高斯混合 Copula 模型

摘要

在统计学领域，混合模型被广泛用于描述具有多峰分布的数据集。其中，高斯混合模型（Gaussian Mixture Model, GMM）因其坚实的统计基础和高效的学习算法而受到了研究者的青睐。然而，GMM 的一个基本假设是每个混合分量都呈正态分布，这对许多现实生活中的数据集来说可能过于严格。在本文中，我们介绍了一类基于 Copula 函数的参数混合模型，即高斯混合 Copula 模型（Gaussian Mixture Copula Model, GMCM）。GMCM 的目标是放宽对混合成分正态性的假设，从而作为 GMM 的一种更具表现力的替代方案。然而，GMCM 的期望最大化和直接似然最大化框架中存在似然函数缺乏封闭形式的问题，且参数存在不可识别性的额外挑战。虽然已经有一些工作对虽然已经有一些工作对这些问题进行了一定程度的调整和改进，但问题仍然存在。在本文复现的论文中，作者为每个问题提供了解决方案。不仅证明了不可识别性的根源，而且还提出了消除该问题的合适先验。此外，还提出了一种有效的数值框架来评估棘手的似然函数，同时还提供其解析导数。最后，提出了将 GMCM 作为来自基本分布的一系列双射映射的视图，使其能够使用现代概率编程语言来实现。

关键词：混合模型；Copula 函数；双射映射；高斯混合 Copula 模型

1 引言

混合模型是指用于描述由不同基础分布生成的观测值组成的总体的概率模型。通常，假设这些分布具有相同的参数形式，但允许具有不同的参数值。在统计学中，混合模型可以用来表征具有多峰分布的数据集。其中，高斯混合模型（Gaussian Mixture Model, GMM）因其良好的统计特性和高效的学习算法得到了广泛使用。在 GMM 中，假设每个混合分量都服从正态分布。对于给定数量的组件，可以使用期望最大化（Expectation-Maximization, EM）算法 [10] 有效地学习 GMM 的参数。GMM 有许多应用任务，包括异常检测 [34]，图像分割 [1] 和运动目标检测 [29] 等。然而，在 GMM 中每个混合构件均呈正态分布的基本假设对于许多现实生活数据集来说通常过于严格。因此，如果数据包含异常值或遵循非高斯分布，GMM 可能无法准确捕获底层结构。

为了解决该问题，Tewari 在 2011 年提出了一个新的混合模型——高斯混合 Copula 模型（Gaussian Mixture Copula Model, GMCM）[31]，其基础是 Copula 函数理论。Copula 函数通过将随机变量边缘分布的估计与它们之间存在的依赖关系分开，来提供一种估计联合分布的方法。该模型放宽了每个分量为正态分布的假设，同时保留了 GMM 的多模特性。虽然 GMCM 相比于 GMM 更具有表现力，但是 GMCM 参数的估计仍然是一个挑战，GMCM 的

期望最大化和直接似然最大化框架都缺乏封闭形式的似然函数。后续有工作对该问题进行进一步推进 [6,19,20,27]。这些方法提出了一些可以缓解问题的近似方法，但是问题仍然没有得到解决。对目前 GMCM 存在的问题总结如下：

(1) GMCM 存在参数不可识别性的固有问题。

(2) 它的似然函数没有封闭的分析形式，因此不适合用于参数估计的 EM 框架。

(3) 与第二个原因相伴随，由于缺乏分析梯度（数值梯度的计算成本很高），即使直接似然最大化（通过基于梯度的方法）也变得困难。

在 2023 年，Tewari 重新审视 GMCM，为每个问题提供了解决方案，帮助 GMCM 充分发挥其潜力 [30]。该论文不仅证明了不可识别性的根源，而且还提出了消除该问题的合适先验。此外，还提出了一种有效的数值框架来评估棘手的似然函数，同时还提供其解析导数。在 EM 算法求解上，提出了 GMCM 的 EM 算法的正确公式。最后，提出了将 GMCM 作为来自基本分布的一系列双射映射（bijective mappings）的视图，这为使用现代概率编程语言综合 GMCM 铺平了道路。该论文经过了严格的审稿过程且由 2023 年的国际机器学习大会（International Conferenceon Machine Learning，ICML）收录接收，说明其研究内容和成果得到了学术界的认可。因此，该论文所提出的方法具有一定的参考价值和学习意义。本文对该论文进行详细的介绍以及梳理，并对复现结果进行了展示。

2 相关工作

2.1 Copula 构造

Copula 是一个多元分布函数，其边缘分布都是在单位区间上的均匀分布。众所周知，任何连续随机变量都可以通过概率积分变换转化为单位区间内的均匀随机变量。因此，Copula 可以用于将不同的边缘分布“耦合”在一起，构造新的多元分布。该方法将多元分布分为两个组成部分，即所有边缘分布和 Copula 函数，为多元建模提供了非常灵活的框架。 [25] 和 [18] 是关于这个主题的综合参考书目。对于广泛使用的介绍，请参见 [17] 以及 [15]。在高维的构造上，一些工作提出通过假设随机变量之间的树结构依赖关系从双变量 Copula 合成多元 Copula [2,8,18]。另外，最近的工作中还有基于变分方法的高维 Copula 函数构造方法 [28]，以及通过神经网络来估计高维 Copula 函数 [32]。

基于 Copula 函数的模型在各个领域都受到了广泛的关注。精算师使用 Copula 函数来模拟相关死亡率和损失 [16,33]，在金融分析上 Copula 函数是常用的一种非线性相关研究工具 [3,9]，而生物学上使用 Copula 函数来建模相关事件时间和竞争风险 [14]。此外人们还尝试寻找 Copula 理论和机器学习之间的协同作用，以构建高保真数据驱动模型 [13]。

2.2 高斯混合 Copula 模型

Tewari 于 2011 年首先提出了基于 Copula 函数的 GMCM 来捕获数据的多模分布 [31]。在统计学中，多峰分布是一种具有多个模式的概率分布，即分布具有多个局部峰值，这是在现实数据中经常观察到的特征。Li 等人 [23] 研究了 GMCM 的一个具体案例，设计了一种

称为可重复性分析的 meta 分析方法，以验证多个高通量基因组学实验的可靠性和一致性，使得 GMCMs 在应用中取得了一些成功。Bilgrau 等人 [6] 进一步推进了这项工作，指出了 GMCM 参数估计的挑战，并提出了解决方案，后续实现了该改进模型的 R 语言包 [5]。Rajan 和 Bhattacharya [27] 对 GMCM 进行扩展，构建灵活的混合数据类型 (连续和离散) 生成式模型。Kasa [19] 等人解决了基因表达数据集中常见的高维情况下 GMCM 估计的可扩展性问题。Kasa 和 Rajan [20] 探讨了自动微分的作用，以获得 GMCM 棘手的似然函数的梯度。然而，这些工作只是部分地解决了 GMCM 的参数不可辨识性及其似然函数 (与梯度) 难以解决的基本问题。为了解决这些仍然存在的问题，Tewari 于 2023 年继续完善 GMCM [30]，这项工作为每个问题提供了解决方案，试图帮助 GMCM 实现其全部潜力。

需要注意的是，还有一些将 Copula 函数与混合模型相结合的工作 [22,24]。不同的是，这些工作是指有限混合模型，其中每个成分密度使用 Copula 定义，通过选择任意的 Copula 密度和每个成分的边际来构建新的混合模型。例如，高斯 Copula 混合模型（Gaussian Copula Mixture Model, GCMM）[24] 采用高斯 Copula 对成分相依关系进行建模。相比之下，GMCM 是一种特定的 Copula 模型，其目标是寻求一个单一的 Copula 分布来捕捉整个多峰相依结构。

3 本文方法

3.1 本文方法概述

在介绍本文方法之前，我们先对 Copula 函数和高斯混合模型（Gaussian Mixture Model, GMM）进行了解。Copula 这个单词来自于拉丁语，意思是“连接”。最早是由 Sklar 在 1959 年提出的。

> Sklar 定理：随机变量 $\boldsymbol{x}$ 的联合分布 $F(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_d)$ 可以表示为边缘分布 $F_i(\boldsymbol{x}_i)$ 的函数。
>
> $$F(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_d) = C(F_1(\boldsymbol{x}_1), F_2(\boldsymbol{x}_2), \ldots, F_d(\boldsymbol{x}_d)) \tag{1}$$

对于 d 维的随机变量的联合分布，可以将其分解为这 d 个维度的边缘分布和一个 Copula 函数，从而将变量的随机性和耦合性分离开来。其中，随机变量的随机性由边缘分布进行描述，不同维之间的耦合性由 Copula 函数进行表述，即一个联合分布关于相关性的性质由其 Copula 函数决定。从公式 1中也可以看出，Copula 可以被视为 d 维随机向量 $\boldsymbol{u}$ 的累积分布函数（Cumulative Distribution Function，CDF），使得 $\boldsymbol{u}_j \sim \text{Uniform}(0,1)$，$j \in \{1, 2, \ldots, d\}$。公式 1表明我们可以用边缘 CDF 和 Copula 来编写联合 CDF。相反，如果我们知道联合 CDF 和边际 CDF，我们可以通过以下方式得到 Copula 函数

$$C(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d) = F(F_1^{-1}(\boldsymbol{u}_1), F_2^{-1}(\boldsymbol{u}_2), \ldots, F_d^{-1}(\boldsymbol{u}_d)), \tag{2}$$

其中 $\boldsymbol{u}_j = F_j(\boldsymbol{x}_j)$，$F_j^{-1}(\boldsymbol{u}_j) = \boldsymbol{x}_j$，且 $F_j^{-1}(\cdot)$ 为 $F_j(\cdot)$ 的逆函数。

鉴于 Copula 可以被视为均匀随机向量的 CDF，我们就可以考虑相应的概率密度函数（Probability Density Function，PDF）。这种密度函数称为 Copula 密度。则随机变量 $\boldsymbol{x}$ 的 PDF 可以写为

$$f(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_d) = c(F_1(\boldsymbol{x}_1), F_2(\boldsymbol{x}_2), \ldots, F_d(\boldsymbol{x}_d)) f_1(\boldsymbol{x}_1) f_2(\boldsymbol{x}_2) \ldots f_d(\boldsymbol{x}_d), \tag{3}$$

其中 $f_j(\boldsymbol{x}_j)$ 是 $\boldsymbol{x}_j$ 的 PDF，$c(\cdot)$ 是 Copula 密度函数。可以将 $c(\cdot)$ 视为联合 CDF 为 $C(\cdot)$ 的均匀随机向量对应的 PDF。同样的，我们可以得到 Copula 密度函数为

$$c(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d) = \frac{f(F_1^{-1}(\boldsymbol{u}_1), F_2^{-1}(\boldsymbol{u}_2), \ldots, F_d^{-1}(\boldsymbol{u}_d))}{f_1(F_1^{-1}(\boldsymbol{u}_1)) f_2(F_2^{-1}(\boldsymbol{u}_2)) \ldots f_d(F_d^{-1}(\boldsymbol{u}_d))} \tag{4}$$

如图 1所示，得到边缘 PDF 后与 Copula 密度函数相乘，可以得到符合数据分布的联合 PDF。

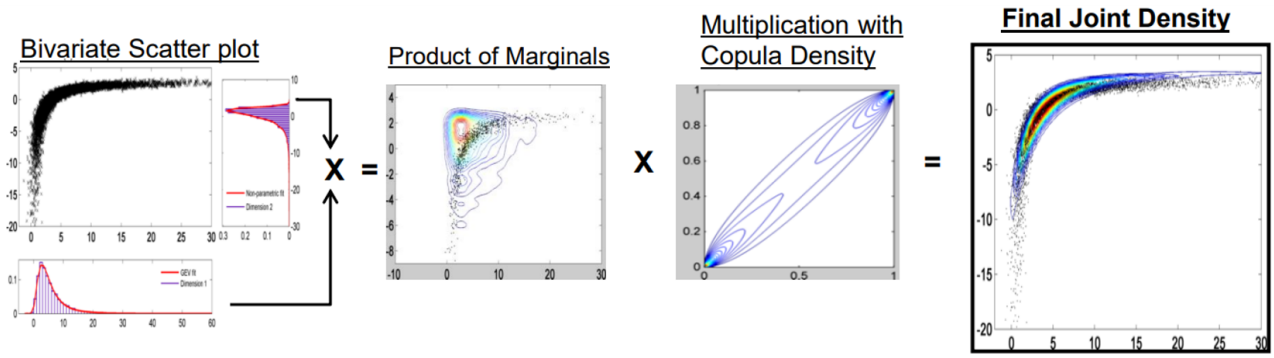


图 1. Copula 的作用

而 GMM 使用多个高斯分布的组合来刻画数据分布，可以看成是多个高斯分布的加权叠加。有 m 个高斯分量组成的 GMM 的 PDF 定义如下：

$$\psi(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_d; \Theta) = \sum_{k=1}^{m} \alpha^{(k)} \phi(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_d; \theta^{(k)}) \tag{5}$$

其中 $\alpha^{(k)} \geq 0$ 为每个分量的权重，且满足 $\sum_{k=1}^{M} \alpha^{(k)} = 1$。$\theta^{(k)}$ 表示与第 k 个分量的均值向量 $\boldsymbol{\mu}^{(k)}$ 和协方差矩阵 $\Sigma^{(k)}$ 相关的参数。参数集 Θ 综合了所有分量的权重、均值向量和协方差矩阵。

在学习了 Copula 函数以及 GMM 后，接下来对本文的高斯混合 Copula 模型（GMCM）进行介绍。将 GMM 分布代入到公式4中，可以得到本文高斯混合 Copula（Gaussian mixture Copula，GMC）密度函数的表达式：

$$\zeta(\boldsymbol{u}; \Theta) = \frac{\psi(\Psi^{-1}(\boldsymbol{u}); \Theta)}{\prod_{r=1}^{d} \psi_r(\Psi_r^{-1}(\boldsymbol{u}_r); \Theta^r)} \tag{6}$$

其中 $\Psi(\cdot)$（和 $\Psi^{-1}(\cdot)$）为 GMM 的联合 CDF（及其逆函数），$\Psi_r(\cdot)$（和 $\Psi_r^{-1}(\cdot)$）为沿 r 维的边缘 CDF（及其逆函数）。在得到 GMC 密度函数后，联合概率密度可以通过单独学习的边缘密度 $f_r(\boldsymbol{x}_r)$ 表示为：

$$p(\boldsymbol{x}; \Theta) = \zeta(\boldsymbol{u}; \Theta) \cdot \prod_{r=1}^{d} f_r(\boldsymbol{x}_r) \tag{7}$$

所提出的 GMCM 将边缘分布的估计与不同模式中存在的依赖结构解耦。这种解耦允许彼此独立地估计边际分布，同时借用 GMM 的依赖结构。图 2展示了 GMCM 相比于 GMM 的表现能力。可以看到，GMCM 拟合分布更紧密，且生成的随机样本与训练数据集更相似，表明 GMCM 相比于 GMM 更具有表现力。

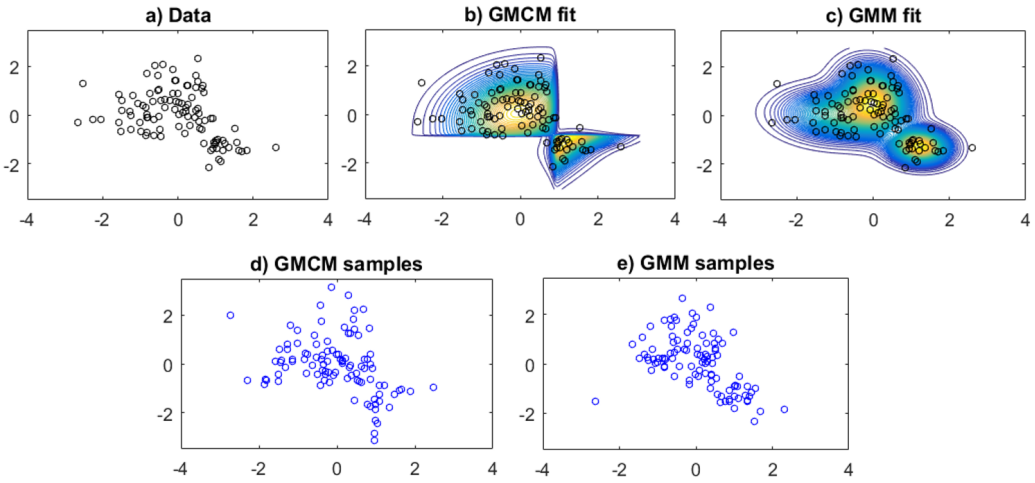


图 2. (a) 表示一个包含 100 个样本的二维数据集，(b-c) 分别为训练集上 GMCM 和 GMM 的拟合，(d-e) 表示由 GMCM 和 GMM 这两个拟合分布产生的随机样本。

尽管具有优越的表达能力，GMCM 参数的估计仍然存在一些一些突出问题。GMCM 的期望最大化和直接似然最大化框架都必须解决缺乏封闭形式的似然函数，该论文提出了一种有效的数值框架来估计棘手的似然函数，同时还提供其解析导数，且提出了一种可证明正确的 GMCM 广义 EM 算法。此外，过去的工作提到了参数不可识别性的额外挑战，本文不仅证明了不可识别性的根源，而且还提出了消除该问题的合适先验。为了让 GMCM 能够成为数据建模者手中的另一种多元建模工具，论文提出了将 GMCM 视为来自基本分布的一系列双射映射的视图，使其适用于现代概率编程框架并利用其内置的自动微分功能。接下来对每个技术点进行简要的介绍。

3.2 GMCM 的双射变换视图

在概率论中，概率积分变换（也称为均匀的普适性）是指将任意给定连续分布的随机变量的数据值转换成具有标准均匀分布的随机变量的结果。例如 $\boldsymbol{x}$ 是随机变量，而 $\boldsymbol{u}$ 是该随机变量的分布，即 $\boldsymbol{u} = F(\boldsymbol{x})$。那么，$\boldsymbol{u}$ 服从均匀分布，$\boldsymbol{u} \sim \text{Uniform}(0, 1)$。同样的，连续型随机变量的分布也可以从均匀分布得到。例如我们想得到分布为 $F(\boldsymbol{x})$ 的随机变量，可以从一组均匀分布的数据 $\boldsymbol{u}$ 得到，此时 $\boldsymbol{x} = F^{-1}(\boldsymbol{u})$。

基于概率积分变换，可以通过一系列双射变换来变换简单基分布（例如高斯分布）来构成联合分布的方法。只要相应的雅可比矩阵的变换（正向和逆向）和行列式被明确定义，可以将任意一组双射简单地链接到基分布，以产生高度表达的联合分布。通过图 3中的示例进行说明，其中基分布是 GMM。然后通过两个双射映射对基分布进行变换；第一个为 GMM 分布的边缘 CDF$\Psi_r(\boldsymbol{z}_r)$，第二个是先验学习的边缘 CDF 的逆函数 $F_r^{-1}(\boldsymbol{u}_r)$。因此，GMCM 诱导的生成过程可以指定如下：

$$
\begin{aligned}
&\boldsymbol{z} \sim GMM(\Theta) \\
&\boldsymbol{u} = [\Psi_1(\boldsymbol{z}_1; \Theta^1), \Psi_2(\boldsymbol{z}_2; \Theta^2), \ldots, \Psi_d(\boldsymbol{z}_d; \Theta^d)] \\
&\boldsymbol{x} = [F_1^{-1}(\boldsymbol{u}_1), F_2^{-1}(\boldsymbol{u}_2), \ldots, F_d^{-1}(\boldsymbol{u}_d)]
\end{aligned} \tag{8}
$$

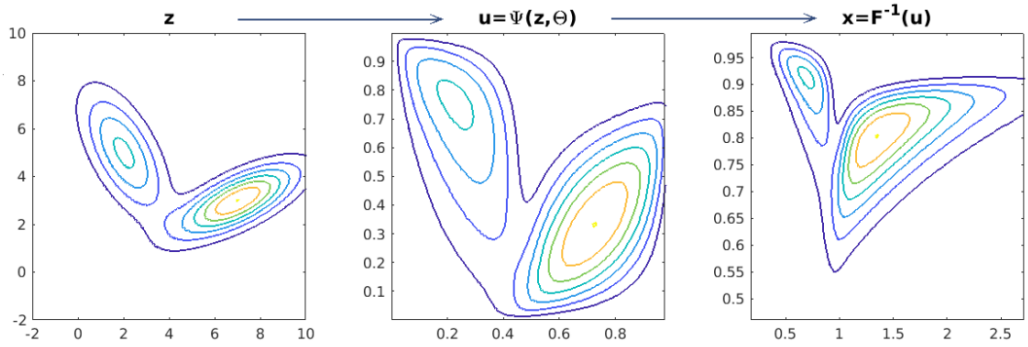


图 3. GMCM 变换的图示

图 4描述了 GMCM 中涉及的双射变换。变换空间 X 中的联合密度可以通过将样本反转到基空间 Z 并利用变量变换定理（the change of variables formula） [26] 来获得。具体公式如下，其中括号内的表达式对应于两个变换的雅可比行列式：

$$p(\boldsymbol{x}) = \left(\prod_{r=1}^{d} \frac{dF_r(\boldsymbol{x}_r)}{d\boldsymbol{x}_r}\right) \cdot \left(\prod_{r=1}^{d} \frac{d\Psi_r^{-1}(\boldsymbol{u}_r)}{d(\boldsymbol{u}_r)}\right) \cdot \psi(\boldsymbol{z}) \tag{9}$$

此时该联合密度可以进一步写为公式 7，即 GMC 密度函数和单独学习的边缘密度 $f_r(\boldsymbol{x}_r)$ 的乘积。利用 GMCM 的双射变换视图，可以用 TensorFlow Probability [12] 提供简洁方便的 API 来构建这种转换。

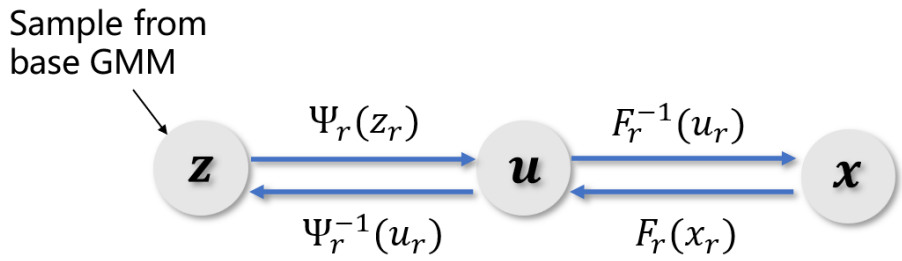


图 4. GMCM 涉及的变换的图示

3.3 GMCM 的极大似然估计

在 GMCM 的极大似然估计中，其中首先学习边缘分布，然后通过最大化方程 10给出的对数似然函数来估计 GMC 函数的参数。

$$\mathcal{L}_\zeta(\Theta|\boldsymbol{U}) = \sum_{i=1}^{n} \log[\zeta(\boldsymbol{U}_{:i}; \Theta)] \tag{10}$$

其中函数 $\zeta(\cdot)$ 是由公式 6给出的 GMC 密度函数，$\boldsymbol{U}$ 为大小为 $d \times n$ 的矩阵。假设从训练数据集 $\boldsymbol{X}$ 中学习了边缘分布，则通过学习的边缘分布函数对数据集 $\boldsymbol{X}$ 进行变换后可以得到 $\boldsymbol{U}$，即 $\boldsymbol{u}_j = F_j(\boldsymbol{x}_j)$，$j \in \{1, 2, \ldots, d\}$。作为一个连续且平滑的函数，可以使用任何基于梯度的算法来最大化 $\mathcal{L}_\zeta(\Theta|\boldsymbol{U})$。但是直接计算梯度的计算成本很高，主要体现在以下两方面。

1. $\Psi_r^{-1}(\boldsymbol{u}_r)$ 无法得到解析形式
   令 $\boldsymbol{z}_r = \Psi_r^{-1}(\boldsymbol{u}_r)$，则 $\boldsymbol{u}_r$ 可以表示为

$$\boldsymbol{u}_r = \Psi_r(\boldsymbol{z}_r; \Theta^r) = \frac{1}{2}\sum_{l=1}^{m} \alpha^l \left[1 + \operatorname{erf}(\frac{\boldsymbol{z}_r - \boldsymbol{\mu}_r^l}{\sqrt{2\Sigma_{rr}^l}})\right] \tag{11}$$

   可以验证 $\boldsymbol{z}_r = \Psi_r^{-1}(\boldsymbol{u}_r)$ 无法显式导出，因此需要数值计算的方法。Bilgrau [6] 等人提出了用线性插值来近似逆函数，并对 $\operatorname{erf}(\cdot)$ 进行经验近似。对于样本大小为 n，维度为 d，基分布 GMM 的分量个数为 m，插值网格大小为 g 的情况，当 $n, d \gg m$ 时，公式 10的计算成本为 $O(mdg + nd\log g)$，其中第一项是由于在 d 维度的 g 个网格点上所涉及的成本，第二项是线性插值的成本。

2. 对参数的偏导计算复杂
   由于 GMCM 的似然函数缺乏闭合形式，分子和分母中带有指数项的累加的对数，因此偏导的推导更具挑战性。在先前的工作中 [6,31]，通过有限差分（finite difference，FD）法来对 GMCM 对数似然函数梯度进行近似。尽管对于小问题有效，但该方案对于问题规模的扩展性很差。由于 GMC 分布具有 $O(m + md + md^2)$ 个参数，因此 FD 近似的复杂度为 $O(md^2 C_{\mathcal{L}(\Theta|\boldsymbol{U})})$（当 $d \gg m$），$C_{\mathcal{L}(\Theta|\boldsymbol{U})}$ 为计算公式10的成本。因此整体复杂度为 $O(m^2 d^3 g + nmd^3 \log g)$。

对于上述极大似然估计的难点，虽然已经有工作在一定程度上缓解了该问题，但是没有完全解决极大似然估计的参数求解。在以下缺陷的驱使下：

- 复杂度与插值的网格尺寸有关
- 近似计算导致低质量梯度的可能

论文提出了提出一个数值方案来估计 Ψ_r^{-1}，同时提供相同的解析偏导数。在计算 Ψ_r^{-1} 时，通过一种计算高效的替代方法，由于 $\boldsymbol{z}_r = \Psi_r^{-1}(\boldsymbol{u}_r)$，所需的求逆替换为寻求表达式 $\boldsymbol{u}_r - \Psi_r(\boldsymbol{z}_r; \Theta^r)$ 的根。与基于线性插值的方法相比，采用求根算法来获得逆函数所需的计算次数要少得多。论文中使用的求根算法为 Chandrupatla 算法 [11]。

GMCM 的 Ψ_r^{-1} 计算解决了部分问题，还需要求解关于 Θ^r 的 Ψ_r^{-1} 的偏导数。虽然 Ψ_r^{-1} 的解析式无法得到，但是这些偏导数可以通过分析获得。利用正向函数 $\Psi_r(\cdot)$ 以及 $\boldsymbol{z}_r = \Psi_r^{-1}(\boldsymbol{u}_r)$，调用欧拉链式法则解析地计算偏导数。

$$\frac{d\boldsymbol{z}_r}{d\theta} = \frac{d\boldsymbol{z}_r}{d\boldsymbol{u}_r} \cdot \frac{d\boldsymbol{u}_r}{d\theta} = \frac{\frac{d\Psi_r(\boldsymbol{z}_r)}{d\theta}}{\frac{d\Psi(\boldsymbol{z}_r)}{d\boldsymbol{z}_r}} \tag{12}$$

分母中的表达式对于所有偏导数都是相同的，只是单变量 GMM 的密度函数。分子关于 $\theta \in \{\alpha_k, \mu_k, \Sigma_k\}$ 的偏导数可以通过应用矩阵微积分恒等式来导出。在 TensorFlow Probability 中可以在函数中嵌入自定义导数，虽然程序原则上可以通过迭代，数值计算等方法进行自动微分，但是如果大量的自动微分容易导致浮点精度错误、内存使用过多和计算速度缓慢等问题。所以导出解析导数是有必要的。

3.4 GMCM 的 EM 算法

EM 算法是一种迭代算法，用于统计中以找到概率模型中参数的最大似然估计，该估计依赖于不可观察的隐藏变量。该算法在机器学习中有极为广泛的用途，例如常被用来学习 GMM 的参数。它的计算方法中每一次迭代都分两步，其中一个为期望步（E 步），另一个为极大步（M 步）。其基本思想是：首先根据已经给出的观测数据，估计出模型参数的值；然后再依据上一步估计出的参数值估计缺失数据（隐变量）的值，再根据估计出的缺失数据加上之前已经观测到的数据重新再对参数值进行估计，然后反复迭代，直至最后收敛，迭代结束。具体算法流程如图 5所示。

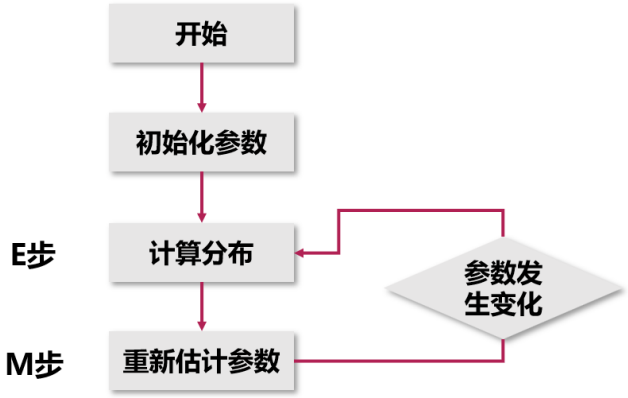


图 5. EM 算法的流程

虽然 GMCM 和 GMM 有相同的参数集，但是用于 GMM 参数估计的 EM 算法并不直接适用于 GMCM。因为 GMCM 的似然函数没有封闭形式，边缘密度函数的逆函数 $\Psi_r^{-1}(\cdot)$ 需要进行数值求解，因此 GMCM 使用 EM 算法要困难的许多。在之前的工作中 [4,31] 提出了参数更新规则，调整 GMM 的 EM 算法来学习 GMCM 的参数，但是这些规则不会最大化 GMCM 对数似然的真实下限。因此，需要进行额外的检查和更正，以确保 EM 更新期间对数似然单调增加，这类算法被成为伪 EM（pseudo-EM，PEM）算法。论文根据文献 [7] 对 EM 算法框架进行重新梳理，具体内容如下。

假设隐变量为 $\boldsymbol{y}$，是与观测数据 $\boldsymbol{U}$ 共同出现的 n 维隐变量向量，则完整的对数似然函数为

$$\mathcal{L}_{comp}(\Theta|\boldsymbol{U},\boldsymbol{y}) = \sum_{i=1}^{n} \log \left(\frac{\alpha^{\boldsymbol{y}_i} \phi(\boldsymbol{Z}_{:i}; \Theta^{\boldsymbol{y}_i})}{\prod_{r=1}^{d} \psi_r(\boldsymbol{Z}_{ri}; \Theta^r)} \right) \tag{13}$$

隐变量 $\boldsymbol{y}_i$ 表示高斯分量的索引，即样本 $\boldsymbol{U}_{:i}$ 属于哪个分量，如果第 i 个样本是由第 k 个混合分量生成的，则 $\boldsymbol{y}_i = k$。函数 $\phi(\cdot)$ 是多元高斯密度，$\Theta^{\boldsymbol{y}_i}$ 和 Θ_r 分别表示与分量 $\boldsymbol{y}_i$ 和维度 r 相关的参数。此外，$\boldsymbol{Z}_{:i}$ 和 $\boldsymbol{Z}_{ri}$ 分别用于表示 $\Psi^{-1}(\boldsymbol{U}_{:i})$ 和 $\Psi_r^{-1}(\boldsymbol{U}_{ri})$。分母不取决于隐变量 $\boldsymbol{y}_i$，因为边缘密度 $\Phi_r(\cdot)$ 不是特定于组件的。E 步骤涉及根据给定数据和当前参数估计（即 $\hat{\Theta}$）的

隐变量的后验分布，推导完整对数似然函数（公式 13）的期望值。作者推导出正确的期望为

$$
\begin{aligned}
Q(\Theta, \hat{\Theta}) = & \sum_{i=1}^{n} \sum_{\boldsymbol{y}_i=1}^{m} \left(\log(\alpha^{\boldsymbol{y}_i}) - \frac{\log(|\Sigma^{\boldsymbol{y}_i}|)}{2} \right) \boldsymbol{G}_{i\boldsymbol{y}_i} \\
& - \sum_{i=1}^{n} \sum_{\boldsymbol{y}_i=1}^{m} \left(\frac{\overline{\boldsymbol{Z}}_{:i}^{T} (\Sigma^{\boldsymbol{y}_i})^{-1} \overline{\boldsymbol{Z}}_{:i}}{2} \right) \boldsymbol{G}_{i\boldsymbol{y}_i} \\
& - \sum_{i=1}^{n} \sum_{r=1}^{d} \log \left(\psi_r \left(\boldsymbol{Z}_{ri}; \Theta^r \right) \right)
\end{aligned}
\tag{14}
$$

其中 $\overline{\boldsymbol{Z}}_{:i} = \boldsymbol{Z}_{:i} - \mu^{\boldsymbol{y}_i}$，且表示 $\boldsymbol{G}_{i\boldsymbol{y}_i}$ 给定第 i 个样本和当前参数估计的第 $\boldsymbol{y}_i$ 个分量的后验概率，公式如下：

$$
G_{i\boldsymbol{y}_i} = \frac{\alpha^{\boldsymbol{y}_i} \phi \left(\Psi^{-1}(\boldsymbol{U}_{:i}); \hat{\Theta}^{\boldsymbol{y}_i} \right)}{\sum_{j=1}^{m} \alpha^{j} \phi \left(\Psi^{-1}(\boldsymbol{U}_{:i}); \hat{\Theta}^{j} \right)}
\tag{15}
$$

与 GMM 不同，GMCM 中的 E 步并没有完全消除指数项之和上的对数。在进行期望最大化时，公式 15的最大化不会产生模型参数的封闭式更新。因此 M 步中需要基于梯度进行更新。

3.5 GMCM 参数的不可识别性

GMCM 存在参数不可识别性问题，即真实参数无法被唯一地识别，具体如下所述：

> 令 $\boldsymbol{U}$ 为 m 分量 GMC 分布生成的数据集，其真实参数集 $\Theta^* = \{\boldsymbol{\mu}^{l*}, \Sigma^{l*}, \alpha^{l*}\}_{l=1}^{m}$，则真实模型的对数似然表示为 $\mathcal{L}_\zeta(\Theta^* | \boldsymbol{U})$。定义另一个参数集 $\Theta = \{\boldsymbol{A}\boldsymbol{\mu}^{l*} + \boldsymbol{b}, \boldsymbol{A}^T \Sigma^{l*} \boldsymbol{A}, \alpha^{l*}\}_{l=1}^{m}$，其中 $\boldsymbol{A}$ 为任意对角正定矩阵，$\boldsymbol{b}$ 为实值向量。那么，$\mathcal{L}_\zeta(\Theta | \boldsymbol{U}) = \mathcal{L}_\zeta(\Theta^* | \boldsymbol{U})$

这说明有多个参数集对应同个对数似然，此时对数似然函数有多个极大值。Bilgrau 等人 [6] 注意到 GMCM 中这种形式的不可识别性。尽管没有证明这一点，他们提出了一个临时解决方案。然而，不可识别性问题在某些条件下仍然存在。这里提出了一种替代解决方案，在目标函数中加入高斯先验项。

> 将 $\boldsymbol{g} \in \mathbb{R}^d$ 和 $\boldsymbol{h} \in \mathbb{R}_+^d$ 表示为任意实值向量，后者为正值。如果 $\boldsymbol{g}$ 和 $\boldsymbol{h}$ 满足以下两个条件，则通过 $\Theta = \{\boldsymbol{\mu}^l, \Sigma^l, \alpha^l\}_{l=1}^{m}$ 参数化的 GMC 分布是可识别的。

$$
\sum_{l=1}^{m} \alpha^l \boldsymbol{\mu}_r^l = \boldsymbol{g}_r, \quad \forall r \in \{1, 2, \ldots, d\}
\tag{16}
$$

$$
\sum_{l=1}^{m} \left[\alpha^l \left(\Sigma_{rr}^l + \left(\boldsymbol{\mu}_r^l \right)^2 \right) \right] - \boldsymbol{g}_r^2 = \boldsymbol{h}_r, \quad \forall r \in \{1, 2, \ldots, d\}
\tag{17}
$$

$\boldsymbol{g}$ 和 $\boldsymbol{h}$ 的选择相当任意。为了方便起见，前者可以设置为全零向量，后者设置为全 1 向量。在参数估计期间，这些约束可以以高斯先验形式指，如公式 18以及 19所示。

$$
\mathcal{N} \left(\sum_{l=1}^{m} \alpha^l \boldsymbol{\mu}_r^l \middle| \boldsymbol{g}_r, \sigma \right), \quad \forall r \in \{1, 2, \ldots, d\}
\tag{18}
$$

$$\mathcal{N}\left(\sum_{l=1}^{m}\left[\alpha^{l}\left(\Sigma_{rr}^{l}+\left(\boldsymbol{\mu}_{r}^{l}\right)^{2}\right)\right]-\boldsymbol{g}_{r}^{2}|\boldsymbol{h}_{r},\sigma\right),\quad\forall r\in\{1,2,\ldots,d\}\tag{19}$$

这些先验的强度可以通过参数 σ 控制（值越大，先验越弱）。在实验过程中，$\sigma=0.01$ 的值可以很好地平衡这些先验和 GMCM 似然。

4 复现细节

4.1 与已有开源代码对比

在这个项目中，我主要借鉴了论文作者的开源代码。在深入理解和学习这些代码的基础上，我进行了一些必要的改进，梳理了代码逻辑。具体来说，我为代码增加了详细的注释，使其更易于理解，方便其他研究人员更好地了解其运行机制和功能。在源代码的基础上，我还发现了原论文中一些错误，并在报告中修正过来。在模型的初始化上，由于 GMCM 的结果受参数初始值影响，我增加了用 Kmeans++ 来进行参数初始化。此外，我还增加了一项新功能，即结果可视化。通过可视化方式展示结果，能更直观、更清晰地展示该论文的研究成果。因此，我在两个不同的数据集上实现了结果可视化，包括概率密度的等高线可视化以及聚类结果可视化，以便我们从多个角度和维度评估和理解模型性能。

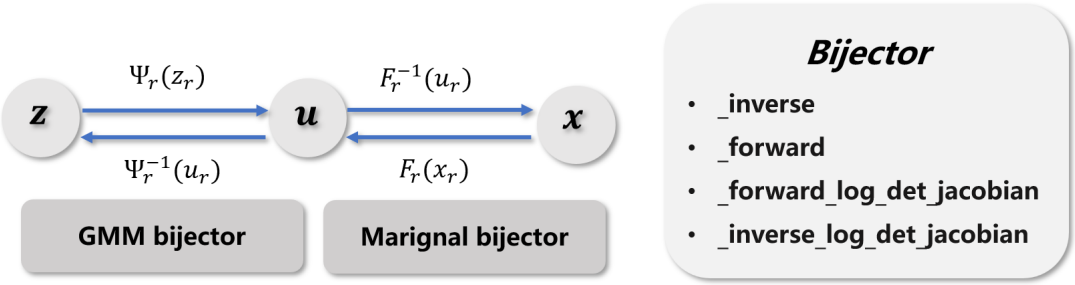
代码结构主要由以下三个部分组成：

- 双射器

  GMCM 可以看成是是变换分布，通过两个双射映射对基分布进行变换：第一个为基本 GMM 分布的边缘分布函数 $\Psi_r(\cdot)$，第二个是预先学习的边缘分布的逆函数 $F_r^{-1}(\cdot)$。所以在代码中，实现了两个双射器，包括从变量 $\boldsymbol{z}$ 到变量 $\boldsymbol{u}$ 的 GMM_bijector 以及从变量 $\boldsymbol{u}$ 到变量 $\boldsymbol{x}$ 的 Marginal_bijector，如图 6a所示。对于 Bijetor 类的实现，假设实现 x 到 y 的映射 $y=g(x)$，主要方法包括：

  - forward：正向映射函数，将随机变量 x 映射为 y。
  - inverse：反向映射函数，将随机变量 y 映射回 x。
  - forward_log_jacobian：正向映射的对数雅可比行列式，用于计算正向映射的概率密度函数。
  - inverse_log_jacobian：反向映射的对数雅可比行列式，用于计算反向映射的概率密度函数。

  在 GMM_bijector 中，inverse 函数的计算比较复杂，将求逆的过程替换为寻求表达式 $\boldsymbol{u}_r-\Psi_r(\boldsymbol{z}_r;\Theta^r)$ 的根，通过 Chandrupatla 算法求根并根据公式计算偏导值。由于需要通过 $\boldsymbol{z}$ 的概率密度来计算 $\boldsymbol{x}$ 的概率密度，如公式 9所示，所以两个双射器都需要实现对数雅可比行列式的计算，正向和反向的对数雅可比行列式互为相反数，所以只需要计算其中一个方向就可得到相反方向的对数雅可比行列式。



(a) 两个双射器　　(b) Bijector 类结构

图 6. 双射器的实现

- GMCM

  GMCM 的实现依赖于两个主要的类：GMCM 类和 GMC 类。GMCM 类是模型的主体，它内部包含一个 GMC 对象。GMC 类定义了 GMC 变换分布（即 GMM 变换），以及两个用于实现 GMCM 参数可识别性的正则项。此外，GMC 类还负责参数的初始化和模型的训练过程。在模型训练中，我们通过最大化直接似然来学习 GMC 分布，采用 Adam 优化器 [21] 进行最大化，其默认学习率为 1E−3。如果选择增加正则项，则将两个正则项加入到对数似然函数中，作为优化的目标函数。

  在 GMCM 类中，我们定义了 GMCM 的变换分布，包括预先学习的边缘分布以及基于 GMM 的边缘分布。如果指定了预处理变换，那么该变换会被包含在变换链中。预处理双射变换采用对数变换，通过减轻数据的重尾性质，以提高对边缘分布的学习效果。此外，GMCM 类还包括预先学习边缘分布的 learn_marginals 方法（通过 GMM 来学习每个维度的边缘分布，最佳构件数通过 BIC 准则来确定），以及拟合 GMCM 模型的方法。如果指定了预处理变换，我们会对训练数据和验证数据进行相应的变换。如果没有预先学习边缘分布，我们会调用 learn_marginals 方法来学习边缘分布。然后，根据学习到的边缘分布来初始化边缘变换。最后，我们初始化 GMC 对象，并调用 GMC 的训练函数来拟合分布。由于 GMCM 模型具有可边缘化的特性，我们在类中实现了返回指定维度的边缘 GMCM 模型的功能。与 GMM 类似，GMCM 存在隐变量来表明变量属于哪一个簇，因此我们可以计算给定变量下的分量概率，概率最大的分量即为变量所属的簇。因此，我们还实现了预测标签的聚类功能。

  总的来说，GMC 类是一个基础类，它定义了 GMC 分布的基本结构和方法，包括参数的初始化、模型的拟合等。而 GMCM 类是一个更高级的类，它包含了一个 GMC 对象，并在此基础上添加了更多的功能，如学习边缘分布、进行数据变换等，从而实现了完整的 GMCM 模型。这种设计使得代码更加模块化，易于理解和维护。

- 其他功能函数

  实现了基础方法，包括数据的划分，GMM 最佳模型的确定，多个模型参数与一个参数向量之间的转换，通过 kmeans++ 来初始化参数等。由于这些方法是可以重复调用的，所以选择将它们定义在类外部，这样可以提高代码的模块化和可重用性。

4.2 实验环境搭建

以下是搭建实验环境的步骤：

1. 安装 Python

   我们的代码需要 Python 3.7 或更高版本，可以从 Python 官方网站下载并安装。

2. 创建虚拟环境

   虚拟环境可以帮助你管理项目的依赖项，避免不同项目之间的依赖冲突。

3. 安装依赖性

   需要安装以下依赖项，可以通过 pip 来安装。

   - python>=3.7
   - numpy>=1.19.5
   - TensorFlow>=2.5.0
   - TensorFlow-Probability>=0.13.0
   - scikit-learn>=0.23.2
   - pandas>=1.1.3

4. 运行代码

   最后，可以运行文件夹中的代码。确保所有代码文件（如 main.py）在当前目录下。然后，可以在命令行中使用 Python 来运行你的代码：

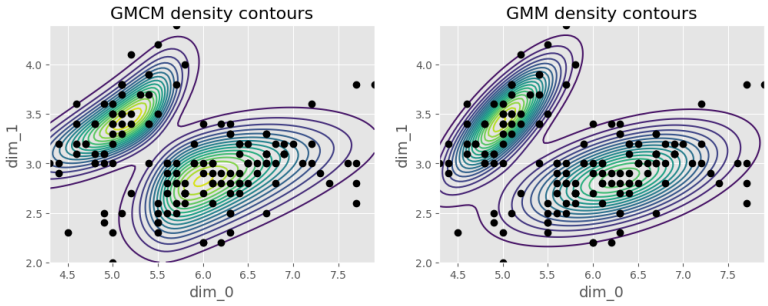
```
python main.py
```

5. 修改数据集（可选）

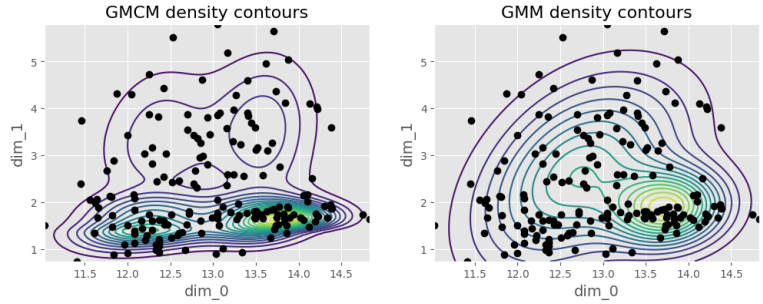
   在 main.py 中可以修改需要的数据集，要注意的是，如果不需要对数据进行对数变换的预处理，请注释 main.py 的第 42 行，取消第 43 行的注释。

5 实验结果分析

在实验部分，我选择了 wine 和 iris 这两个数据集进行研究。在这两个数据集上拟合 GMCM，并选择两个维度对概率密度函数进行了可视化展示，其中 wine 数据集的混合分量为 3，iris 数据集的混合分量为 2 并需要进行预处理变换（对数变换），其中结果如图 7所示。我们可以看出，与 GMM 相比，GMCM 得到的概率密度更能代表样本点的基本分布，在拟合上更加贴近点的分布边缘，且从形状上也放宽了正态性的假设，说明 GMCM 可以成为数据建模者手中的另一种多元建模工具，它提供了 GMM 的简单性和灵活性（边缘化），同时更具表现力。



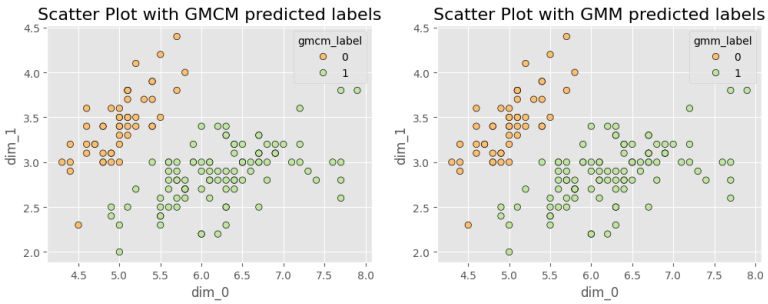
(a) iris 数据集



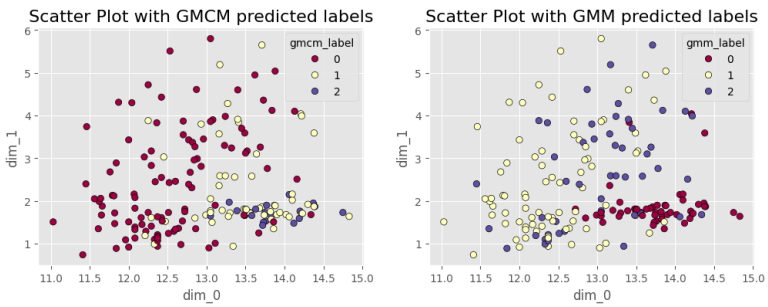
(b) wine 数据集

图 7. 二维密度图

另外，我们还对聚类的结果进行了可视化，如图 8所示。可以看到在两个数据集上 GMM 和 GMCM 的聚类结果没有太大的区别。在 iris 数据集上，由于数据比较简单，两种模型都能很好的进行聚类。在 wine 数据集上，由于簇之间没有明显的分界，所以我们肉眼上没办法判断哪个模型效果更好。但是总的来说，GMCM 如果用在聚类上也可以得到不错的效果的。



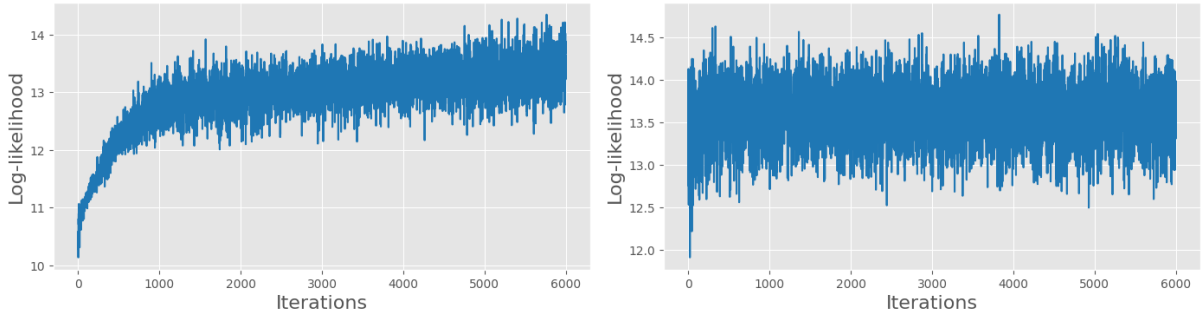
(a) iris 数据集



(b) wine 数据集

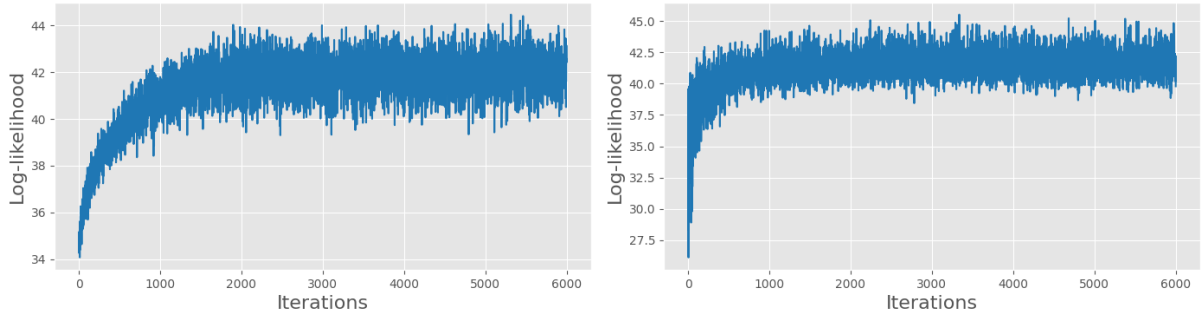
图 8. 聚类结果

为了体现在论文中增加的用 k-means++ 初始化方法对模型的影响，我们分别用随机初始化还有 k-means++ 初始化来进行实验，记录在两个数据集上的迭代过程中似然函数值的变换情况。结果如图 9和 10所示。可以明显的看到，相比于随机初始化，k-means++ 初始化下模型能够更快收敛。当我们使用 k-means 算法初始化 GMCM 时，可以将 k-means 的簇信息（包括中心点，每个簇的大小以及方差）用作 GMM 的初始参数。这样做的好处在于，k-means 的输出已经大致反映了数据的聚类结构，因此可以为 GMCM 提供一个相对合理的初始状态，从而加快 GMCM 的收敛速度。



(a) 随机初始化　　(b) k-means++ 初始化

图 9. iris 数据集上的迭代过程



(a) 随机初始化　　(b) k-means++ 初始化

图 10. wine 数据集上的迭代过程

因此，GMCM 是 GMM 的一个更具表现力的替代方案，它具有相同的参数化来编码多模态依赖结构。这使得 GMCM 不仅能够捕捉到数据的复杂分布特性，而且还能保持模型的参数可解释性和操作性。

6　总结与展望

本文中对一种高斯混合模型（Gaussian Mixture Model, GMM）的更具表现力的替代方法——高斯混合 Copula 模型（Gaussian Mixture Copula Model, GMCM）进行介绍。GMCM 的主要动机是能够表征具有多个非高斯模式的连续多元数据集的概率分布。对于这类数据集，GMM 可能过于严格，因为它对构成模式的正态性强加了基本假设。然而，GMCM 将边缘

分布的估计与不同模式中存在的依赖结构解耦，这种解耦允许我们相互独立地估计边缘分布，同时借用 GMM 的依赖结构。

尽管 GMCM 具有良好的表现，但仍存在一些问题。首先，参数的不可识别性已经得到广泛认可，但在之前的工作中并未得到广泛解决。在这篇论文中，作者通过增加可识别性先验的方式来缓解这个问题。其次，GMCM 似然估计的棘手问题。虽然这个固有问题仍然存在，但作者在论文中提出的高级数值方案和相关的解析偏导数对缓解该问题大有帮助。此外，作者还提出了一种正确的适用于 GMCM 的 EM 算法。之前的尝试提出了参数更新规则，但需要进行额外的检查和更正，以确保更新期间对数似然单调增加。作者在论文中指出，GMCM 无法享受到与 GMM 一样的 EM 算法带来的好处，因此为参数估计规定了直接似然最大化。为了让 GMCM 能够成为数据建模者手中的另一种多元建模工具，作者在论文中提出了将 GMCM 视为来自基本分布的一系列双射映射的视图，使其适用于现代概率编程框架并利用其内置的自动微分功能。在实验上，我们进行了两组数据集的可视化，验证了 GMCM 相比于 GMM 有更紧密的拟合效果，且脱离了高斯分布的假设。在实验上，我们进行了两组数据集的可视化，验证了 GMCM 相比于 GMM 有更紧密的拟合效果，且放宽了有关混合成分正态性的假设。

通过复现这篇论文，我对论文的整体内容有了更深入的理解。在后续的工作中，我计划使用其他概率密度估计工具来估计先验的边缘密度，以及探索如何从 GMCM 模型中提取出不同模式之间存在的依赖结构。

参考文献

[1] Peng An, Zhiyuan Wang, and Chunjiong Zhang. Ensemble unsupervised autoencoders and gaussian mixture model for cyberattack detection. *Information Processing & Management*, 59(2):102844, 2022.

[2] Tim Bedford and Roger M Cooke. Vines–a new graphical model for dependent random variables. *The Annals of Statistics*, 30(4):1031–1068, 2002.

[3] Ahmed BenSaïda, Sabri Boubaker, and Duc Khuong Nguyen. The shifting dependence dynamics between the g7 stock markets. *Quantitative Finance*, 18(5):801–812, 2018.

[4] Sakyajit Bhattacharya and Vaibhav Rajan. Unsupervised learning using gaussian mixture copula model. In *21st International Conference on Computational Statistics. Geneva, Switzerland*, 2014.

[5] Anders Ellern Bilgrau, Poul Svante Eriksen, and Martin Bøgsted. Gmcm: Fast estimation of gaussian mixture copula models. https://github.com/AEBilgrau/GMCM, 2017.

[6] Anders Ellern Bilgrau, Poul Svante Eriksen, Jakob Gulddahl Rasmussen, Hans Erik Johnsen, Karen Dybkær, and Martin Bøgsted. Gmcm: Unsupervised clustering and meta-analysis using gaussian mixture copula models. *Journal of Statistical Software*, 70(2), 2016.

[7] Jeff A Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International computer science institute*, 4(510):126, 1998.

[8] Claudia Czado. Pair-copula constructions of multivariate copulas. In *Copula Theory and Its Applications: Proceedings of the Workshop Held in Warsaw, 25-26 September 2009*, pages 93–109. Springer, 2010.

[9] Giovanni De Luca and Paola Zuccolotto. Dynamic tail dependence clustering of financial time series. *Statistical papers*, 58:641–657, 2017.

[10] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

[11] Pedro Díez. A note on the convergence of the secant method for simple and multiple roots. *Applied mathematics letters*, 16(8):1211–1215, 2003.

[12] Joshua V Dillon, Ian Langmore, Dustin Tran, Eugene Brevdo, Srinivas Vasudevan, Dave Moore, Brian Patton, Alex Alemi, Matt Hoffman, and Rif A Saurous. Tensorflow distributions. *arXiv preprint arXiv:1711.10604*, 2017.

[13] Gal Elidan. Copulas in machine learning. In *Copulae in Mathematical and Quantitative Finance: Proceedings of the Workshop Held in Cracow, 10-11 July 2012*, pages 39–60. Springer, 2013.

[14] Gabriel Escarela and Jacques F Carriere. Fitting competing risks with an assumed copula. *Statistical Methods in Medical Research*, 12(4):333–349, 2003.

[15] NI Fisher. Copulas. *Encyclopedia of Statistical Sciences*, 2, 2004.

[16] Edward W Frees and Ping Wang. Credibility using copulas. *North American Actuarial Journal*, 9(2):31–48, 2005.

[17] Christian Genest and Jock MacKay. The joy of copulas: Bivariate distributions with uniform marginals. *The American Statistician*, 40(4):280–283, 1986.

[18] Harry Joe. *Multivariate models and multivariate dependence concepts*. CRC press, 1997.

[19] Siva Rajesh Kasa, Sakyajit Bhattacharya, and Vaibhav Rajan. Gaussian mixture copulas for high-dimensional clustering and dependency-based subtyping. *Bioinformatics*, 36(2):621–628, 2020.

[20] Siva Rajesh Kasa and Vaibhav Rajan. Improved inference of gaussian mixture copula model for clustering and reproducibility analysis using automatic differentiation. *Econometrics and Statistics*, 22:67–97, 2022.

[21] D Kinga, Jimmy Ba Adam, et al. A method for stochastic optimization. In *International conference on learning representations (ICLR)*, volume 5, page 6. San Diego, California;, 2015.

[22] Ioannis Kosmidis and Dimitris Karlis. Model-based clustering using copulas with applications. *Statistics and computing*, 26:1079–1099, 2016.

[23] Qunhua Li, James B Brown, Haiyan Huang, and Peter J Bickel. Measuring reproducibility of high-throughput experiments. *Annals of Applied Statistics*, 5(3):1752–1779, 2011.

[24] Matthieu Marbac, Christophe Biernacki, and Vincent Vandewalle. Model-based clustering of gaussian copulas for mixed data. *Communications in Statistics-Theory and Methods*, 46(23):11635–11656, 2017.

[25] Roger B Nelsen. *An introduction to copulas*. Springer, 2006.

[26] A Pranklin. *Introduction to the Theory of Statistics*. 1974.

[27] Vaibhav Rajan and Sakyajit Bhattacharya. Dependency clustering of mixed data with gaussian mixture copulas. In *IJCAI*, pages 1967–1973, 2016.

[28] Michael Stanley Smith, Rubén Loaiza-Maya, and David J Nott. High-dimensional copula variational approximation through transformation. *Journal of Computational and Graphical Statistics*, 29(4):729–743, 2020.

[29] Moch Arief Soeleman, Aris Nurhindarto, Muslih Muslih, W Karis, Muljono Muljono, Farikh Al Zami, and R Anggi Pramunendar. Adaptive threshold for moving objects detection using gaussian mixture model. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(2):1122–1129, 2020.

[30] Ashutosh Tewari. On the estimation of gaussian mixture copula models. In *International Conference on Machine Learning*, pages 34090–34104. PMLR, 2023.

[31] Ashutosh Tewari, Michael J Giering, and Arvind Raghunathan. Parametric characterization of multimodal distributions with non-gaussian modes. In *2011 IEEE 11th international conference on data mining workshops*, pages 286–292. IEEE, 2011.

[32] Zhi Zeng and Ting Wang. Neural copula: A unified framework for estimating generic high-dimensional copula functions. *arXiv preprint arXiv:2205.15031*, 2022.

[33] Rui Zhou and Min Ji. Modelling mortality dependence: An application of dynamic vine copula. *Insurance: Mathematics and Economics*, 99:241–255, 2021.

[34] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International conference on learning representations*, 2018.