# One-versus-the-Rest

We extend SVM's in order to support multi-class classification problems.

Consider the training dataset

$$D = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \ldots, (\overline{x}_l, y_l)\} \subset \mathbb{R}^n \times \{1, \ldots, M\},$$

where the label $y_i$ can take on any label in $\{1, \ldots, M\}$ with $i = 1, \ldots, n$ and $M > 2$.

The most popular technique for multi-class classification is called ***one-versus-the-rest*** classification.

Here we build $M$ decision surfaces, say $g^1, \ldots, g^M$, each trained to separate one class from the rest. That is, the decision surface $g^1$ is trained to separate the class labeled $1$ from all other classes, the decision surface $g^2$ is trained to separate the class labeled $2$ from all other classes, *etc*.

In order to classify an unknown point $\overline{x}$ we use a ***voting scheme*** based on which one of the $M$ decision surfaces returns the ***largest value*** for the point $\overline{x}$. We interpret this as selecting the decision surface that separates the point $\overline{x}$ with the ***highest confidence*** from the majority of points.

# Models: Training Sets

In order to train $M$ binary decision surfaces we have to construct appropriate binary training datasets.

Let $p \in \{1, \ldots, M\}$, then for each decision surface $g^p$ we construct the binary training set $D^p = D^p_+ \cup D^p_-$ where

$$\begin{aligned} D^p_+ &= \{(\overline{x}, +1) \mid (\overline{x}, y) \in D \wedge y = k\}, \\ D^p_- &= \{(\overline{x}, -1) \mid (\overline{x}, y) \in D \wedge y \neq k\}. \end{aligned}$$

**Observation:** The set $D^p_+$ contains all the points that are members of the class $p$ and the set $D^p_-$ contains all the remaining points.

# Models: Decision Surfaces

Using our binary training datasets we construct the decision surfaces

$$g^p(\overline{x}) = \sum_{i=1}^{|D^p|} y_i^p \alpha_i^p k(\overline{x}_i, \overline{x}) - b^p$$

with $p \in \{1, \ldots, M\}$ and $(\overline{x}_i, y_i) \in D^p$.

**Note:** The function $g^p(\overline{x})$ returns a signed real value and this value can be interpreted as the distance from the decision surface to the point $\overline{x}$. This value can also be interpreted as a ***confidence value***; the larger the value the more confident we are that the point $\overline{x}$ belong to the positive class.

# Models: Voting Scheme

We can use the confidence value as our criterion to pick the best decision surface: we pick the decision surface with the largest confidence value. That is, we assign point $\overline{x}$ to the class whose decision surface returns the largest value for this point.

More formally, we can construct the decision function $\hat{f}: \mathbb{R}^n \rightarrow \{1, \ldots, M\}$ for our multi-class classification problem as follows,

$$\hat{f}(\overline{x}) = \underset{k}{\operatorname{argmax}}\, g^p(\overline{x}),$$

where $p \in \{1, \ldots, M\}$ and $\overline{x} \in \mathbb{R}^n$.

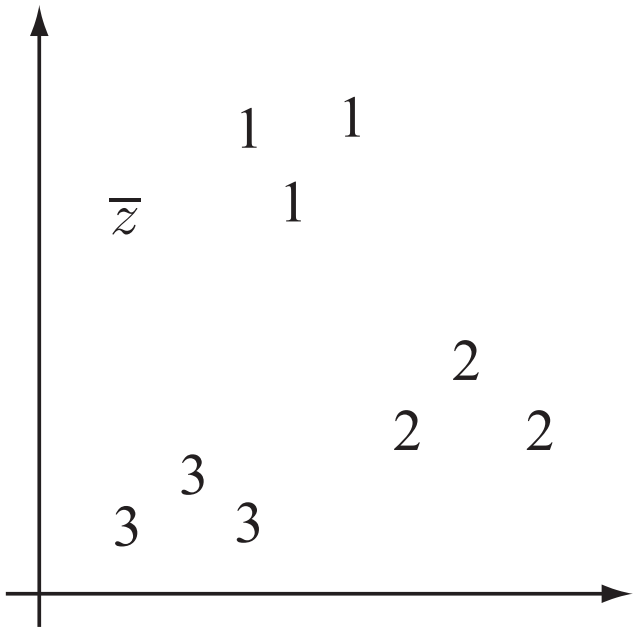
**Note:** The function $\operatorname{argmax}$ returns the value of $p$ that maximizes the function $g^p$.

# An Example

As an example we look at a classification problem with three classes where the training dataset $D$ is defined as

$$D = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \ldots, (\overline{x}_l, y_l)\} \subset \mathbb{R}^2 \times \{1, 2, 3\},$$

with $l = 9$.



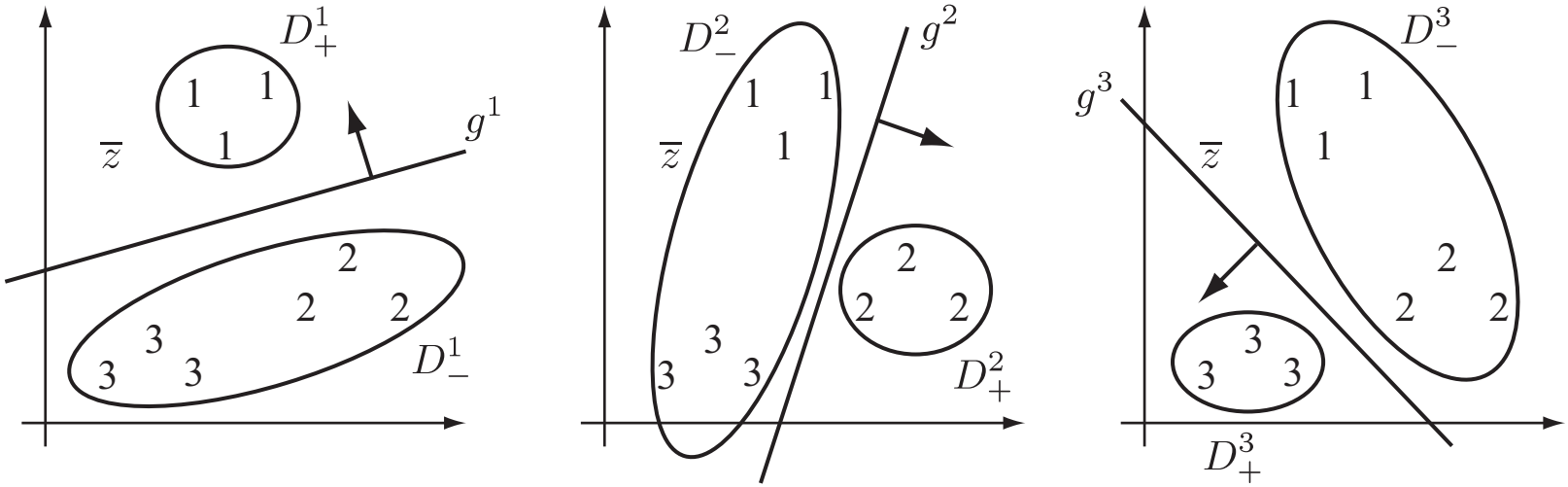

The label of point $\overline{z}$ is unknown and should be estimated from the dataset.

# An Example

We can proceed with our construction and build three training datasets, one for each decision surface that separates one class from the rest,

$$\begin{aligned} D^1 &= D^1_+ \cup D^1_-, \\ D^2 &= D^2_+ \cup D^2_-, \\ D^3 &= D^3_+ \cup D^3_-. \end{aligned}$$

We then construct the decision surfaces, $g^1$, $g^2$, and $g^3$, one for each of these datasets, respectively.




Now: for $\hat{f}: \mathbb{R}^n \rightarrow \{1, 2, 3\}$, $\hat{f}(\overline{z}) = ?$

# An Example

**Observation:** The training sets for the individual decision surfaces in one-versus-the-rest tend to be highly unbalanced.

# Pairwise Classification

Solution: build decision surfaces for ***each pair of classes***.

Given the training set

$$D = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \ldots, (\overline{x}_l, y_l)\} \subset \mathbb{R}^n \times \{1, 2, \ldots, M\},$$

we let $g^{p,q} : \mathbb{R}^n \rightarrow \{p, q\}$ denote the decision surface that separates the pair of classes $p$ and $q$ with $p \neq q$ and $\{p, q\} \subset \{1, 2, \ldots, M\}$. We train each decision surface,

$$g^{p,q}(\overline{x}) = \sum_{i=1}^{|D^{p,q}|} y_i \alpha_i^{p,q} k(\overline{x}_i, \overline{x}) - b^{p,q},$$

on the data set,

$$D^{p,q} = D^p \cup D^q,$$

where

$$D^p = \{(\overline{x}, y) \mid (\overline{x}, y) \in D \wedge y = p\},$$

and

$$D^q = \{(\overline{x}, y) \mid (\overline{x}, y) \in D \wedge y = q\}.$$

# Pairwise Classification

Once we have constructed all the pairwise decision surfaces $g^{p,q}$ using the corresponding training sets $D^{p,q}$, we can classify an unknown point by applying each of the $M(M-1)/2$ decision surfaces to this point keeping track of how many times the point was assigned to what class label. The class label with the highest count is then considered the label for the unknown point.

# Pairwise Classification

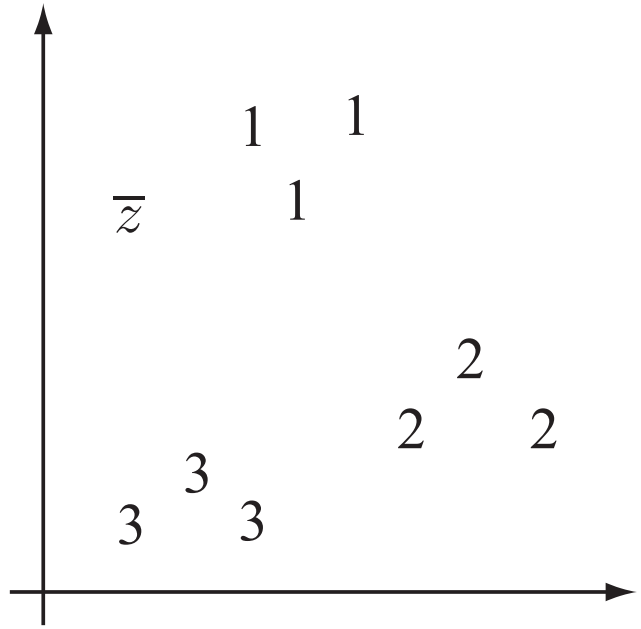
*// multi-class training set*
**let** $D = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \ldots, (\overline{x}_l, y_l)\} \subset \mathbb{R}^n \times \{1, 2, \ldots, M\}$
*// point to be classified*
**let** $\overline{z} \in \mathbb{R}^n$
*// initialize the counter for the labels to zero*
**let** $cnt[1 \ldots M] = \overline{0}$
*// loop through all possible pairs of labels*
**for** $p = 1$ **to** $M$ **do**
  **for** $q = p + 1$ **to** $M$ **do**
    *// construct the decision surface for this pair of labels*
    **let** $D^{p,q} = D^p \cup D^q$
    **train** $g^{p,q}$ **on** $D^{p,q}$
    *// classify the unknown point with the current decision surface*
    *// and increment the appropriate counter*
    **if** $g^{p,q}(\overline{z}) == p$ **then**
      $cnt[p] + +$
    **else**
      $cnt[q] + +$
    **end if**
  **end for**
**end for**
*// return the label with the largest count*
**return** $\text{argmax}_{i=1,\ldots,M}\ (cnt[i])$

# An Example

Consider the same training set from before,

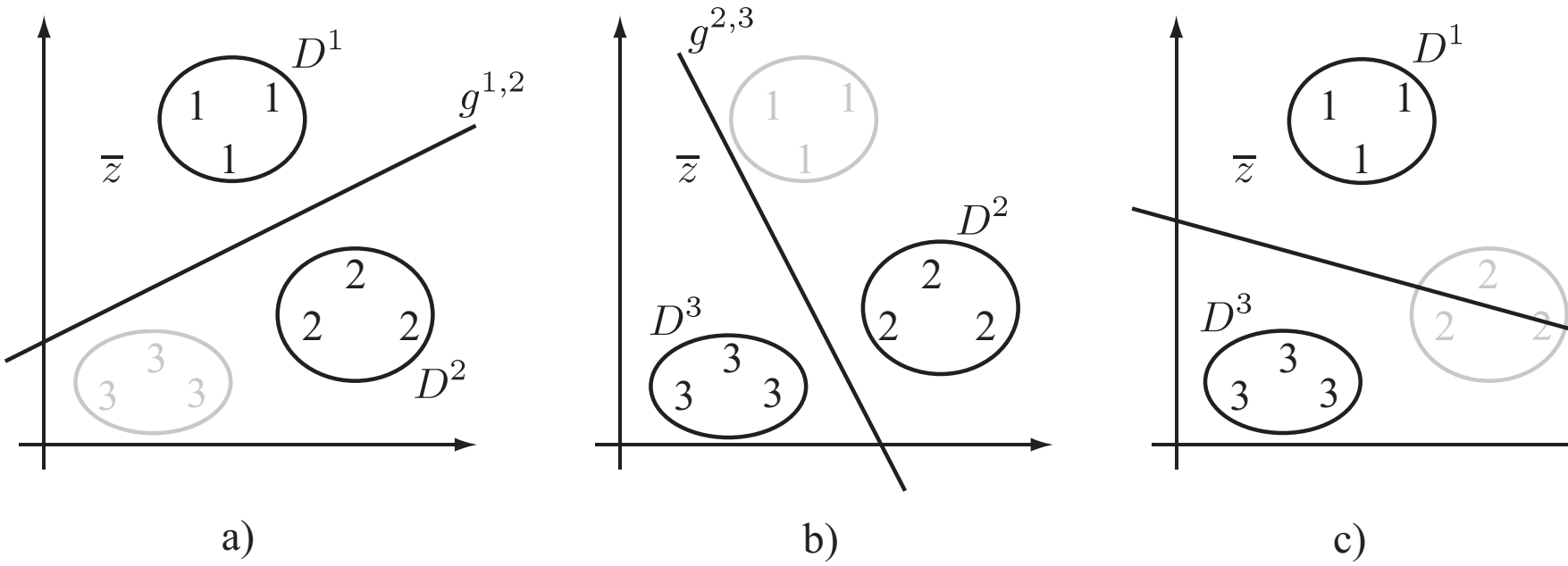
$$D = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \ldots, (\overline{x}_l, y_l)\} \subset \mathbb{R}^2 \times \{1, 2, 3\},$$

with $l = 9$.




The label of point $\overline{z}$ is unknown and should be estimated from the dataset.

# An Example




The three training data sets, $D^p \cup D^q$ and the corresponding decision surfaces $g^{p,q}$ with $\{p, q\} \subset \{1, 2, 3\}$; part a) shows the case for $p = 1$ and $q = 2$, part b) for $p = 2$ and $q = 3$, and part c) for $p = 3$ and $q = 1$. Here, the point $\overline{z}$ belongs to class $1$ because the largest score,

| Class $1$ | Class $2$ | Class $3$ |
|---|---|---|
| $2$ | $0$ | $1$ |

# Observations

- Both the One-versus-the-Rest and the Pairwise Classification schemes coincide in the classification of point $\overline{z}$.
- In pairwise classification the training sets are more balanced, however, we have to construct $M(M-1)/2$ of them together with their corresponding decision surfaces.
- The package 'e1071' implements the pairwise classification scheme for multi-class classification.