Since there are 3 classes there will be 3 hyperplanes, one for each pair. Let's compute hyperplane between class 1 & 2. Same procedure can be followed for the other two.

The training set H consists of vectors from 3 classes and is given by $H = \{ H_1, H_2, H_3 \}$

The eqn. of the hyperplane is given by

$$g(\underline{x}) = \underline{w}^T \underline{x} - w_0 = 0$$

Lets set a convention by stating that a vector falls on the positive side of the plane then that vector will be classified as class 1 and if it falls on the -ve side it will be classified as class 2.

Thus.
$$\underline{w}_{12}^T \underline{x}_i - w_0 = \begin{cases} > 0 & \text{if } \underline{x}_i \in H_1 \\ < 0 & \text{if } \underline{x}_i \in H_2 \end{cases}$$

This equation can be rewritten as

$$\underline{\hat{x}}_i^T \underline{w} = \begin{cases} > 0 & \text{if } \underline{\hat{x}}_i \in H_1 \\ < 0 & \text{if } \underline{\hat{x}}_i \in H_2 \end{cases}$$

where $\underline{w} = \begin{pmatrix} w_{12} \\ -w_0 \end{pmatrix}$ and $\underline{x}_i = \begin{pmatrix} x_i \\ 1 \end{pmatrix}$

Replacing every vector $\hat{\underline{x}}_i$ in H_2 by its negative the above equation becomes

$$\hat{\underline{x}}_i^T \underline{w} > 0 \qquad i = 1(1)n$$

Taking into account all the modified elements in $\{H_1, H_2\}$ the equation becomes. $A\underline{w} > \underline{0}$ where

$$A = \begin{pmatrix} \hat{\underline{x}}_1^T \\ \hat{\underline{x}}_2^T \\ \vdots \\ \hat{\underline{x}}_n^T \end{pmatrix}$$

Let $\underline{b}$ be an user chosen ~~off~~ sets and is given by

$$\underline{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad b_i > 0$$

Thus $A\underline{w} = \underline{b}$

Hence $\underline{w} = A^+\underline{b}$ where A^+ is the pseudoinverse of A

In the Ho Kashyap procedure, to begin with w and b contains arbitray values. Using gradient descent, by minimizing. the error, correct values of w and b are estimated. The error is given by $J_H(\underline{w}, \underline{b}) = (A\underline{w} - \underline{b})^T (A\underline{w} - \underline{b})$

Our target is to make $J_H(\underline{w}, \underline{b}) = 0$

Taking partial differentiation wrt to $\underline{w}$ & $\underline{b}$ in ~~..~~ turn we get

$$\nabla_w J(\underline{w}, \underline{b}) = 2A^T(A\underline{w} - \underline{b}) \quad \text{and}$$

$$\nabla_b J(\underline{w}, \underline{b}) = -2(A\underline{w} - \underline{b}).$$

Putting $\nabla_w J(\underline{w}, \underline{b}) = 0$

$$A^T A \underline{w} = A^T \underline{b}$$

$$\text{or } \underline{w} = (A^T A)^{-1} A^T \underline{b}$$

$$\text{or } \underline{w} = A^{+} \underline{b} \text{ where } A^{+} = (A^T A)^{-1} A^T$$

The algorithm followed for the simulation follows.:-

0) Start with arbitrary $b^{(1)}$ where $b_i > 0$. I have ~~ch~~ chosen $b_i = 1$. Arbitrary value for w is also chosen repeat steps 1 to 4.

1) $e^{(K)} = Aw^{(K)} - b^{(K)}$

2) $b^{(K+1)} = b^{(K)} + \alpha[e^{(K)} + |e^{(K)}|]$ [I have chosen $\alpha = 0.1$]

3) Solve for $w^{(K+1)}$ using $b^{(K+1)}$

$$w^{(K+1)} = A^{+} b^{(K+1)}$$

4) $K = K+1$

Repeat the steps untill $b^{(K+1)} = b^{(K)}$

This approach is called Gradient Descent as here gradually we descent ourselves to the min value of $J(\underline{w}, \underline{b})$.