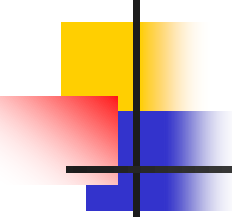




第六章 语音识别-关键技术

- HMM训练
- Viterbi解码



三个基本问题

(1) 已知模型 $\lambda = f(A, B, \pi)$, 计算指定特征矢量序列 $\mathbf{Y}$ 的出现概率 $P_r(\mathbf{Y} | \lambda)$ 。 (评价问题)

"前向-后向"算法

(2) 已知特征矢量序列 $\mathbf{Y}$ 及模型 $\lambda = f(A, B, \pi)$, 计算最大似然状态序列 $\mathbf{X}$ 。 (识别问题)

Viterbi算法

(3) 获得若干 $\mathbf{Y}$ 后, 如何修正 A 、 B 和 π , 以使 $\mathbf{Y}$ 出现概率最大。 (学习问题)

Baum - Welch算法



HMM训练

- 问题：如何获得“好”的模型
 - 评价标准：最大似然、最大区分
 - 方法论：数据驱动
 - 算法实现：**B**矩阵的参数估计

B矩阵的参数估计

- 状态观测概率矩阵

$$B = [b_i(o_t)]$$

$$b_i(o_t) = P_r[y(t) = o_t \mid x(t) = s_i]$$

- 刻画的是状态 s_i 所对应声学特征 o_t 的概率分布

- 随机分布，且非典型函数分布

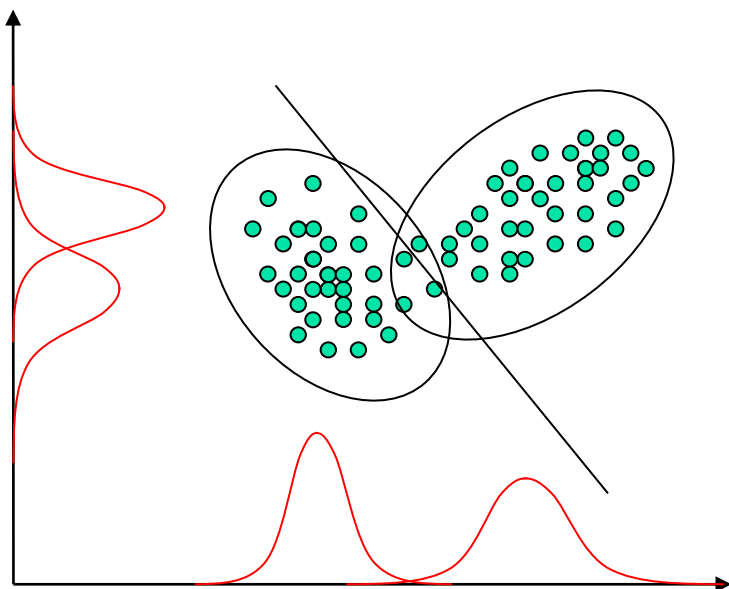
- GMM(Gaussian Mixed Model)

$$P(o_t) = \sum_{i=1}^M w_i N(o_t, \mu_i, \sigma_i)$$

- 原则上，增加混合高斯数目，
可无限逼近任意概率分布

- HMM-GMM

- B矩阵在识别中起关键性作用





基本算法

- 识别模型的训练

- 聚类：类别与语音特征间的关系
- 最大似然：概率测度下的聚类及概率参数估计

K-means

EM



K-means

- 聚类分析：依照某种距离测度，实现观察样本的分类
- K-means，一种动态聚类法

Step1: K个聚类中心初始值， $z_1(1)$, $z_2(1)$, ..., $z_k(1)$

Step2: 样本集 $\{x\}$ 依照距离最小原则分配到K个聚类中心的某个中心j

$$j = \arg \min_i D(x, z_i(k)), \quad i = 1, 2, \dots, K$$

Step3: 利用分类结果重新计算聚类中心

$$z_j(k+1) = \frac{1}{N} \sum_{x \in S_j} x, \quad j = 1, 2, \dots, K$$

Step4: 迭代结束判决， $z_j(k+1)$ vs. $z_j(k)$

- 问题，结果非全局最优
 - K值选取
 - 初识值选取



Expectation Maximum

- 最大似然分析：对于观察样本集 X ，并已知概率模型的当前估计 λ^k ，有新的估计 λ^{k+1} ，使得

$$p(X | \lambda^{k+1}) \geq p(X | \lambda^k)$$

- 期望最大化： λ^{k+1} 为未知，估计 λ^{k+1} 使对数似然函数 $\log[p(X | \lambda^{k+1})]$ 的期望最大

$$E(\log(p(X | \lambda^{k+1}))) = \sum_{i=1}^K p(X, S_i | \lambda^k) \log(p(X, S_i | \lambda^{k+1}))$$

- GMM参数 (w, u, v) 计算

GMM的解释

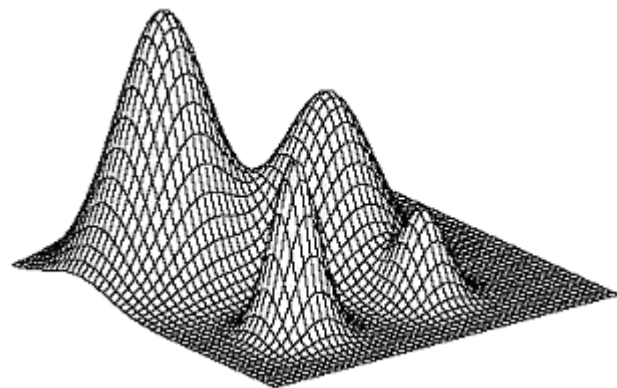
- 对应于某个状态的特征向量分布

$$p_i(\vec{x}) = \frac{1}{(2\pi)^{D/2} \left| \sum_i \right|^{1/2}} e^{-(1/2)(\vec{x}-\mu_i)^t \sum_i^{-1} (\vec{x}-\mu_i)}$$

$$p(\vec{x} | \lambda) = \sum_{i=1}^M w_i p_i(\vec{x})$$

$$\lambda = (w_i, \mu_i, \sum_i^{-1}), i = (1, \dots, M)$$

- 对应于概率分布
 - 表示概率密度函数
 - 足够的混合分量,可逼近任意分布



step1: K-means得到初始GMM估计（硬分类）

GMM参数计算

$Num(X)$ （硬分类）计次函数

step2: 分类概率 / 分类概率累计（软分类）

- c_{kt} 软分类 - 软计次
- c_k 累积软计次

step3: 重新计算权重 w^k 、均值 u^k 、方差 v^k

step4: 迭代结束判决, u^k vs. u^{k+1}

$$\hat{w}_k = \frac{Num(X \in S_k)}{Num(X)}$$

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{X \in S_k} X$$

$$\hat{r}_k^{-1} = \sum_{X \in S_k} (X - \hat{\mu}_k)(X - \hat{\mu}_k)'$$

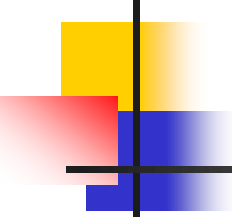
$$c_{kt} = p(k|x_t, \lambda) = \frac{w_k N(x_t|S_k)}{\sum_{l=1}^K w_l N(x_t|S_l)}$$

$$c_k = \sum_{t=1}^T c_{kt}$$

$$\hat{w}_k = \frac{c_k}{T}$$

$$\hat{\mu}_k = \frac{\sum_{t=1}^T c_{kt} x_t}{c_k}$$

$$\hat{r}_k^{-1} = \frac{\sum_{t=1}^T c_{kt} (x_t - \hat{\mu}_k)(x_t - \hat{\mu}_k)'}{c_k}$$



Baum-Welch算法

- step1: 重新估计每个状态 j 的模型参数

$$\hat{\mu}_j = \frac{\sum_{t=1}^T L_j(t) \mathbf{o}_t}{\sum_{t=1}^T L_j(t)}$$

$$\hat{\Sigma}_j = \frac{\sum_{t=1}^T L_j(t) (\mathbf{o}_t - \mu_j)(\mathbf{o}_t - \mu_j)^T}{\sum_{t=1}^T L_j(t)}$$

- step2: 计算 t 时刻对于模型 M 处于状态 j (状态集 L 种状态中的一种)的概率

$$\begin{aligned} L_j(t) &= P(x(t) = j | \mathbf{O}, M) \\ &= \frac{P(\mathbf{O}, x(t) = j | M)}{P(\mathbf{O} | M)} \end{aligned}$$

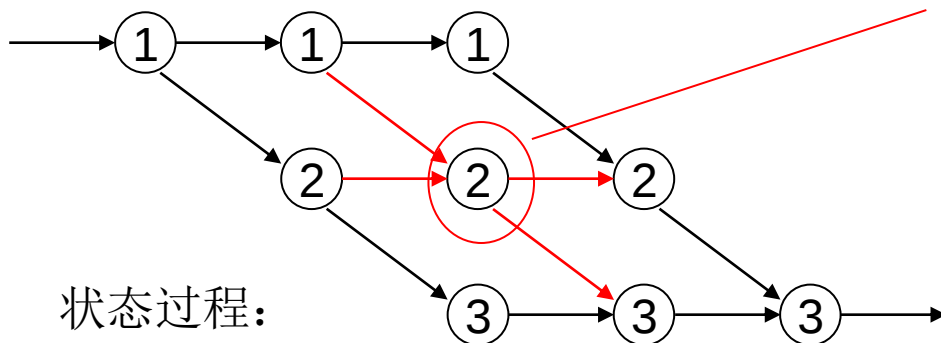
$$= \frac{1}{P} \alpha_j(t) \beta_j(t)$$

$$P = P(\mathbf{O} | M)$$

- step3: 利用新的状态概率, 重新计算模型参数
- step4: 迭代结束判决, $P(\mathbf{O} | M)$ 是否增大, 是则step1

多状态路径

特征向量: y_1 y_2 y_3 y_4 y_5



$$L_j(t) = \frac{1}{P} \alpha_j(t) \beta_j(t)$$

■ 状态路径

- 多个可能的模型
- 路径数目估计 N^T

可能的状态序列

s1,s1,s1,s2,s3; s1,s1,s2,s2,s3
s1,s1,s2,s3,s3; s1,s2,s2,s2,s3
s1,s2,s2,s3,s3; s1,s2,s3,s3,s3



Forward-Backward算法

- 模型 M 对应于特征序列 O , 某个时刻 t 处在状态 j 的概率

$$\alpha_j(t)\beta_j(t) = P(O, x(t) = j|M).$$

- 前向概率 $\alpha_j(t) = P(o_1, \dots, o_t, x(t) = j|M).$

$$\alpha_j(t) = \left[\sum_{i=2}^{N-1} \alpha_i(t-1)a_{ij} \right] b_j(o_t)$$
$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T)a_{iN}. \quad P(O|M) = \alpha_N(T).$$

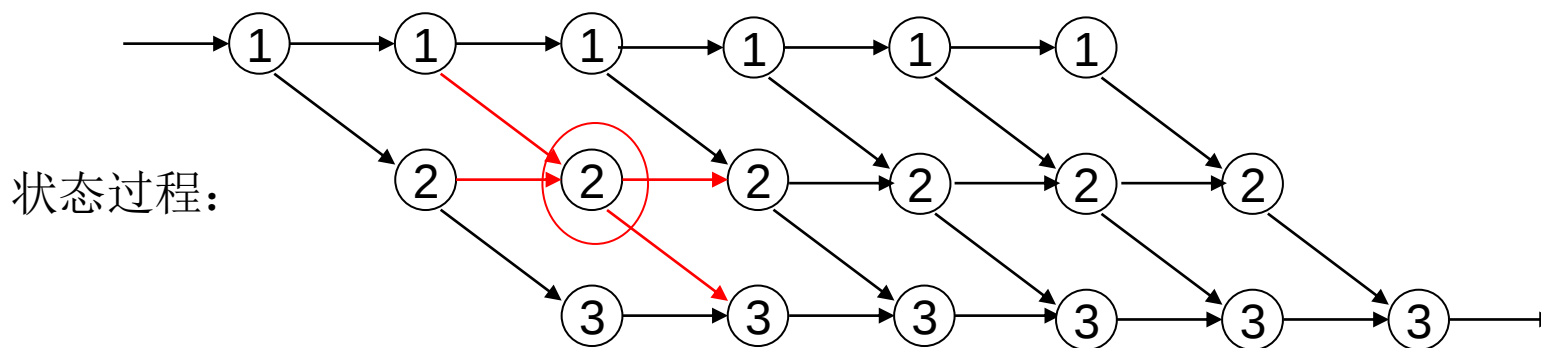
- 后向概率 $\beta_j(t) = P(o_{t+1}, \dots, o_T | x(t) = j, M).$

$$\beta_i(t) = \sum_{j=2}^{N-1} a_{ij}b_j(o_{t+1})\beta_j(t+1)$$
$$\beta_1(1) = \sum_{j=2}^{N-1} a_{1j}b_j(o_1)\beta_j(1).$$

Forward-Backward算法

- $P(O|M)$ 计算量, N 为状态数, T 为特征序列长度
 - 原为 $O(N^T)$
 - 前-后向算法 $O(N^2T)$
 - 与特征序列长度关系: 指数 vs 线性

特征序列: y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8





模型训练过程

- **Step1: 训练数据时间标注 (state一级)**
 - 标注信息: 注音/模型符号序列
 - 算法: Viterbi
 - 模型: 初始模型 λ^0
- **Step2: 模型 λ^1 参数估计**
 - 标注信息: state时间标注
 - 算法: EM
- **Step3: 模型 λ^n 参数估计**
 - 标注信息: 注音/模型符号序列
 - 算法: Baum-Welch
 - 模型: 模型 λ^{n-1}
 - 迭代3-4次



第六章 语音识别-关键技术

- HMM训练
- Viterbi解码



Viterbi搜索

- HMM模型 – Viterbi Search Algorithm
 - 状态序列 X 对观察特征向量序列 Y 的刻画
 - 训练：音段边界标注
 - 识别：比较各模型最优状态序列下的概率



Viterbi搜索

- HMM模型 – Viterbi Search Algorithm

- 状态序列 X 对观察特征向量序列 Y 的刻画
- 概率计算/最优状态路径

$$P(X, Y \mid \lambda) = P(y_1, y_2, \dots, y_N, x_1, x_2, \dots, x_N \mid \pi, A, B)$$

$$\delta_n(i) = \max_{x_1, x_2, \dots, x_{n-1}} P(y_1, y_2, \dots, y_n, x_1, x_2, \dots, x_n = S_i \mid \pi, A, B) \quad (i = 1, 2, \dots, L)$$

Viterbi搜索

声学特征序列

y_{n-1}

y_n

y_{n+1}

y_N

状态序列

s_1

s_2

s_i

s_L

s_j

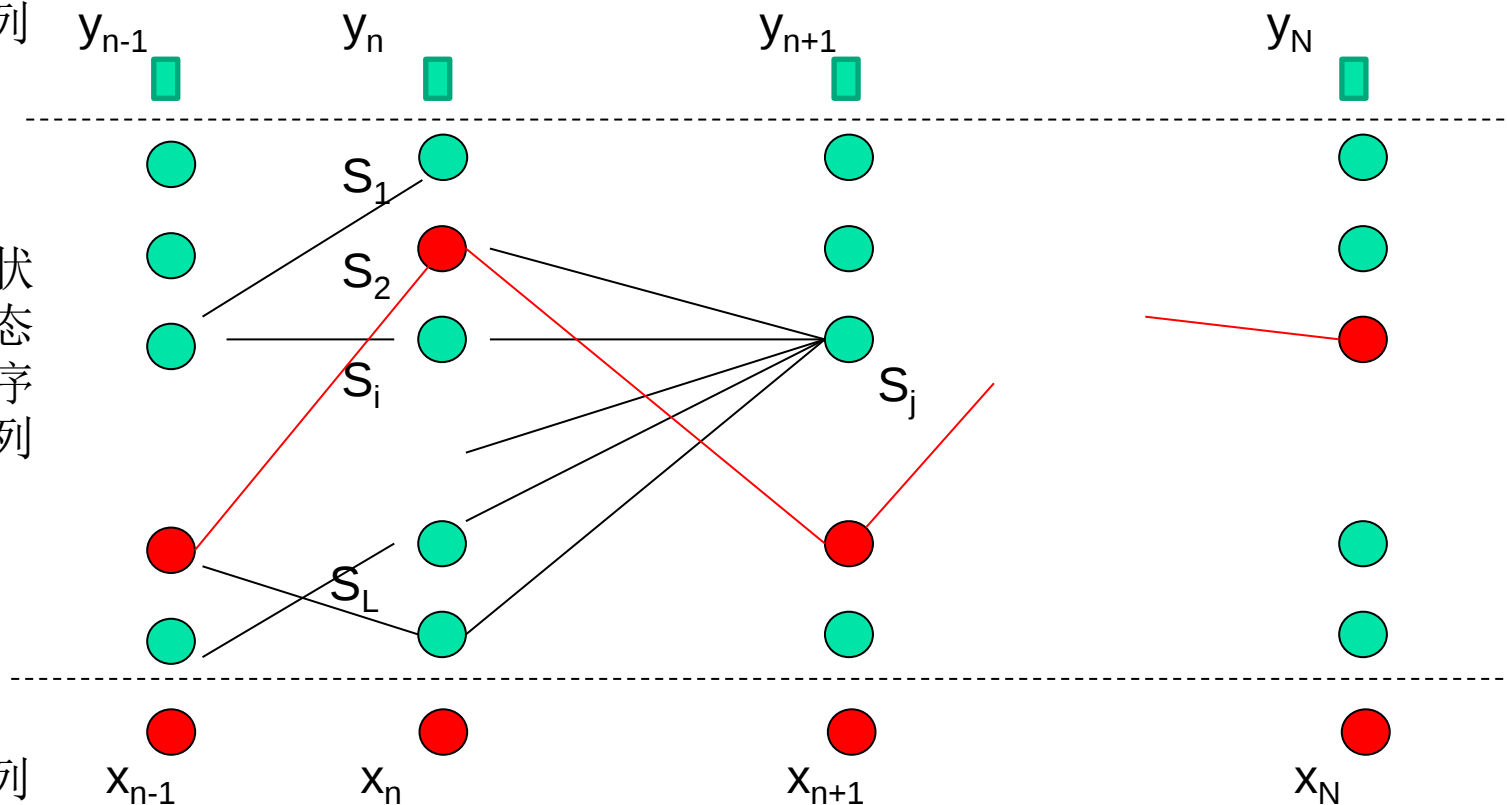
最优状态序列

x_{n-1}

x_n

x_{n+1}

x_N





Viterbi搜索

- HMM模型 – Viterbi Search Algorithm

- 状态序列X对观察特征向量序列Y的刻画
- 概率计算
- 路径搜索

$$\delta_{n+1}(j) = \max_i (\delta_n(i) A_{ij}) \cdot b_j(y_{n+1})$$

- 最优完全路径概率

$$\delta_N(j) = \max_i (\delta_N(i))$$



Viterbi最优路径搜索

- Step 1 初始化

$$\delta_1(i) = \pi_i b_i(y_1) \quad (i = 1, 2, \dots, L)$$

- Step 2 计算概率、最优路径

$$\delta_{n+1}(j) = \max_i (\delta_n(i) A_{ij}) \cdot b_j(y_{n+1})$$

$$\varphi_{n+1}(j) = \arg \max_i (\delta_n(i) A_{ij}) \quad (i, j = 1, 2, \dots, L)$$

- Step 3 确定最优完全路径

$$\hat{l}_N = \arg \max_j (\delta_N(j))$$

- Step 4 路径回溯

$$\hat{l}_n = \varphi_{n+1}(\hat{l}_{n+1})$$

动态规划DP算法-基音检测

■ 代价函数CostFunction()

转移代价: $CostT(n, i, j) = Diff(F0(Can_n(i)), F0(Can_{n+1}(j)))$

目标代价: $CostG(n, i) = Dist(ac(Can_n(i)))$

累计代价: $CostF(n+1, j) = \min_i (CostF(n, i) + CostT(n, i, j)) + CostG(n+1, j)$

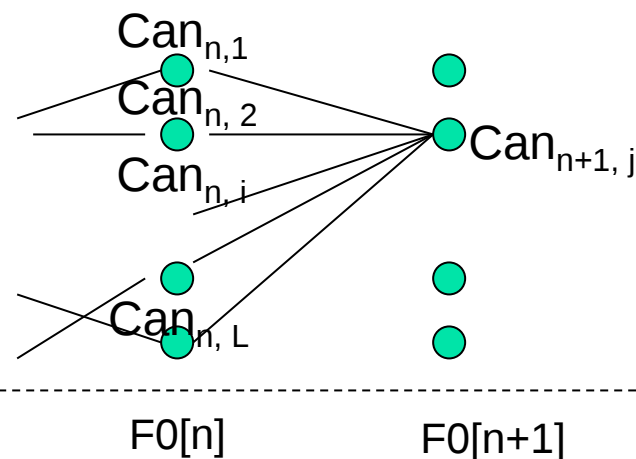
$Can_n(i)$ 表示第n帧的第i个候选

$Diff()$ 表示两个变量间的差异, $F0()$ 表示基频

$Dist()$ 表示距离测度, $ac()$ 表示自相关函数

■ 动态搜索: $F0[1], F0[2], \dots, F0[N]$

- 局部最小化
- 全局最小化





Viterbi搜索

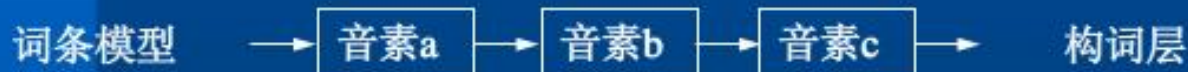
- DP算法的一种形式
 - 概率得分构造代价函数
- 扩展使用
 - 单一HMM（孤立词）vs 多HMM序列（连续语音）
 - 声学模型 vs 语言模型
 - Beam search
 - Token passing (HTK)

语音识别的解码结构

基于HMM框架的连续语音识别系统



Ngram
Beam搜索



发音词典



HMM模型
Viterbi打分



特征提取



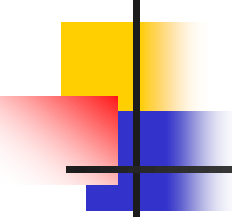
挑战

■ 问题

- 数据 (年龄、性别、语言、方言、主题、情绪)
- 噪声(背景噪声、信道噪声、突发干扰)
- 口语现象(重复、顿措、语序颠倒.....)

■ 探索

- 自上而下：语言模型、对话模型、特定领域
- 自下而上：发音模型、韵律特征引入、基于听觉的语音增强、特定场景
- 近期：DNN（Deep Neuron Network）突破



作业5

- 课程中心平台-“作业5”

- 截止时间12月4日（周日）
- 格式：Word文档，命名规范：姓名_学号_作业5.doc

- 内容

- （1）HMM利用有限状态刻画语音过程，请简要说明HMM三个基本参数，并如何用以刻画状态序列 $\mathbf{X}$ 与语音特征序列 $\mathbf{O}$ 间的关系-计算两者间的联合概率
- （2）基于最大似然解码的语音识别系统可用公式 $\hat{w} = \arg \max_w P(o | w)P(w)$ 加以描述,请说明公式及其中符号的意义
- （3）Viterbi常应用于确定HMM最优状态序列，请说明最优路径搜索及对应的最优完全路径概率计算过程
- （4）传统的语音识别系统由4个主要模块构成：前端、声学模型、语言模型、解码。绘制由此构成的系统框图，说明模块功能及之间的关系



实验3 基频检测器

- 课程中心平台-“实验3”
 - 截止时间12月11日（周日）
 - 格式：Word文档，命名规范：姓名_学号_实验+.doc
- 目的：通过Matlab编程
 - 掌握动态规划DP算法
 - 加深对语音基频参数特性的理解
- 实验平台及材料
 - 下载安装“VoiceBox - matlab工具”
 - Matlab代码'test.m'，功能包括WAV文件读入、语音/参数波形绘制
 - 发音'tang1'的录音文件'tang1.wav'



实验3 基频检测器

■ 实验内容

- 基于自相关系数AC的局部极大值构成基音周期候选点集，通过预设基频范围进行初步筛选；
- 采用动态规划DP算法，通过代价函数CostFunction() - 涉及目标、转移代价，利用帧同步搜索代价最小路径，检测出基频
- 相关算法描述参见课件(计算机语音技术_3_特征分析-基音检测)

■ 实验报告

- 基频检测器总体性设计思路，包括短时自相关函数、基音周期候选集、动态规划等功能模块说明；给出算法关键部分的伪代码
- 结合孤立音节录音'tang1'的基频检测曲线，介绍代码及参数调试过程，对检测结果进行分析
- 自行录制一段连续语音发音，对检测结果进行分析，针对观察到的问题优化算法