Sound Classification Overview

EE3662: Digital Signal Processing Lab
#Lecture11 – Dec. 13, 2021

Prof. Chi-Chun Lee, Yi-Wen Liu
TAs: 邱信豪、許暐彤、陳舫慶、陳靖杰

BIIC

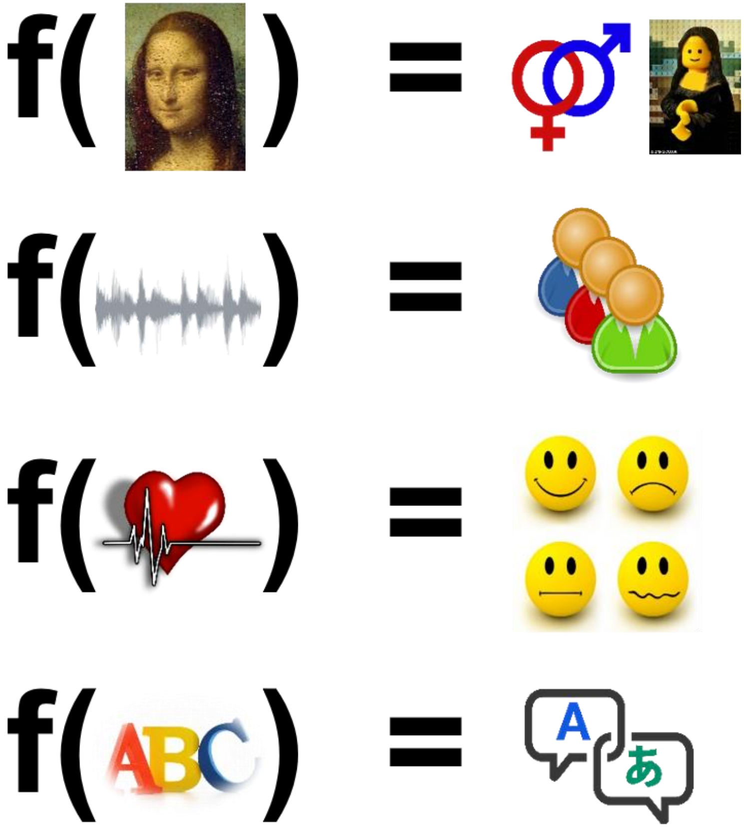
Machine Learning Concept

$$f : X \rightarrow Y$$

◆Mapping from data to label

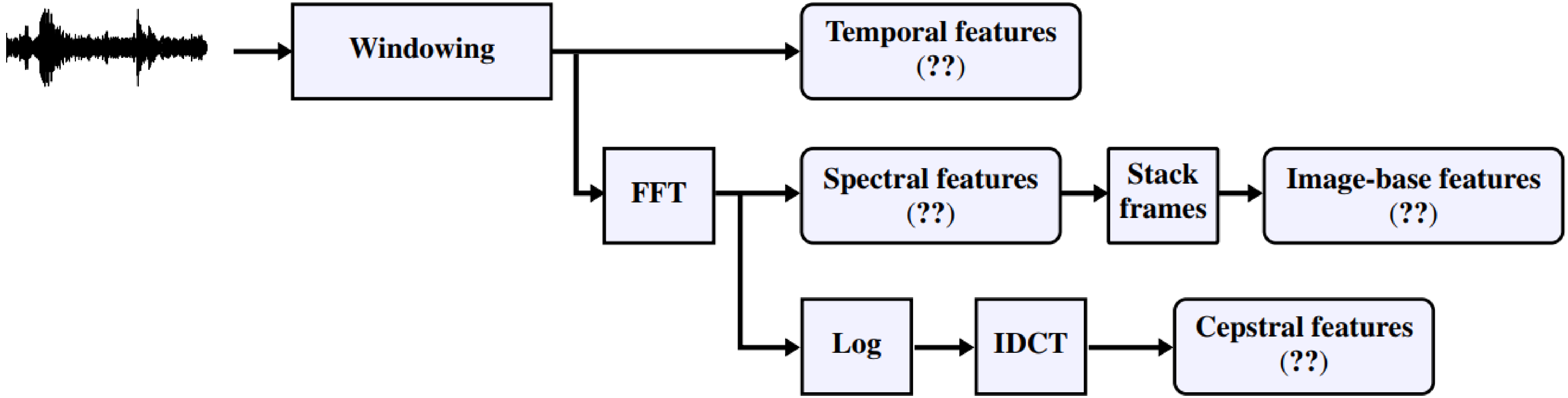
◆Input domain *X* : word sequence, audio, video, physiological signal

◆Output domain *Y* : label, sequence tags, probability



Features

Taxonomy of Acoustic Features



Temporal Features

- Short time energy
  - of frame n: $E_n = \sum_{m=n-N+1}^{n} x^2(m)$
    - windowed squared for x[n]
    - window: rectangular window

Indicator for silence detection

General short-time energy equation: $\sum_{m=-\infty}^{\infty}[x[m]w[n-m]]^2$

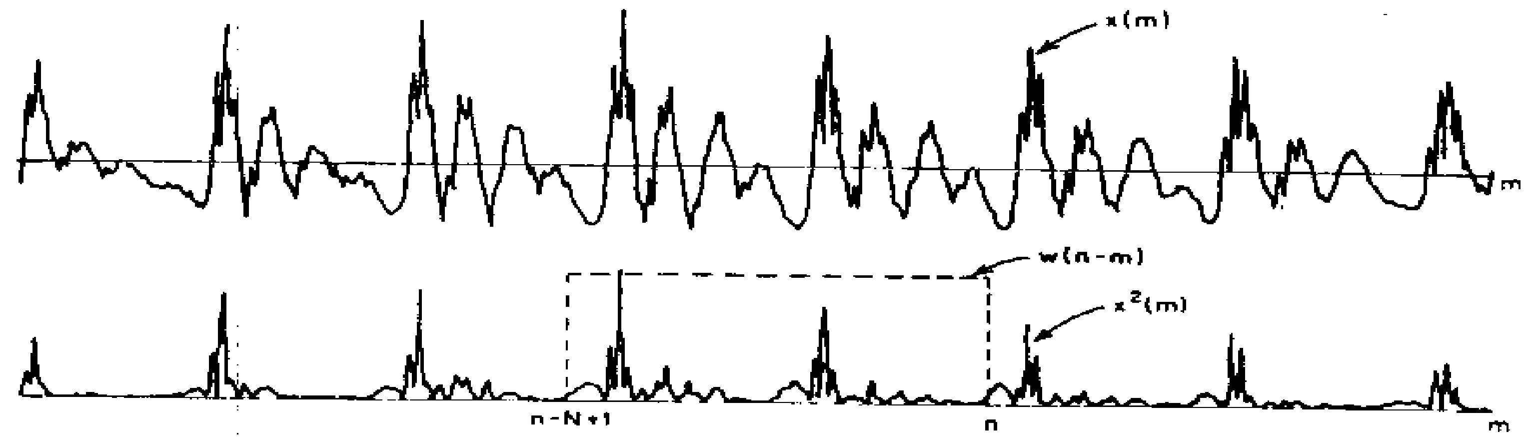
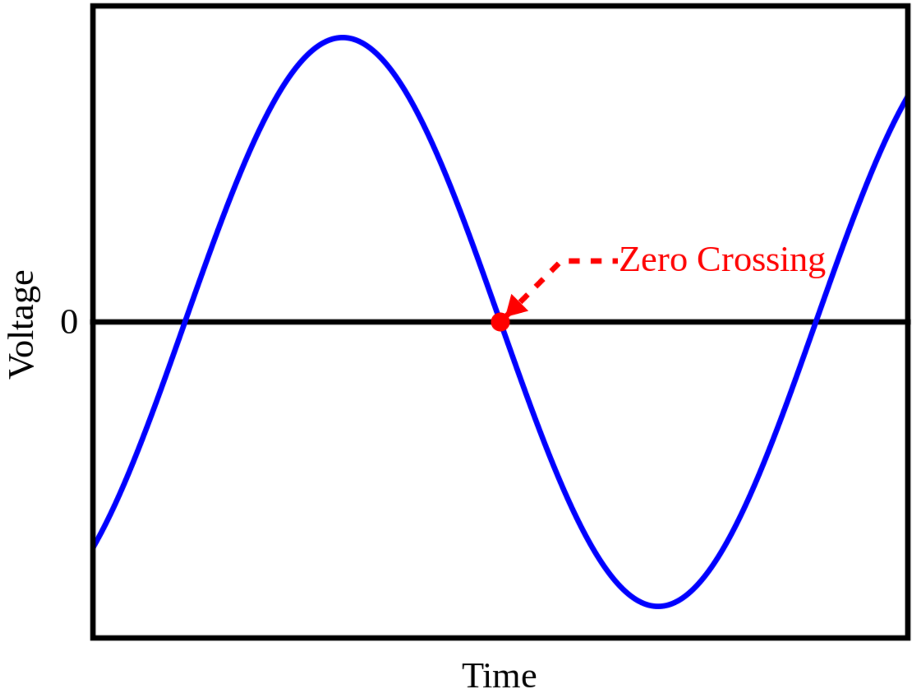


Fig. 4.2 Illustration of the computation of short-time energy.

Temporal Features

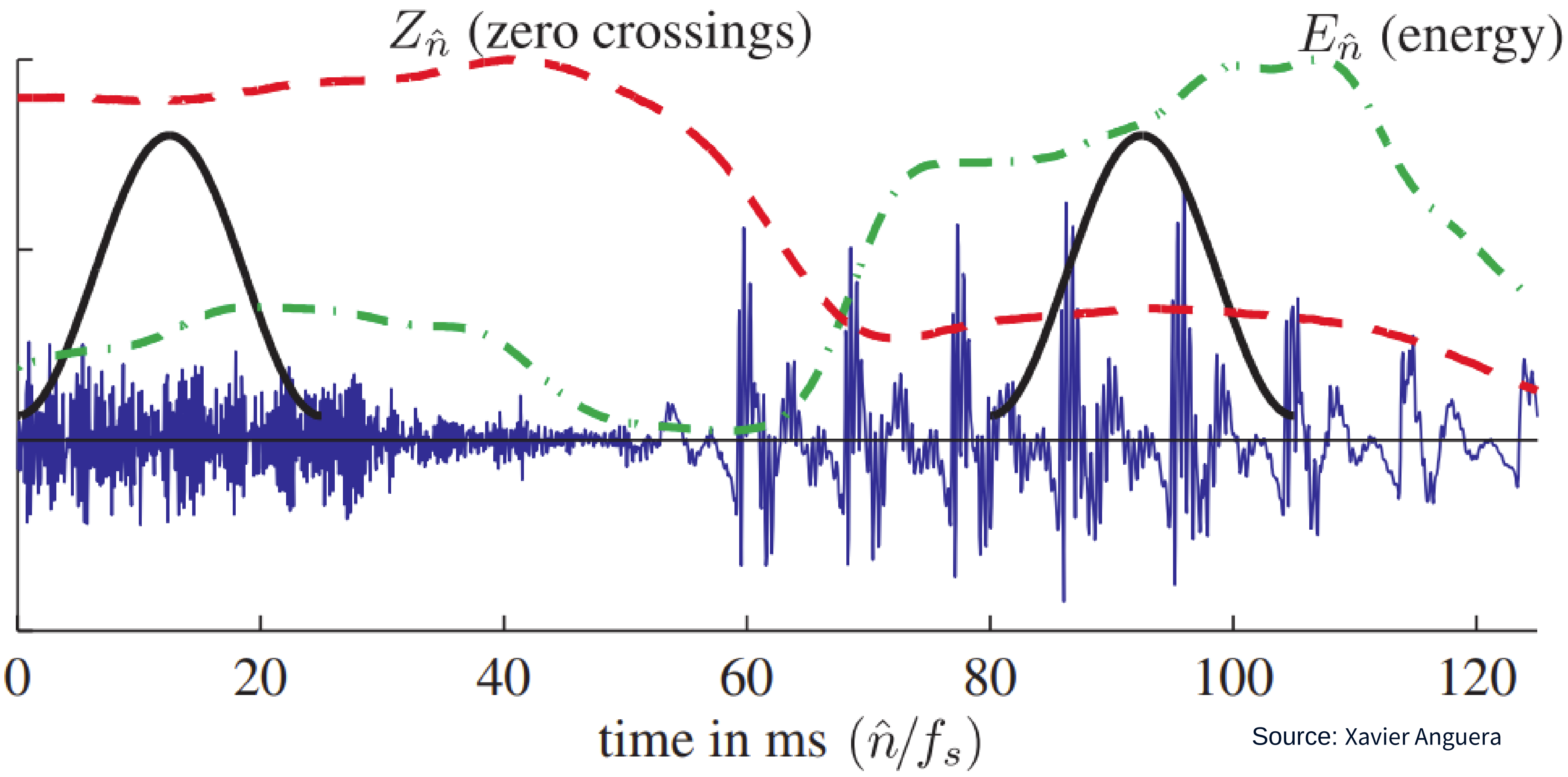
- Short time zero crossing
  - The subsequent samples have different signs
  - Measures how rapidly signal changes
  - Captures frequency content

$$Z_n = \sum_{m=-\infty}^{\infty} \left| \frac{sign[x(m)] - sign[x(m-1)]}{2} \right| w(n-m)$$



Unvoiced region:
lower energy
higher zero-crossing rate

Voiced region:
higher energy
lower zero-crossing rate

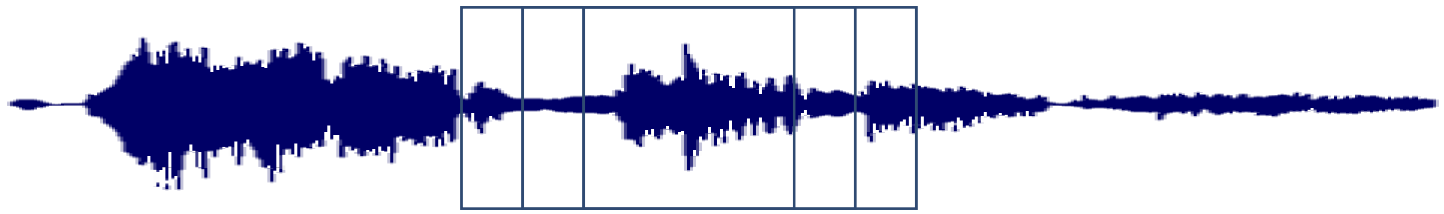


Source: Xavier Anguera

Temporal Features

- Short time autocorrelation
  - $R_n(k) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)x(m+k)w(n-m-k)$
    - How similar $x(m)$ is to $x(m+k)$
    - k is the lag parameter

  - $R_n(k)$ for voiced speech: periodic (not for unvoiced)
  - $R_n(k)$ peaks occur at lag (k) intervals approximately equal to pitch period



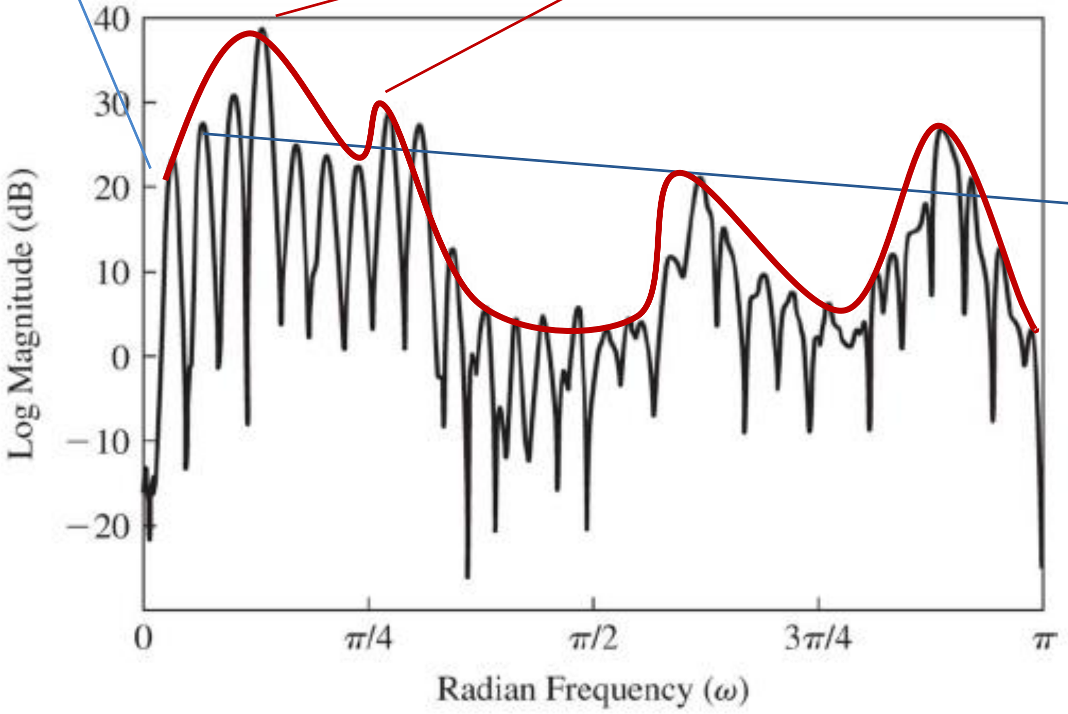
Spectral Features

- ◆Spectral descriptors: slope, flux, roll-off …
- ◆Formants: bandwidth, relative energy …
- ◆ Harmonics: relative difference/ratio of energy …

Spectral Features

Fundamental frequency (pitch): source

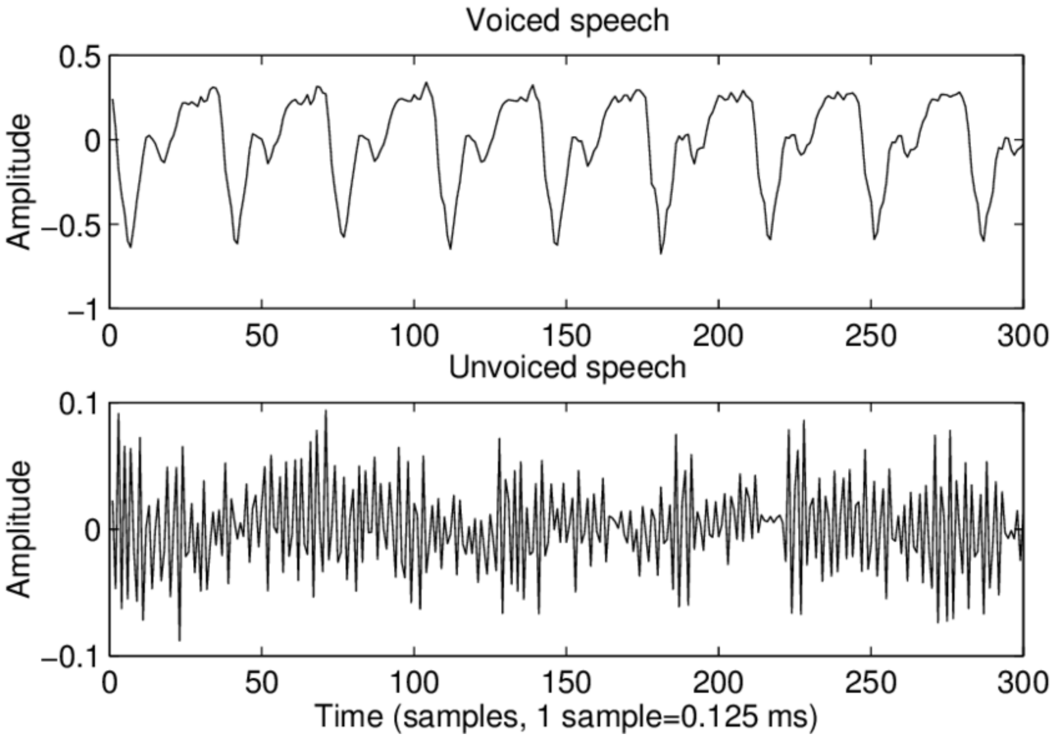
Formants

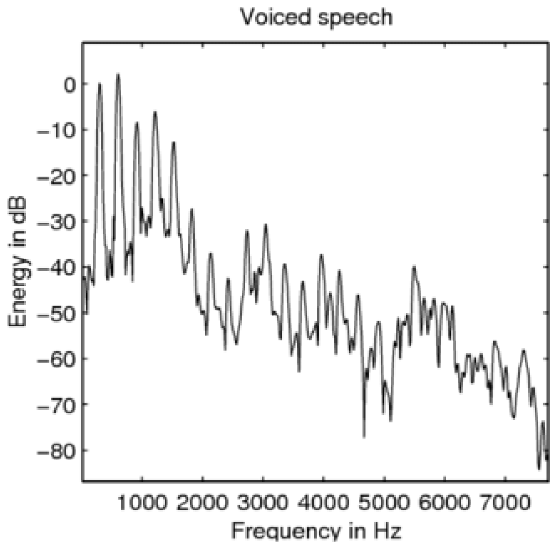


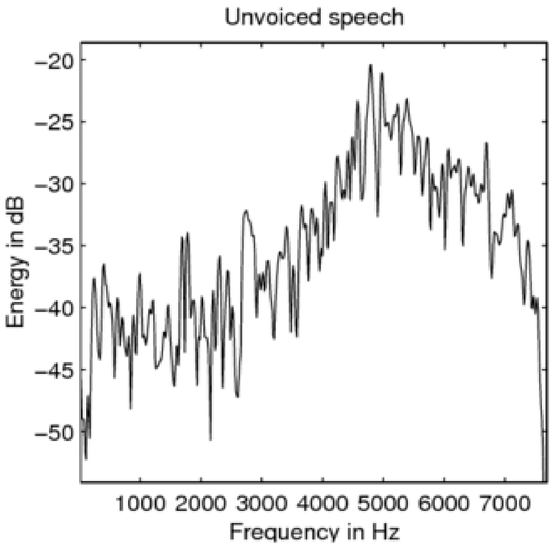
Harmonics: multiples of f0

Spectral Features

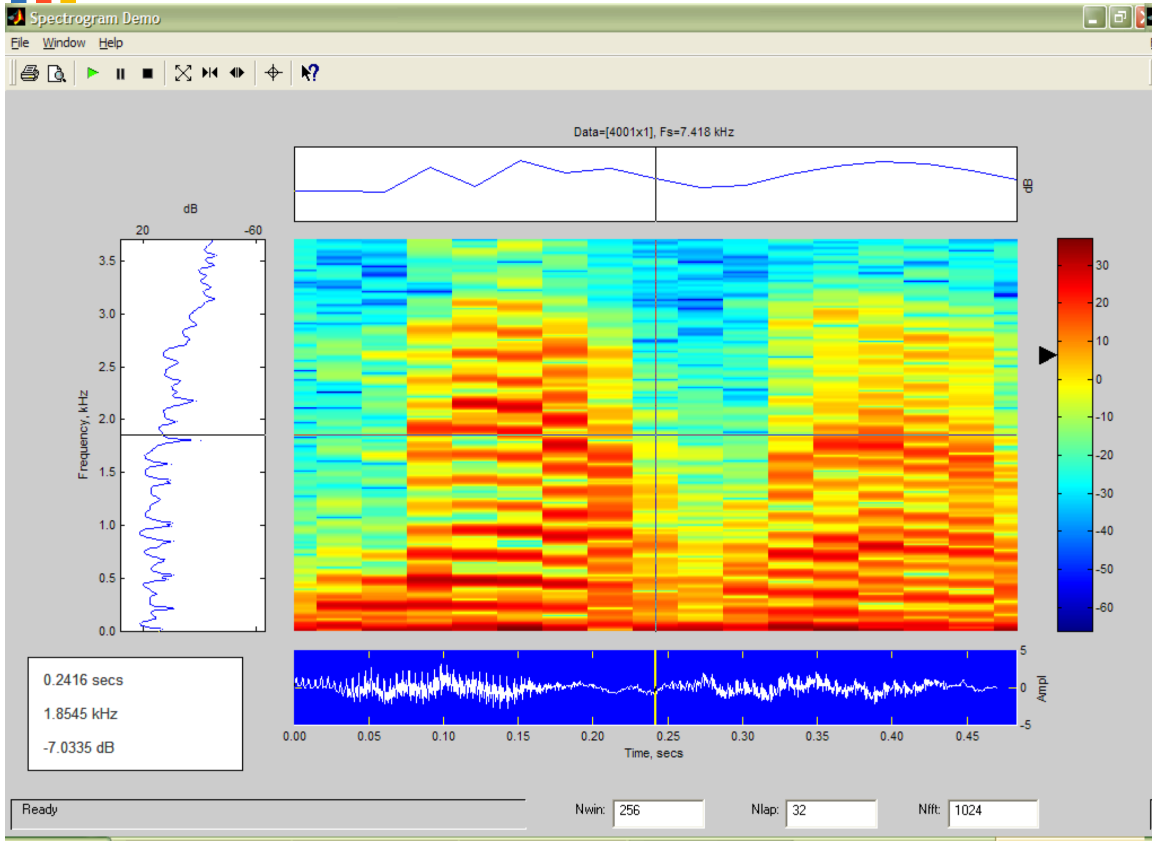
Typical voiced and unvoiced speech have different distribution in spectrum

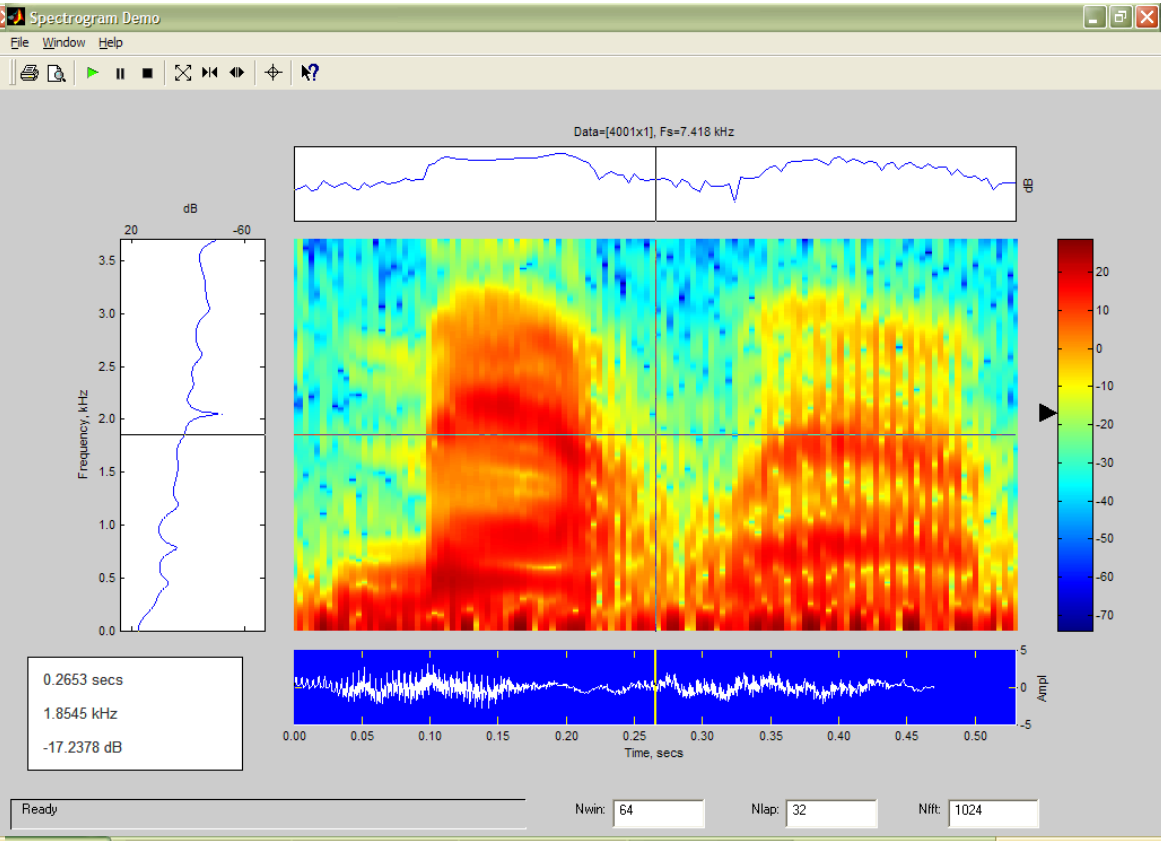






Spectrogram





Cepstral Features

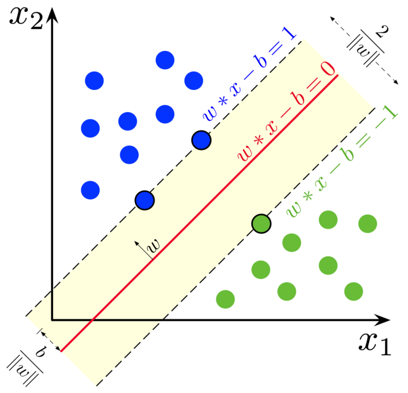
◆Features based on cepstrum

- Mel-frequency cepstral coefficients (MFCC)
- Linear prediction cepstral coefficients (LPCC)

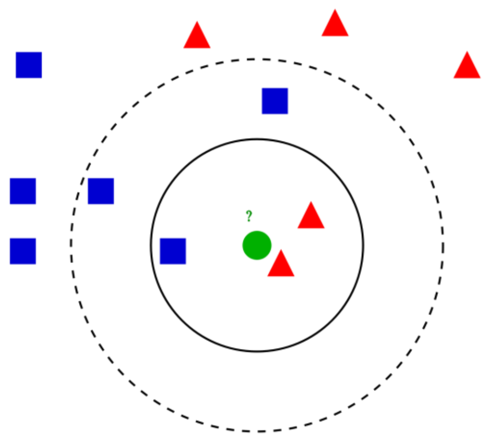
Models

Classifiers

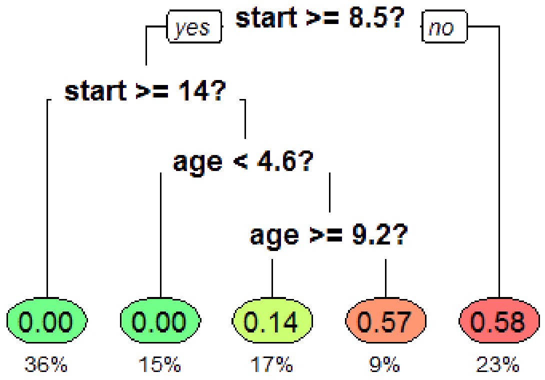
This Lab



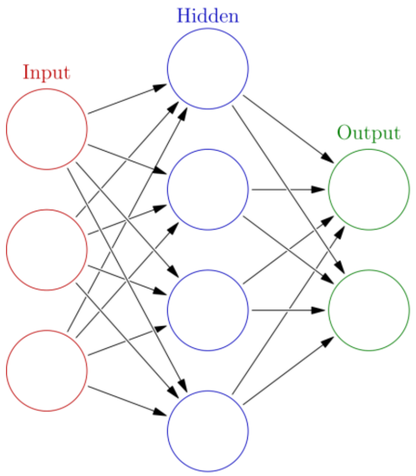
Support Vector Machine



K-nearest neighbors



Decision Tree

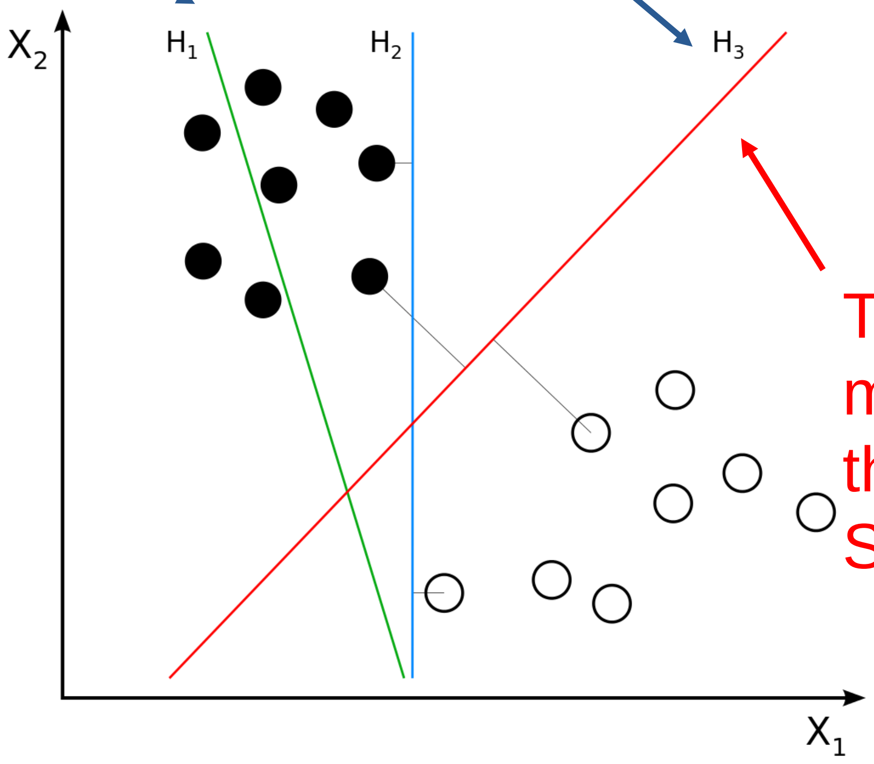


Neural network

BIIC

Linear Classifiers

The margin varies with different decision boundaries.



The one with maximum margin is the simplest type of SVM (Linear SVM).

Support Vector Machine

Maximum Margin Classifier: Find w, b that

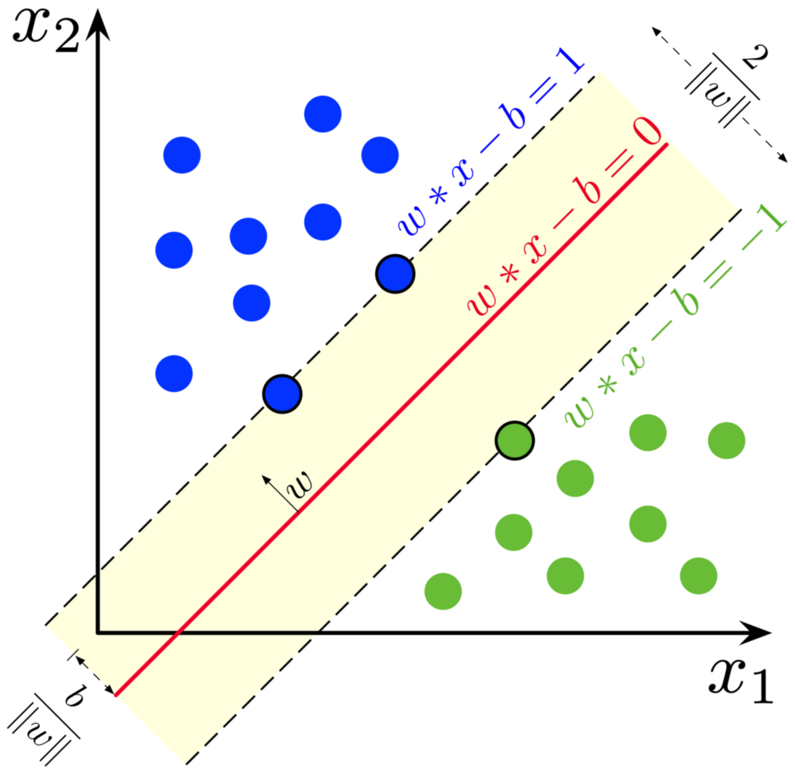
1. Get all samples correct

$$\begin{cases} wx_i + b \geq 1 \text{ for } y_i = +1 \\ wx_i + b \leq -1 \text{ for } y_i = -1 \end{cases}$$

$$\Rightarrow \quad y_i(wx_i + b) \geq 1 \text{ for all samples}$$

2. Maximize margin

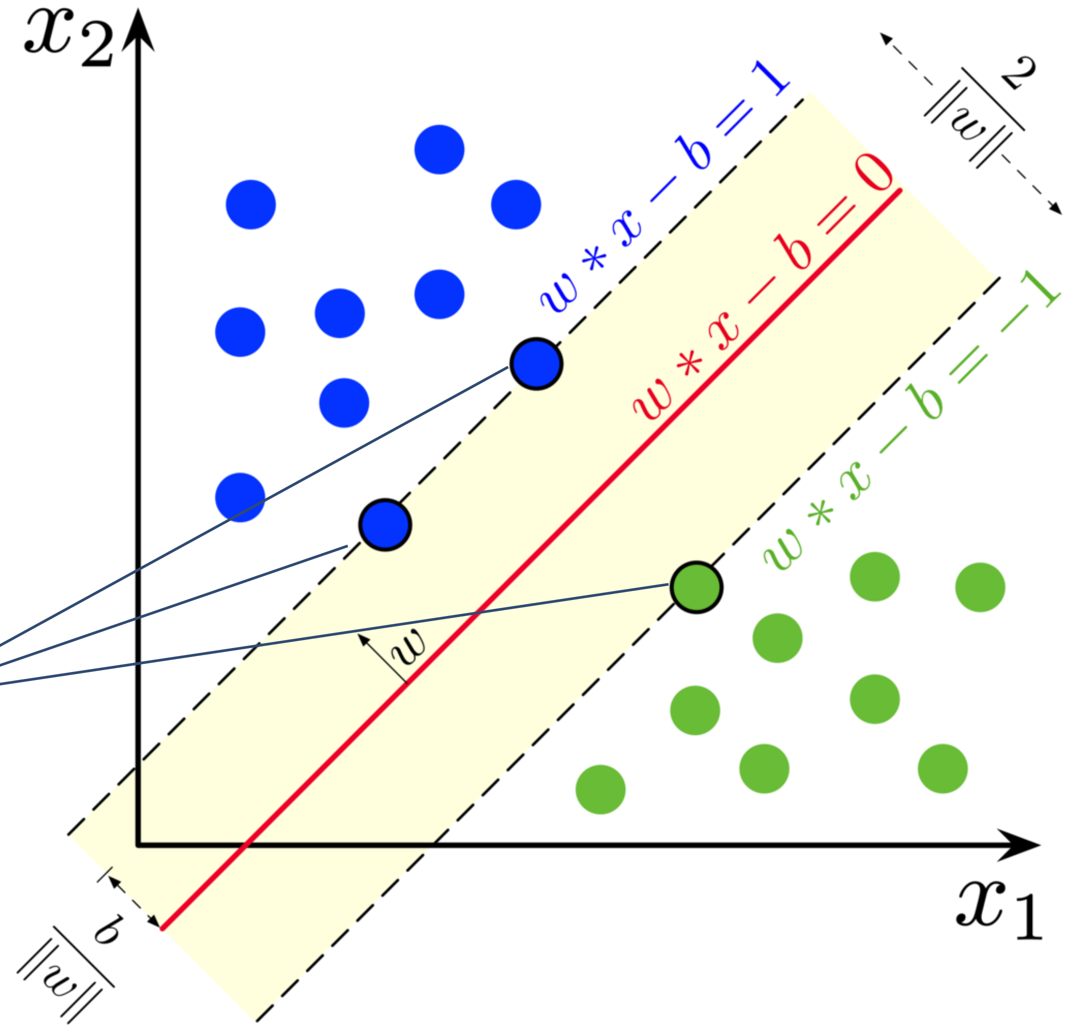
$$argmax \frac{2}{\|w\|} \quad \Rightarrow \quad argmin \frac{1}{2}\|w\|^2$$



Support Vector Machine

- Solution to this problem:
  - $w = \sum_n^N a_n t_n x_n$
  - $b = y_k - w^T x_k$ for any x_k such that $\alpha_k \neq 0$
  - Either $a_n = 0$ or $y_n y(x_n) = 1$

- Each non-zero a_n implies the corresponding x_n is a support vector



Support Vectors
$a_n \neq 0$

Support Vector Machine

◆The classifying function:

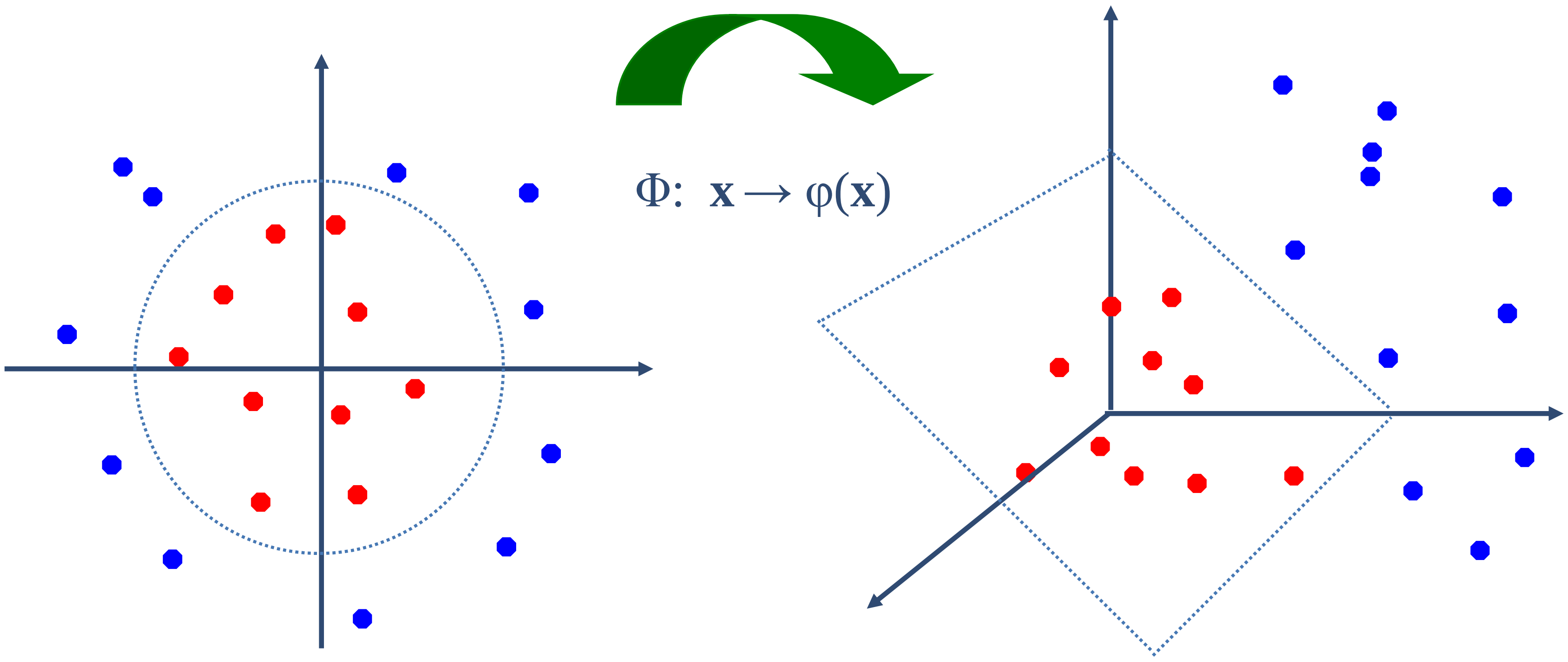
- $y(x) = w^T x + b = \sum_n^N a_n t_n \, x_n^T x + b$

◆Output y only relies on $x_n^T x$ the inner product between test sample and support vectors

◆We can use kernel functions $k(x, x')$ to replace simple $x_n^T x$

Non-Linear SVM

Data samples are mapped into high-dimensional space through kernels functions, and we find the hyperplane



Some Kernel Functions

- Linear: $\langle x, x' \rangle$
- Polynomial: $(\Gamma\langle x, x' \rangle + \boldsymbol{r})^{\boldsymbol{d}}$
- Gaussian radial-basis function (rbf):
  $$\exp(-\Gamma\|x - x'\|^2)$$
- Sigmoid: $\tanh(\Gamma\langle x, x' \rangle + \boldsymbol{r})$

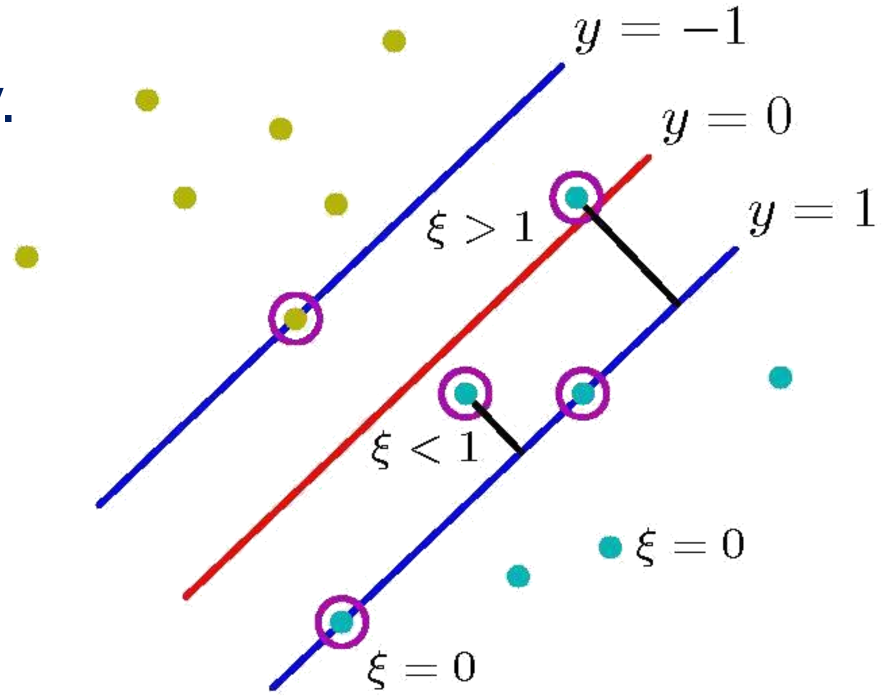
Overlapping Distribution

◆When the data is not completely separable, strong kernel could lead to overfitting.

◆We allow misclassification but with penalty.

◆A penalty variable ξ is increased

by the distance from that boundary



Overlapping Distribution

- The objective to be minimized becomes

$$\frac{1}{2}||w||^2 + C\sum_{n=1}^{N}\xi_n$$

C → ∞ for separable data

Large C: high accuracy but poor generalization
Small C: low accuracy but good generalization

BIIC

Multiclass Classification

◆One-vs-One:
- Train on every two classes
- Total of n_class * (n_class - 1) / 2 models
- The class with most votes as final output

◆One-vs-Rest:
- Train on one class and the remaining as others
- Total of n_class models
- The class with highest decision score as final output

Evaluation Metrics

Confusion matrix

Prediction

Ground Truth		Yes	No
	Yes	TP (True Positive)	FN (False Negative)
	No	FP (False Positive)	TN (True Negative)

Precision = TP/(TP+FP)
Recall = TP /(TP+FN)
Accuracy: (TP+FN)/(TP+FN+TN+FP)

P(positive): Predict YES
N(negative): Predict NO

T(True): Predict Correctly
F(False): Predict Wrongly

Cross Validation

Better generalization to unknown data & finding parameters