概率统计大作业

2023 年 12 月 17 日

姓名: 管昊
学号: 522030910072
指导教师: 熊德文
学院: 电子信息与电气工程学院

目录

摘要

本项目通过编写两份 Python 代码，使用 numpy 库生成随机数模拟混合高斯分布数据，并利用 matplotlib 库进行可视化，研究了不同参数对分布形状的影响。第一份代码通过直方图可视化混合高斯分布，考察均值、标准差和权重参数的变化对分布的影响。第二份代码则用于验证中心极限定理，生成不同数量的数据样本，并分析其对样本均值分布的影响。报告详细阐述了代码的实现机制，并结合理论解释了数据分布的特点。

关键词：样本均值，样本方差，混合高斯分布，中心极限定理。

1 准备工作

首先，编写 python 代码 test.py 代码生成 QQ 图（Quantile-Quantile 图）验证 NumPy 生成的正态分布随机数是否合理，过程如下。

代码首先生成了 1000 个均值为 0，标准差为 1 的正态分布随机数，然后计算并显示了这些随机数的平均值和标准差，接着绘制了这些数据的直方图和 QQ 图来视觉上验证其正态性。直方图显示了数据的分布，而 QQ 图则用于检查样本数据是否符合正态分布的理论分位数。如图 1 所示，QQ 图上的点大致沿着参考线排列，故生成的随机数是合理的正态分布。下面开始实验。

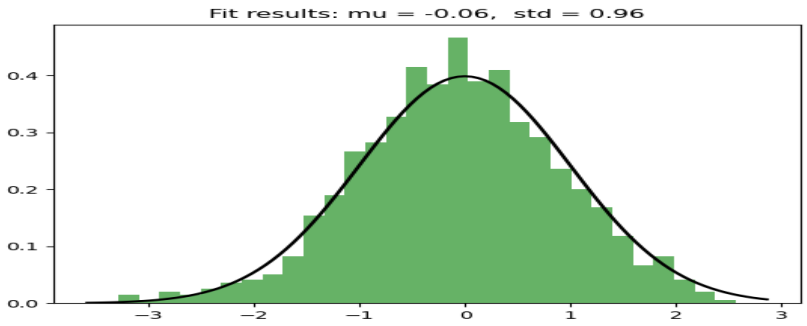

图 1: 验证正态分布的随机数

2 混合高斯分布的模拟与可视化

2.1 混合高斯分布

高斯混合模型就是正态分布曲线精确地量化事物，它是一个将事物分解为若干的基于正态分布曲线形成的模型。混合高斯分布是一种统计分布，它由多个正态分布的组合构成。这种分布在统计学和机器学习领域内非常常见，特别是在处理具有多峰（即数据分布在多个区域集中）的复杂数据集时。下面是混合高斯分布的主要特征：

基本定义 混合高斯分布由多个高斯分布混合而成。每个高斯分布称为一个组分（component）。每个组分有其参数：每个高斯分布具有自己的均值（mean）和标准差（standard deviation）。每个高斯分布组分都有一个权重，表示该组分在整个混合分布中的比重。

数学表示 混合高斯分布可以用以下数学公式表示：

$$p(x) = \sum_{i=1}^{K} \pi_i \mathcal{N}(x|\mu_i, \sigma_i^2)$$

其中：

- $p(x)$ 是混合高斯分布。
- K 是高斯分布组分的数量。
- π_i 是第 i 个组分的权重，且一般地 $\sum_{i=1}^{K} \pi_i = 1$。
- $\mathcal{N}(x|\mu_i, \sigma_i^2)$ 是具有均值 μ_i 和方差 σ_i^2 的高斯分布。

特征

- 灵活性：由于可以包含多个组分，混合高斯分布能够模拟各种形状的数据分布，尤其是非单峰分布。
- 适用性：在现实世界的数据分析中，混合高斯模型被广泛应用于聚类、分类以及概率密度估计等任务。
- 参数估计：混合高斯模型的参数通常通过期望最大化（EM）算法进行估计。

2.2 理论分析

推导得出混合高斯分布的期望与方差：

$$\begin{aligned} E(Z) &= E(X) + E(\eta E(Y)) \\ &= \mu_1 + p\mu_2, \\ D(Z) &= D(X) + D(\eta^2 E(Y^2)) - E^2(\eta Y) \\ &= D(X) + E(\eta^2)E(Y^2) - E^2(\eta Y) \\ &= \sigma_1^2 + p\sigma_2^2 + (1-p)p\mu_2^2. \end{aligned}$$

因此 Z 的概率密度函数图像（和频率直方图）应当呈现出两个峰值，分别位于 $\mu_1, \mu_1 + \mu_2$ 处且峰值大小取决于 p, σ_1, σ_2 称均值在 $\mu_1, \mu_1 + \mu_2$ 处的峰为 X 峰、V 峰，则可以知道：p 越小、σ_1 越小，则 X 峰越高；p 越大，$\sigma_1^2 + \sigma_2^2$ 越小，则 V 峰越高

2.3 实验目的

探究混合高斯分布的参数对其频率分布直方图的影响

2.4 代码解析

任务一代码[1]定义了一系列函数，用于生成混合高斯分布样本并绘制其直方图，下面是对代码的具体解析。

[1]详细代码在附录一中

2.4.1 函数定义

- `generate_mixed_gaussian`: 此函数接受两组均值（`miu1` 和 `miu2`）、标准差（`std1` 和 `std2`）和混合参数（`yitap`），生成一个混合高斯分布的样本。
- `plot`: 该函数负责将生成的样本绘制成直方图。
- `plot_mixed_gaussian`: 它结合上述两个函数，生成样本并绘制图形。

2.4.2 参数变化

- 原始数据：设置为均值为-10 和 10，标准差为 1 和 2，权重为 0.8。
- 变化均值：提高第二个高斯分布的均值到 20，观察峰值的移动。
- 变化标准差：分别增加第二个高斯分布的标准差为 4 或第一个高斯分布的标准差为 4，观察分布的扩散。
- 变化权重：调整混合比例为 0.5 和 1，观察两个高斯分布混合后峰值的变化。

2.5 结果分析

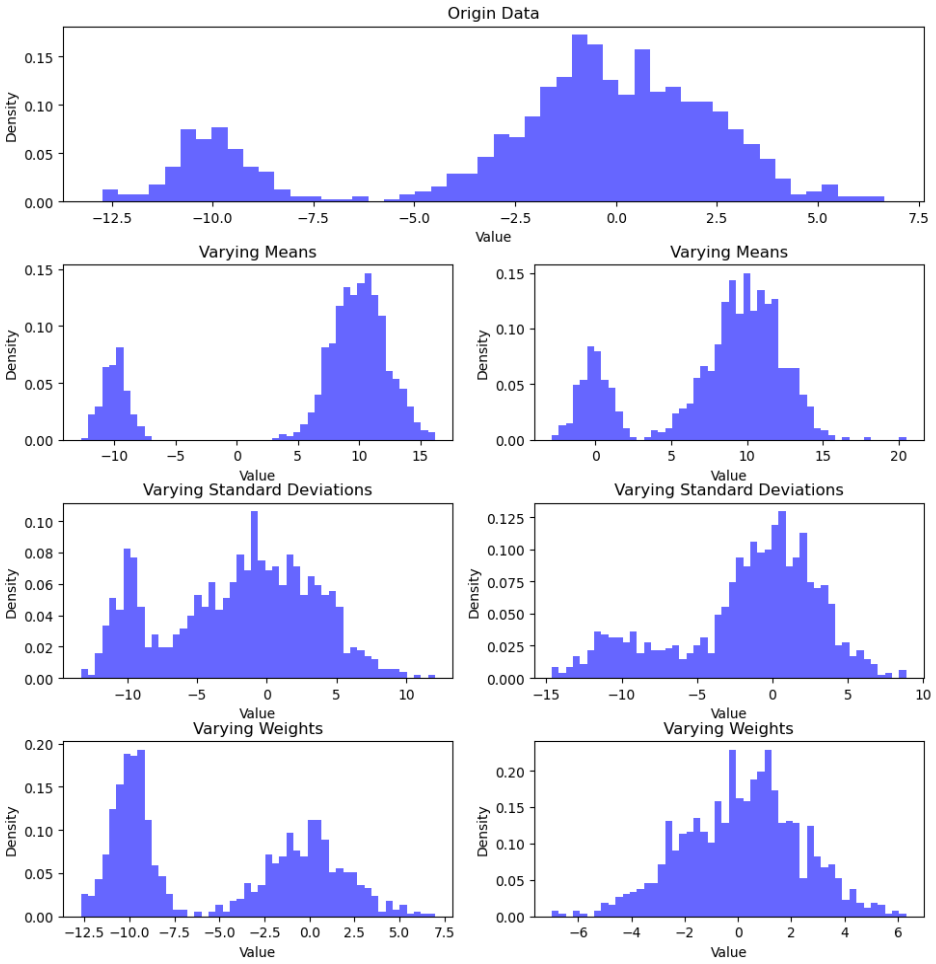

图 2: 实验结果

从这四张图中可以看出，X 峰（即图中左侧峰）的中心位置大致处于 mu_1 处，V 峰（即图中右侧峰）的中心位置大致处于 $\mu_1 + \mu_2$ 处。这与理论分析相一致。

上图中展示的四个直方图，分别代表不同参数的变化，以下是对每个直方图的详细分析：

- **原始数据（第一行）**此直方图显示了一个双峰分布，表示存在两个具有不同均值的高斯成分。两个显著的峰值表明这两个分布的均值相距较远，因此在值的分布上形成了两个中心点。这些峰值紧密聚集表明两个成分的标准差都较小。

- **变化的均值（第二行）**如图可以看出：左侧峰对应 μ_1，右侧峰对应 $\mu_1 + \mu_2$，且改变 μ_1 会改变两个峰的位置，而改变 $\mu_1 + \mu_2$ 只改变第二个峰位置。与原始数据相比，其中一个峰值向右移动，表明其中一个高斯成分的均值增大。这种移动在混合中创建了一个向右偏斜的单个峰值。如果原始成分的均值接近，增加其中一个的均值会使峰值分离，但如果该成分也有较大的权重或更多样本，整体分布仍然会显示出向较大均值倾斜的中心趋势。

- **变化的方差（第二行）**此直方图呈现了一个更为分散的分布，有一个主要峰值和一个较小的峰值。增加一个成分的标准差会使数据点在均值周围更广泛地分布，导致直方图看起来更加扁平和宽敞。这表示分布中存在更大的变异性。当我们修改方差，分析组分中高斯分布的方差对混合高斯分布的影响时。可以发现，两者的方差影响两个峰的宽度。当两个高斯分布的方差过大时，混合高斯分布的两峰逐渐重叠。且可以看出，σ_1 影响两个峰的宽度，而 σ_2 仅影响 V 峰的宽度。这与理论分析也一致。

- **变化的权重（第三行）**直方图显示了一个双峰但不对称的分布。调整高斯成分的权重会改变它们对最终混合分布的贡献。如果对应较小峰值的成分权重减小，那么在混合分布中这个峰值就会变得不那么明显。可见的主峰和次峰表明一个成分相对于另一个成分具有更大的权重。当 p 等于 1 时，如右下图所示，混合高斯分布将退化成为以 $\mu_1 + \mu_2$ 为均值，以 $\sigma_1^2 + \sigma_2^2$ 为方差的高斯分布

2.6 结论

通过实验，我们得出以下结论呢：调整混合高斯分布的均值、方差（标准差）和系数（权重）会以不同的方式影响数据的分布。以下是每个参数调整对数据分布造成的影响：

- **均值 μ（Mean）**：调整单个高斯分布的均值会改变该分布数据点的中心位置。如果两个高斯分布的均值彼此接近，混合分布会倾向于表现出单峰特征。如果均值相差很远，混合分布会表现出多峰特征，在我设置的参数下，左侧峰对应 μ_1，右侧峰对应 $\mu_1 + \mu_2$。均值的变化也可以改变分布的对称性，特别是在不同高斯分布具有不同权重的情况下。

- **方差 σ^2（Variance）**：方差或标准差的大小决定了单个高斯分布的数据点围绕其均值的散布程度。较小的方差意味着数据点更集中，导致峰值更尖锐。较大的方差意味着数据点更分散，导致峰值更平坦和更宽。不同分布的方差差异可以导致混合分布中出现宽窄不一的峰值。

- **系数 η（Coefficient）**：在混合高斯分布中， 取 1 的概率 p 是决定各个分量相对重要性的关键因素。当 p 值增加时，对应的高斯分量在整个混合分布中的影响增强，但由于公式中 μ_1 对应的高斯分布前面的系数恒为 1，使得 μ_1 对应的高斯分布的权重永远大于 μ_2 对应的高斯分布。当 p=1 时，混合高斯分布退化成为高斯分布。p 的选择和调整能够显著影响混合高斯分布的整体形态，对于精确地捕捉和描述数据分布特性至关重要

总结来说，改变混合高斯分布中的参数会导致数据分布的形状显著变化，这些变化反映了不同高斯组分对最终分布形状的不同影响。调整均值、标准差和权重参数可以控制分布的中心位置、分散程度和各组分的相对重要性，从而适应不同的数据分析需求。

在实践中，通过细致地调整这些参数，可以控制混合高斯分布的整体形状，以模拟复杂的现实世界数据分布。例如，可以通过调整参数来拟合具有特定统计特性的实验数据，或者在机器学习中用于对特定数据集进行建模。

3 中心极限定理的应用

3.1 中心极限定理

中心极限定理（Central Limit Theorem, CLT）是概率论中的一个基本定理，它解释了为什么在现实世界中许多分布看起来都接近正态分布，即使它们的底层分布不是正态的。这组定理是数理统计学和误差分析的理论基础，指出了大量随机变量之和近似服从正态分布的条件。这个定理的主要内容可以总结如下：

3.1.1 定理描述

假设 $X_1, X_2, \ldots, X_n$ 是一组独立同分布的随机变量，它们具有共同的期望 μ 和方差 $\sigma^2 (\sigma^2 > 0)$。考虑这些随机变量的和 $S_n = X_1 + X_2 + \ldots + X_n$。

中心极限定理指出，随着 n 的增加，标准化变量

$$Z_n = \frac{S_n - n\mu}{\sigma\sqrt{n}}$$

的分布趋近于标准正态分布 $N(0,1)$。也就是说，无论原始随机变量 X_i 的分布如何，只要它们独立同分布且具有有限的期望和方差，随机变量和的分布都会趋向正态分布。

3.1.2 应用

中心极限定理说明，对于样本量足够大的样本均值，无论总体分布如何（除了极少数情况），其分布都将近似正态分布。这使得我们可以使用正态分布的性质来进行样本均值的推断，即使总体分布未知。这在抽样调查、品质控制等领域中非常有用。

3.1.3 限制

- 中心极限定理不适用于没有有限方差的分布，例如柯西分布。
- 样本数量 n 需要足够大，这个“大”取决于原始分布的特性。如果原始分布已经接近正态分布，那么即使 n 较小，中心极限定理的近似也可能是合理的。
- 中心极限定理提供的是一个渐进结果，它告诉我们 n 趋于无穷时的行为，但并不给出对于特定 n 的分布的精确描述。

中心极限定理的一个经典应用是在抽样调查和实验设计中。例如，如果从任何一个总体中随机抽取样本，计算它们的平均值，然后重复这个过程很多次，那么这些平均值的分布就会形成一个正态分布。这就是为什么正态分布在统计学中如此重要，因为它允许我们进行假设检验和构建置信区间，即使我们不知道总体的确切分布。

中心极限定理的美妙之处在于它提供了一种强大的途径，通过使用样本均值来估计总体均值，并且随着样本量的增大，估计的准确度越高。在这次实验中，我们通过模拟混合高斯分布来观察中心极限定理。

3.2 实验目的

探究中心极限定理在混合高斯分布随机数上的应用和效果。通过计算机生成的随机数验证随着样本量 n 的增加，样本均值的分布将越来越接近正态分布。

3.3 代码解析

任务二代码[2]演示了中心极限定理（CLT）的应用。通过多次抽取样本并计算均值，可以观察到随着样本量的增加，样本均值的分布越来越接近正态分布。下面是对代码的解释。

3.3.1 函数定义

- `generate_mixed_gaussian`: 与第一份代码中的同名函数功能相同。
- `problem2`: 该函数实现了 CLT 的模拟过程。它通过多次生成混合高斯分布的样本，计算样本均值，并绘制其分布的直方图。

3.3.2 参数设置

代码中预设了一个样本量列表 `nlist`，其中包含从 2 到 5000 不等的样本量。通过改变样本量，可以观察样本均值分布的变化。

3.4 结果分析

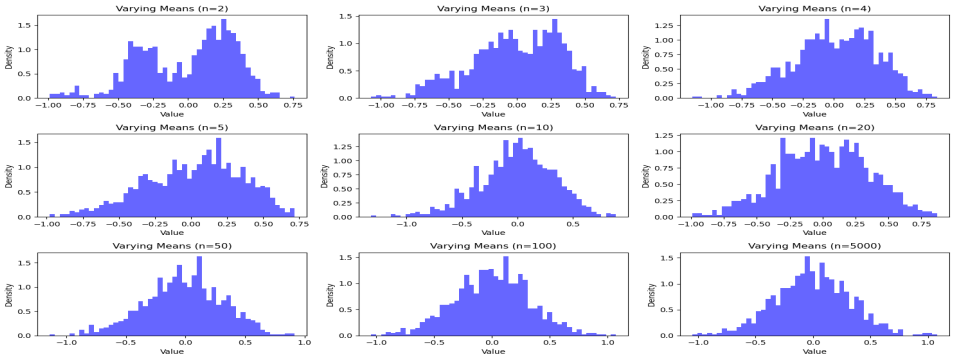

图 3: 实验结果

在上图中，有一系列直方图展示了不同样本大小（n）的混合高斯分布的统计特性。从左上角到右下角，图像分别对应着样本大小为 2、3、4、5、10、20、50、100 和 5000 的情况。以下是对这些直方图的分析：

- **样本大小的影响**：直方图的形状变化揭示了样本大小（n）对估计混合高斯分布的影响。每个直方图反映了通过模拟混合高斯分布得到的统计量的分布，其中统计量通常是样本均值。

[2]详细代码见附录 2

- **小样本大小（n=2, 3, 4, 5）**：对于较小的样本大小（n=2 到 5），直方图呈现出较多的随机性和不规则性。在 n = 2 时图像依然可以看到较为明显的两个峰值，这是因为当样本量很小时，随机性对结果的影响较大，因此样本倾向于表现为其本身的分布，图像可能会有所不同，导致统计量的分布看起来比较粗糙且不平滑。
- **中等样本大小（n=10, 20）**：当样本量增加到 10 和 20 时，直方图开始显现出更平滑且更接近正态分布的形状。中心极限定理（CLT）说明，即使原始数据不是正态分布的，样本均值的分布随着样本量的增加而趋向于正态分布。这里的直方图开始反映这一理论，峰值更加明显，且分布变得对称。
- **大样本大小（n=50, 100, 5000）**：对于更大的样本量（n=50, 100, 5000），直方图的形状变得非常平滑，并且明显呈现出了正态分布的特征，即单一的对称峰和两侧的尾部逐渐减少。这符合中心极限定理的预期，即样本均值将形成一个钟形的正态分布。对于 n=5000 的情况，直方图的形状几乎完美地符合正态分布的典型形态，表明了大数定律和中心极限定理在这里的应用。

3.5 结论

通过观察不同样本大小的混合高斯分布，我们可以看到，随着样本量的增加，样本均值的分布越来越趋于正态分布。这些直方图是中心极限定理在混合高斯分布上应用的直观展示。即使原始数据可能不是正态分布的，样本均值也会随着样本量的增加而趋近于正态分布。这一现象是统计学中非常重要的，因为它允许我们对大型样本集的性质进行推断，即使我们只能观察到来自该分布的有限样本。

4 遇到的问题和解决方法

如何形成混合高斯分布 在作业的刚开始我本想通过根据 分别生成一部分 X 正态分布的数据和 Y 正态分布的数据 concatenate 来获得混合正态分布，但尝试后发现这样生成的效果不太好，于是改用分别使用 numpy 中的 random.normal 函数生成 X 和 Y 的值后调用 numpy 中的 random.binomial 函数随机生成 η 的数值后使用等式 $Z = X + Y * \eta$ 求得 Z 的值。

验证中心极限定理的过程中结果不明显的难题 为了验证中心极限定理，我对于不同的数据量的生成了 50 次混合高斯分布后画出他们的均值的频率直方图，此时我才用的参数依然是第一个实验中的原始参数，但由于原始参数两个样本均值相差太小，频率直方图两个峰之间距离较近导致不同数据量画出样本均值的频率直方图之后差距不太大，实验现象不明显。后来我通过调整样本均值的大小，成功获得了较为明显的实验图像——当数据量增大时，图像越来越像正态分布的图像，如此很好地验证了中心极限定理。

对题目的误解 在刚开始做大作业时，我没有太搞懂任务二的要求，在询问同学和了解了题目的目的后缕清了题目的思路，较好地完成了代码实现。

5 总结和感想

本报告通过两份代码的分析和图形结果讨论，得出以下结论：混合高斯分布可以通过调整均值、标准差和混合比例来控制分布形状。中心极限定理对样本均值分布具有普遍适用性，验证了即使原

始数据不是正态分布，样本均值的分布也会随着样本量增加而趋于正态分布。这些发现对理解复杂数据分布有重要意义，同时展示了统计学在数据分析中的应用。

完成这次大作业后，我深刻地理解了混合高斯分布和中心极限定理的重要性，意识到数据科学和统计学中不可或缺的一部分，它帮助我们理解数据背后的规律和趋势，为决策提供了有力的支持。

此外，在这次项目中，我学会了如何使用混合高斯分布来模拟和描述复杂的数据分布。通过调整均值、标准差和混合比例，我能够控制分布的形状，这对于实际问题中的数据建模非常有用。我也学会了如何利用中心极限定理来理解样本均值的分布，这对于推断总体参数和进行假设检验至关重要。

最后，这次项目还让我更深入地了解了编程和数据可视化的技能。通过编写代码和绘制图表，我能够清晰地展示我的分析过程和结果，使得报告更具可读性和说服力。

总的来说，这次项目让我不仅学到了有关概率统计和数据分析的知识，还培养了解决问题和沟通复杂概念的能力。我相信这些技能将在未来的学习中发挥重要作用，帮助我更好地应对数据驱动的挑战。因此，我认为学好概率统计是非常有意义的，它为我们提供了解世界和做出明智决策的工具和方法。

A 附录一：任务一的问题和代码

设定不同的参数 $\mu_1, \sigma_1^2, \mu_1, \sigma_2^2, p$ 生成 5000（也可以是别的数）个混合高斯分布的随机数并画出其频率分布直方图。代码如下：

Listing 1: **任务一代码**

```
import numpy as np
import matplotlib.pyplot as plt

# 创建一个4行2列的子图布局，第一行单独一个大图
fig = plt.figure(figsize=(10, 12))
gs = fig.add_gridspec(4, 2)  # 4 rows, 2 columns grid
axes = [fig.add_subplot(gs[0, :])]  # First row spans all columns
axes += [fig.add_subplot(gs[i, j]) for i in  range(1, 4) for j in  range(2)]  # Remaining rows

# 定义一个函数，用于生成混合高斯分布的样本并绘制直方图
def plot_mixed_gaussian(miu1, std1, miu2, std2, yitap, title, seq):
    samples = generate_mixed_gaussian(miu1=miu1, std1=std1, miu2=miu2, std2=std2, yitap=yitap, n
        =1000)
    plot(samples, title, seq)

# 定义绘制直方图的函数
def plot(samples, title, seq):
    # 绘制频率直方图
    axes[seq].hist(samples, bins=50, density=True, alpha=0.6, color='blue')
    axes[seq].set_title(title)
    axes[seq].set_xlabel('Value')
    axes[seq].set_ylabel('Density')

# 生成混合高斯分布样本的函数
def generate_mixed_gaussian(n, miu1, std1, miu2, std2, yitap):
    X = np.random.normal(miu1, std1, n)
    Y = np.random.normal(miu2, std2, n)
    yita_samples = np.random.binomial(1, yitap, n)
    Z = X + Y * yita_samples
    return Z

# 使用定义的函数绘制不同条件下的混合高斯分布
plot_mixed_gaussian(miu1=-10, std1=1, miu2=10, std2=2, yitap=0.8, title="Origin Data", seq=0)
plot_mixed_gaussian(miu1=-10, std1=1, miu2=20, std2=2, yitap=0.8, title="Varying Means", seq=1)
plot_mixed_gaussian(miu1=0, std1=1, miu2=10, std2=2, yitap=0.8, title="Varying Means", seq=2)
plot_mixed_gaussian(miu1=-10, std1=1, miu2=10, std2=4, yitap=0.8, title="Varying Standard Deviations
    ", seq=3)
plot_mixed_gaussian(miu1=-10, std1=2, miu2=10, std2=2, yitap=0.8, title="Varying Standard Deviations
    ", seq=4)
plot_mixed_gaussian(miu1=-10, std1=1, miu2=10, std2=2, yitap=0.5, title="Varying Weights", seq=5)
plot_mixed_gaussian(miu1=-10, std1=1, miu2=10, std2=2, yitap=1, title="Varying Weights", seq=6)

# 调整子图布局
plt.tight_layout()
```

```
# 显示图形
plt.show()
```

B 附录二：任务二的问题和代码

对应自己设定的参数，EZ，DZ 为高斯分布的期望与方差，用计算机生成 1000 每组 n 个）混合高斯分布的随机数 $Z_{i,j}$ 并分别计算

$$U_i = \frac{1}{\sqrt{nDZ}}\{\Sigma_{j=1}^{n} Z_{1,j} - nE(Z)\}; \quad i = 1, 2, \ldots, 1000$$

画出频率直方图。

实现代码如下：

Listing 2: **任务二代码**

```
import numpy as np
import matplotlib.pyplot as plt

# 生成混合高斯分布样本的函数
def generate_mixed_gaussian(n, miu1 = -2, std1 = 1, miu2 = 2, std2 = 1, yitap = 0.5):
    X = np.random.normal(miu1, std1, n)  # 生成第一个高斯分布的样本
    Y = np.random.normal(miu2, std2, n)  # 生成第二个高斯分布的样本
    yita = np.random.binomial(1, yitap, n)  # 生成混合的伯努利试验
    Z = X + Y * yita  # 混合两个分布
    return Z

# 创建一个3行3列的子图布局
fig, axes = plt.subplots(3, 3, figsize=(15, 8))

# 用于模拟不同样本量下的分布并绘图的函数
def problem2(n, seq, miu1, std1, miu2, std2, yitap, title):
    # 计算混合分布的期望值和方差
    EZ = miu1 + miu2 * yitap
    DZ = (std1 ** 2 + miu1 ** 2) + yitap * (std2 ** 2 + miu2 ** 2) - EZ ** 2

    U = []

    # 生成1000个样本并计算每个样本的Ui
    for i in  range(1000):
        samples = generate_mixed_gaussian(n=n, miu1=miu1, std1=std1, miu2=miu2, std2=std2, yitap=
            yitap)
        Ui = (samples.mean() * n - n * EZ) / np.sqrt(n * DZ)
        U.append(Ui)

    # 绘制Ui值的直方图
    axes[seq // 3, seq % 3].hist(U, bins=50, density=True, alpha=0.6, color='blue')
    axes[seq // 3, seq % 3].set_title(f"{title} (n={n})")
    axes[seq // 3, seq % 3].set_xlabel('值')
    axes[seq // 3, seq % 3].set_ylabel('密度')
```

```
34
35 # 不同的样本量进行测试
36 nlist = [2, 3, 4, 5, 10, 20, 50, 100, 5000]
37
38 # 对每个样本量运行模拟
39 for i in  range(9):
40     problem2(nlist[i], i, miu1=-10, std1=1, miu2=10, std2=2, yitap=0.8, title="变化的均值")
41
42 # 调整子图布局
43 plt.tight_layout()
44
45 # 展示图形
46 plt.show()
```