华南理工大学

South China University of Technology

《机器学习》课程实验报告

学　　院　软件学院

专　　业　软件工程

组　　员　张驰，李龙康

学　　号　201530613603，201530612002

邮　　箱　916583810@qq.com

指导教师　吴庆耀

提交日期　2017 年 12 月 22 日

1. 实验题目：基于 AdaBoost 算法的人脸分类

2. 实验时间：2017 年 12 月 9 日

3. 报告人:张驰，李龙康

4. 实验目的:

1）深入理解 AdaBoost 算法原理
2）熟悉人脸检测的基本方法
3）学会利用 AdaBoost 算法解决人脸分类问题，将理论和实际工程接轨
4）体验机器学习的完整过程

5. 数据集以及数据分析：

本实验提供 1000 张图片，其中 500 张是含有人脸的 RGB 图片，储存在./datasets/original/face 内；另外 500 张是不含有人脸的 RGB 图，储存在./datasets/original/nonface 内。

6. 实验步骤:

1. 读取数据集数据。读取图片，将全部图片转成大小为 24*24 的灰度图，数据集正负类样本的个数和比例不限，数据集标签形式不限。
2. 处理数据集数据，提取 NPD 特征。使用 *feature.py* 中 *NPDFeature* 类的方法提取特征。（提示：因为预处理数据集的时间比较长，可以用 pickle 库中的 dump()函数将预处理后的特征数据保存到缓存中，之后可以使用 load()函数读取特征数据）
3. 将数据集切分为训练集和验证集，本次实验不切分测试集。
4. 根据 *ensemble.py* 中的预留的接口编写 *AdaboostClassifier* 所有函数。以下为 *AdaboostClassifier* 类中的 fit()方法的思路：
   4.1 初始化训练集的权值 ω,每一个训练样本被赋予相同的权值。
   4.2 训练一个基分类器，基分类器可以使用 sklearn.tree 库中 DecisionTreeClassifier(注意训练的时候需要将权重 ω 作为参数传入)。
   4.3 计算基分类器在训练集上的分类误差率 ε 。
   4.4 根据分类误差率 ε，计算参数 α 。
   4.5 更新训练集的权值 ω 。
   4.6 重复以上 4.2-4.6 的步骤进行迭代，迭代次数为基分类器的个数。
5. 用 *AdaboostClassifier* 中的方法在验证集上进行预测并计算精确率,并用 sklearn.metrics 库的 classification_report()函数将预测结果写入 *report.txt* 中。
6. 整理实验结果并完成实验报告。

7. 代码内容:

Algorithm 2: Adaboost

Input: $D = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$, **where** $\mathbf{x}_i \in X, y_i \in \{-1, 1\}$
Initialize: Sample distribution w_m
Base learner: $\mathcal{L}$

$w_1(i) = \frac{1}{n}$
for m=1,2,...,M **do**
  $h_m(x) = \mathcal{L}(D, w_m)$
  $\epsilon_m = \sum_{i=1}^{n} w_m(i)\mathbb{I}(h_m(\mathbf{x}_i) \neq y_i)$
  if $\epsilon_m > 0.5$ **then**
    break
  end
  $\alpha_m = \frac{1}{2}\log\frac{1-\epsilon_m}{\epsilon_m}$
  $w_{m+1}(i) = \frac{w_m(i)}{z_m}e^{-\alpha_m y_i h_m(\mathbf{x}_i)}$, **where** $i = 1, 2, ..., n$ **and** $z_m = \sum_{i=1}^{n} w_m(i)e^{-\alpha_m y_i h_m(\mathbf{x}_i)}$
end
Output: $H(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m h_m(\mathbf{x})$

8.实验总结：adaBoost 是 boosting 方法中最流行的一种算法。它是以弱分类器作为基础分类器，输入数据之后，通过加权向量进行加权，；在每一轮的迭代过程中都会基于弱分类器的加权错误率，更新权重向量，从而进行下一次迭代。并且会在每一轮迭代中计算出该弱分类器的系数，该系数的大小将决定该弱分类器在最终预测分类中的重要程度。显然，这两点的结合是 adaBoost 算法的优势所在。优点：泛化错误率低，容易实现，可以应用在大部分分类器上，无参数调整。缺点：对离散数据点敏感