

# Letter dataset

Code:

```
clear;
A = importdata('letter-recognition.data.xlsx');

set=A.data;
label=A.textdata;

tp= 50; %percent in training set

sigma=1;
c=10;
e=10^-6;

k=1;
for i=1:length(label)
    if(strcmp(label(i,1),'A')==1)
        class(k,1)= 1;
        new_set(k,:)=set(i,:);
        k=k+1;
    elseif(strcmp(label(i,1),'B')==1)
        class(k,1)= 2;
        new_set(k,:)=set(i,:);
        k=k+1;
    elseif(strcmp(label(i,1),'C')==1)
        class(k,1)= 3;
        new_set(k,:)=set(i,:);
        k=k+1;
    end
end

new_set(:,17)=class;

new_set(randperm(length(new_set)),:)=new_set;

tpa= ceil(length(new_set)*tp/100);

X_train=new_set(1:tpa,1:16);
train_label= new_set(1:tpa,17);

X_test= new_set(tpa+1:length(new_set),1:16);
test_label= new_set(tpa+1:length(new_set),17);

alpha_a=ones(tpa+1,1);
alpha_b=ones(tpa+1,1);
alpha_c=ones(tpa+1,1);

for i=1:length(train_label)
    if(train_label(i,1)==1)
        Y_a(i,1)= 1;
    end
end
```

```

        Y_b(i,1)= -1;
        Y_c(i,1)= -1;
    elseif(train_label(i,1)==2)
        Y_a(i,1)= -1;
        Y_b(i,1)= 1;
        Y_c(i,1)= -1;
    else
        Y_a(i,1)= -1;
        Y_b(i,1)= -1;
        Y_c(i,1)= 1;
    end
end

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;

K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j1=1:5000
    j1
    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_a(k,1).*(alpha_a'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_a(k,1)*K1(:,k));
    end

    alpha_a_old = alpha_a;

    alpha_a=(K0+c*sm1+eye(size(K0,1)))/(c*sm2);

    if (sum(abs(alpha_a_old-alpha_a))<0.01)
        j
        break;
    end
end

result_a = (alpha_a'*K1)';

```

```

for j2=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_b(k,1).*(alpha_b'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_b(k,1)*K1(:,k));
    end

    alpha_b_old = alpha_b;

    alpha_b=(K0+c*sm1+eye(size(K0,1)))\(c*sm2);

    if (sum(abs(alpha_b_old-alpha_b))<0.01)
        j
        break;
    end
end

result_b = (alpha_b'*K1)';

for j3=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_c(k,1).*(alpha_c'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_c(k,1)*K1(:,k));
    end

    alpha_c_old = alpha_c;

    alpha_c=(K0+c*sm1+eye(size(K0,1)))\(c*sm2);

    if (sum(abs(alpha_c_old-alpha_c))<0.01)
        j
        break;
    end
end

result_c = (alpha_c'*K1)';

train_res = [result_a result_b result_c];

```

```

for i=1:size(train_res)
    [y p] = max(train_res(i,1:3));
    train_res(i,4)=p;
end

train_res(:,5)=train_label;

err_train=sum(train_res(:,5)~=train_res(:,4))/length(train_res)

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_a,1)
        if(j>size(X_train,1))
            value=value+alpha_a(j,1);
        else
            value=value+alpha_a(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_b,1)
        if(j>size(X_train,1))
            value=value+alpha_b(j,1);
        else
            value=value+alpha_b(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,2)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_c,1)
        if(j>size(X_train,1))
            value=value+alpha_c(j,1);
        else
            value=value+alpha_c(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,3)=value;
end

for i=1:size(test_res)
    [y p] = max(test_res(i,1:3));
    test_res(i,4)=p;
end

test_res(:,5)=test_label;

```

```
err_test = sum(test_res(:,5)~=test_res(:,4))/length(test_res)
```

## Function for computing the kernel

```
function [ value ] = kernel( x1,x2,sigma )  
  
value= exp(-(norm(x1-x2))^2)/(2*sigma*sigma);  
  
end
```

# Forest dataset

Code:

```
clear;

sigma=40;
c=1;
e=10^-6;

%(d,1) (h,2) (o,3) (s,4)

train_data= importdata('training.csv');
test_data= importdata('testing.csv');

train_set=train_data.data;
test_set=test_data.data;

X_train=train_set;

X_test=test_set;

train_label=train_data.textdata(2:199,1);
test_label=test_data.textdata(2:326,1);

for i=1:length(train_set)
    if (strcmp(train_label(i,1),'d')==1)
        k=1;
    elseif(strcmp(train_label(i,1),'h')==1)
        k=2;
    elseif(strcmp(train_label(i,1),'o')==1)
        k=3;
    else
        k=4;
    end

    train_class(i,1)=k;

    for j=1:4
        if(j==k)
            Y(i,j)=1;
        else
            Y(i,j)= -1;
        end
    end
end

for i=1:length(test_set)
    if (strcmp(test_label(i,1),'d')==1)
        k=1;
    elseif(strcmp(test_label(i,1),'h')==1)
        k=2;
```

```

elseif(strcmp(test_label(i,1),'o')==1)
    k=3;
else
    k=4;
end

test_class(i,1)=k;

end

Y_1=Y(:,1);
Y_2=Y(:,2);
Y_3=Y(:,3);
Y_4=Y(:,4);

tpa=length(X_train);

alpha_1=ones(tpa+1,1);
alpha_2=ones(tpa+1,1);
alpha_3=ones(tpa+1,1);
alpha_4=ones(tpa+1,1);

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;
K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j1=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_1(k,1).*(alpha_1'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_1(k,1)*K1(:,k));
    end

    alpha_1_old = alpha_1;

    alpha_1=(K0+c*sm1)\(c*sm2);

```

```

        if (sum(abs(alpha_1_old-alpha_1))<0.0001)
            j1
            break;
        end
    end

result_1 = (alpha_1'*K1)';

for j2=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_2(k,1).*(alpha_2'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_2(k,1)*K1(:,k));
    end

    alpha_2_old = alpha_2;

    alpha_2=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_2_old-alpha_2))<0.0001)
        j2
        break;
    end
end

result_2 = (alpha_2'*K1)';

for j3=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_3(k,1).*(alpha_3'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_3(k,1)*K1(:,k));
    end

    alpha_3_old = alpha_3;

    alpha_3=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_3_old-alpha_3))<0.0001)
        j3
        break;
    end
end

```



```

end

result_3 = (alpha_3'*K1)';

for j4=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_4(k,1).*(alpha_4'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+((1+z(k))/(2*z(k)))*Y_4(k,1)*K1(:,k);
    end

    alpha_4_old = alpha_4;

    alpha_4=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_4_old-alpha_4))<0.0001)
        j4
        break;
    end
end

result_4 = (alpha_4'*K1)';

train_res=[result_1 result_2 result_3 result_4];

for i=1:size(train_res,1)
    [y p] = max(train_res(i,1:4));
    train_res(i,5)=p;
end

train_res(:,6)=train_class;

err_train=sum(train_res(:,5)~=train_res(:,6))/length(train_res)

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_1,1)
        if(j>size(X_train,1))
            value=value+alpha_1(j,1);
        else
            value=value+alpha_1(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end
end

```

```

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_2,1)
            if(j>size(X_train,1))
                value=value+alpha_2(j,1);
            else
                value=value+alpha_2(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,2)=value;
    end

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_3,1)
            if(j>size(X_train,1))
                value=value+alpha_3(j,1);
            else
                value=value+alpha_3(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,3)=value;
    end

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_4,1)
            if(j>size(X_train,1))
                value=value+alpha_4(j,1);
            else
                value=value+alpha_4(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,4)=value;
    end

    for i=1:size(test_res,1)
        [y p] = max(test_res(i,1:4));
        test_res(i,5)=p;
    end

    test_res(:,6)=test_class;

err_test = sum(test_res(:,5)~=test_res(:,6))/length(test_res)

```

# Tic-tac-toe dataset

Code:

```
clear;
A = importdata('tic-tac-toe.csv.xlsx');

set=A.data.Sheet1;
label=A.textdata.Sheet1;

sigma=1;
c=10;
e=10^-6;

% err_train =
%
%      0
%
%
% err_test =
%
%      0.0418

tp= 50; %percent in training set

tpa= ceil(length(set)*tp/100);

for i=1:length(label)
    if(strcmp(label(i,1),'positive')==1)
        class(i,1)= 1;
    else
        class(i,1)= 2;
    end
end

set(:,28)=class;

set(randperm(length(set)),:)=set;

X_train=set(1:tpa,1:27);
train_label= set(1:tpa,28);

X_test=set(tpa+1:length(set),1:27);
test_label= set(tpa+1:length(set),28);

alpha_a=ones(tpa+1,1);
alpha_b=ones(tpa+1,1);

for i=1:length(train_label)
    if(train_label(i,1)==1)
        Y_a(i,1)= 1;
    end
end
```

```

        Y_b(i,1)= -1;
    else
        Y_a(i,1)= -1;
        Y_b(i,1)= 1;
    end
end

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;

K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j1=1:5000
    j1
    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_a(k,1).*(alpha_a'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_a(k,1)*K1(:,k));
    end

    alpha_a_old = alpha_a;

    alpha_a=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_a_old-alpha_a))<0.01)
        j
        break;
    end
end

result_a = (alpha_a'*K1)';

for j2=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_b(k,1).*(alpha_b'*K1(:,k))));
    end
end

```

```

end

sm1=0;
sm2=0;
for k=1:size(X_train,1)
    sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
    sm2=sm2+(((1+z(k))/(2*z(k)))*Y_b(k,1)*K1(:,k));
end

alpha_b_old = alpha_b;

alpha_b=(K0+c*sm1)\(c*sm2);

if (sum(abs(alpha_b_old-alpha_b))<0.01)
    j
    break;
end
end

result_b = (alpha_b'*K1)';

train_res = [result_a result_b]

for i=1:size(train_res)
    [y p] = max(train_res(i,1:2));
    train_res(i,3)=p;
end

train_res(:,4)=train_label;

err_train=sum(train_res(:,3)~=train_res(:,4))/length(train_res)

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_a,1)
        if(j>size(X_train,1))
            value=value+alpha_a(j,1);
        else
            value=value+alpha_a(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_b,1)
        if(j>size(X_train,1))
            value=value+alpha_b(j,1);
        else
            value=value+alpha_b(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
end

```

```
        end
        test_res(i,2)=value;
    end

    for i=1:size(test_res)
        [y p] = max(test_res(i,1:2));
        test_res(i,3)=p;
    end

    test_res(:,4)=test_label;

    err_test = sum(test_res(:,3)~=test_res(:,4))/length(test_res)
```

# Magic dataset

Code:

```
clear;
A = importdata('magic04.xlsx');

set=A.data;
label=A.textdata;

tp= 50; %percent in training set

sigma=1;
c=10;
e=10^-6;

for i=1:length(label)
    if(strcmp(label(i,1),'g')==1)
        class(i,1)= 1;
    else
        class(i,1)= 2;
    end
end

set(:,11)=class;

set(randperm(length(set)),:)=set;

set=set(1:2000,:);

tpa= ceil(length(set)*tp/100);

X_train=set(1:tpa,1:10);
train_label= set(1:tpa,11);

X_test=set(tpa+1:length(set),1:10);
test_label= set(tpa+1:length(set),11);

alpha_a=ones(tpa+1,1);
alpha_b=ones(tpa+1,1);

for i=1:length(train_label)
    if(train_label(i,1)==1)
        Y_a(i,1)= 1;
        Y_b(i,1)= -1;
    else
        Y_a(i,1)= -1;
        Y_b(i,1)= 1;
    end
end
end
```

```

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;

K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j1=1:5000
    j1
    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_a(k,1).*(alpha_a'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_a(k,1)*K1(:,k));
    end

    alpha_a_old = alpha_a;

    alpha_a=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_a_old-alpha_a))<0.01)
        j
        break;
    end
end

result_a = (alpha_a'*K1)';

for j2=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_b(k,1).*(alpha_b'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));

```



```

        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_b(k,1)*K1(:,k));
    end

    alpha_b_old = alpha_b;

    alpha_b=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_b_old-alpha_b))<0.01)
        j
        break;
    end
end

result_b = (alpha_b'*K1)';

train_res = [result_a result_b];

for i=1:size(train_res)
    [y p] = max(train_res(i,1:2));
    train_res(i,3)=p;
end

train_res(:,4)=train_label;

err_train=sum(train_res(:,3)~=train_res(:,4))/length(train_res)

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_a,1)
        if(j>size(X_train,1))
            value=value+alpha_a(j,1);
        else
            value=value+alpha_a(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_b,1)
        if(j>size(X_train,1))
            value=value+alpha_b(j,1);
        else
            value=value+alpha_b(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,2)=value;
end

for i=1:size(test_res)

```

```
        [y p] = max(test_res(i,1:2));  
        test_res(i,3)=p;  
end  
  
test_res(:,4)=test_label;  
  
err_test = sum(test_res(:,3)~=test_res(:,4))/length(test_res)
```

# Cancer dataset

Code:

```
clear;
A = importdata('cancer_data.csv');

set = A(:,2:11);

sigma=10;
c=10;
e=10^-6;

tp= 50; %percent in training set

tpa= ceil(length(set)*tp/100);

set(randperm(length(set)),:)=set;

train=set(1:tpa,:);
test=set(tpa+1:length(set),:);

X_train=train(:,1:9);
train_class=train(:,10);

X_test=test(:,1:9);
test_class=test(:,10);

alpha_a=ones(tpa+1,1);
alpha_b=ones(tpa+1,1);

for i=1:length(train_class)
    if(train_class(i,1)==4)
        Y_a(i,1)= -1;
        Y_b(i,1)= 1;
        train_label(i,1)=2;
    else
        Y_a(i,1)= 1;
        Y_b(i,1)= -1;
        train_label(i,1)=1;
    end
end

for i=1:length(test_class)
    if(test_class(i,1)==2)
        test_label(i,1)= 1;
    else
```

```

        test_label(i,1)= 2;
    end
end

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;

K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_a(k,1).*(alpha_a'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_a(k,1)*K1(:,k));
    end

    alpha_a_old = alpha_a;

    alpha_a=(K0+c*sm1+eye(size(K0,1)))\(c*sm2);

    if (sum(abs(alpha_a_old-alpha_a))<0.01)
        j
        break;
    end
end

result_a = (alpha_a'*K1)';

for j=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_b(k,1).*(alpha_b'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));

```

```

        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_b(k,1)*K1(:,k));
    end

    alpha_b_old = alpha_b;

    alpha_b=(K0+c*sm1+eye(size(K0,1))\ (c*sm2);

    if (sum(abs(alpha_b_old-alpha_b))<0.01)
        j
        break;
    end
end

result_b = (alpha_b'*K1)';

train_res = [result_a result_b]

for i=1:size(train_res)
    [y p] = max(train_res(i,1:2));
    train_res(i,3)=p;
end

train_res(:,4)=train_label;

err_train=sum(train_res(:,3)~=train_res(:,4))/length(train_res)

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_a,1)
        if(j>size(X_train,1))
            value=value+alpha_a(j,1);
        else
            value=value+alpha_a(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_b,1)
        if(j>size(X_train,1))
            value=value+alpha_b(j,1);
        else
            value=value+alpha_b(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,2)=value;
end

for i=1:size(test_res)

```

```
        [y p] = max(test_res(i,1:2));  
        test_res(i,3)=p;  
end  
  
test_res(:,4)=test_label;  
  
err_test = sum(test_res(:,3)~=test_res(:,4))/length(test_res)
```

# Handwritten letter dataset

Code:

```
clear;

sigma=50;
c=1;
e=10^-6;

train_set= importdata('optdigits.tra.txt');
test_set= importdata('optdigits.tes.txt');

X_train=train_set(:,1:64);

X_train=X_train(1:200,:);

X_test=test_set(:,1:64);

train_label=train_set(1:200,65);
test_label=test_set(:,65);

for i=1:length(X_train)
    k=train_set(i,65);
    for j=1:10
        if(j==k+1)
            Y(i,j)=1;
        else
            Y(i,j)= -1;
        end
    end
end

Y_0=Y(:,1);
Y_1=Y(:,2);
Y_2=Y(:,3);
Y_3=Y(:,4);
Y_4=Y(:,5);
Y_5=Y(:,6);
Y_6=Y(:,7);
Y_7=Y(:,8);
Y_8=Y(:,9);
Y_9=Y(:,10);

tpa=length(X_train);

alpha_0=ones(tpa+1,1);
alpha_1=ones(tpa+1,1);
alpha_2=ones(tpa+1,1);
alpha_3=ones(tpa+1,1);
alpha_4=ones(tpa+1,1);
alpha_5=ones(tpa+1,1);
```

```

alpha_6=ones(tpa+1,1);
alpha_7=ones(tpa+1,1);
alpha_8=ones(tpa+1,1);
alpha_9=ones(tpa+1,1);

for i=1:tpa
    for j=1:tpa
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;
K0(tpa+1,:)=0;
K0(:,tpa+1)=0;
K0(tpa+1,tpa+1)=0;

K1=K;
K1(tpa+1,:)=1;

for j0=1:5000
    j0
    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_0(k,1).*(alpha_0'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+((1+z(k))/(2*z(k)))*Y_0(k,1)*K1(:,k);
    end

    alpha_0_old = alpha_0;

    alpha_0=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_0_old-alpha_0))<0.01)
        j0
        break;
    end
end

result_0 = (alpha_0'*K1)';

for j1=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_1(k,1).*(alpha_1'*K1(:,k))));
    end

    sm1=0;
    sm2=0;

```



```

for k=1:size(X_train,1)
    sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
    sm2=sm2+(((1+z(k))/(2*z(k)))*Y_1(k,1)*K1(:,k));
end

alpha_1_old = alpha_1;

alpha_1=(K0+c*sm1)\(c*sm2);

if (sum(abs(alpha_1_old-alpha_1))<0.01)
    j1
    break;
end
end

result_1 = (alpha_1'*K1)';

for j2=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_2(k,1).*(alpha_2'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_2(k,1)*K1(:,k));
    end

    alpha_2_old = alpha_2;

    alpha_2=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_2_old-alpha_2))<0.01)
        j2
        break;
    end
end

result_2 = (alpha_2'*K1)';

for j3=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_3(k,1).*(alpha_3'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_3(k,1)*K1(:,k));
    end
end

```

```

alpha_3_old = alpha_3;

alpha_3=(K0+c*sm1)\(c*sm2);

if (sum(abs(alpha_3_old-alpha_3))<0.01)
    j3
    break;
end
end

result_3 = (alpha_3'*K1)';

for j4=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_4(k,1).*(alpha_4'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_4(k,1)*K1(:,k));
    end

    alpha_4_old = alpha_4;

    alpha_4=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_4_old-alpha_4))<0.01)
        j4
        break;
    end
end

result_4 = (alpha_4'*K1)';

for j5=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_5(k,1).*(alpha_5'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_5(k,1)*K1(:,k));
    end

    alpha_5_old = alpha_5;

    alpha_5=(K0+c*sm1)\(c*sm2);

```

```

        if (sum(abs(alpha_5_old-alpha_5))<0.01)
            j5
            break;
        end
    end

result_5 = (alpha_5'*K1)';

for j6=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_6(k,1).*(alpha_6'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_6(k,1)*K1(:,k));
    end

    alpha_6_old = alpha_6;

    alpha_6=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_6_old-alpha_6))<0.01)
        j6
        break;
    end
end

result_6 = (alpha_6'*K1)';

for j7=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_7(k,1).*(alpha_7'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_7(k,1)*K1(:,k));
    end

    alpha_7_old = alpha_7;

    alpha_7=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_7_old-alpha_7))<0.01)
        j7
        break;
    end
end

```

```

        end
    end

result_7 = (alpha_7'*K1)';

for j8=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_8(k,1).*(alpha_8'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_8(k,1)*K1(:,k));
    end

    alpha_8_old = alpha_8;

    alpha_8=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_8_old-alpha_8))<0.01)
        j8
        break;
    end
end

result_8 = (alpha_8'*K1)';

for j9=1:5000

    for k=1:tpa
        z(k,1)=max(e,abs(1-Y_9(k,1).*(alpha_9'*K1(:,k))));
    end

    sm1=0;
    sm2=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_9(k,1)*K1(:,k));
    end

    alpha_9_old = alpha_9;

    alpha_9=(K0+c*sm1)\(c*sm2);

    if (sum(abs(alpha_9_old-alpha_9))<0.01)
        j9
        break;
    end
end

result_9 = (alpha_9'*K1)';

```

```
train_res=[result_0 result_1 result_2 result_3 result_4 result_5 result_6  
result_6 result_7 result_8 result_9];
```

```
for i=1:size(train_res,1)  
    [y p] = max(train_res(i,1:10));  
    train_res(i,11)=p-1;  
end
```

```
train_res(:,12)=train_label;
```

```
err_train=sum(train_res(:,11)~=train_res(:,12))/length(train_res)
```

```
for i=1:size(X_test,1)  
    value=0;  
    for j=1:size(alpha_0,1)  
        if(j>size(X_train,1))  
            value=value+alpha_0(j,1);  
        else  
value=value+alpha_0(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);  
        end  
    end  
    test_res(i,1)=value;  
end
```

```
for i=1:size(X_test,1)  
    value=0;  
    for j=1:size(alpha_1,1)  
        if(j>size(X_train,1))  
            value=value+alpha_1(j,1);  
        else  
value=value+alpha_1(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);  
        end  
    end  
    test_res(i,2)=value;  
end
```

```
for i=1:size(X_test,1)  
    value=0;  
    for j=1:size(alpha_2,1)  
        if(j>size(X_train,1))  
            value=value+alpha_2(j,1);  
        else  
value=value+alpha_2(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);  
        end  
    end  
    test_res(i,3)=value;  
end
```

```

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_3,1)
        if(j>size(X_train,1))
            value=value+alpha_3(j,1);
        else
            value=value+alpha_3(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,4)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_4,1)
        if(j>size(X_train,1))
            value=value+alpha_4(j,1);
        else
            value=value+alpha_4(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,5)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_5,1)
        if(j>size(X_train,1))
            value=value+alpha_5(j,1);
        else
            value=value+alpha_5(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,6)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_6,1)
        if(j>size(X_train,1))
            value=value+alpha_6(j,1);
        else
            value=value+alpha_6(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,7)=value;
end

```

```

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_7,1)
            if(j>size(X_train,1))
                value=value+alpha_7(j,1);
            else
value=value+alpha_7(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,8)=value;
    end

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_8,1)
            if(j>size(X_train,1))
                value=value+alpha_8(j,1);
            else
value=value+alpha_8(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,9)=value;
    end

    for i=1:size(X_test,1)
        value=0;
        for j=1:size(alpha_9,1)
            if(j>size(X_train,1))
                value=value+alpha_9(j,1);
            else
value=value+alpha_9(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
            end
        end
        test_res(i,10)=value;
    end

    for i=1:size(test_res,1)
        [y p] = max(test_res(i,1:10));
        test_res(i,11)=p-1;
    end

    test_res(:,12)=test_label;

err_test = sum(test_res(:,11)~=test_res(:,12))/length(test_res)

```

# Iris dataset

Code:

```
clear;
load fisheriris;

sigma=1;
c=1;
e=10^-6;

tp= 50; %percent in training set

tpe= 50*tp/100;

class1=meas(1:50,:);
class2=meas(51:100,:);
class3=meas(101:150,:);

class1(randperm(50),:)=class1;
class2(randperm(50),:)=class2;
class3(randperm(50),:)=class3;

train1=class1(1:tpe,:);
test1=class1(tpe+1:50,:);

train2=class2(1:tpe,:);
test2=class2(tpe+1:50,:);

train3=class3(1:tpe,:);
test3=class3(tpe+1:50,:);

X_train=[train1; train2; train3];
X_test=[test1; test2; test3];

one=0;
one(1:(2*tpe),1)=1;

Y1(1:tpe,1)=1;
Y2(1:tpe,1)=-1;

Y_a=[Y1;Y2;Y2];
Y_b=[Y2;Y1;Y2];
Y_c=[Y2;Y2;Y1];

one(1:(100-(tpe*2)),1)=1;
```



```

alpha_a=ones(3*tpe+1,1);
alpha_b=ones(3*tpe+1,1);
alpha_c=ones(3*tpe+1,1);

for i=1:3*tpe
    for j=1:3*tpe
        K(i,j)= kernel(X_train(i,:),X_train(j,:),sigma);
    end
end

K0=K;

K0(3*tpe+1,:)=0;
K0(:,3*tpe+1)=0;
K0(3*tpe+1,3*tpe+1)=0;

K1=K;
K1(3*tpe+1,:)=1;

for j=1:5000

    for k=1:3*tpe
        z(k,1)=max(e,abs(1-Y_a(k,1).*(alpha_a'*K1(:,k))));
    end

    sm1=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
    end

    sm2=0;

    for k=1:size(X_train,1)
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_a(k,1)*K1(:,k));
    end

    alpha_a_old = alpha_a;

    alpha_a=inv(K0+c*sm1)*c*sm2;

    if (sum(abs(alpha_a_old-alpha_a))<0.0001)
        j
        break;
    end
end

result_a = (alpha_a'*K1)';

for j=1:5000

    for k=1:3*tpe

```

```

        z(k,1)=max(e,abs(1-Y_b(k,1).*(alpha_b'*K1(:,k))));
    end

    sm1=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
    end

    sm2=0;

    for k=1:size(X_train,1)
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_b(k,1)*K1(:,k));
    end

    alpha_b_old = alpha_b;

    alpha_b=inv(K0+c*sm1)*c*sm2;

    if (sum(abs(alpha_b_old-alpha_b))<0.0001)
        j
        break;
    end
end

result_b = (alpha_b'*K1)';

for j=1:5000

    for k=1:3*tpe
        z(k,1)=max(e,abs(1-Y_c(k,1).*(alpha_c'*K1(:,k))));
    end

    sm1=0;
    for k=1:size(X_train,1)
        sm1=sm1+(K1(:,k)*K1(:,k)')/(2*z(k));
    end

    sm2=0;

    for k=1:size(X_train,1)
        sm2=sm2+(((1+z(k))/(2*z(k)))*Y_c(k,1)*K1(:,k));
    end

    alpha_c_old = alpha_c;

    alpha_c=inv(K0+c*sm1)*c*sm2;

    if (sum(abs(alpha_c_old-alpha_c))<0.0001)
        j
        break;
    end
end

result_c = (alpha_c'*K1)';

```

```

train_res = [result_a result_b result_c];

for i=1:size(train_res)
    [y p] = max(train_res(i,1:3));
    train_res(i,4)=p;
end

value=0;
for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_a,1)
        if(j>size(X_train,1))
            value=value+alpha_a(j,1);
        else
            value=value+alpha_a(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,1)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_b,1)
        if(j>size(X_train,1))
            value=value+alpha_b(j,1);
        else
            value=value+alpha_b(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,2)=value;
end

for i=1:size(X_test,1)
    value=0;
    for j=1:size(alpha_c,1)
        if(j>size(X_train,1))
            value=value+alpha_c(j,1);
        else
            value=value+alpha_c(j,1)*kernel(X_train(j,:),X_test(i,:),sigma);
        end
    end
    test_res(i,3)=value;
end

for i=1:size(test_res)
    [y p] = max(test_res(i,1:3));
    test_res(i,4)=p;
end

sp1=one(1:50-tpe,1);
sp2=2*sp1;
sp3=3*sp1;

sp=[sp1;sp2;sp3];

```

```
test_res(:,5)=sp;  
train_res(:,5)=sp;  
err_train=sum(train_res(:,4)~=train_res(:,5))/length(train_res)  
err_test = sum(test_res(:,4)~=test_res(:,5))/length(test_res)
```