简介

本课程重点讲解11大机器学习经典算法，将算法的原理推导、模型优化作为核心内容，对代码能力较差的同学可能不是十分友好，鉴于同学之前出现的各种问题，本文予以总结，希望能对同学们的学习有所帮助。

涉及到的python库

在这里将用到的几个库列出来，大家可以有针对性的去学习一下。

1. matplotlib
   常用的数据可视化库，用于绘制各种图表，主要用到其中的pyplot模块。
2. numpy
   用于处理数据的python，因为机器学习中的数据常常是数组形式，维数一般较高，而numpy库提供了很多处理和计算多维数组的方法。
3. scikit-learn
   常用的机器学习库，封装了很多机器学习算法和调参优化方法，内容较为庞杂，试学部分主要涉及到以下内容，可以针对性的学习这部分内容。

```
linear_model.LinearRegression()  # 线性回归模型
datasets.load_iris()  # sklearn库封装好了一些经典数据集，这里是加载iris数据集到内存中
neighbors.KNeighborsClassifier()  # K近邻分类模型
model_selection.KFold()  # K折交叉验证模块
model_selection.GridSearchCV()  # 网格搜索选取参数
```

易出现的问题

1. 模型训练时传入的数据。
   这里主要要注意的是传入数据的维度，因为sklearn中封装的算法是对数据维度有要求的，而numpy数组在切片后会有维度丢失的现象(切片后会将值为1的维度展开)，会造成数组降维，如下所示：

```
>>> data = np.array(
    [[152, 51], [156, 53], [160, 54], [164, 55], [168, 57], [172, 60], [176,
62], [180, 65], [184, 69], [188, 72]])
>>> data.shape
(10, 2)  # 此时可以看到data是一个二维数组
>>> data[:,0].shape
(10,)  # 而切片后本应是(10,1)的二维数组，却变成了(10,)第二个维度丢失了，变成了一维数组
```

所以在传入数据前应检查一下数组形状，如果维度丢失，可以使用reshape函数来修正。

2. 3.3项目第一段读取数据文件报错，imread不可用，是由于scipy版本变更导致，可将load_data.py文件中`from scipy.misc import imread`改为`from PIL.Image import open as imread`，其他地方无需改动。
3. 3.3项目imshow函数绘制显示图片报错的问题。
   很多同学直接拿到图片数据就使用imshow进行绘制，但会报这样一个错误，

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for
floats or [0..255] for integers).
```

这个错误出现的原因是是数据类型出现了问题，在imshow函数里，要求传入图片的RGB是[0,1]的浮点数或者[0,255]的整数，但是咱们读出来的图片的RGB实际上是`[216., 184., 140.]`，可以看到它的范围是[0,255]，但是每一个值却是小数类型，所以这里需要做一个处理，要么将值映射到[0,1]，要么将小数类型转成整数。以下是示例代码，可供参考：

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

# X作为单张图片数据
# 第一种，映射到[0,1]
plt.imshow(X/255)

# 第二种，转换成整数类型
X = X.astype(np.int32)
plt.imshow(X)
```

3. 随机采样时随机种子的设置。
   在取样的时候设置一个随机种子，随机种子其实就决定着每次产生的随机数，随机种子不变则每一次运行都产生相同的随机数，在项目里也就是每次运行选出的图片都不变，以免多次运行时选出不同的训练集测试集影响模型的准确率。
4. 抽样图片展示，思路提示：
   通过subplot函数将figure对象分为50个子区域(5*10,10个种类，每类5张)，通过for循环分别在每一个子区域绘制图片，注意绘制时要设置不显示坐标轴，还要判断当前行数，如果是第一行，要给图片加上标题(这类的标签)。
5. 有关3.3项目交叉验证太慢的问题。
   很多同学跑了很久最终也没有跑出结果还导致电脑卡死，在这里个大家提点小建议，作业中要求使用`GridSearchCV`网格搜索，在实例化`GridSearchCV`对象时有一个参数`n_jobs`，`n_jobs`表示使用的CPU核数，`n_jobs = -1`表示使用全部CPU，在这里不建议大家使用全部，最好留一个核以免CPU超负荷卡死，反而影响程序运行，另一点，大家可以将`GridSearchCV`换成`RandomizedSearchCV`随机搜索，可以大大减少训练时间。

结束语

以上是这段时间根据学员常出现的问题总结得来，希望同学们能多多思考，有疑问积极到答疑群内提出，老师们会继续去总结改进，贪心会继续努力，为大家提供更好的服务！