人工智能基础

编程作业 2

http://staff.ustc.edu.cn/~linlixu/ai2016spring/

完成截止时间：2016/6/30
提交至：ustc_ai2016@163.com
助教： 郭俊良(guojunll@mail.ustc.edu.cn)
蒋亮(jal@mail.ustc.edu.cn)
张超(zhangchao5656@gmail.com)

实验目的：

本次实验考虑机器学习中传统的监督学习问题与非监督学习，基于两个经典应用：手写数字识别和图片去噪，并结合课上介绍的相应学习算法，在数据集上分别进行实验，以加强对相关算法原理及应用的理解。

Part 1. 手写数字分类(75%)

数据集介绍：

USPS 手写数字识别数据集，我们将对其中的 3 和 8 两个数字进行分类，每张图片表示为一个 16x16 像素的黑白图片，对于每一个像素，用 1 个 8 bit 数字(0-255 之间)表示其灰度值。一个 16x16 的图片，总共有 256 个像素，因此对于每张图片，可以用一个 256 个元素的向量表示。而在标记信息中，0 表示当前样本为数字 3，1 表示当前样本为数字 8。

注：我们对每个数据集进行了一定的划分，保留了每个数据集的一部分作为对实验结果的评价之一，余下的部分放在了课程主页上供下载。

训练与测试

在监督学习中，训练数据带有标号，在训练的过程中需要从训练数据 traindata 和其对应的标号 trainlabel 中学习相应的分类模型。

在测试过程中，用学习到的模型对测试集中的数据 testdata 作预测，并将预测结果与测试数据的真实标签 testlabel 进行比较，从而度量分类模型的性能。

$$\text{Accuracy} = \frac{\sum_{i \in test\ set} I(\text{predict}_i = testlabel_i)}{\#of\ test\ size}$$

实验要求：

1．实现一个朴素贝叶斯分类器(10%)

提交一个 Matlab 函数 nbayesclassifier，函数形式为

```
function [ypred,accuracy]= nbayesclassifier (traindata,
                    trainlabel, testdata, testlabel, threshold)
```

其中 threshold 为用于判断类别的后验概率的阈值，即如果 P(digit=8|x) > threshold 则判别为数字 8。要求函数返回对测试数据的预测 ypred，以及通过与真实标号比较计算得到的分类正确率 accuracy。ypred 与 trainlabel 和 testlabel 形式相同。

2．实现一个最小二乘分类器(引入规范化项后)(10%)

1)．对引入了 L2 规范化项之后的最小二乘分类问题进行推导。即求解以下优化问题：

$$min_w(X\boldsymbol{w}-y)^2+\lambda\|\boldsymbol{w}\|^2$$

2)．基于 1 中的结果，实现并提交一个 Matlab 函数 lsclassifier

```
function [ypred,accuracy] = lsclassifier(traindata, trainlabel,
                              testdata, testlabel, lambda)
```

3．实现一个支持向量机分类器 (15%)

提交一个 Matlab 函数 softsvm

```
function  [ypred,accuracy]  =  softsvm(traindata,  trainlabel,
                                testdata, testlabel , sigma, C)
```

其中 C 为 soft margin SVM 的控制参数，sigma 为控制核函数的参数，当 sigma=0 时，使用线性核函数$\mathrm{K}(x_i,x_j)=x_i{}^Tx_j$，其他情况则使用 RBF 核函数$\mathrm{K}(x_i,x_j)=e^{-\frac{\|x_i-x_j\|^2}{sigma^2}}$

4．在不同数据集上使用交叉验证选择各个算法的参数(15%)

实现交叉验证（代码需要提交），在各个数据集上：

- 使用 5-fold 交叉验证为每个算法挑选适当的参数(Naïve Bayes 中的 threshold，最小二乘法中的 Lambda，SVM 中的 sigma 和 C)；
- 对每一个算法：
  - 返回一个矩阵，表示每一个参数（参数组合）在每一个 fold 上的正确率（若有 10 个参数，则返回 10x5 的矩阵）；
  - 挑选在 5 个 fold 中平均正确率最高的参数（参数组合）

在实验报告中需要记录交叉验证的结果，即对于每个参数(参数组合)在 5 个

fold 上的平均正确率。

5．实验报告(15%)

总结以上的实验结果，并对实验结果进行分析。

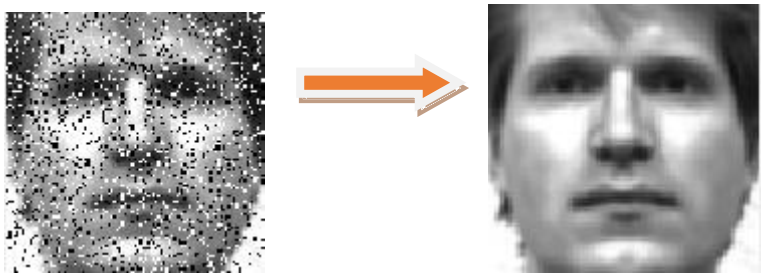
6．实验测试结果评价(10%)

对于这部分，保存每个算法在相应数据集上对应的最佳参数并提交。如对于分类算法，需要保存 Naïve Bayes 和 SVM 在相应数据集上使用 5-fold 交叉验证得到的参数(Naïve Bayes 的 threshold，Least Squares 的 lambda，SVM 的 sigma 和 C)，保存文件名统一为“数据集名”_parameters.mat。我们将会基于你们的算法代码以及最优参数，在保留下来的一部分数据上进行测试，并度量各个算法的性能。

Part 2. 图片去噪(25%)

在这部分实验中，我们以人脸图片数据为例，通过 PCA 算法对数据进行降维，保留数据中的主要信息，进一步检验 PCA 消除数据中噪音的效果。

在训练过程中通过 PCA 算法来计算投影矩阵。测试时将带有噪音的图片通过投影矩阵投影至低维空间，保留图片的主要信息，再投影至原空间完成重构，在此过程中会消除噪音的效果。



数据集介绍：

我们提供的是 YaleFace 中的人脸数据集，其中训练数据集为 60 张正常情况下的人脸图片，测试集共 6 个样本，每张均包含了一定的噪声。每张照片的大小是 50x50 的黑白图，对于照片中像素中的每一个像素，用 1 个 8 bit 数字(0-255 之间)表示其灰度值。一个 50x50 的图片，总共有 2500 个像素，因此对于每张图片，可以用一个 2500 个元素的向量表示。

在课程主页上下载 YaleFace.mat，在 Matlab 中 load 数据。有 train_data,test_data,ground_truth 三个矩阵，对应训练数据和测试数据和用于对比的无噪声数据，ground_truth 和 test_data 一一对应。其中数据都已进行归一化(每个元素在 0~1 范围)。

为了实现对于图片的去噪，我们对于训练数据用 PCA 算法计算得到投影矩阵 proj_matrix，对于测试样本 y 的重构需要先将其投影至低维空间，从而保留图像中人脸的主要信息，再对原图像进行重构。将重构得到的图像与我们提供的 ground_truth 图片作对比，得到两者之差的平方加和平均得到重构误差 recons_error。例如，A 为重构的图片，B 为 ground_truth，recons_error=$\frac{\sum_{i=1}^{50}\sum_{j=1}^{50}|A_{ij}-B_{ij}|^2}{50\times 50}$。

实验要求（注意以下有些过程可能会需要运行一定时间）

1．实现一个 PCA 降维算法（10%）

提交一个 Matlab 函数 myPCA ，函数形式为：
function[proj_matrix,recons_data,recons_error]=reconsPCA(train_data,test_data, ground_truth,threshold)；其中 threshold 表示特征值的累计贡献率。即选择前 m 个特征向量，使得

$$\frac{\text{Sum(first m}-1\text{ eigenvalues)}}{\text{Sum(all eigenvalues)}} < threshold \leq \frac{\text{Sum(first m eigenvalues)}}{\text{Sum(all eigenvalues)}}$$

2．实验验证 PCA 算法效果及实验报告（15%）

a）检验随着 threshold 不同取值，PCA 选择的降维维度以及对应的重构效果会有什么变化，重构效果可从视觉上即恢复的图片以及重构误差两方面来评价。
b）讨论为什么 PCA 能够去噪并提出改进方案。
c）在实验报告中总结以上的实验结果。
d）当 threshold=0.95 时，提交对于每个测试样本重构之后的图片，请按照测试样本的索引进行命名，例如对于第 1 个测试样本可以保存为 1.jpg。同时提交 recons_error.mat，即测试样本和 ground_truth 之间的 error，长度为 6，元素的顺序也是按照测试样本的顺序排列。

备注：

1.矢量化编程是提高算法速度的一种有效方法，其思想就是尽量使用高度优化的数值运算操作来实习学习算法。例如，假设为向量，需要计算，在 Matlab 中可以用以下方式实现：

```
z = 0;
for i = 1 : n
   z = z + x(i) * y(i);
end
```

或者可以更简单的写为：

```
z = x' * y;
```

很显然，第二段程序代码不仅简单，而且运行速度更快。

通常，一个编写 Matlab 程序的诀窍是：**代码中尽可能避免显示的 for 循环**

特别是对于核函数的计算，希望能够尽量使用矢量化编程的思想来减小计算复杂度，我们将根据算法的优化进行相应加分。

2.SVM 求解二次优化问题可以使用 Matlab 函数 quadprog，可以输入 help quadprog 查看函数使用帮助。

3.Naive Bayes 算法中的 threshold 的取值可以从[0.5 0.6 0.7 0.75 0.8 0.85 0.9]中取值；最小二乘分类器中的 lambda 可以从[1e-4 0.01 0.1 0.5 1 5 10 100 1000 5000 10000]中取值；SVM 中的参数有高斯核参数 sigma 以及 C，其中 sigma 的取值范围由数据决定：假设数据集为$\{(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_n}, y_n)\}$，令 $\mathrm{d} = \frac{\sum_{i,j}(\mathbf{x_i}-\mathbf{x_j})^2}{n^2}$，则 sigma 从[0.01d 0.1d d 10d 100d]中取值，C 可以取[1 10 100 1000]。

4. 第二部分实验中每个图片的表示是一个长为 2500 的向量，对于图片的显示需要先将向量 reshape 为一个大小为 50x50 的矩阵 image_mat，然后使用 imshow(image_mat),便可查看图片。具体函数的使用可利用 matlab 中的 help 命令查询。计算特征方程可使用 matlab 提供的 eig 函数,可以用 help eig 查询使用方法。

5.提交格式为“学号_姓名.rar”,请将两个实验的文件分别放在 part1 和 part2 文件夹中。对于 part1 中的实验,除了包含必须的.m 文件之外,还需要将 5-fold 交叉验证得到的各个函数对应正确率最好的参数保存到“数据集名”_parameter.mat 文件中同时提交，该.mat 文件中应该只有 4 个参数（分别名为 threshold, lambda, sigma, C）。part2 中需要提交当 threshold=0.95 时每个测试样本重构之后的图片以及误差向量 recons_error.mat,具体命名格式 part2 已经指出。