CS 4900/5900: Machine Learning

Fisher's Linear Discriminant

Razvan C. Bunescu

School of Electrical Engineering and Computer Science

bunescu@ohio.edu

Supervised Learning

- **Task** = learn an (unkown) function $t : X \rightarrow T$ that maps input instances $\mathbf{x} \in X$ to output targets $t(\mathbf{x}) \in T$:
  - **Classification**:
    - The output $t(\mathbf{x}) \in T$ is one of a finite set of discrete categories.
  - **Regression**:
    - The output $t(\mathbf{x}) \in T$ is continuous, or has a continuous component.

- Target function $t(\mathbf{x})$ is known (only) through (noisy) set of training examples:

  $(\mathbf{x}_1,t_1), (\mathbf{x}_2,t_2), \ldots (\mathbf{x}_n,t_n)$

Three Parametric Approaches to Classification

1) Discriminant Functions: construct $f : X \rightarrow T$ that directly assigns a vector $\mathbf{x}$ to a specific class C_k.
   - Inference and decision combined into a single learning problem.
   - *Linear Discriminant*: the decision surface is a hyperplane in X:
     - Fisher 's Linear Discriminant
     - Perceptron
     - Support Vector Machines

Three Parametric Approaches to Classification

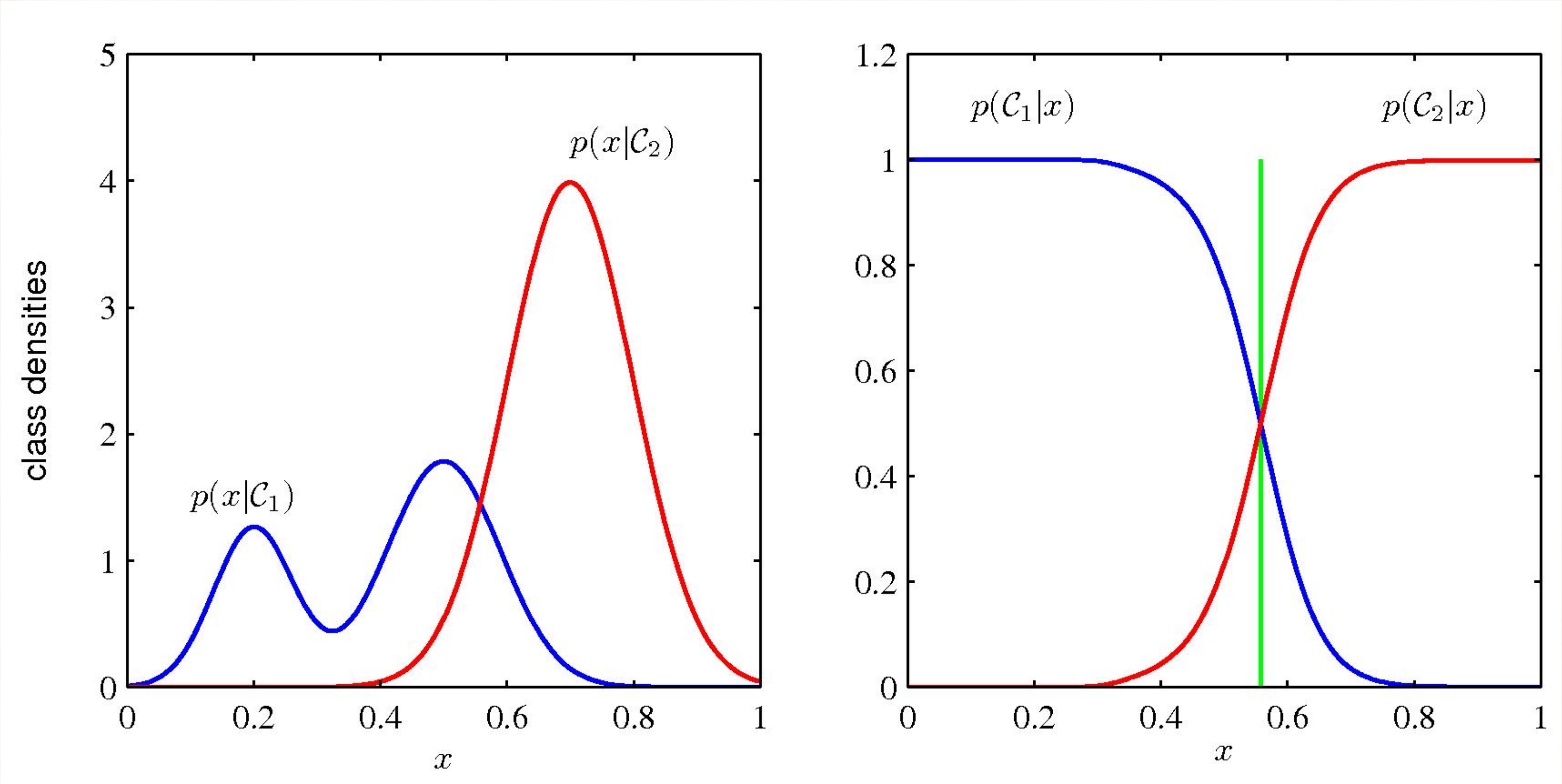
2) Probabilistic Discriminative Models: directly model the posterior class probabilities $p(C_k \mid \mathbf{x})$.
   - Inference and decision are separate.
   - Less data needed to estimate $p(C_k \mid \mathbf{x})$ than $p(\mathbf{x} \mid C_k)$.
   - Can accommodate many overlapping features.
     - Logistic Regression
     - Conditional Random Fields

Three Parametric Approaches to Classification

3) Probabilistic Generative Models:

- Model class-conditional $p(\mathbf{x} \mid C_k)$ as well as the priors $p(C_k)$, then use Bayes's theorem to find $p(C_k \mid \mathbf{x})$.
  - or model $p(\mathbf{x}, C_k)$ directly, then marginalize to obtain the posterior probabilities $p(C_k \mid \mathbf{x})$.
- Inference and decision are separate.
- Can use $p(\mathbf{x})$ for *outlier* or *novelty detection*.
- Need to model dependencies between features.
  - Naïve Bayes.
  - Hidden Markov Models.

Generative vs. Discriminative



Left-hand mode has no effect on posterior class probabilities.

Linear Discriminant Functions: Two classes ($K = 2$)

- Use a linear function of the input vector:

$$y(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + w_0$$

weight vector

bias = – threshold

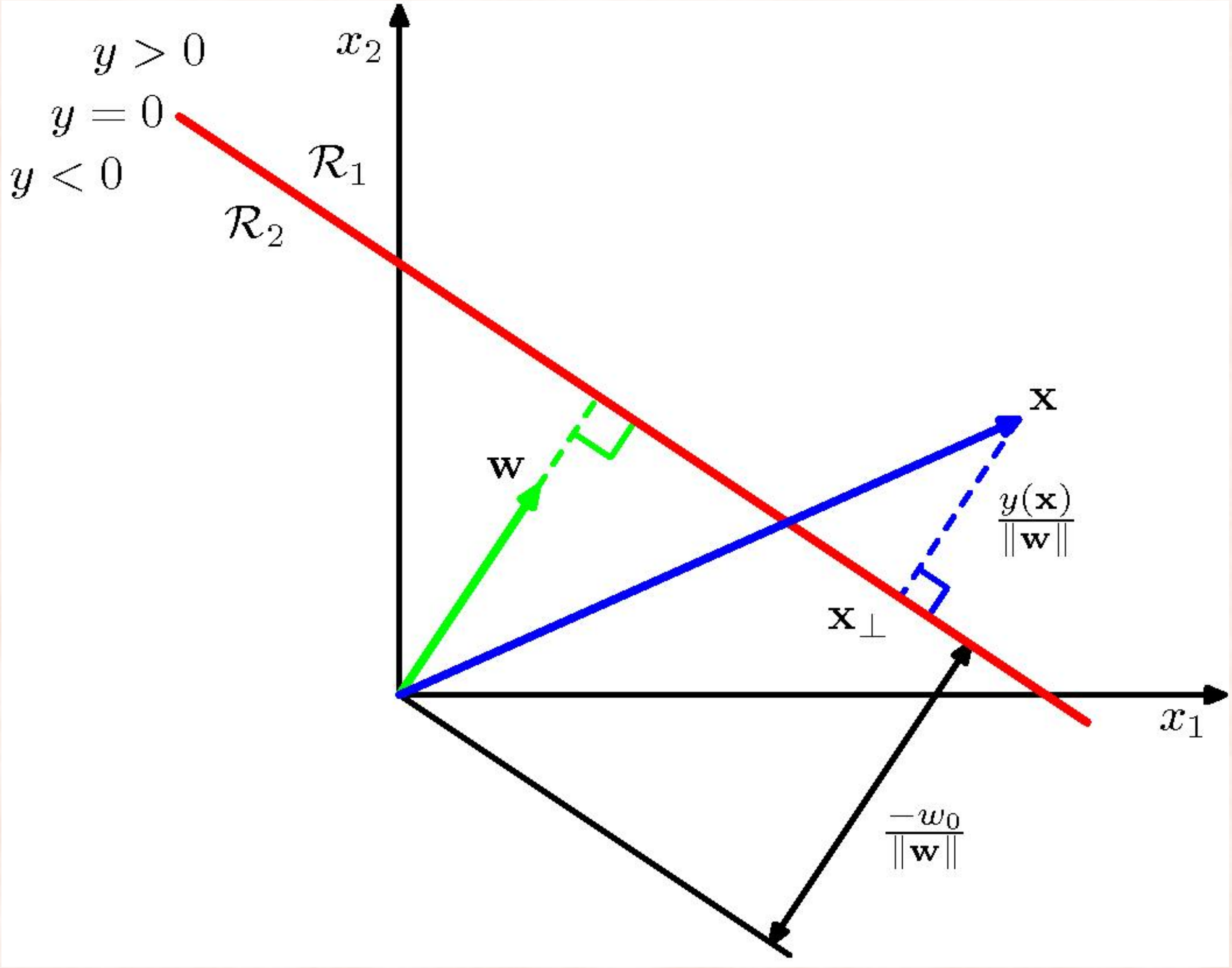
- Decision:

$\mathbf{x} \in C_1$ if $y(\mathbf{x}) \geq 0$, otherwise $\mathbf{x} \in C_2$.

$\Rightarrow$ decision boundary is hyperplane $y(\mathbf{x}) = 0$.

- Properties:
  - $\mathbf{w}$ is orthogonal to vectors lying within the decision surface.
  - w_0 controls the location of the decision hyperplane.

Linear Discriminant Functions: Two Classes ($K = 2$)



Linear Discriminant Functions: Multiple Classes (K > 2)

1) Train K or K–1 *one-versus-the-rest* classifiers.
2) Train K(K–1)/2 *one-versus-one* classifiers.

3) Train K linear functions:

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \varphi(\mathbf{x}) + w_{k0}$$

- Decision:

$\mathbf{x} \in C_k$ if $y_k(\mathbf{x}) > y_j(\mathbf{x})$, for all $j \neq k$.

$\Rightarrow$ decision boundary between classes C_k and C_j is hyperplane defined by $y_k(\mathbf{x}) = y_j(\mathbf{x})$ i.e. $(\mathbf{w}_k - \mathbf{w}_j)^T \varphi(\mathbf{x}) + (w_{k0} - w_{j0}) = 0$

$\Rightarrow$ same geometrical properties as in binary case.

Linear Discriminant Functions: Multiple Classes (K > 2)

4) More general ranking approach:

$$y(\mathbf{x}) = \arg\max_{t \in T} \mathbf{w}^T \varphi(\mathbf{x}, t) \quad \text{where} \quad T = \{c_1, c_2 ..., c_K\}$$

- It subsumes the approach with K separate linear functions.

- Useful when T is very large (e.g. exponential in the size of input **x**), assuming inference can be done efficiently.

Linear Discriminant Functions: Two Classes (K = 2)

- What algorithms can be used to learn $y(\mathbf{x}) = \mathbf{w}^T\varphi(\mathbf{x}) + w_0$?

  Assume a training dataset of $N = N_1 + N_2$ examples in C_1 and C_2.

  – Fisher's Linear Discriminant
  – Perceptron:
    - Voted/Averaged Perceptron
    - Kernel Perceptron
  – Support Vector Machines:
    - Linear
    - Kernel

Fisher's Linear Discriminant

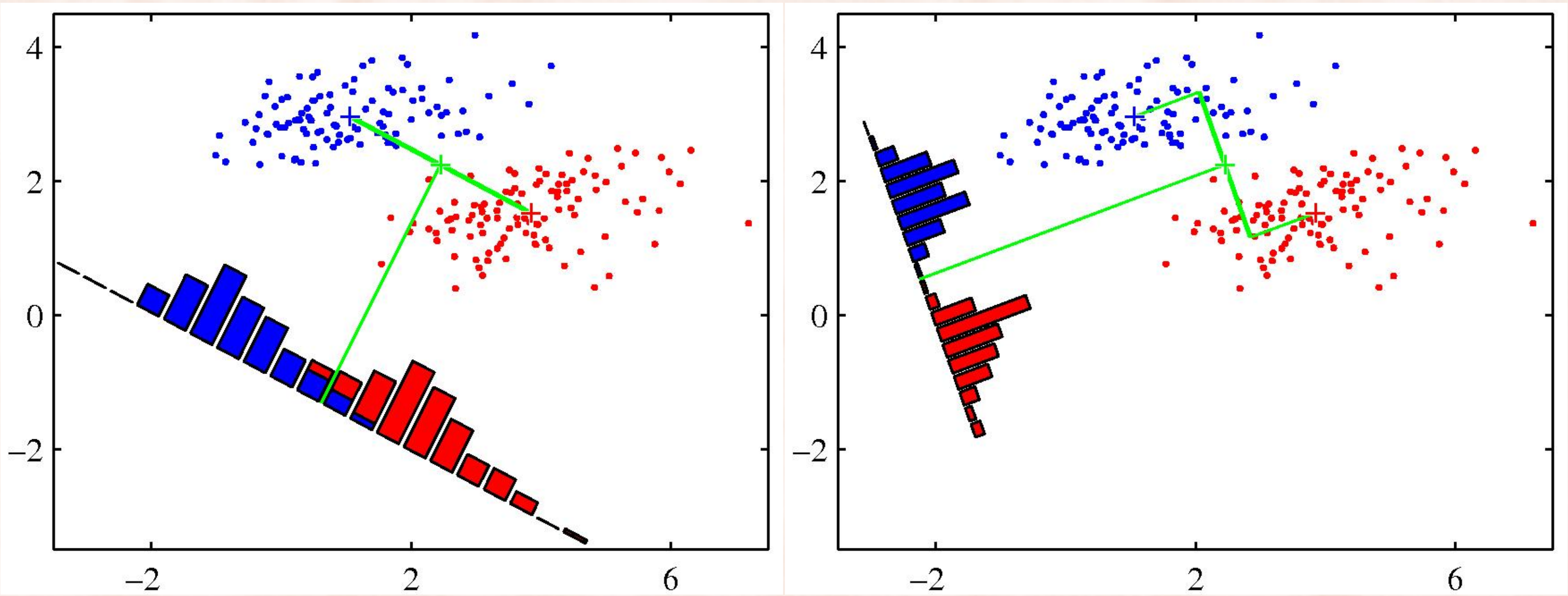
- Discriminant function $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ can be interpreted as follows:
  1. Project D-dimensional $\mathbf{x}$ down to one dimension $\Rightarrow \mathbf{w}^T\mathbf{x}$
  2. Use a threshold $-w_0$ to classify $\mathbf{x} \Rightarrow$

     $\mathbf{x} \in C_1$, if $\mathbf{w}^T\mathbf{x} \geq -w_0$

     $\mathbf{x} \in C_2$, otherwise.

- Fisher's idea:
  - Maximize the **between-class separation** of projected dataset.
  - Minimize the **within-class variance** of projected dataset.

Fisher’s Linear Discriminant



Line joining the class means vs. Line inferred with Fisher’s criterion.

Fisher's Linear Discriminant

1) Measure of the separation between the classes is the *between class variance*:

$$\left.\begin{aligned}\mathbf{m}_1 &= \frac{1}{N_1}\sum_{n\in C_1}\mathbf{x}_n \\ \mathbf{m}_2 &= \frac{1}{N_2}\sum_{n\in C_2}\mathbf{x}_n\end{aligned}\right] \Rightarrow m_2 - m_1 = \mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1) \Rightarrow \boxed{(m_2 - m_1)^2}$$

Fisher's Linear Discriminant

2) Measure of the *within-class variance*:

$$\left.\begin{aligned} s_1^2 &= \sum_{n \in C_1} (\mathbf{w}^T \mathbf{x}_n - m_1)^2 \\ s_2^2 &= \sum_{n \in C_2} (\mathbf{w}^T \mathbf{x}_n - m_2)^2 \end{aligned}\right] \Rightarrow \boxed{s_1^2 + s_2^2}$$

Fisher's Linear Discriminant

- Maximize the between-class separation and minimize the within-class variance $\Rightarrow$ Fisher's criterion:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} J(\mathbf{w}) \quad \text{, where } J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}$$

- The objective function can be rewritten as:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_\mathrm{B} \mathbf{w}}{\mathbf{w}^T \mathbf{S}_\mathrm{W} \mathbf{w}}$$

where

$$\mathbf{S}_\mathrm{B} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^\mathrm{T}$$

$$\mathbf{S}_\mathrm{W} = \sum_{n \in C_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^\mathrm{T} + \sum_{n \in C_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^\mathrm{T}$$

Fisher's Linear Discriminant

- Optimization formulation:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} J(\mathbf{w}) = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}$$

- Solution:

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = 0 \Rightarrow (\mathbf{w}^T S_W \mathbf{w}) S_B \mathbf{w} = (\mathbf{w}^T S_B \mathbf{w}) S_W \mathbf{w}$$

$$\Rightarrow S_B \mathbf{w} = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}} S_W \mathbf{w} \quad \Rightarrow \boxed{S_B \mathbf{w} = \lambda\, S_W \mathbf{w}}$$

generalized eigenvalue problem

- If S_W is nonsingular:

$$\Rightarrow \boxed{S_W^{-1} S_B \mathbf{w} = \lambda \mathbf{w}}$$

conventional eigenvalue problem

Fisher's Linear Discriminant

- No need to solve the eigenvalue problem:

  $\mathbf{S}_B\mathbf{w} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T\mathbf{w}$ is a vector in the direction $(\mathbf{m}_2 - \mathbf{m}_1)$

- The norm of **w** is immaterial, only its direction is important.

  $\Rightarrow$ can take

  $$\boxed{\mathbf{w} = \mathrm{S}_{\mathrm{W}}^{-1}(\mathbf{m}_2 - \mathbf{m}_1)}$$

- How to find w_0:
  - Assume $p(\mathbf{w}^T\mathbf{x}|C_1)$ and $p(\mathbf{w}^T\mathbf{x}|C_2)$ are Gaussians.
  - Estimate means and variances using maximum likelihood.
  - Use decision theory to find $\mathbf{w}_0$ i.e. $p(-\mathbf{w}_0|C_1) = p(-\mathbf{w}_0|C_2)$

Supplementary Reading

- PRML Section 1.4 (The Curse of Dimensionality).
- PRML Section 1.5 (Decision Theory).
- PRML Section 4 (Linear Models for Classification):
  - 4.1.1 to 4.1.4.