PRML Assignment 1 Report

PART I

在第一部分中，需要设计三个高斯分布，然后在三个分布上采样以作为后续分类模型的输入。

生成高斯分布的采样可以使用 numpy.random.multivariate_normal 函数.

第二部分使用的数据可以直接使用 source.genDis(means, covs, [1/3, 1/3, 1/3], 5000)来生成

PART II

样本点分为 A,B,C 三类，多分类问题我们可以写出 softmax 形式的后验概率 (Bishop, 2006)

$$p(C_k|x) = \frac{exp(a_k)}{\sum_j exp(a_j)}$$

$$a_k = \ln p(x|C_k)p(C_k)$$

生成模型

$$p(\boldsymbol{T}|\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma}) = \prod_{n=1}^{N}\prod_{i=1}^{3}(\pi_i N(\boldsymbol{x}_n|\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i))^{t_{ni}}$$

由最大化$\log\left(p(\boldsymbol{T}|\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma})\right)$

$$\sum \pi_i = 1, maximize \sum_{i=1}^{3} N_i \ln \pi_i$$

$$\therefore \boldsymbol{\pi} = \frac{\boldsymbol{N}}{N}$$

$$\forall i \in \{1,2,3\}, maximize \sum_{t_{ni}=1} \ln \frac{1}{2\pi}\frac{1}{|\boldsymbol{\Sigma}_i|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}_n^T\boldsymbol{\Sigma}_i^{-1}x - \boldsymbol{\mu}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{x}_n - \boldsymbol{x}_n^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i + \boldsymbol{\mu}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i)}$$

对$\boldsymbol{\mu}_i$,

$$\sum_{t_{ni}=1} -\boldsymbol{\Sigma}_i^{-1}\boldsymbol{x}_n - \boldsymbol{\Sigma}_i^{-1^T}\boldsymbol{x}_n + \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i + \boldsymbol{\Sigma}_i^{-1^T}\boldsymbol{\mu}_i = 0$$

由$\boldsymbol{\Sigma}_i$正定，得到

$$\sum_{t_{ni}=1} \boldsymbol{x}_n - \boldsymbol{\mu}_i = 0$$

$$\Leftrightarrow \boldsymbol{\mu}_i = \frac{1}{N_i}\sum_{n=1}^{N} t_{ni}\boldsymbol{x}_n$$

对$\boldsymbol{\Sigma}_i$,

$$maximize\ -\frac{1}{2}N_i \ln|\boldsymbol{\Sigma}_i| + \sum_{t_{ni}=1} -\frac{1}{2}(\boldsymbol{x}_n^T - \boldsymbol{\mu}_i^T)\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu}_i)$$

$$-\frac{1}{2}N_i\frac{1}{|\boldsymbol{\Sigma}_i|}adj(\boldsymbol{\Sigma}_i)+\frac{\partial}{\partial\boldsymbol{\Sigma}_i}\sum_{t_{ni}=1}-\frac{1}{2}(\boldsymbol{x}_n^T-\boldsymbol{\mu}_i^T)\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_i)=0$$

$$-\frac{1}{2}N_i\boldsymbol{\Sigma}_i^{-1}+\frac{\partial}{\partial\boldsymbol{\Sigma}_i}\sum_{t_{ni}=1}-\frac{1}{2}tr(\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_i)(\boldsymbol{x}_n^T-\boldsymbol{\mu}_i^T))=0$$

$$-\frac{1}{2}N_i\boldsymbol{\Sigma}_i^{-1}+\sum_{t_{ni}=1}\frac{1}{2}\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_i)(\boldsymbol{x}_n^T-\boldsymbol{\mu}_i^T)\boldsymbol{\Sigma}_i^{-1}=0$$

$$\boldsymbol{\Sigma}_i=\frac{1}{N_i}\sum_{n=1}^{N}t_{ni}(\boldsymbol{x}_n-\boldsymbol{\mu}_i)(\boldsymbol{x}_n^T-\boldsymbol{\mu}_i^T)$$

训练出参数后，我们取使a_k最大的$k\in\{1,2,3\}$来确定分类。即$p(C_k|x)\propto p(x|C_k)p(C_k)$

判别模型

让

$$a_k(x)=\boldsymbol{w}_k^T\boldsymbol{x}$$

其中$\boldsymbol{x}$第一维为 1,因此 w 表示样本空间+1 维的列向量。则

$$\tilde{\boldsymbol{y}}=\mathrm{softmax}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{x})$$

使用交叉熵损失函数, $\boldsymbol{y}$使用 one-hot,

$$L(\boldsymbol{W})=\sum_{n=1}^{N}\boldsymbol{y}log(\tilde{\boldsymbol{y}})$$

使用小批量随机梯度下降学习 W 矩阵，有

$$\boldsymbol{W}_{next}=\boldsymbol{W}+\alpha\sum_{n\ in\ batch}\boldsymbol{x}_n(\boldsymbol{y}_n-\tilde{\boldsymbol{y}}_n)^T$$

训练完成后反代会 softmax 得到后验概率。

PART III

观察到生成模型无论在什么数据集下的效果都非常好。这是可以想见的：生成模型是使用高斯分布，也就是原始数据分布，因此在本 assignment 中能得到高斯分布的所有参数并且用来生成数据，而判别模型则只能用来分类。

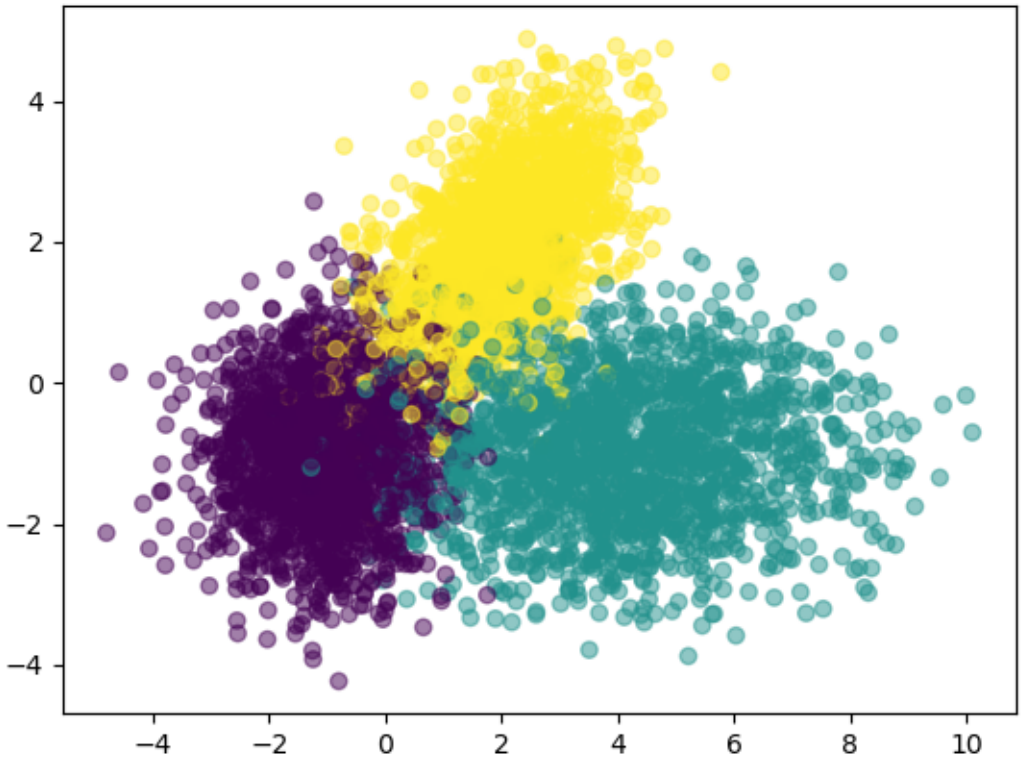

Figure 1 原始数据集，共 5k 点

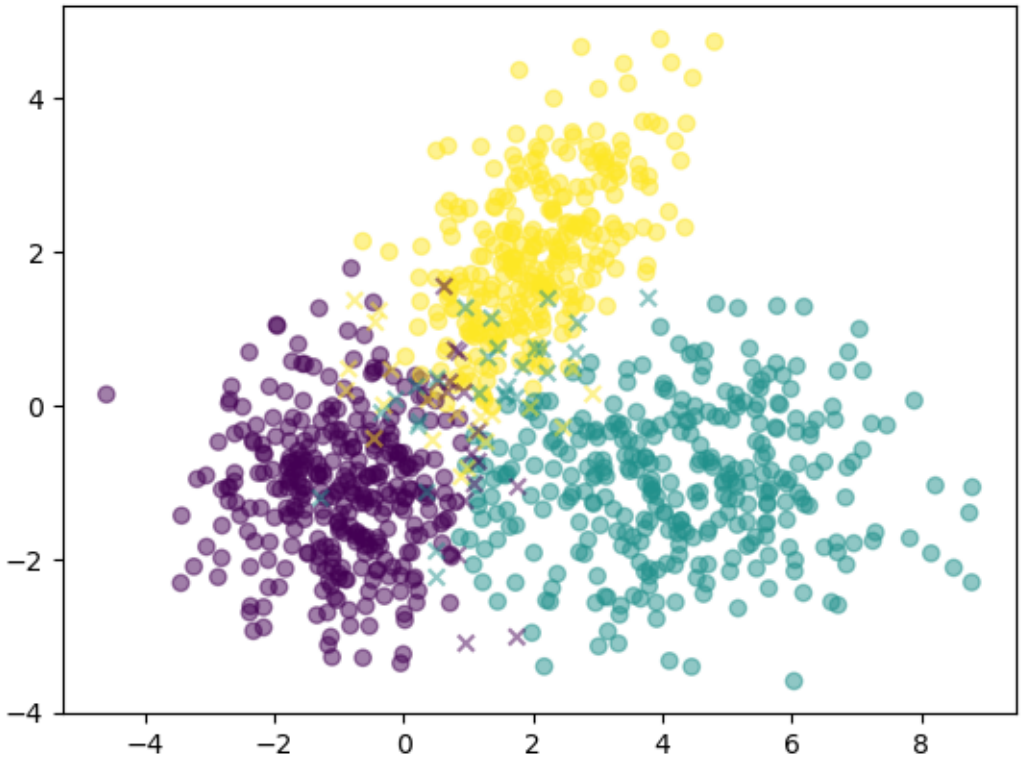

Figure 2 生成模型，准确率 0.936

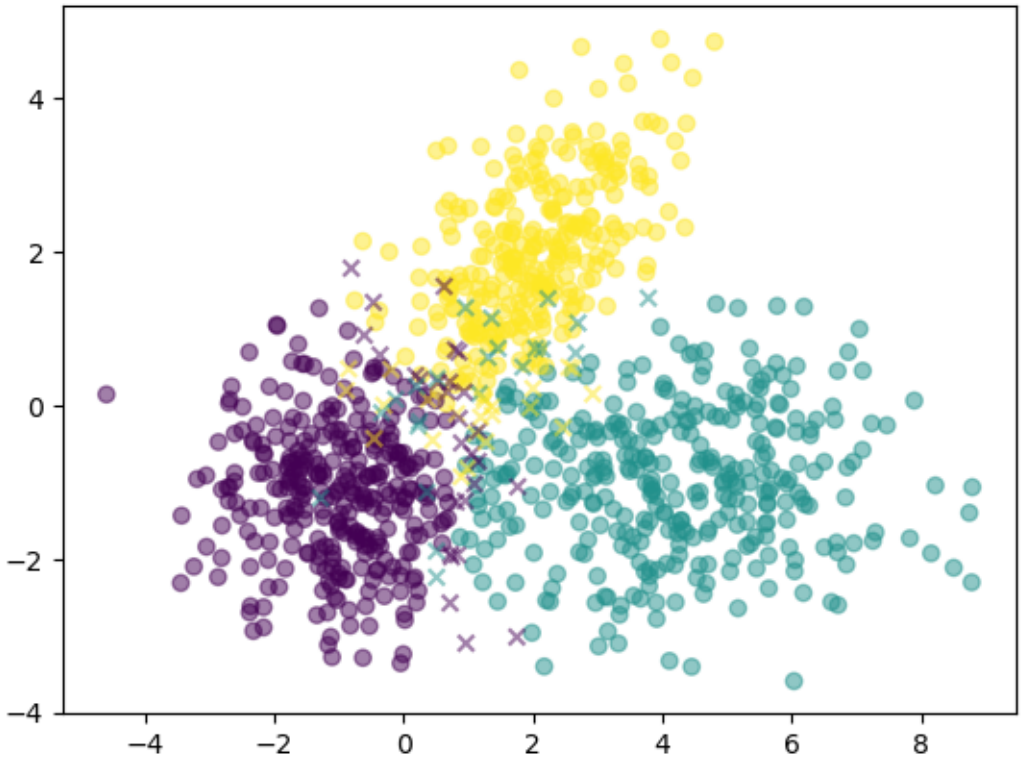

Figure 3 判别模型，alpha0.05, 400 个 epoch，batch = 40.虽然训练准确率停止在 0.936 但是测试集准确率（上图）为 0.927，过拟合了

判别模型得到的决策边界是线性的，但是我没能通过调整原始数据协方差使得判别模型正确率相比生成模型有较大下降。增大数据的维度、数据分布的数量后生成和判别模型都还是能做的很好。但是对原始数据的平移会对训练速度造成较大干扰：

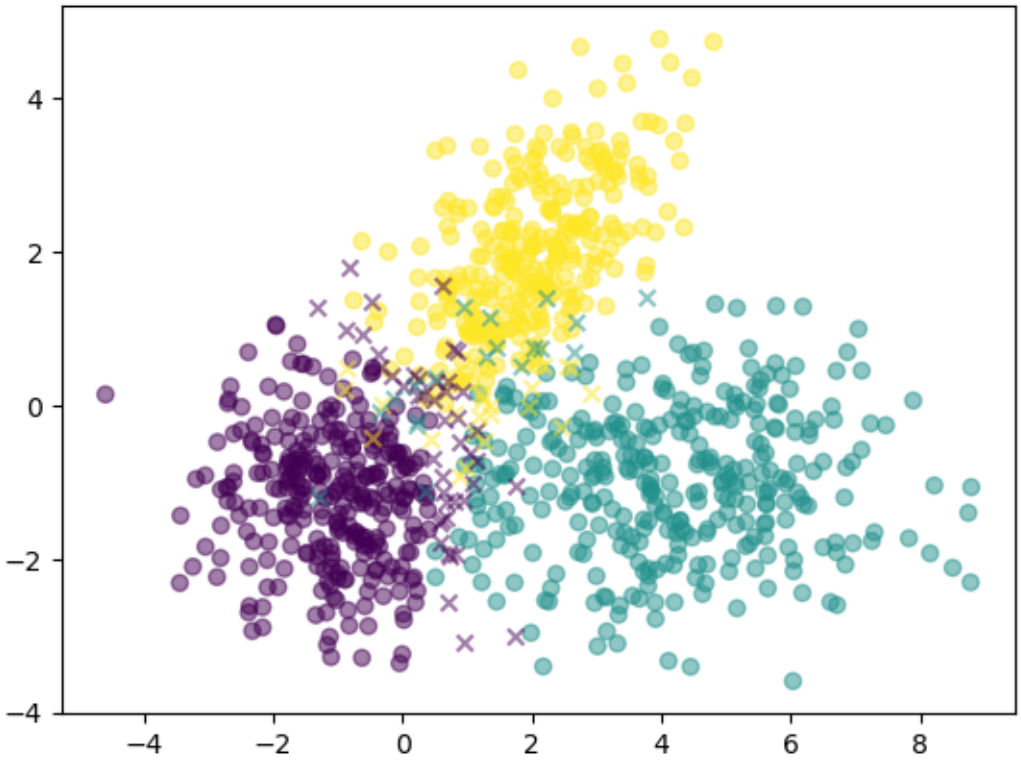

Figure 4 训练参数不变，仅 epoch 变为 100，正确率 0.913

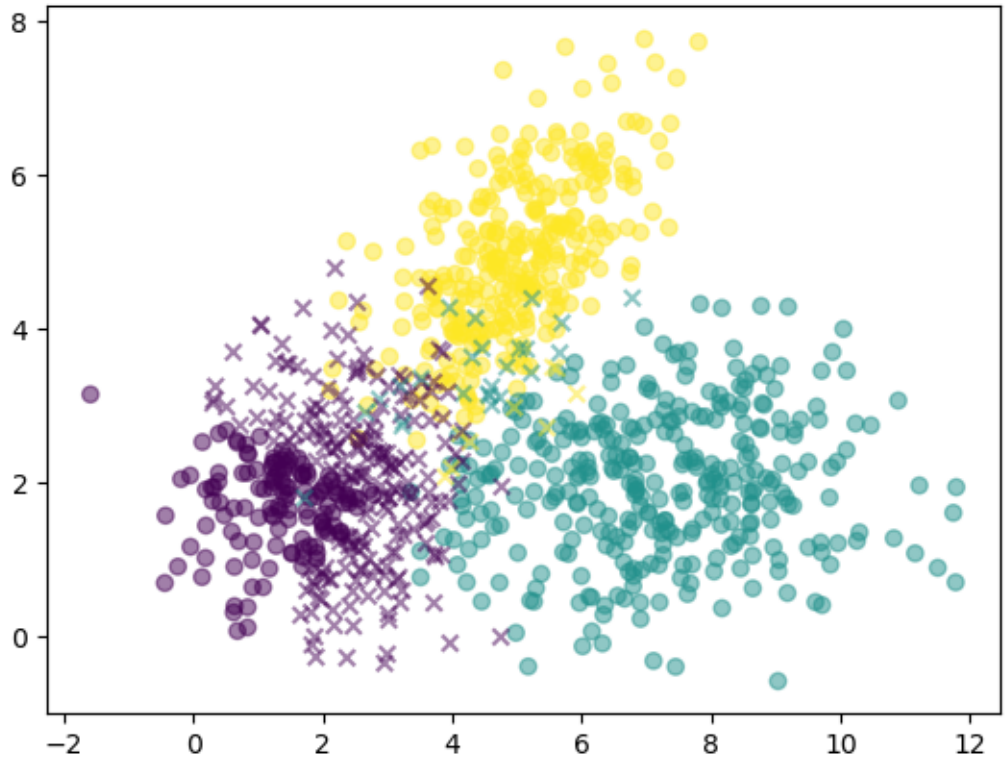

Figure 5 数据集进行平移，100epoch 没能完成训练

一方面可能与初始 W 矩阵有关。另一方面我认为可能是由于是对预测 y softmax 过后的交叉熵可能限制了学习的速度。神经网络有时候最后不经过 softmax 层，而是训练完成后加入 softmax 层，不知道是不是出于同样的原因。

并且当数据线性不可分的情况下，我们最好加大 batch_size

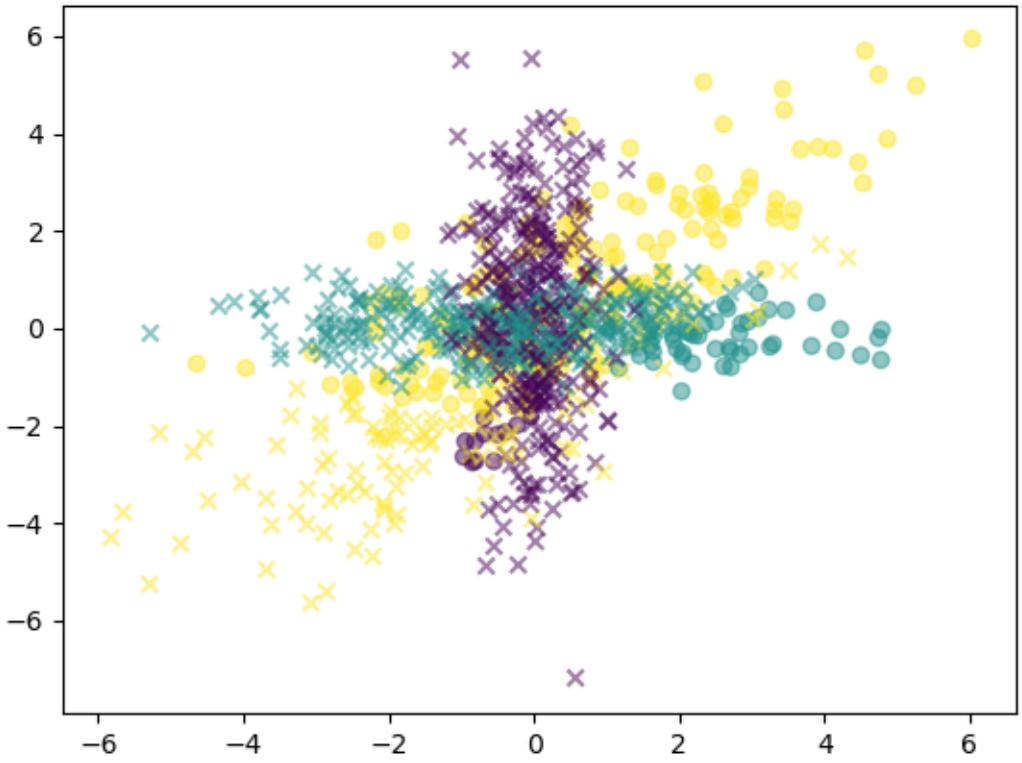

Figure 6 三个数据中心完全重合，生成模型能达到 0.714 正确率，而判别模型可想而知无法获得较好的解

总结

Generative model 不需要设置超参，训练速度更快，可以说在本次 assignment 的模型中完胜。但是这是由于事先知道正确的数据模型，使得生成模型上限和直接使用数据生成参数来做判断相同，可能在其他情况就没有这么方便。

虽然 python 功底很差，但是还是磕磕绊绊完成了 assignment。感觉对自己是一次很好的锻炼。