

Machine Learning Lab 1

Classifier

[实验要求]

给出一份 Python 代码 (Lab1.py)，学生需要利用 scikit-learn 库的相关方法将代码中的关键函数补充完整，从而实现 KNN、逻辑回归、决策树与 SVM 分类。在实现分类的基础上，学生可对相关分类器的参数进行调整，来检验这些参数对分类结果的影响。除此以外，学生还需要利用 matplotlib 库对分类结果进行可视化，同样以补充函数的方式进行。

[数据]

实验采用 Scikit-learn 库自带的 Iris 鸢尾花数据集，数据集内包含 3 类共 150 条记录，每类各 50 个数据，每条记录都有 4 项特征：花萼长度、花萼宽度、花瓣长度、花瓣宽度，可以通过这 4 个特征预测鸢尾花卉属于 (iris-setosa, iris-versicolour, iris-virginica) 中的哪一品种。

[实验说明]

1. 需要学生进行补充的函数（有且仅有以下几个）

```
def Vis(test, pred):  
    # Matplotlib Visualization  
    # Tip: Using Scatter Figure  
    pass  
  
def KNN(test):  
    # KNN Implementation  
    pass  
  
def Logistic(test):  
    # Logistic Regression Implementation  
    pass  
  
def DecisionTree(test):  
    # Decision Tree Implementation  
    pass  
  
def SVM(test):  
    # SVM Implementation  
    pass
```

其中，Vis 函数接受样本特征与标签作为参数，需要输出散点图，可使用 matplotlib 库中的方法。不同的标签可以用不同颜色的点来表示，而特征的坐标则可选用鸢尾花样本中的两个特征，可以进行任意选取，查看哪些特征与类别的

关联性强。

其他几个函数对应不同的分类器，可选用 Scikit-Learn 库中对应的方法进行实现。这只需要几行代码即可完成，因此如果在实现这一部分时写了过多的代码，请思考是否走了弯路。

2. 分类器的调用

分类器通过 Predict 函数进行调用，只需要调整该函数的第一个字符串参数即可。参数值分别为 KNN, Logistic, DecisionTree 和 SVM。注：此处固定了输入格式，如果了解怎样使输入更灵活，请参照正则表达式相关内容。

```
y_pred = Predict('SVM',X_test)
```

3. Vis 函数的调用

Vis 函数默认接受测试数据。但若有兴趣，可将测试数据改为训练数据以查看训练集分布情况。

4. 其他部分

其他部分均固定，无需任何修改。若有兴趣，可调整 train_test_split 函数的参数实现不同的数据集划分方式。

[实验环境]

实验在 Python 环境下运行，需要安装的库有 Scikit Learn, matplotlib 与 Numpy。

Scikit learn 文档: https://scikit-learn.org/stable/user_guide.html

Numpy 文档: <https://numpy.org/doc/stable/>

Matplotlib 文档: <https://matplotlib.org/stable/tutorials/index.html>