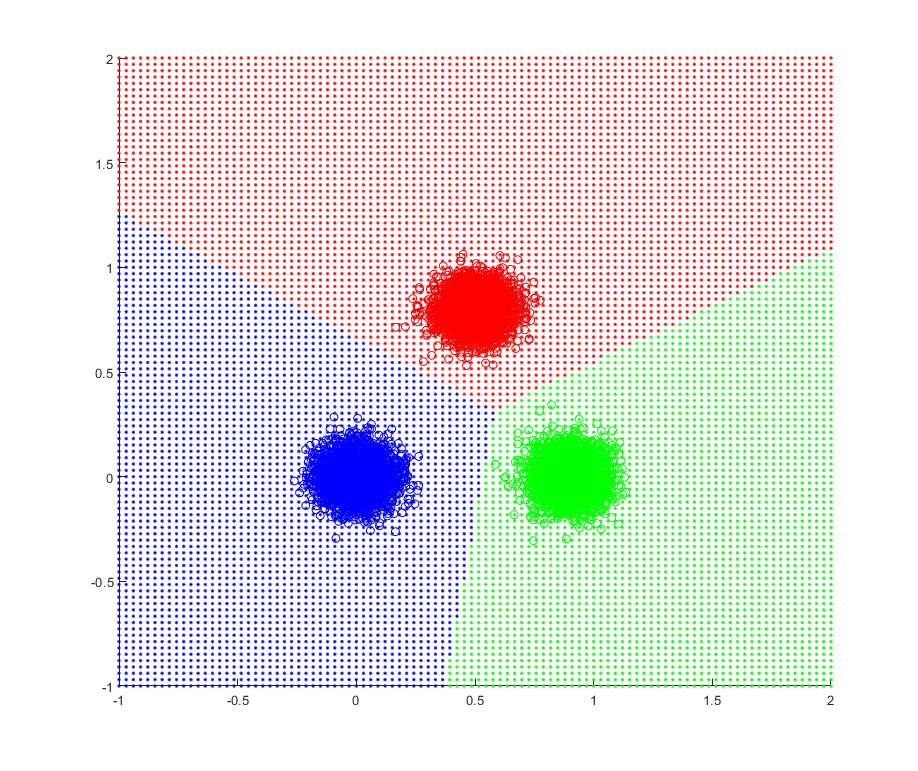
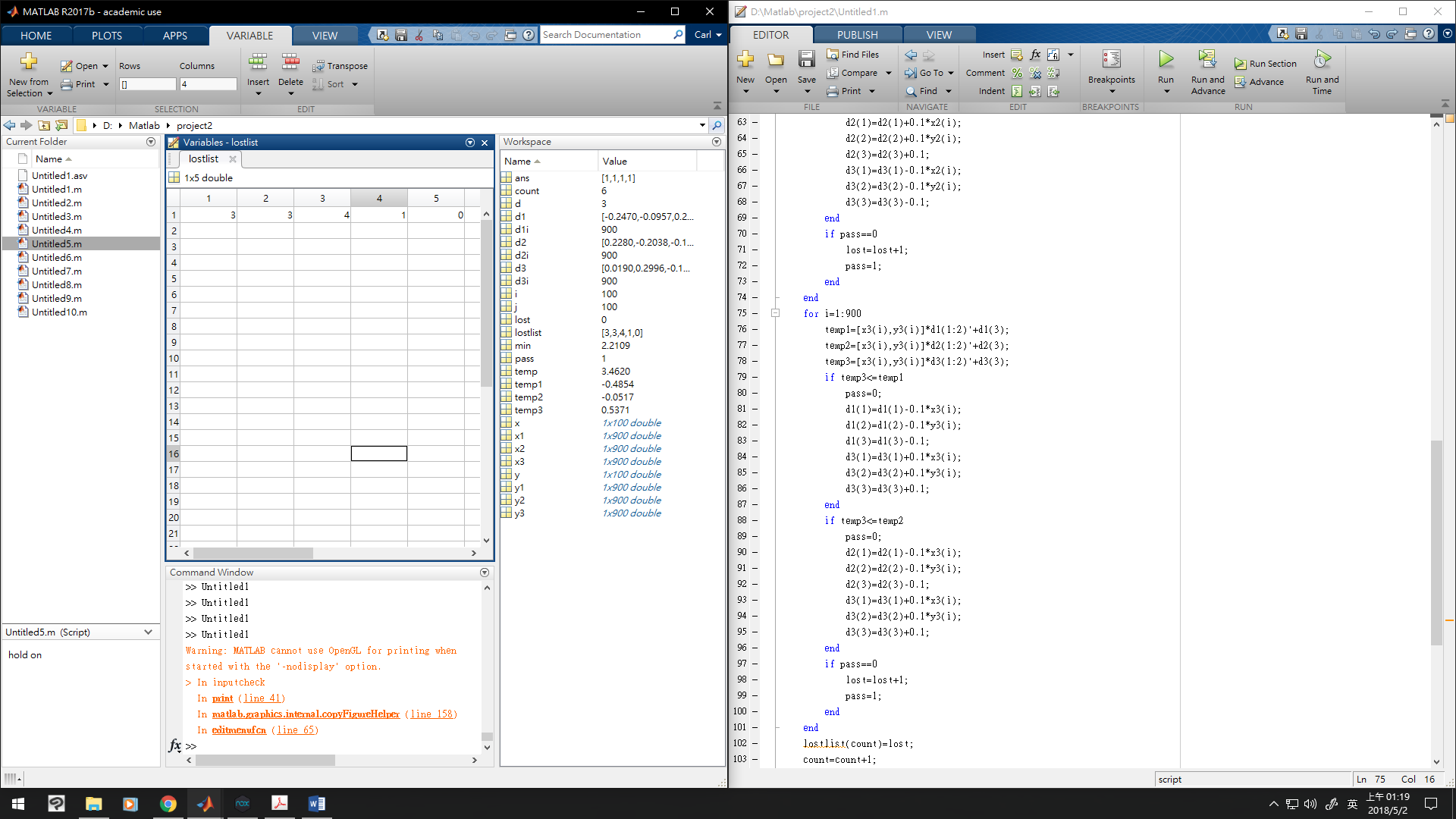
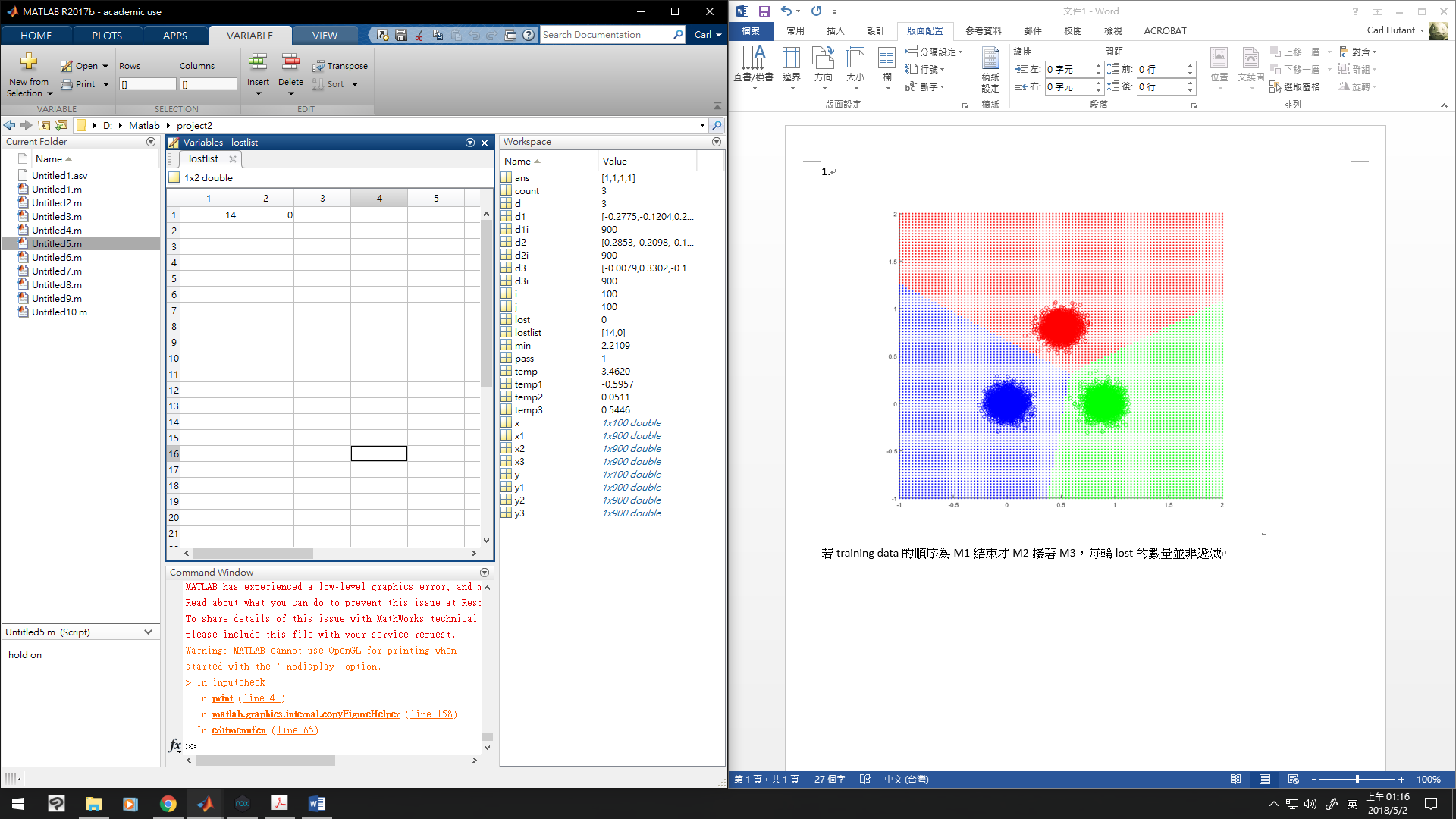
1.a multi-class linear discriminant functions



若training data的順序為M1結束才M2接著M3，每輪lost的數量並非遞減，但數量較低



順序為交錯，則每輪數量較多，但花費較少輪就完成



分割線及分割的區域會交於一點

x1=random('Normal',0,0.08,1,900);

y1=random('Normal',0,0.08,1,900);

x2=random('Normal',0.9,0.08,1,900);

y2=random('Normal',0,0.08,1,900);

x3=random('Normal',0.5,0.08,1,900);

y3=random('Normal',0.8,0.08,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

if temp1<=temp2

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp1<=temp3

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

for i=1:900

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

if temp2<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp2<=temp3

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

for i=1:900

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

if temp3<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if temp3<=temp2

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=temp2 && temp1>=temp3

plot(x(i),y(j),'.b');

elseif temp2>=temp3

plot(x(i),y(j),'.g');

else

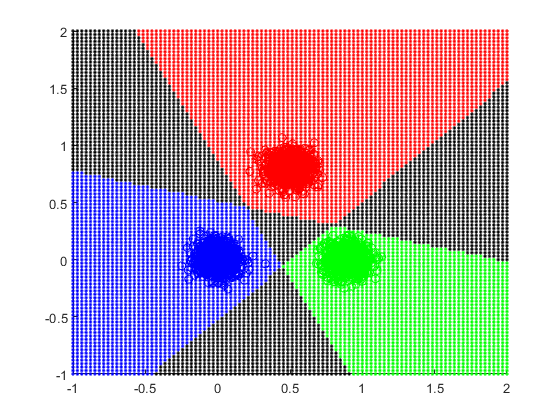
plot(x(i),y(j),'.r');

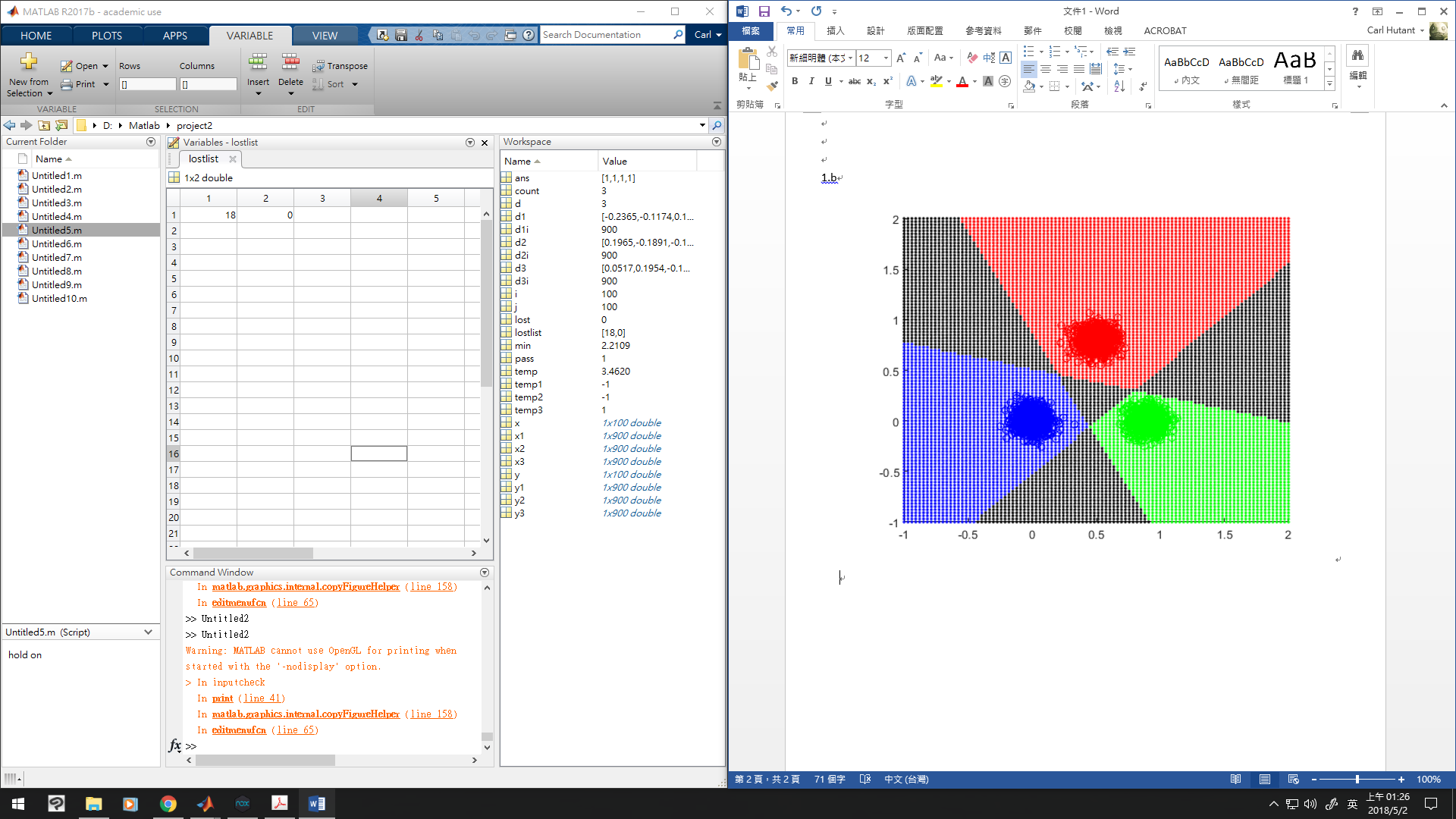
end

end

end

1.b multi-class linear perceptrons (hard limiter function)



分割線沒有交於一點，在不改變training model的條件下無法解決此問題，因為只要分割線相交就會產生同時被歸類為兩者的區域極不被歸類為任意者的區域，更平均分布的training data與提升function order只是減少模糊地帶的面積，除非能使分割線彼此成為漸近的關係也許能解決，但終究會有不被歸類為任意者的區域。

x1=random('Normal',0,0.08,1,900);

y1=random('Normal',0,0.08,1,900);

x2=random('Normal',0.9,0.08,1,900);

y2=random('Normal',0,0.08,1,900);

x3=random('Normal',0.5,0.08,1,900);

y3=random('Normal',0.8,0.08,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==-1

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

end

if temp2==-1

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

end

if temp3==-1

pass=0;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1 && temp2==-1 && temp3==-1

plot(x(i),y(j),'.b');

elseif temp1==-1 && temp2==1 && temp3==-1

plot(x(i),y(j),'.g');

elseif temp1==-1 && temp2==-1 && temp3==1

plot(x(i),y(j),'.r');

else

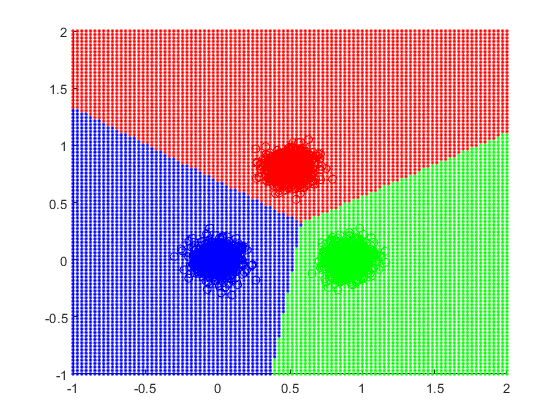
plot(x(i),y(j),'.k');

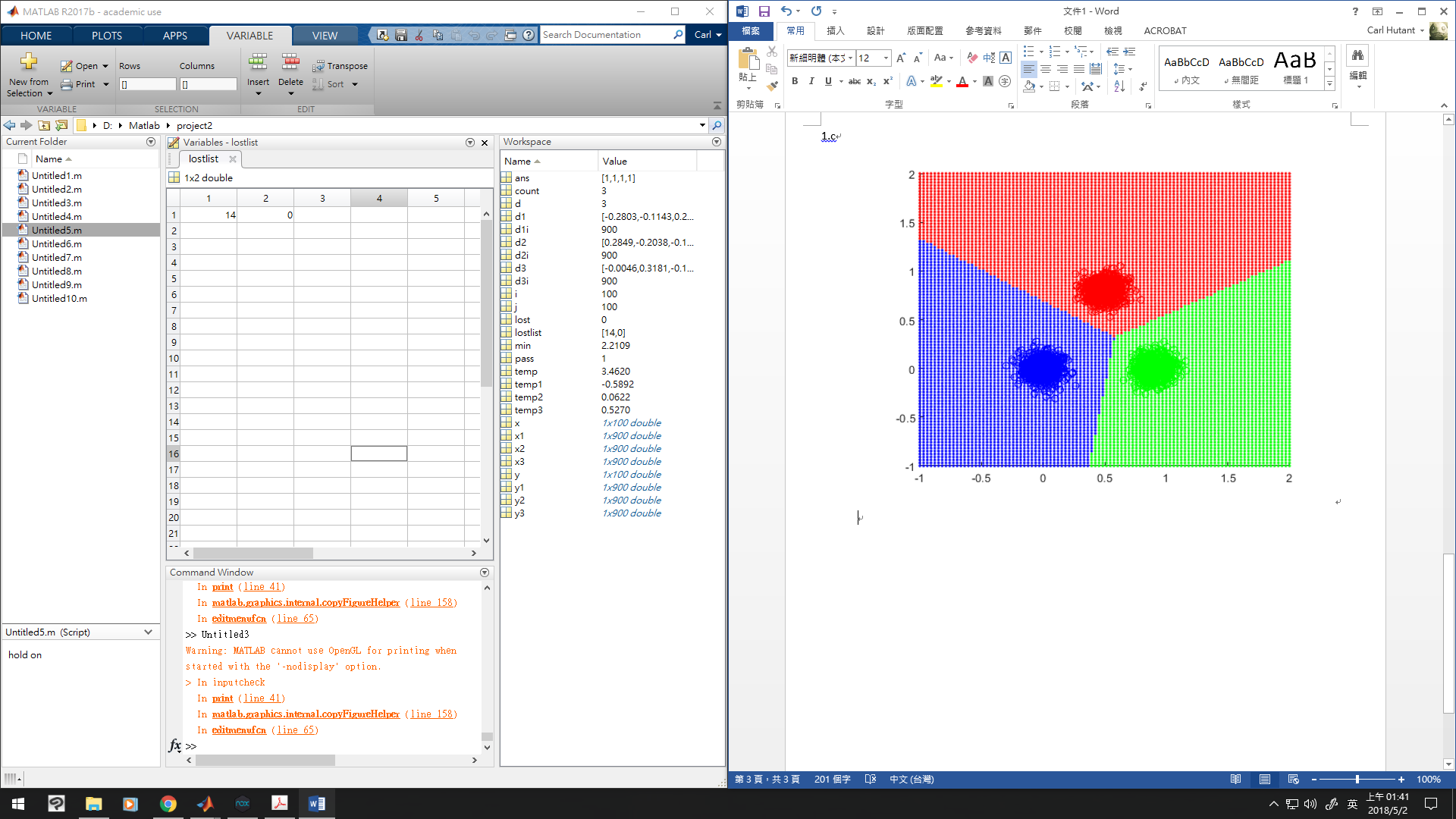
end

end

end

1.c multi-class linear perceptrons (sigmoidal function)



到目前為止，大多數的方法都會在一輪training後就收斂，應該是因為不同class距離遠的緣故，分割的區域有交於一點

x1=random('Normal',0,0.08,1,900);

y1=random('Normal',0,0.08,1,900);

x2=random('Normal',0.9,0.08,1,900);

y2=random('Normal',0,0.08,1,900);

x3=random('Normal',0.5,0.08,1,900);

y3=random('Normal',0.8,0.08,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp1<=temp2

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp1<=temp3

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp2<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp2<=temp3

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp3<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if temp3<=temp2

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=temp2 && temp1>=temp3

plot(x(i),y(j),'.b');

elseif temp2>=temp3

plot(x(i),y(j),'.g');

else

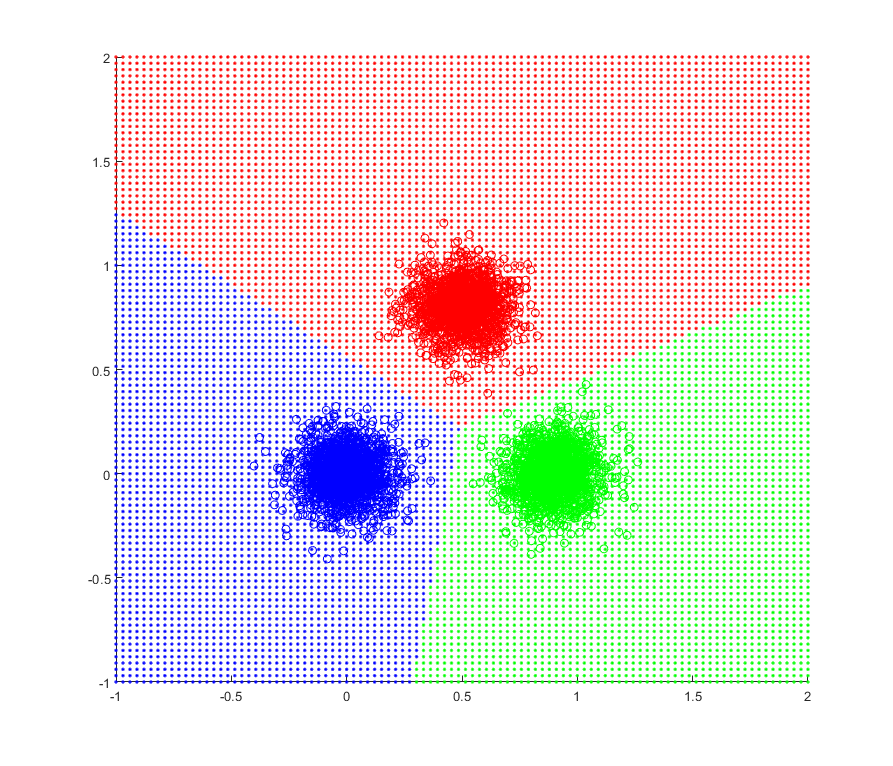
plot(x(i),y(j),'.r');

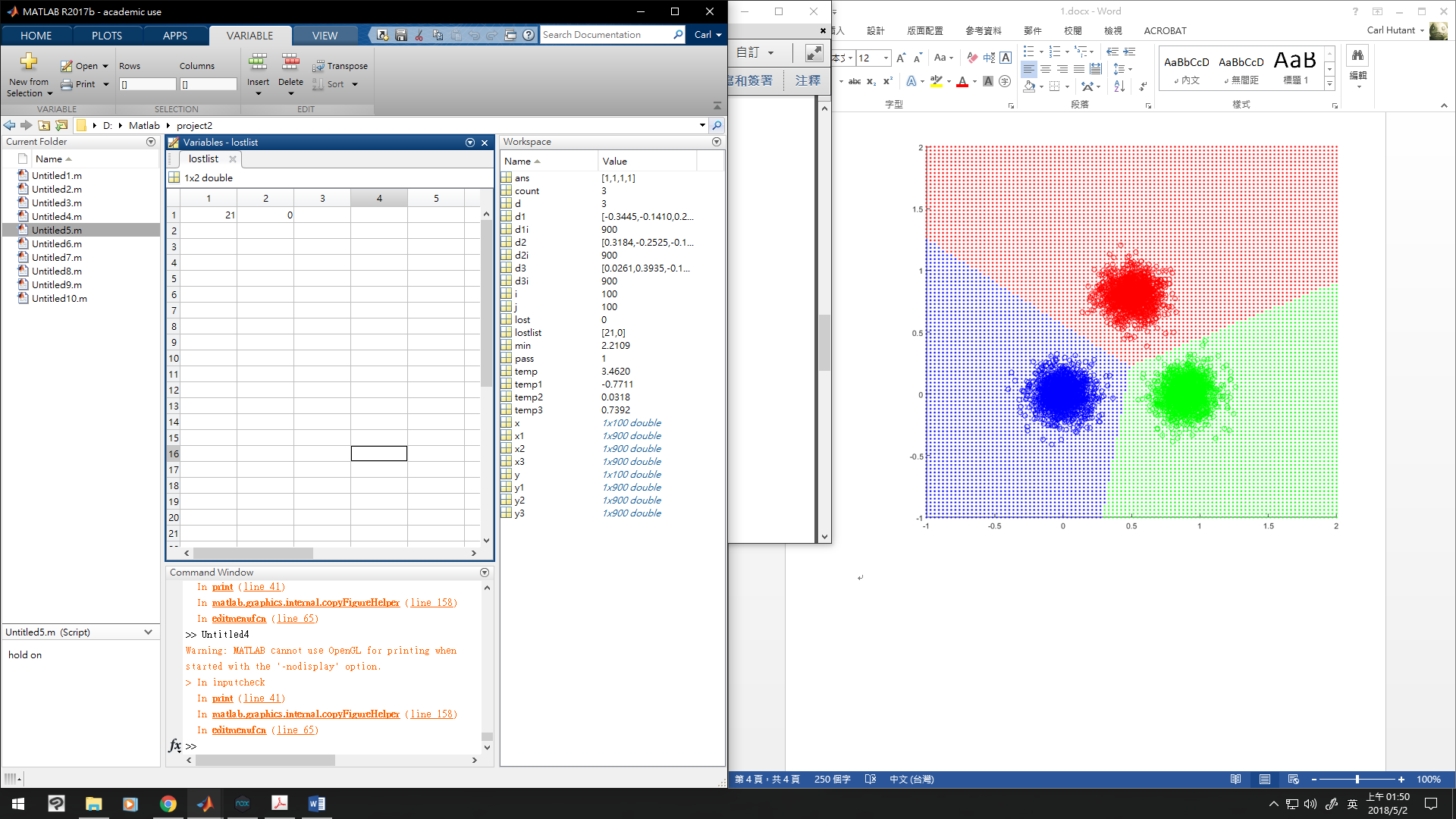
end

end

end

2.a multi-class linear discriminant functions





x1=random('Normal',0,0.12,1,900);

y1=random('Normal',0,0.12,1,900);

x2=random('Normal',0.9,0.12,1,900);

y2=random('Normal',0,0.12,1,900);

x3=random('Normal',0.5,0.12,1,900);

y3=random('Normal',0.8,0.12,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

if temp1<=temp2

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp1<=temp3

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

if temp2<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp2<=temp3

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

if temp3<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if temp3<=temp2

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=temp2 && temp1>=temp3

plot(x(i),y(j),'.b');

elseif temp2>=temp3

plot(x(i),y(j),'.g');

else

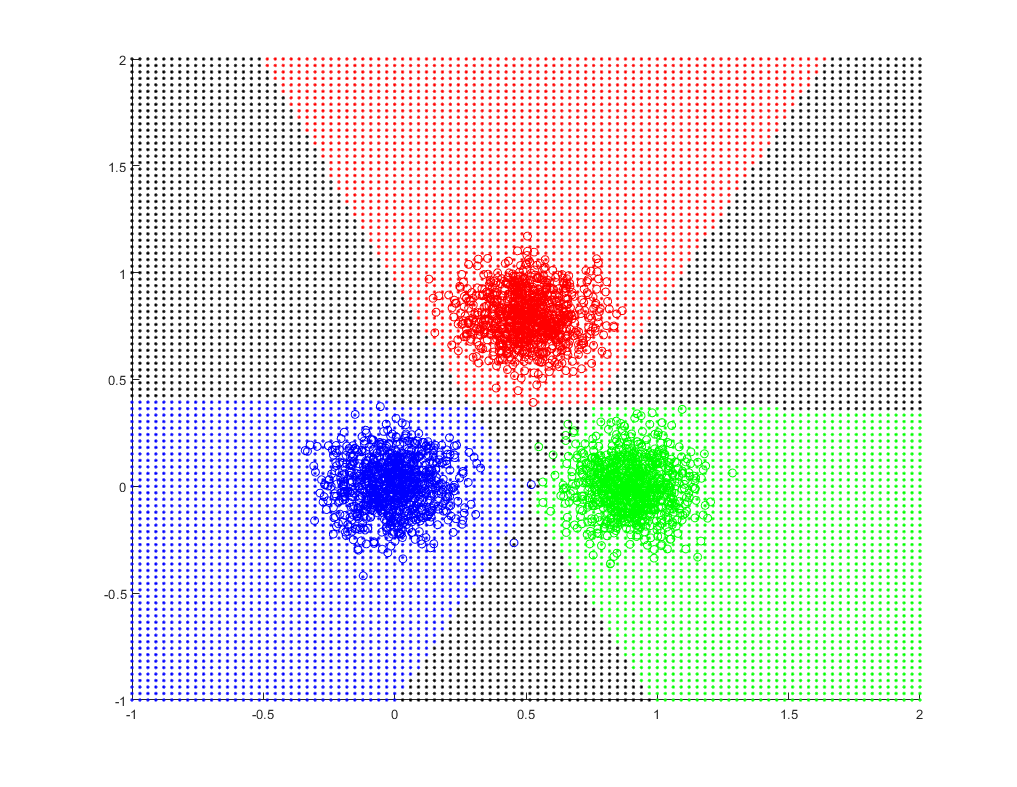
plot(x(i),y(j),'.r');

end

end

end

2.b multi-class linear perceptrons (hard limiter function)



在100輪之後仍然沒有收斂，以下是lost

47 27 14 7 16 14 6 12 6 6 6 6 16 12 13 13 9 7 6 6 6 4 6 6 6 8 6 6 6 6 6 6 6 6 8 6 6 6 6 6 6 6 6 8 6 6 6 6 6 6 6 8 6 6 6 6 6 6 6 8 6 6 6 6 6 6 6 8 6 6 6 6 6 6 8 6 6 6 6 6 6 9 11 6 6 6 6 6 6 6 6 6 6 9 7 7 6 6 4 6

x1=random('Normal',0,0.12,1,900);

y1=random('Normal',0,0.12,1,900);

x2=random('Normal',0.9,0.12,1,900);

y2=random('Normal',0,0.12,1,900);

x3=random('Normal',0.5,0.12,1,900);

y3=random('Normal',0.8,0.12,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==-1

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

end

if temp2==-1

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

end

if temp3==-1

pass=0;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1 && temp2==-1 && temp3==-1

plot(x(i),y(j),'.b');

elseif temp1==-1 && temp2==1 && temp3==-1

plot(x(i),y(j),'.g');

elseif temp1==-1 && temp2==-1 && temp3==1

plot(x(i),y(j),'.r');

else

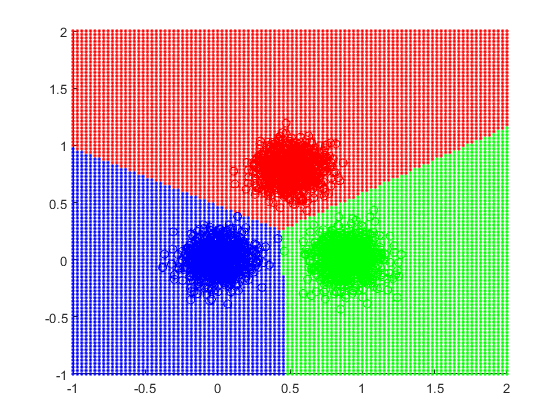
plot(x(i),y(j),'.k');

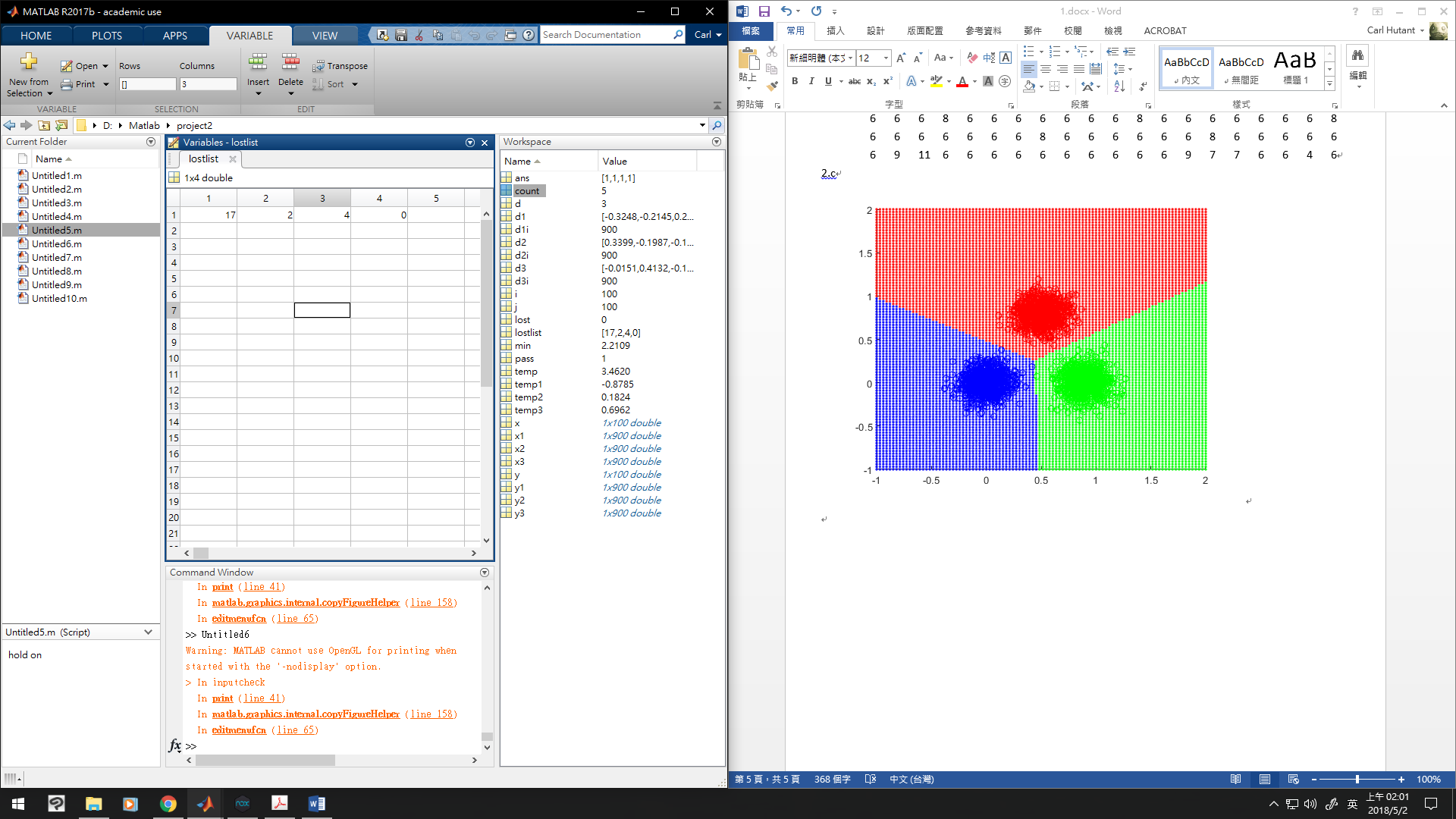
end

end

end

2.c multi-class linear perceptrons (sigmoidal function)



改用sigmoidal function後就可收斂

x1=random('Normal',0,0.12,1,900);

y1=random('Normal',0,0.12,1,900);

x2=random('Normal',0.9,0.12,1,900);

y2=random('Normal',0,0.12,1,900);

x3=random('Normal',0.5,0.12,1,900);

y3=random('Normal',0.8,0.12,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0];

d2=[0,0,0];

d3=[0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i),y1(i)]\*d1(1:2)'+d1(3);

temp2=[x1(i),y1(i)]\*d2(1:2)'+d2(3);

temp3=[x1(i),y1(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp1<=temp2

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d2(1)=d2(1)-0.1\*x1(i);

d2(2)=d2(2)-0.1\*y1(i);

d2(3)=d2(3)-0.1;

end

if temp1<=temp3

pass=0;

d1(1)=d1(1)+0.1\*x1(i);

d1(2)=d1(2)+0.1\*y1(i);

d1(3)=d1(3)+0.1;

d3(1)=d3(1)-0.1\*x1(i);

d3(2)=d3(2)-0.1\*y1(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i),y2(i)]\*d1(1:2)'+d1(3);

temp2=[x2(i),y2(i)]\*d2(1:2)'+d2(3);

temp3=[x2(i),y2(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp2<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x2(i);

d1(2)=d1(2)-0.1\*y2(i);

d1(3)=d1(3)-0.1;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

end

if temp2<=temp3

pass=0;

d2(1)=d2(1)+0.1\*x2(i);

d2(2)=d2(2)+0.1\*y2(i);

d2(3)=d2(3)+0.1;

d3(1)=d3(1)-0.1\*x2(i);

d3(2)=d3(2)-0.1\*y2(i);

d3(3)=d3(3)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i),y3(i)]\*d1(1:2)'+d1(3);

temp2=[x3(i),y3(i)]\*d2(1:2)'+d2(3);

temp3=[x3(i),y3(i)]\*d3(1:2)'+d3(3);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp3<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x3(i);

d1(2)=d1(2)-0.1\*y3(i);

d1(3)=d1(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if temp3<=temp2

pass=0;

d2(1)=d2(1)-0.1\*x3(i);

d2(2)=d2(2)-0.1\*y3(i);

d2(3)=d2(3)-0.1;

d3(1)=d3(1)+0.1\*x3(i);

d3(2)=d3(2)+0.1\*y3(i);

d3(3)=d3(3)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i),y(j)]\*d1(1:2)'+d1(3);

temp2=[x(i),y(j)]\*d2(1:2)'+d2(3);

temp3=[x(i),y(j)]\*d3(1:2)'+d3(3);

if temp1>=temp2 && temp1>=temp3

plot(x(i),y(j),'.b');

elseif temp2>=temp3

plot(x(i),y(j),'.g');

else

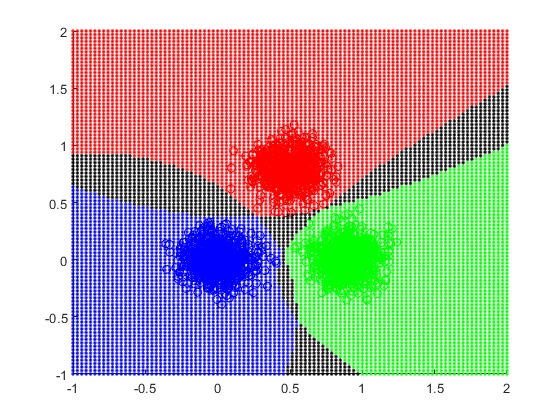
plot(x(i),y(j),'.r');

end

end

end

2.d multi-class non-linear perceptrons (hard limiter / sigmoidal function)



使用hard limiter function雖然有解但會有無法分辨的區域，lost: 37 5 2 2 2 2 8 8 6 4 4 4 4 4 4 3 2 2 2 3 4 4 3 2 2 2 2 2 2 2 2 0

x1=random('Normal',0,0.12,1,900);

y1=random('Normal',0,0.12,1,900);

x2=random('Normal',0.9,0.12,1,900);

y2=random('Normal',0,0.12,1,900);

x3=random('Normal',0.5,0.12,1,900);

y3=random('Normal',0.8,0.12,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0,0,0];

d2=[0,0,0,0,0];

d3=[0,0,0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d1(1:4)'+d1(5);

temp2=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d2(1:4)'+d2(5);

temp3=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d3(1:4)'+d3(5);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==-1

pass=0;

d1(1)=d1(1)+0.1\*x1(i)^2;

d1(2)=d1(2)+0.1\*y1(i)^2;

d1(3)=d1(3)+0.1\*x1(i);

d1(4)=d1(4)+0.1\*y1(i);

d1(5)=d1(5)+0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x1(i)^2;

d2(2)=d2(2)-0.1\*y1(i)^2;

d2(3)=d2(3)-0.1\*x1(i);

d2(4)=d2(4)-0.1\*y1(i);

d2(5)=d2(5)-0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x1(i)^2;

d3(2)=d3(2)-0.1\*y1(i)^2;

d3(3)=d3(3)-0.1\*x1(i);

d3(4)=d3(4)-0.1\*y1(i);

d3(5)=d3(5)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d1(1:4)'+d1(5);

temp2=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d2(1:4)'+d2(5);

temp3=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d3(1:4)'+d3(5);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x2(i)^2;

d1(2)=d1(2)-0.1\*y2(i)^2;

d1(3)=d1(3)-0.1\*x2(i);

d1(4)=d1(4)-0.1\*y2(i);

d1(5)=d1(5)-0.1;

end

if temp2==-1

pass=0;

d2(1)=d2(1)+0.1\*x2(i)^2;

d2(2)=d2(2)+0.1\*y2(i)^2;

d2(3)=d2(3)+0.1\*x2(i);

d2(4)=d2(4)+0.1\*y2(i);

d2(5)=d2(5)+0.1;

end

if temp3==1

pass=0;

d3(1)=d3(1)-0.1\*x2(i)^2;

d3(2)=d3(2)-0.1\*y2(i)^2;

d3(3)=d3(3)-0.1\*x2(i);

d3(4)=d3(4)-0.1\*y2(i);

d3(5)=d3(5)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d1(1:4)'+d1(5);

temp2=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d2(1:4)'+d2(5);

temp3=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d3(1:4)'+d3(5);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1

pass=0;

d1(1)=d1(1)-0.1\*x3(i)^2;

d1(2)=d1(2)-0.1\*y3(i)^2;

d1(3)=d1(3)-0.1\*x3(i);

d1(4)=d1(4)-0.1\*y3(i);

d1(5)=d1(5)-0.1;

end

if temp2==1

pass=0;

d2(1)=d2(1)-0.1\*x3(i)^2;

d2(2)=d2(2)-0.1\*y3(i)^2;

d2(3)=d2(3)-0.1\*x3(i);

d2(4)=d2(4)-0.1\*y3(i);

d2(5)=d2(5)-0.1;

end

if temp3==-1

pass=0;

d3(1)=d3(1)+0.1\*x3(i)^2;

d3(2)=d3(2)+0.1\*y3(i)^2;

d3(3)=d3(3)+0.1\*x3(i);

d3(4)=d3(4)+0.1\*y3(i);

d3(5)=d3(5)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i)^2,y(j)^2,x(i),y(j)]\*d1(1:4)'+d1(5);

temp2=[x(i)^2,y(j)^2,x(i),y(j)]\*d2(1:4)'+d2(5);

temp3=[x(i)^2,y(j)^2,x(i),y(j)]\*d3(1:4)'+d3(5);

if temp1>=0

temp1=1;

else

temp1=-1;

end

if temp2>=0

temp2=1;

else

temp2=-1;

end

if temp3>=0

temp3=1;

else

temp3=-1;

end

if temp1==1 && temp2==-1 && temp3==-1

plot(x(i),y(j),'.b');

elseif temp1==-1 && temp2==1 && temp3==-1

plot(x(i),y(j),'.g');

elseif temp1==-1 && temp2==-1 && temp3==1

plot(x(i),y(j),'.r');

else

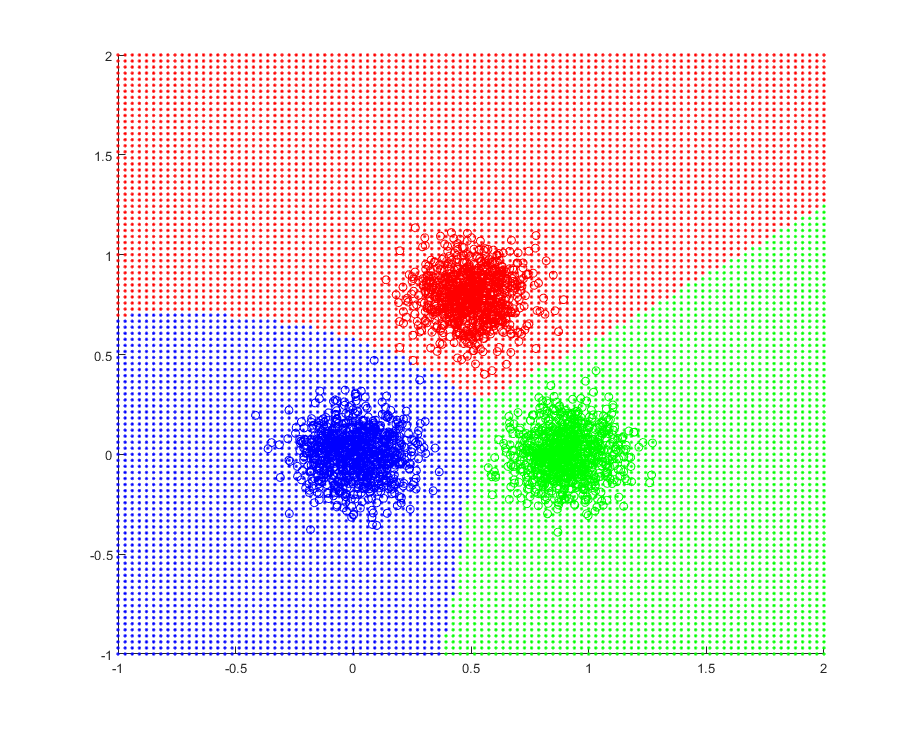
plot(x(i),y(j),'.k');

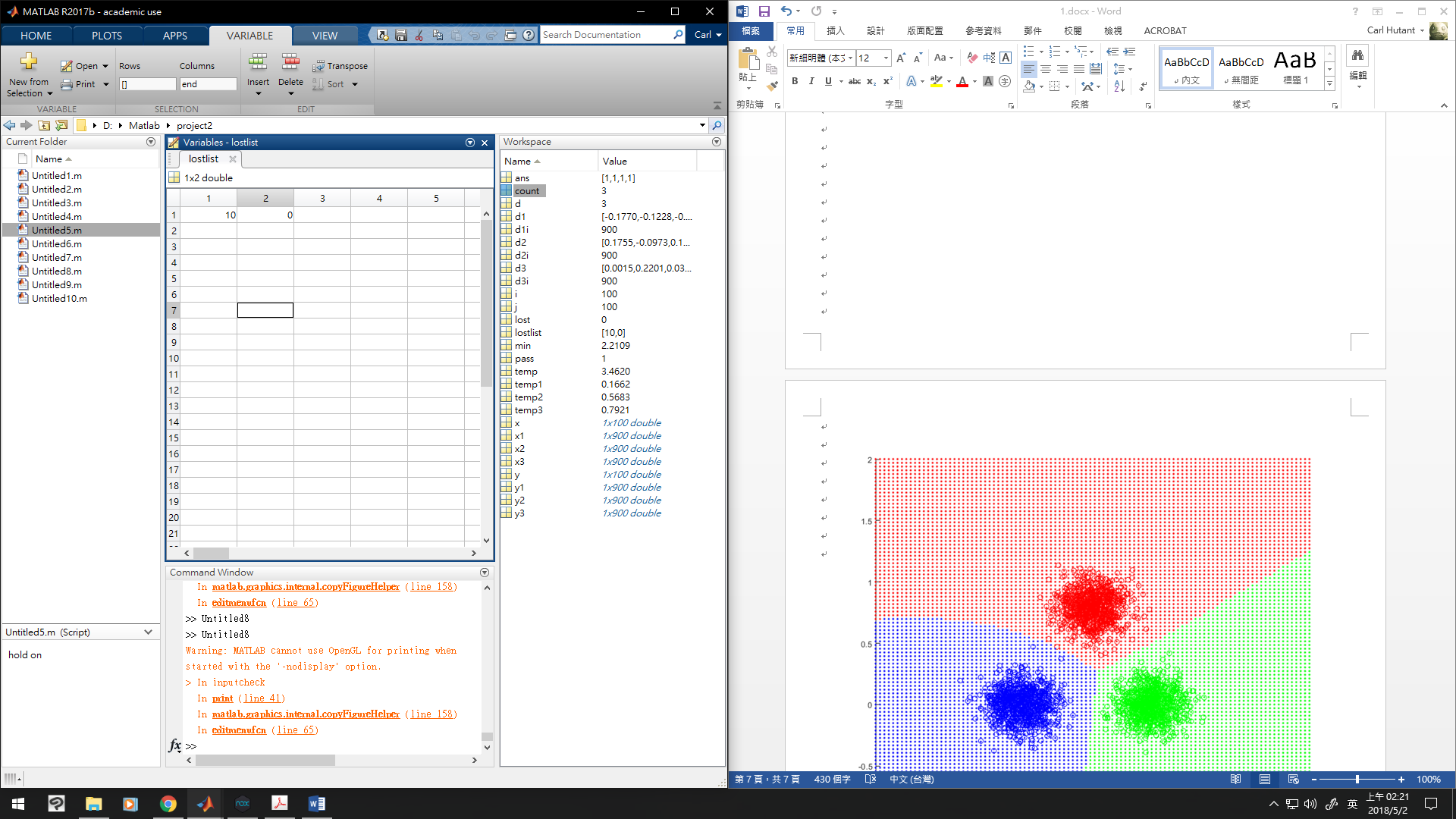
end

end

end

hold off



使用sigmoidal function會解決無法分辨區域的問題

x1=random('Normal',0,0.12,1,900);

y1=random('Normal',0,0.12,1,900);

x2=random('Normal',0.9,0.12,1,900);

y2=random('Normal',0,0.12,1,900);

x3=random('Normal',0.5,0.12,1,900);

y3=random('Normal',0.8,0.12,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

d1=[0,0,0,0,0];

d2=[0,0,0,0,0];

d3=[0,0,0,0,0];

pass=0;

count=1;

lost=1;

lostlist=0;

while lost~=0 && count<=100

lost=0;

pass=1;

for i=1:900

temp1=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d1(1:4)'+d1(5);

temp2=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d2(1:4)'+d2(5);

temp3=[x1(i)^2,y1(i)^2,x1(i),y1(i)]\*d3(1:4)'+d3(5);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp1<=temp2

pass=0;

d1(1)=d1(1)+0.1\*x1(i)^2;

d1(2)=d1(2)+0.1\*y1(i)^2;

d1(3)=d1(3)+0.1\*x1(i);

d1(4)=d1(4)+0.1\*y1(i);

d1(5)=d1(5)+0.1;

d2(1)=d2(1)-0.1\*x1(i)^2;

d2(2)=d2(2)-0.1\*y1(i)^2;

d2(3)=d2(3)-0.1\*x1(i);

d2(4)=d2(4)-0.1\*y1(i);

d2(5)=d2(5)-0.1;

end

if temp1<=temp3

pass=0;

d1(1)=d1(1)+0.1\*x1(i)^2;

d1(2)=d1(2)+0.1\*y1(i)^2;

d1(3)=d1(3)+0.1\*x1(i);

d1(4)=d1(4)+0.1\*y1(i);

d1(5)=d1(5)+0.1;

d3(1)=d3(1)-0.1\*x1(i)^2;

d3(2)=d3(2)-0.1\*y1(i)^2;

d3(3)=d3(3)-0.1\*x1(i);

d3(4)=d3(4)-0.1\*y1(i);

d3(5)=d3(5)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d1(1:4)'+d1(5);

temp2=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d2(1:4)'+d2(5);

temp3=[x2(i)^2,y2(i)^2,x2(i),y2(i)]\*d3(1:4)'+d3(5);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp2<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x2(i)^2;

d1(2)=d1(2)-0.1\*y2(i)^2;

d1(3)=d1(3)-0.1\*x2(i);

d1(4)=d1(4)-0.1\*y2(i);

d1(5)=d1(5)-0.1;

d2(1)=d2(1)+0.1\*x2(i)^2;

d2(2)=d2(2)+0.1\*y2(i)^2;

d2(3)=d2(3)+0.1\*x2(i);

d2(4)=d2(4)+0.1\*y2(i);

d2(5)=d2(5)+0.1;

end

if temp2<=temp3

pass=0;

d2(1)=d2(1)+0.1\*x2(i)^2;

d2(2)=d2(2)+0.1\*y2(i)^2;

d2(3)=d2(3)+0.1\*x2(i);

d2(4)=d2(4)+0.1\*y2(i);

d2(5)=d2(5)+0.1;

d3(1)=d3(1)-0.1\*x2(i)^2;

d3(2)=d3(2)-0.1\*y2(i)^2;

d3(3)=d3(3)-0.1\*x2(i);

d3(4)=d3(4)-0.1\*y2(i);

d3(5)=d3(5)-0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

temp1=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d1(1:4)'+d1(5);

temp2=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d2(1:4)'+d2(5);

temp3=[x3(i)^2,y3(i)^2,x3(i),y3(i)]\*d3(1:4)'+d3(5);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp3<=temp1

pass=0;

d1(1)=d1(1)-0.1\*x3(i)^2;

d1(2)=d1(2)-0.1\*y3(i)^2;

d1(3)=d1(3)-0.1\*x3(i);

d1(4)=d1(4)-0.1\*y3(i);

d1(5)=d1(5)-0.1;

d3(1)=d3(1)+0.1\*x3(i)^2;

d3(2)=d3(2)+0.1\*y3(i)^2;

d3(3)=d3(3)+0.1\*x3(i);

d3(4)=d3(4)+0.1\*y3(i);

d3(5)=d3(5)+0.1;

end

if temp3<=temp2

pass=0;

d2(1)=d2(1)-0.1\*x3(i)^2;

d2(2)=d2(2)-0.1\*y3(i)^2;

d2(3)=d2(3)-0.1\*x3(i);

d2(4)=d2(4)-0.1\*y3(i);

d2(5)=d2(5)-0.1;

d3(1)=d3(1)+0.1\*x3(i)^2;

d3(2)=d3(2)+0.1\*y3(i)^2;

d3(3)=d3(3)+0.1\*x3(i);

d3(4)=d3(4)+0.1\*y3(i);

d3(5)=d3(5)+0.1;

end

if pass==0

lost=lost+1;

pass=1;

end

end

lostlist(count)=lost;

count=count+1;

end

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

temp1=[x(i)^2,y(j)^2,x(i),y(j)]\*d1(1:4)'+d1(5);

temp2=[x(i)^2,y(j)^2,x(i),y(j)]\*d2(1:4)'+d2(5);

temp3=[x(i)^2,y(j)^2,x(i),y(j)]\*d3(1:4)'+d3(5);

temp1=1/(1+exp(-temp1));

temp2=1/(1+exp(-temp2));

temp3=1/(1+exp(-temp3));

if temp1>=temp2 && temp1>=temp3

plot(x(i),y(j),'.b');

elseif temp2>=temp3

plot(x(i),y(j),'.g');

else

plot(x(i),y(j),'.r');

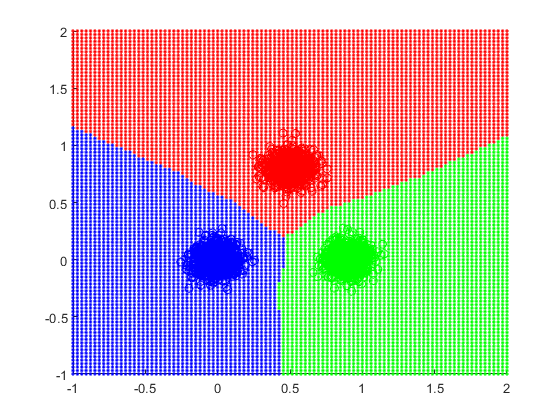
end

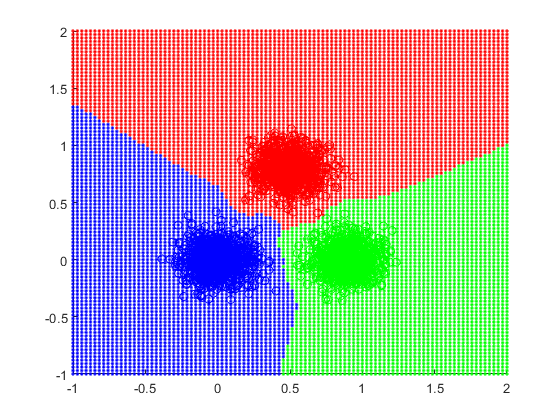
end

end

hold off

3.





分類的實作是使用最簡單的算法，並沒有實作加速的部分

x1=random('Normal',0,0.08,1,900);

y1=random('Normal',0,0.08,1,900);

x2=random('Normal',0.9,0.08,1,900);

y2=random('Normal',0,0.08,1,900);

x3=random('Normal',0.5,0.08,1,900);

y3=random('Normal',0.8,0.08,1,900);

% hold on

% plot(x1,y1,'.b');

% plot(x2,y2,'.g');

% plot(x3,y3,'.r');

hold on

plot(x1,y1,'ob');

plot(x2,y2,'og');

plot(x3,y3,'or');

x=linspace(-1,2,100);

y=linspace(-1,2,100);

for i=1:100

for j=1:100

min=100;

d=0;

for d1i=1:900

temp=(x(i)-x1(d1i))^2+(y(j)-y1(d1i))^2;

if min>temp

min=temp;

d=1;

end

end

for d2i=1:900

temp=(x(i)-x2(d2i))^2+(y(j)-y2(d2i))^2;

if min>temp

min=temp;

d=2;

end

end

for d3i=1:900

temp=(x(i)-x3(d3i))^2+(y(j)-y3(d3i))^2;

if min>temp

min=temp;

d=3;

end

end

if d==1

plot(x(i),y(j),'.b');

elseif d==2

plot(x(i),y(j),'.g');

else

plot(x(i),y(j),'.r');

end

end

end