INT104 ARTIFICIAL INTELLIGENCE

L10- Unsupervised Learning II
Gaussian mixture model (GMM)

Fang Kang
Fang.kang@xjtlu.edu.cn

Xi'an Jiaotong-Liverpool University
西交利物浦大学

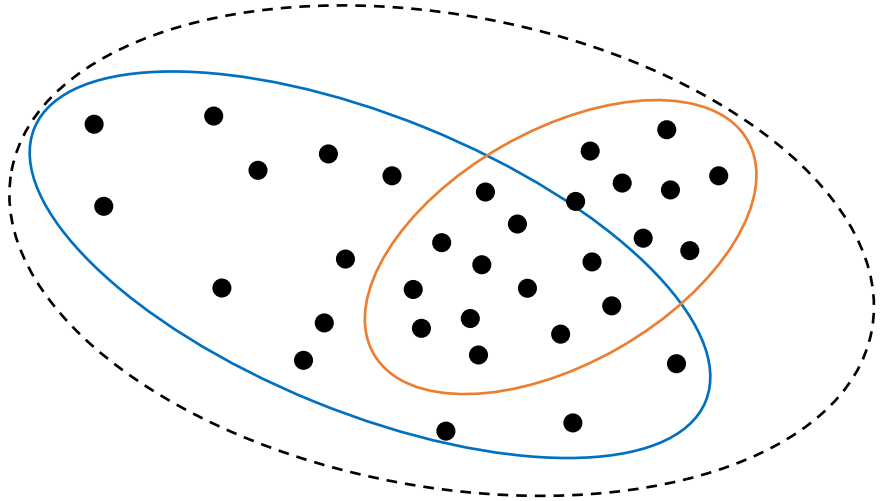
CONTENT

- Mixture Gaussian Model and EM method
  - Gaussian distribution
  - Mixture of gaussians
  - EM (Expectation-Maximization) method
- AI for Application

Motivation

K-means make ***hard*** assignments to data points: $x^{(i)}$ must belong to one of the clusters $1,2,\cdots,K$

Sometimes, one data point can belong to multiple clusters

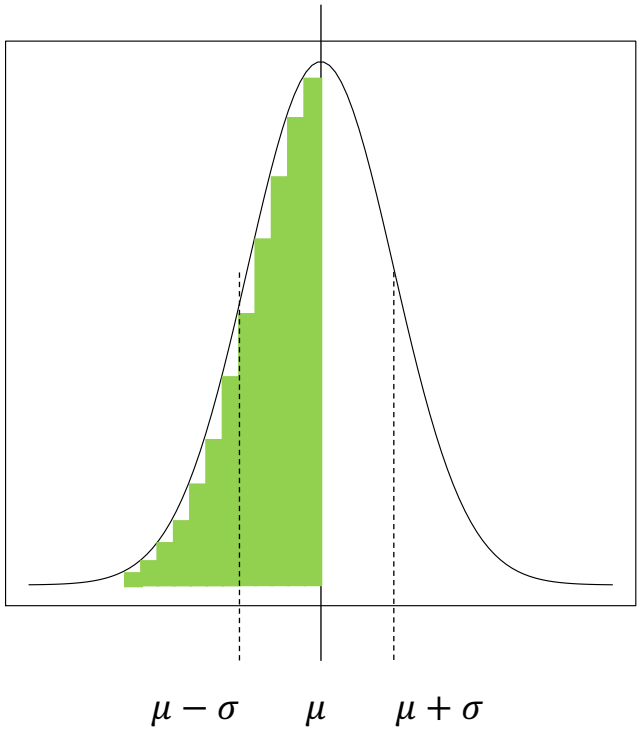


- Clusters may overlap
- Hard assignment may be simplistic
- Need a ***soft*** assignment:
  data points belong to clusters with different ***probabilities***

Gaussian (Normal) distribution

1-D (univariate) Gaussian $\mathcal{N}(\mu, \sigma)$

Probability density function (PDF): $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ μ: mean σ: standard deviation



$$P(x < \mu) = \int_{-\infty}^{\mu} p(x)dx = 0.5 = P(x > \mu)$$

$$P(x < \mu - \sigma) = \int_{-\infty}^{\mu} p(x)dx \approx 0.157 = P(x > \mu + \sigma)$$

Gaussian is ubiquitous

In biology, the *logarithm* of various variables
- Measures of size: length, height, weight, …
- Blood pressure of adult humans

In finance, the logarithm of change rates
- Price indices
- Stock market indices

In linguistics
- Word frequency
- Sentence length

Tend to have a Gaussian distribution

Gaussian model

μ and σ fully define a gaussian distribution

Use them as parameter $\theta = (\mu, \sigma)$ to define the model:
suppose each data point is randomly *drawn* from the distribution

μ, σ are **unknown**, but they can be learned (estimated) from **data**

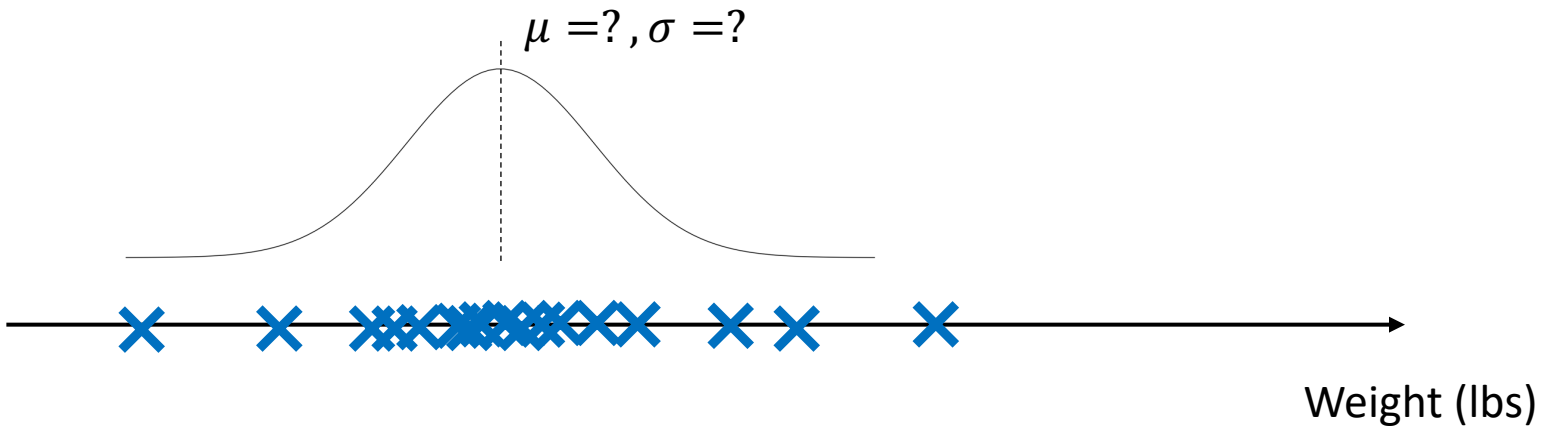
Job: find the parameters that best fit the data

What is "best fit"? → **Maximum Likelihood Estimation** (**MLE**)

Gaussian model example

Data: weight of Salmon fish. Assumption: The weight is from a Gaussian distribution

Task: to estimate the μ, σ of Salman



Maximum Likelihood Estimation (MLE)

Given m data points $X = \{x^{(1)}, \cdots, x^{(m)}\}$ Fit a Gaussian model $\mathcal{N}(\mu, \sigma)$, $\theta = (\mu, \sigma)$

PDF at $x^{(i)}$: $p(x^{(i)}|\theta) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^{(i)}-\mu)^2}{2\sigma^2}}$ ⇨ How likely it is to observe $x^{(i)}$ given θ

Assuming all data points are independent, then the likelihood of observing the whole dataset:

$$p(X|\theta) = \prod_{i=1}^{m} p(x^{(i)}|\theta) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^{(i)}-\mu)^2}{2\sigma^2}}$$

A good estimation of θ needs to maximize $p(X|\theta)$, the **likelihood** of data given the parameters

Maximum Likelihood Estimation (MLE) (cont.)

Likelihood function: $$\mathcal{L}(\theta) = p(X|\theta) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^{(i)}-\mu)^2}{2\sigma^2}}$$

It is easier to work with **log-likelihood**:

$$\mathcal{LL}(\theta) = \log\big(\mathcal{L}(\theta)\big) = -\frac{m\log(2\pi)}{2} - m\log(\sigma) - \sum_{i=1}^{m} \frac{(x^{(i)}-\mu)^2}{2\sigma^2}$$

Goal: find the $\theta = (\mu, \sigma)$ that maximizes $\mathcal{LL}(\theta)$

Maximum Likelihood Estimation (MLE) (cont.)

$$\mathcal{LL}(\theta) = \log\big(\mathcal{L}(\theta)\big) = -\frac{m\log(2\pi)}{2} - m\log(\sigma) - \sum_{i=1}^{m}\frac{(x^{(i)}-\mu)^2}{2\sigma^2}$$

Take the derivative of $\mathcal{LL}(\theta)$ w.r.t μ and σ

$$\frac{\partial\mathcal{LL}(\theta)}{\partial\mu} = -\frac{1}{\sigma^2}\sum_{i=1}^{m}(x^{(i)}-\mu) = -\frac{1}{\sigma^2}\left[\sum_{i=1}^{m}x^{(i)} - m\mu\right]$$

$$\frac{\partial\mathcal{LL}(\theta)}{\partial\sigma} = -\frac{m}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{m}(x^{(i)}-\mu)^2$$

$\mathcal{LL}(\theta)$ has extreme values when $\frac{\partial\mathcal{LL}(\theta)}{\partial\mu} = 0$ and $\frac{\partial\mathcal{LL}(\theta)}{\partial\sigma} = 0$

$$\Rightarrow \quad \mu = \frac{1}{m}\sum_{i=1}^{m}x^{(i)} = \bar{X} \qquad \sigma = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\mu)^2}$$

Mean of data
(sample mean)

Variance of data
(sample variance)

When μ is estimated by $\bar{X}$,

$\sigma = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(x^{(i)}-\bar{X})^2} = \sqrt{Var(X)}$

in order to get an unbiased estimate

These are the reasonable estimates of μ and σ from the data

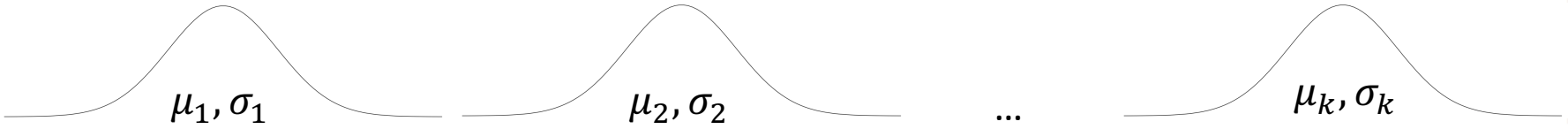
Mixture of Gaussians

Previous example has the assumption that data are drawn from ***one*** Gaussian distribution $\mathcal{N}(\mu, \sigma)$

What if there are ***multiple*** Gaussian distributions: $\mathcal{N}(\mu_1, \sigma_1)$, $\mathcal{N}(\mu_2, \sigma_2)$, …, $\mathcal{N}(\mu_k, \sigma_k)$

How do we generate the data?

Step 1: Draw from k distributions with probabilities $Q_1, Q_2, \cdots, Q_k$

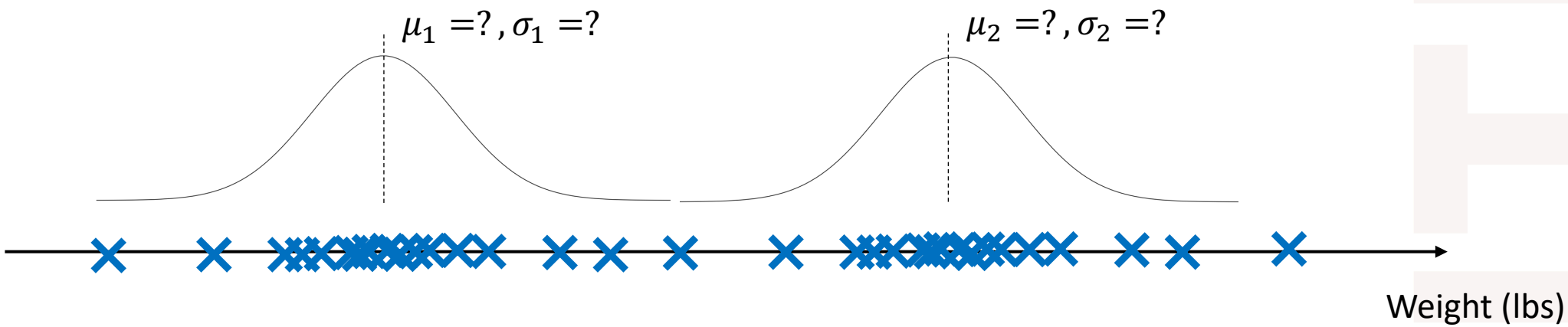


Step 2: Suppose distribution j is chosen, draw a data point from $\mathcal{N}(\mu_j, \sigma_j)$

$$p\left(x^{(i)} \middle| \mu_j, \sigma_j\right) = \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(x^{(i)}-\mu_j)^2}{2{\sigma_j}^2}}$$

Example of 2 Gaussians

Weights of two kinds of fish: Salmon & Tuna fish



How a data point is generated

A data point $x^{(i)}$ is generated according to the following process:

First, select the fish *kind* with

- Probability ϕ_S of being Salmon
- Probability ϕ_T of being Tuna
- $\phi_S + \phi_T = 1$

Given the fish *kind*, generate the data point from the corresponding Gaussian distribution

- $p\left(x^{(i)}\middle|S\right) \sim \mathcal{N}(\mu_S, \sigma_S)$ for Salmon
- $p\left(x^{(i)}\middle|T\right) \sim \mathcal{N}(\mu_T, \sigma_T)$ for Tuna

Introduce latent (unobserved) variable

Model parameters: $\Theta = (\phi_S, \phi_T, \mu_S, \mu_T, \sigma_S, \sigma_T)$

Parameters for mixture probabilities

Parameters for each Gaussian distribution

For each data point $x^{(i)}$, we don't know if it is a Salmon or Tuna

Let $z^{(i)}$ be the latent random variable indicating which Gaussian distribution $x^{(i)}$ is from

$z^{(i)} = 1$ for Salmon, $z^{(i)} = 2$ for Tuna

Then the likelihood of $x^{(i)}$ is:

$$p(x^{(i)}|\Theta) = \sum_{z^{(i)}} p(x^{(i)}, z^{(i)}|\Theta)$$

Let Q_i be the distribution of $z^{(i)}$
s.t. $\sum_{z^{(i)}} Q_i(z^{(i)}) = 1$
$Q_i(z^{(i)} = j)$ is the probability of $z^{(i)} = j$

Rewrite the likelihood

$$p(x^{(i)}|\Theta) = \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)}|\Theta)}{Q_i(z^{(i)})}$$

Log likelihood of data

The likelihood of the whole data: $\mathcal{L}(\theta) = p(X|\Theta) = \prod_{i=1}^{m} p\left(x^{(i)}, z^{(i)}|\Theta\right) = \prod_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)}|\Theta)}{Q_i(z^{(i)})}$

Log likelihood: $\mathcal{LL}(\theta) = \sum_{i=1}^{m} \log\left(\sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)}|\Theta)}{Q_i(z^{(i)})}\right) = \sum_{i=1}^{m} \log\left(Q_i(z^{(i)}=1) \frac{p\left(x^{(i)}, z^{(i)}|\Theta\right)}{Q_i(z^{(i)}=1)} + Q_i(z^{(i)}=2) \frac{p\left(x^{(i)}, z^{(i)}|\Theta\right)}{Q_i(z^{(i)}=2)}\right)$

It is difficult to take the derivative of $\mathcal{LL}(\theta)$ w.r.t. $\phi_S, \phi_T, \mu_S, \mu_T, \sigma_S, \sigma_T$, and solve them analytically

Solution: Instead of maximizing $\mathcal{LL}(\theta)$, we can maximize the lower bound of $\mathcal{LL}(\theta)$

Idea: Find some expression E, s.t. $\mathcal{LL}(\theta) \geq E$. When we maximize E, $\mathcal{LL}(\theta)$ is also maximized.

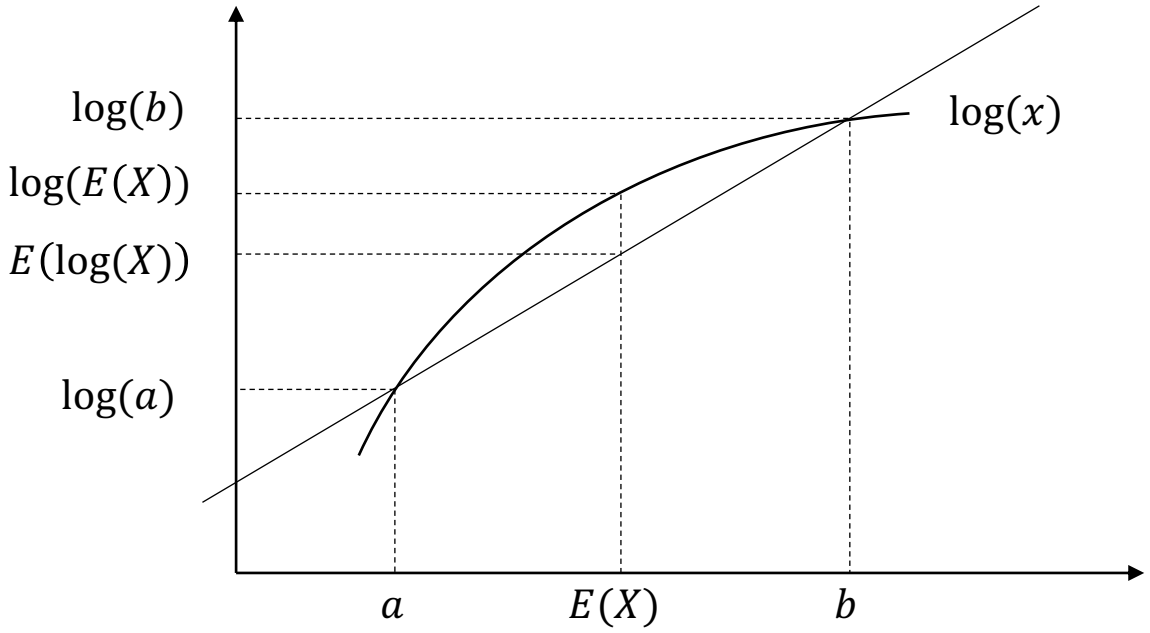
E should have a form that is easier to calculate derivatives

Find the lower bound of $\mathcal{LL}(\theta)$ (optional)

$$\mathcal{LL}(\theta) = \sum_{i=1}^{m} \log\left(\underbrace{Q_i\left(z^{(i)} = 1\right)}_{\text{Probability}} a + \underbrace{Q_i\left(z^{(i)} = 2\right)}_{\text{Probability}} b \right)$$

Let a, b be two values of a random variable X

Then $Q_i\left(z^{(i)} = 1\right)a + Q_i\left(z^{(i)} = 2\right)b$ is the expectation of $E(X)$



Because log(x) is convex $\log\left(E(X)\right) \geq E(\log(X))$

$$\mathcal{LL}(\theta) \geq \sum_{i=1}^{m} Q_i\left(z^{(i)} = 1\right)\log(a) + Q_i\left(z^{(i)} = 2\right)\log(b)$$

$$= \sum_{i=1}^{m}\sum_{z^{(i)}} Q_i\left(z^{(i)}\right)\log\left(\frac{p(x^{(i)}, z^{(i)}|\Theta)}{Q_i(z^{(i)})}\right)$$

We need to replace $Q_i\left(z^{(i)}\right)$ with something we know

Jensen's inequality: $f(E(X)) \geq E(f(X))$, when f is convex

How to estimate Q_i (optional)

$$\mathcal{LL}(\theta) \geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \log\left(\frac{p(x^{(i)}, z^{(i)}|\Theta)}{Q_i(z^{(i)})}\right)$$

$Q_i(z^{(i)})$ is unknown, but we can guess it after observing $x^{(i)}$

I.e., after observing a data point $x^{(i)}$, we can "guess" which distribution it is from

A **reasonable** way to guess:

If $x^{(i)}$ is drawn from Salmon, then the likelihood of $x^{(i)}$ is $p(x^{(i)}|S)p(S) = p(x^{(i)}|\mu_S, \sigma_S)\phi_S$

$$\frac{1}{\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}$$

If $x^{(i)}$ is drawn from Tuna, then the likelihood of $x^{(i)}$ is $p(x^{(i)}|T)p(T) = p(x^{(i)}|\mu_T, \sigma_T)\phi_T$

Then the chance of $x^{(i)}$ being Salmon is:

$$p(S|x^{(i)}) = \underbrace{\frac{p(x^{(i)}|S)p(S)}{p(x^{(i)}|S)p(S) + p(x^{(i)}|T)p(T)}}_{\text{Posterior, } w_S^{(i)}}$$

The chance of $x^{(i)}$ being Tuna is:

$$p(T|x^{(i)}) = \underbrace{\frac{p(x^{(i)}|T)p(T)}{p(x^{(i)}|S)p(S) + p(x^{(i)}|T)p(T)}}_{\text{Posterior, } w_T^{(i)}}$$

New form of Log-likelihood function (optional)

$$\mathcal{LL}(\theta) \geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i\left(z^{(i)}\right) \log\left(\frac{p\left(x^{(i)}, z^{(i)} \middle| \Theta\right)}{Q_i\left(z^{(i)}\right)}\right) = \sum_{i=1}^{m} w_S^{(i)} \log\left(\frac{p\left(x^{(i)}, z^{(i)} = 1 \middle| \Theta\right)}{w_S^{(i)}}\right) + w_T^{(i)} \log\left(\frac{p\left(x^{(i)}, z^{(i)} = 2 \middle| \Theta\right)}{w_T^{(i)}}\right) = \mathcal{LL}'(\theta)$$

$$p\left(x^{(i)}, z^{(i)} = 1 \middle| \Theta\right) = p\left(x^{(i)} \middle| \mu_S, \sigma_S\right)\phi_S = \frac{\phi_S}{\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}$$

$$p\left(x^{(i)}, z^{(i)} = 2 \middle| \Theta\right) = p\left(x^{(i)} \middle| \mu_T, \sigma_T\right)\phi_T = \frac{\phi_T}{\sqrt{2\pi}\sigma_T} e^{-\frac{(x^{(i)}-\mu_T)^2}{2\sigma_T^2}}$$

Treating w_S and w_T as known, the derivatives of $\mathcal{LL}'(\theta)$ is much easier to calculate

$$\lfloor \mathcal{LL}(\theta) \rfloor = \mathcal{LL}'(\theta) = \sum_{i=1}^{m} w_S^{(i)} \log\left(\frac{\phi_S}{w_S^{(i)}\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}\right) + w_T^{(i)} \log\left(\frac{\phi_T}{w_T^{(i)}\sqrt{2\pi}\sigma_T} e^{-\frac{(x^{(i)}-\mu_T)^2}{2\sigma_T^2}}\right)$$

Maximizing $\mathcal{LL}'(\theta)$ (optional)

$$[\mathcal{LL}(\theta)] = \mathcal{LL}'^{(\theta)} = \sum_{i=1}^{m} w_S^{(i)} \log\left(\frac{\phi_S}{w_S^{(i)}\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}\right) + w_T^{(i)} \log\left(\frac{\phi_T}{w_T^{(i)}\sqrt{2\pi}\sigma_T} e^{-\frac{(x^{(i)}-\mu_T)^2}{2\sigma_T^2}}\right)$$

$$\frac{\partial \mathcal{LL}'(\theta)}{\partial \mu_S} = \sum_{i=1}^{m} \frac{\partial}{\partial \mu_S}\left[w_S^{(i)} \log\left(\frac{\phi_S}{\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}\right)\right] = \sum_{i=1}^{m} w_S^{(i)}(x^{(i)} - \mu_S) = 0 \quad \Rightarrow \quad \mu_S = \frac{\sum_{i=1}^{m} w_S^{(i)} x^{(i)}}{\sum_{i=1}^{m} w_S^{(i)}}$$

$$\frac{\partial \mathcal{LL}'(\theta)}{\partial \sigma_S} = \sum_{i=1}^{m} \frac{\partial}{\partial \sigma_S}\left[w_S^{(i)} \log\left(\frac{\phi_S}{\sqrt{2\pi}\sigma_S} e^{-\frac{(x^{(i)}-\mu_S)^2}{2\sigma_S^2}}\right)\right] = \sum_{i=1}^{m} w_S^{(i)}[(x^{(i)} - \mu_S)^2 - \sigma_S^2] = 0 \quad \Rightarrow \quad \sigma_S^2 = \frac{\sum_{i=1}^{m} w_S^{(i)}(x^{(i)} - \mu_S)^2}{\sum_{i=1}^{m} w_S^{(i)}}$$

Find the terms that only depends on ϕ_S and ϕ_T ⟶ ϕ_S and ϕ_T cannot take any value Under constraint: $\phi_S + \phi_T = 1$

$$\mathcal{LL}'(\theta) = \sum_{i=1}^{m} w_S^{(i)} \log(\phi_S) + w_T^{(i)} \log(\phi_T) \longrightarrow \text{Construct a Lagrangian:} \quad \mathcal{L}(\phi_S) = \left(\sum_{i=1}^{m} w_S^{(i)} \log(\phi_S) + w_T^{(i)} \log(\phi_T)\right) + \beta(\phi_S + \phi_T - 1)$$

$$\frac{\partial \mathcal{L}(\phi_S)}{\partial \phi_S} = \frac{\sum_{i=1}^{m} w_S^{(i)}}{\phi_S} + \beta = 0 \quad \Rightarrow \quad \phi_S = \frac{\sum_{i=1}^{m} w_S^{(i)}}{-\beta} \quad \phi_T = \frac{\sum_{i=1}^{m} w_T^{(i)}}{-\beta} \quad \Rightarrow \quad -\beta = \sum_{i=1}^{m}\left(w_S^{(i)} + w_T^{(i)}\right) = m$$

Solutions of maximizing $\mathcal{L}\mathcal{L}'(\theta)$ (optional)

$$\begin{cases} \mu_S = \dfrac{\sum_{i=1}^{m} w_S^{(i)} x^{(i)}}{\sum_{i=1}^{m} w_S^{(i)}} \\ \sigma_S^2 = \dfrac{\sum_{i=1}^{m} w_S^{(i)} \left(x^{(i)} - \mu_S\right)^2}{\sum_{i=1}^{m} w_S^{(i)}} \\ \phi_S = \dfrac{\sum_{i=1}^{m} w_S^{(i)}}{m} \end{cases} \qquad \begin{cases} \mu_T = \dfrac{\sum_{i=1}^{m} w_T^{(i)} x^{(i)}}{\sum_{i=1}^{m} w_T^{(i)}} \\ \sigma_T^2 = \dfrac{\sum_{i=1}^{m} w_T^{(i)} \left(x^{(i)} - \mu_T\right)^2}{\sum_{i=1}^{m} w_T^{(i)}} \\ \phi_T = \dfrac{\sum_{i=1}^{m} w_T^{(i)}}{m} \end{cases}$$

Repeatedly update all parameters, $\phi_S, \phi_T, \mu_S, \mu_T, \sigma_S, \sigma_T$ until convergence

In which, $$w_S^{(i)} = p\left(S|x^{(i)}\right) = \frac{p\left(x^{(i)}|S\right)\phi_S}{p\left(x^{(i)}|S\right)\phi_S + p\left(x^{(i)}|T\right)\phi_T}$$

$$w_T^{(i)} = p\left(T|x^{(i)}\right) = \frac{p\left(x^{(i)}|S\right)\phi_S}{p\left(x^{(i)}|S\right)\phi_S + p\left(x^{(i)}|T\right)\phi_T}$$

E-M (Expectation-Maximization) Algorithm (1-D Gaussian)

Assume the data $\{x^{(i)}\}$ are drawn from *k* Gaussian distributions with probabilities $\phi_1, \phi_2, \cdots, \phi_k$
Each distribution has parameters μ_j, σ_j $(j = 1,2, \cdots, k)$

Randomly initialize all parameters $\phi_1, \phi_2, \cdots, \phi_k$ and μ_j, σ_j $(j = 1,2, \cdots, k)$

Repeat until convergence {

E-step: For each $x^{(i)}$, compute the expectation of which distribution it is from

$$w_j^{(i)} := p\left(z^{(i)} = j \middle| x^{(i)}\right) = \frac{p(x^{(i)}|\mu_j, \sigma_j)\phi_j}{\sum_j p(x^{(i)}|\mu_j, \sigma_j)\phi_j} \qquad \text{For } j = 1,2, \cdots, k$$

M-step: Update the parameters (as if $w_j^{(i)}$ is correct) by maximizing the likelihood:

$$\underbrace{\mu_j := \frac{\sum_{i=1}^m w_j^{(i)} x^{(i)}}{\sum_{i=1}^m w_j^{(i)}}}_{\text{Weighted average}} \qquad \sigma_j^2 := \frac{\sum_{i=1}^m w_j^{(i)} \left(x^{(i)} - \mu_j\right)^2}{\sum_{i=1}^m w_j^{(i)}} \qquad \phi_j := \frac{\sum_{i=1}^m w_j^{(i)}}{m} \qquad \text{For } j = 1,2, \cdots, k$$

}

Compare with *K*-means

Randomly initialize all *k* centroids $\mu_1, \mu_2, \cdots, \mu_k$

Repeat until convergence {

E-step: For each $x^{(i)}$, assign it to the closest centroid

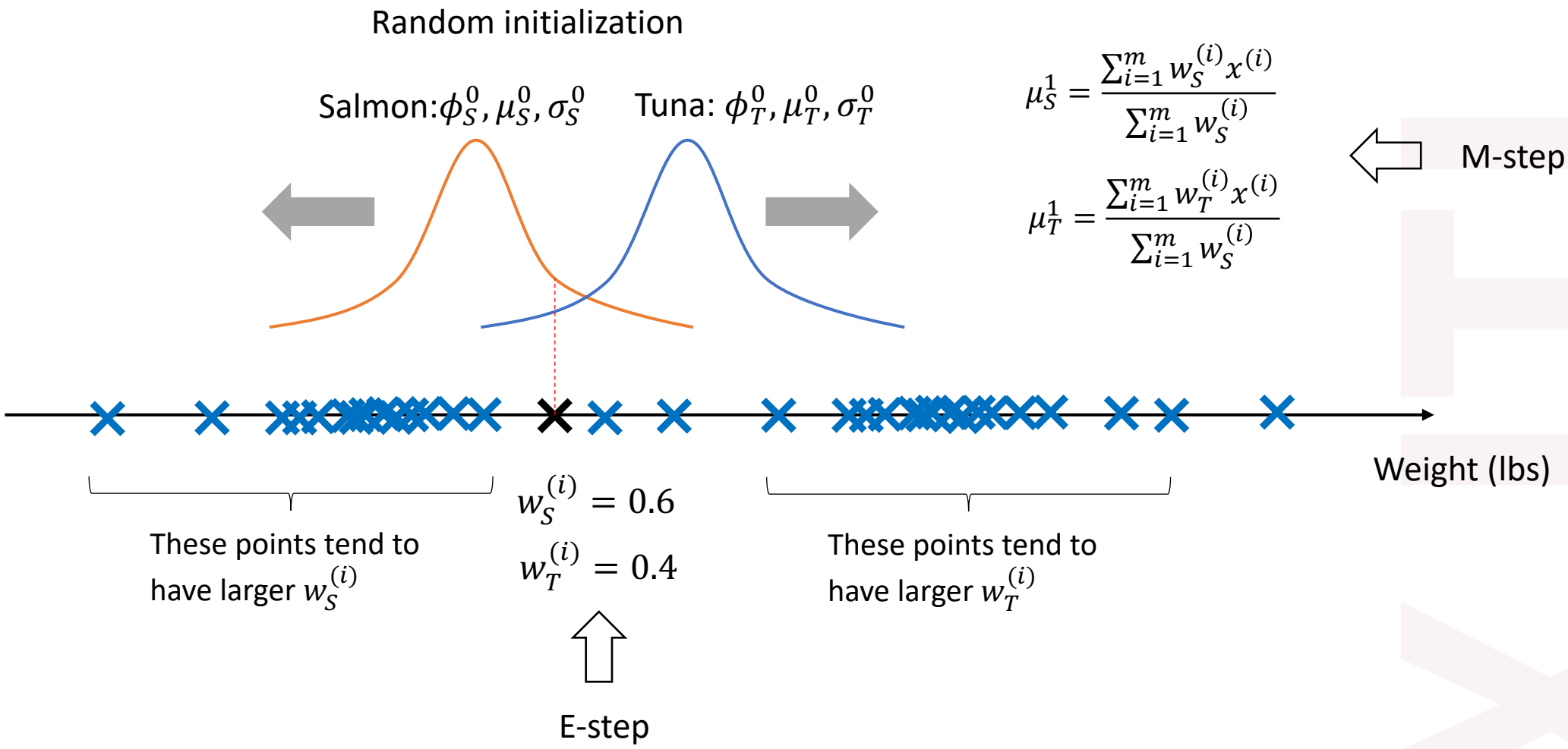
$$c^{(i)} := \arg\min_j \left\| x^{(i)} - \mu_j \right\|^2$$

M-step: Update the positions of centroids

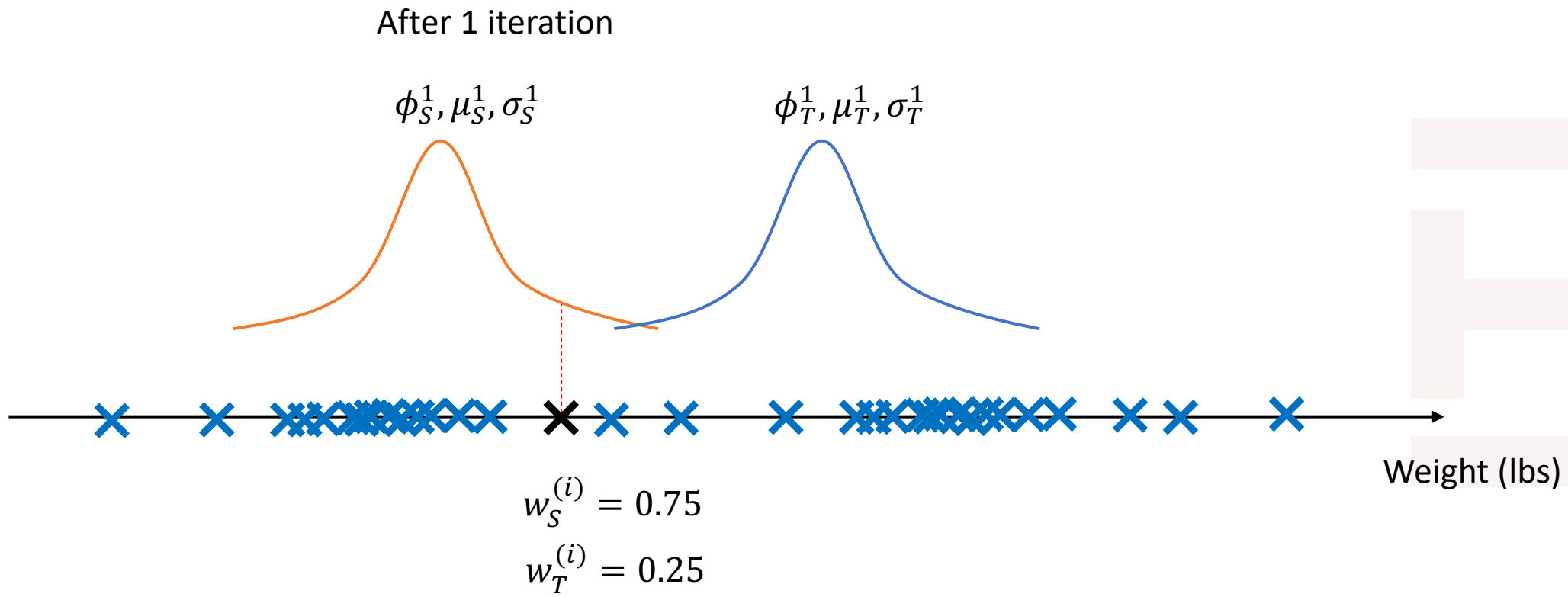
$$\mu_j := \frac{\sum_{i=1}^{m} 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c^{(i)} = j\}}$$

}

Demonstration with k = 2, 1-D Gaussian

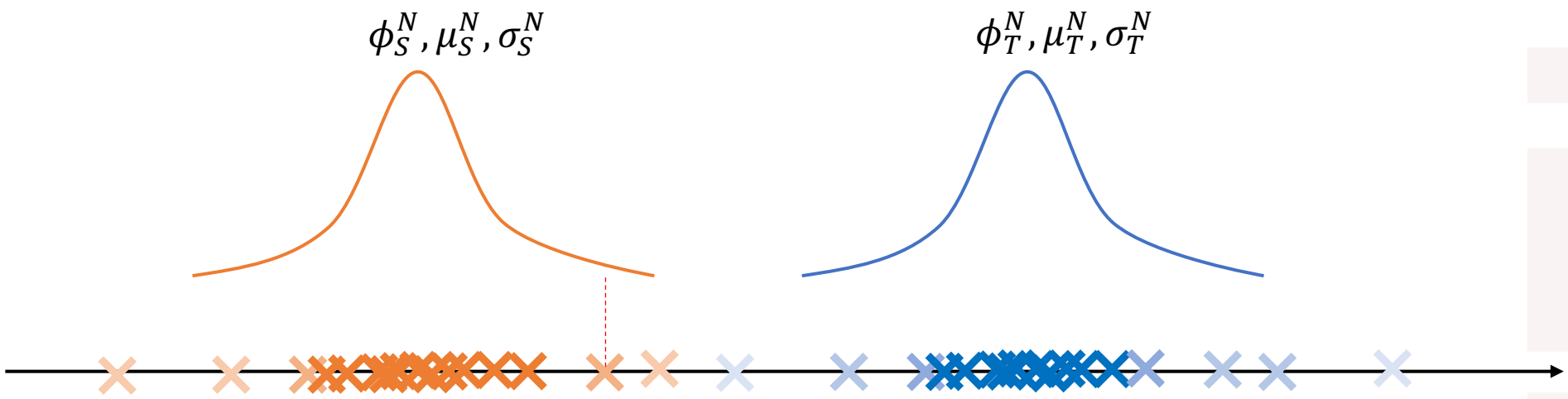


Demonstration with k = 2, 1-D Gaussian



Demonstration with k = 2, 1-D Gaussian

After N iterations, all parameters converge



$w_S^{(i)} = 0.77$

$w_T^{(i)} = 0.23$

This data point is 0.77 chance a Salmon, and 0.23 chance a Tuna

No hard assignment!

What about multivariate Gaussians?

A random vector $X = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix}$ is said to have a multivariate Gaussian distribution

If its probability density function is: $$p(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

Mean: $\mu \in \mathbb{R}^n$ Covariance matrix: Σ

Property: $$\frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) dx_1 dx_2 \cdots dx_n = 1.$$

Covariance matrix

If X_i, Y_j are a pair of 1-D random variables

Then the covariance is defined as: $Cov[X_i, Y_j] = E\left[\left(X - E(X_i)\right)\left(Y - E(Y_j)\right)\right] = E[X_i Y_j] - E(X_i)E(Y_j)$

If $X = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix}$, $Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}$ are a pair of n-D random variables

Then the covariance matrix Σ is a $n \times n$ symmetric matrix

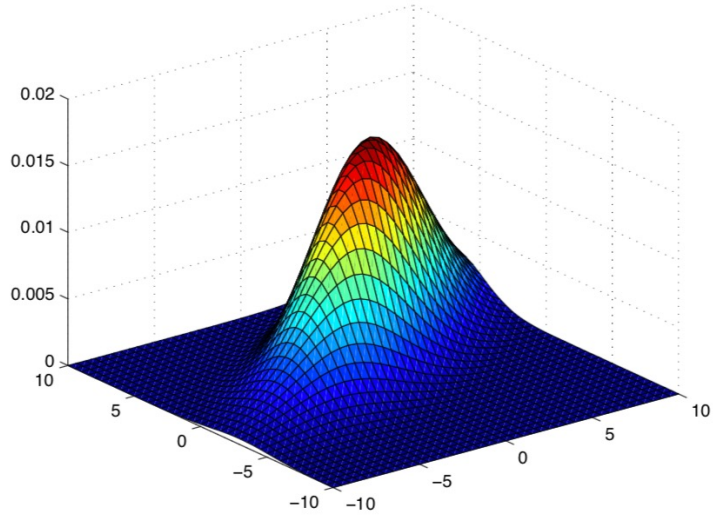
whose (i, j) th entry is $Cov[X_i, Y_j]$

$$\Sigma = \begin{bmatrix} Cov[X_1, Y_1] & Cov[X_1, Y_2] & \cdots & Cov[X_1, Y_n] \\ Cov[X_2, Y_1] & Cov[X_2, Y_2] & & Cov[X_2, Y_n] \\ \vdots & \vdots & & \vdots \\ Cov[X_n, Y_1] & Cov[X_n, Y_2] & & Cov[X_n, Y_n] \end{bmatrix}$$

When n=2, 2-D Gaussian distribution

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2 \\ \sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

$$p(x) = \frac{1}{2\pi \begin{vmatrix} \sigma_1^2 & \sigma_1\sigma_2 \\ \sigma_1\sigma_2 & \sigma_2^2 \end{vmatrix}^{1/2}} \exp\left(-\frac{1}{2} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2 \\ \sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}^{-1} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \right)$$



Special case: covariance matrix is diagonal

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

$$p(x) = \frac{1}{2\pi \begin{vmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{vmatrix}^{1/2}} \exp\left(-\frac{1}{2} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}^{-1} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \right)$$

$$= \frac{1}{2\pi\sqrt{\sigma_1^2 \sigma_2^2}} \exp\left(-\frac{1}{2} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \begin{bmatrix} 1/\sigma_1^2 & 0 \\ 0 & 1/\sigma_2^2 \end{bmatrix}^{-1} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \right)$$

$$= \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{1}{2\sigma_1^2}(x_1 - \mu_1)^2 - \frac{1}{2\sigma_2^2}(x_2 - \mu_2)^2 \right)$$

$$= \underbrace{\frac{1}{2\pi\sigma_1} \exp\left(-\frac{1}{2\sigma_1^2}(x_1 - \mu_1)^2 \right)}_{\text{PDF for } x_1} \cdot \underbrace{\frac{1}{2\pi\sigma_2} \exp\left(-\frac{1}{2\sigma_2^2}(x_2 - \mu_2)^2 \right)}_{\text{PDF for } x_2}$$

⇨ Product of two independent 1-D Gaussian distribution

Contours of 2-D Gaussians

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

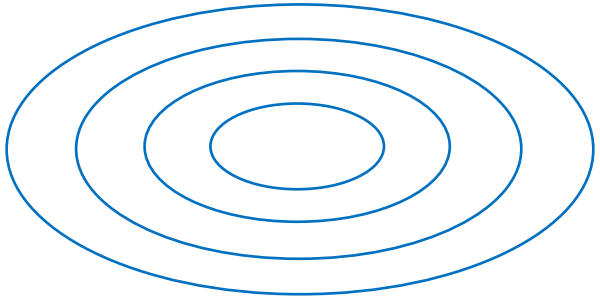
$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

$$p(x) = \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{1}{2\sigma_1^2}(x_1 - \mu_1)^2 - \frac{1}{2\sigma_2^2}(x_2 - \mu_2)^2 \right)$$

To draw contours, let $p(x)$ be a constant

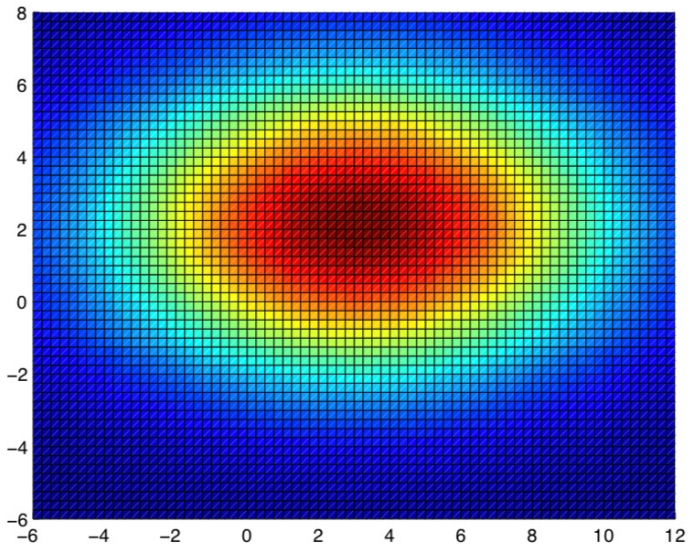
$$p(x) = c \quad \Rightarrow \quad 1 = \frac{(x_1 - \mu_1)^2}{2\sigma_1^2 \log\left(\frac{1}{2\pi c\sigma_1\sigma_2}\right)} + \frac{(x_2 - \mu_2)^2}{2\sigma_2^2 \log\left(\frac{1}{2\pi c\sigma_1\sigma_2}\right)}$$



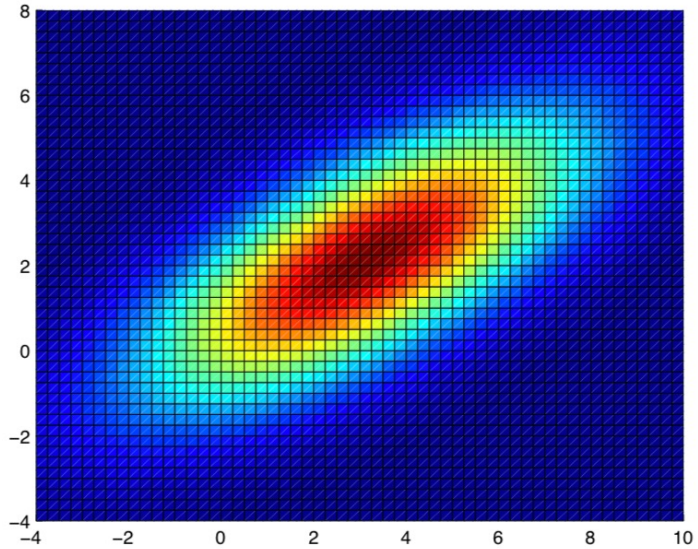
$$1 = \frac{(x_1 - \mu_1)^2}{r_1^2} + \frac{(x_2 - \mu_2)^2}{r_2^2}$$

An ellipse!

Covariance matrix decides the shape of ellipse



$$\mu = \begin{pmatrix} 3 \\ 2 \end{pmatrix} \qquad \Sigma = \begin{bmatrix} 25 & 0 \\ 0 & 9 \end{bmatrix}$$



$$\mu = \begin{pmatrix} 3 \\ 2 \end{pmatrix} \qquad \Sigma = \begin{bmatrix} 10 & 5 \\ 5 & 5 \end{bmatrix}$$

E-M algorithm for mixture of multivariate gaussians

Assume the data $\{x^{(i)}\}$ are drawn from *k n*-D Gaussian distributions with probabilities $\phi_1, \phi_2, \cdots, \phi_k$
Each distribution has parameters μ_j, Σ_j $(j = 1,2, \cdots, k)$

Randomly initialize all parameters $\phi_1, \phi_2, \cdots, \phi_k$ and μ_j, Σ_j $(j = 1,2, \cdots, k)$

Repeat until convergence {

E-step: For each $x^{(i)}$, compute the expectation of which distribution it is from

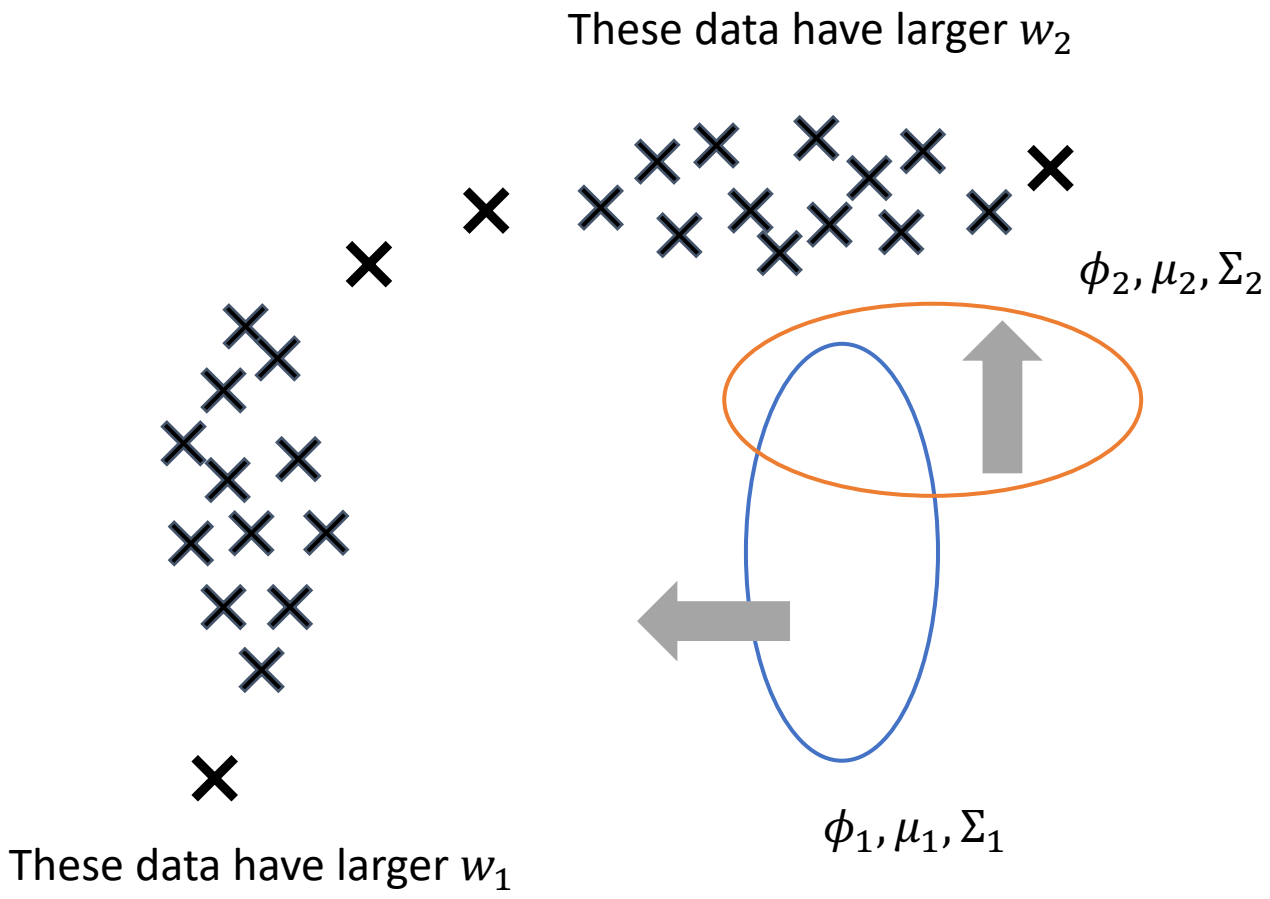
$$w_j^{(i)} := p\left(z^{(i)} = j \middle| x^{(i)}\right) = \frac{p(x^{(i)}|\mu_j, \Sigma_j)\phi_j}{\sum_j p(x^{(i)}|\mu_j, \Sigma_j)\phi_j} \qquad \text{For } j = 1,2, \cdots, k$$

M-step: Update the parameters (as if $w_j^{(i)}$ is correct) by maximizing the likelihood:

$$\mu_j := \frac{\sum_{i=1}^m w_j^{(i)} x^{(i)}}{\sum_{i=1}^m w_j^{(i)}} \qquad \Sigma_j := \frac{\sum_{i=1}^m w_j^{(i)} \left(x^{(i)} - \mu_j\right)\left(x^{(i)} - \mu_j\right)^T}{\sum_{i=1}^m w_j^{(i)}} \qquad \phi_j := \frac{\sum_{i=1}^m w_j^{(i)}}{m} \qquad \text{For } j = 1,2, \cdots, k$$

}

Demo of learning a mixture of 2-D Gaussians



Random initialization

For each $x^{(i)}$, compute

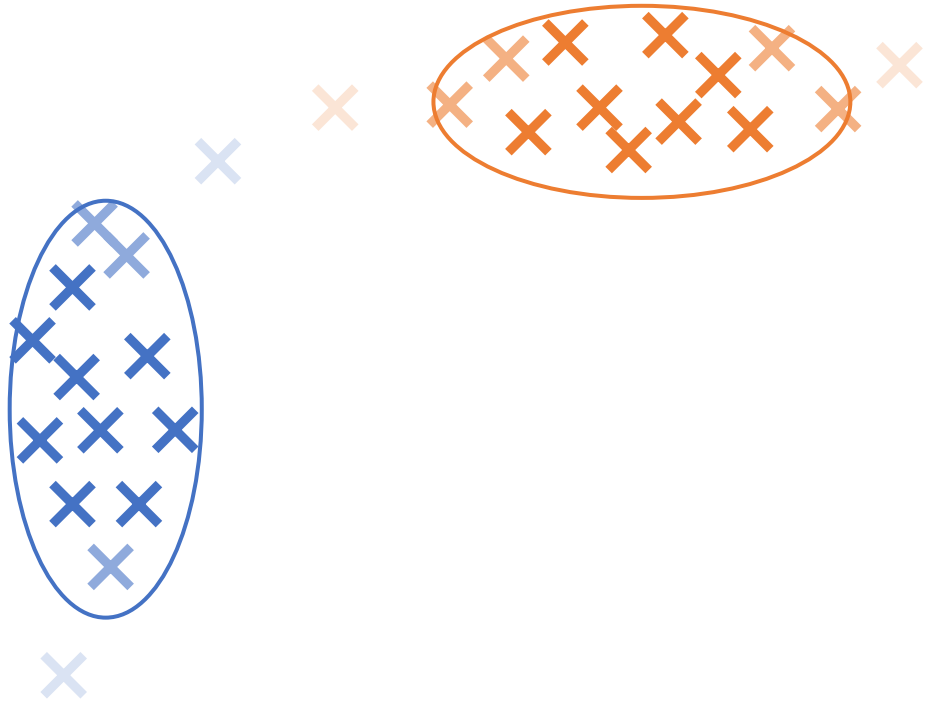
$$w_1^{(i)} := \frac{p(x^{(i)}|\mu_1, \Sigma_1)\phi_1}{p(x^{(i)}|\mu_1, \Sigma_1)\phi_1 + p(x^{(i)}|\mu_2, \Sigma_2)\phi_2}$$

$$w_2^{(i)} := \frac{p(x^{(i)}|\mu_2, \Sigma_2)\phi_2}{p(x^{(i)}|\mu_1, \Sigma_1)\phi_1 + p(x^{(i)}|\mu_2, \Sigma_2)\phi_2}$$

Update:

$$\mu_1 := \frac{\sum_{i=1}^m w_1^{(i)} x^{(i)}}{\sum_{i=1}^m w_1^{(i)}} \qquad \mu_2 := \frac{\sum_{i=1}^m w_2^{(i)} x^{(i)}}{\sum_{i=1}^m w_2^{(i)}}$$

Demo of learning a mixture of 2-D Gaussians (cont.)



CONTENT

- Mixture Gaussian Model and EM method
  - Gaussian distribution
  - Mixture of gaussians
  - EM (Expectation-Maximization) method
- AI for Application

Content

Outlines

01 Basic Knowledge of Sound

02 Communication Acoustics

03 AI for Audio Application

Basic Knowledge of Sound

XJTLU | SCHOOL OF FILM AND TV ARTS

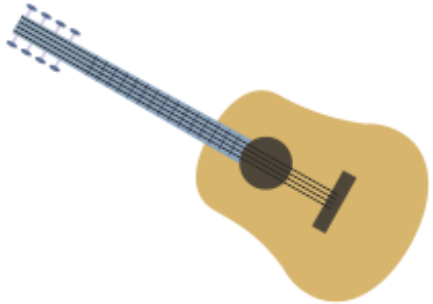
Xi'an Jiaotong-Liverpool University 西交利物浦大學

Basic Knowledge of Sound

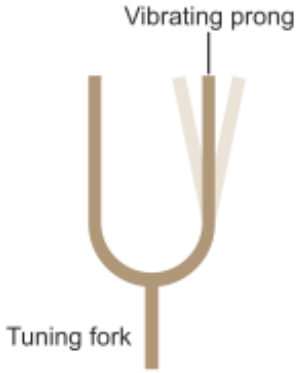
Production of Sound

- **Produced by vibration of an object**
- **Vibrations cause air molecules to oscillate**
- **Change in air pressure creates a wave**

When we hit the drum, membrane of drum vibrates producing sound.



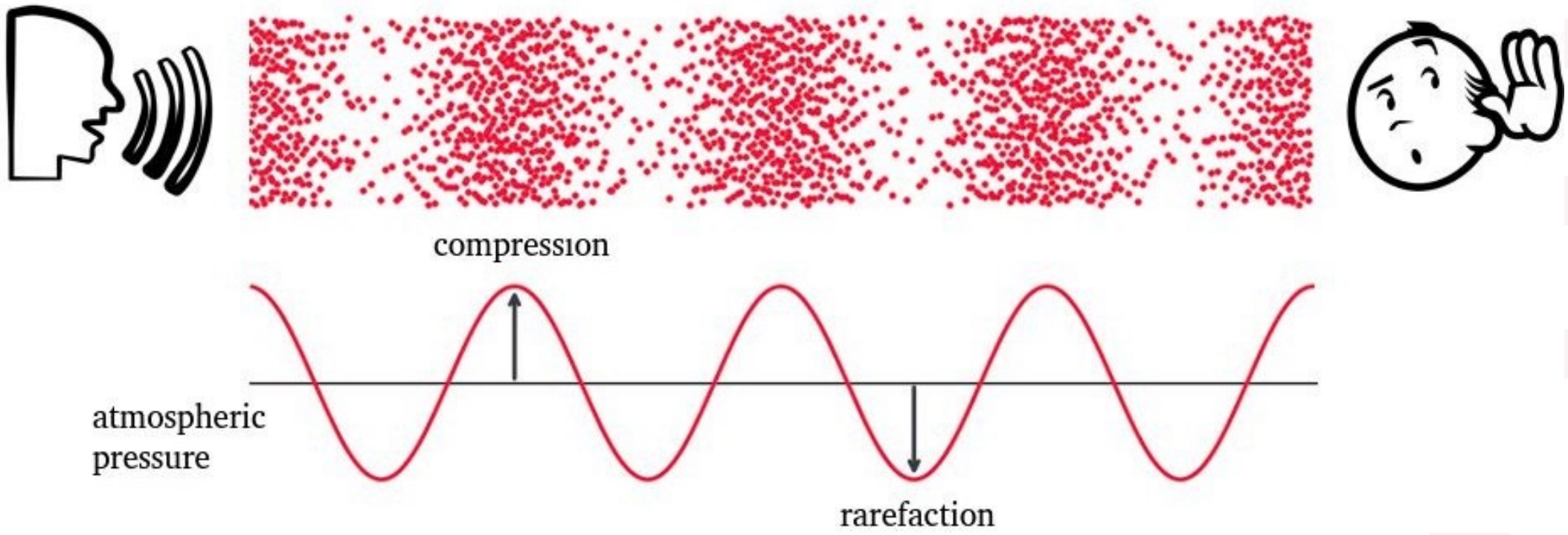
When we play a guitar, the string on it makes to and fro motion and produces sound.



Sound produced by vibrating prong of tuning fork.

Basic Knowledge of Sound

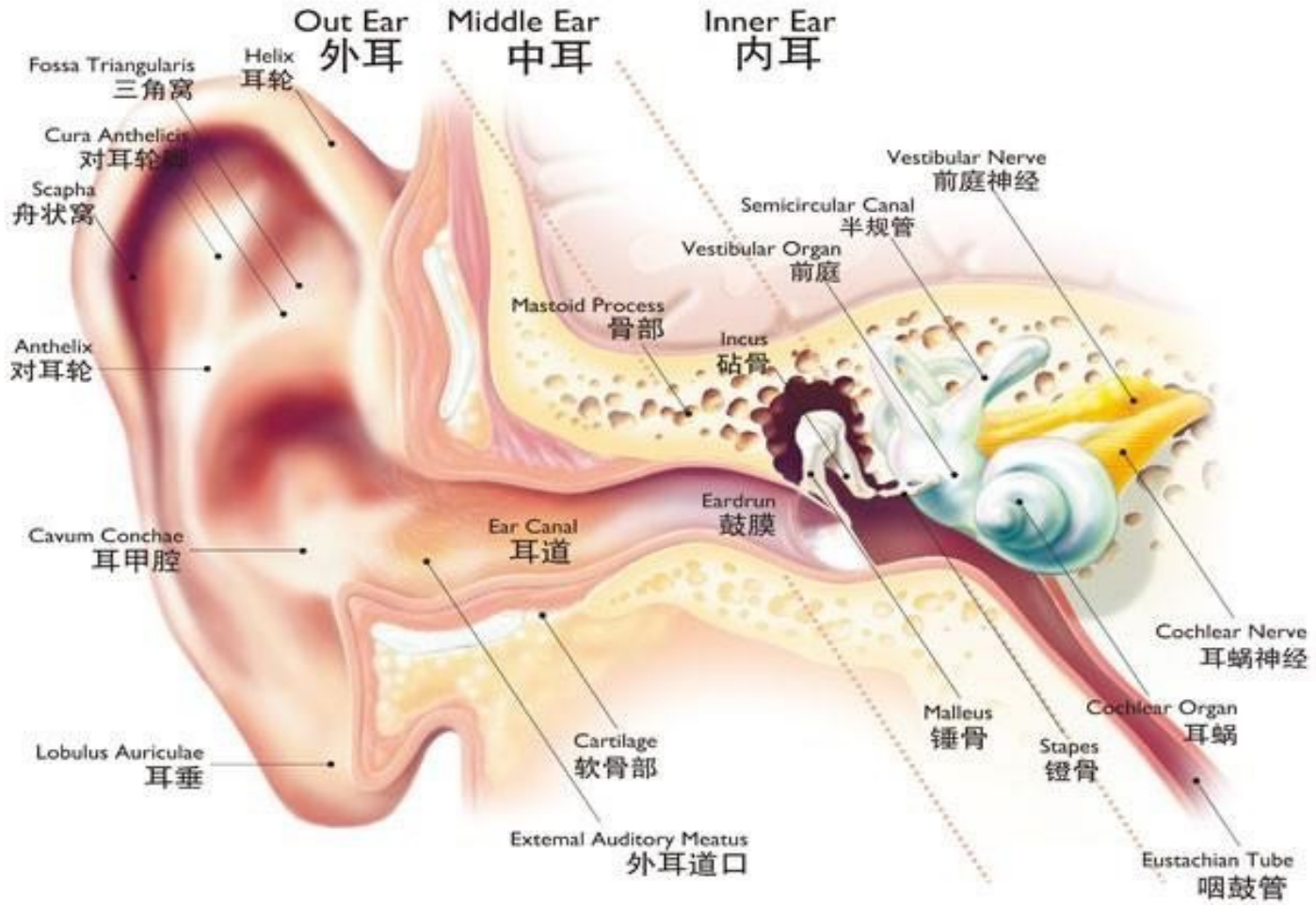
Propagation of Sound



Basic Knowledge of Sound

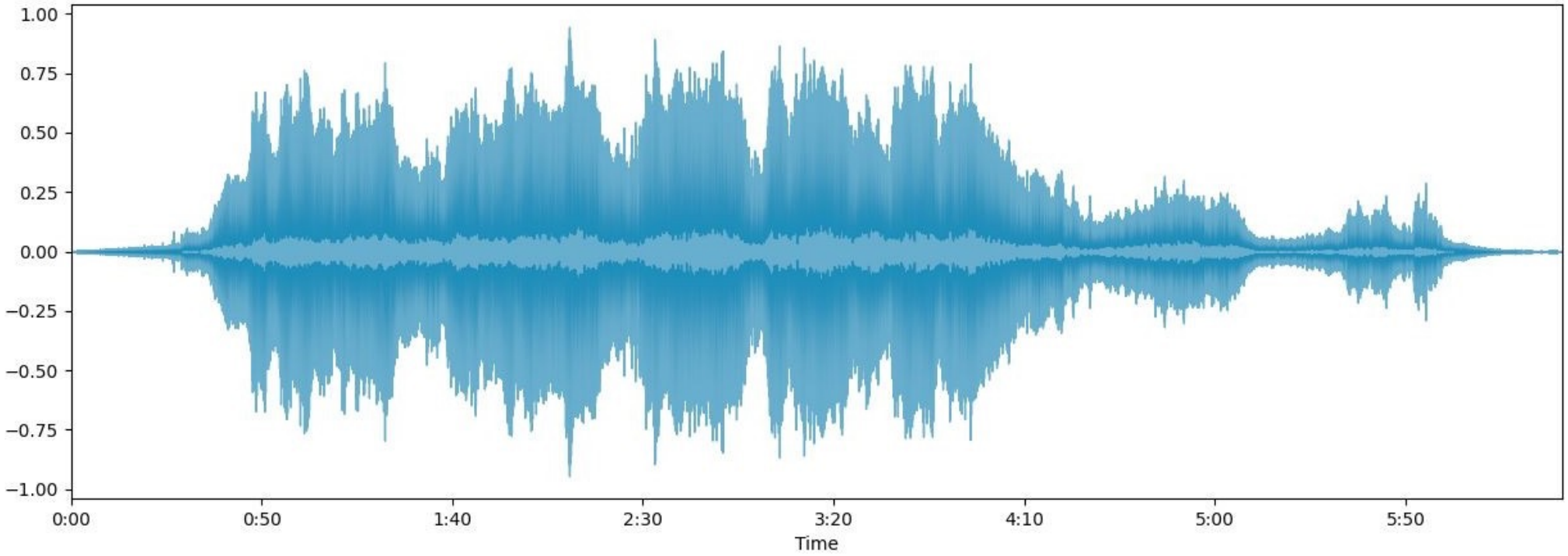
Reception of Sound

Sound
↓
Out Ear
↓
Eardrum
↓
Cochlear Organ
↓
Nerve
↓
Brain
↓



Basic Knowledge of Sound

Sound Waveform



Basic Knowledge of Sound

Waveform

- Carries multifactorial information:
  - Intensity
  - Timbre
  - Frequency

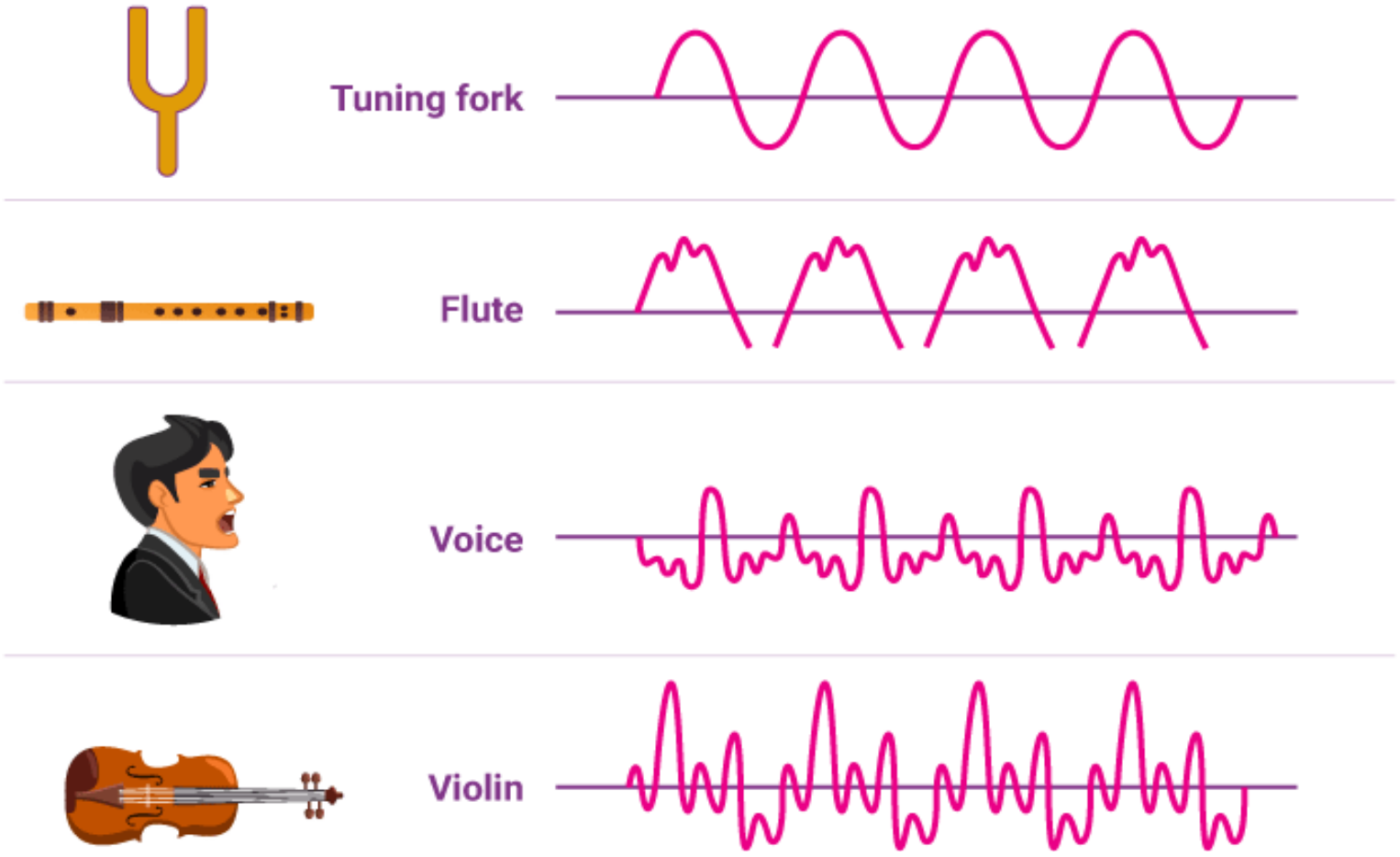
Basic Knowledge of Sound

Intensity Level

Source	Intensity	Intensity level	× TOH
Threshold of hearing (TOH)	10^{-12}	0 dB	1
Whisper	10^{-10}	20 dB	10^{2}
Pianissimo	10^{-8}	40 dB	10^{4}
Normal conversation	10^{-6}	60 dB	10^{6}
Fortissimo	10^{-2}	100 dB	10^{10}
Threshold of pain	10	130 dB	10^{13}
Jet take-off	10^{2}	140 dB	10^{14}
Instant perforation of eardrum	10^{4}	160 dB	10^{16}

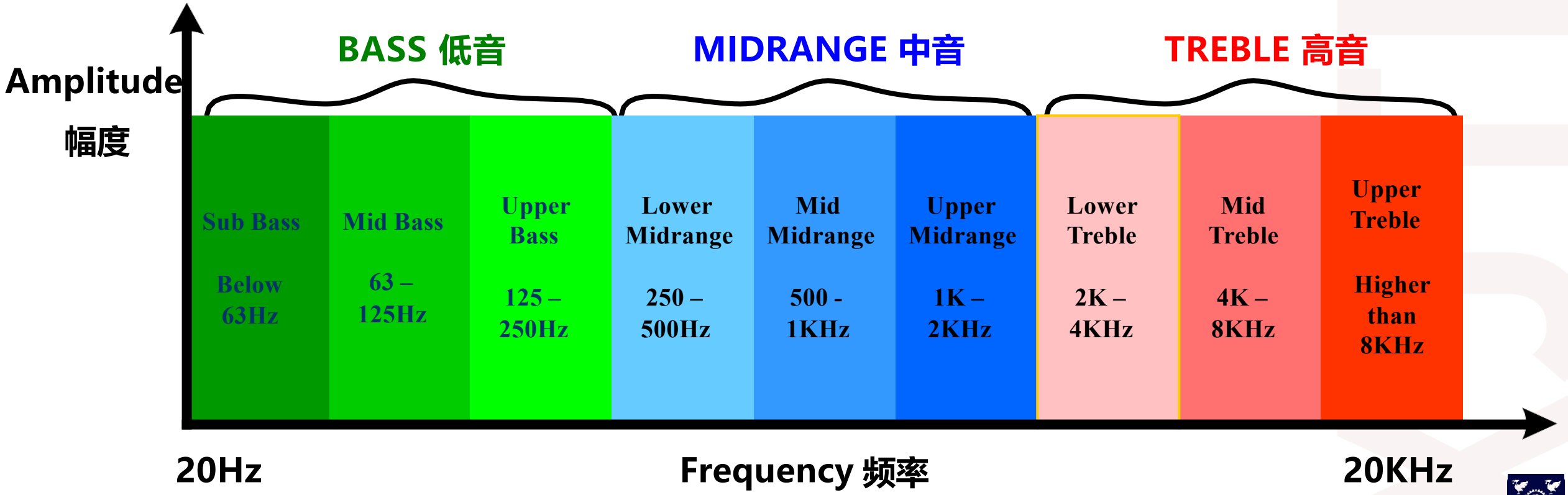
Basic Knowledge of Sound

Timbre of Sound



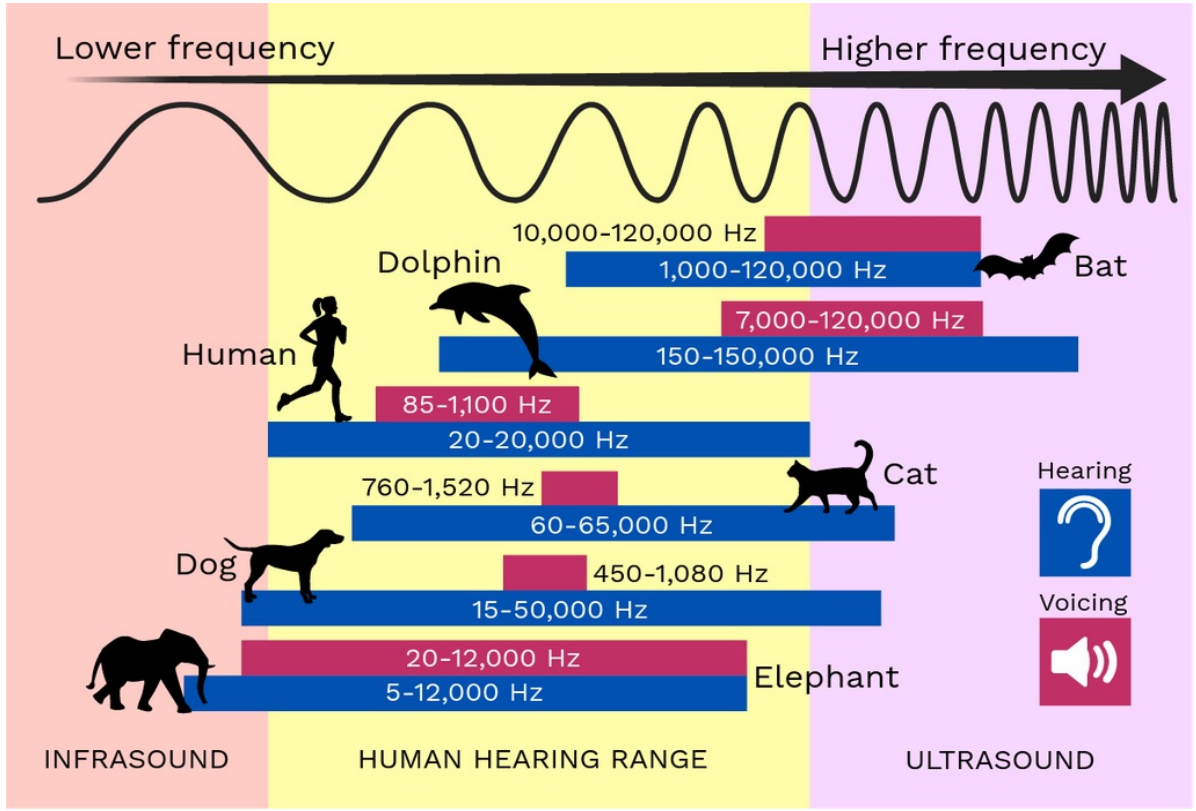
Basic Knowledge of Sound

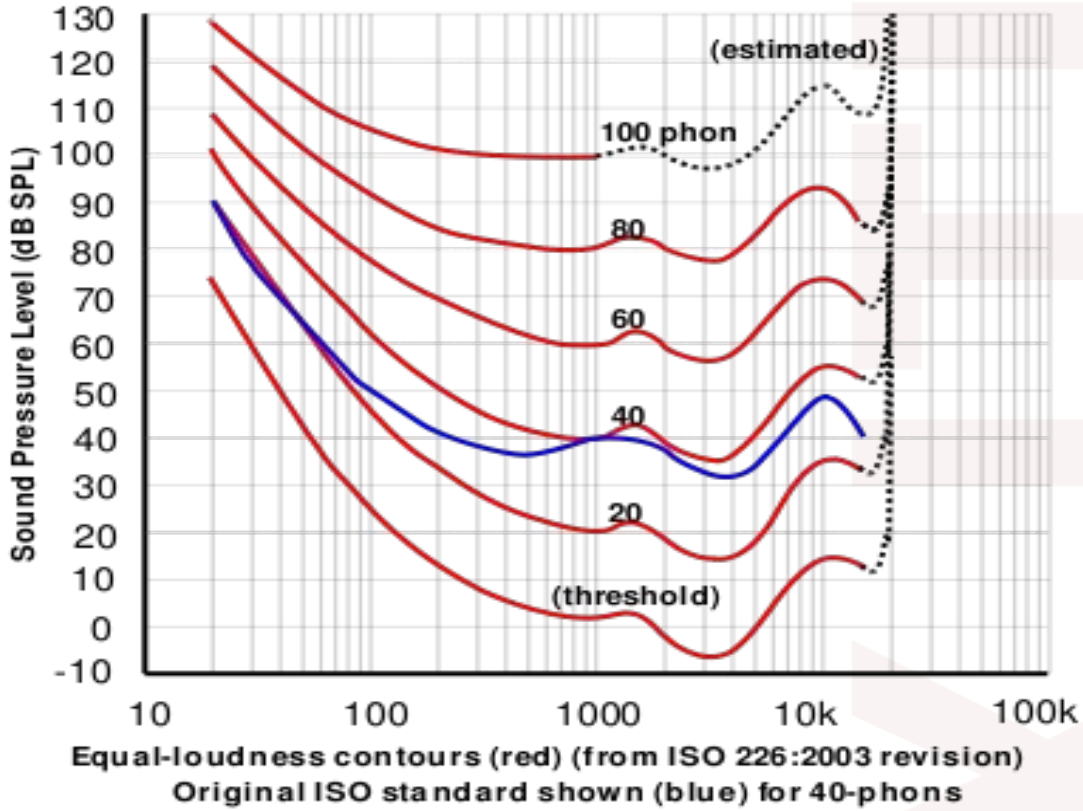
Frequency Band



Basic Knowledge of Sound

Hearing Range and Equal Loudness Contours

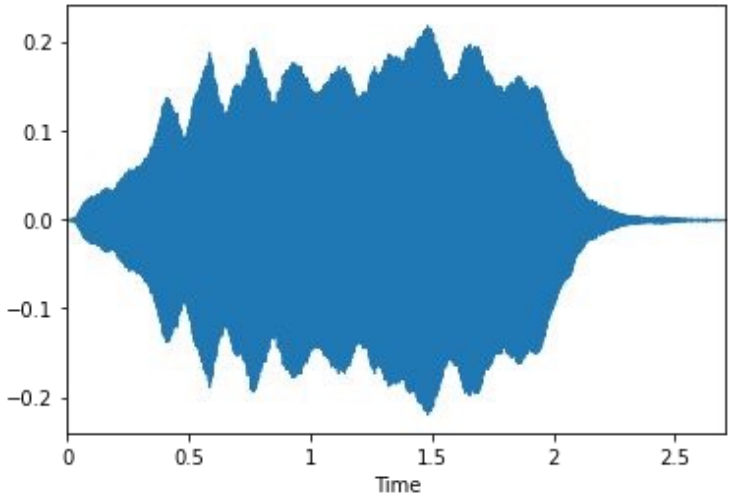


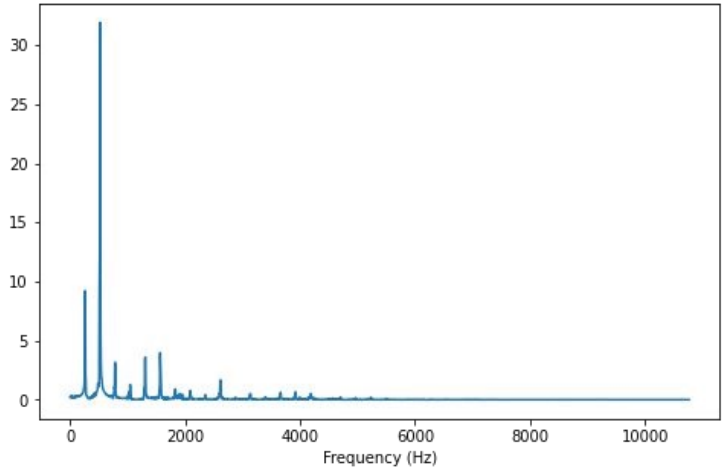


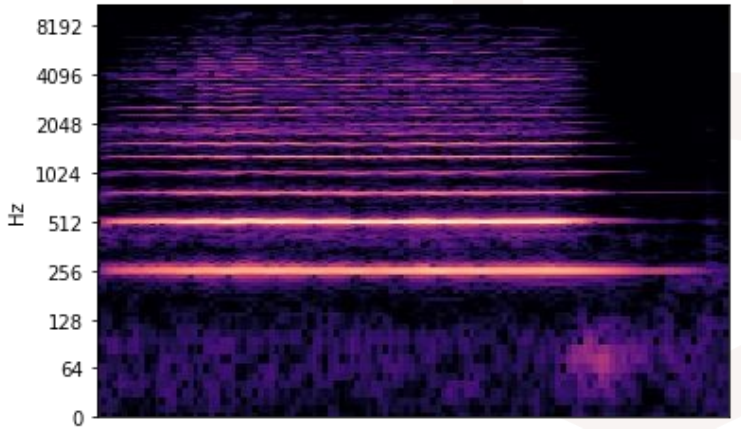
Basic Knowledge of Sound

Signal Domain

- Time domain
- Frequency domain
- Time-frequency representation







Basic Knowledge of Sound

Fourier Transform

- **Fourier Transform allows us to decompose a signal into its constituent frequencies, revealing the frequency content of the signal.**

$$F(\omega) = F\ [f(t)] = \int_{-\infty}^{\infty} f(t)e^{-jwt}dt$$

Basic Knowledge of Sound

Short-Time Fourier Transform (STFT)

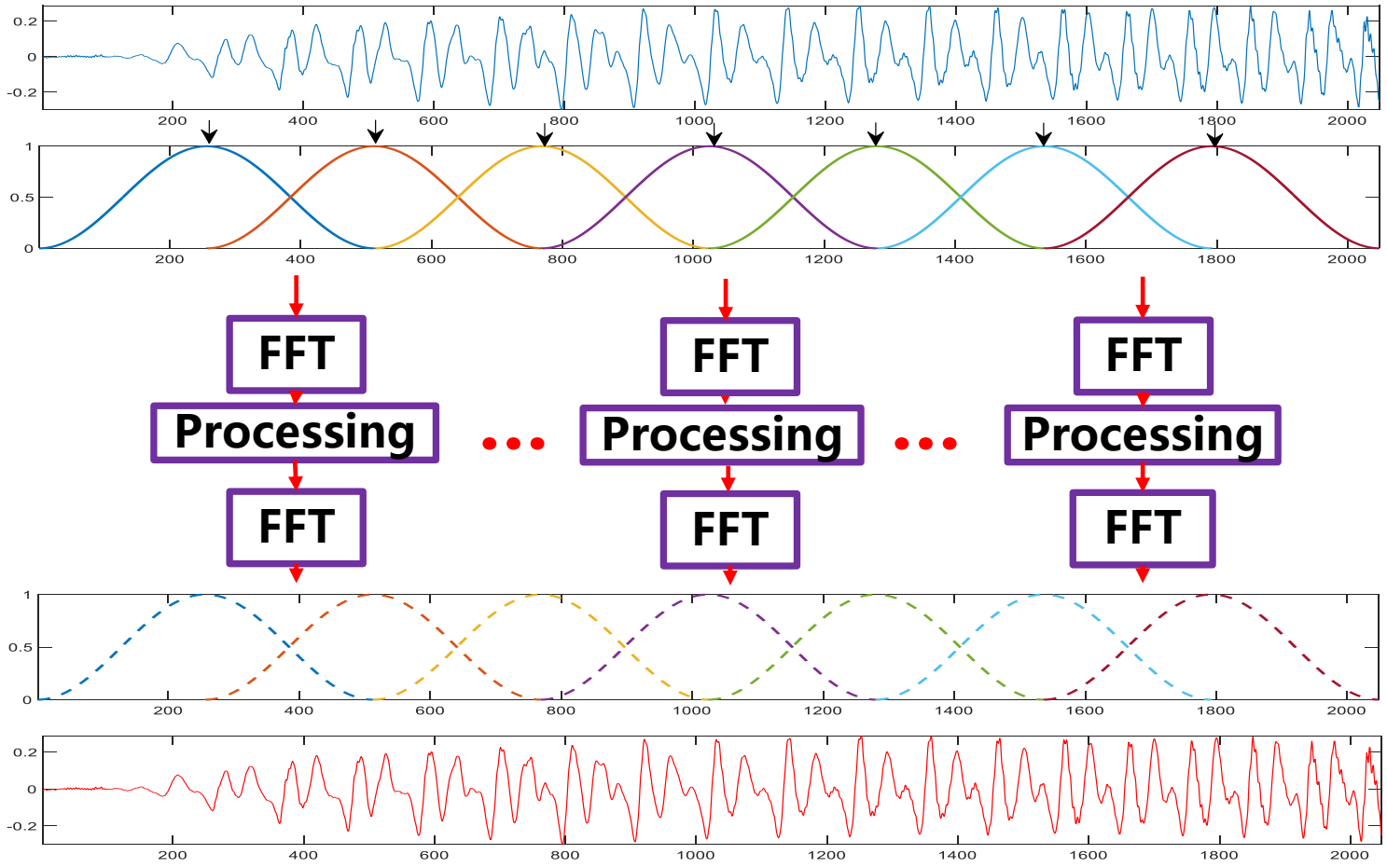
Framing

Window

Transform

Window

Synthesis



Basic Knowledge of Sound

Level of Features

Instrumentation, Key, Chords, Rhythm, Tempo, Lyrics, Genre, Mood

High-level

Pitch, Note onsets, Fluctuation patterns, MFCCs

Mid-level

Amplitude Envelope, Energy, Spectral centroid, Spectral flux

Low-level

Communication Acoustics

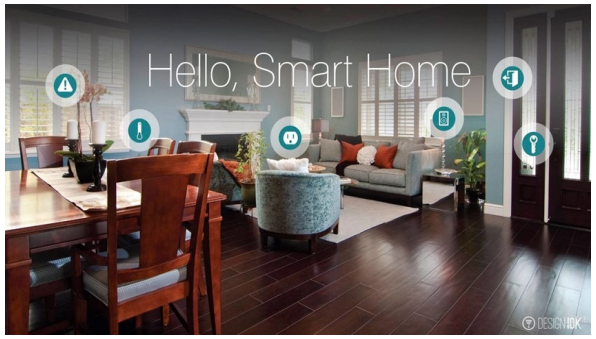
XJTLU | SCHOOL OF FILM AND TV ARTS

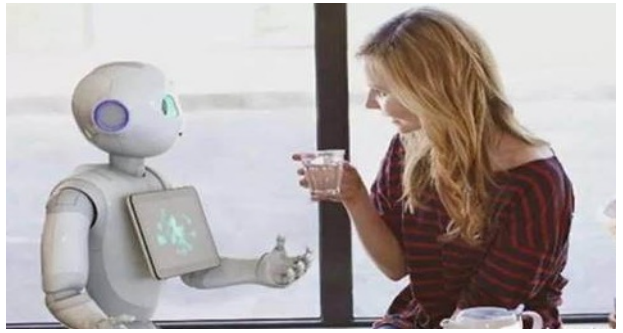
Xi'an Jiaotong-Liverpool University 西交利物浦大學

Communication Acoustics

Concepts and Applications

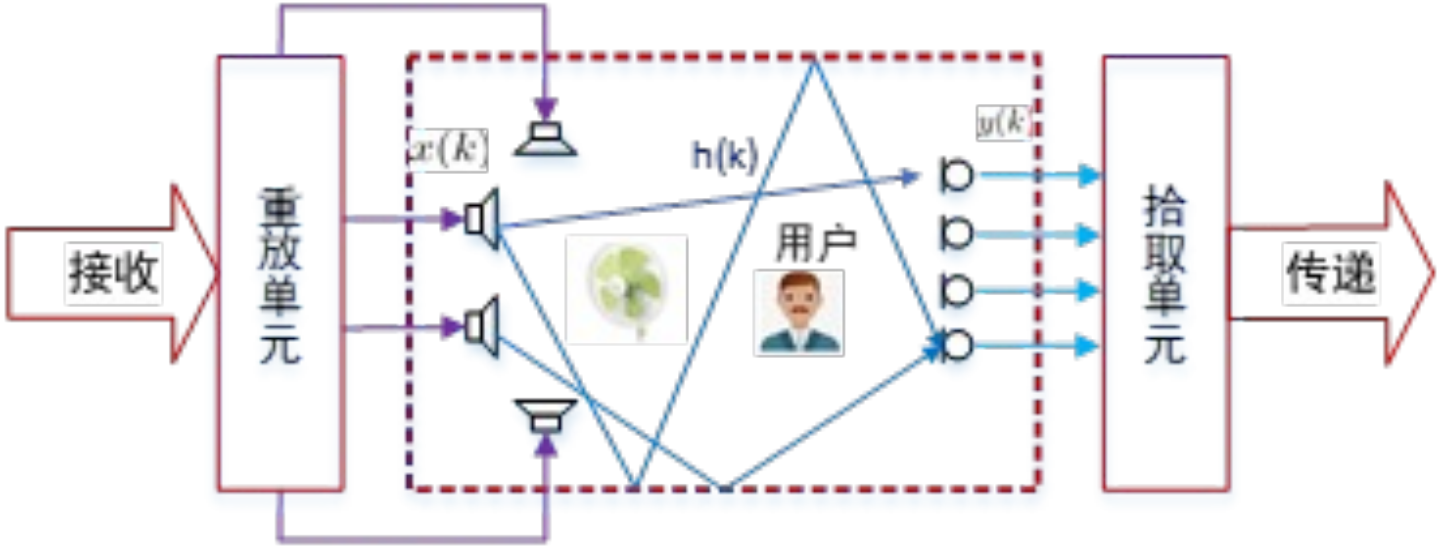
- Studies acoustic problems in communication.
- Focused on the exchange of information between people and people, people and machines, and machines and machines.





Communication Acoustics

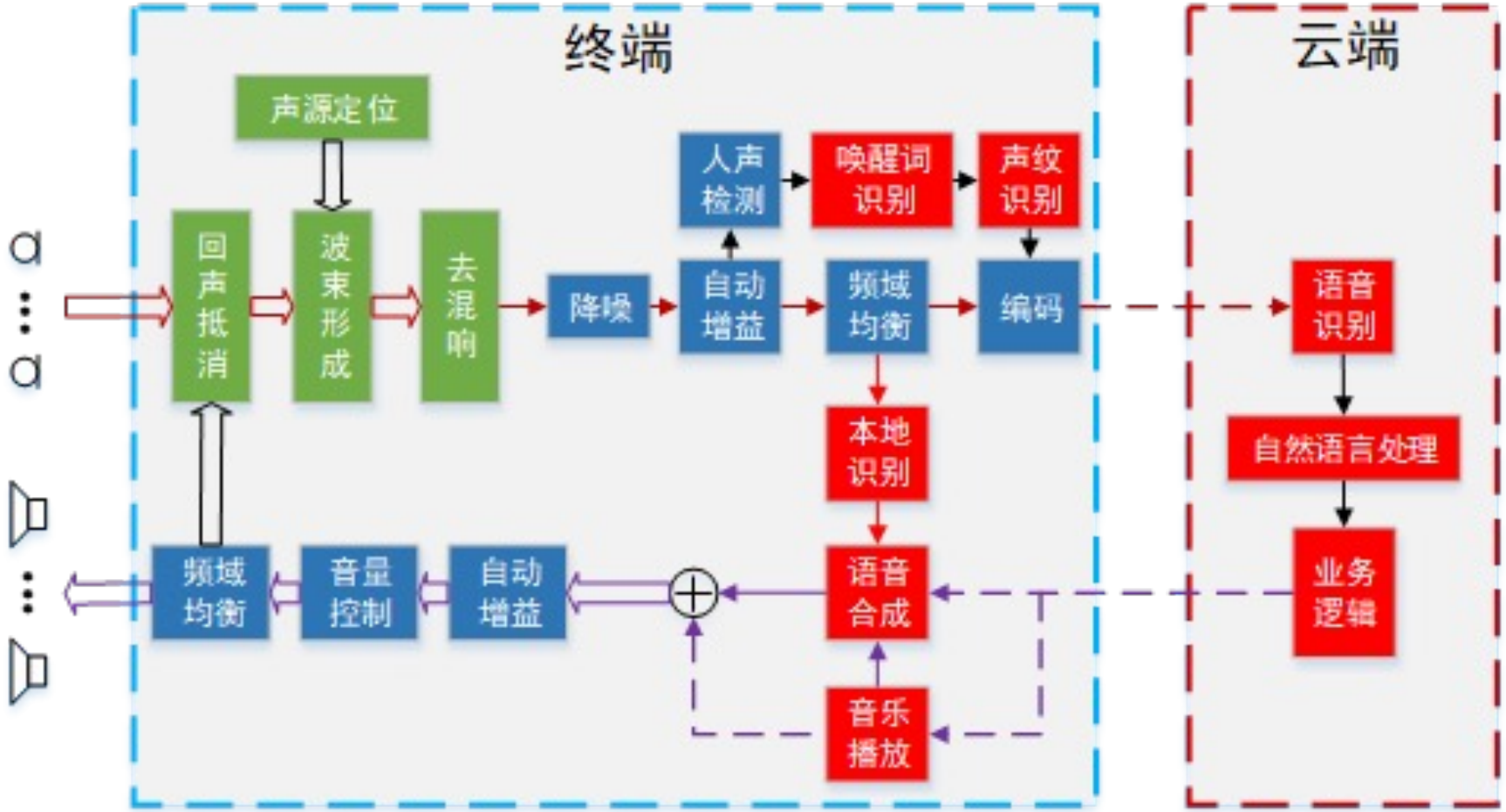
Acoustic Signal Acquire and Process



- Interfering factors：noise, echo and reverberation.
- Noise：Stationary noise and Non-stationary noise.
- Echo：Speaker sound is collected by a microphone.
- Reverberation：Speaker sound is affected by the room.

Communication Acoustics

Acoustic Framework of Smart Speaker



Communication Acoustics

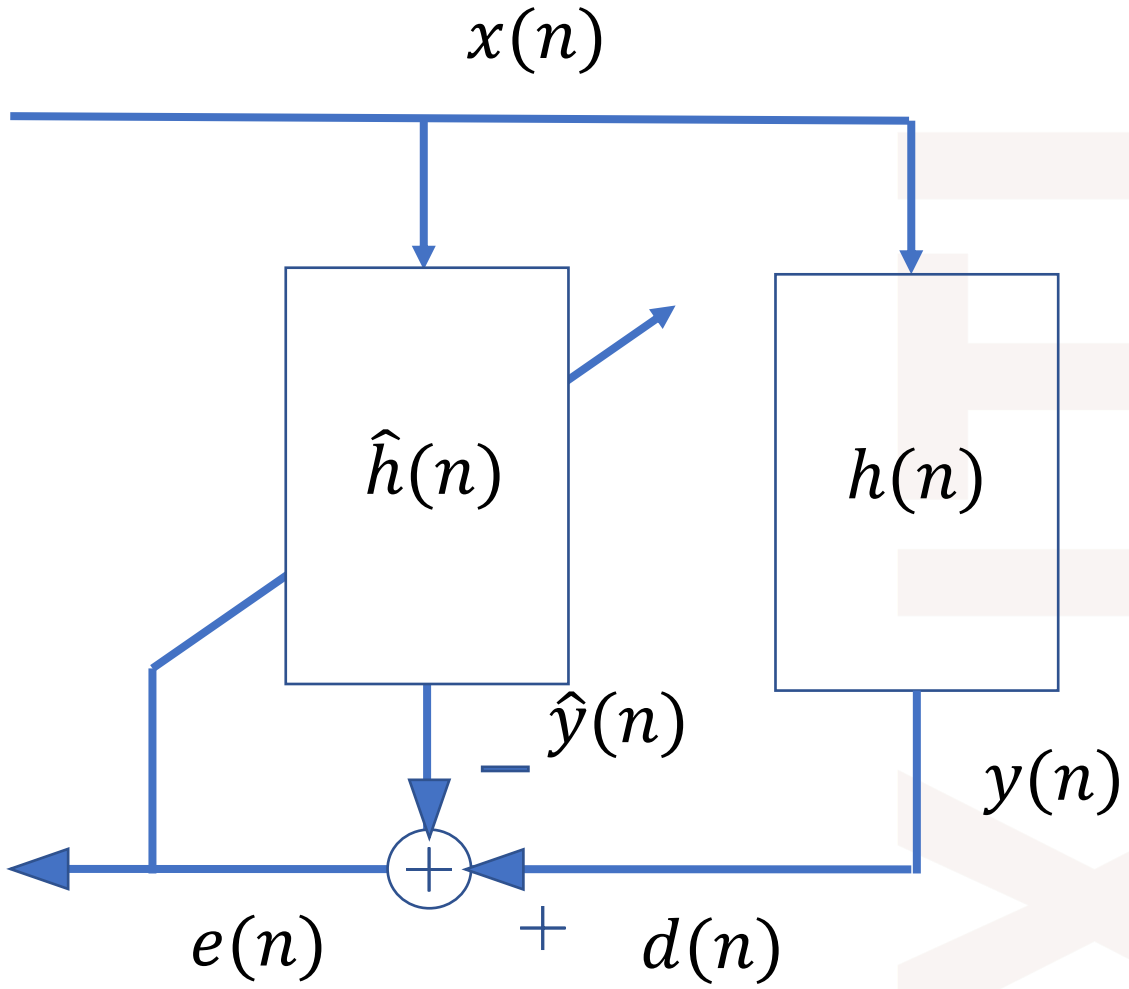
Acoustic Echo Cancellation

- **Echo from remote side:**
  - **The speaker will hear his own voice with delay, which is annoying and confusing.**
  - **The speech from the remote speaker will be affected by the echo signal.**
- **Echo on both sides:**
  - **Worst-case scenario. Feedback can make communication impossible.**

Communication Acoustics

Acoustic Echo Cancellation

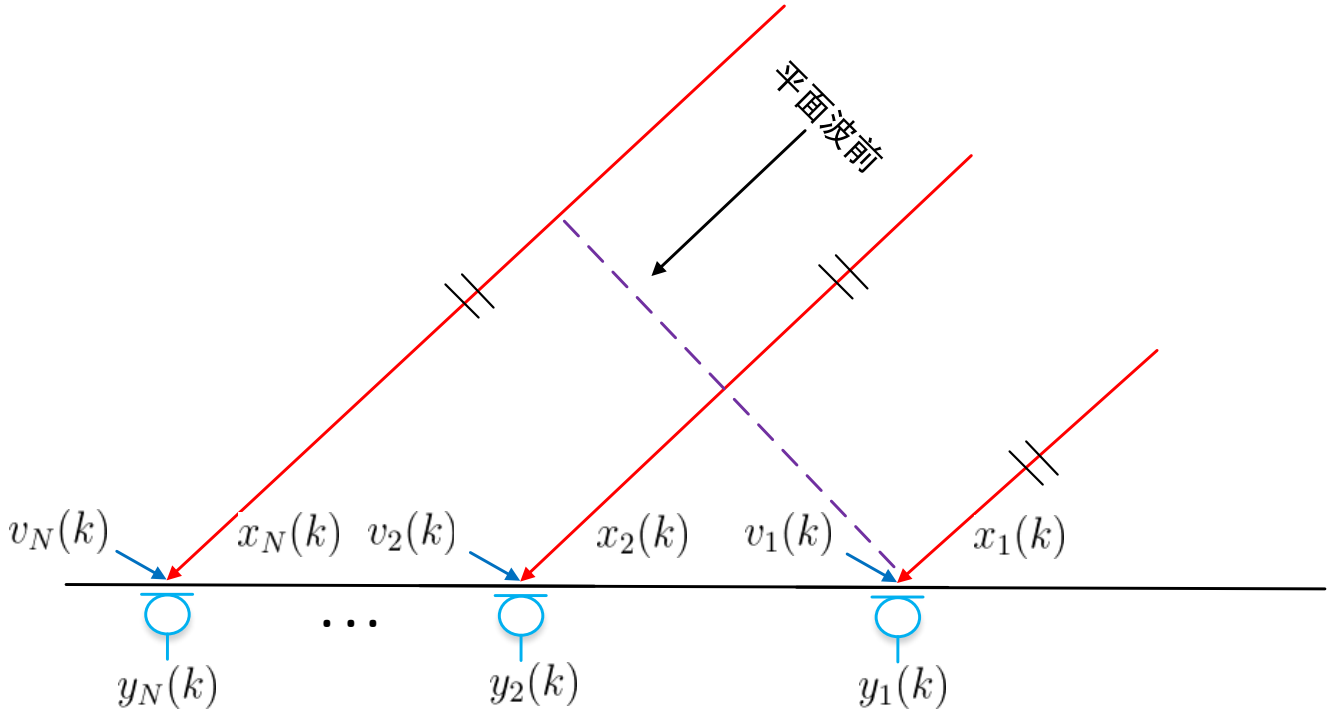
- The effect of the room can be modelled as a FIR filter.
- If we can estimate this filter, we can subtract the echo from the recorded signal before sending.



Communication Acoustics

Source Localization

- **Sound source localization aims to determine the spatial location of a sound source in the environment based on the acoustic signals received by a set of microphones.**

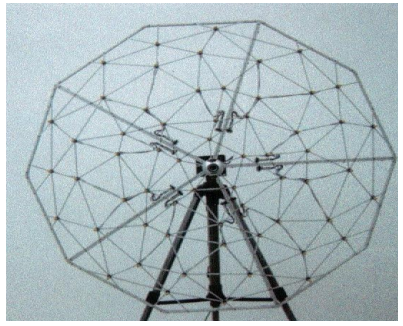


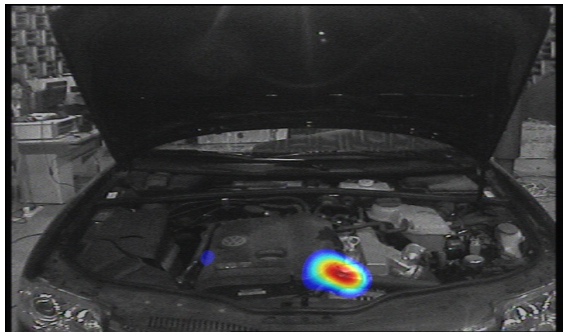
$$x_1(n) = s(n)$$
$$x_2(n) = s(n-D)$$
$$\vdots$$
$$x_N(n) = s(n-(N-1)D)$$
$$D = ?$$

Communication Acoustics

Source Localization - Sound Camera

- Use a microphone array to measure the distribution of sound fields. It can be used to measure the location of sound emitted by an object and the state of sound radiation.





Communication Acoustics

Active Noise Control

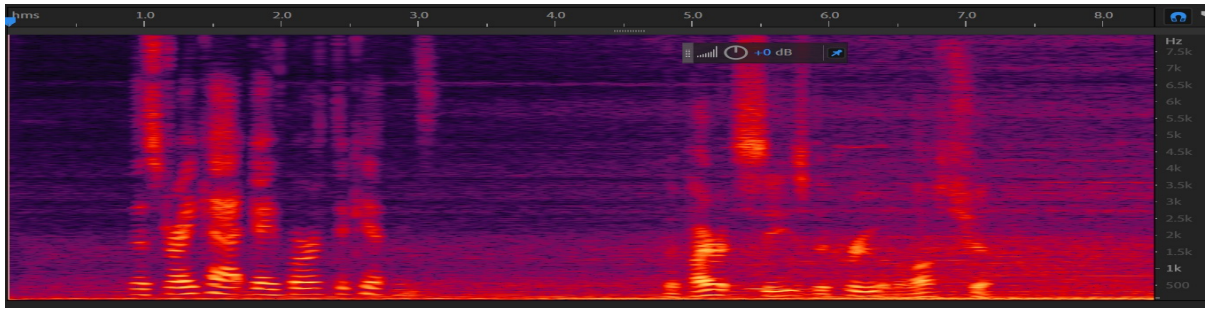
Communication Acoustics

Active Noise Control - Example

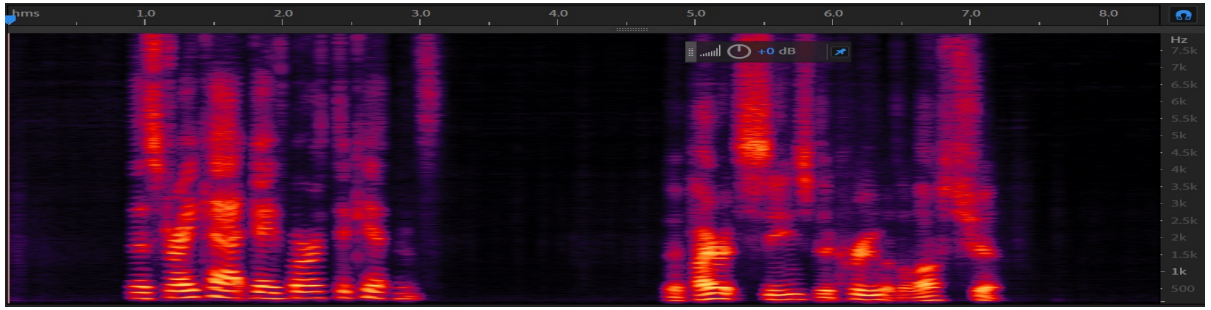
Communication Acoustics

Speech Enhancement

Improve the quality and intelligibility of a speech signal by removing various types of noise interference.



Raw Mic



Proposed

Communication Acoustics

Speech Enhancement – Types of Noise

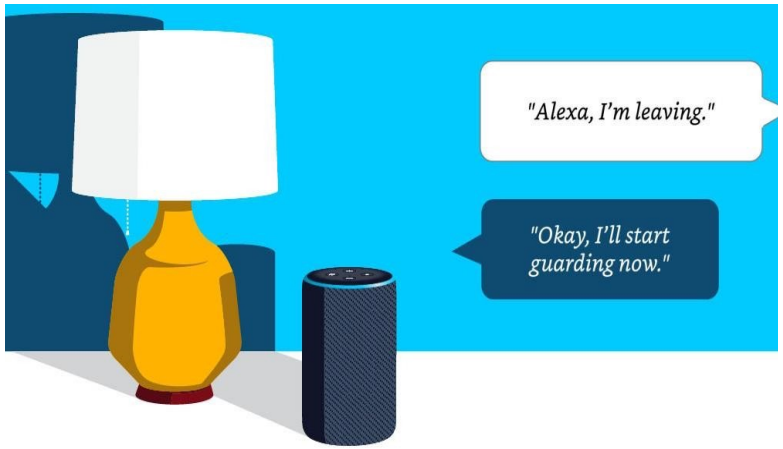
- Stationary noise
- Quasi stationary noise
- Non-stationary noise

Common Types: white, pink, music, babble, bus, car, metro, office, railway, restaurant, street, traffic, workshop, airport, station

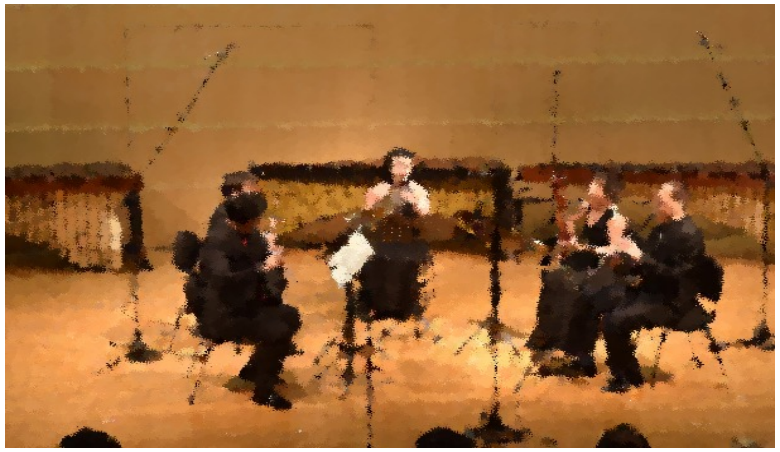
Communication Acoustics

Source Separation

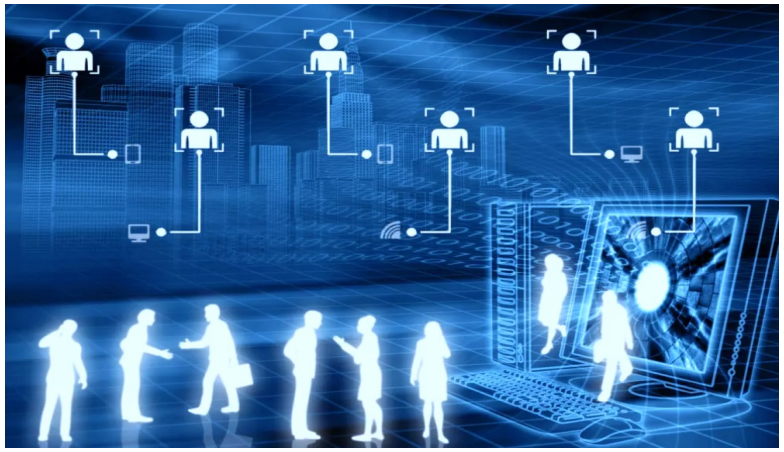
Cocktail Party Problem



Speech Recognition in Noisy Environment



Music Analysis



Security Monitoring

Communication Acoustics

Source Separation - Example

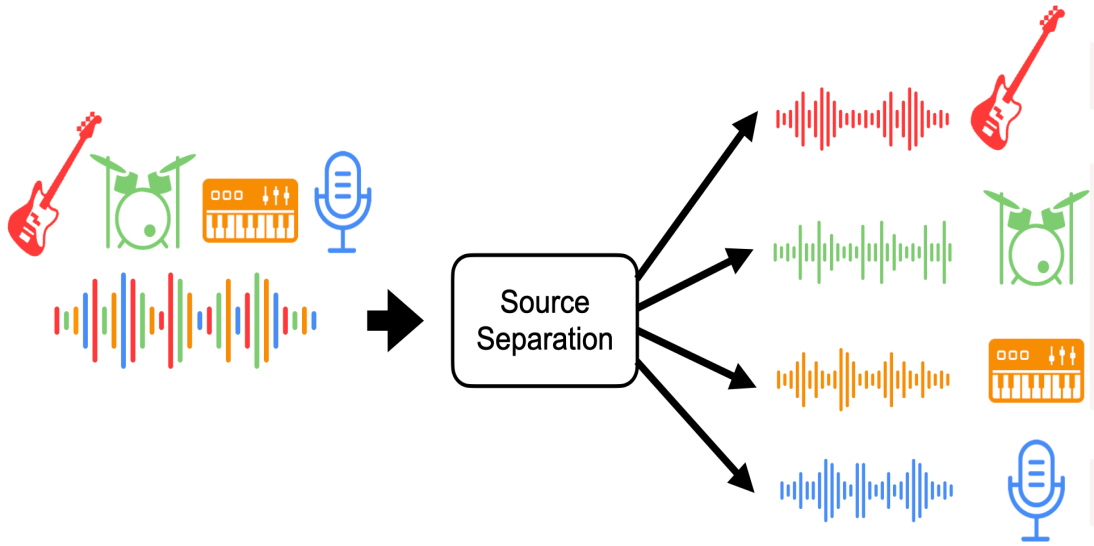
Noisy Speech

Target Speech

Mixture

Est1

Est2



Mixture

Vocal

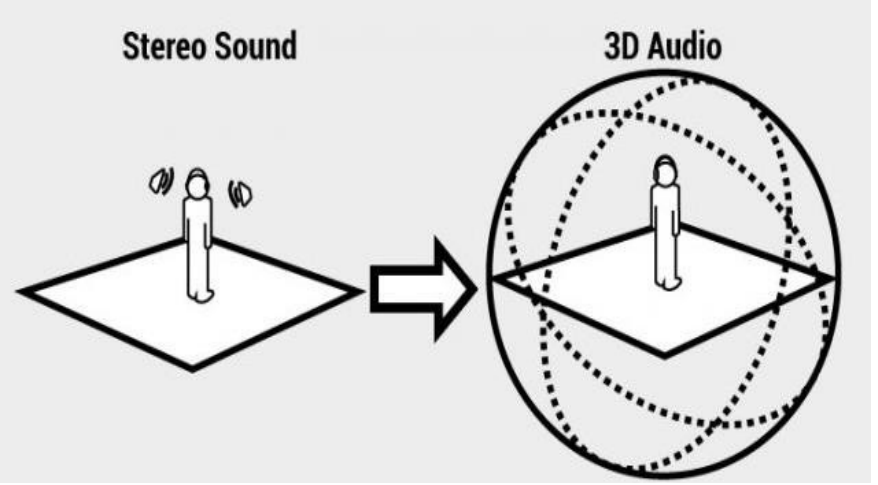
Drum

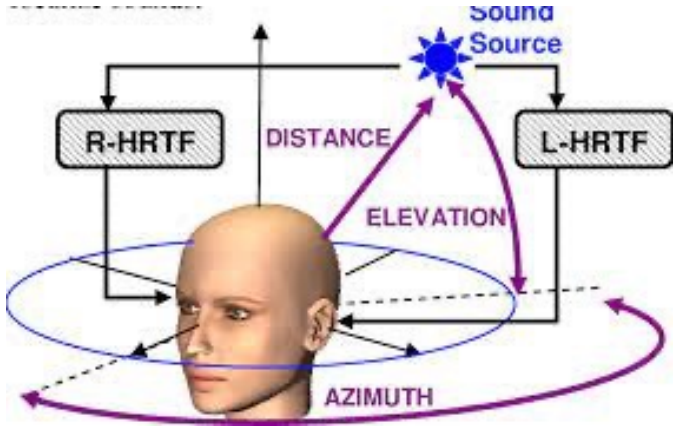
Bass

Others

Communication Acoustics

3D Sound Reproduction





Communication Acoustics

Some References

[1] R. Martin, U. Heute, C. Antweiler, Advances in Digital Speech Transmission, Wiley, 2008.

[2] P. Naylor and D. Nikolay, eds. Speech dereverberation. Springer Science & Business Media, 2010.

[3] B. Michael, and D. Ward, eds. Microphone arrays: signal processing techniques and applications. Springer Science & Business Media, 2001.

[4] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, Advances in Network and Acoustic Echo Cancellation. Springer-Verlag, Berlin, Germany, 2001.

[5] P. Loizou. Speech enhancement: theory and practice. CRC press, 2013.

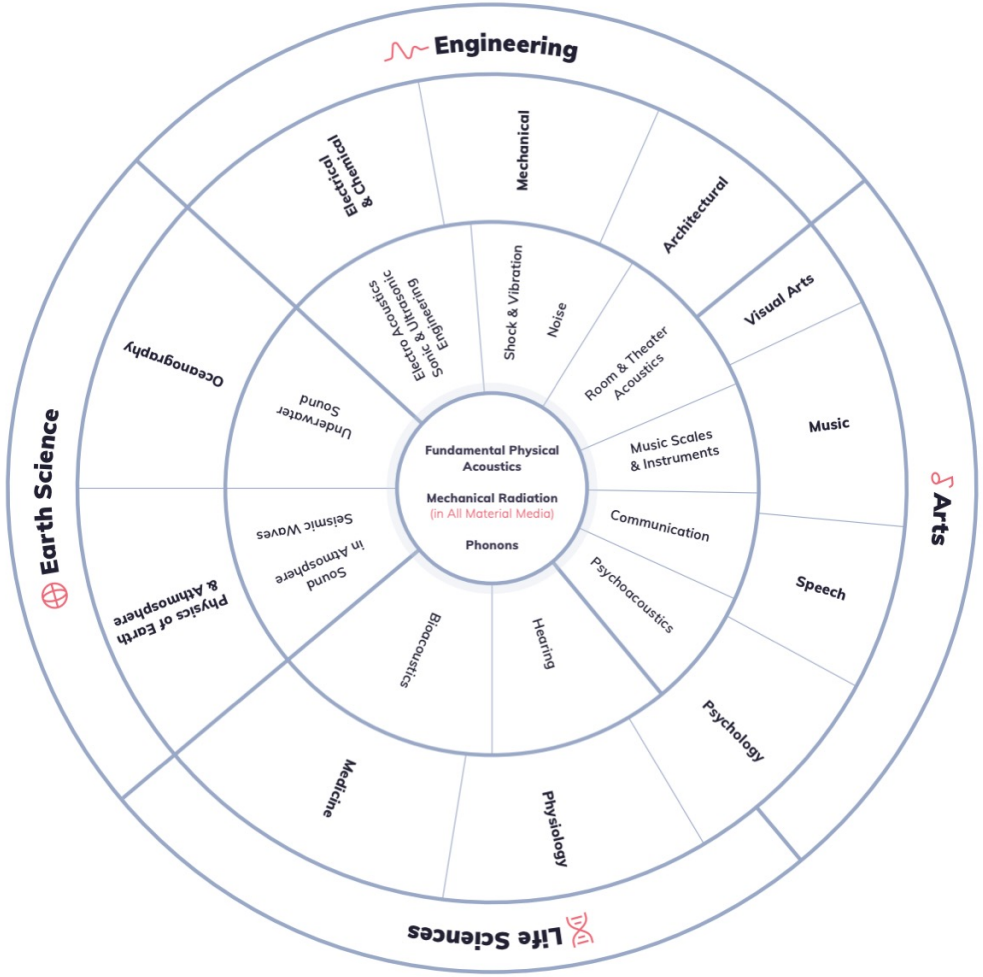
AI for Audio Application

XJTLU | SCHOOL OF FILM AND TV ARTS

Xi'an Jiaotong-Liverpool University 西交利物浦大學

AI for Audio Application

Lindsay's Wheel of Acoustic



- **Audio, speech, and acoustics are collectively referred to as sound.**
- **It is the second largest application of artificial intelligence.**

AI for Audio Application

DCASE Challenge

- Approach: "A rising tide lifts all boats"
- Before 2011: No "home" for everyday sound recognition, little data
- 2012: Plan, collect sounds, "IEEE AASP Challenge", WASPAA slot
- DCASE 2013, Oct 2013 (at WASPAA): 24 submissions
- Now: Series of Challenges and Workshops: http://dcase.community/
- DCASE 2016, Sep 2016 (at EUSIPCO): 82 submissions
- DCASE 2017, Nov 2017, Munich: 200+ submissions
- DCASE 2018, Nov 2018, Surrey, UK: 650+ submissions
- DCASE 2019, Oct 2019, New York: 1000+ submissions

AI for Audio Application

DCASE Challenge

Acoustic Scene Classification

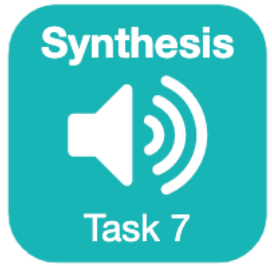
Machine Condition Monitoring

Sound Event Localization and Detection

Weakly Supervised Sound Event Detection

Bioacoustics Event Detection

Audio Captioning

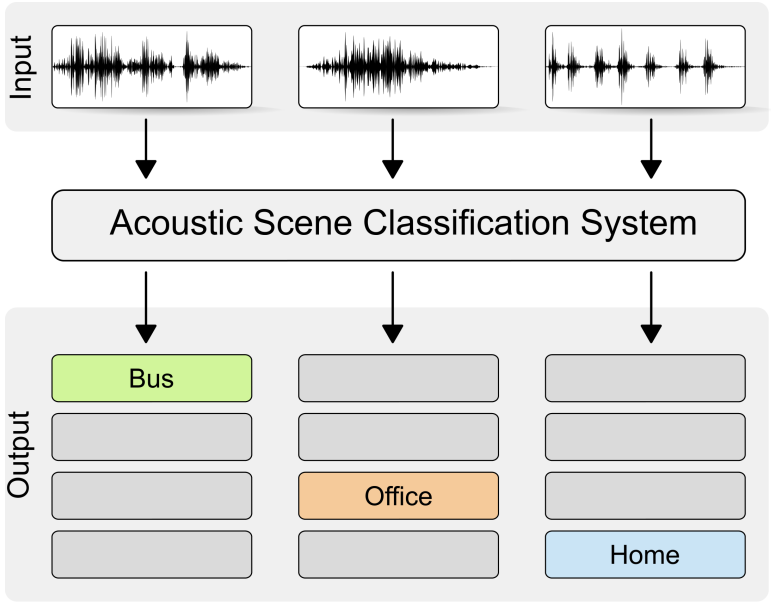


Foley Sound Synthesis

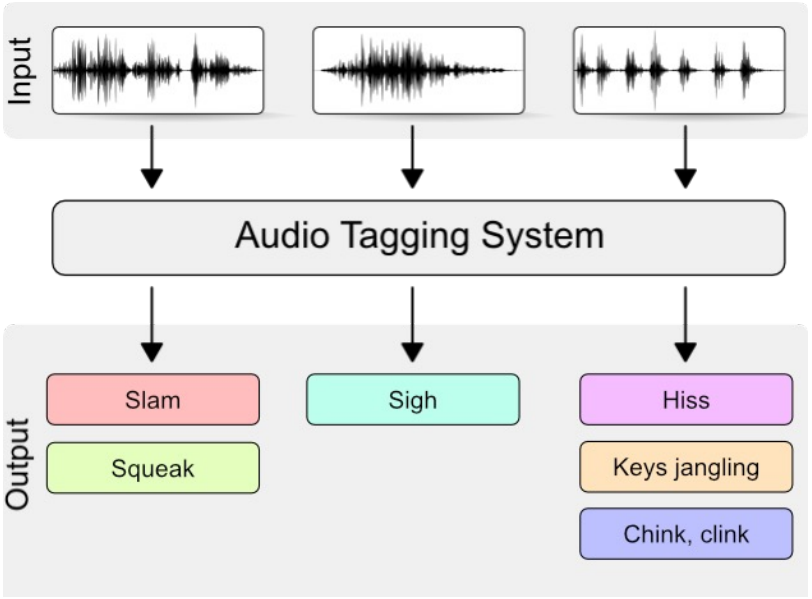
AI for Audio Application

Computational Analysis of Sound Scenes and Events

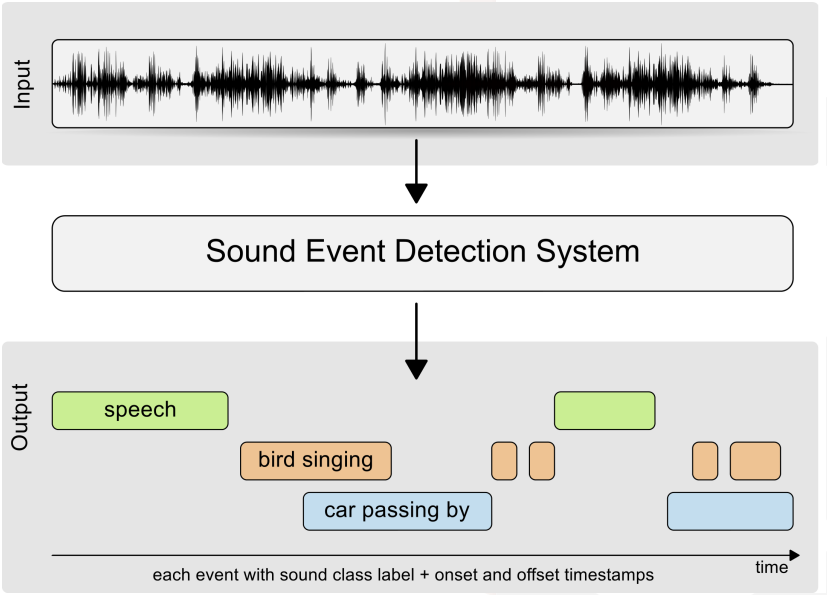
Acoustic Scene Classification



Audio Tagging

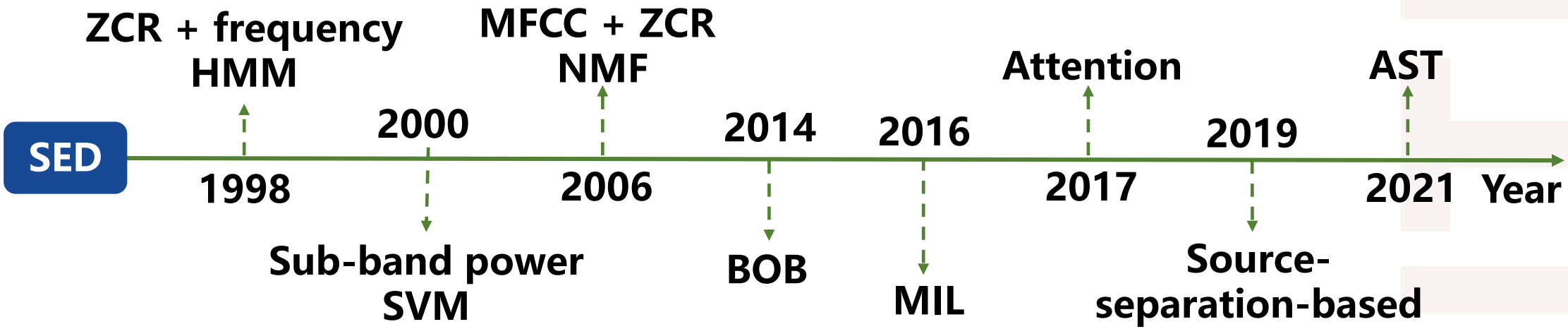


Sound Event Detection



AI for Audio Application

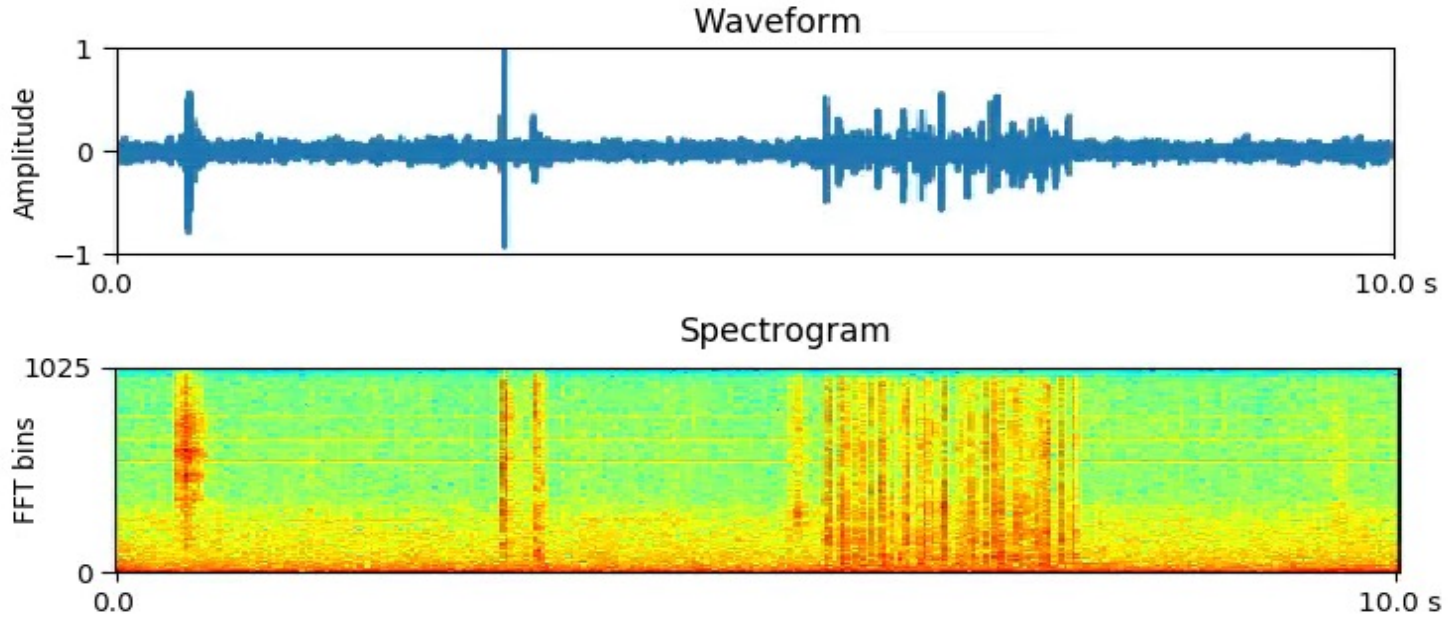
History of SED



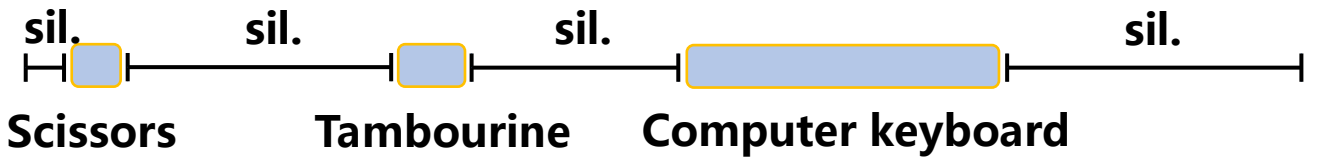
[1] Liu, Zhu, Jincheng Huang, and Yao Wang. "Classification TV programs based on audio information using hidden Markov model." In 1998 IEEE Second Workshop on Multimedia Signal Processing (Cat. No.98EX175), pp. 27-32. IEEE, 1998.
[2] Li, Stan Z., and Guo-dong Guo. "Content-based audio classification and retrieval using SVM learning." In First IEEE Pacific-Rim Conference on Multimedia, Invited Talk, Australia. 2000.
[3] Benetos, Emmanouil, Margarita Kotti, and Constantine Kotropoulos. "Musical instrument classification using non-negative matrix factorization algorithms and subset feature selection." In 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, vol. 5, pp. V-V. IEEE, 2006.
[4] Plinge A, Grzeszick R, Fink G A. A bag-of-features approach to acoustic event detection[C]//2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014: 3704-3708.
[5] Kumar A, Raj B. "Audio event detection using weakly labeled data". Proceedings of the 24th ACM international conference on Multimedia. ACM, 2016: 1038-1047.
[6] Xu Y, Kong Q, Wang W, et al. Large-scale weakly supervised audio classification using gated convolutional neural network[C]//2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2018: 121-125.
[7] Kong Q, Xu Y, Sobieraj I, et al. Sound event detection and time–frequency segmentation from weakly labelled data[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2019, 27(4): 777-787.

AI for Audio Application

Strong Labeled Data is Costly to Acquire



The goal of Sound Event Detection(SED):



→ **Strongly labeled data**

The goal of Audio Tagging(AT):

Computer keyboard, Tambourine, Scissors

→ **Weakly labeled data**

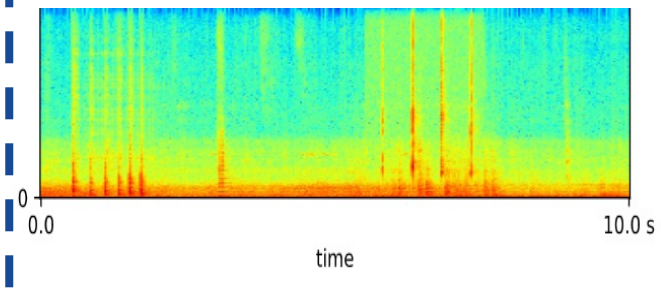
AI for Audio Application

A Large Amount of Weakly Labeled Data is Available

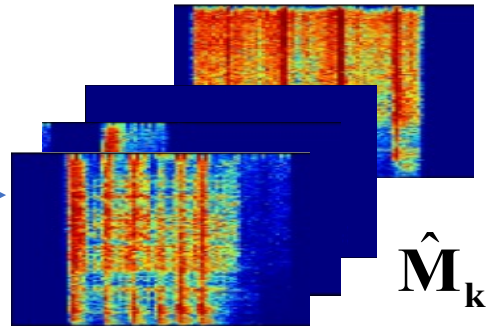
AI for Audio Application

SED Based on Source Separation Framework

$\mathbf{X} \in \mathbb{R}_{+}^{T \times F}$



G1



$\hat{\mathbf{M}}_k$

G2

Prediction $\hat{y}_k$	Label y_k
0.8	1
0.2	0
0.6	1

$\hat{\mathbf{M}} = [\hat{M}_k(t,f)] \in \mathbb{R}_{+}^{K \times T \times F}$

$\hat{\mathbf{y}} = [\hat{y}_k]^{\mathrm{T}} \in \mathbb{R}_{+}^{1 \times K}$

Sound event detection

$$\hat{y}_k(t) = \sum_{f=1}^{F} w_k(f) \cdot \hat{M}_k(t,f),$$

Source Separation

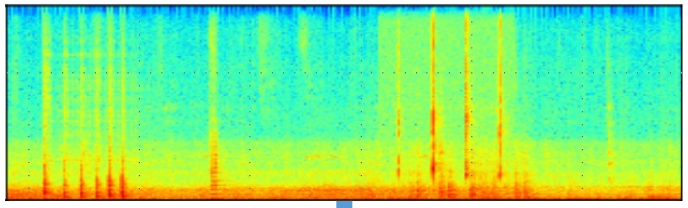
$$\hat{x}_k(n) = IFFT(\hat{\mathbf{M}}_k(t,f) \mathrm{e}\ |\mathbf{X}(t,f)| \cdot e^{j\angle \mathbf{X}(t,f)})$$

G1: Neural network $\mathbf{X} \rightarrow \hat{\mathbf{M}}$

G2: Pooling function $\hat{\mathbf{M}} \rightarrow \hat{\mathbf{y}}$

AI for Audio Application

A Multi-task Learning (MTL) Method for SED



$1 \times 64 \times 311$

Conv(3,3)

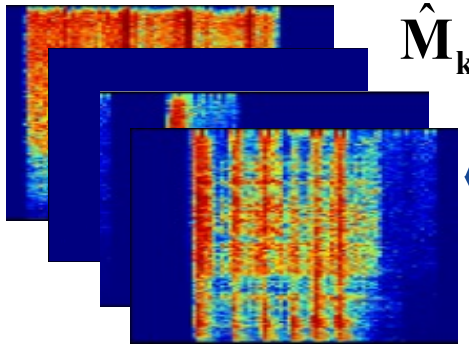
Depth Conv(3,3)&Atrous(2)

Point Conv(1,1)

Conv(1,1)

Estimated T-F Masks

$K \times 64 \times 311$



$\hat{\mathbf{M}}_k$

Global Pooling Function

Main task: SED

Prediction $\hat{y}_k$	Label y_k
0.8	1
0.2	0
0.6	1

Cross Entropy $L_{sed}(\theta)$

Overlap Measure

Auxiliary task: SS

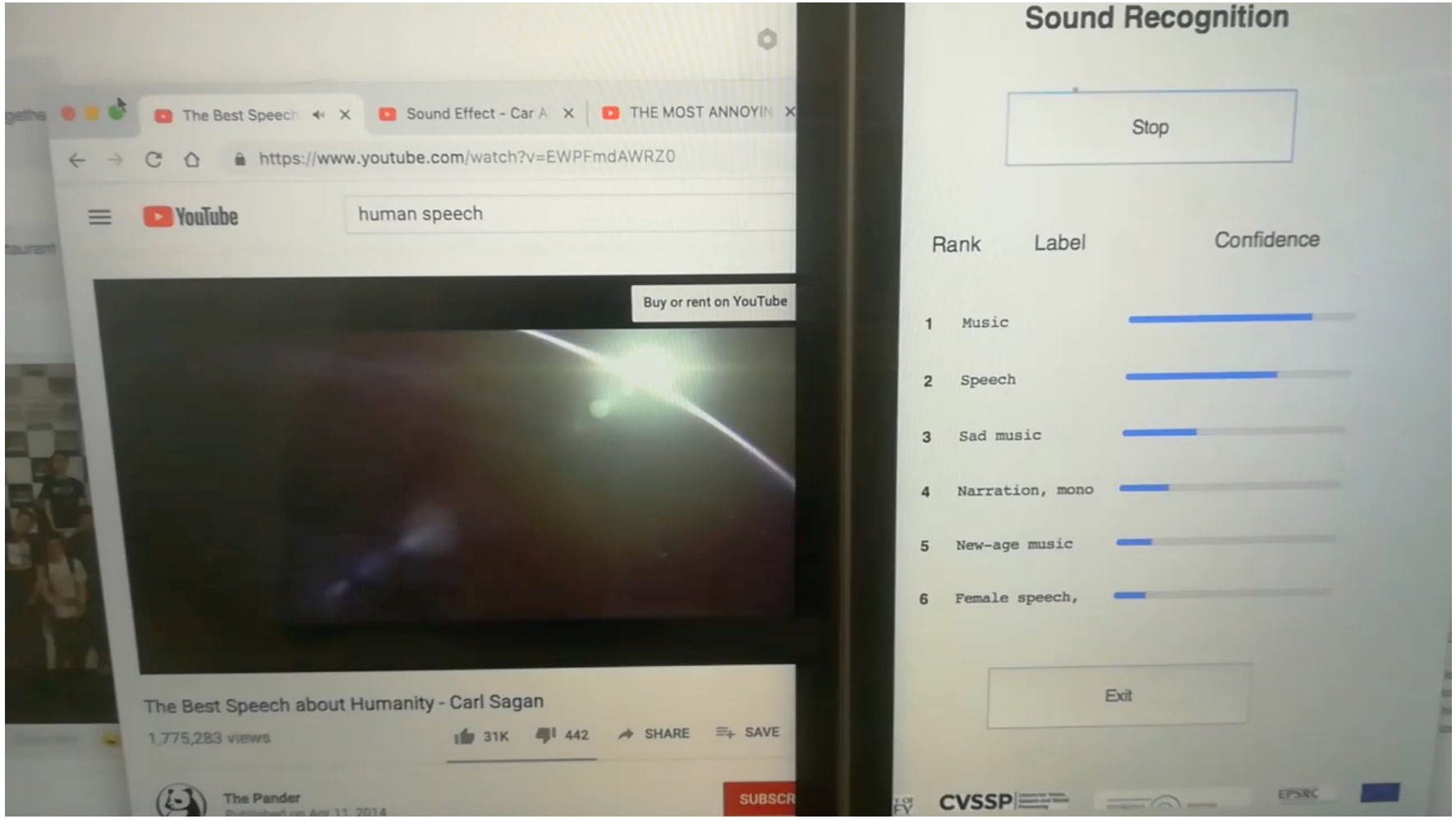
Prediction $\hat{Overlap}$	Label $Overlap$
0.2	0

Segment Loss $L_{ss}(\theta)$

$$L(\theta) = \lambda_1 L_{sed}(\theta_{share}, \theta_{sed}) + \lambda_2 L_{ss}(\theta_{share}, \theta_{ss})$$

AI for Audio Application

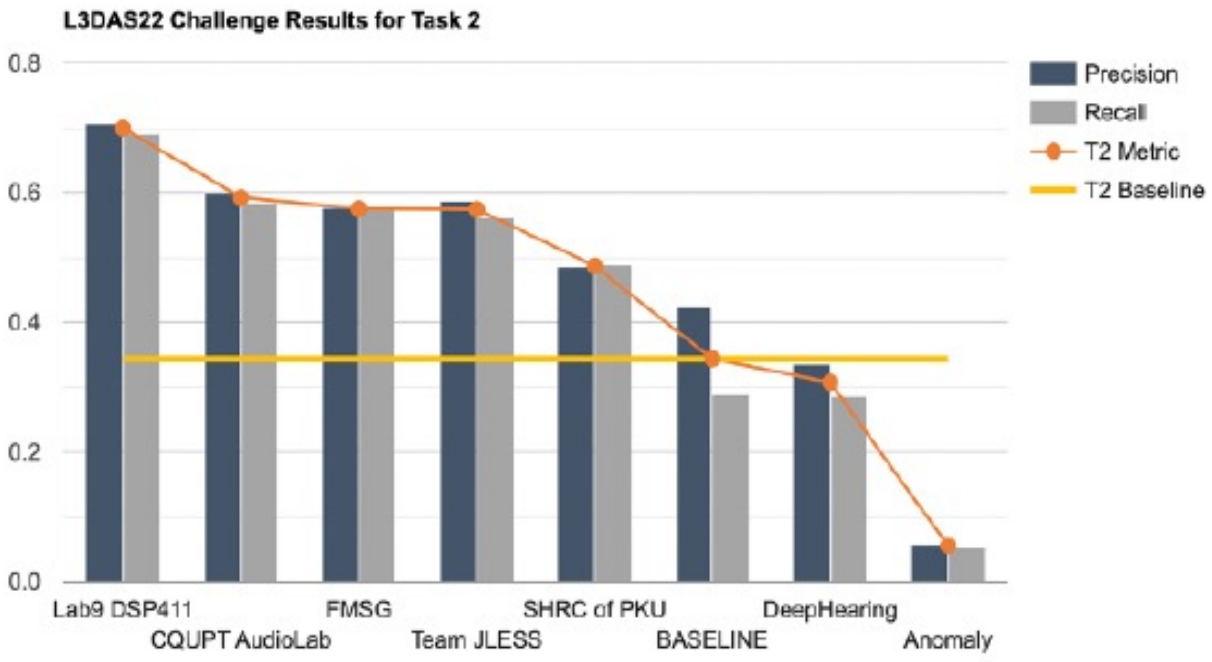
Sound Event Detection - Example

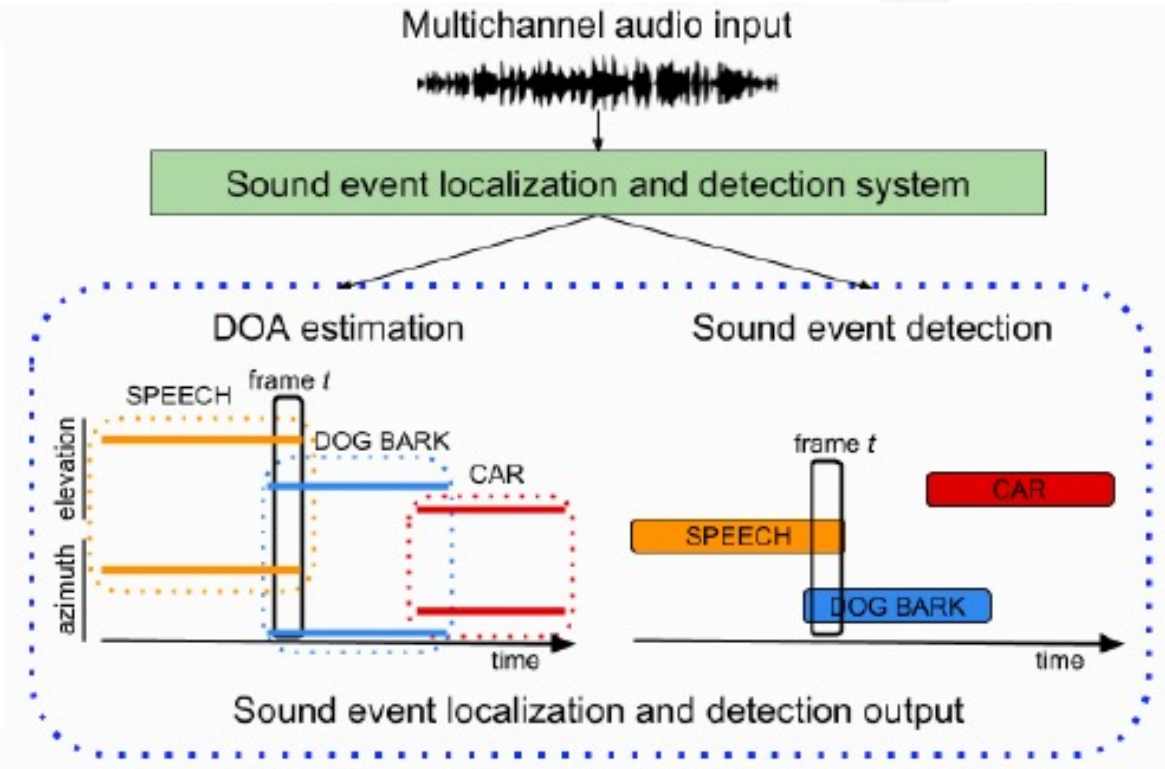


AI for Audio Application

Sound Event Localization and Detection

Detect the category, start and end time of ongoing acoustic events, and locates their spatial position and direction.

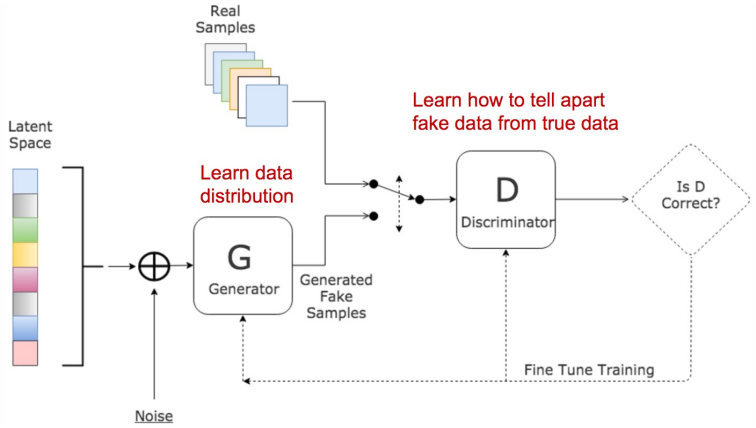




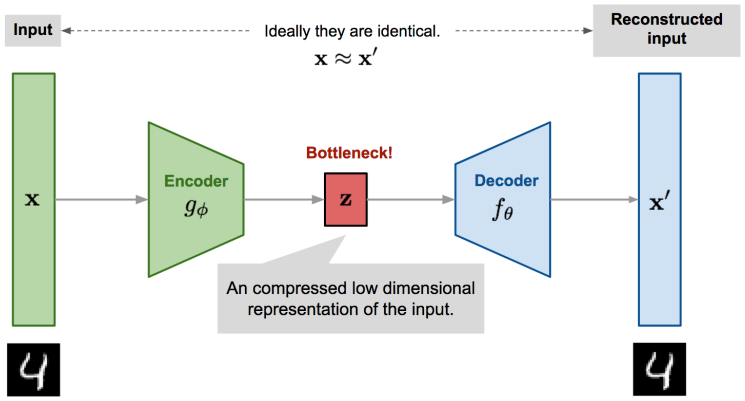
AI for Audio Application

Generative Models

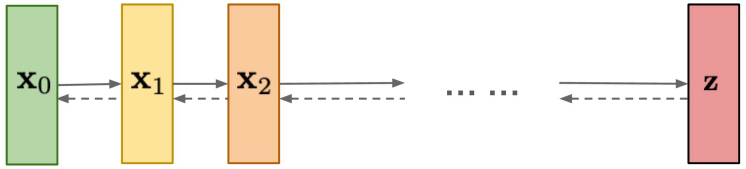
对抗生成网络（GANs）



自编码器与变分自编码器 (AE, VAE)



扩散模型 (Diffusion)



Diffusion models:
Gradually add Gaussian noise and then reverse

AI for Audio Application

Generative Models - Example

THANK YOU

VISIT US

WWW.XJTLU.EDU.CN

FOLLOW US

@XJTLU

Xi'an Jiaotong-Liverpool University
西交利物浦大学

XJTLU | SCHOOL OF FILM AND TV ARTS