Machine Learning (521289S) Linear Discriminant Functions

M.Sc. Antti Isosalo

Physiological Signal Analysis Team
Center for Machine Vision and Signal Analysis (CMVS)
University of Oulu

Linear Discriminant Functions

- In MLE, for example, we assumed that the forms for the underlying probability densities were known, and used the training samples to perform parameter estimation.

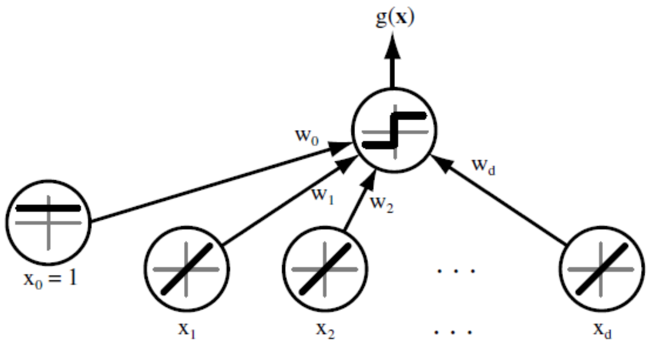
- ***Here we shall instead assume, that we know the proper forms for the discriminant functions***, and use the samples to estimate the values of parameters of the classifier.

- Determining the discriminant functions does not require knowledge of the forms of underlying probability distributions.

Linear Discriminant Functions Are Attractive in Their Simplicity

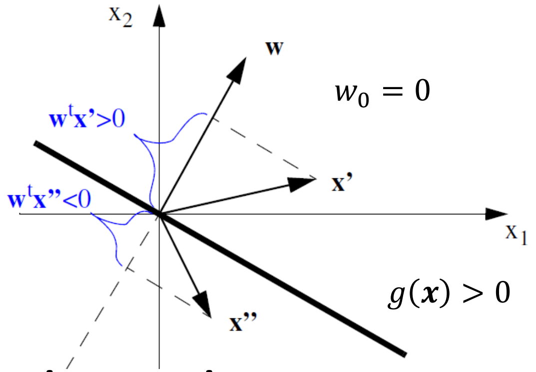
- As we have seen before, linear discriminant functions can be optimal if the underlying distributions are co-operative, such as Gaussians having equal covariance, as might be obtained through an intelligent ***choice of feature detectors***.
- Even when they are not optimal, we might be willing to sacrifice some performance in order to gain the advantage of their simplicity.
- Linear discriminant functions are ***relatively easy to compute*** and in the absence of information suggesting otherwise, linear classifiers are an attractive candidates for initial, trial classifiers.
- You will come across linear discriminant functions also when we look into ***neural networks*** later on this course.

Linear Discriminant Functions



- Discriminant function, that is linear combination of the components of *feature vector* $\boldsymbol{x}$, can be written as
$$g(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0,$$
where $\boldsymbol{w}$ is a *weight vector* and w_0 is a *bias constant*.
- In general, each class is assigned its own discriminant function and a *pattern* is classified into a class based on discriminant function that gives the highest value at point $\boldsymbol{x}$.
- In a two category case the discriminant functions can be combined to one
$$g_1(\boldsymbol{x}) > g_2(\boldsymbol{x}),$$
$$g_1(\boldsymbol{x}) - g_2(\boldsymbol{x}) > 0,$$
$$g(\boldsymbol{x}) > 0$$
- Then we can say: Decide ω_1 if $g(\boldsymbol{x}) > 0$ and decide ω_2 if $g(\boldsymbol{x}) < 0$.
  - If $g(\boldsymbol{x}) = 0$, $\boldsymbol{x}$ can usually be assigned to either class, but in Chapter 5 the course book chooses to leave the assignment undefined.
- In other words, decide ω_1 if the inner product $\boldsymbol{w}^T\boldsymbol{x} > -w_0$, otherwise ω_2.

Linear Discriminant Functions (cont.)



- To define a decision surface (hyperplane when $g(\boldsymbol{x})$ is linear) separating points assigned to ω_1 from points classified to ω_2, we write

$$g(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0 = [w_1 \quad w_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + w_0 \equiv 0.$$

- If $w_0 = 0$, then the hyperplane would be positioned to *origin,* and $\boldsymbol{w}^T\boldsymbol{x} \equiv 0$, $\boldsymbol{w} \cdot \boldsymbol{x} = 0$, $\boldsymbol{w} \perp \boldsymbol{x}$, saying $\boldsymbol{w}$ is normal (perpendicular) to any vector lying in the hyperplane.
- If $w_0 > 0$, then origin is found from the positive side of the decision surface, and if $w_0 < 0$, then origin is found on the negative side of the decision surface.
- If we were to have, e.g., $w_0 > 0$, then $g(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0$, and on the decision surface $\boxed{\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\xi} \equiv 0}$ for some $\boldsymbol{\xi}$, $\boxed{-\boldsymbol{w}^T\boldsymbol{\xi} = w_0}$, $\boldsymbol{w} \perp \big(\boldsymbol{x} - (-\boldsymbol{\xi})\big)$, resulting ***a shift of origin*** and allowing more flexible positioning of the hyperplane between classes.

Linear Discriminant Functions: Augmented Vector Representation

- So, a linear discriminant function divides the feature space by a decision surface.
- The orientation of the surface is determined by the normal vector $\boldsymbol{w}$ (weight vector), and the location of the surface is determined by the bias w_0.
- We can also write

$$g(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0$$

$$g(\boldsymbol{x}) = w_0 \cdot \underbrace{1}_{x_0} + \begin{bmatrix} w_1 & w_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$g(\boldsymbol{x}) = \underbrace{\begin{bmatrix} w_0 & w_1 & w_2 \end{bmatrix}}_{\boldsymbol{a}} \underbrace{\begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{y}}$$

$$g(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{y},$$

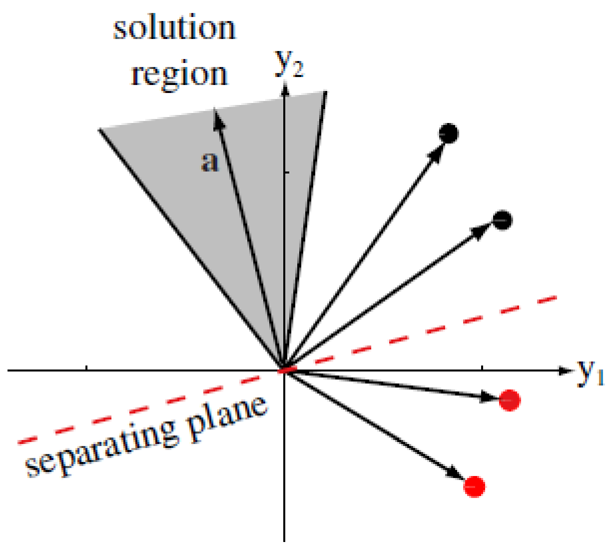
where $\boldsymbol{a}$ is an ***augmented weight vector*** and $\boldsymbol{y}$ is an ***augmented feature vector***.

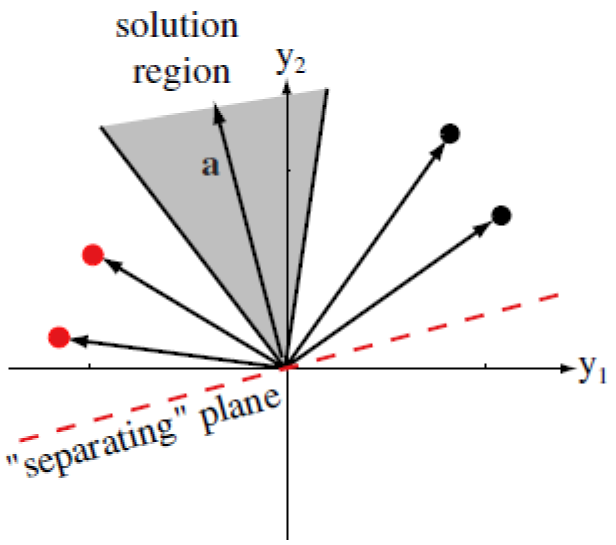
  - You can see that the decision surface now lies in origin in $\boldsymbol{y}$ space.

Linear Discriminant Functions: Linear Separation and Solution Vector

- Now we want to use *samples* in our feature vector $\boldsymbol{y}$ to determine the weight vector $\boldsymbol{a}$ in a linear discriminant function $g(\boldsymbol{x}) = \boldsymbol{a}^{\boldsymbol{T}}\boldsymbol{y}$.
- If we are able to find vector $\boldsymbol{a}$ for a discriminant function which classifies correctly all samples, we say that classes ω_1 and ω_2 are ***linearly separable***.
- Let's decide that we achieve a correct classification to ω_1 if $\boldsymbol{a}^{\boldsymbol{T}}\boldsymbol{y}_{\boldsymbol{i}} > 0$ and correctly to ω_2 if $\boldsymbol{a}^{\boldsymbol{T}}\boldsymbol{y}_{\boldsymbol{i}} < 0$.
- In order to find *solution vector* $\boldsymbol{a}$ for which $\boldsymbol{a}^{\boldsymbol{T}}\boldsymbol{y}_{\boldsymbol{i}} > 0$ for all samples we can replace all feature vectors $\boldsymbol{y}_{\boldsymbol{i}}$ from class ω_2 with their negation $-\boldsymbol{y}_{\boldsymbol{i}}$.
  - All vectors point now to the positive side of the decision boundary.

Linear Discriminant Functions: Linear Separation and Solution Vector (cont.)



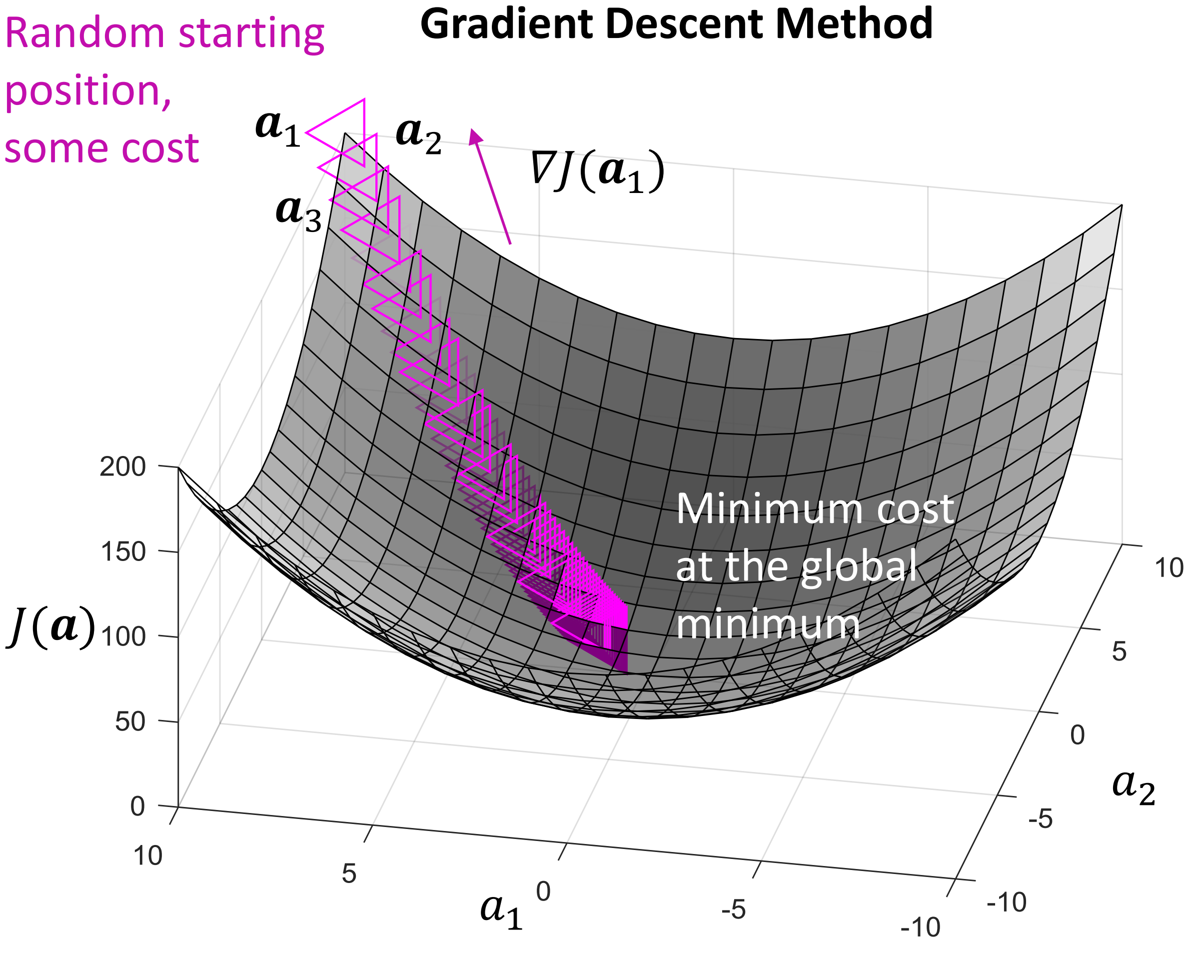


Gradient Search Methods for Finding the Weight Vector

- There are different ways to find a weight vector $\boldsymbol{a}$ such that $\boldsymbol{a}^T \boldsymbol{y_i} > 0$ for all the samples (assuming one exists).
- So, we find a solution for linear systems of inequalities $\boldsymbol{a}^T \boldsymbol{y_i} > 0$ by using a certain criterion function $J(\boldsymbol{a})$ which minimizes when $\boldsymbol{a}$ is our desired solution vector.
- So, *we move to solve a minimization problem* for scalar valued function, and that can be accomplished using ***gradient search*** (e.g. gradient decent procedure).
  - Gradient descent is an optimization algorithm used to find the values of parameters (coefficients) of a function that minimizes a criterion function (cost).

Gradient Decent Procedure

- In basic gradient descent method, we start with some arbitrarily chosen weight vector $\boldsymbol{a_1}$ and compute the gradient vector $\nabla J(\boldsymbol{a}_1)$.
- The next value $\boldsymbol{a}_2$ is obtained by moving some distance from $\boldsymbol{a}_1$ in the direction of steepest descent, i.e., along the negative of the gradient.
- In general, $\boldsymbol{a}_{k+1}$ is obtained from $\boldsymbol{a}_k$ by the equation $\boldsymbol{a}_{k+1} = \boldsymbol{a}_k - \eta(k)\nabla J(\boldsymbol{a}_k)$, where $\eta(k)$ is a ***learning rate*** that sets the step size.
- We hope that our sequence of weight vectors will converge to a solution minimizing the criterion function $J(\boldsymbol{a})$.
- If $\eta(k)$ is too small, convergence is needlessly slow.
- If $\eta(k)$ is too large, the correction process will overshoot and can even diverge.



Here the learning rate was fixed to 0.02