Linear classification

Course of Machine Learning
Master Degree in Computer Science
University of Rome ``Tor Vergata''

Giorgio Gambosi

a.a. 2017-2018

Introduction

Classification

- value t to predict are from a discrete domain, where each value denotes a class
- most common case: disjoint classes, each input has to assigned to exactly one class
- input space is partitioned into decision regions
- in linear classification models decision boundaries are linear functions of input $\mathbf{x}$ ($D-1$-dimensional hyperplanes in the D-dimensional feature space)
- datasets such as classes correspond to regions which may be separated by linear decision boundaries are said linearly separable

- Regression: the target variable $\mathbf{t}$ is a vector of reals
- Classification: several ways to represent classes (target variable values)
- Binary classification: a single variable $t \in \{0, 1\}$, where $t = 0$ denotes class C_0 and $t = 1$ denotes class C_1
- $K > 2$ classes: ``1 of K'' coding. $\mathbf{t}$ is a vector of K bits, such that for each class C_j all bits are 0 except the j-th one (which is 1)

Approaches to classification

Three general approaches to classification

1. find $f : \mathbf{X} \mapsto \{1, \dots, K\}$ (discriminant function) which maps each input $\mathbf{x}$ to some class C_i (such that $i = f(\mathbf{x})$)
2. discriminative approach: determine the conditional probabilities $p(C_j|\mathbf{x})$ (inference phase); use these distributions to assign an input to a class (decision phase)
3. generative approach: determine the class conditional distributions $p(\mathbf{x}|C_j)$, and the class prior probabilities $p(C_j)$; apply Bayes' formula to derive the class posterior probabilities $p(C_j|\mathbf{x})$; use these distributions to assign an input to a class

Discriminative approaches

- Approaches 1 and 2 are discriminative: they tackle the classification problem by deriving from the training set conditions (such as decision boundaries) that , when applied to a point, discriminate each class from the others
- The boundaries between regions are specify by *discrimination functions*

Generalized linear models

- In linear regression, a model predicts the target value; the prediction is made through a linear function $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ (linear basis functions could be applied)
- In classification, a model predicts probabilities of classes, that is values in $[0, 1]$; the prediction is made through a generalized linear model $y(\mathbf{x}) = f(\mathbf{w}^T\mathbf{x} + w_0)$, where f is a non linear activation function with codomain $[0, 1]$
- boundaries correspond to solution of $y(\mathbf{x}) = c$ for some constant c; this results into $w^T\mathbf{x} + w_0 = f^{-1}(c)$, that is a linear boundary. The inverse function f^{-1} is said link function.

Generative approaches

- Approach 3 is generative: it works by defining, from the training set, a model of items for each class
- The model is a probability distribution (of features conditioned by the class) and could be used for random generation of new items in the class
- By comparing an item to all models, it is possible to verify the one that best fits

Discriminant functions

Linear discriminant functions in binary classification

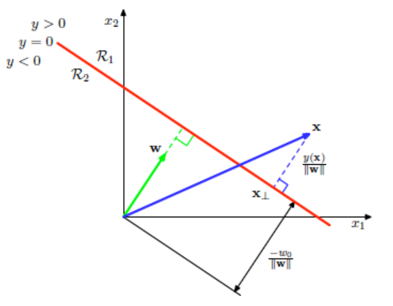
- Decision boundary: $D-1$-dimensional hyperplane $y(\mathbf{x})=0$ of all points s.t. $w_0+\sum_{i=1}^{D} w_i x_i = 0$, that is $\mathbf{w}^T\mathbf{x}=-w_0$, where $\mathbf{w}=(w_1,\ldots,w_D)$
- Given $\mathbf{x}_1, \mathbf{x}_2$ on the hyperplane, $y(\mathbf{x}_1)=y(\mathbf{x}_2)=0.$ Hence,

  $$\mathbf{w}^T\mathbf{x}_1=\mathbf{w}^T\mathbf{x}_2 \implies \mathbf{w}^T(\mathbf{x}_1-\mathbf{x}_2)=\mathbf{w}\cdot(\mathbf{x}_1-\mathbf{x}_2)=0$$

  $\mathbf{x}_1-\mathbf{x}_2$, $\mathbf{w}$ orthogonal
- For any $\mathbf{x}$ s.t. $y(\mathbf{x})=0$, $\mathbf{w}^T\mathbf{x}$ is the length of the projection of $\mathbf{x}$ in the direction of $\mathbf{w}$ (orthogonal to the hyperplane $y(\mathbf{x})=0$)
- By normalizing wrt to $||\mathbf{w}||_2=\sqrt{\sum_i w_i^2}$, we get the length of the projection of $\mathbf{x}$ in the direction orthogonal to the hyperplane, in multiples of $||\mathbf{w}||_2$

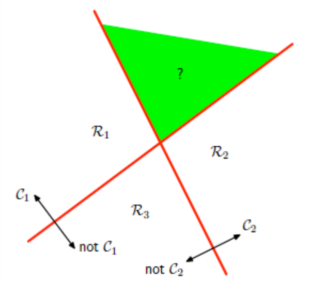
Linear discriminant functions in binary classification

- In general, $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ returns the distance (in multiples of $||\mathbf{w}||$) of $\mathbf{x}$ from the hyperplane
- The sign of the returned value discriminates in which of the regions separated by the hyperplane the point lies



First approach

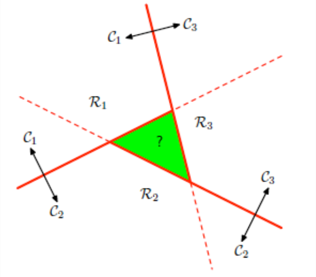
- Define $K-1$ discrimination functions
- Function f_i $(1 \leq i \leq K-1)$ discriminates points belonging to class C_i from points belonging to all other classes: if $f_i(\mathbf{x}) > 0$ then $\mathbf{x} \in C_i$, otherwise $\mathbf{x} \notin C_i$
- The green region belongs to both $\mathcal{R}_1$ and $\mathcal{R}_2$



Linear discriminant functions in multiclass classification

Second approach

- Define $K(K-1)/2$ discrimination functions, one for each pair of classes
- Function f_{ij} $(1 \leq i < j \leq K)$ discriminates points which might belong to C_i from points which might belong to C_j
- Item $\mathbf{x}$ is classified on a majority basis
- The green region is unassigned



Third approach

- Define K linear functions

$$y_i(\mathbf{x}) = \mathbf{w}_i^T\mathbf{x} + w_{i0} \qquad\qquad 1 \leq i \leq K$$

  Item $\mathbf{x}$ is assigned to class C_k iff $y_k(\mathbf{x}) > y_j(\mathbf{x})$ for all $j \neq k$: that is,

$$k = \underset{j}{\operatorname{argmax}}\, y_j(\mathbf{x})$$

- Decision boundary between C_i and C_j: all points $\mathbf{x}$ s.t. $y_i(\mathbf{x}) = y_j(\mathbf{x})$, a $D-1$-dimensional hyperplane

$$(\mathbf{w}_i - \mathbf{w}_j)^T\mathbf{x} + (w_{i0} - w_{j0}) = 0$$

The resulting decision regions are connected and convex

Generalized discriminant functions

- The definition can be extended to include terms relative to products of pairs of feature values (Quadratic discriminant functions)

$$y(\mathbf{x}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D} \sum_{j=1}^{i} w_{ij} x_i x_j$$

  $\frac{d(d+1)}{2}$ additional parameters wrt the $d+1$ original ones: decision boundaries can be more complex

- In general, generalized discrimination functions through set of functions $\phi_i, \ldots, \phi_m$

$$y(\mathbf{x}) = w_0 + \sum_{i=1}^{M} w_i \phi_i(\mathbf{x})$$

Least squares and classification

Linear discriminant functions and regression

- Assume classification with K classes
- Classes are represented through a 1-of-K coding scheme: set of variables $z_1, \dots, z_K$, class C_i coded by values $z_i = 1$, $z_k = 0$ for $k \neq i$
- Discriminant functions y_i are derived as linear regression functions with variables z_i as targets
- To each variable z_i a discriminant function $y_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$ is associated: $\mathbf{x}$ is assigned to the class C_k s.t.

$$k = \underset{i}{\operatorname{argmax}}\, y_i(\mathbf{x})$$

- Then, $z_k(\mathbf{x}) = 1$ and $z_j(\mathbf{x}) = 0$ $(j \neq k)$ if $k = \underset{i}{\operatorname{argmax}}\, y_i(\mathbf{x})$
- Group all parameters together as

$$\mathbf{y}(\mathbf{x}) = \overline{\mathbf{W}}^T \mathbf{x}$$

where the i-th column of $\overline{\mathbf{W}}$ provides the coefficients $\mathbf{w}_i, w_{i0}$

- In general, a regression function provides an estimation of the target given the input: in particular, it estimates the expectation $E[t|\mathbf{x}]$
- Here, $y_i(\mathbf{x})$ can then be seen as a (poor) estimation of $E[z_i|\mathbf{x}]$, the conditional expectation of variable z_i given $\mathbf{x}$
- Observe that in this case $p(z|\mathbf{x})$ is a Bernoulli distribution, as a consequence

$$\begin{aligned} E[z_i|\mathbf{x}] &= P(z_i = 1|\mathbf{x}) \cdot 1 + P(z_i = 0|\mathbf{x}) \cdot 0 \\ &= P(z_i = 1|\mathbf{x}) \\ &= P(C_i|\mathbf{x}) \end{aligned}$$

- Hence, $y_i(\mathbf{x})$ is an estimate of $p(C_i|\mathbf{x})$. However, $y_i(\mathbf{x})$ is not a probability

Learning functions y_i

- Given a training set $(\mathbf{X}, \mathbf{t})$, each regression functions $y_i(\mathbf{x})$ is derived by least squares
- An item in $(\mathbf{X}, \mathbf{t})$ is a pair $(\mathbf{x}_i, \mathbf{t}_i)$, $\mathbf{x}_i \in \mathbb{R}^D$ and $\mathbf{t}_i \in \{0,1\}^K$ with $\sum_j t_{ij} = 1$
- $\overline{\mathbf{W}} \in \mathbb{R}^{(D+1)\times K}$ is the matrix of coefficients of all functions y_i: the i-th column represents the $D+1$ parameters $w_{i0}, \dots, w_{iD}$ of y_i

$$\overline{\mathbf{W}} = \begin{bmatrix} w_{10} & w_{20} & \cdots & w_{K0} \\ w_{11} & w_{21} & \cdots & w_{K1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{1D} & w_{2D} & \cdots & w_{KD} \end{bmatrix}$$

- $\mathbf{y}(\mathbf{x}) = \overline{\mathbf{W}}^T \overline{\mathbf{x}}$ where $\overline{\mathbf{x}} = (1, x_1, \dots, x_d)$

- $\overline{\mathbf{X}} \in \mathbb{R}^{n\times(D+1)}$ is the matrix of feature values for all items in the training set, with a first column of 1 values

$$\overline{\mathbf{X}} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(D)} & x_2^{(D)} & \cdots & x_n^{(D)} \end{bmatrix}$$

- Then, for matrix $\mathbf{Y} = \overline{\mathbf{W}}^T\overline{\mathbf{X}}$, of size $K \times n$, we have

$$(\overline{\mathbf{W}}^T\overline{\mathbf{X}})_{ij} = w_{i0} + \sum_{k=1}^{D} x_j^{(k)} w_{ik} = y_i(\mathbf{x}_j)$$

  that is, the i-th column of $\mathbf{Y}$ contains the values $y_1(\mathbf{x}_i), \ldots, y_K(\mathbf{x}_i)$.

- $\mathbf{Y}_{ij} = y_i(\mathbf{x}_j)$ is compared to item $\mathbf{T}_{ij}$ in the matrix $\mathbf{T}$, of size $K \times n$, of target values, where column j is the 1-of-K coding $t_{j1}, \dots, t_{jK}$ of the class of item $\mathbf{x}_j$

$$(\mathbf{Y} - \mathbf{T})_{ij} = y_i(\mathbf{x}_j) - t_{ji}$$

- Let us consider the diagonal items of $(\mathbf{Y} - \mathbf{T})^T(\mathbf{Y} - \mathbf{T})$. Then,

$$((\mathbf{Y} - \mathbf{T})^T(\mathbf{Y} - \mathbf{T}))_{jj} = \sum_{i=1}^{K} (y_i(\mathbf{x}_j) - t_{ji})^2$$

  That is, for each $\mathbf{x}_j$,

$$((\mathbf{Y} - \mathbf{T})^T(\mathbf{Y} - \mathbf{T}))_{jj} = (y_k(\mathbf{x}_j) - 1)^2 + \sum_{i \neq k} y_i(\mathbf{x}_j)^2$$

  where we assumed $\mathbf{x}_j \in C_k$.

Learning functions y_i

- Summing all elements on the diagonal of $((\mathbf{Y}-\mathbf{T})^T(\mathbf{Y}-\mathbf{T}))$ provides the overall sum, on all items in $\mathbf{X}$, of the squared differences between observed values (class identifiers coded 1-to-K) and values computed by the model, with parameters $\overline{\mathbf{W}}$
- This corresponds to the trace of $((\mathbf{Y}-\mathbf{T})^T(\mathbf{Y}-\mathbf{T}))$. Hence, we have to minimize:

$$E(\overline{\mathbf{W}}) = \frac{1}{2} tr((\overline{\mathbf{W}}^T\overline{\mathbf{X}}-\mathbf{T})^T(\overline{\mathbf{W}}^T\overline{\mathbf{X}}-\mathbf{T}))$$

- Standard approach, solve

$$\frac{\partial E(\overline{\mathbf{W}})}{\partial \overline{\mathbf{W}}} = \mathbf{0}$$

It is possible to show that this results into the set of discriminant functions

$$\mathbf{y}(\mathbf{x}) = \overline{\mathbf{W}}^T \overline{\mathbf{x}} = \mathbf{T}\overline{\mathbf{X}}(\overline{\mathbf{X}}^T \overline{\mathbf{X}})^{-1} \overline{\mathbf{x}}$$

Fisher linear discriminant

Approach

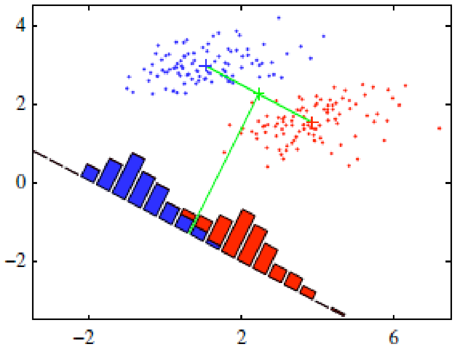
- The idea of *Linear Discriminant Analysis* (*LDA*) is to find a linear projection of the training set into a suitable subspace where classes are as linearly separated as possible
- A common approach is provided by Fisher linear discriminant, where all items in the training set (points in a D-dimensional space) are projected to one dimension, by means of a transformation of the type

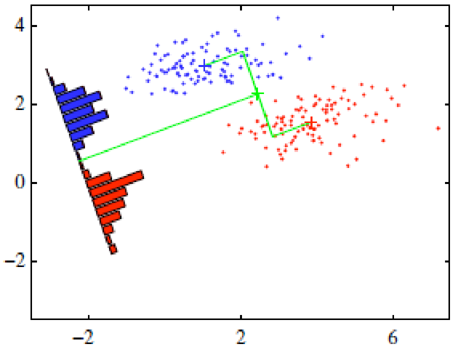
$$y = \mathbf{w} \cdot \mathbf{x} = \mathbf{w}^T \mathbf{x}$$

  where $\mathbf{w}$ is the D-dimensional vector corresponding to the direction of projection (in the following, we will consider the one with unit norm).

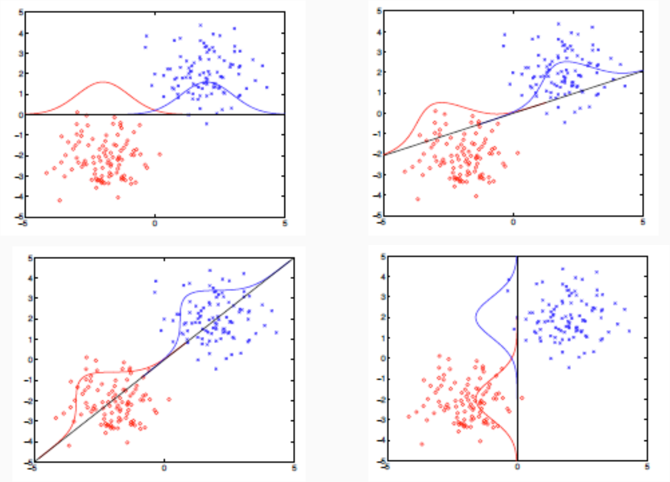
LDA

If $K = 2$, given a threshold $\tilde{y}$, item $\mathbf{x}$ is assigned to C_1 iff its projection $y = \mathbf{w}^T\mathbf{x}$ is such that $y > \tilde{y}$; otherwise, $\mathbf{x}$ is assigned to C_2.





Different line directions, that is different parameters $\mathbf{w}$, may induce quite different separability properties.



Let n_1 be the number of items in the training set belonging to class C_1 and n_2 the number of items in class C_2. The mean points of both classes are

$$\mathbf{m}_1 = \frac{1}{n_1} \sum_{\mathbf{x} \in C_1} \mathbf{x} \qquad \mathbf{m}_2 = \frac{1}{n_2} \sum_{\mathbf{x} \in C_2} \mathbf{x}$$

A simple measure of the separation of classes, when the training set is projected onto a line, is the difference between their mean points

$$m_2 - m_1 = \mathbf{w}^T (\mathbf{m}_2 - \mathbf{m}_1)$$

where $m_i = \mathbf{w}^T \mathbf{m}_i$ is the projection of $\mathbf{m}_i$ onto the line.

Deriving $\mathbf{w}$ in the binary case

- We wish to find a line direction $\mathbf{w}$ such that $m_2 - m_1$ is maximum
- $\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1)$ can be made arbitrarily large by multiplying $\mathbf{w}$ by a suitable constant, at the same time maintaining the direction unchanged. To avoid this drawback, we consider unit vectors, introducing the constraint $||\mathbf{w}||_2 = \mathbf{w}^T\mathbf{w} = 1$
- This results in an optimization with a lagrangian multiplier: we wish to maximize the following function of $\mathbf{w}$ and λ

$$\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1) + \lambda(1 - \mathbf{w}^T\mathbf{w})$$

Deriving $\mathbf{w}$ in the binary case

Setting the gradient of the function wrt $\mathbf{w}$ to $\mathbf{0}$

$$\frac{\partial}{\partial \mathbf{w}}(\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1) + \lambda(1 - \mathbf{w}^T\mathbf{w})) = \mathbf{m}_2 - \mathbf{m}_1 + 2\lambda\mathbf{w} = \mathbf{0}$$

results into

$$\mathbf{w} = \frac{\mathbf{m}_2 - \mathbf{m}_1}{2\lambda}$$

Deriving $\mathbf{w}$ in the binary case

Setting the derivative wrt λ to 0

$$\frac{\partial}{\partial \lambda}(\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1) + \lambda(1 - \mathbf{w}^T\mathbf{w})) = 1 - \mathbf{w}^T\mathbf{w} = 0$$

results into

$$1 - \mathbf{w}^T\mathbf{w} = 1 - \frac{(\mathbf{m}_2 - \mathbf{m}_1)^T(\mathbf{m}_2 - \mathbf{m}_1)}{4\lambda^2} = 0$$

that is

$$\lambda = \frac{\sqrt{(\mathbf{m}_2 - \mathbf{m}_1)^T(\mathbf{m}_2 - \mathbf{m}_1)}}{2} = \frac{||\mathbf{m}_2 - \mathbf{m}_1||_2}{2}$$

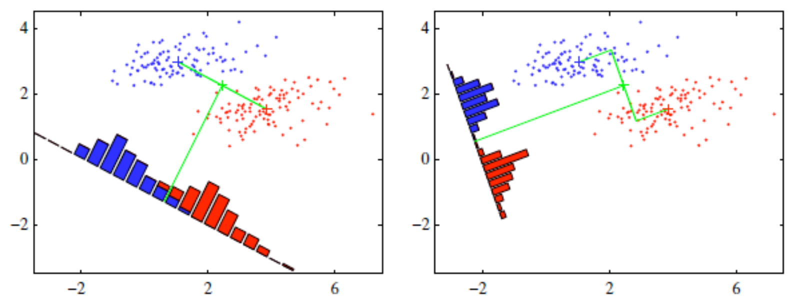
Combining with the result for the gradient,

$$\mathbf{w} = \frac{\mathbf{m}_2 - \mathbf{m}_1}{||\mathbf{m}_2 - \mathbf{m}_1||_2}$$

Deriving $\mathbf{w}$ in the binary case

The direction $\mathbf{w}$ of the line is the one from $\mathbf{m}_1$ to $\mathbf{m}_2$.

This may result in a poor separation of classes.



Projections of classes are dispersed (high variance) along the direction of $\mathbf{m}_1 - \mathbf{m}_2$. This may result in a large overlap.

Deriving $\mathbf{w}$ in the binary case: refinement

- Choose directions s.t. classes projections show as little dispersion as possible
- Possible in the case that the amount of class dispersion changes wrt different directions, that is if the distribution of points in the class is elongated
- We wish then to maximize a function which:
  - is growing wrt the separation between the projected classes (for example, their mean points)
  - is decreasing wrt to the dispersion of the projections of points of each class

Deriving $\mathbf{w}$ in the binary case: refinement

- The within-class variance of the projection of class C_i $(i = 1, 2)$ is defined as
$$s_i^2 = \sum_{\mathbf{x} \in C_i} (\mathbf{w}^T \mathbf{x} - m_i)^2$$
The total within-class variance is defined as $s_1^2 + s_2^2$
- Given a direction $\mathbf{w}$, the Fisher criterion is the ratio between the (squared) class separation and the overall within-class variance, along that direction
$$J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}$$
- Indeed, $J(\mathbf{w})$ grows wrt class separation and decreases wrt within-class variance

Deriving $\mathbf{w}$ in the binary case: refinement

Let $\mathbf{S}_1, \mathbf{S}_2$ be the within-class covariance matrices, defined as

$$\mathbf{S}_i = \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T$$

Then,

$$\begin{aligned}
s_i^2 = \sum_{\mathbf{x} \in C_i} (\mathbf{w}^T \mathbf{x} - m_i)^2 &= \sum_{\mathbf{x} \in C_i} (\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_i)^2 \\
&= \sum_{\mathbf{x} \in C_i} (\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_i)(\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_i) \\
&= \sum_{\mathbf{x} \in C_i} (\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_i)(\mathbf{x}^T \mathbf{w} - \mathbf{m}_i^T \mathbf{w}) \\
&= \sum_{\mathbf{x} \in C_i} \left(\mathbf{w}^T (\mathbf{x} - \mathbf{m}_i)\right) \left((\mathbf{x} - \mathbf{m}_i)^T \mathbf{w}\right) \\
&= \sum_{\mathbf{x} \in C_i} \mathbf{w}^T (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \mathbf{w} \\
&= \mathbf{w}^T \left(\sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \right) \mathbf{w} = \mathbf{w}^T \mathbf{S}_i \mathbf{w}
\end{aligned}$$

Deriving $\mathbf{w}$ in the binary case: refinement

Let also $\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2$ be the total within-class covariance matrix and

$$\mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$$

be the between-class covariance matrix.

Then,

$$\begin{aligned}
J(\mathbf{w}) &= \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} \\
&= \frac{(\mathbf{w}^T\mathbf{m}_2 - \mathbf{w}^T\mathbf{m}_1)^2}{\mathbf{w}^T\mathbf{S}_1\mathbf{w} + \mathbf{w}^T\mathbf{S}_2\mathbf{w}} \\
&= \frac{(\mathbf{w}^T\mathbf{m}_2 - \mathbf{w}^T\mathbf{m}_1)(\mathbf{w}^T\mathbf{m}_2 - \mathbf{w}^T\mathbf{m}_1)}{\mathbf{w}^T\mathbf{S}_1\mathbf{w} + \mathbf{w}^T\mathbf{S}_2\mathbf{w}} \\
&= \frac{\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T\mathbf{w}}{\mathbf{w}^T\mathbf{S}_1\mathbf{w} + \mathbf{w}^T\mathbf{S}_2\mathbf{w}} \\
&= \frac{\mathbf{w}^T\mathbf{S}_B\mathbf{w}}{\mathbf{w}^T\mathbf{S}_W\mathbf{w}}
\end{aligned}$$

Deriving $\mathbf{w}$ in the binary case: refinement

As usual, $J(\mathbf{w})$ is maximized wrt $\mathbf{w}$ by setting its gradient to $\mathbf{0}$

$$\frac{\partial}{\partial \mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}} = 2 \frac{(\mathbf{w}^T \mathbf{S}_B \mathbf{w}) \mathbf{S}_W \mathbf{w} - (\mathbf{w}^T \mathbf{S}_W \mathbf{w}) \mathbf{S}_B \mathbf{w}}{(\mathbf{w}^T \mathbf{S}_W \mathbf{w})(\mathbf{w}^T \mathbf{S}_W \mathbf{w})^T}$$

which results into

$$(\mathbf{w}^T \mathbf{S}_B \mathbf{w}) \mathbf{S}_W \mathbf{w} = (\mathbf{w}^T \mathbf{S}_W \mathbf{w}) \mathbf{S}_B \mathbf{w}$$

Deriving $\mathbf{w}$ in the binary case: refinement

Observe that:

- $\mathbf{w}^T\mathbf{S}_B\mathbf{w}$ is a scalar, say c_B
- $\mathbf{w}^T\mathbf{S}_W\mathbf{w}$ is a scalar, say c_W
- $(\mathbf{m}_2 - \mathbf{m}_1)^T\mathbf{w}$ is a scalar, say c_m

Then, the condition $(\mathbf{w}^T\mathbf{S}_B\mathbf{w})\mathbf{S}_W\mathbf{w} = (\mathbf{w}^T\mathbf{S}_W\mathbf{w})\mathbf{S}_B\mathbf{w}$ can be written as

$$c_B\mathbf{S}_W\mathbf{w} = c_W\mathbf{S}_B\mathbf{w} = c_W(\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T\mathbf{w} = c_W(\mathbf{m}_2 - \mathbf{m}_1)c_m$$

which results into

$$\mathbf{w} = \frac{c_W c_m}{c_B}\mathbf{S}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

Since we are interested into the direction of $\mathbf{w}$, that is in any vector proportional to $\mathbf{w}$, we may consider the solution

$$\hat{\mathbf{w}} = \mathbf{S}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1) = (\mathbf{S}_1 + \mathbf{S}_2)^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

Deriving $\mathbf{w}$ in the binary case: choosing a threshold

Possible approach:

- model $p(y|C_i)$ as a gaussian: derive mean and variance by maximum likelihood

  $$m_i = \frac{1}{n_i}\sum_{\mathbf{x}\in C_i} w^T\mathbf{x} \qquad \sigma_i^2 = \frac{1}{n_i - 1}\sum_{\mathbf{x}\in C_i}(w^T\mathbf{x} - m_i)^2$$

  where n_i is the number of items in training set belonging to class C_i
- derive the class probabilities

  $$p(C_i|y) \propto p(y|C_i)p(C_i) = p(y|C_i)\frac{n_i}{n_1 + n_2} \propto n_i e^{-\frac{(y-m_i)^2}{2\sigma_i^2}}$$

- the threshold $\tilde{y}$ can be derived as the minimum y such that

  $$\frac{p(C_2|y)}{p(C_1|y)} = \frac{n_2}{n_1}\frac{p(y|C_2)}{p(y|C_1)} > 1$$

Let $K > 2$ and assume $D > K$, that is the number of features is greater than the number of classes.

Let also D', $1 < D' < D$, be the dimension of the projection space: then, D' linear transformations $y_k = \mathbf{w}_k^T \mathbf{x}$ ($k = 1, \ldots, D'$) are defined which project a D-dimensional point $\mathbf{x}$ into a D'-dimensional point $\mathbf{y} = (y_1, \ldots, y_{D'})^T$. In short, if $\mathbf{w}_i$ is the i-th column of $\mathbf{W}$,

$$\mathbf{y} = \mathbf{W}^T \mathbf{x}$$

To apply the same criterion of the binary case, we have to define within-class and between-class matrices, both in the D-dimensional and in the D'-dimensional spaces.

The generalization of the within-class covariance matrix is trivial:

$$\mathbf{S}_W = \sum_{i=1}^{K} \mathbf{S}_i = \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T$$

where

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in C_i} \mathbf{x}$$

For what concerns the between-class covariance, we first define the total covariance matrix of the training set

$$\mathbf{S}_T = \sum_{\mathbf{x}} (\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T = \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T$$

where $\mathbf{m}$ is the mean point of the whole training set

$$\mathbf{m} = \frac{1}{n} \sum_{\mathbf{x}} \mathbf{x} = \frac{1}{n} \sum_{i=1}^{K} n_i \mathbf{m}_i$$

This matrix can be decomposed as follows

$$\begin{aligned}
\mathbf{S}_T &= \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i + \mathbf{m}_i - \mathbf{m})(\mathbf{x} - \mathbf{m}_i + \mathbf{m}_i - \mathbf{m})^T \\
&= \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T + \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \\
&= \mathbf{S}_W + \sum_{i=1}^{K} n_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T
\end{aligned}$$

In the identity

$$\mathbf{S}_T = \mathbf{S}_W + \sum_{i=1}^{K} n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$$

we may identify the share of total covariance not caused by within-class covariance as between-class covariance, thus defining the between-class covariance matrix as

$$\mathbf{S}_B = \sum_{i=1}^{K} n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$$

$$\mathbf{s}_W = \sum_{i=1}^{K} \mathbf{s}_i = \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} (\mathbf{W}^T\mathbf{x} - \overline{\mathbf{m}}_i)(\mathbf{W}^T\mathbf{x} - \overline{\mathbf{m}}_i)^T$$

$$\mathbf{s}_B = \sum_{i=1}^{K} n_i (\overline{\mathbf{m}}_i - \overline{\mathbf{m}})(\overline{\mathbf{m}}_i - \overline{\mathbf{m}})^T$$

$$\overline{\mathbf{m}}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in C_i} \mathbf{W}^T\mathbf{x} \qquad \overline{\mathbf{m}} = \frac{1}{n} \sum_{\mathbf{x}} \mathbf{W}^T\mathbf{x}$$

It is also possible to prove that

$$\mathbf{s}_W = \mathbf{W}^T\mathbf{S}_W\mathbf{W} \quad \text{and} \quad \mathbf{s}_B = \mathbf{W}^T\mathbf{S}_B\mathbf{W}$$

LDA and multiclass classification

- Reminder: we need a matrix $\mathbf{W}$ that
  1. increases dispersion of classes (between-class covariance after projection)
  2. decreases the dispersion of points within classes (within-class covariance after projection)
- Different measures of dispersion can be introduced in this framework, such as
  1. the ratio between the determinants of $\mathbf{s}_B$ and $\mathbf{s}_W$

     $$J(\mathbf{W}) = \frac{|\mathbf{s}_B|}{|\mathbf{s}_W|} = |\mathbf{s}_W^{-1} s_B| = |(\mathbf{W}^T \mathbf{S}_W \mathbf{W})^{-1} \mathbf{W}^T \mathbf{S}_B \mathbf{W}|$$

     the determinant is the product of the eigenvalues (and, approximately, of the variances along the distribution axes in a gaussian model)
  2. the trace of the ``ratio'' between $\mathbf{s}_B$ and $\mathbf{s}_W$

     $$J(\mathbf{W}) = \mathrm{tr}(\mathbf{s}_W^{-1} s_B) = tr((\mathbf{W}^T \mathbf{S}_W \mathbf{W})^{-1} \mathbf{W}^T \mathbf{S}_B \mathbf{W})$$

     note that the trace is the sum of the eigenvalues

It is possible to prove that $\mathbf{W}$ is given by the eigenvectors of $\mathbf{S}_B^{-1} \mathbf{S}_W$ corresponding to the D' largest eigenvalues.