

类 Gauss 的主要成员方法说明：

1) **public Map gaussmix(float x[][], float c, float l, int m0, String v0)**

该方法用于初始化 GMM 模型参数。其中，方法参数 x 为 MFCC 特征参数；c 为最小方差的归一化数据；l 的整数部分为最大迭代次数，小数部分为迭代终止阈值；m0 为 GMM 混合数；v0 为 GMM 初始方法设置。

本函数采用 K-均值聚类算法初始化 GMM 模型参数。

2) **public Map gaussmix(float x[][], float c, float l, float m0[], float v0[], float w0[])**

该方法用于估计 GMM 模型参数。其中，方法参数 x 为 MFCC 特征参数；c 为最小方差的归一化数据；l 的整数部分为最大迭代次数，小数部分为迭代终止阈值；m0 为 GMM 模型的均值矩阵；v0 为 GMM 模型的方差矩阵；w0 为 GMM 模型的权重向量。

本函数采用 EM 算法对 GMM 模型参数进行估算。

3) **public Map gaussmixp(float y[][], float m[], float v[], float w[])**

该方法用于计算对数似然度。其中，方法参数 y 为 MFCC 特征参数；m 为 GMM 模型的均值矩阵；v 为 GMM 模型的方差矩阵；w 为 GMM 模型的权重向量。

4) **private Map gaussmix_train_diagonal(float xs[][], float m[], float v[], float w[], float c, float l, float sx0[], float mx0[])**

该方法用于估计 GMM 模型参数，协方差矩阵取对角阵。其中，方法参数 xs 为 MFCC 特征参数；m 为 GMM 模型的均值矩阵；v 为 GMM 模型的方差矩阵；w 为 GMM 模型的权重向量；c 为最小方差的归一化数据；l 的整数部分为最大迭代次数，小数部分为迭代终止阈值；sx0 和 mx0 为 gaussmix() 方法内部使用变量。

5) **public Map kmeans(float d[][], int k, String x0, int l)**

方法为 K-均值聚类算法。其中，方法参数 d 为待分类的数据；k 为 GMM 模型混合数；x0 为初始化方法设置；l 为最大迭代次数。

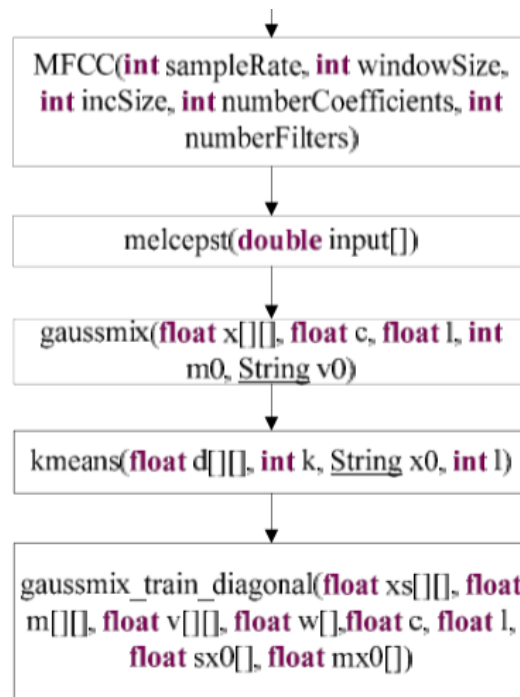


图 4-8 模型训练函数调用示意图

2.5 高斯混合模型

2.5.1 模型描述

高斯混合模型（GMM）本质上是一种多维概率密度函数。其核心思想是在概率空间上其特征向量的分布情形可以用若干个单高斯概率密度函数来线性逼近。

-23-

将其用于说话人识别时，每个说话人对应一个 GMM。

一个 M 阶的 GMM 模型的概率密度函数是由 M 个高斯概率密度函数的加权求和得来的，如下所示：

$$P(o|\lambda) = \sum_{i=1}^M P(o_i|\lambda) = \sum_{i=1}^M C_i P(o|i, \lambda) \quad (2-29)$$

式中 λ ——模型参数集，由均值、协方差、权重组成；

o ——表示 K 维特征参量；

i ——表示高斯分量的序号，GMM 模型的阶数为 M 表示有 M 个高斯分量；
 C_i ——对应于第 i 个高斯分量的混合权重，也就是先验概率。

则有

$$\sum_{i=1}^M C_i = 1 \quad (2-30)$$

式 (2-29) 中， $P(o|q=i, \lambda)$ 可以简化成 $P(o|i, \lambda)$ 的形式，表示序号为 i 的高斯分量，对应可观察的分量 i 的概率密度函数。

对于高斯分量 i ，常用一个维数为 K 的单高斯概率密度函数进行描述，即

$$P(o|i, \lambda) = \frac{1}{(2\pi)^{K/2} |\Sigma_i|^{1/2}} \exp\left[-\frac{(o - \mu_i)^T \Sigma_i^{-1} (o - \mu_i)}{2}\right] \quad (2-31)$$

式中 μ_i ——均值矢量；

Σ_i ——协方差矩阵， $i = 1, \dots, M$ 。

GMM 模型参数集 λ 的表达形式如下：

$$\lambda = \{c_i, \mu_i, \Sigma_i; (i = 1, \dots, M)\} \quad (2-32)$$

式中， λ 由各均值矢量 μ_i 、协方差矩阵 Σ_i 及混合分量的权重组成。其中 Σ_i 可以选择普通矩阵，也可以选择对角矩阵，但通常由于后者的算法简单且性能较好，为了减少计算量， Σ_i 常取后者即对角矩阵^[42]，即

$$\Sigma_i = \text{diag}\{\sigma_{i0}^2, \sigma_{i1}^2, \dots, \sigma_{iK-1}^2\}, \quad 0 \leq k \leq K-1 \quad (2-33)$$

式中 σ_{ik}^2 ——表示对应于高斯分量 i 的特征向量中维数为 k 的分量的方差。

将式 (2-33) 代入式 (2-31) 可得

$$P(o|i, \lambda) = \prod_{k=0}^{K-1} \frac{1}{\sqrt{2\pi}\sigma_{ik}} \exp\left[-\frac{(o_k - \mu_{ik})^2}{2\sigma_{ik}^2}\right] \quad (2-34)$$

式中 o_k ， μ_{ik} ——分别为矢量 o 和矢量 μ_i 的第 k 个分量。

根据公式，GMM 的观察特征矢量与模型匹配的基本示意图如图 2-8 所示：

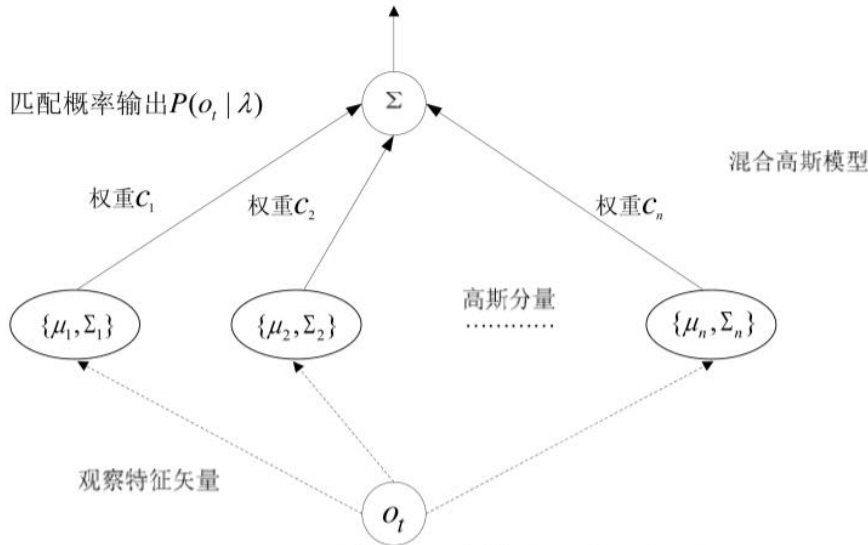


图 2-8 GMM 观察特征矢量与模型匹配的基本示意图

2.5.2 模型参数估计

为说话人建立 GMM 模型，实际上就是通过训练对 GMM 模型的参数进行估计，使其能够最佳匹配训练特征矢量的分布。估计 GMM 模型参数的方法有很多，本文采用的是最常用的就是最大似然（Maximum Likelihood，简称为 ML）估计法。该方法的前提条件是：给定训练的数据，且其目的是：估计 GMM 模型参数，且使估计出的模型的似然值达到最大。对于 T 个训练矢量序列 $O = \{o_1, o_2, \dots, o_T\}$ ，则 GMM 的似然值可用下式来表示：

$$P(O | \lambda) = \prod_{t=1}^T P(o_t | \lambda) \quad (2-35)$$

由于似然函数 $P(O | \lambda)$ 和模型参数集 λ 是非线性函数关系，很难直接求出上式的极大值点，必须引入隐状态来参与计算。因此，为了估计模型参数集 λ ，我们引入期望最大化算法（Expectation Maximization，以下简称为 EM 算法）。求期望（即 Expectation，用 E 来表示）和最大化（即 Maximization，用 M 来表示）是该算法需要实现的两个方面。

EM 算法的估计过程为：从一个初始模型开始，利用最大似然准则，迭代地估计模型参数 $\bar{\lambda}$ ，使 $P(O | \lambda) \leq P(O | \bar{\lambda})$ ，然后再以 $\bar{\lambda}$ 作为新的模型参数开始下一轮迭代过程，直到满足收敛条件。可引入辅助函数 $Q(\lambda, \bar{\lambda})$

$$Q(\lambda, \bar{\lambda}) = \sum_{t=1}^T Q_t(\lambda, \bar{\lambda}) = \sum_{t=1}^T \sum_{i=1}^M \frac{P(o_t, i | \lambda)}{P(o_t | \lambda)} \log P(o_t, i | \bar{\lambda}) \quad (2-36)$$

已知 $P(o_t, i | \lambda) = c_i P(o_t | i, \lambda)$ ，将其代入上式，则有

$$Q(\lambda, \bar{\lambda}) = \sum_{i=1}^M \sum_{t=1}^T \frac{c_i P(o_t | i, \lambda)}{P(o_t | \lambda)} \log \bar{c}_i + \sum_{i=1}^M \sum_{t=1}^T \frac{c_i P(o_t | i, \lambda)}{P(o_t | \lambda)} \log P(o_t, i | \bar{\lambda}) \quad (2-37)$$

令 $\frac{\partial Q(\lambda, \bar{\lambda})}{\partial \bar{c}_i} = 0$ ，则有

$$\bar{c}_i = \frac{\sum_{t=1}^T \frac{c_i P(o_t | i, \lambda)}{P(o_t | \lambda)}}{\left[\sum_{i=1}^M \frac{\sum_{t=1}^T \frac{c_i P(o_t | i, \lambda)}{P(o_t | \lambda)}}{P(o_t | \lambda)} \right]} = \frac{1}{T} \sum_{t=1}^T \frac{c_i P(o_t | i, \lambda)}{P(o_t | \lambda)} \quad (2-38)$$

这样就可将迭代估计 GMM 模型参数的过程分为 Expectation 和 Maximization 两个步骤(即 E-Step 和 M-Step): E-Step, 计算训练数据落在隐状态 i 的概率; M-Step, 以局部最大准则估计 GMM 模型参数集 $\bar{\lambda}$ ，也就是 \bar{c} 、 $\bar{\mu}_i$ 、 $\bar{\sigma}_{ik}^2$ 。

EM 算法的计算过程如下:

a) E-Step : 以 Bayes 公式为依据和基础，计算出训练数据对应于高斯分量 i 的概率:

$$P(q_t = i | o_t, \lambda) = \frac{C_i P(o_t | i, \lambda)}{P(o_t | \lambda)} \quad (2-39)$$

b) M-Step : 分别求出 \bar{c} 、 $\bar{\mu}_i$ 、 $\bar{\sigma}_{ik}^2$:

式 (2-38) 可写成如下形式:

$$\bar{c}_i = \frac{1}{T} \sum_{t=1}^T P(q_t = i | o_t, \lambda) \quad (2-40)$$

同理，均值矢量和方差矩阵可由下式来进行估计:

$$\bar{\mu}_i = \frac{\sum_{t=1}^T P(q_t = i | o_t, \lambda) o_t}{\sum_{t=1}^T P(q_t = i | o_t, \lambda)} \quad (2-41)$$

$$\bar{\sigma}_{ik}^2 = \frac{\sum_{t=1}^T P(q_t = i | o_t, \lambda) (o_{tk} - \mu_{ik})^2}{\sum_{t=1}^T P(q_t = i | o_t, \lambda)}, \quad k = 0, 1, \dots, K-1 \quad (2-42)$$

E-Step 和 M-Step 反复迭代，直至满足收敛条件，即可得到最优的模型参数 λ 。

在实际的训练过程中，通常训练数据比较少，训练环境背景噪声较大，语音易受噪声污染，此时便很可能出现 $\bar{\sigma}_{ik}^2$ 的值过小的情况。然而，过小的方差会极大地影响对整体的似然函数的计算。为了避免此种情况的发生，必须限制方差的范围，即

$$\bar{\sigma}_{ik}^2 = \begin{cases} \bar{\sigma}_{ik}^2, & \bar{\sigma}_{ik}^2 > \bar{\sigma}_{\min}^2 \\ \bar{\sigma}_{\min}^2, & \bar{\sigma}_{ik}^2 \leq \bar{\sigma}_{\min}^2 \end{cases} \quad k = 0, 1, \dots, K-1 \quad (2-43)$$

式中 $\bar{\sigma}_{\min}^2$ ——系统允许的最小方差，推荐取值在 0.01~0.1 之间。

采用 EM 算法估计 GMM 模型参数的流程如图 2-9 所示:

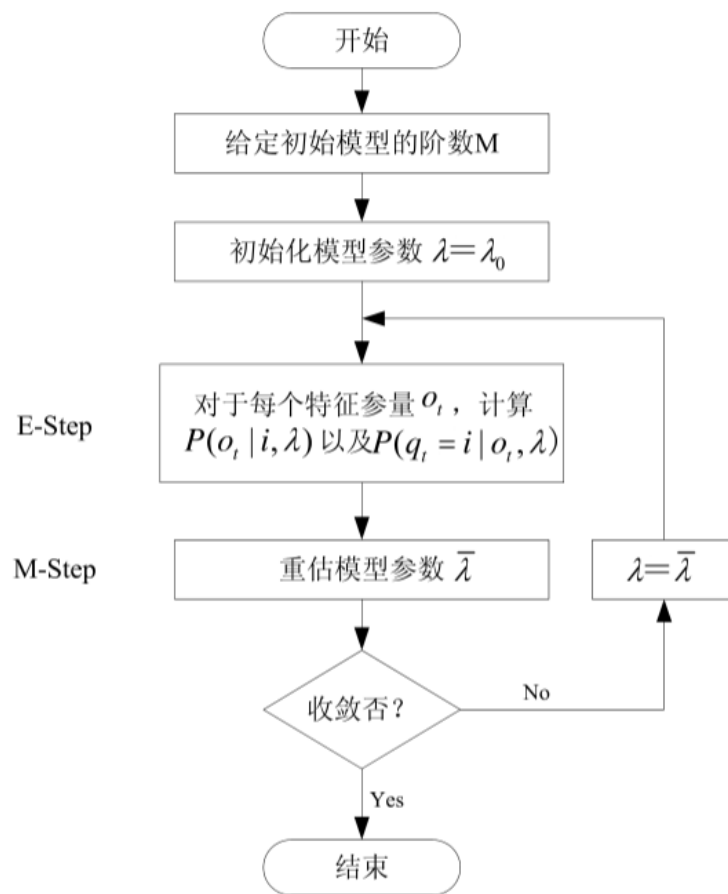


图 2-9 EM 算法估计 GMM 模型参数流程图

2.5.3 模型参数初始化

EM 算法的迭代过程是从一个初始模型开始的，初始模型参数集 λ 的选择会影响算法迭代的速度，其根本原因在于 EM 算法是一个局部最优搜索算法，经过反复迭代可以求出一个最佳解。GMM 模型常用的初始化方法有两种：方法之一，在训练数据中任意取出与 M 个高斯分量相对应 M 组数据，分别计算出每组数据的均值和方差，且将其作为高斯分量的初始的均值和方差，并且使得各分量的权重相同。方法之二，通过 K-均值聚类算法^[43] 对模型参数进行初始化，并划分训练数据到 M 个聚类中，每个高斯分量对应一个聚类，各个高斯分量的初始均值为对应聚类的均值，且每个高斯分量的方差与对应聚类的方差相同，而对应聚类内的数

据量与总数据量的比值即为权重。

GMM 阶数 M 的选择与实际应用有关，一般由实验的结果确定。

K-均值聚类算法^[44]的基本思想是取定将训练数据特征矢量集 $\{x_i\}$ 划分为 M 个类别，以及选择 M 个初始聚类中心，并按最小距离原则将各特征矢量 x_i 逐个分配到 M 类中的某一类，之后不断地计算类的中心和调整各特征矢量的类别，最终使各特征矢量距离其所属类别中心的平方和最小，具体计算步骤如下：

1) 在训练数据特征矢量集中任选 M 个样本作为初始聚类中心： $z_1^{(0)}, z_2^{(0)}, \dots, z_M^{(0)}$ ，令 $k=0$ 。

2) 按最小距离原则，将待分类的特征矢量集 $\{x_i\}$ 中的特征矢量逐个地分给 M 类中的某一类，即

$$\text{如果 } d_{ij}^k = \min[d_{ij}^{(k)}], \quad i = 1, 2, \dots, N \quad (2-44)$$

$$\text{则判 } x_i \in \omega_j^{(k+1)}$$

式中 d_{ij}^k ——表示特征矢量 x_i 与聚类 $\omega_j^{(k)}$ 的中心 $z_j^{(k)}$ 之间的距离，上标表示迭代次数。

这样产生了新的聚类 $\omega_j^{(k+1)}$ ， $j=1, 2, \dots, M$ 。

3) 计算重新分类后的各类中心

$$z_j^{(k+1)} = \frac{1}{n_j^{(k+1)}} \sum_{x_i \in \omega_j^{(k+1)}}^{(k)} x_i, \quad j = 1, 2, \dots, M \quad (2-45)$$

式中 $n_j^{(k+1)}$ ——表示 $\omega_j^{(k+1)}$ 类中所含特征矢量的个数。

4) 如果 $z_j^{(k+1)} = z_j^{(k)}$ ($j=1, 2, \dots, M$)，则迭代结束；否则，令 $k=k+1$ ，转至步骤 2) 开始下一轮迭代。

2.5.4 识别判决

对于一个有 N 个人的说话人识别系统，用 λ_i ($i=1, 2, \dots, N$) 代表第 i 个说话人的 GMM 模型。在进行识别时，假设待识别语音的观察特征矢量序列为 $O = \{o_1, o_2, \dots, o_T\}$ ，则判定该目标说话人为第 n 个说话人的后验概率为^[45]

$$p(\lambda_n | O) = \frac{p(O | \lambda_n) p(\lambda_n)}{p(O)} = \frac{p(O | \lambda_n) p(\lambda_n)}{\sum_{m=1}^N p(O | \lambda_m) p(\lambda_m)} \quad (2-46)$$

$$p(O | \lambda) = \prod_{t=1}^T p(O_t | \lambda) \quad (2-47)$$

式中 $p(O | \lambda_n)$ ——表示第 n 个说话人产生 O 的条件概率；

$p(O)$ ——表示全部说话人条件下 O 的概率；

$p(\lambda_n)$ ——表示第 n 个说话人的先验概率。

由于待识别的说话人相对独立，将其先验概率设为 $p(\lambda_n) = \frac{1}{N}$, $n = 1, 2, \dots, N$, 此外，所有说话人的 $p(O)$ 都相等，则识别判决结果可由最大后验概率准则给出，即

$$n^* = \arg \max_{1 \leq n \leq N} P(\lambda_n | O) = \arg \max_{1 \leq n \leq N} P(O | \lambda_n) \quad (2-48)$$

为了简化计算，通常采用对数似然函数，则闭集说话人辨认的得分公式由下式给出

$$n^* = \arg \max_{1 \leq n \leq N} \ln P(O | \lambda_n) = \arg \max_{1 \leq n \leq N} \sum_{t=1}^T \ln P(o_t | \lambda_n) \quad (2-49)$$