实验一 数据降维与分类任务

1 问题描述

分别利用 PCA 和 LDA 降维技术对葡萄酒数据进行降维处理，在降维后的数据集上训练和测试 logistic 回归分类器，并比较降维技术前后分类器准确率的变化。

2 实现步骤与流程

PCA 降维

求解样本的散布矩阵

$$\mathbf{S}(\boldsymbol{x}) = \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu}_x)(\boldsymbol{x}_i - \boldsymbol{\mu}_x)^{\mathrm{T}}$$

求解散布矩阵最大的 β 个特征值对应的特征向量作为基向量进行投影

$$\mathbf{S}(\boldsymbol{x})\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i \qquad i = 1,\ 2,\ \cdots,\ k$$

投影后的样本特征向量

$$\boldsymbol{y} = \mathbf{W}^{\mathrm{T}}\boldsymbol{x} \Rightarrow \mathbb{R}^{\beta} = \mathbb{R}^{\beta\times\alpha}\mathbb{R}^{\alpha}$$

其中变换矩阵 $\mathbf{W}$ 的列向量即为散布矩阵的特征向量 $\boldsymbol{w}_i$

LDA 降维

求解类内散布矩阵

$$\mathbf{S}_w(\boldsymbol{x}) = \sum_{i=1}^{c} \mathbf{S}_i = \sum_{i=1}^{c} \sum_{\boldsymbol{x}\in\mathcal{X}_i} (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x} - \boldsymbol{\mu}_i)^{\mathrm{T}}$$

求解类间散布矩阵

$$\mathbf{S}_b(\boldsymbol{x}) = \sum_{i=1}^{c} n_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^{\mathrm{T}}$$

如果类内散布矩阵可逆，投影的 β 个基向量满足

$$\mathbf{S}_w^{-1}\mathbf{S}_b \boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i \qquad i = 1,\ 2,\ \cdots,\ k$$

并且对应分别对应最大的 β 个特征值 λ_i。如果类内散布矩阵不可逆，可以将其替换为

$$\mathbf{S}_w \leftarrow \mathbf{S}_w + \epsilon \mathbf{I}_\beta, \quad \epsilon > 0$$

投影后的样本特征向量

$$\boldsymbol{y} = \mathbf{W}^{\mathrm{T}} \boldsymbol{x} \Rightarrow \mathbb{R}^\beta = \mathbb{R}^{\beta \times \alpha} \mathbb{R}^\alpha$$

其中变换矩阵 $\mathbf{W}$ 的列向量即为矩阵 $\mathbf{S}_w^{-1} \mathbf{S}_b$ 的特征向量 $\boldsymbol{w}_i$

logistic 回归

logistic 回归模型

$$\hat{y} = \sigma(z) = \sigma(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b)$$

其中 $\sigma(\cdot)$ 代表 sigmoid 函数，对于 logistic 回归可以使用两种损失函数

- 交叉熵损失

$$\ell_{cross-entropy} = -\sum_{i=1}^{n} \left[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \right]$$

- 负对数似然

$$\ell_{log-likelihood} = \sum_{i=1}^{m} \left[-y_i \hat{y}_i + \log(1 + e^{\hat{y}_i}) \right]$$

本次实验中采取前者进行实现。由于 logistic 回归模型只适用于二分类任务，对于多分类任务，可以通过 OvO、OvM 或 MvM 等方式将 logistic 模型拓展到多分类中，同时也可以考虑 softmax 回归模型

$$\hat{\boldsymbol{y}} = softmax(\mathbf{W}^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{b})$$

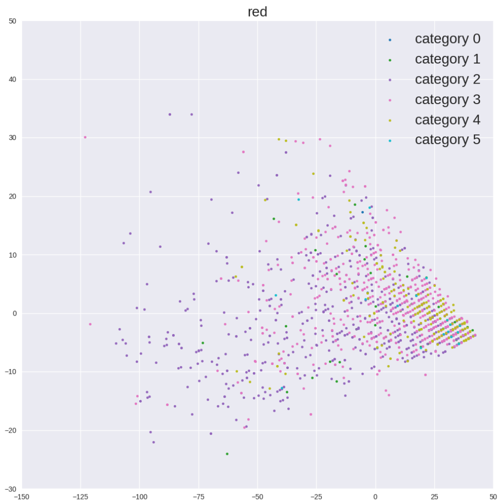
并采取交叉熵损失对模型进行优化。

3 实验结果与分析

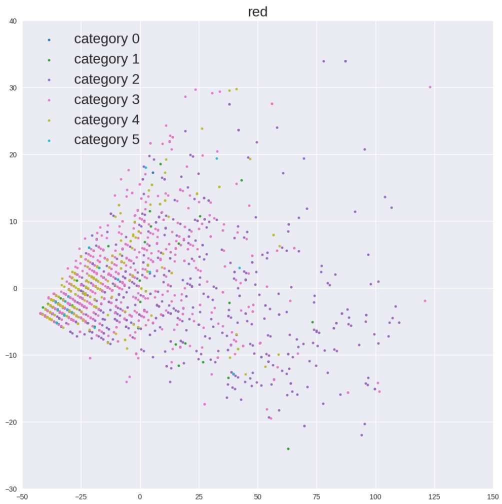
由于实验要求的 MindSpore 框架相关的资料较为匮乏，因此本次实验采取开源机器学习算法库 sklearn 与实验中的算法进行对比。sklearn 是一个开源的基于 python 语言的机器学习工具包。它通过 numpy, scipy 等 python 数值计算的库实现高效的算法应用，并且涵盖了几乎所有主流机器学习算法。

PCA 降维

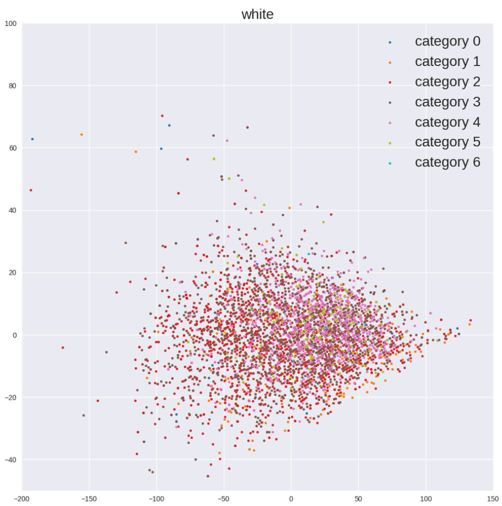
经过实验算法以及 sklearn 算法的 PCA 降维后的样本如下图所示，可以看出，降维后的样本具有较高的散布程度。并且样本经过实验的 PCA 算法降维后与 sklearn 的 PCA 算法相比拥有相同的性态，但二者在某一特征维度上呈镜像对称。



(a) 红葡萄酒（实验）



(b) 红葡萄酒（sklearn）



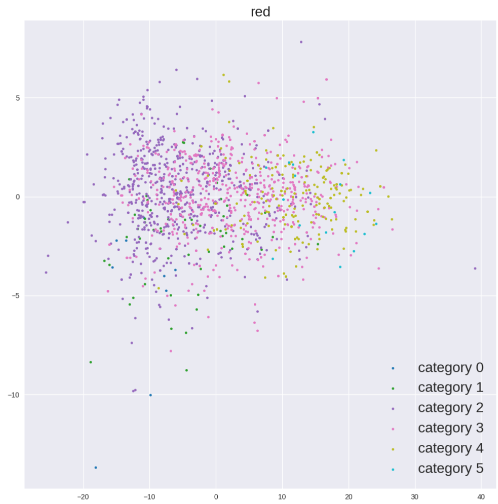
(c) 白葡萄酒（实验）

(d) 白葡萄酒（sklearn）

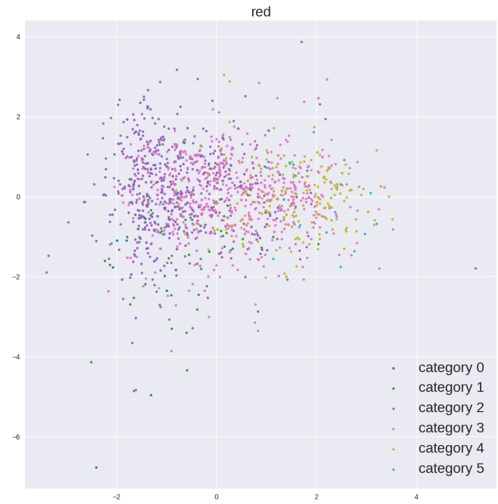
图 1: PCA 降维效果图

LDA 降维

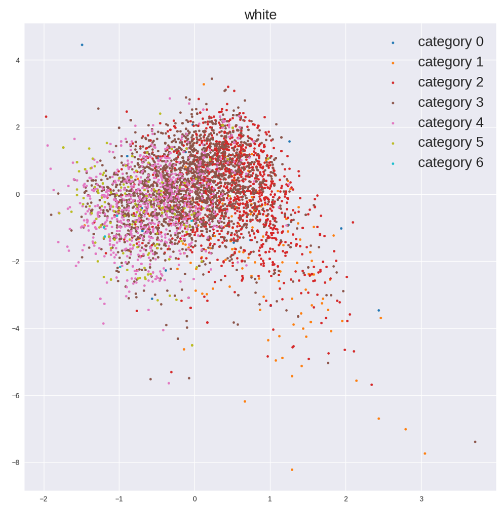
经过实验算法以及 sklearn 算法的 LDA 降维后的样本如下图所示，可以看出，降维后的样本具有相对较好的可分性。并且样本经过实验的 PCA 算法降维后与 sklearn 的 PCA 算法相比拥有相同的性态，与 PCA 不同的是，二者仅在白葡萄酒上呈镜像对称。



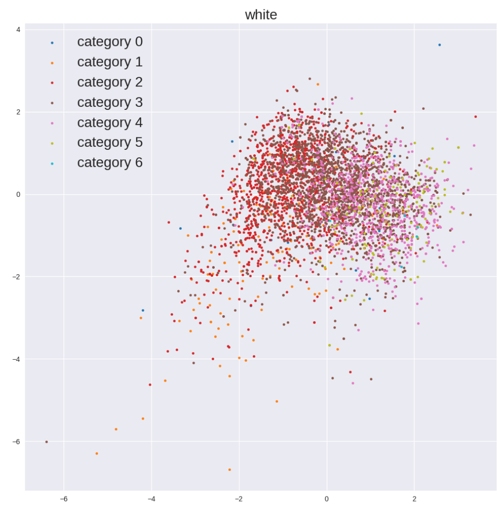
(a) 红葡萄酒（实验）



(b) 红葡萄酒（sklearn）



(c) 白葡萄酒（实验）

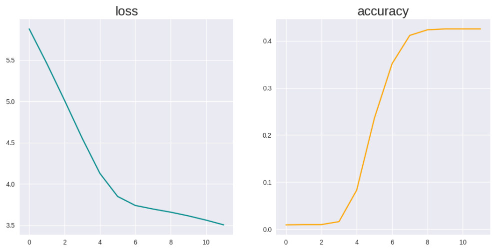


(d) 白葡萄酒（sklearn）

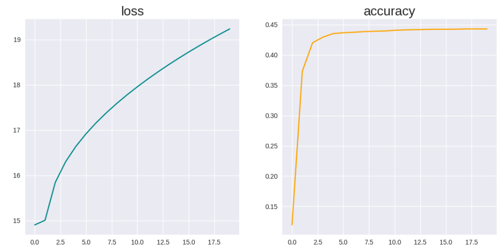
图 2: LDA 降维效果图

3.1 logistic 回归

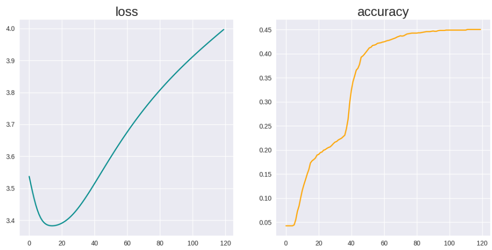
实验中实现的 logistic、softmax 回归在原始数据集、PCA 降维数据集以及 LDA 降维数据集上的训练过程如下图所示，分别展示了模型训练过程中损失函数以及分类准确率的变化。



(a) 红葡萄酒（原始）

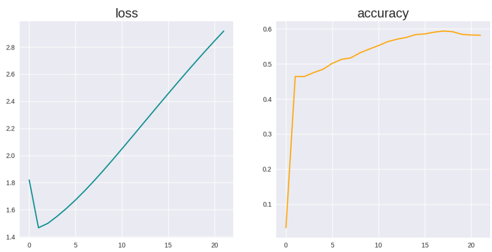


(b) 白葡萄酒（原始）

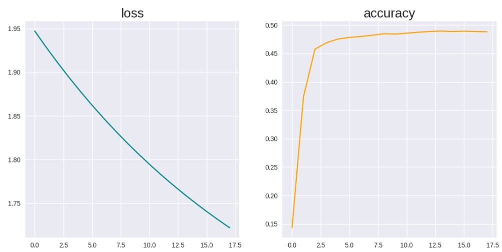


(c) 红葡萄酒（PCA）

(d) 白葡萄酒（PCA）



(e) 红葡萄酒（LDA）



(f) 白葡萄酒（LDA）

图 3: logistic 回归（实验）

从图像中可以看出，原始数据集的模型准确率居中，PCA 降维后的数据集的模型准确率较差，LDA 降维后的数据集的模型准确率最佳。同时，也可以观察到实验中的反常现象：损失函数与准确率一同上升，经过排查暂时没有发现这种现象出现的原因。

相应地，sklearn 提供的 logistic 回归模型在各个数据集上的准确率如下图所示，与实验算法相同的是，LDA 的准确率最高，原始数据次之，PCA 最低。

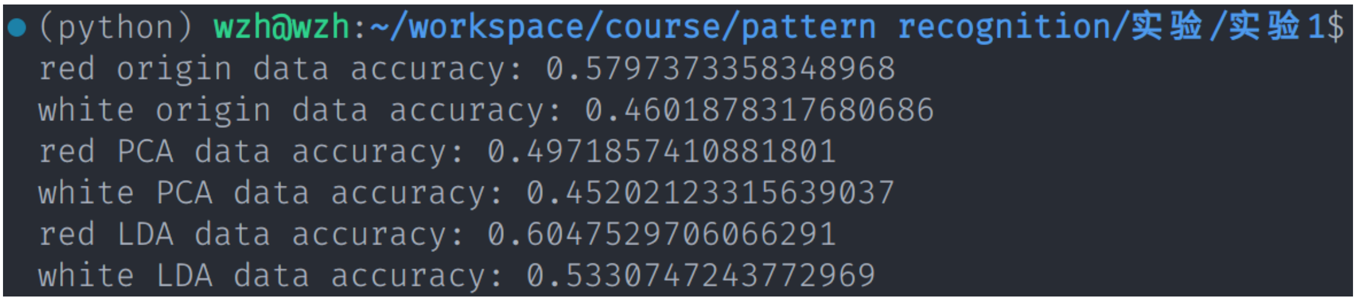


图 4: logistic 回归（sklearn）

4 MindSpore 学习使用心得体会

由于本次实验并未采用 MindSpore，此部分用 sklearn 的学习使用心得体会来代替。

本次实验中调用了 sklearn 中的以下算法接口

- PCA（decomposition PCA）
- LDA（discriminant_analysis LinearDiscriminantAnalysis）
- logistic 回归（linear_model LogisticRegression）

作为实验算法的对照算法，sklearn 提供的算法接口方便调用，并且执行效率较高，能够高效地实现实验的需求。

5 代码附录