

**人工智能与模式识别大作业**

学号： 192050152

姓名： 朱俊杰

课程： 人工智能与模式识别

老师： 孔 万 增

**摘要**

本文使用主成分分析(PCA)方法，对MNIST 数据集中的数据进行降维，保留其主要特征，再将保留后的特征输入至支持向量机(SVM)进行训练，最后计算在MNIST 数据集中的10000个测试集上的准确率结果。

**关键词：**主成分分析；支持向量机；PCA；SVM

1. **方法介绍**

本节主要介绍实验中涉及的两个方法：主成分分析(PCA)方法和支持向量机(SVM)方法。这两个方法都是模式识别中传统、常用的方法，实验结果表明这两种方法的效果较好。主成分分析(PCA)方法主要对数据进行降维，提取其主要特征，SVM是经典的分类方法，能够有效的将数据进行分类。

## 主成分分析(PCA)

我们知道，样本点;在新空间中超平面上的投影是,若所有样本点的投影能尽可能分开，则应该使投影后样本点的方差最大化，如图1所示。

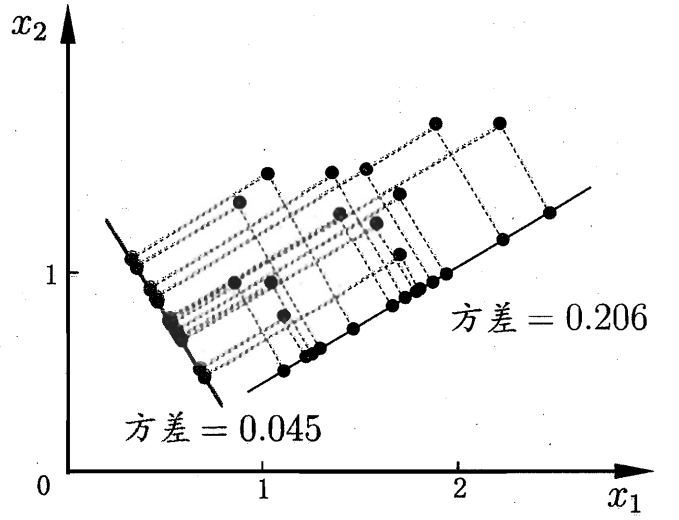


图1 使所有样本的投影尽可能分开,则需最大化投影点的方差

投影后样本点的方差是,于是优化目标可写为

对上式使用拉格朗日乘子法可得

于是，只需对协方差矩阵进行特征值分解，将求得的特征值排序:≥≥…≥，再取前个特征值对应的特征向量构成。这就是主成分分析的解。PCA算法描述如图2所示。

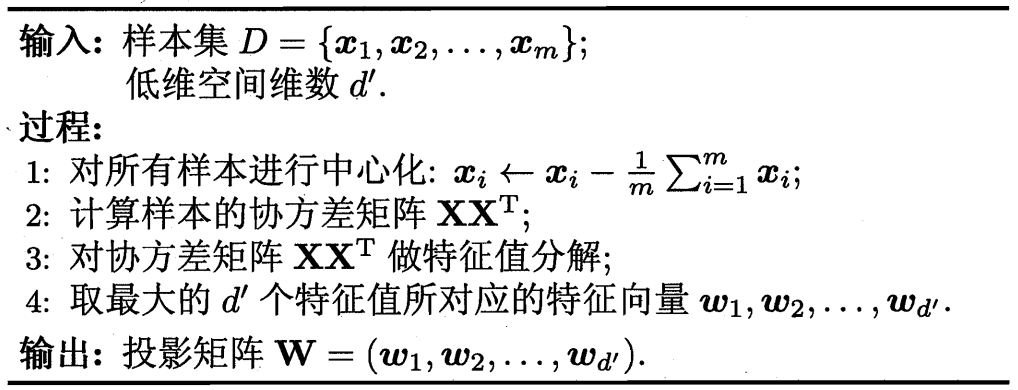


图2 PCA算法

降维后低维空间的维数通常是由用户事先指定，或通过在值不同的低维空间中对邻分类器(或其他开销较小的学习器)进行交叉验证来选取较好的值。对PCA,还可从重构的角度设置-一个重构阈值,例如= 95%,然后选取使下式成立的最小值:

PCA仅需保留与样本的均值向量即可通过简单的向量减法和矩阵-向量乘法将新样本投影至低维空间中。显然,低维空间与原始高维空间必有不同，因为对应于最小的个特征值的特征向量被舍弃了,这是降维导致的结果。但舍弃这部分信息往往是必要的:一方面，舍弃这部分信息之后能使样本的采样密度增大，这正是降维的重要动机;另一方面，当数据受到噪声影响时,最小的特征值所对应的特征向量往往与噪声有关，将它们舍弃能在一定程度，上起到去噪的效果。

## 支持向量机(SVM)

支持向量机是一类按监督学习方式对数据进行二元分类的广义线性分类器，其决策边界是对学习样本求解的最大边距超平面，也就是下面的公式

其中是超平面的法向量，是训练样本集的横坐标， 是训练样本集的个数。

一些线性不可分的问题可能是非线性可分的，即特征空间存在超曲面将正类和负类分开。使用非线性函数可以将非线性可分问题从原始的特征空间映射至更高维的希尔伯特空间，从而转化为线性可分问题，此时作为决策边界的超平面表示如下

式中为映射函数，由于映射函数具有复杂的形式，难以计算其内积，因此可使用核方法，即定义映射函数的内积为核函数以回避内积的显式计算，所以核函数直接决定了支持向量机与核方法的最终性能。

在具体的实验中，调用的是封装好的SVM函数，它会根据提供的训练集去找到合适的分类方法。

1. **特征提取**

本节介绍从MNIST数据集上提取特征，主要包含了两步：一是对数据进行预处理，二是对数据进行PCA降维操作，保留其主要特征。

## 数据预处理

原始MNIST数据集上的图像，首先将其reshape得到。将reshape后得到的图像进行标准化，在本次实验中是将每个像素值除以255，使其每个像素值的大小限定在[0,1]的范围内。

## PCA降维

1. 对样本数据进行中心化，即将样本属性减去属性的平均值；
2. 计算样本的协方差矩阵 ；
3. 对协方差矩阵做特征值分解，求得其特征值和特征向量；
4. 将特征值从大到小排序，筛选出前N个；
5. 将数据转换到新的低维空间中。
6. **分类器训练**

本文采用SVM分类器对经过PCA降维后的MNIST数据集进行训练，样本大小为60000张图像。

1. **实验结果**

本文对MNIST数据集中的10000张测试图像进行测试，同此前的60000张训练图像一样，测试图像也经过数据预处理、PCA降维后输入到SVM中进行分类，保留分类的结果。将SVM预测出来的结果与其真实类别进行比对，计算准确率。实验比较了将数据降到不同维度对分类准确率的影响以及记录了测试所需要的时间，如下表所示。为了节约测试时间，将SVM的最大迭代次数设置为了10次，这会对最后的分类准确率造成一定的影响。

表1 实验结果

|  |  |  |
| --- | --- | --- |
| PCA后维度 | SVM分类准确率(%) | 测试耗时(s) |
| 784 | 49.15 | 40.65 |
| 200 | 79.41 | 24.63 |
| 100 | 88.37 | 18.76 |
| 50 | 93.79 | 11.53 |
| 10 | 97.64 | 8.53 |
| 3 | 79.31 | 5.28 |

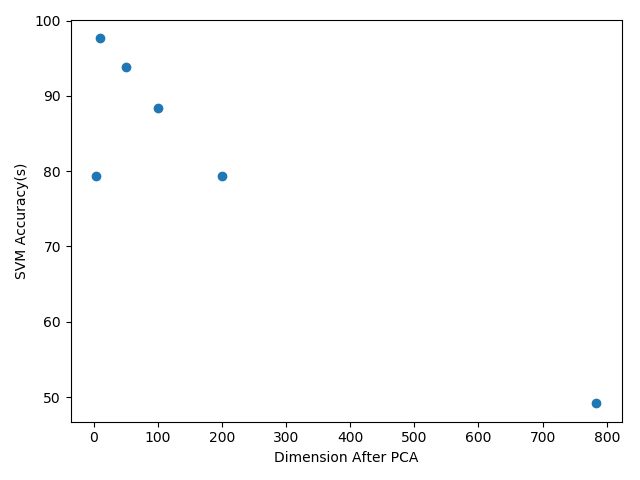


图3 SVM准确率与PCA后维度关系

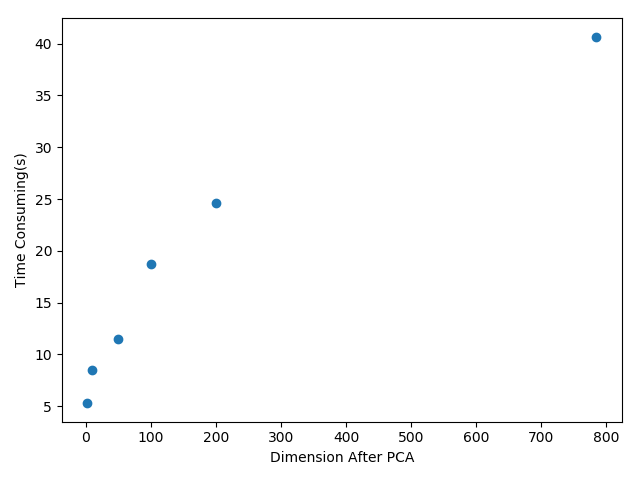


图4 测试耗时与PCA后维度关系

从上表可以看出，维度在一定的范围内，随着维度越来越低，SVM的分类准确率在逐步提升，这是因为PCA降维后，保留下来的都是主要特征，而这些主要特征正是机器在做判断是所需要的依据。但是维度并不是越低越好，在达到一定的维度后，会发现其分类的准确率会减少，这是因为其保留的特征少了，这会干扰到SVM最后对它的分类判断。从测试耗时的角度看，测试的时间会随着维度的降低而减少，这是由于其保留的特征减少了，检测中判断的所需要的特征减少，所以消耗的时间也在减少。

1. **实验核心代码（Python）**

**def** pca(dataMat, topNfeat=10):

*# 1.对所有样本进行中心化（所有样本属性减去属性的平均值）*

meanVals = np.mean(dataMat, axis=0)

meanRemoved = dataMat - meanVals

*# 2.计算样本的协方差矩阵 XXT*

covmat = np.cov(meanRemoved, rowvar=0)

*#print(covmat)*

*# 3.对协方差矩阵做特征值分解，求得其特征值和特征向量，并将特征值从大到小排序，筛选出前topNfeat个*

eigVals, eigVects = np.linalg.eig(np.mat(covmat))

eigValInd = np.argsort(eigVals)

*# 取前topNfeat大的特征值的索引*

eigValInd = eigValInd[:-(topNfeat+1):-1]

*# 取前topNfeat大的特征值所对应的特征向量*

redEigVects = eigVects[:, eigValInd]

*# 4.将数据转换到新的低维空间中*

*# 降维之后的数据*

lowDDataMat = meanRemoved \* redEigVects

*# 重构数据，可在原数据维度下进行对比查看*

reconMat = (lowDDataMat \* redEigVects.T) + meanVals

**return** np.array(lowDDataMat), np.array(reconMat)

from sklearn import svm

clf =svm.SVC(max\_iter=10)

clf.fit(x\_train, y\_train)

prediction=clf.predict(x\_test)

accurancy = np.sum(np.equal(prediction, y\_test)) / 10000

print('accurancy : ', accurancy)

参考文献

1. 周志华《机器学习》