5/4/20

Multi-Class Classification

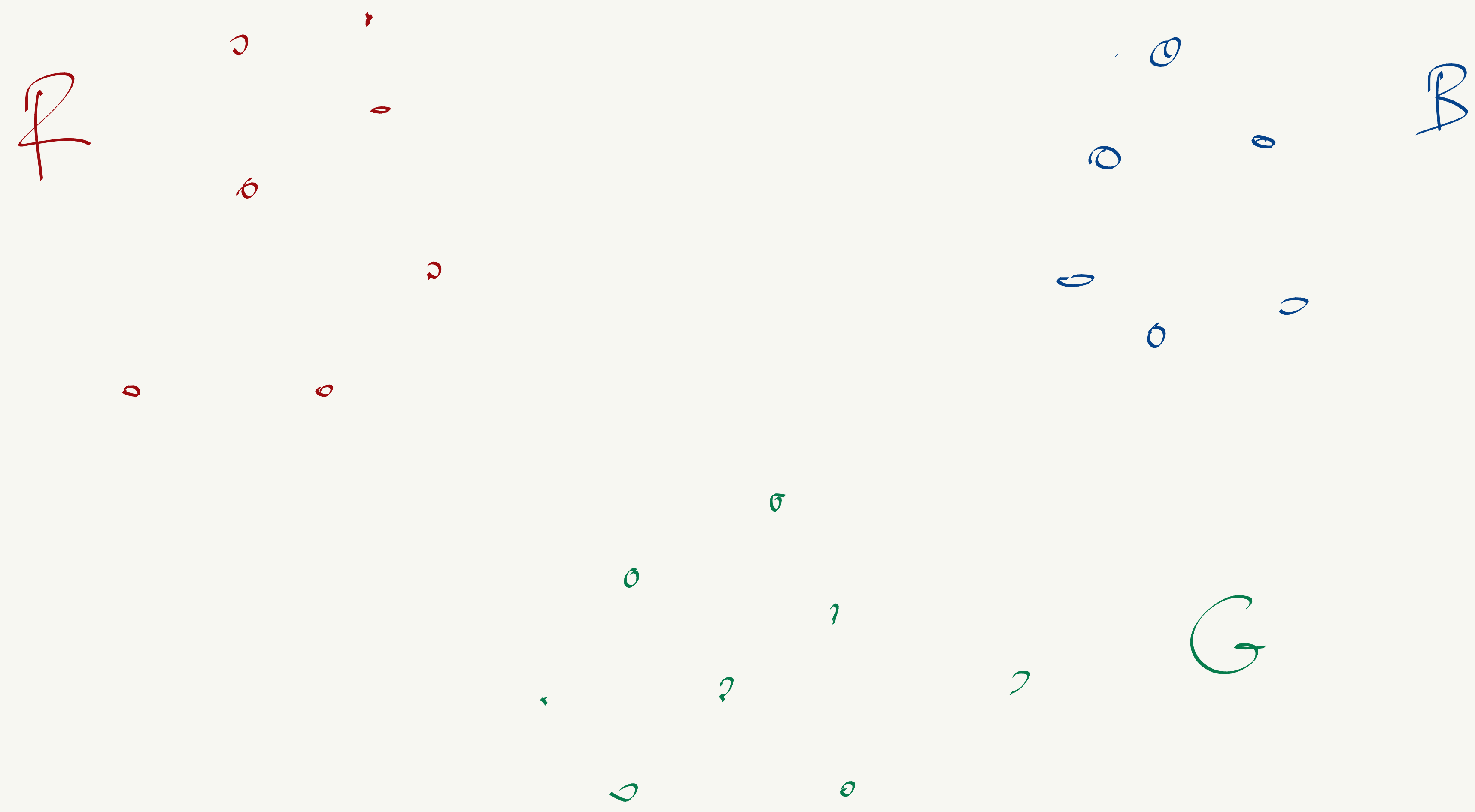
Given data

$$\{(\bar{x}_1, y_1), (\bar{x}_2, y_2), \ldots, (\bar{x}_P, y_P)\}$$

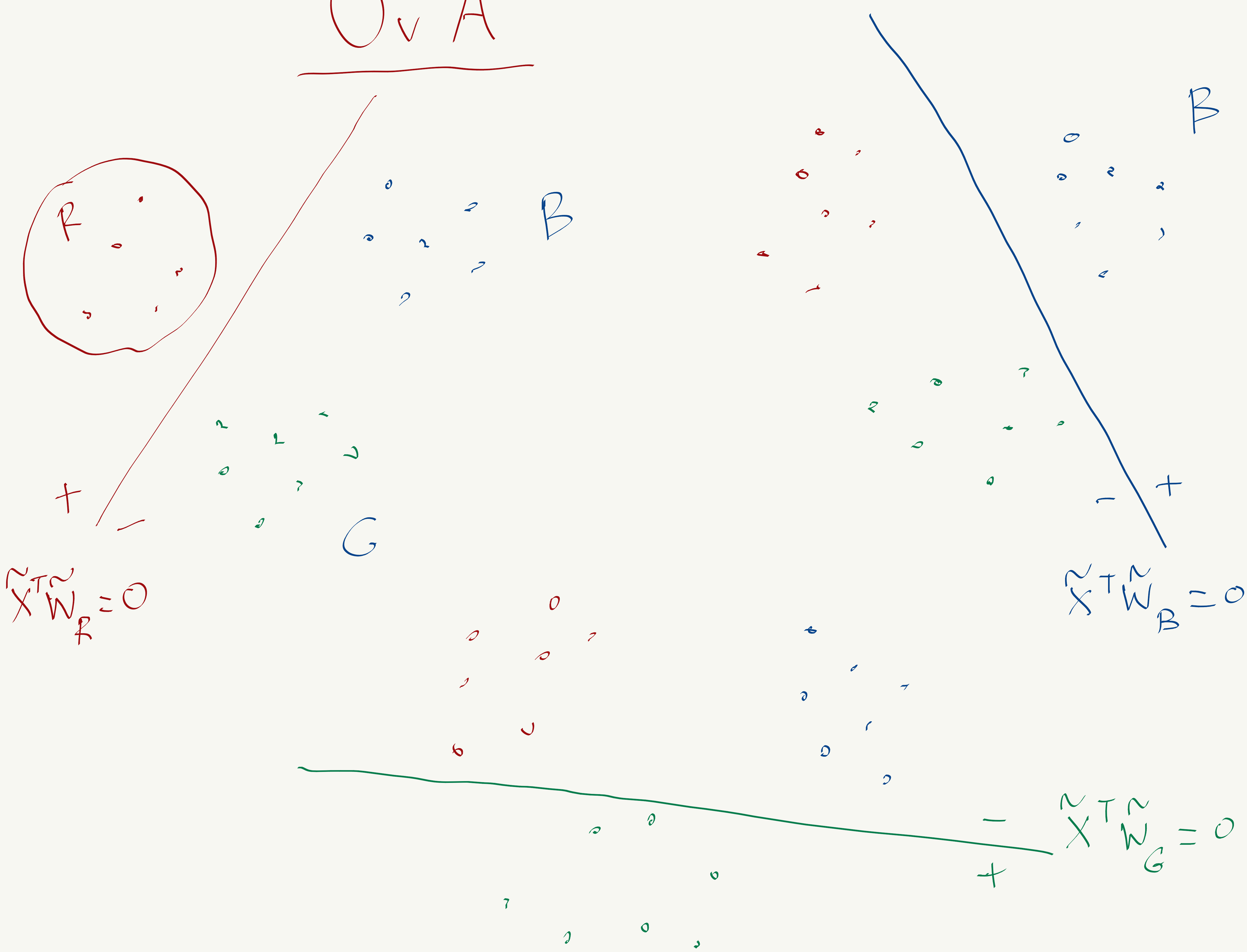
labels $y_p \in \underbrace{\{0, 1, \ldots, C-1\}}_{C \text{ classes}}$

2 Solution Approaches

- One-versus-All (OvA)
- solve the problem direction
  - perceptron
  - softmax


R
B
G

OvA


R
B
G
B
G
+
−
$\tilde{X}^T\tilde{W}_R = 0$
+
−
$\tilde{X}^T\tilde{W}_B = 0$
−
+
$\tilde{X}^T\tilde{W}_G = 0$

OvA

1) • Learn C linear classifiers

2) • $\tilde{x}^T \tilde{w}_c = 0$, $c = 0, \ldots, C-1$

3) • Each individual classifier

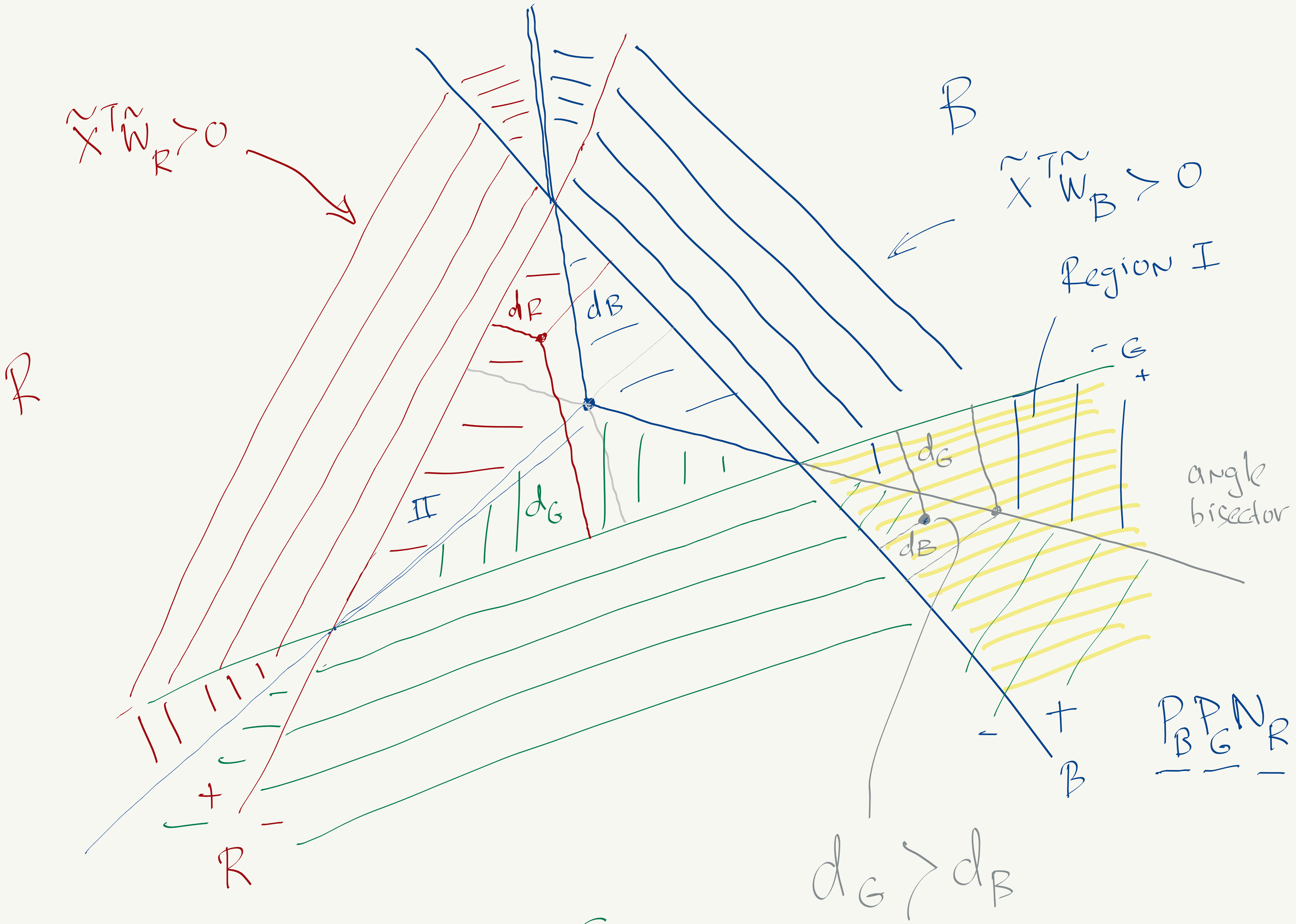
- assigns temporary labels

$$\tilde{y}_p = \begin{cases} 1, & \bar{x}_p \in c \quad (y_p = c) \\ 0, & \text{else} \quad (y_p \neq c) \end{cases}$$

- a point $\bar{x}_p$ belongs to class c

if

$$\boxed{\begin{array}{l} \tilde{x}_p^T \tilde{w}_c > 0 \\ \tilde{x}_p^T \tilde{w}_j < 0 \, , \quad j = 0, \ldots, C-1, \; j \neq c \end{array}}$$



Region I

signed distance of $\bar{x}$ to the j-th boundary $= \dfrac{\tilde{x}^T\tilde{w}_j}{\|\bar{w}\|_2}$

Normalize $\tilde{w}_j \leftarrow \dfrac{\tilde{w}_j}{\|\bar{w}\|_2}$

$\tilde{x} = \begin{bmatrix} 1 \\ \bar{x} \end{bmatrix}$

$\therefore$ Signed distance : $\tilde{x}^T\tilde{w}_j$

$\tilde{w} = \begin{bmatrix} w_0\,(b) \\ \bar{w} \end{bmatrix}$

SO FAR.

$$\boxed{y = \arg\max_{j=0,\dots,C-1} \tilde{x}^T\tilde{w}_j}$$ ✓

Region II

$d_R > d_B > d_G$ (all negative numbers)

$$\boxed{\text{FUSION RULE FOR OvA} \qquad y = \arg\max_{j=0,\ldots,C-1} \tilde{x}^T \tilde{w}_j}$$

$\tilde{w}_j$, $j = 0, \ldots, C-1$

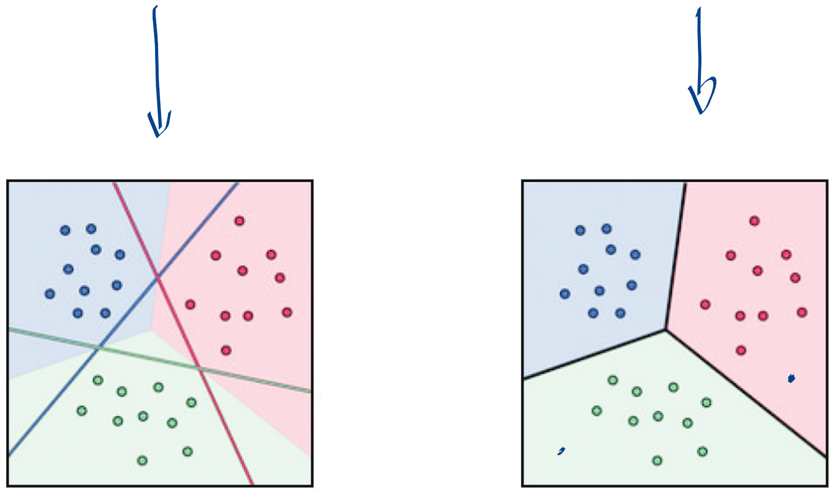


Figure 7.6 The result of applying the fusion rule in Equation (7.12) to the input space of our toy dataset, with regions colored according to their predicted label along with the original two-class decision boundaries (left panel) and fused multi-class decision boundary in black (right panel). See text for further details.

Fusion

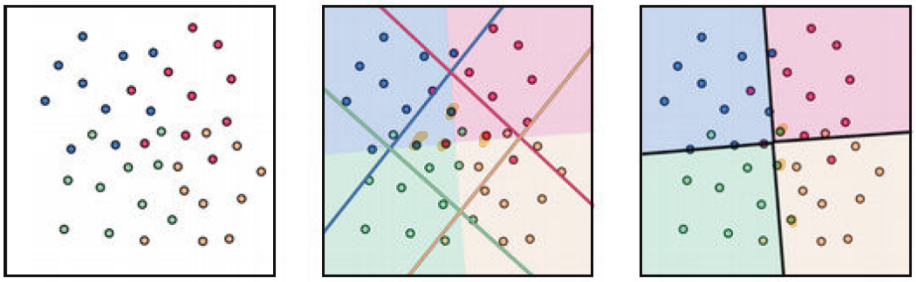


Figure 7.7 Figure associated with Example 7.1. See text for details.

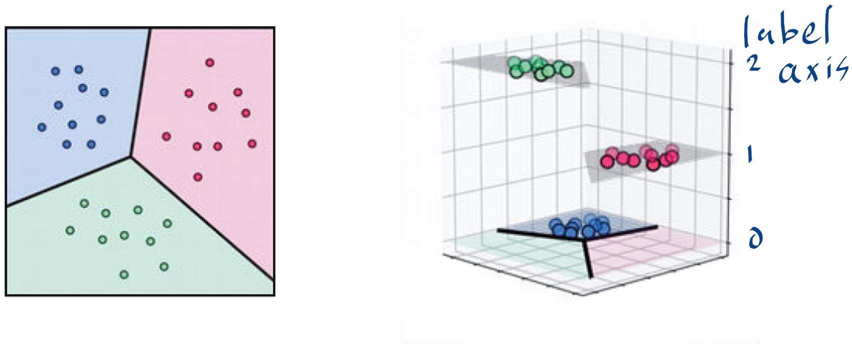


Figure 7.8 Fusion rule for a toy classification dataset with $C = 3$ classes, shown from the perceptron perspective (left panel) and from the regression perspective (right panel). Note that the jagged edges on some of the steps in the right panel are merely an artifact of the plotting mechanism used to generate the three dimensional plot. In reality the edges of each step are smooth like the fused decision boundary shown in the input space. See Example 7.2 for further details.

Multi-class Perceptron

$$\underline{y_p} = \underbrace{\underset{j=0,..,C-1}{\arg\max}}\ \tilde{x}_p^T \tilde{w}_j$$

$$\Rightarrow \tilde{x}_p^T \tilde{w}_{y_p} = \max_{j=0,..,C-1} \tilde{x}_p^T \tilde{w}_j$$

$$\Rightarrow \underbrace{\max_{j=0,..,C-1} \tilde{x}_p^T \tilde{w}_j - \tilde{x}_p^T \tilde{w}_{y_p}} = 0$$

if $\bar{x}_p$ classified correctly

$\tilde{x}_p^T \tilde{w}_{y_p}$

If we consider all P points

→ Cost function

$$g(\tilde{w}_0, \tilde{w}_1, .., \tilde{w}_{C-1}) = \frac{1}{P} \sum_{p=1}^{P} \left[\underbrace{\max_{j=0,..,C-1} \tilde{x}_p^T \tilde{w}_j} - \underbrace{\tilde{x}_p^T \tilde{w}_{y_p}} \right]$$

$$\max(s_0, s_1, \ldots, s_{C-1}) - \underset{\substack{\uparrow \\ \text{constant}}}{z} = \max(s_0 - z, \ldots, s_{C-1} - z)$$

$\therefore$

$$\max_j \left(\tilde{x}_p^T \tilde{w}_j\right) - \tilde{x}_p^T \tilde{w}_{y_p}$$

$$= \max_j \left(\underbrace{\tilde{x}_p^T \left(\tilde{w}_j - \tilde{w}_{y_p}\right)}\right)$$

$$= \max_{\substack{j=0,\ldots,C-1 \\ j \neq y_p}} \left(0, \tilde{x}_p^T \left(\tilde{w}_j - \tilde{w}_{y_p}\right)\right)$$

$$\boxed{g(\cdot) = \frac{1}{P} \sum_p \max_{\substack{j \\ j \neq y_p}} \left(0, \tilde{x}_p^T \left(\tilde{w}_j - \tilde{w}_{y_p}\right)\right)}$$

$$\frac{1}{P} \sum_{c=0}^{C-1} \sum_p \max_{\substack{j \\ j \neq c}} \left(0, \tilde{x}_p^T \left(\tilde{w}_j - \tilde{w}_c\right)\right)$$

Multi-class softmax

$$g(\tilde{w}_0, \ldots, \tilde{w}_{C-1}) = \frac{1}{P}\sum_{p=1}^{P}\left[\max_j \tilde{x}_p^T\tilde{w}_j - \tilde{x}_p^T\tilde{w}_{y_p}\right]$$

$$= \frac{1}{P}\sum_{p=1}^{P}\max\left(\tilde{x}_p^T\tilde{w}_0, \tilde{x}_p^T\tilde{w}_1, \ldots\right) - \tilde{x}_p^T\tilde{w}_{y_p}$$

$$\boxed{\begin{aligned}&\text{soft}(s_0, \ldots, s_{C-1}) = \log\left(\sum_j e^{s_j}\right)\\ &\approx \max(s_0, \ldots, s_{C-1})\end{aligned}}$$

$$= \frac{1}{P}\sum_{p}\text{soft}\left(\sum_{j=0}^{C-1}\tilde{x}_p^T\tilde{w}_j - \tilde{x}_p^T\tilde{w}_{y_p}\right)$$

7.23

$$= \frac{1}{P}\sum_{p=1}^{P}\left[\log\left(\sum_{j=0}^{C-1}e^{\tilde{x}_p^T\tilde{w}_j}\right) - \underbrace{\tilde{x}_p^T\tilde{w}_{y_p}}_{= \log e^{\tilde{x}_p^T\tilde{w}_{y_p}}}\right]$$

$$\boxed{a = \log e^a}$$

$$= \frac{1}{P}\sum_{p=1}^{P}\log\frac{\sum_{j=0}^{C-1}e^{\tilde{x}_p^T\tilde{w}_j}}{e^{\tilde{x}_p^T\tilde{w}_{y_p}}}$$

$$\left(= \frac{1}{P}\sum_p\underbrace{\left(1 + \log\frac{\sum_{j=0, j\neq y_p}^{C-1}e^{\tilde{x}_p^T\tilde{w}_j}}{e^{\tilde{x}_p^T\tilde{w}_{y_p}}}\right)}_{7.24}\right)$$

$$\boxed{g(\cdot) = -\frac{1}{P}\sum_{p=1}^{P}\log\frac{e^{\tilde{x}_p^T\tilde{w}_{y_p}}}{\sum_j e^{\tilde{x}_p^T\tilde{w}_j}}}$$