****

**2018—2019学年第2学期**

**《大数据技术及应用》课程报告**

题目：基于KNN算法的Wine Data分类

学 号 1607040216

专 业 物联网工程

班 级 1602班

姓 名 吕 涵

|  |  |
| --- | --- |
| 得 分 |  |
| 阅卷人 |  |

2019年 6 月 1 日

# 说明

1. 《大数据技术及应用》课程，旨在对学生进行大数据处理能力的综合训练，以提高学生分析问题、建立模型、解决问题能力。所有参加本课程的学生都必须参与算法设计实现、课程报告撰写工作。
2. 《大数据技术及应用》结束后，需要提交**源数据、程序源代码、报告电子版、报告纸质版，压缩后以“学号+班级+姓名”进行命名，材料经班长汇总后交给任课老师**。逾期未提交相关资料者不得参加成绩评定。
3. 指导教师需对学生实践报告填写评语并在相应位置填写成绩并签字。
4. 课程报告要求严格按照本报告模板撰写，条理清晰、内容详尽、论述准确、书写认真。成绩由报告格式设计、内容设计、算法设计、结果描述等几部分组成。
5. 报告格式要求如下：表格、图像进行编号，正文要求宋体、小四，行前缩进2字符，间距1.5倍行距。其他未尽事宜请参考本科毕业设计报告撰写格式说明，纸质报告要求双面打印。
6. 报告可根据内容适当增加页面，但不宜长篇大论，所写内容应紧扣主题。

|  |
| --- |
| **题目 基于KNN算法的Wine Data分类** |
| 一、研究意义 |
| K最近邻(KNN，K-NearestNeighbor)分类算法是[数据挖掘](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E6%8C%96%E6%8E%98/216477" \t "/Users/Lv/Documents\\x/_blank)分类技术中最简单的方法之一。KNN算法的核心思想是如果一个样本在特征空间中的k个最相邻的样本中的大多数属于某一个类别，则该样本也属于这个类别，并具有这个类别上样本的特性。该方法在确定分类决策上只依据最邻近的一个或者几个样本的类别来决定待分样本所属的类别。  KNN方法在原理上依赖于极限定理，但在类别决策时，只与极少量的相邻样本有关。由于KNN方法主要靠周围有限的邻近的样本，而不是靠判别类域的方法来确定所属类别的，因此KNN方法较其他方法更为适合于类域的交叉或重叠较多的待分样本集。  在化学工程、制药和食品分析领域，KNN算法可根据研究人员已知的样本特性与类别对未知样本进行分类，KNN算法在多次工程学的分类操作上都表现出了极高的可靠性与准确性。同时，随着其他技术的不断更新和完善，KNN算法改进算法也应运而生，其改进方向主要分成了分类效率和分类效果两方面。  在本项目中我们针对数据集中的红酒特性进行KNN分类的训练和测试，研究人员事先根据实验中测量到每一份样品的不同属性值综合判断该样本来源于哪一品种，随后我们根据另一部分样本的属性值预测其品种，最后与实验人员的真实测定值比对，评价本算法的准确性。  在数据集中，实验人员对意大利同一地区生产的葡萄酒进行化学分析提取了178份红酒样本的若干个属性来分析，并将其划分为三个不同的品种。同时实验人员确定了三种葡萄酒中，共有13中成分的数量共同影响其品类，这对当地的制酒工业有较为重要的指导意义。  然而，这类工作在工程实践中需投入大量时间和人力成本，并且实际操作时难以保证准确性。但是机器学习的方法可以极大地弥补人力的限制和不足，我们以此样本为例，探讨KNN分类算法在化学工程成分分析领域中的应用可行性与实际价值。 |
| 二、数据描述 |
| 数据来源于UCI Machine Learning Repository 中较为经典的热门数据集Wine Data Set。UCI机器学习库是机器学习社区用于机器学习算法的经验分析的数据库，领域理论和数据生成器的集合，其中数据集可保证真实性与准确性。  Wine Data Set用于红酒的化学成分分析与分类，这些数据包括了三种酒中13种不同成分的数量。13种成分分别为：Alcohol，Malicacid，Ash，Alcalinity of ash，Magnesium，Total phenols，Flavanoids，Nonflavanoid phenols，Proanthocyanins，Color intensity，Hue，OD280/OD315 of diluted wines，Proline。  此外，每一份样本还标记了已知的分类，分别记为“1”，“2”，“3”其中第1类有59个样本，第2类有71个样本，第3类有48个样本。数据源文件中的每个样本的数据都是完整的，没有空缺值与脏数据，可直接读取csv文件进行操作，因此可省略对源数据的清理工作。  前期处理包括数据的读取与预处理，我们使用pandas库读入csv文件并转化为Dataframe对象，直接完成数据的读取。然后随机打乱数据，再按照7：3比例划分样本数据与测试数据。  统计数据得出样本数据共124条，测试数据共54条，之后再分别将样本数据与测试数据存储为本地文件准备下一步训练和测试工作。  C5E311BA-AEB2-42C6-A457-8AF8CA3CF6F0图2.1 数据的读取与预处理 |
| 三、模型描述 |
| 数据预处理完成后，我们得到两个数据集——训练集和与测试集和。本模型的首要功能即将测试集和中每一条数据放入训练集和中进行学习，判断测试数据的分类。第二项功能是在全部学习结束后，将测试分类与实际分类进行对比，评估在某一K值下该模型的分类准确度。最后，我们探讨不同K值对于模型的分类准确度的影响。   1. KNN分类 2. 计算距离：给定测试对象，计算它与训练集中的每个对象的距离；   50F57AED-A313-4699-AB0C-0E9139105BCA我们采用测试对象与所有训练对象属性之间欧几里得距离进行计算，距离公式为：  然而，在数据采集时，我们需考虑到这样一个问题，不同属性之间的计量单位不相同，有可能会造成不同属性之间权重不同，影响分类结果。因此我们在计算之前，要首先进行数据的标准化。  数据的标准化是将数据按比例缩放，使之落入一个小的特定区间。在某些比较和评价的指标处理中经常会用到，去除数据的单位限制，将其转化为无量纲的纯数值，便于不同单位或量级的指标能够进行比较和加权。  其中最典型的就是数据的归一化处理（图3.1），即将数据统一映射到[0,1]区间上。归一化公式为：  2866A8C1-C9E0-44F0-8251-410CBDEDBF96  图3.1 数据的归一化处理   1. 邻居发现：排序确定距离最近的K个训练对象，作为测试对象的近邻；   邻居发现阶段首先对上步归一化结果得出的距离进行降序排序。  94AF3B09-6A0F-43DB-AAEE-34108A456757    94AF3B09-6A0F-43DB-AAEE-34108A456757之后我们根据确定好的K值，选取距离测试对象最近的K个训练样本。  94AF3B09-6A0F-43DB-AAEE-34108A456757最后我们根据索引值返回K紧邻表，用以第3步的学习分类。   1. 学习分类：根据这K个近邻归属的主要类别，对测试对象进行分类。   9A0EE263-4774-46CF-9D0C-7F7975894BA1首先根据上步得到的K近邻列表，统计其中权重最大的类别。这里我们假设所有点的权重都相同，即统计各类别数量之和。  9A0EE263-4774-46CF-9D0C-7F7975894BA1对K个近邻的3种分类数量统计完成后，我们对统计结果进行降序排列，选择最大数量的类别代表测试样本最大概率的类别归属。  对54条测试数据中每一条数据进行学习分类后，统计测试样本分类表并存储至本地。   1. 准确率评估   wpsoffice上步中我们得到了统计测试样本的分类表，在原始数据中这些测试样本已经有了实际分类，我们利用如下统计学公式计算测试样本的正确率来对该模型进行评估。  （三）K值对于模型的分类准确度的影响  在实际操作中我们观察到，当K取不同值时，部分测试数据的分类结果不相同。即对距离最近的不同数目邻居进行分类统计时，统计结果的差异会影响分类结果。我们在项目中探究了K取1~20不同值时，对模型准确率的影响。  在前两步的基础上，我们采用一个循环结构对每一个K值进行学习分类，保留其结果进行分析。  E5C6C53D-C847-45DD-B08B-1ED980339B2A  FF189A9D-261B-4BC2-B6EC-95B3CF00FB8B程序流程图：  图3.2 模型程序流程图 |
| 算法实现 |
| 按照程序流程顺序，我们描述算法实现代码，并且对核心代码进行分析。   1. #Dataframe格式转换np.array  原始数据以csv格式进行存储，我们使用pandas库对数据进行读入得到一个   Dataframe类型对象，为了满足下一步的矩阵运算，我们先将其转换成np.array的矩阵类型数据。  def df2array(train):  train = train.drop('Class', axis=1)#删去分类信息，只处理数据  train = np.array(train)#格式转换为up.array  Mat = np.delete(train, [0, 1], axis=1)#删去index  return Mat   1. #统计K近邻列表   对于测试数据进行归一化后计算欧氏距离，选取距离最近的前K个值并按照索引值建立K近邻列表。  def distance(DataMat,inArr,train\_class,K):  normMat, ranges, minVals = autoNorm(DataMat)#样本数据归一化  Mat = (sqrt(((inArr - minVals) / ranges - normMat) \*\* 2))#计算欧氏距离  result = Mat.sum(axis=1)#降序排列提取索引值  index= result.argsort()#选取距离最近的前K个值  index= list(index[:K]) #按照索引值建立K近邻列表  neighbour=[]  for i in index:  neighbour.append(train\_class[int(i)])  return neighbour#返回K近邻列表   1. #KNN分类测试与模型评估   DataMat为样本数据,TestMat为测试数据，将TestMat置入DataMat中学习分类，所有样本分类结束后根据统计学公式计算分类准确率  def ClassTest(DataMat,TestMat,test,train\_class,K):  test\_class = []#建立分类表  for inArr in TestMat:#测试数据依次分类  neighbour = distance(DataMat, inArr,train\_class,K)  target = classify(neighbour)  test\_class.append(target)   test['test\_Class'] = test\_class  test.to\_csv('test\_classify.csv')#分类结果写入测试数据文件  #对比实际分类，计算KNN分类准确率  true\_class=test['Class']  Count=0  for item in range(len(true\_class)):  if true\_class[item]==test\_class[item]:  Count+=1  print("分类正确:"+str(Count)+"条")  print("总共分类:"+str(len(true\_class))+"条")  print("分类准确率:"+str(round(Count\*100/len(true\_class),2))+"%")  return Count/len(true\_class)   1. 主函数   if \_\_name\_\_ == '\_\_main\_\_':  #读取训练样本数据与测试数据  train=pd.read\_csv('train.csv')  test=pd.read\_csv('test.csv')  train\_class=list(train['Class'])#提取训练样本分类列表  #训练样本数据与测试数据转化为np.array  DataMat=df2array(train)  TestMat=df2array(test)  ClassTest(DataMat,TestMat,test,train\_class,K=13)#选取K值，执行分类测试   1. K值分析（附加内容）   K值对于模型的分类准确度的影响，使用matplotlib作图分析在1~20范围内，K取不同值时的分类准确度。  plt.stem(x,rate,linefmt=':')#做取样图 x\_ticks = np.linspace(1,20,20)#限定x轴刻度 y\_ticks = np.linspace(0.9,1,11)#限定y轴刻度 plt.xticks(x\_ticks) plt.yticks(y\_ticks) plt.xlim(0,21,1)#限定x轴范围 plt.ylim(0.9,1,0.01)#限定x轴范围 plt.title('K值与分类准确率关系',fontproperties=font,size=15)#写入标题 plt.xlabel('K值', fontproperties=font,size=12)#写入x轴标签名称 plt.ylabel('分类准确率', fontproperties=font,size=12)#写入y轴标签名称 y=[] for xy in zip(x, rate):  plt.annotate(s="(%s)" % round(xy[1],2), xy=xy, xytext=(-15, 10), textcoords='offset points')#作数据描述 plt.show()#图像显示 |
| 五、运行结果及意义说明 |
| （一）K近邻表输出  803BCD22-1947-4333-9CA9-751BFB1C1AB71BD5545F-E550-495F-B2C2-7CE603D5450F图5.1 K近邻表输出(部分)  我们输出K=13时的测试数据的K近邻表，我们可以看到解释器输出了54\*13的矩阵。54行对应着54条测试数据，13对应着每一条的测试数据的前13个近邻。   1. 统计近邻列表中分类个数   A2B682BA-A9B9-4781-B5E9-D807C75A24DA675DC8C7-596D-4303-BB1C-92D4C0C08C88  图5.2 K近邻表列表分类统计(部分)  K近邻表列表分类统计结果输出54个词典类型数据，每个字典对应着每一条测试数据的K近邻表列表分类统计情况。  以图5.2中红圈标记的字典{1: 2, 2: 4, 3: 7}为例进行说明，该字典说明了此条测试数据的K近邻表列表中，有2个类别一邻居、4个类别2邻居和7个类别3邻居。  我们分别对每个测试样本的K近邻表列表进行排序，找出K个邻居中出现最多次数的类别，即可依此来指代测试样本的类别。   1. KNN模型分类结果  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Named | Class | test\_Class | Named | Class | test\_Class | | 125 | 2 | 2 | 80 | 2 | 2 | | 37 | 1 | 1 | 97 | 2 | 2 | | 90 | 2 | 2 | 13 | 1 | 1 | | 136 | 3 | 3 | 104 | 2 | 2 | | 28 | 1 | 1 | 173 | 3 | 3 | | 62 | 2 | 2 | 147 | 3 | 3 | | 74 | 2 | 2 | 29 | 1 | 1 | | 81 | 2 | 1 | 75 | 2 | 2 | | 43 | 1 | 1 | 168 | 3 | 3 | | 101 | 2 | 2 | 53 | 1 | 1 | | 160 | 3 | 3 | 105 | 2 | 2 | | 103 | 2 | 2 | 40 | 1 | 1 |   图5.3 KNN模型分类结果(部分)  所用测试数据分类完成之后存入csv文档中便于后期的分析，我们节选部分分类结果进行说明。列表中Named表示数据序号，Class为样本真实分类，test\_Class为该算法执行的分类结果。  观察图5.3可得出，在以上数据中绝大多数的分类都准确命中，只有极少数的测试数据（例如81号数据）分类出现偏离。出现偏移的影响因素有多种，这些因素直接决定了模型的准确度。   1. K取值对模型准确度的影响探究   Figure_1 在对模型准确度的众多影响因素中，K值的选取是最为关键的一个，因此我们尝试在此模型的基础上在1~20之间调整K值的数值，观察不同K值下模型的准确度。  图5.4 K值与分类准确率关系  分析图5.4我们可以得知，K值在1~20取值范围内，模型准确率一直在动态变化，并无明显线性关系。  当K取12、20时模型准确率最低，只有93%；当K取3、6时准确率最高，为98%。其他取值条件下，模型准确率基本分布于94%与96%上。  在使用该模型进行分类时，准确度应当是最优先考虑的因素，但同时也必须考虑时间和内存成本。例如为取得较高准确率而采用较大K值时，K近邻表的维数就会相应扩大，因此对表的建立与操作会占用较多的时间与内存，影响模型的学习效率。  因此，我们应当根据分类任务应用场景对模型准确度做不同的要求，尽可能在模型算法的执行上平衡准确率与时间、内存成本，以达到效率最大化。 |
| 六、总结 |
| 本次实验针对机器学习经典数据集Wine Data Set构建了一个KNN算法分类模型，并且使用测试结果对该模型进行准确性评价。实验进行过程中，我们也探讨了K取值对模型准确度的影响，整个项目取得了较为满意的结果。  首先我们巩固了课程中学习到的KNN算法，明确了该算法的六项步骤：  收集数据、数据预处理、设计算法分析数据、测试算法、使用算法和评价算法。KNN又包含几项核心算法，如数据归一化、K近邻表的构建和近邻列表中的分类统计等。  本项目完全采用Python3编译，程序中涉及多种数据类型之间的转换和对数据对象的操作，如Dataframe和np.array的转化，dict类型与list类型的转化，dict类型数据的value排序和key索引。本项目的算法代码设计工作，使我学习到了许多关键的Python3语法和数据结构知识，也提高了我们对这门高效简洁的脚本语言——Python的掌握能力。  本项目的整体工作涉数据挖掘、数据处理、数据分析和数据可视化技术，各种技术在Python语言的统一下灵活结合，及其高效的完成预定工作。在项目的20多次样本测试工作中，平均每次耗时0.03秒，样本平均准确率95.18%，符合项目预期。  本次实践项目使我对大数据处理和机器学习领域有了新的理论和实践认识，为我以后的学习和工作积累了宝贵经验，也对我今后的职业素养提升指导了方向。  学生签字： |
| **指导教师评语**  指导教师签字：  年 月 日 |