人工智能作业报告

SY2303206 张一辉

摘要

本实验内容为基于已有的身高数据，使用 EM 算法来对混合高斯模型进行参数估计并且进行预测。实验通过已有的代码生成两个由男生身高和女生身高两个正态分布构成的混合高斯模型，通过 EM 算法，对模型的参数进行估计。

引言

极大似然估计是一种常用的参数估计方法，常用于统计学、机器学习、人工智能等领域。其基本思想是：已知样本的分布，通过估计参数来确定分布的概率密度函数或概率质量函数。具体来说，假设样本 $x_1, x_2, \ldots, x_n$ 来自某个分布 P_θ，其中 θ 是分布的参数。极大似然估计的目标是找到一个参数 $\hat{\theta}$，使得观测到的样本 $x_1, x_2, \ldots, x_n$ 出现的概率最大。EM 算法是一种常用的统计学习算法，用于解决含有隐变量的概率模型参数估计问题。其基本思想是，通过迭代的方式，不断地估计参数，从而找到最优的参数值。

研究方法

高斯模型:

高斯模型是一种常用的概率模型，也称为正态分布模型或高斯分布模型。在高斯模型中，随机变量的取值服从高斯分布（也称为正态分布），其概率密度函数形式为：

$$f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

其中，μ 是分布的均值，σ^2 是分布的方差，x 是随机变量的取值。

高斯模型的特点是对称且钟形的曲线，曲线的峰值对应于分布的均值。当 σ 较小时，曲线较陡峭；当 σ 较大时，曲线较平缓。

高斯混合高斯模型是一种常用的概率模型，它是若干个高斯模型的加权和，表示数据集中可能存在多个不同的高斯分布。具体来说，混合高斯模型可以表示为：

$$f(x) = \sum_{i=1}^{k} \pi_i f_i(x)$$

其中，$f_i(x)$ 表示第 i 个高斯分布的概率密度函数，π_i 表示第 i 个高斯分布的权重，$\sum_{i=1}^{k} \pi_i = 1$。混合高斯模型的特点是可以对数据集中的不同类别进行建模，混合高斯模型的参数估计通常使用 EM 算法进行求解

EM 算法原理:

EM 算法是求解混合高斯模型的一种常用方法。混合高斯模型中包含多个高斯分布，每个高斯分布都有自己的均值和方差。EM 算法通过迭代计算每个高斯分布的均值和方差，以及每个高斯分布在总体中的权重。

EM 算法求解混合高斯模型的基本步骤如下:

初始化模型参数: 首先需要对模型参数进行初始化，包括每个高斯分布的均值、方差和权重。在数据不复杂时通常可以随机抽取数值作为均值，根据经验设定较为符合的方差，每个高斯分布的权重可以初始化为均等分布，即每个高斯分布的权重都为 $1/k$。

E 步骤: 对于每个数据点x_i，计算其属于每个高斯分布的概率 $p(z_i = j|x_i, \theta)$，其中 $j \in [1, k]$，θ 表示当前模型参数，即每个高斯分布的均值、方差和权重。

$$p(z_i = j \mid x_i, \theta) = \frac{\pi_j f_j(x_i)}{\sum_{l=1}^{k} \pi_l f_l(x_i)}$$

其中，$f_j(x_i)$ 表示第 j 个高斯分布在 x_i 处的概率密度函数值，π_j 表示第 j 个高斯分布的权重

M 步骤: 使用 E 步骤计算出的每个数据点属于每个高斯分布的概率，重新估计每个高斯分布的均值、方差和权重。

每个高斯分布的均值可以根据加权平均的方法进行估计:

$$\mu_j = \frac{\sum_{i=1}^{n} p(z_i = j \mid x_i, \theta) x_i}{\sum_{i=1}^{n} p(z_i = j \mid x_i, \theta)}$$

每个高斯分布的方差可以根据加权平均的方法进行估计:

$$\sigma_j^2 = \frac{\sum_{i=1}^{n} p(z_i = j \mid x_i, \theta)\left(x_i - \mu_j\right)^2}{\sum_{i=1}^{n} p(z_i = j \mid x_i, \theta)}$$

每个高斯分布的权重可以根据所有数据点属于该高斯分布的概率之和除以数据总数进行估计:

$$\pi_j = \frac{\sum_{i=1}^{n} p(z_i = j \mid x_i, \theta)}{n}$$

终止条件: 重复执行 E 步骤和 M 步骤，直到模型收敛，即模型参数不再发生明显变化或达到最大迭代次数。通常可以选定一定最大迭代次数或者使用对数似然函数来检查模型收敛，如果对数似然函数的增量小于一个设定的阈值，就认为模型收敛。

预测: 在模型收敛后，对于一个新的数据点 x，可以通过计算其在所有高斯分布中的概率来进行分类，选择概率最大的高斯分布作为其所属的类别。

实验内容

EM 算法具体推导:

该问题中，共有两个高斯分布，分别设为 $N(\mu_1,\sigma_1^2)$ $N(\mu_2,\sigma_2^2)$

即 $f_1(x)=\frac{1}{\sqrt{2\pi}\sigma_1}e^{\left(\frac{-(x-\mu_1)^2}{2\sigma_1^2}\right)}$ $f_2(x)=\frac{1}{\sqrt{2\pi}\sigma_2}e^{\left(\frac{-(x-\mu_2)^2}{2\sigma_2^2}\right)}$

$$f(x)=\sum_{i=1}^{k}\pi_i f_i(x)=\pi_1 f_1(x)+\pi_2 f_2(x)$$

初始化：

$\pi_1=\pi_2=\frac{1}{k}=0.5$

μ_1 与 μ_2 从样本中随机抽取

$\sigma_1=\sigma_2=1$

E 步：对于 x_i

x_i 属于分布1的概率 $\pi_{i1}=\frac{p(x_i|f_1(x))\cdot\pi_1}{p(x_i|f_1(x))\cdot\pi_1+p(x_i|f_2(x))\cdot\pi_2}$

x_i 属于分布2的概率 $\pi_{i2}=\frac{p(x_i|f_2(x))\cdot\pi_2}{p(x_i|f_1(x))\cdot\pi_1+p(x_i|f_2(x))\cdot\pi_2}$

M 步：根据 E 步计算的结果重新估计每个高斯分布

$$\mu_1=\frac{\pi_{11}x_1+\pi_{21}x_2+\cdots+\pi_{N1}x_N}{\pi_{11}+\pi_{21}+\cdots+\pi_{N1}}$$

$$\sigma_1=\sqrt{\frac{\pi_{11}(x_1-\mu_1)^2+\cdots+\pi_{N1}(x_N-\mu_1)^2}{\pi_{11}+\pi_{21}+\cdots+\pi_{N1}}}$$

$$\mu_2=\frac{\pi_{12}x_1+\pi_{22}x_2+\cdots+\pi_{N2}x_N}{\pi_{12}+\pi_{22}+\cdots+\pi_{N2}}$$

$$\sigma_2=\sqrt{\frac{\pi_{12}(x_1-\mu_2)^2+\cdots+\pi_{N2}(x_N-\mu_2)^2}{\pi_{12}+\pi_{22}+\cdots+\pi_{N2}}}$$

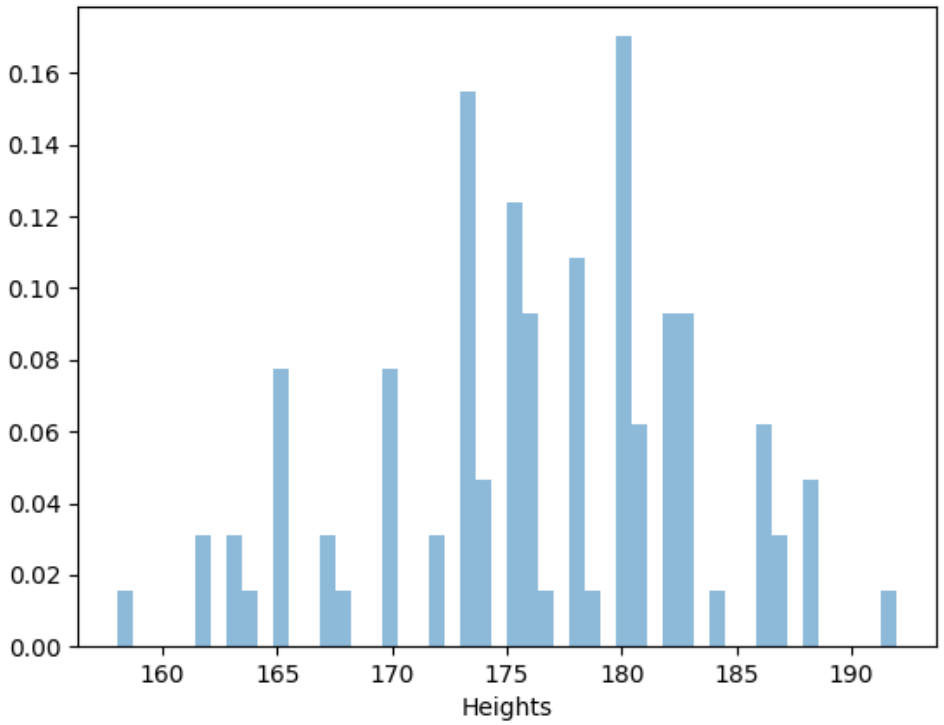
同时更新每个分布的概率

$$\pi_1=\frac{\pi_{11}+\cdots+\pi_{N1}}{N}$$

$$\pi_2=\frac{\pi_{12}+\cdots+\pi_{N2}}{N}$$

算法实现:

首先使用已有的身高数据，保存在 student_height.csv 文件中。绘制身高的分布如下图



在 EM 算法中，设定初始权重为 0.5，0.5，初始均值通过随机抽取两个数值，初始方差均设定为 1。

初始设定值如下:

Initial means: [176 180]

Initial standard deviations: [1. 1.]

Initial weights: [0.5 0.5]

迭代次数 300 次，得到结果如下:

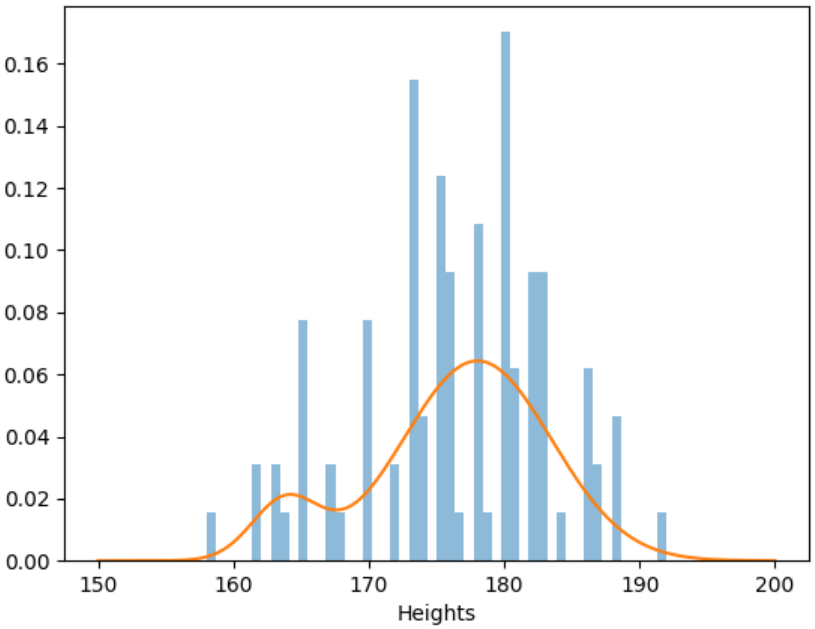
Final means: [164.01002858 176.01056606]

Final standard deviations: [3.12794657 4.99734404]

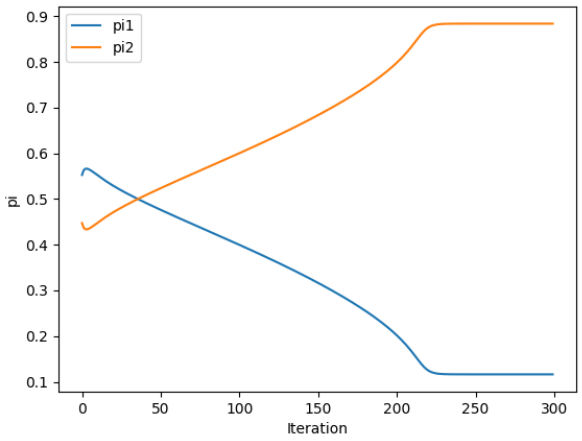
Final weights: [0.24359935 0.75640065]

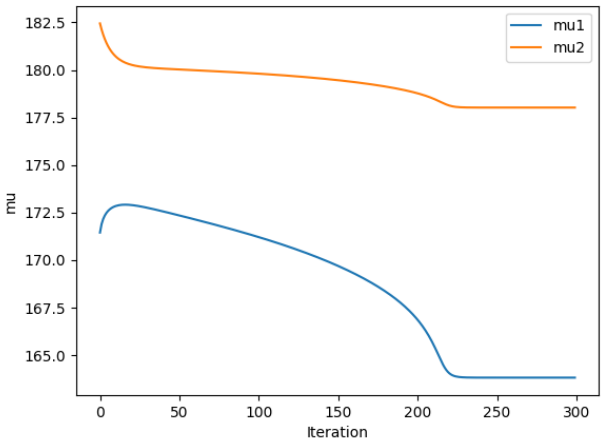
和真实数值相比，得到结果十分接近，对比结果与分布图如下

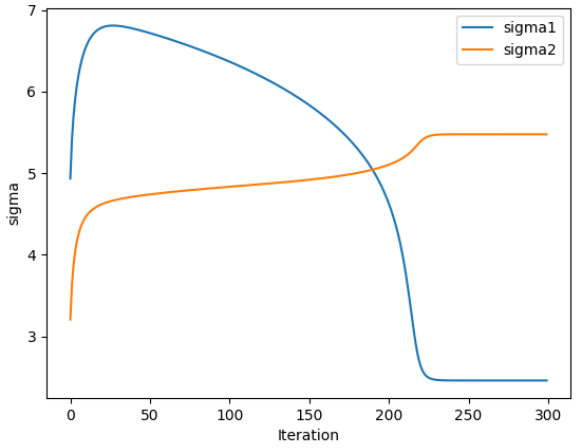
	真实值	预测值
男生数量	78	84
女生数量	17	11
男生均值	178.33	176.01
女生均值	167.35	164.01
男生标准差	5.532	4.997
女生标准差	5.3461495	3.128
判断正确率	89.47%	



将历次迭代的权重，均值，方差依次绘制







可以看到，大约在迭代至 220 次的时候，结果收敛到了定值。

结论

经过计算我们得到了混合高斯模型的参数，和准确值相比，EM 算法得到的结果十分接近。

实验过程中发现不同的初始参数影响着迭代次数，对于本实验采用的参数生成的方法，在一些参数情况下，200 次以内能够收敛，而部分情况下则需要 250 次以内才能够收敛得到。本实验主要是根据实际设定了最大计算次数，此外还可以通过设定最小差值，当若干次计算的结果相差小于该值时认为收敛了。