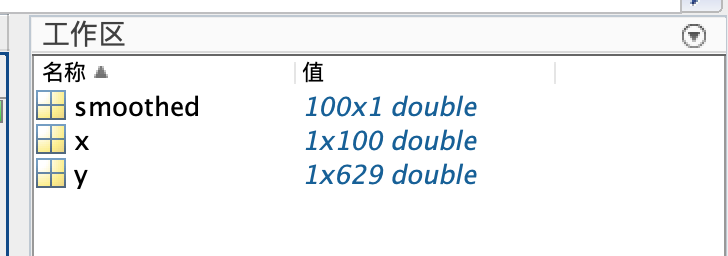
end

实验结果及分析：

图像显示平滑效果良好。

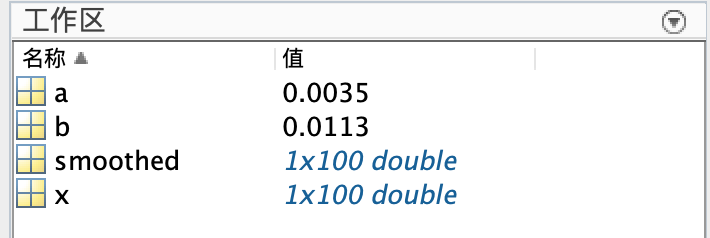
对比两种方法，发现卷积法效率比循环法高很多，所用时间相差一个数量级。

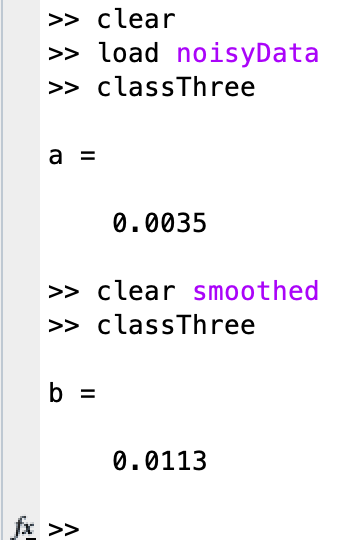




图表, 散点图

描述已自动生成





题目：8 Plot a circle.

代码：

getCircle.m

function [x,y] = getCircle(center,r)

t = 0:0.01:2\*pi;

x = r\*cos(t)+center(1);

y = r\*sin(t)+center(2);

end

concentric.m

clf; clear; clc;

figure(1);

c = jet(5);

[x,y] = getCircle([0,0],5);

plot(x,y,'LineWidth',10,'Color',c(1,:))

hold on

[x,y] = getCircle([0,0],10);

plot(x,y,'LineWidth',7,'Color',c(2,:))

hold on

[x,y] = getCircle([0,0],15);

plot(x,y,'LineWidth',5,'Color',c(3,:))

hold on

[x,y] = getCircle([0,0],20);

plot(x,y,'LineWidth',3,'Color',c(4,:))

hold on

[x,y] = getCircle([0,0],22);

plot(x,y,'LineWidth',2,'Color',c(5,:))

hold on

olympic.m

clf; clear; clc;

figure(2);

[x,y] = getCircle([-1,0],0.4);

plot(x,y,'b','LineWidth',10)

hold on

[x,y] = getCircle([-0.5,-0.5],0.4);

plot(x,y,'y','LineWidth',10)

hold on

[x,y] = getCircle([0,0],0.4);

plot(x,y,'k','LineWidth',10)

hold on

[x,y] = getCircle([0.5,-0.5],0.4);

plot(x,y,'g','LineWidth',10)

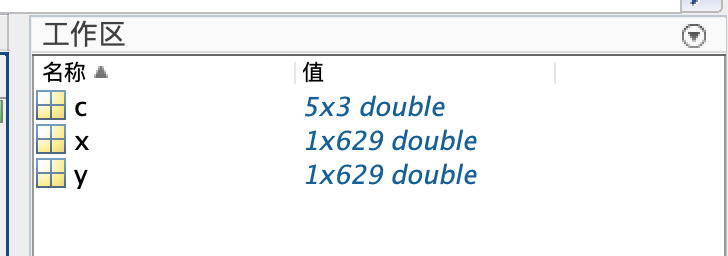
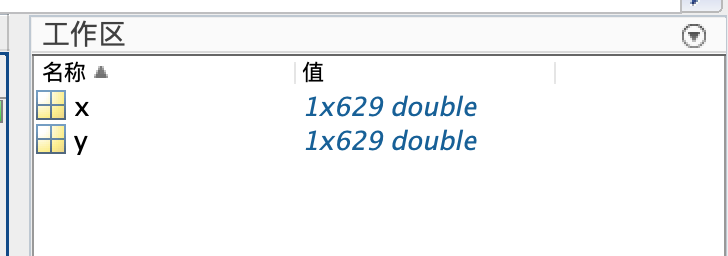
hold on

[x,y] = getCircle([1,0],0.4);

plot(x,y,'r','LineWidth',10)

hold on

实验结果及分析：

图片包含 气泡图

描述已自动生成图片包含 形状

描述已自动生成

可画出图形，函数编写正确

题目：9 Logical indexing and piecewise plots. Transistor(晶体管) I-V curves.

代码：

clear;clc;

K = 50e-6;

Vgs = [0 1 2 3 4 5];

Vds = linspace(0,5,1000);

Vt = ones(1,6);

% Ids = ones(length(Vds),6);

difference = Vgs-Vt;

inds = find(difference<=0);

%difference(inds) = 0;

Ids = K\*(Vds'\*difference-0.5\*(Vds.^2)'\*ones(1,6));

for n=1:6

index = find(Vds>difference(n));

Ids(index,n) = 0.5\*K\*difference(n)^2;

end

Ids(:,inds) = 0;

Ids = 10^6\*Ids;

plot(Vds,Ids,'LineWidth',2)

ylim([0 500])

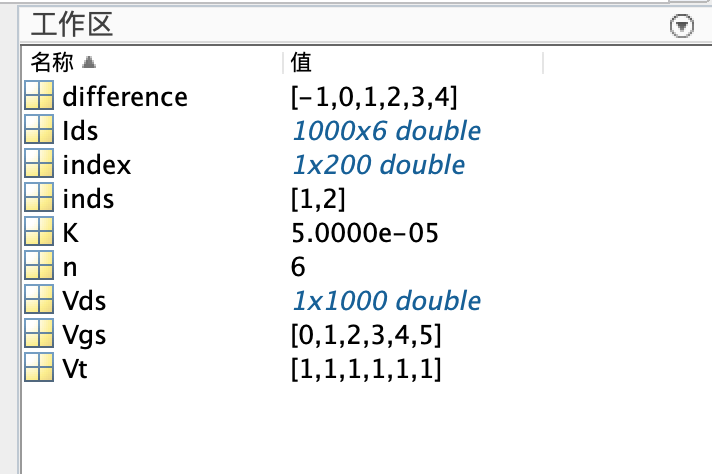
title('Transistor Drain Current')

xlabel('Vds (V)')

ylabel('Ids (ÂµA)')

legend('Vgs=0V','Vgs=1V','Vgs=2V','Vgs=3V','Vgs=4V','Vgs=5V')

实验结果及分析：



图表

描述已自动生成

函数图像符合题设。

题目：Optional Problems：9. Functions.

代码：

clear;clc;

v = 4;

thetas = linspace(0,60,500);

distances = ones(1,500);

for n=1:500

distances(n) = throwBall(v,thetas(n));

end

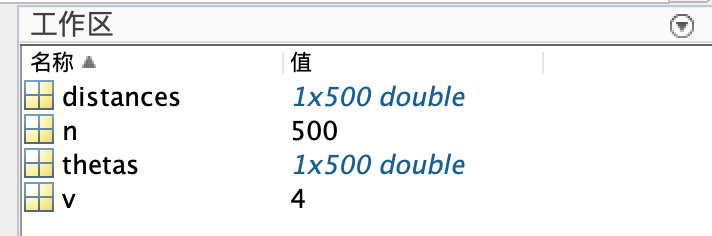
plot(thetas,distances,'r','LineWidth',2)

title('Distance of ball throw as a function of release angle (v=4 m/s)')

xlabel('Initial Angle (deg)')

ylabel('Destance throwm (m)')

实验结果及分析：



图表

描述已自动生成

通过图像可知，当v=4m/s时，倾角约30°时抛出距离越大。

题目：Optional Problems：10. Smoothing nonuniformly sampled data.

代码：

函数：

function smoothed=rectFilt\_m(x,width)

smoothed = x;

unChange = x(:,1);

change = x(:,2);

v=0:width:10;

for n=1:20

ind = find(unChange>(unChange(n)-width/2) & unChange<(unChange(n)+width/2));

change(n) = sum(change(ind))/length(ind);

end

smoothed(:,2) = change;

end

测试代码：

clear;

load optionalData

smoothed = rectFilt\_m(x,2);

plot(x(:,1),x(:,2),'.','MarkerSize',20)

hold on

plot(smoothed(:,1),smoothed(:,2),'--','LineWidth',3)

vq = interp1(smoothed(:,1),smoothed(:,2),0:0.1:10);

plot(0:0.1:10,vq,'k')

实验结果及分析：

实验结果与题目基本一致，可验证函数编写正确

图表, 折线图

描述已自动生成

题目：Optional Problems：11. Buy and sell a stock.

代码：

function endValue=tradeStock(initialInvestment, price, buy, sell)

transactionCost = 12.95;

stocks = 0;

endValue = initialInvestment;

for n=1:length(price)

if ~isempty(find(sell==n, 1)) && stocks~=0

if endValue>=transactionCost

endValue = endValue + stocks\*price(n) - transactionCost;

stocks = 0;

end

elseif ~isempty(find(buy==n, 1))

if endValue>=transactionCost + price(n)

if mod((endValue-transactionCost),price(n)) >= transactionCost

stocks = stocks+floor((endValue-transactionCost)/price(n));

endValue = mod((endValue-transactionCost),price(n));

else

stocks = stocks+floor((endValue-transactionCost)/price(n))-1;

endValue = mod((endValue-transactionCost),price(n))+price(n);

end

end

end

end

endValue = endValue + stocks\*price(end);

end

实验结果及分析：

实验结果表明函数符合题设要求，测试数据正确。

