
```
% ECE 403 Assignment 2
% Problem 1.6
% Noah Rondeau V00816430

% load the data and extract the two classes of interest
load D_iris;
D = D_iris(1:4,51:150);

% prepare the data
D1 = D(:,1:50);
D2 = D(:,51:100);
rand('state',15)
r1 = randperm(50);
D1train = D1(:,r1(1:40));
D1test = D1(:,r1(41:50));
rand('state',16)
r2 = randperm(50);
D2train = D2(:,r2(1:40));
D2test = D2(:,r2(41:50));
Dtrain = [D1train D2train];
Dtest = [D1test D2test];

% class labels
y = [ones(40,1); -ones(40,1)];

% add the ones and take transpose to make ones at right column
Xhat = [Dtrain' ones(80, 1)];

% calculate the weights and biases
wb = pinv(Xhat)*y;
b = wb(end);
w = wb(1:end-1);

fprintf('The weights are:\nW = \n');
disp(w);
fprintf('The bias is\nb = %f\n\n', b);

% now classify the testing data
predictions = w'*Dtest + b;
class_predictions = sign(predictions);

% compare to expected labels and compute accuracy
class_labels = [ones(1,10) -ones(1,10)];
% this sum(x == y) computes the number of times that the equivalent
% elements are equal, handy shortcut
accuracy = sum(class_predictions == class_labels) /
    length(class_labels);

fprintf('The accuracy is %f\n\n', accuracy);

figure(1);
confusionchart(class_labels, class_predictions);
```

```
title('Confusion matrix');
```

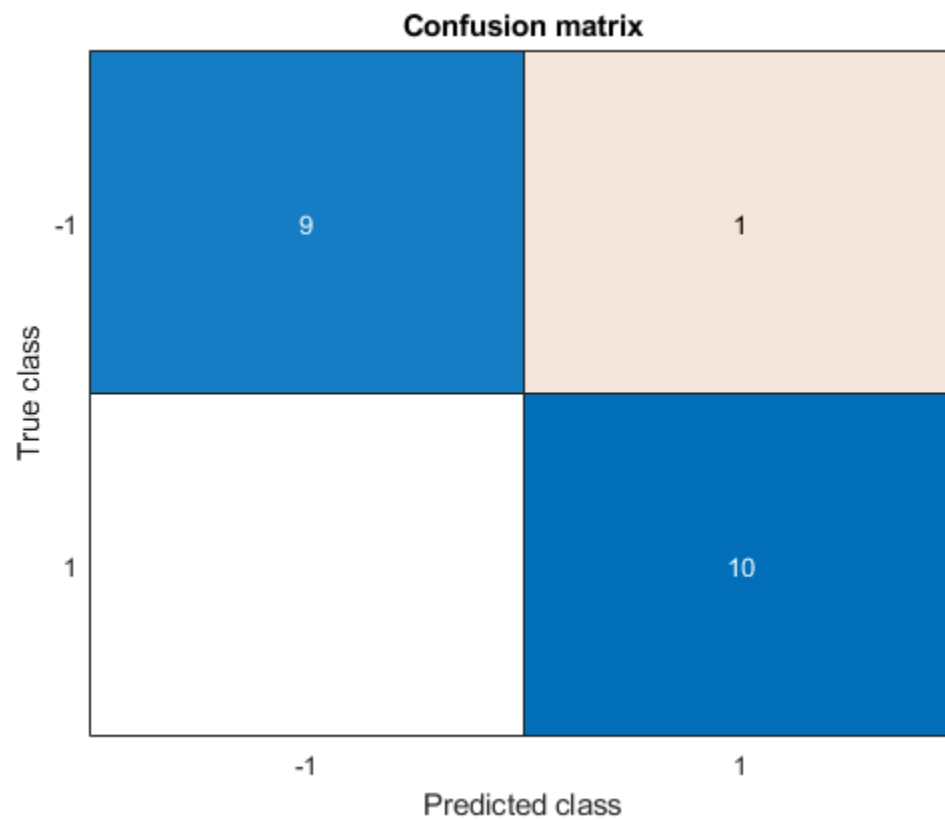
The weights are:

```
W =  
    0.2789  
    0.7447  
   -0.6800  
   -1.4144
```

The bias is

```
b = 1.828548
```

The accuracy is 0.950000



Published with MATLAB® R2018b