Supervised Learning

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

- Supervision = set of training examples:
  \[(x_1, t_1), (x_2, t_2), \ldots, (x_n, t_n)\]
Three Parametric Approaches to Classification

1) **Discriminant Functions**: construct $f : X \rightarrow T$ that directly assigns a vector $x$ 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 $p(C_k \mid x)$.
   - Inference and decision are separate.
   - Less data needed to estimate $p(C_k \mid x)$ than $p(x \mid C_k)$.
   - Can accommodate many overlapping features.
     - Logistic Regression
     - Conditional Random Fields
Three Parametric Approaches to Classification

3) Probabilistic Generative Models:
   - Model class-conditional $p(x \mid C_k)$ as well as the priors $p(C_k)$, then use Bayes’s theorem to find $p(C_k \mid x)$.
     • or model $p(x, C_k)$ directly, then marginalize to obtain the posterior probabilities $p(C_k \mid x)$.
   - Inference and decision are separate.
   - Can use $p(x)$ 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.

Lecture 03
Linear Discriminant Functions: Two classes \((K = 2)\)

- Use a linear function of the input vector:
  \[
y(x) = w^T \varphi(x) + w_0
  \]
  - **weight vector**
  - **bias = - threshold**

- Decision:
  - \(x \in C_1\) if \(y(x) \geq 0\), otherwise \(x \in C_2\).
  - \(\Rightarrow\) decision boundary is hyperplane \(y(x) = 0\).

- Properties:
  - \(w\) is orthogonal to vectors lying within the decision surface.
  - \(w_0\) controls the location of the decision hyperplane.
Linear Discriminant Functions: Two Classes ($K = 2$)
Feature Scaling
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:
   \[ y_k(x) = w_k^T \varphi(x) + w_{k0} \]

- **Decision:**
  \[ x \in C_k \text{ if } y_k(x) > y_j(x), \text{ for all } j \neq k. \]
  \[ \Rightarrow \text{decision boundary between classes } C_k \text{ and } C_j \text{ is hyperplane defined by } y_k(x) = y_j(x) \text{ i.e. } (w_k - w_j)^T \varphi(x) + (w_{k0} - w_{j0}) = 0 \]
  \[ \Rightarrow \text{same geometrical properties as in binary case.} \]
Linear Discriminant Functions: Multiple Classes ($K > 2$)

4) More general ranking approach:

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

- 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 $y(x) = w^T \varphi(x) + w_0$?
  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 $y(x) = w^T x + w_0$ can be interpreted as follows:
  1. Project D-dimensional $x$ down to one dimension $\Rightarrow w^T x$
  2. Use a threshold $-w_0$ to classify $x$ $\Rightarrow$
     - $x \in C_1$, if $w^T x \geq -w_0$
     - $x \in C_2$, 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.
1) Measure of the separation between the classes is the *between class variance*:

\[
\begin{align*}
\mathbf{m}_1 &= \frac{1}{N_1} \sum_{n \in C_1} \mathbf{x}_n \\
\mathbf{m}_2 &= \frac{1}{N_2} \sum_{n \in C_2} \mathbf{x}_n \\
\end{align*}
\]

\[
\mathbf{m}_2 - \mathbf{m}_1 = \mathbf{w}^T (\mathbf{m}_2 - \mathbf{m}_1) \quad \Rightarrow \quad (\mathbf{m}_2 - \mathbf{m}_1)^2
\]

Fisher’s Linear Discriminant
Fisher’s Linear Discriminant

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

\[
s_1^2 = \sum_{n \in C_1} (w^T x_n - m_1)^2
\]

\[
s_2^2 = \sum_{n \in C_2} (w^T x_n - m_2)^2
\]

\[s_1^2 + s_2^2\]
Fisher’s Linear Discriminant

- Maximize the between-class separation and minimize the within-class variance \( \Rightarrow \) Fisher’s criterion:
  \[
  w^* = \arg \max_w J(w), \text{ where } J(w) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}
  \]

- The objective function can be rewritten as:
  \[
  J(w) = \frac{w^T S_B w}{w^T S_W w}
  \]

where
\[
S_B = (m_2 - m_1)(m_2 - m_1)^T
\]
\[
S_W = \sum_{n \in C_1} (x_n - m_1)(x_n - m_1)^T + \sum_{n \in C_2} (x_n - m_2)(x_n - m_2)^T
\]
Fisher’s Linear Discriminant

• Optimization formulation:

\[ w^* = \arg \max_w J(w) = \arg \max_w \frac{w^T S_B w}{w^T S_W w} \]

• Solution:

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

\[ \Rightarrow S_B w = \frac{w^T S_B w}{w^T S_W w} S_W w \Rightarrow S_B w = \lambda S_W w \]

• If \( S_W \) is nonsingular:

\[ \Rightarrow S_W^{-1} S_B w = \lambda w \]

Lecture 03
Fisher’s Linear Discriminant

• No need to solve the eigenvalue problem:
  \[ S_B w = (m_2 - m_1)(m_2 - m_1)^T w \] is a vector in the direction \( m_2 - m_1 \)

• The norm of \( w \) is immaterial, only its direction is important.
  \[ \Rightarrow \text{can take} \quad w = S_W^{-1}(m_2 - m_1) \]

• How to find \( w_0 \):
  – Assume \( p(w^T x | C_1) \) and \( p(w^T x | C_2) \) are Gaussians.
  – Estimate means and variances using maximum likelihood.
  – Use decision theory to find \( w_0 \) i.e. \( p(-w_0 | C_1) = p(-w_0 | C_2) \)
Reading Assignment

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