1. Least squares:

The least squares method is based on the criterion of minimizing mean squared error. Let's consider the 2-class case:

Let C0 and C1 denote the two classes. Thus, if y(i) = 0 then Xi ∈ C0 and if y(i) = 1 then Xi ∈ C1

Let n0 and n1 denote the number of examples(features) of each class. (n = n0 + n1)

For any W, y are the one-dimensional data that we get after projection.

$$y_i = w^T x_i$$

We now determine the parameter matrix w, by minimizing a sum-of-squares error function:

$$E_D[w] = \frac{1}{2} T_R (xw - T)^T (xw - T)$$

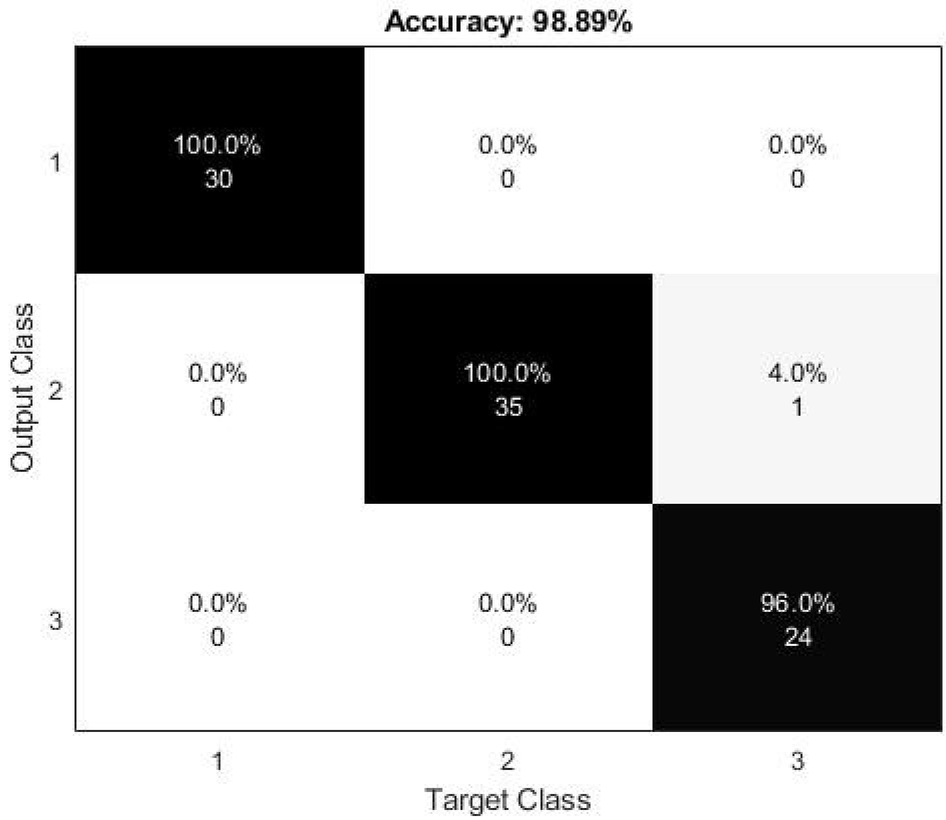
Setting the derivative with respect to w, to zero, and rearranging, we then obtain the solution for w, in the form:

$$w = (x^T x)^{-1} x^T T = x^{\dagger} T$$

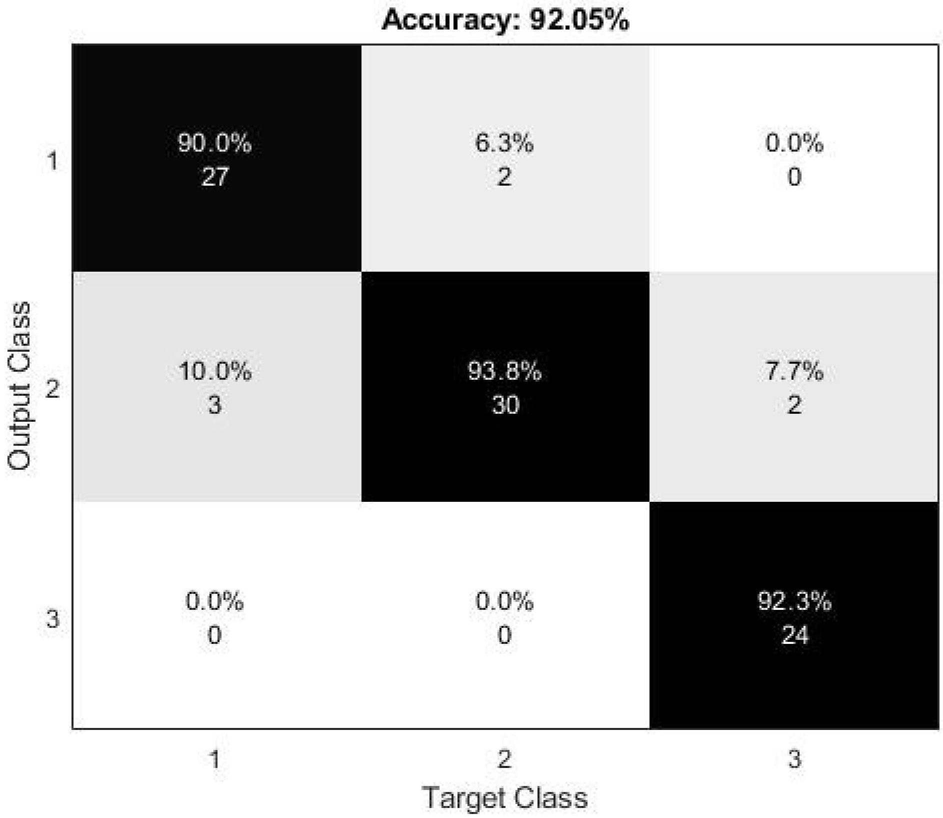
where $x^{\dagger}$ is the pseudo-inverse of the matrix x. We then obtain the discriminant function in the form.

Following are the Confusion matrices for different datasets for both training and test sets:

a. Wine Data

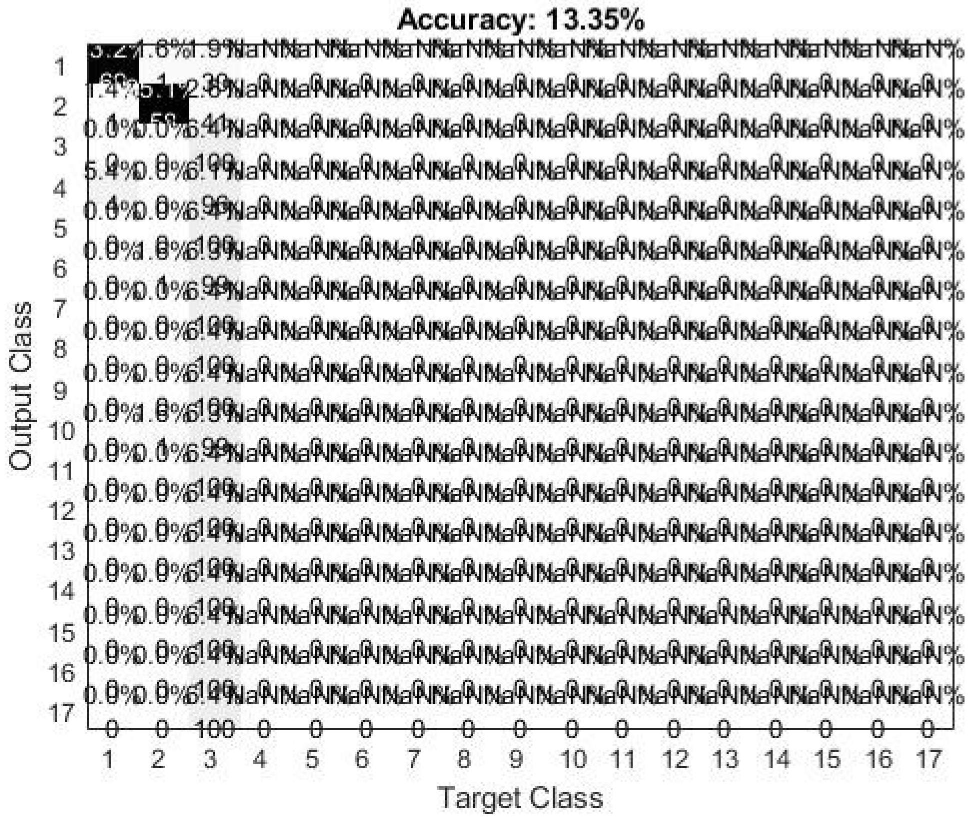


Training Data Set

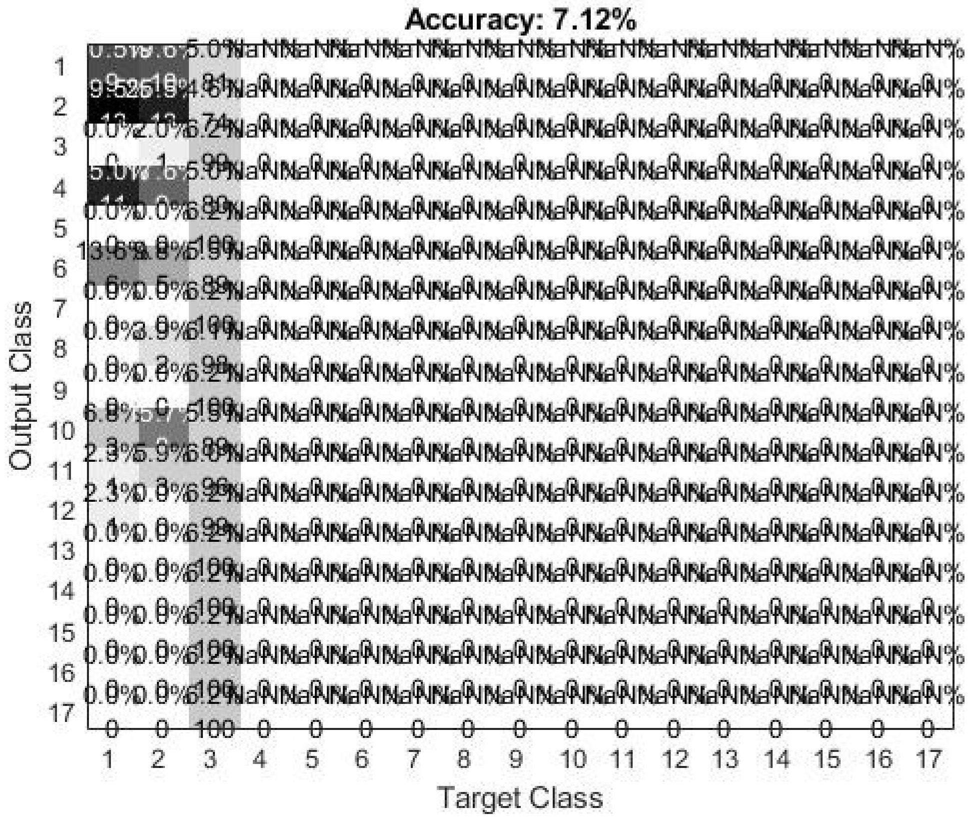


Test Data Set

b. Wallpaper Data

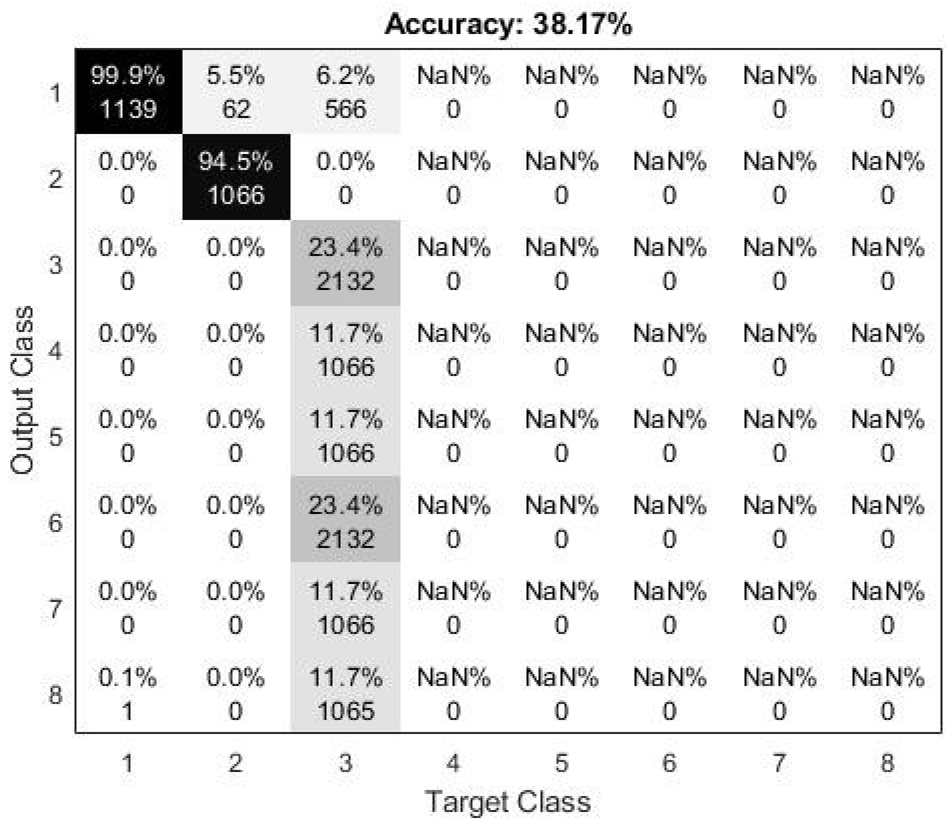


Training Data Set

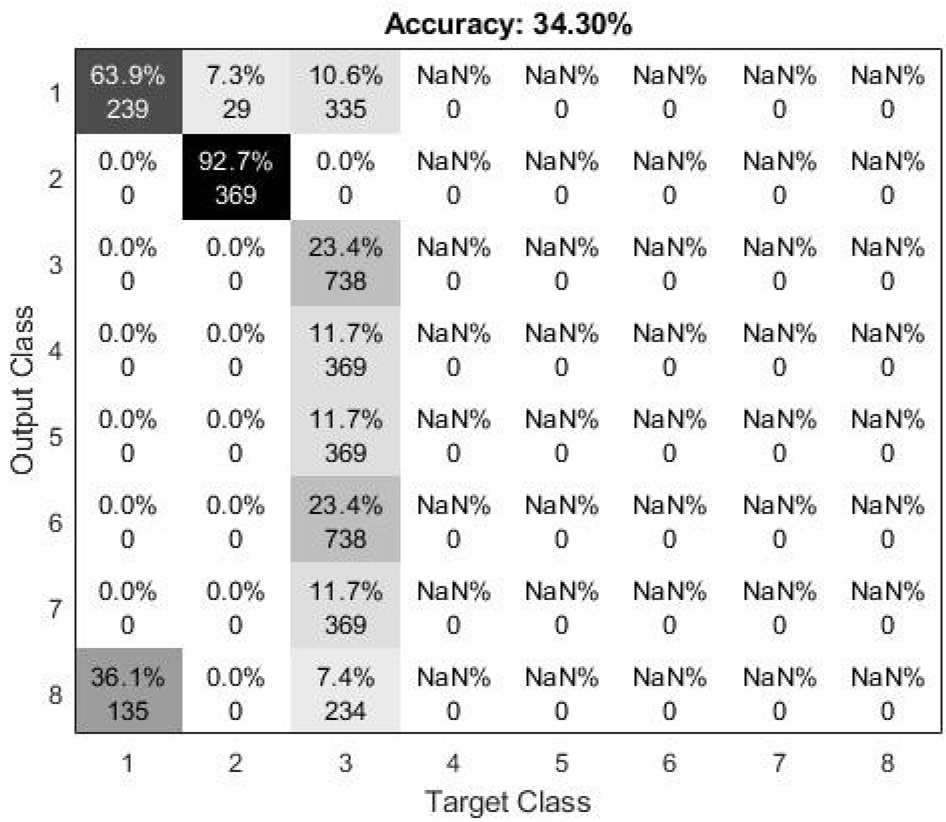


Test Data Set

c. Taiji Data

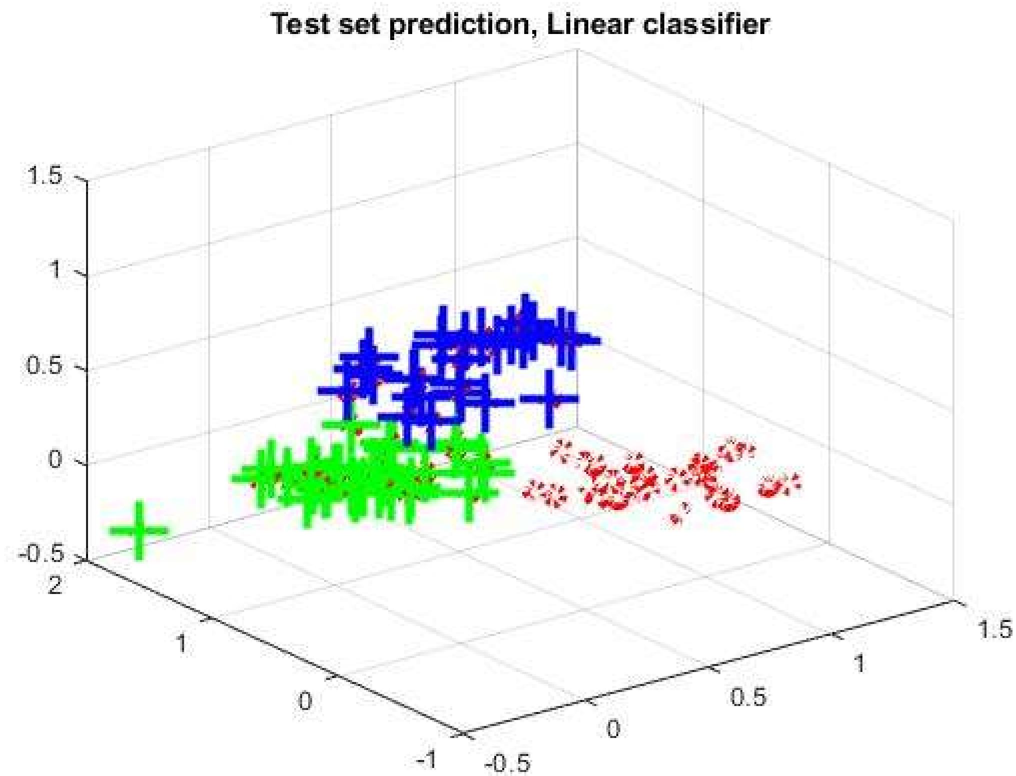


Training Data Set



Test Data Set

Below is the 3-Dimensional figure for Wine testing:



2. Fisher LDA:

The least squares method is based on the criterion of minimizing mean squared error. Let's consider the 2-class case:

Let C0 and C1 denote the two classes. Thus, if y(i) = 0 then Xi ∈ C0 and if y(i) = 1 then Xi ∈ C1

Let n0 and n1 denote the number of examples(features) of each class. (n = n0 + n1)

For any W, z are the one-dimensional data that we get after projection.

$$z_i = w^T x_i$$

Let $M_0$and M_1be the means of data from the two classes:

$$M_0 = \frac{1}{n_0} \sum_{i} x_i$$

$$M_1 = \frac{1}{n_1} \sum_{\cdot} x_i$$

we want a W that maximizes $[m_0 - m_1]^2$

However, we have to make this scale independent.

Also, the distance between means should be viewed relative to the variances.

Thus, we define:

$$s_0^2 = \sum_{x_i \in C_0} (w^T x_i - m_0)^2 \qquad s_1^2 = \sum_{x_i \in C_1} (w^T x_i - m_1)^2$$

These give us the variances (upto a factor) of the two classes in the projected data.

We want large separation between $M_0$and M_1 relative to the variances.

Hence, we can take our objective to be to maximize:

$$J_w = \frac{(M_1 - M_0)^2}{s_0^2 + s_1^2}$$

$$(M_1 - M_0)^2 = [w^T M_1 - w^T M_0]^2$$

$$(M_1 - M_0)^2 = W^T (M_1 - M_0)(M_1 - M_0)^T W$$

$$= W^T S_B \, W$$

$$S_B = (M_1 - M_0)(M_1 - M_0)^T$$

S_B is a d × d matrix

We can similarly write s_0^2 and s_1^2 also as quadratic forms.

We know

$$s_0^2 = \sum_{x_i \in C_0} (w^T x_\text{i} - m_0)^2 \qquad\qquad s_1^2 = \sum_{x_i \in C_1} (w^T x_\text{i} - m_1)^2$$

$$s_0^2 + s_1^2 = W^T S_w \, W$$

S_w is also d × d matrix and is called within class scatter matrix

Thus,

$$J_w = \frac{W^T S_B \, W}{W^T S_w \, W}$$

We want to find a W that maximizes J_w:

J_w is not affected by scaling of W.

Maximizing ratio of quadratic forms is a standard optimization problem

Differentiating w.r.t. W and equating to zero, we get:

$$\frac{2 S_B \, W}{W^T S_w \, W} - \frac{W^T S_B \, W}{W^T S_w \, W} * \frac{2 \, S_w W}{W^T S_w \, W} = 0$$

This Implies, $S_B \, W$ is in the same direction as $S_w \, W$

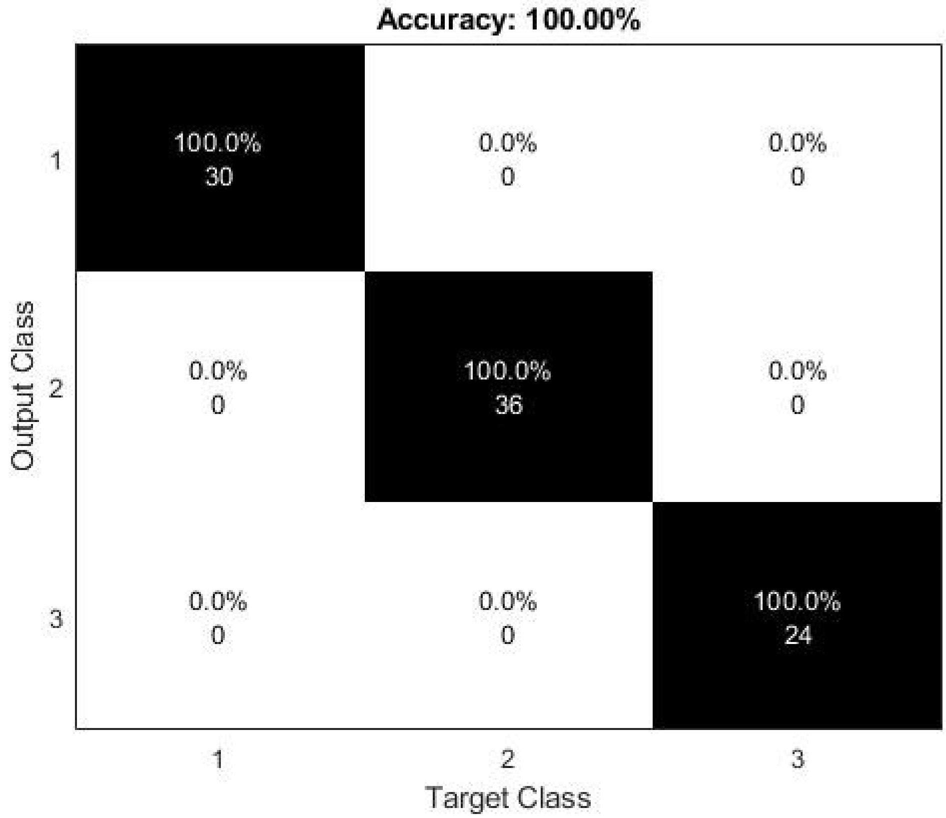
Thus, any maximizer of J_w has to satisfy $S_w \, W$ = λ $S_b \, W$ for some constant λ

This is known as the generalized eigen value problem.

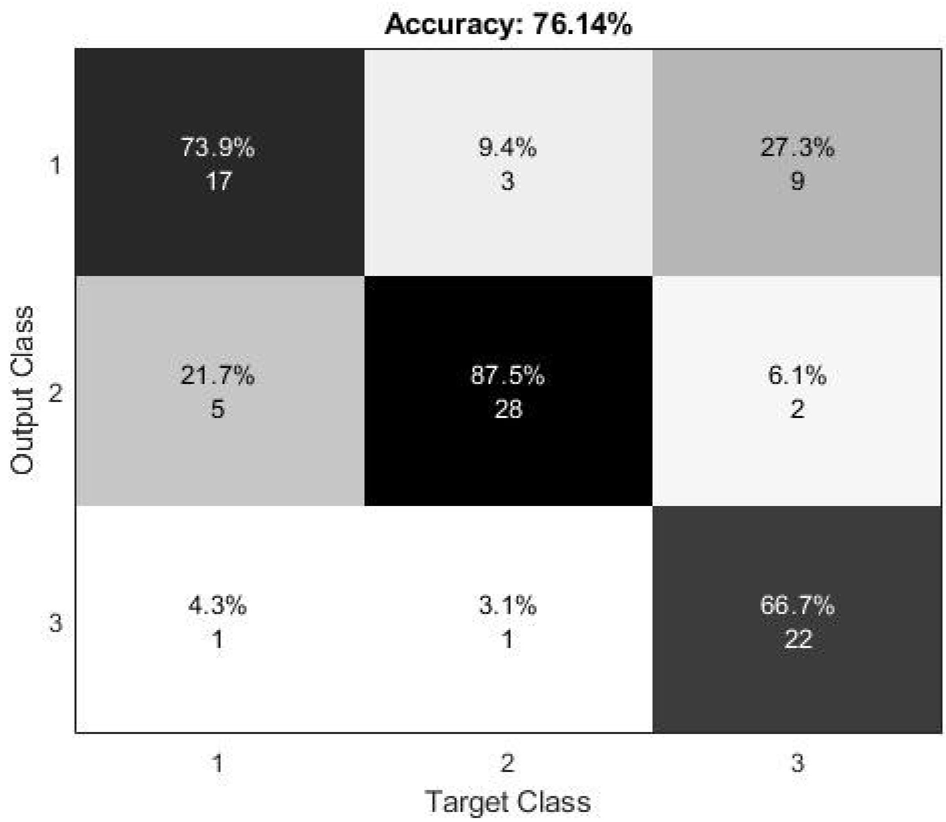
By solving the generalized eigen value problem we can find the best direction W.

Following are the Confusion matrices for different datasets for both training and test sets:

a) Wine dataset:

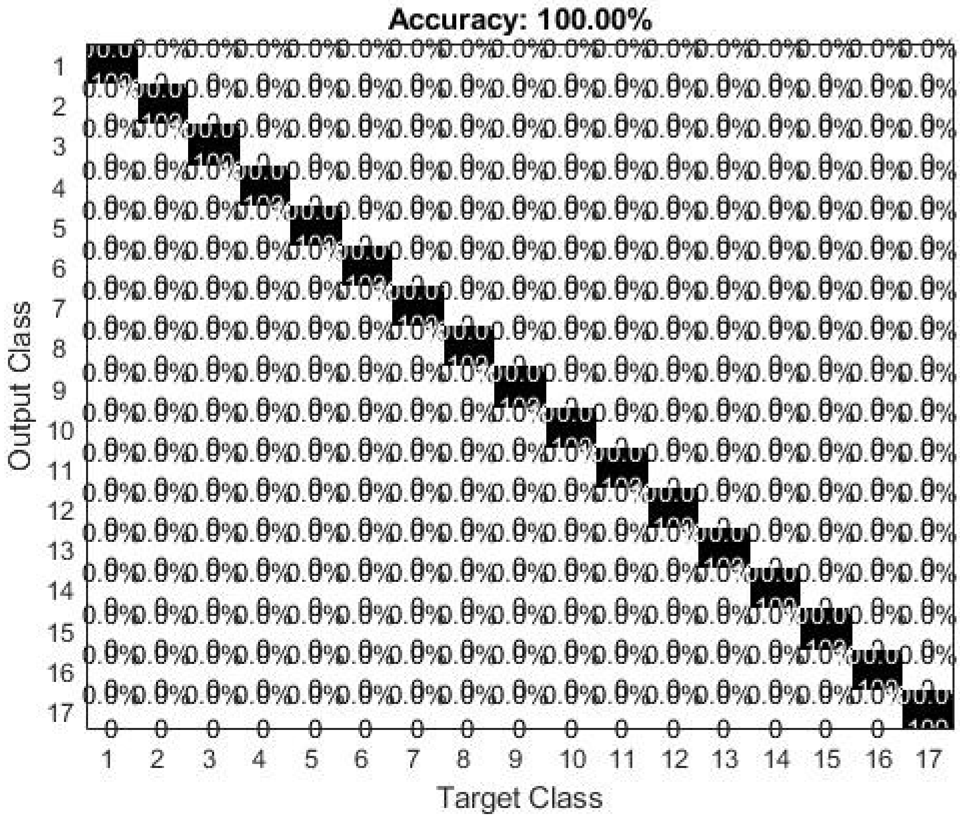


Training Data Set

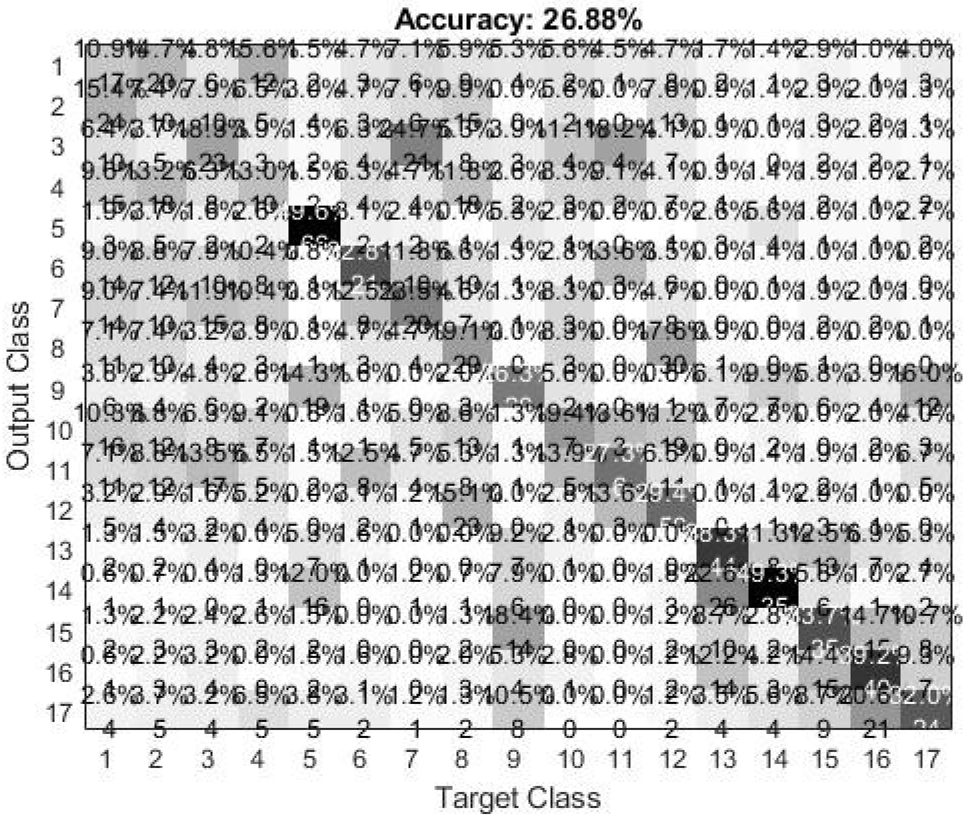


Test Data Set

b. Wallpaper Data Set:

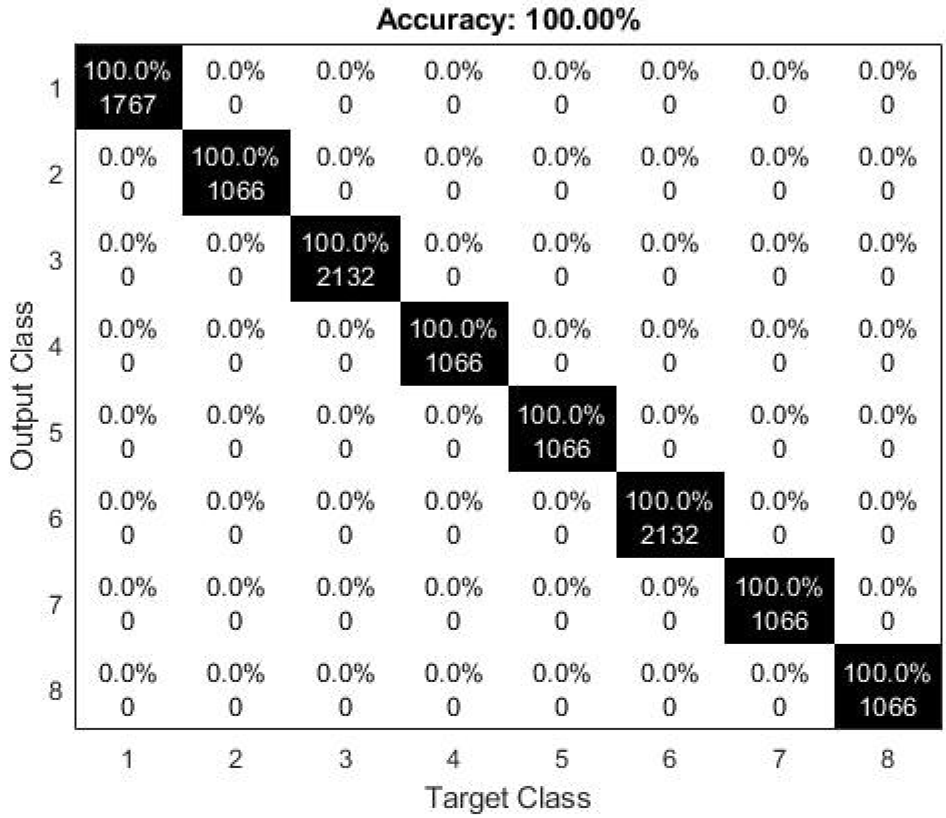


Training Data Set

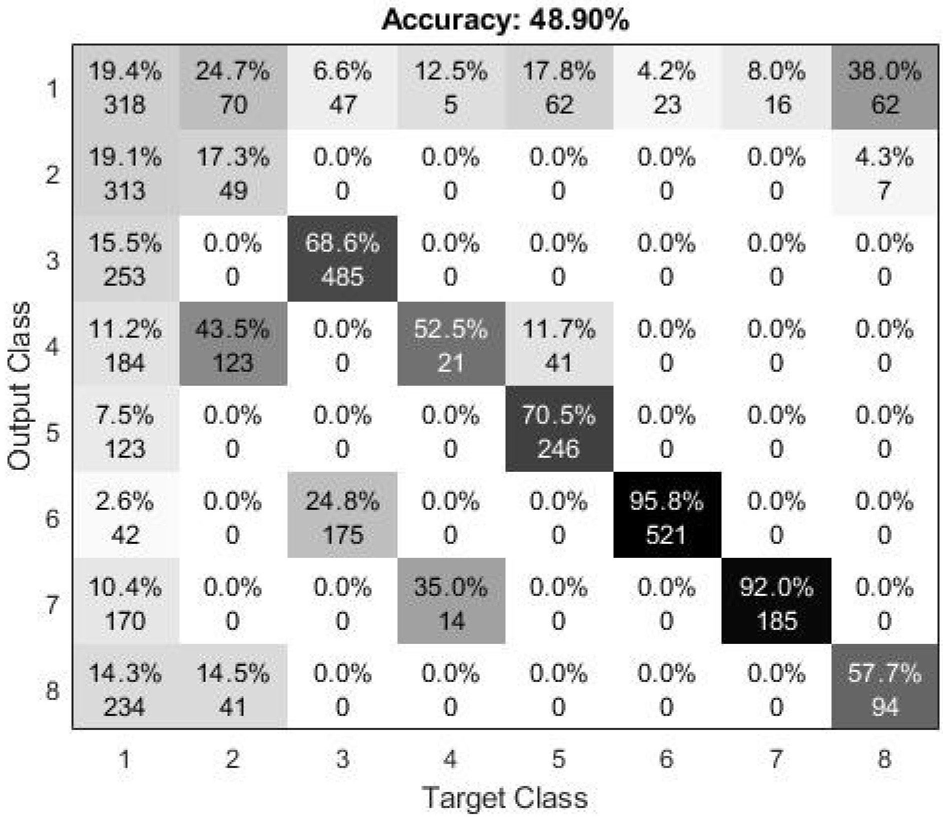


Test Data Set

c. Taiji Data Set:



Training Data Set



Test Data Set

3. Fisher LDA as special case of Linear Regression(for 2 classes):

Let’s say the targets for class C1 be N/N1, where N1 is the number of patterns in class C1, and N is the total number of patterns.

For class C_2, we shall take the targets to be $-N/N_2$, where N_2 is the number of patterns in class C_2

The sum-of-squares error function can be written:

$$E = \frac{1}{2}\sum_{1}^{N}[w^T x_N + w_0 - t_n]^2$$

Setting the derivatives of E with respect to w_0 and $\mathbf{w}$ to zero, we obtain respectively

$$\sum_{1}^{N}[w^T x_N + w_0 - t_n]x_N \quad = 0$$

making use of our choice of target coding scheme for the t_n, we obtain an expression for the bias in the form.

$$\omega_0 = -w^T m$$

where we have used

$$\tilde{\Sigma} t_n = N_1 \frac{N}{N_1} - N_2 \frac{N}{N_2} = 0$$

and where $\mathbf{m}$ is the mean of the total data set and is given by:

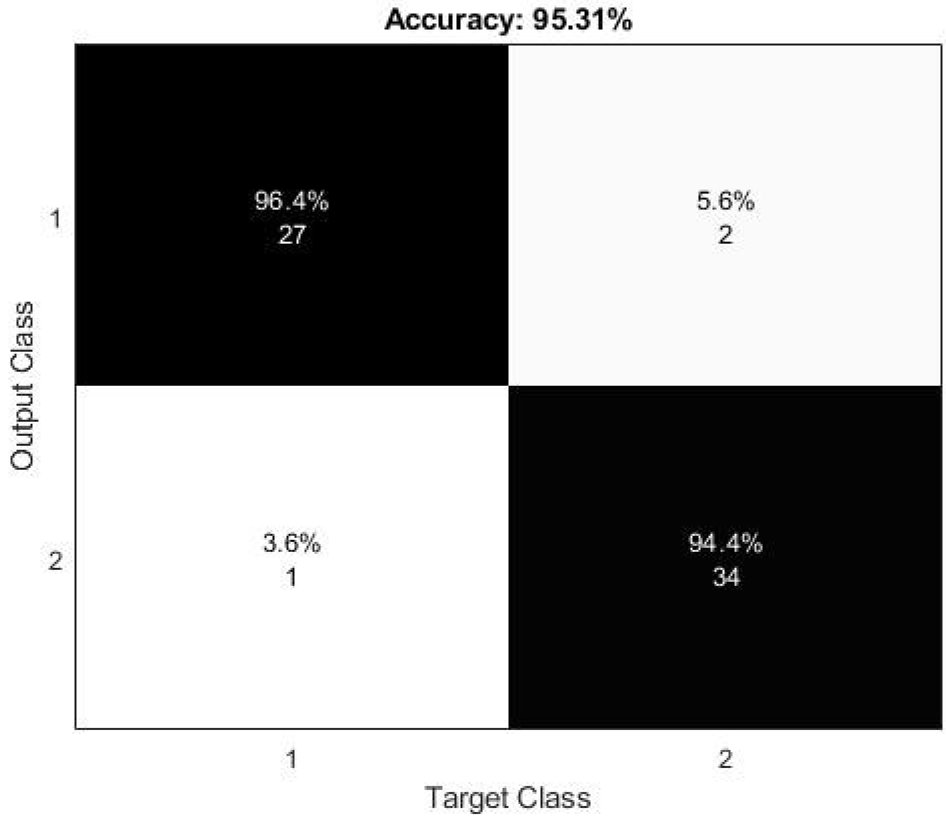
$$M = \frac{1}{N}(N_1 M_1 + N_2 M_2)$$

Thius we can write:

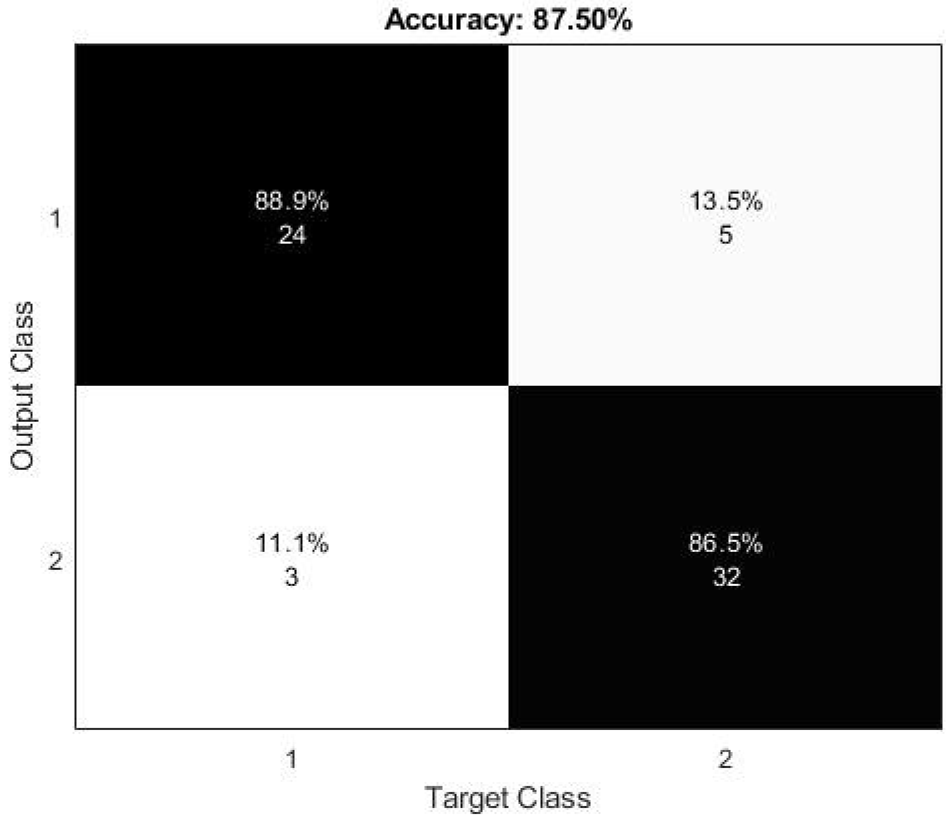
$$\left(S_{w \frac{+N_1 N_2}{N}} S_B\right) w = N(M_1 - M_2)$$

$$w \propto S_w^{-1}(M_2 - M_1)$$

Below are the confusion matrices for wine data after removing class 3:



Least Square Test Data Set



Fisher LDA Test Data Set